

Traian Anghel

**DICȚIONAR
DE
INFORMATICĂ**

Redactare: Marieva Ionescu
Tehnoredactare: Cristina Gvinda

Date despre autor:

Traian Anghel, profesor gr. I la Grupul Școlar Industrial „Anghel Saligny” din Brăila, absolvent al Facultății de Fizică din cadrul Universității București și al unui curs postuniversitar de specializare în informatică la Universitatea „Dunărea de Jos” din Galați. A publicat lucrări în care este reflectat interesul său pentru programarea Web (folosind AJAX, JavaScript, PHP, MySQL), serviciile Web, interacțiunea om-calculator, Web-ul semantic (metadate, microformate, ontologii) și utilizarea în sistemul educațional a instrumentelor și a resurselor Web.

Autorul vă invită să vizitați și blogul acestei cărți, la adresa <http://dictionarinfo.blogspot.com/>

Editura CORINT

Redacția și administrația:

Str. Mihai Eminescu nr. 54 A, sector 1, București
Tel./Fax: 021.319.47.97, 021.319.47.99

Difuzarea:

Calea Plevnei nr. 145, sector 6, cod poștal 060012, București
Tel.: 021.319.88.22; 021.319.88.33; 021.319.88.55; 021.319.88.77
Fax: 021.319.88.66

E-mail: vanzari@edituracorint.ro; Magazin virtual: www.edituracorint.ro

Descrierea CIP a Bibliotecii Naționale a României
ANGHEL, TRAIAN

Dicționar de informatică / Traian Anghel. - București:
Corint, 2010

ISBN 978-973-135-547-4

81'374.2:004=135.1

ISBN: 978-973-135-547-4

Toate drepturile asupra acestei lucrări sunt rezervate Editurii CORINT,
parte componentă a GRUPULUI EDITORIAL CORINT.

Format: 16/61x86

Coli tipo: 28

Tiparul executat la: Imprimeria OLTENIA

Traian Anghel

**DICTIONAR
,
DE
INFORMATICĂ**

CORINT

Cuvânt-înainte

Introducere

Învățarea bazată pe mediile electronice de comunicare și de furnizare a materialelor de învățare, cunoscută ca *e-Learning*, este practică astăzi la scară din ce în ce mai mare. Cei mai mulți dintre elevi și studenți sunt familiarizați cu folosirea calculatorului personal, precum și cu tehnologiile moderne care oferă posibilitatea înregistrării imaginilor, mișcărilor și sunetelor, având deprinderea de a utiliza aparate foto și camere de filmat digitale, telefoane și alte dispozitive mobile cu funcții avansate. Adesea, aceștia își partajează proiectele prin intermediul unor instrumente colaborative *online* furnizate de noul Web, așa cum sunt serviciile pentru partajarea conținutului (*YouTube, Flickr*), rețelele sociale (*Hi5, Facebook, MySpace*), blogurile, *wiki*-urile etc. Abilitatea de utilizare a TIC (Tehnologia Informației și a Comunicațiilor) este văzută – în cadrul strategiei Lisabona¹ – ca o nouă alfabetizare, care, împreună cu formele tradiționale de educație, le permite tuturor cetățenilor să participe la „societatea cunoașterii”. De aceea, cunoașterea termenilor de bază ai informaticii a devenit obligatorie pentru toate persoanele implicate atât în domeniul educațional (elevi, studenți și profesori, indiferent de disciplina predată), cât și în celelalte domenii ale sistemului social global.

Despre informatică

Informatica este un complex de discipline având ca scop procesarea informației prin intermediul sistemelor automate, în special cu ajutorul calculatorului. Acesta reprezintă dispozitivul prin care sunt implementate conceptele teoretice ale informaticii, crearea și dezvoltarea sa bazându-se pe câteva discipline aplicative (fizică, electronică, telecomunicații etc). Conform lui Edsger Dijkstra², „în informatică ai de-a face cu calculatorul la fel cum în astronomie ai de-a face cu telescopul”. Astfel, informatica este un sistem de gândire formal (ca și matematica), realizabil prin intermediul calculatorului. Așa cum afirmă Grigore Moisil³, „Informatica restabilește nu numai unitatea matematicilor pure și a celor aplicate, a tehnicii concrete și a matematicilor abstracte, dar și pe cea a științelor naturii, ale omului și ale

¹ Cunoscută și ca Agenda Lisabona sau Procesul Lisabona, reprezintă un set de obiective, domenii prioritare de acțiune, ținte și măsuri pentru orientarea politicilor de creștere economică și ocupare a forței de muncă, având ca scop realizarea obiectivului strategic al Uniunii Europene de a deveni cea mai competitivă și dinamică economie bazată pe cunoaștere. Strategia a fost adoptată la Consiliul European extraordinar de la Lisabona din 23–24 martie 2000 și reînnoită de Consiliul European de la Bruxelles din 22–23 martie 2005.

² Edsger Dijkstra (1930–2002), informatician olandez, cunoscut în special pentru algoritmul drumului minim într-un graf, algoritm care îi poartă numele.

³ Grigore C. Moisil (1906–1973), academișian, profesor la Facultatea de Matematică a Universității din București, fondatorul școlii de logică și informatică.

societății. Reabilitează conceptele de abstract și de formal și împacă arta cu știința, nu numai în sufletul omului de știință, unde erau întotdeauna împăcate, ci și în filosofarea lor”.

Iată câteva dintre disciplinele incluse în cadrul informaticii: algoritmi și structuri de date, limbaje de programare, arhitectura calculatoarelor, sisteme de operare, inginerie software, calcule numerice și simbolice, sisteme de gestiune a bazelor de date, rețele de calculatoare, securitatea informației, criptografie, teoria jocurilor, inteligența artificială.

Elementele implicate în procesarea informației alcătuiesc un sistem informatic. Acesta include calculatoare, diverse elemente hardware utilizate în transmiterea datelor, datele prelucrate și software-ul folosit.

Despre carte

Acest dicționar le oferă cititorilor posibilitatea de a avea acces, folosind o singură sursă, la definițiile unui număr mare de termeni utilizați în informatică și în studiul sistemelor informatice. Definiind circa două mii de termeni, lucrarea clarifică într-o manieră simplă și accesibilă sensul unor cuvinte și expresii incluse în vocabularul de bază al informaticii. Dicționarul a fost elaborat astfel încât să poată fi consultat de utilizatori cu diverse niveluri de pregătire în domeniul calculatoarelor, de la începători la specialiști. Această abordare ține seama de faptul că tehnica de calcul este utilizată de un număr din ce în ce mai mare de persoane, în diverse scopuri.

Un număr important de termeni se referă la utilizarea calculatoarelor în activitatea de predare-învățare-evaluare în educație (*e-Learning*). Aceștia sunt definiți *in extenso*, oferindu-le educatorilor un util material didactic. De asemenea, sunt prezentate pe scurt biografiile unor personalități științifice (programatori, matematicieni, ingineri) care au adus contribuții decisive la dezvoltarea informaticii și a sistemelor informatice. În sfârșit, lucrarea include prezentări ale unor aplicații populare, atât proprietare, cât și *open source*, precum și ale unor site-uri care oferă diverse resurse, software și servicii folosite adesea și în educație.

Ținând seama că majoritatea termenilor utilizați în informatică provin din limba engleză, fiecare termen definit este însoțit de echivalentul său englezesc, inclus între paranteze. Totuși, atunci când varianta englezească a unui termen a pătruns ca neologism în limba română sau când este preferată celei românești (în manuale, lucrări științifice, dar și în limbajul comun), termenul este inclus în limba engleză (de exemplu *heap*), iar între paranteze este conținut – atunci când există – echivalentul său românesc („grămadă”).

Destinatarii cărții

Cartea de față se adresează în special educatorilor (cadre didactice) și educabililor (elevi și studenți), dar poate fi consultată, în general, de toate persoanele care

utilizează calculatorul. Dicționarul le poate fi util celor care activează în domeniul educațional, atât în învățământul preuniversitar (gimnazial și liceal, la toate filierele, profilurile și specializările), cât și în cel universitar. În particular, lucrarea poate fi folosită de către elevii claselor cu specializarea matematică-informatică (inclusiv intensiv informatică), profilul real, filiera teoretică.

Convenții utilizate

Pentru a facilita parcurgerea dicționarului, au fost utilizate următoarele convenții de reprezentare a conținutului acestuia:

- extensiile de fișiere, URL-urile, precum și numele aplicațiilor, organizațiilor, consorțiilor, grupurilor, instituțiilor și conferințelor au fost scrise utilizând stilul *italic*;
- au fost utilizate acronimele *e.g.* pentru *exempli gratia* („de exemplu”, în limba latină) și *i.e.* pentru *id est* („cu alte cuvinte”, „adică”, în limba latină).

Traian Anghel

Dicționarul

Numere și simboluri

& (*ampersand*). Caracter utilizat în sirurile de interogare incluse în URL-uri pentru separarea perechilor de forma *nume_data = valoare_data*.

***** (*asterisk, asterisc*). Numit și caracter *wildcard*, este folosit în căutările realizate pe calculatorul propriu al utilizatorului. De asemenea, poate fi utilizat într-o manieră similară și în căutările realizate pe Web. Caracterul respectiv trebuie folosit în expresii de căutare incluse între ghilimele. Vor fi returnate toate paginile care conțin în locul caracterului *wildcard* orice alt cuvânt sau grup de cuvinte. De exemplu, căutarea în care se folosește expresia „*open * access*” va returna și pagini care conțin expresia „*open and unrestricted access*”.

// (*dublu slash*). Împreună cu caracterul „:” (două puncte), este folosit în URL pentru separarea protocolului de numele mașinii-gazdă. A se vedea și : și **URL**.

: (*două puncte*). Simbol utilizat în URL după numele protocolului, împreună cu **//** (*dublu slash*), pentru separarea acestuia de numele mașinii-gazdă. A se vedea și **//** și **URL**.

<> (*paranteze unghiulare*). **1.** Pereche de simboluri utilizate pentru definirea marcajelor în limbajele de marcare (e.g., XML, HTML). **2.** Pereche de simboluri utilizate pentru evidențierea adresei în antetul mesajelor de *e-mail*. A se vedea și **mesaj de e-mail**.

@ (*a rond*). Citit și *at*, este utilizat în adresele de *e-mail* pentru a separa numele contului de numele domeniului. A se vedea și **adresă de e-mail**.

1educat. Portal specializat în prezentarea ofertei de cursuri a mai multor furnizori, conținând informații despre planificarea carierei și mediul educațional (<http://www.1educat.ro/>). De asemenea, le oferă utilizatorilor un asistent *online*.

100BaseT. Numit și *FastEthernet*, este un standard Ethernet pentru rețelele locale de calculatoare care folosesc cablu bifilar torsadat și lucrează la rate de transfer de 100 megabiți pe secundă. A se vedea și **Ethernet** și **Mbps**.

1GL (*first-generation language*). A se vedea **limbaj de programare la nivel mașină**.

200. Cod de stare HTTP generat de serverul Web, având semnificația de O.K. și indicând faptul că resursa solicitată de client a fost furnizată, cererea acestuia fiind astfel satisfăcută. A se vedea și **cod de stare HTTP**.

2GL (*second-generation language*). A se vedea **limbaj de asamblare**.

301. Cod de stare HTTP generat de serverul Web, semnificând redirectionarea permanentă (*Moved Permanently*) a cererii către o altă locație (în care a fost mutată resursa cerută). A se vedea și **cod de stare HTTP**.

3GL (*third-generation language*). A se vedea **limbaj de nivel înalt**.

400. Cod de stare HTTP generat de serverul Web, indicând o cerere eronată (*Bad Request*) a clientului, care nu poate fi satisfăcută deoarece conține o eroare de sintaxă. A se vedea și **cod de stare HTTP**.

401. Cod de stare HTTP generat de serverul Web, indicând o cerere neautorizată (*Unauthorized*), care nu poate fi satisfăcută deoarece implică existența unui antet *Authorization*, iar acesta nu a fost furnizat odată cu cererea. A se vedea și **cod de stare HTTP**.

402. Cod de stare HTTP generat de serverul Web, indicând o cerere cu plată (*Payment Required*) care nu poate fi

satisfăcută deoarece nu a fost furnizat antetul *ChangeTo* (plătit pentru). A se vedea și **cod de stare HTTP**.

403. Cod de stare HTTP generat de serverul Web atunci când cererea clientului nu poate fi satisfăcută deoarece accesul este interzis (*Forbidden*). A se vedea și **cod de stare HTTP**.

404. Cod de stare generat de serverul Web, având semnificația *Not Found* (lipsă), atunci când resursa solicitată nu a fost găsită. A se vedea și **cod de stare HTTP**.

4GL (*fourth-generation language*). A se vedea **limbaj din generația a patra**.

A

AboutUs.org. Director Web de tip wiki, fondat în 2006 de Ray King, în Portland, SUA. Ca oricare aplicație de tip wiki, le oferă utilizatorilor posibilitatea de a adăuga intrări sau de a modifica informațiile existente (<http://www.aboutus.org/>). A se vedea și **director Web și wiki**.

AcademiaOnline. Sistem de *e-Learning* dezvoltat și implementat de compania InsideMedia, care pune la dispoziție cursuri *online* pentru afaceri, Tehnologia Informației și Comunicațiilor și limbi străine (<http://www.academiaonline.ro/>). A se vedea și **e-Learning**.

accelerated learning (*învățare accelerată*). Teorie elaborată de Colin Rose, conform căreia stilurile de învățare predominante în învățarea tradițională sunt: vizual, auditiv și tactil/kinestezic. Oamenii pot învăța citind (vizual), ascultând (auditiv), văzând (vizual), vorbind (auditiv) sau efectuând o activitate (tactil/kinestezic). Primele trei moduri de învățare enumerate sunt pasive, în vreme ce ultimele două sunt active.

Modurile active favorizează o rată înaltă a retenției (*i.e.*, reținerii informațiilor, deprinderilor și competențelor). Oamenii își amintesc foarte bine ceea ce fac/practică în timp real. De asemenea, o combinație între activitatea practică și vorbire (despre ceea ce învață) determină un grad mare de retenție. În sfârșit, vorbirea (despre ceea ce învață) favorizează retenția. Modurile pasive se află mai jos decât cele active pe scara retenției. Combinația dintre ascultare și văz produce o rată a retenției mai mică decât cea determinată de vorbire, fiind urmată de ascultare, văz și citit. A se vedea și **Memletic**.

accelerator (*accelerator*). Tastă (sau combinație de taste) prestabilită, utilizată (utilizate) într-o aplicație, în scopul de a mări viteza cu care se realizează o acțiune (*e.g.*, selectarea unui cuvânt). Mai poartă și numele de *tastă de scurtătură* (*shortcut key*).

Accept. Unul dintre anteturile cererii HTTP. Permite clientului să precizeze – prin intermediul unei negocieri conduse de server – tipurile MIME pe care le poate recunoaște și care pot fi incluse în răspunsul serverului (*e.g.*, `Accept: image/gif, image/jpeg`). A se vedea și **cerere HTTP și MIME**.

Accept-Language. Unul dintre anteturile cererii HTTP. Permite specificarea unei limbi în care este scris documentul solicitat, în situația în care pe server există versiuni ale acestuia în diverse limbi. A se vedea și **cerere HTTP**.

acces de la distanță (*remote access*). Utilizarea unui dispozitiv (e.g., calculator) de la distanță.

acces deschis (OA, *Open Access*). Reprezintă accesul *online* liber, imediat și permanent la textul complet al materialelor științifice și educaționale publicate în Internet. Accesul deschis presupune că oricare utilizator care are acces la Internet, indiferent de locul unde se află, poate citi, descărca, stoca, printa, utiliza conținutul digital al articolelor. În mod obișnuit, un articol OA are copyright și restricții de licențiere limitate. Ceea ce face posibil accesul deschis este Web-ul și acordul autorului sau al titularului dreptului de autor.

Cele mai importante momente din istoria mișcării *Open Access* sunt: conferința de la Budapesta (februarie 2002), care a adoptat *Inițiativa cu privire la Accesul Deschis* (BOAI, *Budapest Open Access Initiative*); conferința de la Berlin (20–22 octombrie 2003), care a adoptat *Declarația cu privire la Accesul Deschis în Științe exacte și Științe umaniste* (*Berlin Declaration on Open Access in the Sciences and Humanities*); lansarea spre semnare în ianuarie 2007 a petiției pentru garantarea accesului public la rezultatele cercetărilor finanțate din fonduri publice.

Cele mai importante documente adoptate pe plan internațional referitoare la accesul deschis sunt: declarația BOAI (*Budapest Open Access Initiative*), care furnizează

definiția accesului deschis și fundamentează filosofia acestei mișcări (14 februarie 2002); *Bethesda Statement on Open Access Publishing* (20 iunie 2003); *Berlin Declaration on Open Access to Knowledge in the Sciences and the Humanities* (22 octombrie 2003).

accesibilitate (*accessibility*). Proprietatea unui sistem (software sau hardware) de a fi utilizabil de către persoanele cu anumite deficiențe fizice (e.g., imobilitate parțială, deficiențe de auz, de vedere). A se vedea și **tehnologie cu accesibilitate**.

achiziție (*acquisition*). Termen folosit în *marketing*-ul pe Internet pentru a denumi momentul în care un vizitator al site-ului devine client sau cumpărător. În mod obișnuit, este momentul în care acesta realizează una dintre următoarele trei acțiuni: cumpără un produs, furnizează date de contact și este interesat de un produs sau se înscrie pentru a primi *newsletter*-ul oferit de compania în proprietatea căreia se află site-ul.

achiziție de date (*data acquisition*). Extragerea datelor dintr-o sursă externă unui calculator (aflată în lumea reală) cu scopul de a genera informații care pot fi manipulate (*i.e.*, analizate, prelucrate, stocate și afișate) de calculator.

ACM (*Association for Computing Machinery – Asociația pentru mașini de calcul*). Fondată de Richard Hamming, este prima societate științifică și

educațională din domeniul științei calculatoarelor. A fost înființată în 1947 la New York, SUA, unde își are centrul administrativ (<http://www.acm.org/>). A se vedea și **Hamming, Richard Wesley**.

acronim (*acronym*). Cuvânt format din litere (primele, de obicei) sau din grupuri de litere ale unor cuvinte (e.g., HTTP este un acronim pentru **H**yper**T**ext **T**ransfer **P**rotocol; Benelux este un acronim care comprimă numele a trei țări vecine: **B**elgium, **N**etherlands și **L**uxembourg).

ActionScript. Limbaj de scripting, complet orientat pe obiect, dezvoltat inițial de Macromedia și ulterior de Adobe pentru a fi utilizat în mediile *Flash* (e.g., *Adobe Flash*). Limbajul, având la bază standardul ECMA, le permite dezvoltatorilor să adauge interactivitate animațiilor *Flash*. A se vedea și **Adobe Flash** și **limbaj de scripting**.

ActiveX. Cadru de lucru care permite componentelor software reutilizabile (denumite *controale*) care îndeplinesc o funcție sau un set de funcții particulare în aplicațiile rulate în sistemele bazate pe Microsoft Windows să interacționeze într-o modalitate independentă de limbajul de programare folosit pentru implementarea lor. ActiveX a fost elaborat de Microsoft în 1996, având la bază tehnologiile COM și OLE. A se vedea și **COM**, **controale ActiveX** și **OLE**.

Ada. Limbaj de programare de nivel înalt, imperativ, orientat pe obiect. A fost proiectat inițial de o echipă condusă de Jean Ichbiah (între 1977 și 1983) și extins ulterior de S. Tucker Taft (între 1992 și 1995). Este utilizat în aplicații *mission-critical* (e.g., software pentru domeniul aviatic). Numele limbajului a fost dat în onoarea contesei Ada Lovelace (fiica lordului Byron), considerată a fi primul programator. A se vedea și **programare orientată pe obiect**, **limbaj de nivel înalt** și **Lovelace, Ada**.

adaptor de rețea (*network adapter*). Numit și *adaptor LAN* (*LAN adapter*) sau *placă de rețea* (*network card*), este un dispozitiv utilizat pentru conectarea unui calculator la o rețea locală. Fiecare placă de rețea este identificată printr-un număr unic pe 48 de biți (furnizat de IEEE), denumit adresă MAC sau adresă fizică, stocat într-o memorie ROM proprie. A se vedea și **adresă fizică**, **IEEE**, **rețea de calculatoare** și **ROM**.

adaptor wireless (WNIC, *Wireless Network Interface Controller*). Placă de rețea utilizată pentru conectarea calculatorului la o rețea fără fir. Un dispozitiv WNIC este conectat la calculator folosind magistrala PCI sau portul USB. A se vedea și **adaptor de rețea** și **USB**.

adâncimea culorii (*color depth*). Caracteristică a imaginilor în format *raster*, egală cu numărul de biți utilizați pentru stocarea culorii unui pixel. Culoarea imaginilor *raster* necomprimate poate fi stocată utilizând pentru

un pixel: un bit (*monochrome*), 8 biți (*grayscale* și *colour-map*) sau 24 de biți (*truecolor*).

În imaginile *monochrome*, fiecare pixel este stocat folosind un singur bit (0 sau 1). Astfel, o imagine monocromă, cu dimensiunile 640x480, necesită 37,5 kB pentru stocare ($640 \times 480 / 8 = 38\,400$ octeți, adică $38\,400 / 1024 = 37,5$ kB). În imaginile *grayscale*, culoarea unui pixel este stocată folosind un octet. O astfel de imagine, cu dimensiunile 640x480, necesită 300 kB pentru stocare ($640 \times 480 = 307\,200$ octeți, adică $307.200 / 1024 = 300$ kB).

În imaginile *color pe 8 biți (colour-map)*, culoarea unui pixel este stocată folosind un octet. Imaginile de acest tip suportă 256 de culori simultan, dintr-o paletă incluzând milioane de posibilități. O astfel de imagine, cu dimensiunile de 640x480, necesită 300 kB pentru stocare (ca și o imagine *grayscale*).

În imaginile *color pe 24 de biți (truecolor)*, culoarea fiecărui pixel este reprezentată utilizând trei octeți și un model cromatic (e.g., RGB – Red, Green, Blue și CMY – Cyan, Magenta, Yellow). Suportă $256 \times 256 \times 256 = 16\,777\,216$ de posibile combinații de culori. O astfel de imagine necesită 900 kB pentru stocare ($640 \times 480 \times 3 = 921\,600$ octeți, adică $921\,600 / 1024 = 900$ kB). Cele mai multe imagini color pe 24 de biți sunt, de fapt, imagini color pe 32 de biți în care fiecare pixel are alocat un octet suplimentar utilizat pentru a stoca o valoare de tip *alpha channel*, aceasta

reprezentând informații despre transparența imaginii. A se vedea și **formate raster**.

add-on (add-on). Program de mici dimensiuni care extinde funcțiile unei aplicații (e.g., browsere). Există chiar și un depozit oficial de *add-on*-uri pentru software-ul dezvoltat de Mozilla (*Mozilla Firefox*, *Mozilla Thunderbird*, *SeaMonkey* și *Mozilla Sunbird*), la adresa <https://addons.mozilla.org/>. A se vedea și **browser Web**.

ADL (Advanced Distributed Learning – Inițiativa de învățare avansată distribuită). Proiect comun al instituțiilor guvernamentale, agenților economici și comunității academice, având ca scop fundamentarea unui nou mediu de învățare distribuită care să permită interoperabilitatea instrumentelor de învățare și a conținuturilor la scară globală (<http://www.adlnet.gov/>). ADL produce SCORM, standard pentru administrarea obiectelor de învățare. A se vedea și **obiect de învățare** și **SCORM**.

administrator de sistem (system administrator). Persoană având rolul de a opera, administra și menține în stare de funcționare optimă un sistem de calcul multiutilizator sau o rețea de comunicații. Deține contul rădăcină (i.e., contul de superutilizator). A se vedea și **cont-rădăcină** și **superutilizator**.

Adobe Acrobat. Software pentru crearea documentelor PDF, produs și distribuit de Adobe Systems. Este

realizat în trei variante: *Adobe Acrobat Standard*, *Adobe Acrobat Pro* și *Adobe Acrobat Pro Extended*. A se vedea și **Adobe Reader**, **Adobe Systems** și **PDF**.

Adobe Fireworks. Editor comercial pentru grafică bitmap și vectorială (fost *Macromedia Fireworks*), produs și distribuit de Adobe Systems. A se vedea și **Adobe Systems**.

Adobe Flash. Platformă multimedia (fostă *Macromedia Flash*) dezvoltată și distribuită de compania americană Adobe Systems, utilizată pentru crearea animațiilor *Flash*. A se vedea și **Adobe Systems**, **Flash** și **SWF**.

Adobe Flash Player. Software creat de Macromedia și dezvoltat astăzi de Adobe Systems, distribuit gratuit, funcționând ca *plugin* în browser, utilizat pentru vizualizarea animațiilor și filmelor în format *Flash* incluse în paginile Web. Poate rula fișiere SWF create cu *Adobe Flash*, *Adobe Flex* sau cu instrumente dezvoltate de alte companii. Are versiuni pentru numeroase platforme, inclusiv mobile (e.g., Windows, Linux, Solaris, Mac OS X, Windows Mobile, iPhone OS, Symbian OS, Palm OS, Pocket PC). A se vedea și **Adobe Flash**, **Adobe Flex** și **Flash**.

Adobe Flex. Kit de dezvoltare furnizat de compania Adobe Systems în scopul creării aplicațiilor RIA, bazat pe platforma *Adobe Flash*. În februarie 2008, Adobe Systems a lansat *Flex 3 SDK* sub licență *open source* Mozilla

Public License. A se vedea și **Adobe Flex Builder** și **RIA**.

Adobe Flex Builder. IDE dezvoltat de compania Adobe Systems pe baza platformei *Eclipse*, utilizat pentru crearea aplicațiilor RIA și *desktop*. A se vedea și **Adobe Flex**, **IDE** și **RIA**.

Adobe Illustrator. Editor grafic vectorial, creat și distribuit de Adobe Systems. A fost lansat în 1986, pentru calculatoarele Apple Macintosh. Începând cu 1992, *Illustrator* are versiuni și pentru Windows. A se vedea și **Adobe Systems**.

Adobe InDesign. Unul dintre liderii aplicațiilor de DTP (*desktop publishing*), creat de Adobe Systems și lansat în 1999. A se vedea și **Adobe Systems** și **DTP**.

Adobe Photoshop. Aplicație utilizată pentru editarea imaginilor digitale, creată de frații Thomas și John Knoll. Aceștia au semnat în 1988 un contract de licențiere cu Adobe Systems. Împreună cu *Photoshop* este livrat și *Adobe Image Ready*, un editor de imagini bitmap. A se vedea și **Adobe Systems**.

Adobe Reader. Program gratuit oferit de compania Adobe Systems, utilizat pentru vizualizarea documentelor în format PDF. A se vedea și **Adobe Acrobat**, **Adobe Systems** și **PDF**.

Adobe Systems. Una dintre cele mai mari companii americane de software, fondată în decembrie 1982 de John Warnock și Charles Geschke

(<http://www.adobe.com/>). Cele mai cunoscute produse ale companiei sunt incluse în pachetul de programe *Creative Suite*, care conține, printre altele, *Adobe Acrobat*, *After Effects*, *Dreamweaver*, *Fireworks*, *Flash*, *Illustrator*, *InDesign*, *Photoshop*, *Premiere*). De asemenea, compania mai produce *Adobe Premiere*, *PageMaker*, *GoLive*, *LiveMotion* și *Adobe Flex*. A se vedea și **Adobe Acrobat**, **Adobe Fireworks**, **Adobe Flash**, **Adobe Illustrator**, **Adobe InDesign** și **Adobe Photoshop**.

adresă de e-mail. Este asociată unui cont de *e-mail*, având forma `nume_utilizator@nume_calculator`, în care prima parte reprezintă numele (*i.e.*, identificatorul) utilizatorului, iar cea de-a doua, situată după caracterul @, permite identificarea calculatorului pe care se află căsuța poștală (*i.e.*, server de *e-mail*), organizația de care aparține acesta și/sau țara (domeniul). A se vedea și **poșta electronică**.

adresă fizică (*physical address*). Denumită și **adresă MAC** (*Media Access Control*), este adresa unui adaptor Ethernet (denumit și *placă de rețea* sau *adaptor de rețea*), fiind înscrisă în hardware-ul acestuia. Adresa fizică este alcătuită din 6 octeți (*i.e.*, 12 cifre hexazecimale). Fiecare producător are grijă ca adresa MAC să fie unică, rețelele Ethernet bazându-se pe această unicitate în livrarea mesajelor. A se vedea și **adaptor de rețea**, **ARP** și **RARP**.

adresă IP (*IP address*). În sistemul de adresare IPv4 (*Internet Protocol*, versiunea 4) este un număr binar pe 32 de biți (4 octeți) care identifică în mod unic o gazdă (un calculator) conectată la Internet. În termeni strict tehnici, o adresă IP identifică o interfață capabilă să trimită și să primească pachete IP. Un sistem poate avea interfețe multiple. Oricum, atât gazdele, cât și routerele trebuie să aibă fiecare cel puțin câte o adresă IP. Este exprimată prin valorile în sistem zecimal ale celor patru octeți, separate prin puncte, convenție denumită *Dot-Decimal Notation* (*e.g.*, 127.0.0.1).

O adresă IP este alcătuită din următoarele două părți: identificatorul (sau numărul) de rețea și identificatorul (sau numărul) de gazdă. Fiecare identificator (*id*) este reprezentat prin intermediul unui număr de biți: $adresa\ IP = \langle id\ rețea \rangle \langle id\ gazda \rangle$. Numerele de rețea sunt atribuite în mod centralizat de InterNIC. Primul sau primii (doi sau trei) octeți identifică rețeaua la care este conectată gazda, iar restul biților identifică gazda. Prin intermediul celor 32 de biți pot fi specificate aproximativ patru miliarde de adrese IP.

O adresă IP în care toți biții identificatorului de gazdă sunt înlocuiți cu valoarea 0 se numește *adresă IP de rețea*, reprezentând și adresa IP a primei gazde incluse în rețea. O adresă de rețea definește un *spațiu de adrese*, acesta incluzând toate adresele IP care au același identificator de rețea.

O serie de adrese IP sunt rezervate pentru utilizări speciale. O adresă de acest fel este 0.0.0.0, care identifică gazda locală, fiind utilizată în timpul procesului de *boot*.

Tot pentru utilizări speciale sunt rezervate și adresele în care una dintre componente (de rețea sau de gazdă) are toți biții 0 sau 1: toți biții 0 înseamnă *aceasta (this)*, adică *această rețea* (adresă IP cu <identificator rețea> = 0) sau *această gazdă* (<identificator gazda> = 0). Când o gazdă vrea să comunice în rețea, dar nu cunoaște încă adresa de rețea, poate trimite pachete având <identificator rețea> = 0. Alte gazde din rețea vor interpreta adresa ca însemnând *această rețea*, iar răspunsul lor va conține adresa de rețea pe care expeditorul o va înregistra pentru a o folosi în continuare; toți biții 1 înseamnă *toate (all)*, adică *toate rețelele* sau *toate gazdele*. De exemplu, adresa 130.20.255.255 înseamnă *toate gazdele* din rețeaua 130.20 (de clasă B). Adresele în care toți biții identificatorului de gazdă sunt egali cu 1 (numite *adrese de broadcast* sau *de difuzare*) permit mașinilor să trimită pachete de difuzare în LAN-uri la distanță (tuturor gazdelor incluse într-o astfel de rețea), oriunde în Internet.

Tot pentru utilizare specială sunt păstrate și adresele 127.x.y.z (*i.e.*, incluse în spațiul de adrese definit de rețeaua de clasă A 127.0.0.0). Acestea sunt rezervate pentru testarea și depanarea programelor în bucla locală

(*loopback*). Pachetele IP trimise către o astfel de adresă nu pleacă, de fapt, din mașina locală, fiind totuși tratate ca pachete sosite. A se vedea și **clasă de adrese IP și InterNIC**.

adrese IP private (*private IP address*). Aceste adrese (care nu sunt atribuite de InterNIC) sunt utilizate în mod obișnuit în rețele locale mici din instituții de învățământ, aeroporturi, hoteluri, firme mici etc. Calculatoarele cu adrese IP private nu pot fi conectate direct la Internet. În mod corespunzător, calculatoarele din afara rețelei locale nu pot comunica direct cu un calculator cu o astfel de adresă. Comunicarea se face utilizând un *router* sau un dispozitiv similar care suportă tehnica NAT (*Network Address Translation*). Tehnica amintită schimbă – la plecarea datelor – o adresă privată într-una publică (vizibilă în Internet). La returnul traficului, adresa privată este refăcută, pentru ca datele întoarse să ajungă la calculatorul de origine. Spațiul adreselor IP private este alcătuit din următoarele trei grupuri:

10.0.0.0–10.255.255.255;
172.16.0.0–172.31.255.255;
192.168.0.0–192.168.255.255.

A se vedea și **adresă IP și NAT**.

adresă MAC (*MAC address*). A se vedea **adresă fizică**.

adware. Program de tip *malware* care include reclame sau alte tipuri de materiale publicitare, încărcate automat și afișate după instalare. De asemenea, aceste programe trimit automat către un destinatar prestabilit informații despre

calculator sau activitățile utilizatorului, fără permisiunea acestuia. În ceea ce privește resursele sistemului, programele *adware* ocupă spațiu pe hard-disk, consumă din puterea procesorului, folosesc lățime de bandă și scad viteza de transfer în rețea. A se vedea și **malware** și **spyware**.

AeL (*Advanced e-Learning*). Soluție completă de *e-Learning* dezvoltată de compania românească SIVCO și utilizată pentru managementul învățării și al conținutului educațional, reprezentând un suport integrat pentru predare, învățare și evaluare. *AeL* are rolul de a sprijini educatorii, studenții, dezvoltatorii de conținut, precum și pe alți participanți la procesul educațional (e.g., manageri și personal administrativ).

Platforma *AeL* poate administra și livra diverse tipuri de conținut educațional: materiale interactive multimedia, tutoriale, exerciții, simulări, jocuri educaționale etc.

Cele mai importante caracteristici ale *AeL* sunt următoarele: are o interfață prietenoasă, adaptabilă, diferențiată pe roluri, grupuri și drepturi de acces; rolurile, grupurile, utilizatorii și drepturile de acces sunt foarte ușor de administrat; este bazat pe standarde, fiind compatibil cu MathML, SCORM și IMS Content Packaging; este ușor de instalat și administrat; oferă suport multi-lingvistic și regional, *AeL* fiind ușor de transpus într-o altă limbă și ușor de reconfigurat. A se vedea **IMS Content Packaging** și **SCORM**.

AES (*Advanced Encryption Standard – Standard avansat de criptare*). Algoritm pentru criptarea datelor, denumit și *Rijndael*, adoptat ca standard de NIST (*National Institute of Standards and Technology*). Este utilizat pe scară largă în aplicații, înlocuind cifrul DES (*Data Encryption Standard*), considerat nesigur. Algoritmul AES a fost dezvoltat de criptograful belgieni Joan Daemen și Vincent Rijmen. A se vedea și **DES**.

agent-utilizator (*user-agent*). Ortografiată și *user agent*, este o aplicație-client utilizată pentru un protocol de rețea particular. În mod obișnuit, termenul este folosit în legătură cu accesul la Web. Agenții de utilizator Web sunt browsere Web, clienți de *e-mail*, roboții motoarelor de căutare, telefoane mobile, browsere Web folosite de utilizatorii cu dizabilități etc. A se vedea și **User-Agent**.

agent de consultare (*advisory agent*). Agent inteligent care oferă consultare și suport pentru decizie. Agenții de acest tip decid în ceea ce privește informațiile de care au nevoie, le caută și le utilizează pentru a face recomandări.

agregator (*aggregator*). Software-client sau o aplicație Web care permite abonarea la *feed-uri* în scopul organizării și vizualizării facile a conținutului lor. Este cunoscut și ca *feed reader* (cititor de *feed-uri*) sau *news reader* (cititor de știri).

Agregatoarele pot fi încadrate în următoarele categorii: bazate pe Web, pentru *desktop*, incluse în aplicații, bazate pe *e-mail*, în linie de comandă, pentru dispozitive mobile.

Agregatoarele pentru *feed*-uri bazate pe Web, numite și agregatoare *online*, sunt de două tipuri: de sine stătătoare (e.g., *Bloglines*) și găzduite de motoarele clasice de căutare (e.g., *Google Reader*).

Agregatoarele pentru *desktop* pot fi descărcate și instalate pe calculatorul utilizatorului. În funcție de platforma pe care funcționează, agregatoarele sunt incluse în următoarele categorii: *cross-platform*, pentru platforma Windows, pentru platforma Linux, pentru platforma Mac OS. Iată câteva dintre cele mai utilizate agregatoare pentru *desktop*: *FeedDemon*, *FeedReader*, *RSSBandit*, *WinRSS*, *NewsLife*, *Akregator*).

Dintre aplicațiile care conțin agregatoare pentru *feed*-uri sau care le pot include ca *plugin*-uri pot fi amintite: browserele (e.g., *Firefox*, *Internet Explorer 7*, *Opera*); clienții de *e-mail* (e.g., *Mozilla Thunderbird*, *Outlook Express*); sistemele de gestiune a bazelor de date (e.g., *IBM Lotus Notes/Domino*).

Popularitatea crescândă a formatelor RSS și Atom i-a determinat pe producătorii de browsere să implementeze posibilitatea de a vizualiza acest tip de știri. Browserele *Firefox*, *Opera* (începând cu versiunea 8.0) și *Internet Explorer* (începând cu versiunea 7) permit vizualizarea *feed*-urilor RSS și

Atom în format XHTML, având implementată și opțiunea de abonare (e.g., *Live Bookmarks* în *Firefox*). De asemenea, *Safari* – browserul Web pentru Mac OS X (și, de curând, pentru Windows) – include un agregator de *feed*-uri RSS și Atom. Dacă pagina Web încărcată în browser are setată în secțiunea antet informații despre *feed*-urile incluse (*RSS Autodiscovery*), în bara de adrese va fi inclus icon-ul RSS. Efectuând un click pe iconul amintit, va fi afișată o listă cu *feed*-urile respective. După selectarea unuia dintre ele, acesta va fi afișat de cititorul inclus în browser, iar utilizatorul va avea și posibilitatea de abonare la *feed*. A se vedea și **feed**.

Aiken, Howard (1900–1973). Fizician american, pionier în domeniul calculatoarelor. În 1940, la Harvard, a proiectat prima mașină complexă electromecanică de calcul, denumită Mark I, construită apoi de specialiști ai firmei IBM. Calculatorul folosea relee electromagnetice controlate electronic și un sistem de introducere, stocare și prezentare a rezultatelor pe cartele perforate. Pentru realizările sale, în 1970 a primit medalia Edison. A se vedea și **Mark I**.

AIML (*Artificial Intelligence Markup Language*). Limbaj bazat pe XML, utilizat pentru crearea programelor de tip *chatbot* (e.g., ALICE). A se vedea și **ALICE**, *chatbot* și **XML**.

AJASON. Bibliotecă PHP 5 (pentru server) și JavaScript (pentru client)

dezvoltată sub licență GNU GPL, care implementează tehnica AJAX (<http://ajason.fantastic-bits.de/>).

Formatul utilizat pentru datele schimbate între client și server este JSON. Pentru utilizarea sa sunt necesare un server Web cu modulul PHP inclus și un browser în care este activat suportul JavaScript. Caracteristicile principale ale AJASON sunt următoarele: este orientată complet pe obiect, utilizează apeluri asincrone ale funcțiilor PHP și ale metodelor obiectelor JavaScript; între client și server pot fi schimbate date complexe, inclusiv tablouri și proprietăți ale obiectelor; utilizează funcții *callback* JavaScript pentru procesarea răspunsului serverului; pe client sunt raportate erorile AJASON generate la nivelul serverului. A se vedea și **AJAX**, **JavaScript**, **JSON** **PHP** și **programare orientată pe obiect**.

AJAX (*Asynchronous JavaScript and XML*). Set de tehnologii interconectate folosit pentru crearea de aplicații Web avansate. Principalul avantaj al acestei tehnici (implementată în browserele moderne) este gradul ridicat de interactivitate al aplicațiilor Web care o folosesc, datorat schimbului asincron de pachete mici de date cu serverul, astfel încât nu trebuie reîncărcată întreaga pagină pentru a reflecta opțiunile și selecțiile realizate de utilizator. Acest mod de lucru mai are ca rezultat, alături de îmbunătățirea interactivității, creșterea vitezei și funcționalităților aplicațiilor.

Tehnologiile folosite în AJAX sunt: limbaje standardizate de prezentare a datelor (XHTML și CSS); standardul DOM, folosit pentru redare și interacțiune; formatul XML (sau alte formate, precum XSLT, utilizate pentru schimbul de date și manipularea acestora); obiectul XMLHttpRequest, folosit pentru transferul asincron de date; ECMAScript, limbaj utilizat pentru procesarea datelor.

Ca oricare altă aplicație Web, o aplicație bazată pe AJAX conține, în principiu, două părți: o parte care se execută pe client și o parte care se execută pe server. Ceea ce caracterizează partea care se execută pe client a aplicațiilor AJAX este că aceasta conține o logică de control specifică paginii, introdusă prin intermediul JavaScript, care permite o interacțiune bazată pe evenimente (*e.g.*, *click*, *mouseover*, *mouseout*, *blur*, *change*, *keyup*, *keydown*). Această interacțiune determină o separare clară a datelor de logica de prezentare. O pagină Web poate fi împărțită în mai multe fragmente (piese), conținutul fiecăruia putând fi actualizat independent de al celorlalte, prin intermediul AJAX. Astfel, nu este necesară reîncărcarea întregii pagini atunci când se dorește actualizarea conținutului unui anumit fragment al paginii.

AJAX impune o arhitectură pe partea de server diferită de cea a aplicațiilor Web clasice, compatibilă cu modelul de interacțiune specific. Într-o aplicație

Web clasică, partea care se execută pe server se concentrează pe generarea de conținut HTML sau de altă natură, de fiecare dată când o acțiune a clientului determină un apel către server. Atunci, clientul va reîmprospăta întreaga pagină pentru fiecare răspuns al serverului. În schimb, partea de server a aplicațiilor bazate pe AJAX este alcătuită din componente care generează date XML, acestea din urmă fiind transformate (prin intermediul foilor de stil XSLT și CSS) și injectate drept conținut al fragmentelor documentului HTML. Astfel, pagina Web poate fi considerată ca un container, în ale cărui fragmente serverul injectează conținut actualizabil la apariția unor evenimente.

Dintre posibilele utilizări ale AJAX (care se constituie totodată și ca tipuri de aplicații Web bazate pe această tehnică) pot fi menționate: validarea în timp real a datelor introduse în câmpurile formularelor: anumite date introduse într-un formular (*e.g.*, identificatori, coduri poștale, numere seriale) care necesită validare pe server pot suferi acest proces înainte ca utilizatorul să determine trimiterea întregului său conținut; autocompletare: o dată introdusă într-un câmp al unui formular (*e.g.*, adresă de *e-mail*, nume) poate fi autocompletată odată ce utilizatorul a introdus câteva caractere; reîmprospătarea datelor incluse în diverse fragmente ale paginii: tabele, reprezentări grafice, diagrame, clasamente etc.; notificări ale serverului: pot fi determinate notificări ale serverului către client realizate prin diverse

mesaje, precum și reîmprospătarea datelor sau redirectionări; construirea unor interfețe cu utilizatorul sofisticate: pot fi realizate diverse controale (*e.g.*, meniuri, arbori, bare de progres), care vor fi utilizate fără reîmprospătarea întregii pagini. A se vedea și **CSS**, **DOM**, **XHTML**, **XML**, **XMLHttpRequest** și **XSLT**.

Akregator. Agregator RSS pentru KDE (Linux). A se vedea și **agregator** și **feed**.

alfa (*alpha*). Ceea ce se referă la un produs software aflat în versiune alfa. Aceasta este o versiune primară, neconținând toate caracteristicile care urmează să fie implementate în produs. În mod obișnuit, un produs software trece prin două etape de testare înainte de a fi considerat finalizat: alfa și beta. În stadiul alfa, produsul este testat de obicei numai de utilizatorii membri ai organizației care îl dezvoltă. A se vedea și **beta**.

alfanumeric (*alphanumeric*). Ceea ce este format din litere, cifre sau o combinație a acestora. Setul de caractere alfanumerice include cifrele de la 0 la 9, literele mici (de la a la z) și literele mari (de la A la Z) ale alfabetului latin, în total 62 de caractere (10 cifre, 26 de litere mici și 26 de litere mari). A se vedea și **sortare alfanumerică**.

algebra booleană (*boolean algebra*). Algebră definită axiomatic, dezvoltată în anul 1840 de George Boole. Are la

bază: elementele $\{0,1\}$; două operații binare numite SAU și ȘI (notate simbolic + sau U și, respectiv, x sau \cap); o operație unară numită NU (sau negație, notată simbolic -). Algebra booleană este unul dintre elementele fundamentale aflate la baza tehnicii de calcul. A se vedea și **Boole, George**.

ALGOL (*ALGO*rithmic Language). Primul limbaj de programare structurat, procedural, dezvoltat de un comitet de specialiști în informatică europeni și americani. Aceștia s-au întâlnit în 1958 la ETH Zurich (*Institutul Federal de Tehnologie*), din munca lor rezultând prima specificație a limbajului, ALGOL 58 (denumit inițial, provizoriu, IAL: *International Algorithmic Language*). Mai târziu, au fost formulate încă două specificații majore: ALGOL 60 și ALGOL 68. A avut o influență decisivă asupra multor limbaje de programare dezvoltate ulterior. A se vedea și **limbaj de programare**.

algoritm (*algorithm*). Termen utilizat în diverse domenii (*e.g.*, matematică, informatică) pentru a desemna o secvență finită de pași (sau *instrucțiuni*) care rezolvă o problemă legată adesea de efectuarea unor calcule sau de procesarea unor date.

Termenul „algoritm” este legat de numele matematicianului și astrologului persan Abu Ja'far Mohammed ibn Musa al-Khowarizmi (sec. VIII-IX) și, literal, înseamnă „din orașul Khowarizmi” („al-Khowarizmi”). Acest oraș se numește în prezent Khiva și se află în Uzbekistan. Al-Khowarizmi a

scris o carte de matematică având titlul (în latină) *Algorithmi de numero indorum*. Atât el, cât și alți matematicieni ai Evului Mediu înțelegeau prin *algoritm* o regulă folosită pentru efectuarea calculelor aritmetice (*e.g.*, dublări, înjumătățiri, înmulțiri de numere). Deși termenul a fost utilizat mai târziu și de Leibnitz (în secolul al XVIII-lea), precum și de Kronecker și Dedekind (în secolul al XIX-lea), abia în deceniile 3 și 4 ale secolului XX teoria (recursivității și) algoritmilor începe să se constituie ca atare, prin lucrările lui Skolem, Ackermann, Gödel, Turing și alții. În această perioadă, gândirea algoritmică s-a transformat dintr-un instrument matematic particular într-o modalitate esențială de abordare a problemelor în diverse domenii.

Un algoritm are două componente: domeniul algoritmului și descrierea propriu-zisă a algoritmului. Orice algoritm lucrează asupra unor date de intrare care sunt prelucrate, obținându-se astfel rezultatul scontat. Domeniul algoritmului este reprezentat de mulțimea obiectelor asupra cărora este aplicat; trebuie ca aceasta să fie numărabilă (*e.g.*, submulțimile unei mulțimi finite, mulțimea numerelor naturale). În acest domeniu sunt incluse și rezultatele de ieșire. Algoritmul este alcătuit din instrucțiuni, fiecare conținând operații elementare care se aplică asupra datelor din domeniul algoritmului. Parcurgerea unui algoritm de la început până la sfârșit este denumită *execuția algoritmului*.

Programul este exprimarea unui algoritm într-un limbaj de programare.

Caracteristicile unui algoritm sunt: generalitate (trebuie să rezolve o întreagă clasă de probleme); finitudine (trebuie să includă un număr finit de pași, indiferent de datele de intrare); determinism (la un moment dat se efectuează o singură operație bine specificată); optimalitate (trebuie să fie executat într-un număr minim de pași); corectitudine (trebuie să furnizeze o soluție corectă).

În funcție de soluția aleasă pentru implementare, un algoritm poate fi: recursiv (se folosește pe el însuși, repetat), iterativ (repetitiv), serial sau paralel, determinist sau aleator, exact sau aproximativ. De asemenea, în funcție de paradigma utilizată, algoritmii pot fi: *backtracking*; *divide et impera*; de tip *greedy*; de tip programare dinamică; genetici, probabilistici, euristici.

Studiul algoritmilor presupune parcurgerea următoarelor etape: stabilirea cerințelor și a datelor de intrare și de ieșire; proiectarea algoritmilor (se face în limbaj natural, în pseudocod de cele mai multe ori sau, mai rar, folosind schemele logice); codificarea algoritmilor folosind un limbaj de programare; validarea algoritmilor (se demonstrează că aceștia rezolvă corect problema); analiza algoritmilor (se caută și alți algoritmi care să rezolve problema dată; dacă sunt găsiți, se vor compara toți algoritmii între ei în ceea ce privește ușurința înțelegerii și

eficiența). A se vedea și **pseudocod și schemă logică**.

algoritm de căutare (*search algorithm*). **1.** Utilizat în informatică, este responsabil pentru căutarea în diverse structuri de date (*e.g.*, tablouri, liste, baze de date, fișiere, arbori, grafuri). Cei mai cunoscuți algoritmi de căutare în tablouri sunt căutarea secvențială (numită și liniară) și căutarea binară. **2.** În legătură cu căutarea în Internet folosind motoarele de căutare, semnifică un algoritm responsabil pentru luarea deciziei în ceea ce privește rezultatele care îi vor fi returnate utilizatorului. Din cauza diferenței dintre algoritmii utilizați, este posibil ca motoare de căutare diferite (*e.g.*, *Google* și *Yahoo!*) să genereze rezultate diferite pentru aceeași expresie de căutare. Motorul de căutare *Google* calculează un așa-numit *PageRank* (rangul paginii) pentru fiecare pagină indexată, care stabilește rangul (poziția) acesteia în cadrul rezultatelor amintite. A se vedea și **căutare binară, căutare secvențială, motor de căutare și PageRank**.

algoritm de planificare (*scheduling algorithm*). Algoritm utilizat într-un sistem de operare pentru a acorda proceselor și fluxurilor de date acces la resursele sistemului (*e.g.*, timp de procesor, lățime de bandă pentru comunicație). Scopul utilizării unui algoritm de planificare este echilibrarea sistemului și atingerea unui nivel ridicat al calității serviciilor oferite de sistemele de operare moderne.

Cel mai simplu algoritm este *round robin*, care acordă – pe rând – timpi egali de execuție fiecărui proces. De asemenea, pot fi utilizați și algoritmi complecși, care iau în considerare niveluri de prioritate ale proceselor, ceea ce permite unora dintre acestea să utilizeze mai mult timp resursele sistemului în raport cu altele. Astfel, *kernel*-ul poate utiliza oricând oricâte resurse are nevoie în scopul asigurării funcționării corecte a sistemului, ceea ce înseamnă că are o prioritate infinită (*i.e.*, cea mai mare). A se vedea și **kernel**, **planificator**, **round robin** și **sistem de operare**.

algoritm de sortare (*sorting algorithm*). Utilizat în matematică și informatică, este responsabil pentru așezarea elementelor unei liste într-o ordine specificată (de obicei numerică sau lexicografică). Cei mai populari algoritmi de acest tip sunt algoritmi de sortare prin comparație: metoda bulelor (*bubble sort*), sortarea rapidă (*quick sort*), sortarea prin interclasare (*merge sort*), sortarea prin metoda arborelui heap (*heap sort*), sortarea prin selecție (*selection sort*) și sortarea prin inserție (*insertion sort*). A se vedea și **algoritm**, **sortare prin inserție**, **sortare prin interclasare**, **sortare prin metoda arborelui heap**, **sortare prin metoda bulelor**, **sortare prin selecție** și **sortare rapidă**.

algoritm Lempel-Ziv (*Lempel-Ziv algorithm*). Algoritm de comprimare fără pierderi a unui fișier. Practic, se referă la doi algoritmi (LZ1 și LZ2),

elaborați de Abraham Lempel (1977) și, respectiv, de Jacob Ziv (1978). Aceștia reprezintă baza pentru alți algoritmi de comprimare fără pierderi (*e.g.*, LZW, LZSS, LZMA). A se vedea și **algoritm**, **algoritm LZW** și **comprimare fără pierderi**.

algoritm LZW (*LZW algorithm*). Algoritm de comprimare fără pierderi creat de Abraham Lempel, Jacob Ziv și Terry Welch. Publicat de Welch în 1984, reprezintă o implementare îmbunătățită a algoritmului Lempel-Ziv. A se vedea și **algoritm Lempel-Ziv**.

alias (*alias*). Unul dintre numele simbolice alternative ale unei gazde având o adresă IP dată. A se vedea și **adresă IP**.

ALICE (*Artificial Linguistic Internet Computer Entity*). *Chatbot* care a câștigat premiul Loebner de trei ori (în 2000, 2001 și 2004), fără a fi totuși capabil să treacă testul Turing. Regulile de conversație respectate de ALICE sunt specificate printr-un limbaj special, AIML (*Artificial Intelligence Markup Language*). A se vedea și **chatbot**.

Allen, Gardner Paul (n. 1953). Antreprenor american și cofondator în anul 1975 (împreună cu Bill Gates) al Microsoft Corporation (companie denumită inițial Micro-Soft). A se vedea și **Gates, William Henry III**.

Allman, Eric (n. 1955). Programator american, creatorul programului de

poștă electronică *sendmail*, unul dintre cele mai vechi din această categorie. A se vedea și **poștă electronică** și **sendmail**.

alocarea resurselor (*resource allocation*). Distribuirea facilităților existente într-un sistem de calcul către diversele aplicații care rulează la un moment dat. Operația este realizată de sistemul de operare. A se vedea și **sistem de operare**.

ALTAIR 8800. Microcalculator proiectat și lansat în anul 1975 de MITS (*Micro Instrumentation and Telemetry Systems*). Este bazat pe microprocesorul Intel 8080 și considerat primul calculator personal. A se vedea și **microcalculator**.

Altavista. Motor de căutare lansat în 1995, încă activ, aflat în proprietatea *Yahoo!* (<http://www.altavista.com/>). A se vedea și **motor de căutare**.

alterare (*corruption*). Proces de degradare a datelor existente în memoria de lucru a calculatorului sau aflate pe un dispozitiv de stocare (e.g., hard-disk, CD, DVD), care conduce la imposibilitatea de a mai fi utilizate.

alt text (*text alternativ*). Text care constituie descrierea unei imagini, afișat de browser numai atunci când acesta (din diverse motive) nu poate afișa imaginea însăși. Textul trebuie să fie o descriere cât mai exactă, dar succintă, a imaginii. Este util roboților motoarelor de căutare pentru a indexa

imaginile, persoanelor care dispun de conexiuni slabe la Internet (și nu doresc să încarce anumite imagini), dar și persoanelor cu deficiențe de vedere care utilizează browsere speciale, pentru a le face accesibil conținutul imaginilor. Iată un exemplu de utilizare în limbajul HTML a atributului:

```
<img src = "exemplu.jpg"  
      alt = "exemplu">
```

A se vedea și **HTML**.

AMBULANT SMIL Player. Player *open source* pentru documentele create folosind limbajul SMIL. Are versiuni pentru platformele Linux, Mac OS X, Windows, Windows TabletPC și Windows PocketPC. Furnizează suport pentru subseturile de specificații Language, Mobile, Extended Mobile și Basic. Dezvoltarea proiectului *AMBULANT SMIL Player* este posibilă datorită unui grant acordat de *NLnet Foundation*. A se vedea și **SMIL**.

AMD (*Advanced MicroDevices*). Companie multinațională americană, producător în domeniul semiconducătorilor, înființată în 1969 de Jerry Sanders III, Ed Turney, John Carrey și alții (<http://www.amd.com/>). Este al doilea producător mondial de microprocesoare bazate pe arhitectura x86, după Intel. Acestea (e.g., Athlon, Athlon XP, Athlon 64, Opteron, Phenom) sunt utilizate în calculatoarele compatibile IBM PC.

analiza rețelelor sociale (*social network analysis*). Domeniu de analiză a relațiilor sociale în care acestea sunt

referite în termeni de noduri și legături. *Nodurile (nodes)* sunt actori individuali în rețele, iar *legăturile (ties)* sunt relații stabilite între actori. Acestea din urmă pot să fie de diverse tipuri. Cercetările arată că rețelele sociale operează la mai multe niveluri, de la cel al familiei până la nivel național, având un rol important în determinarea căilor de rezolvare a unor probleme, în funcționarea organizațiilor și în atingerea propriilor scopuri de către indivizi. A se vedea și **rețea socială**.

ancoră (anchor). În HTML, reprezintă sursa sau destinația unei hiperlegături. Utilizând atributul `href`, ancora devine o hiperlegătură către un alt document (vezi primul exemplu) sau către o altă parte a documentului curent (vezi al doilea exemplu):

```
<a href = "URL_tinta">  
  Catre o alta pagina</a>  
<a href = "#nume_ancora">  
  Catre ancora</a>
```

Folosind unul dintre atributele `name` sau `id`, ancora devine ținta hiperlegăturii în documentul curent, ca în următorul exemplu:

```
<a name="nume_ancora">Ancora</a>
```

A se vedea și **atribut, hiperlegătură, HTML și text-ancoră**.

Andreessen, Marc (n. 1971). Investitor, antreprenor și inginer software, cunoscut mai ales în calitate de coautor al primului browser grafic, *Mosaic*. A fondat în 1994 Netscape Communications Corporation, aflată începând cu 1998 în proprietatea companiei AOL (<http://netscape.aol>

[com/](http://www.mozilla.com/)). A se vedea și **browser Web și Mosaic**.

Android. Sistem de operare *open source* pentru dispozitive mobile (e.g., telefoane inteligente), bazat pe Linux, dezvoltat inițial de compania Google și apoi de OHA (*Open Handset Alliance*) sub licență Apache, lansat în anul 2008 (<http://www.android.com/>). A se vedea și **licența Apache, OHA și telefon inteligent**.

animație GIF (GIF animated). Animație rezultată prin suprapunerea succesivă în aceeași poziție a unor imagini diferite în format GIF. A se vedea și **GIF**.

anonymous (anonim). Numele standard cu care un utilizator se poate conecta la un server FTP public. Utilizatorii care se conectează la server vor introduce *anonymous* atunci când li se solicită numele de utilizator, iar ca parolă vor furniza adresa de *e-mail*. Clienții FTP moderni se conectează transparent la serverele FTP publice, furnizând automat o parolă standard (care depinde de client). Multe servere FTP nu permit accesul anonim, din motive legate de securitate, iar cele care îl permit acordă utilizatorilor numai dreptul de a descărca fișiere. A se vedea și **FTP**.

ANSI (American National Standards Institute). Organizație privată non-profit, care supraveghează dezvoltarea standardelor pentru produse, servicii, procese și sisteme în SUA (<http://www.ansi.org/>). De asemenea,

ANSI coordonează standardele americane cu cele internaționale, operație desfășurată pentru ca produsele americane să poată fi utilizate în întreaga lume.

antet (header). 1. Referitor la o pagină, zonă aflată în partea superioară a acesteia. Poate să includă numărul paginii, precum și alte informații. A se vedea și **subsol**. 2. Una dintre cele două secțiuni incluse într-un document HTML. Conține marcaje al căror conținut nu este afișat în browser. A se vedea și **HTML**.

antet de mesaj (message header). Structură de octeți plasată înaintea unui mesaj, cu scopul de a furniza informații cu privire la sursa, destinația și structura acestuia (e.g., lungime, format). A se vedea și **mesaj**.

antivirus (antivirus). Program utilizat pentru a împiedica infectarea unui sistem cu viruși informatici sau pentru a-i elimina. A se vedea și **dezinfecție** și **virus de calculator**.

AOL LLC. Fostă America Online, este o companie americană furnizoare de servicii Internet și media, fondată în 1983, aflată în proprietatea lui Tim Warner (<http://www.aol.com/>).

Apache. Cel mai popular (începând cu aprilie 1996) și unul dintre cele mai rapide, eficiente și funcționale servere Web, dezvoltat de *Apache Software Foundation* (<http://www.apache.org/>) sub licență *open source* (licență Apache). Serverul își are originea într-o

serie de corecții (*patch-uri*) create pentru NCSA Web Server, iar denumirea sa provine de la *Apache server*. A fost dezvoltat inițial pentru platformele UNIX, Linux și BSD, fiind portat ulterior și pe Windows.

Serverul Web Apache este alcătuit dintr-un nucleu relativ mic, împreună cu o serie de module. Acestea din urmă pot fi compilate static în server, sau – de cele mai multe ori – încărcate dinamic la execuție (*runtime*).

Comportamentul serverului Web Apache și al modulelor sale poate fi controlat prin modificarea valorilor directive-lor incluse în fișierul principal de configurare, numit *httpd.conf*, utilizând directiva `<Directive>...</Directive>`. Fișierele de configurare de tipul *.htaccess* controlează comportamentul serverului la nivelul directorului (și al subdirectoarelor sale) în care este inclus un astfel de fișier. A se vedea și **licență Apache** și **server Web**.

Apache Software Foundation (ASF). Fundație americană non-profit înființată în martie 1999, care are ca scop sprijinirea proiectelor software denumite generic *Apache*, inclusiv *Apache HTTP Server* (<http://www.apache.org/>). ASF este o comunitate descentralizată de programatori, software-ul produs de aceștia fiind distribuit sub licență Apache (Apache License). A se vedea și **Apache** și **licența Apache**.

apel de funcție (function call). Apel care pasează controlul programului

unei funcții. Se realizează prin specificarea în program a numelui funcției, însoțit de lista parametrilor actuali (numiți și efectivi), dacă aceștia există, plasată între paranteze rotunde. După ce funcția este executată, controlul este întors către următoarea instrucțiune din corpul programului. O funcție poate returna o valoare. Funcția apelată poate să fie definită în program sau în alt fișier. A se vedea și **funcție**.

API (*Application Programming Interface, Interfață de programare a aplicațiilor*). Set de instrumente și/sau proceduri folosite pentru realizarea aplicațiilor software. API le furnizează programatorilor un mecanism pentru a utiliza funcționalitățile unui set de module fără a avea acces la codul-sursă. În particular, folosirea unui API în Web nu presupune cumpărarea sau deținerea unei licențe, acestea fiind calificate ca *deschise (open)*. API-urile au determinat dezvoltarea rapidă a serviciilor Web 2.0 și au favorizat crearea unor aplicații denumite *mashups*, care utilizează date din variate surse, transformând Web-ul într-o experiență multimedia complexă. A se vedea și **mashup, serviciu Web și Web 2.0**.

aplicație Web (*Web application*). Ansamblu de pagini Web interconectate, având un conținut generat dinamic și care oferă o funcționalitate specifică (e.g., motor de căutare, magazin virtual, client de *e-mail*).

O aplicație Web este rezidentă pe un server dedicat, fiind accesată de utilizatori prin intermediul unei rețele

(e.g., Internet). Dacă se dorește, aplicațiile Web pot fi disponibile numai în intranet sau extranet, nefiind publice.

În mod obișnuit, o aplicație Web este structurată pe următoarele trei niveluri (*three-tier architecture*): nivelul I – browserul Web; nivelul II sau nivelul intermediar – serverul Web și tehnologia utilizată pentru generarea dinamică a conținutului; nivelul III – sursa/sursele de date. O aplicație Web structurată pe trei niveluri funcționează astfel: browserul trimite cereri nivelului intermediar, acesta efectuează interogări ale bazei de date și generează automat un răspuns pe care îl returnează browserului. Transferul cererii și al răspunsului (către/de la nivelul intermediar) sunt realizate prin intermediul serverului Web.

Funcționalitățile aplicațiilor Web sunt asigurate în cea mai mare parte de server. Din acest motiv, utilizatorii folosesc un client care nu necesită performanțe deosebite, numit *thin client* (i.e., browserul Web). Prin comparație, aplicațiile *desktop* sunt bazate pe un client cu performanțe ridicate, numit *thick client* (sau *fat client*), care oferă cea mai mare parte a funcționalităților. A se vedea și **server Web și site Web**.

aplicații Internet avansate (AIP, *Advanced Internet Applications*). Aplicații bazate pe un set de tehnologii Internet complexe, care realizează activități de proiectare, gestionare a datelor și acces de la distanță la servicii Internet.

Apple, Inc. Fost Apple Computer, Inc. (până în ianuarie 2007), este un producător american de calculatoare personale (denumite MacIntosh), software (e.g., sisteme de operare: Mac OS X, Mac OS X Server; aplicații: *iTunes*, *QuickTime Player*) și dispozitive mobile (e.g., iPod, iPhone, iPad) (<http://www.apple.com/>).

Compania a fost înființată de Steve Jobs și Stephen Wozniak în aprilie 1976. A fost prima companie care a utilizat tehnologia GUI în sistemele sale de operare. În 1977, compania a lansat Apple II, primul calculator personal. De atunci, piața de calculatoare preasamblate a oferit unor largi categorii de persoane posibilitatea de a avea acces la sisteme de calcul, impulsivând astfel dezvoltarea continuă a software-ului. A se vedea și **GUI**, **iPod**, **iPhone** și **iTunes**.

applet. Componentă software care rulează în contextul altui program (e.g., browser Web). Spre deosebire de un program, un *applet* nu poate rula independent, ci într-un container furnizat de programul-gazdă. Iată câteva exemple de tipuri de *applet*-uri: *Java Applets*, *Adobe Flash Applet*, *Media Player Applet*.

arbore (*tree*). În programare, structură de date care generează o structură arborescentă ierarhică, incluzând un set de noduri legate. Mai exact, un arbore este un graf neorientat conex și fără cicluri, în care un nod are zero sau mai multe noduri-copil și cel mult un nod-părinte. Nodul fără părinte se numește

nod-rădăcină. Un nod fără copii se numește *nod-frunză*.

Se poate da și o definiție recursivă unui arbore. Astfel, acesta este un set finit de unul sau mai multe noduri, care îndeplinește condițiile: există un nod unic numit rădăcina arborelui; celelalte noduri sunt repartizate în $k > 0$ seturi disjuncte, fiecare set fiind la rândul său un arbore. Tot recursiv se poate defini nivelul unui nod: nivelul nodului-rădăcină este 1; nivelul oricărui nod este nivelul părintelui + 1 (nivelul unui nod este 1 + numărul de arce care alcătuiesc un „drum” de la nodul rădăcină la el). Un *subarbore* (*subtree*) al unui arbore A este un arbore care conține un nod al arborelui A și toți descendenții acestuia.

Nodurile unui arbore pot fi traversate (enumerare) în următoarele moduri (între paranteze sunt oferite definiții recursive): în *preordine* (*pre-order*; se vizitează mai întâi rădăcina, apoi se traversează pe rând în *preordine* toți subarborii); în *postordine* (*post-order*; se traversează pe rând în *postordine* toți subarborii, apoi se vizitează rădăcina); în *inordine* (*in-order*; se traversează întâi în *inordine* primul subarbore – stâng, se vizitează rădăcina, se traversează pe rând în *inordine* toți ceilalți subarbori). A se vedea și **arbore binar**, **graf** și **structură de date**.

arbore binar (*binary tree*). În programare, colecție de noduri cu următoarele proprietăți: orice nod are 0, 1 sau 2 succesori (copii); orice nod, cu excepția nodului-rădăcină, are un

singur nod predecesor (părinte); nodul-rădăcină nu are predecesori; nodurile-copil sunt ordonate: copilul stâng, copilul drept (dacă un nod are un singur copil, trebuie menționat care este acela). Altfel spus, arborele binar este o mulțime finită de noduri care este fie vidă, fie un arbore ordonat în care fiecare nod are cel mult doi descendenți. A se vedea și **arbore**.

arbore binar complet (*complete binary tree*). Numit și *heap*, este un arbore binar în care fiecare nod care nu este frunză are exact doi descendenți. Se demonstrează că un arbore binar complet care are n noduri terminale, toate situate pe același nivel, are în total $2n-1$ noduri. A se vedea și **arbore binar și heap**.

Archie. Introdus în anul 1990, este un instrument utilizat pentru indexarea arhivelor FTP publice, permițând utilizatorilor să găsească diverse fișiere (*Archie* este o prescurtare de la *archive*, *arhivă*). Este considerat a fi primul motor de căutare în Internet. Versiunea originală a fost scrisă de Alan Emtage, Bill Heelan și J. Peter Deutsh, studenți la McGill University din Montreal. Servele *Archie* păstrează indexuri ale serverelor FTP indexate, iar clienții le accesează. Astăzi mai sunt în funcțiune numai câteva servere *Archie*. A se vedea și **FTP**.

argument implicit (*default parameter*). În programare, argument (al unei *funcții*) pentru care se precizează o valoare implicită, astfel încât nu va

mai fi obligatoriu ca acesta să primească o altă valoare la apelul funcției, atribuirea explicită a unei noi valori fiind totuși posibilă dacă se dorește acest lucru. Argumentele implicite trebuie plasate în partea de sfârșit a listei de argumente (deoarece un argument care urmează unuia implicit trebuie să fie, de asemenea, implicit). Iată un script scris în limbajul de programare PHP în care se definește și se utilizează o funcție (având numele `multiplica()`) a cărei listă de argumente include unul implicit:

```
<?php
function multiplica($x, $y=2)
{
    return $x*$y;
}
echo multiplica(4,5).
    "<br />";
// Se afiseaza 20
echo multiplica(4);
// Se afiseaza 8
?>
```

În exemplul oferit mai sus se poate observa poziționarea argumentului implicit după cel explicit. A se vedea și **funcție**.

arhitectura client-server (*client-server architecture*). Cunoscută și ca *modelul client-server* (*client-server model*), arhitectură care descrie relația dintre două entități cu roluri diferite: una dintre acestea – *clientul* – este activă (trimite cereri și așteaptă și recepționează replicile serverului), iar cealaltă – *serverul* – este pasivă (așteaptă cereri ale clientului, le primește, le procesează și îi trimite

acestui rezultat obținut). Altfel spus, în cadrul acestei arhitecturi, informațiile sunt cerute de clienți și oferite de servere.

Clienții și serverele nu sunt neapărat calculatoare, ci și programe, ceea ce înseamnă că un calculator poate fi simultan server și client. Totuși, în mod obișnuit, programele-server sunt instalate pe calculatoare puternice, pentru a putea deservi rapid numărul mare de clienți care solicită informații.

Programul-client interacționează direct cu utilizatorii prin intermediul unei interfețe grafice. În schimb, în mod obișnuit, programul-server nu interacționează direct cu utilizatorii. Când are nevoie de informații, clientul se conectează la server și îi adresează o cerere. Acesta trimite înapoi un răspuns care poate conține informația solicitată sau precizează că nu dispune de ea. Clientul poate continua să emită cereri, eventual către alte servere.

Aplicațiile Web funcționează conform modelului client-server. Pentru Web, navigatoarele (sau browserele) sunt clienți, iar serverele Web sunt servere.

Principiul descris oferă portabilitate față de platforma hardware, execuție multitasking, execuție multiprocesoare, execuție multifilară (execuția simultană a mai multor fire de execuție).

În principiu, serviciile Internet necesită servere și clienți specializați. Astfel, există servere Web, servere FTP, servere de *e-mail* etc. și, corespunzător, clienți Web (numiți *browsere* sau *navigatoare*), clienți FTP etc. Totuși, în

mod obișnuit, un browser Web este atât client Web, cât și client FTP și client de *e-mail*, dar există și clienți specializați pe câte un singur serviciu. A se vedea și **browser Web și server Web**.

arhitectura von Neumann (*von Neumann architecture*). Arhitectură de calculator imaginată de John von Neumann (1945), care stă la baza calculatoarelor electronice moderne, descrisă în articolul *First Draft of a Report on the EDVAC*. În viziunea lui Neumann, un calculator digital are patru părți componente principale: unitatea aritmetică-logică, UAL (*Arithmetic Logic Unit*, ALU); o zonă de memorie pentru stocarea datelor și a programelor; o unitate de control (realizează transferul datelor și instrucțiunilor între memorie și UAL); dispozitive de intrare/ieșire I/E (sau I/O, de la *input/output*). Aceste părți sunt interconectate prin intermediul unei *magistrale* (*bus*) și conduse în tactul unui *ceas* (*clock*). Deși designul și performanțele calculatoarelor s-au îmbunătățit extrem de mult în comparație cu anii 1940, arhitectura von Neumann se află practic în continuare la baza tuturor mașinilor de calcul utilizate astăzi. A se vedea și **von Neumann, John**.

arhitectură (*architecture*). Planul unui dispozitiv hardware sau produs software. Arhitecturile deschise se referă la sistemele compatibile cu dispozitive sau software create de diverși producători. În schimb, arhitecturile închise se referă la sisteme care includ

caracteristici proprietare, neprezentând compatibilitate cu sistemele altor producători.

arhitectură peer-to-peer (*peer-to-peer architecture*). Arhitectură corespunzătoare unei rețele formate din minim două calculatoare în care nu există servere dedicate și o organizare ierarhică, toate calculatoarele fiind egale. Astfel, într-o rețea *peer-to-peer* (de la egal la egal) fiecare calculator are atât rol de client, cât și de server. Acest lucru înseamnă că nu există un administrator care să se ocupe de rețea, fiecare utilizator trebuind să aibă grijă de propriul sistem (este atât administratorul, cât și utilizatorul acestuia). Costul unei astfel de soluții este în general mic, nefiind necesar să se achiziționeze un sistem care să joace rol de server. Din acest motiv, o rețea *peer-to-peer* este mai ieftină decât o rețea bazată pe server. A se vedea și **arhitectură**.

arhivă (*archive*). 1. Fișier sau grup de fișiere comprimate. 2. Director de fișiere în Internet care poate fi accesat folosind FTP. A se vedea și **FTP**.

arhivă cu autoextragere (*self-extracting archive*). Fișier executabil care conține unul sau mai multe fișiere comprimate (arhivate). Pentru decomprimare este necesară rularea fișierului respectiv. A se vedea și **arhivă (1)**.

ARP (*Address Resolution Protocol, Protocolul de rezoluție a adreselor*). Protocol la nivelul rețea în modelul de

referință TCP/IP, realizează asocierea dintre adresa IP și adresa fizică ale unui nod de rețea.

Pentru a se putea transmite pachete IP trebuie asociate (sau mapate) adreselor IP adrese fizice, deoarece un cadru fizic purtând pachetul IP trebuie să plece de la o adresă fizică și să ajungă la o altă adresă fizică. Nodul-sursă, care nu cunoaște adresa fizică a nodului-destinație, lansează un pachet ARP cu adresa IP proprie și adresa IP de destinație, pe adresa de difuzare, adică spre toate stațiile. Nodul-destinație își recunoaște adresa IP, după care transmite către sursă adresa sa fizică. În scopul reducerii traficului în rețea, fiecare nod reține un tabel (*address resolution cache*) în care sunt incluse adrese IP și adresele fizice corespunzătoare. Tabelul este actualizat pe măsură ce sunt utilizate adrese IP noi. A se vedea și **adresă IP, RARP, IP și TCP/IP**.

ARPANET (*Advanced Research Projects Agency Network*). Rețea de calculatoare considerată predecesorul Internetului, creată de ARPA, o agenție a DoD (*Department of Defense, Ministerul Apărării* al SUA). A fost prima rețea operațională bazată pe comutarea de pachete. ARPANET a început să funcționeze la sfârșitul anului 1969, având patru noduri: University of California din Los Angeles (UCLA), Stanford Research Institute (SRI), University of California, Santa Barbara (UCSB) și University of Utah. În 1971, rețeaua ARPANET avea deja

15 noduri/23 de gazde. În 1972, Robert Kahn realizează la ICCC (*International Computer Communication Conference*) prima prezentare publică a rețelei ARPANET. În anul 1973 au fost conectate la rețea, pentru prima oară, noduri din afara continentului american (primul nod conectat a fost University College of London).

În anul 1980, ARPANET asigura conexiuni între aproximativ 400 de calculatoare răspândite în universități, centre guvernamentale și militare. La aceste calculatoare aveau acces peste 10.000 de persoane.

În 1983 (1 ianuarie), protocoalele NCP (*Network Control Protocol*) utilizate pentru comunicații în rețeaua ARPANET (dezvoltate de *Network Working Group*, finalizate în decembrie 1970 și implementate în rețea în 1971–1972) au fost înlocuite cu suita de protocoale TCP/IP. Tot în anul 1983, rețeaua a fost împărțită în două componente. Una dintre ele, denumită în continuare ARPANET (desființată ulterior, în 1990), interconecta centrele universitare și academice, iar cealaltă, denumită MILNET (care există și astăzi), era utilizată în scopuri militare (U.S.A.). A se vedea și **comutare de pachete și TCP/IP**.

articol de meniu (*menu item*). Opțiune dintr-un meniu. Dacă este disponibilă, poate fi selectată cu tastatura sau cu *mouse*-ul. Dacă opțiunea nu este disponibilă (neputând fi, astfel, selectată), este afișată cu o altă culoare (e.g., gri). A se vedea și **meniu**.

asamblor (*asssembler*). Program care translatează programele scrise într-un limbaj de asamblare în programe executabile, acestea din urmă fiind alcătuite din instrucțiuni scrise în limbaj-mașină. A se vedea și **cod-mașină și limbaj de asamblare**.

ASCII (*American Standard Code for Information Interchange*). Standard utilizat în domeniul calculatoarelor și echipamentelor de comunicație, bazat pe alfabetul englezesc, definește o schemă (un sistem) de codificare pe 7 biți prin atribuirea unor valori numerice elementelor unui set de 128 de caractere (e.g., litere mari și mici, cifre, semne de punctuație, caractere de control). Proiectarea standardului a început în 1960, prima ediție a acestuia fiind publicată în 1963. O revizie importantă s-a produs în 1967, iar cea mai recentă actualizare a avut loc în 1986. A se vedea și **ASCII extins și setul de caractere ASCII**.

ASCII extins (*extended ASCII*). Orice set de caractere asociat valorilor ASCII cuprinse între 128 și 255. Caracterele asociate codurilor ASCII extinse variază în funcție de platformă, aplicație, limba utilizată etc. Folosind valorile ASCII suplimentare pot fi reprezentate diacritice, simboluri speciale, elemente grafice etc.

Din cele 256 de valori (care pot fi reprezentate folosind un octet), prima jumătate (incluzând valori reprezentate pe 7 biți) este folosită de un set comun de caractere (denumit setul de caractere ASCII), cealaltă (incluzând valori

reprezentate pe 8 biți) fiind dedicată codificării specifice, dependentă de limbă.

Există mai multe extensii ale standardului ASCII, acestea fiind descrise de standardele ISO 8859-x, unde x este un număr zecimal format din una sau două cifre (de la 1 la 16). Dintre acestea poate fi remarcat ISO 8859-2 (denumit și *Latin 2*), utilizat pentru codificarea limbilor indo-europene răsăritene (inclusiv româna). În acest standard există o problemă referitoare la diacriticele românești ș (Ș) și ț (Ț), acestea fiind reprezentate folosind sedila, și nu virgula, după cum cer regulile specificate de *Institutul de Lingvistică al Academiei Române*. Problema a fost rezolvată în ISO 8859-16 (Latin 10), standard utilizat pentru codificarea caracterelor folosite în limbile sud-est europene (albaneză, ungară, croată, română, poloneză), dar și în alte limbi (franceză, germană și italiană). A se vedea și **ASCII**, **setul de caractere ASCII**, **UTF-8** și **UTF-16**.

asistent personal digital (*Personal Digital Assistant*). A se vedea **PDA**.

asistență contextuală (*context-sensitive help*). Formă de asistență directă utilizată în mediile cu interfață grafică, în care programul folosit de utilizator afișează informații referitoare la comanda curentă sau la operația în curs de desfășurare. Poate fi implementată folosind *tooltip*-uri, schimbarea formei cursorului (semn de întrebare), urmată – după execuția unui click – de

aparitia *help*-ului contextual etc. A se vedea și **interfață grafică** și **tooltip**.

Ask.com. Denumit încă *Ask Jeeves* în Marea Britanie, este un motor de căutare general, creat în 1996 în California, SUA, de Garret Gruener și David Warthen (<http://www.ask.com/>). Oferă servicii localizate în câteva țări (Franța, Marea Britanie, Italia, Spania, Germania, Olanda, Japonia). A se vedea și **motor de căutare**.

asociativitate (*associativity*). În matematică, este una dintre proprietățile unei operații binare. O astfel de operație se numește asociativă dacă într-o expresie care conține de două sau mai multe ori operatorul corespunzător, ordinea operațiilor nu contează atâta vreme cât ordinea operandilor nu se schimbă. Astfel, o operație binară * definită pe mulțimea M este asociativă dacă pentru orice elemente x, y și z incluse în M:

$$(x * y) * z = x * (y * z).$$

De exemplu, adunarea numerelor reale este asociativă, deoarece:

$$(x + y) + z = x + (y + z).$$

De asemenea, înmulțirea numerelor reale este asociativă. În schimb, scăderea, împărțirea și ridicarea la putere sunt operații neasociative. A se vedea și **comutativitate** și **operator**.

asociativitatea operatorilor (*operator associativity*). Caracteristică a operatorilor care determină ordinea de evaluare într-o expresie atunci când doi operatori succesivi au aceeași precedență. Sunt

posibile două cazuri: asociativitate de la stânga la dreapta și asociativitate de la dreapta la stânga. Asociativitatea celor mai mulți operatori este de la stânga la dreapta. A se vedea și **operator** și **precedența operatorilor**.

asociere (*binding*). Asocierea unor valori simbolurilor utilizate într-un program. Se poate face în timpul compilării (asociere statică) sau în timpul execuției programului (asociere dinamică). A se vedea și **asociere în timpul compilării** și **asociere în timpul execuției**.

asociere în timpul compilării (*compile time binding*). Numită și *asociere statică* (*static binding*), este procesul prin care simbolurile utilizate într-un program sunt rezolvate (*i.e.*, li se atribuie valori) în timpul compilării. Această abordare determină creșterea timpului necesar compilării programului, dar și scăderea celui de execuție (*runtime*). A se vedea și **asociere, asociere în timpul execuției și runtime** (1, 2).

asociere în timpul execuției (*runtime binding*). Numită și *asociere dinamică* (*dynamic binding*), este procesul prin care simbolurile utilizate într-un program sunt rezolvate (*i.e.*, li se atribuie valori) în timpul execuției (*runtime*). Această abordare determină creșterea timpului de execuție a programului respectiv. A se vedea și **asociere, asociere în timpul compilării și runtime** (1, 2).

ASP. Prima platformă de programare pentru server dezvoltată de Microsoft, utilizată pentru crearea paginilor Web dinamice. Astăzi, este înlocuită de ASP.NET. A se vedea și **ASP.NET**.

ASP.NET (*Active Server Pages*). Instrument de programare pentru crearea paginilor Web dinamice, dezvoltat de Microsoft. Orice limbaj de scripting compatibil cu standardul Active Scripting poate fi folosit în ASP. Limbajul de programare clasic utilizat în ASP este VBScript. ASP.NET este inclus în platforma .NET, beneficiind de avantajele acesteia (*e.g.*, compatibilitate cu peste 20 de limbaje). A se vedea și **ASP**.

atac (*attack*). Acțiune rău intenționată întreprinsă de o persoană în scopul de a obține acces neautorizat la diverse tipuri de date existente pe un sistem sau pentru a provoca refuzul unui serviciu oferit de acesta.

Atanasoff, John Vincent (1903–1995). Fizician american, cunoscut pentru crearea primului calculator electronic digital, denumit Atanasoff-Berry Computer (ABC). ABC, conceput în 1937 și testat cu succes în 1942, a fost proiectat cu scopul de a rezolva sisteme de ecuații liniare, nefiind un calculator programabil. A se vedea și **calculatorul Atanasoff-Berry**.

ATM (*Automated Teller Machine*). Dispozitiv computerizat care le furnizează clienților unei instituții financiare posibilitatea de a efectua

tranzacții financiare într-un spațiu public fără intervenția directă a unui angajat al acesteia. Scoțianul James Goodfellow este creditat pentru inventarea ATM-ului.

Utilizatorul este identificat pe baza unei cărți de credit (*credit card*) și a unui număr de identificare personal (*Personal Identification Number*, PIN). Cardul conține un *chip* care include un număr unic de identificare a cardului, precum și o serie de informații de securitate (e.g., data expirării, CVV – *Card Verification Value*) utilizate pentru prevenirea fraudelor. Folosind un ATM, utilizatorii pot avea acces la propriul cont bancar și pot efectua verificarea soldului, retrageri de numerar, plăți (e.g., plata facturilor), împrumuturi etc.

Platformele software utilizate în mod obișnuit în ATM-uri sunt OS/2, Microsoft Windows și Linux. A se vedea și **Goodfellow, James**.

atribut (*attribute*). Informație suplimentară atașată unui element folosit în HTML, XML sau în alt limbaj de marcare. Valorile atributului sunt stabilite în cadrul definiției tipului de document sau pot fi specificate prin construcția sintactică `atribut = "valoare"`, plasată în marcajul de început al unui element. A se vedea și **DTD** și **element**.

atribut de date (*data attribute*). 1. Caracteristică a unei date (e.g., lungime, valoare, metodă de reprezentare). 2. În teoria bazelor de date relaționale, informație structurală referitoare la date, care descrie

contextul și semnificația lor. A se vedea și **bază de date relațională** și **date**.

atribut XML (*XML attribute*). Componentă a unui document XML, furnizează informații suplimentare despre o apariție particulară a unui element într-un astfel de document, fiind inclus în interiorul marcajului de început, imediat după numele acestuia. Un element poate să conțină niciunul, unul sau mai multe atribute. Orice atribut trebuie precizat printr-un nume (*nume-atribut*) și o valoare (*valoare-atribut*), utilizându-se următoarea sintaxă: `nume-atribut = "valoare-atribut"`.

În scrierea și utilizarea atributelor trebuie respectate următoarele reguli: valorile acestora vor fi incluse obligatoriu între ghilimele; nu sunt acceptate atribute fără valoare; atributele sunt *case sensitive*; dacă un element conține mai multe atribute, ordinea în care acestea sunt precizate nu prezintă importanță.

De multe ori, aceeași informație poate fi precizată în două moduri: prin utilizarea elementelor-copil și, respectiv, prin utilizarea atributelor. În acest caz, este recomandată utilizarea elementelor-copil, pentru ca structurile de date XML să poată fi descrise mai ușor. Specificațiile XML precizează două atribute predefinite: `xml:space` și `xml:lang`. Primul este utilizat pentru a indica modul în care sunt tratate spațiile albe, având ca valori posibile `default` (implicit) și `preserve`. A doua valoare (`xml:space="preserve"`) implică

păstrarea spațiilor albe. Al doilea atribut predefinit precizează limba utilizată pentru alcătuirea conținutului textual al elementului. A se vedea și **XML**.

autentificare (*authentication*). Acțiunea prin care se verifică dacă un utilizator are permisiunea de a accesa o resursă sau de a executa o operație. În mod obișnuit, autentificarea se realizează folosind un nume de utilizator și o parolă. A se vedea și **autentificare Web**.

autentificare Web (*Web authentication*). Cunoscută și ca autentificare HTTP, utilizează o schemă de tip *solicitare/răspuns* pentru autentificarea utilizatorilor (inclusă în HTTP/1.1), în scopul de a permite accesul acestora la o zonă protejată a unui site sau a unei aplicații Web. Zona este, de fapt, un subdirector inclus în directorul-rădăcină al site-ului/aplicației sau chiar directorul-rădăcină în întregime.

Pentru autentificarea Web sunt utilizate metodele *Basic* și *Digest*, care, pentru serverul Web *Apache*, sunt implementate în modulele *mod_auth* și, respectiv, *mod_auth_digest*.

Una dintre deficiențele metodei *Basic* este aceea că numele de utilizator și parola sunt complet expuse în rețeaua pe care o traversează. Orice parte intermediară care poate intercepta sau vedea conținutul comunicării poate recupera datele de identificare. Din acest motiv, specificațiile HTTP recomandă utilizarea metodei *Basic* numai în situația în care cerințele de

securitate ale aplicației sunt minime.

Metoda *Digest* utilizează principiile criptografiei pentru a evita transmiterea în rețea a parolelor. În locul transmiterii unei parole, browserul dovedește serverului că se află în posesia ei. În acest scop, clientul creează și trimite serverului un mesaj *digest* (rezumat), cunoscut și ca mesaj *hash* (încurcat), utilizând parola și o valoare furnizată de server. Serverul reface mesajul *digest* cunoscând valoarea furnizată browserului și parola (asociată utilizatorului și stocată pe server). Dacă rezultatele sunt identice, serverul va ști că browserul cunoaște parola corectă. Algoritmii folosiți pentru calculul mesajului *digest* se bazează pe așa-numitele *funcții cu un singur sens*. O astfel de funcție realizează o transformare relativ simplă, dar extrem de greu de realizat în sens invers. Astăzi, sunt utilizați cu precădere doi algoritmi *digest*. Unul este Message Digest 5 (MD5), creat de Ron Rivest în 1991, iar celălalt Secure Hash Algorithm (SHA), dezvoltat de *National Institute of Standards and Technology* (din Statele Unite). A se vedea și **autentificare, MD5 și SHA**.

autostrada informațională (*information highway*). Concept lansat în anul 1992, desemnează Internetul și infrastructura sa, văzute ca un ansamblu de servicii care oferă cantități foarte mari de informații și realizează transferul lor către utilizatori. A se vedea și **Internet**.

authoring tool (AT). Pachet software folosit de dezvoltatori pentru a crea și împacheta un conținut care va fi livrat utilizatorilor finali. O altă definiție a instrumentelor AT este următoarea: un AT este un program care le permite utilizatorilor să scrie aplicații hipertext sau multimedia. Instrumentele de acest tip oferă posibilitatea de a crea aplicații finale prin stabilirea unor conexiuni între obiecte de diverse tipuri (*e.g.*, secvențe de text, imagini, clipuri video). Prin definirea unor relații între obiecte și prin stabilirea unei ordini de apariție a acestora, autorii (*i.e.*, cei care utilizează AT-urile) pot produce aplicații atractive și utile. Cele mai multe AT-uri suportă și limbaje de scripting proprii, care permit crearea unor aplicații sofisticate.

Instrumentele AT au numeroase întrebuințări, însă cel mai adesea sunt utilizate pentru a crea module pentru *e-Learning*. Acestea sunt realizate respectând o serie de standarde internaționale, de exemplu SCORM.

Instrumentele AT pot fi clasificate, în funcție de gradul lor de complexitate, astfel: instrumente bazate pe șabloane (*template-based tools*) – conținutul este creat folosind șabloane și casete de dialog. (practic, nu necesită o pregătire prealabilă a celui care le utilizează); instrumente generale (*general authoring tools*) – în această categorie sunt incluse majoritatea instrumentelor AT, cu ajutorul lor fiind creată cea mai mare parte a conținutului utilizat în *e-Learning* (necesită o pregătire

prealabilă a utilizatorului, dar îi oferă acestuia posibilitatea de a-și manifesta creativitatea); instrumente pentru simulări (*high-end simulation tools*) – sunt utilizate pentru a produce simulări multimedia sofisticate, care includ conținut video și audio (utilizatorului îi sunt necesare câteva săptămâni sau luni pentru a învăța să le utilizeze, dar gradul de libertate în ceea ce privește creativitatea în utilizarea acestora este foarte mare).

În selecția unui instrument AT, creatorii de conținut vor trebui să țină seama de: tipul conținutului care va fi creat (*e.g.*, curs, tutorial, test); modul de prezentare (*e.g.*, Web, CD/DVD, fișier executabil); grupul-țintă căruia i se adresează conținutul creat (*i.e.*, nivelul de pregătire al studenților); respectarea standardelor pentru *e-Learning* (*e.g.*, SCORM). De asemenea, trebuie avute în vedere cunoștințele necesare pentru utilizarea instrumentelor, prețul acestora și existența licențelor. A se vedea și **e-Learning** și **SCORM**.

Authorization. Unul dintre anteturile cererii HTTP. Este folosit în situația în care accesul la resursă necesită autentificare, după primirea unui răspuns de tip 401 (Unauthorized). A se vedea și **401** și **cerere HTTP**.

Authorware. Instrument de tip CAT (*Course Authoring Tool*) utilizat pentru crearea aplicațiilor (materialelor) pentru *e-Learning* bogate în efecte vizuale, care pot fi livrate pe Web, pe CD/DVD sau direct în rețelele corporațiilor. Proiectat inițial (de

Macromedia) pentru a realiza materiale CBT (*Computer-Based Training*) pe CD-uri, a fost adaptat pentru a crea module care – folosind o serie de *plugin*-uri proprietare – pot rula în browserele Web. Deoarece dispune de un limbaj de scripting propriu, oferă posibilitatea creării unor simulări cu interactivitate ridicată, care includ și părți audio și video. *Authorware* le oferă utilizatorilor un număr de obiecte predefinite pentru diverse etape ale unui curs, care salvează o parte importantă din timpul alocat creării acestuia. Pentru a exploata toate posibilitățile și avantajele oferite de *Authorware*, sunt necesare o pregătire specializată și exersarea acestui instrument.

Utilizând *Authorware*, pot fi create aplicații/cursuri (*Application*) și teste (*Quiz*). Programul este disponibil și în varianta de evaluare pentru 30 de zile și poate fi descărcat – ca și alte produse Adobe Systems – după crearea unui cont de utilizator pe site-ul companiei. A se vedea și **Course Authoring Tool**.

autoritate de certificare (*certification authority*). Companie care semnează

certIFICATE cu cheie publică, acestea din urmă garantând identitatea serverului Web. Pentru a pregăti un server Web să accepte conexiuni securizate, este necesar ca administratorul să creeze un certificat cu cheie publică (numit și certificat digital), care trebuie să fie semnat de o autoritate de certificare. Browserele actuale recunosc certificate semnate de principalele autorități de certificare (e.g., VeriSign, GeoTrust, Comodo). A se vedea și **certificat digital** și **semnătură digitală**.

avatar (*avatar*). Reprezentare abstractă a unui utilizator al unei aplicații de tip mesagerie instantanee sau al unei lumi virtuale (e.g., *Second Life*). A se vedea și **Second Life**.

AVI (*Audio Video Interleave*). Format multimedia de tip container pentru Windows, introdus de Microsoft în 1992, având extensia de fișier *.avi*. Permite includerea într-un singur fișier a unor secvențe audio și video. Fișierele AVI suportă *streaming*-ul audio și video multiplu. Tipurile MIME utilizate sunt `video/avi`, `video/msvideo` și `video/x-msvideo`. A se vedea și **MIME**.

B

Babbage, Charles (1791–1871). Inginer mecanic și matematician englez, numit „părintele informaticii”, deoarece a fost primul care a imaginat – în anul 1837 – o mașină de calcul programabilă, denumită *mașina analitică* (*analytical machine*). Mașina nu a putut fi construită în vremea aceea din cauza limitărilor tehnologice. După schițele sale, în anul 1991 a fost construită o mașină de calcul perfect funcțională. A se vedea și **Lovelace, Ada**.

backbone (*coloană vertebrală*). 1. Termen care desemnează ansamblul principalelor trasee de date (e.g., comerciale, guvernamentale, academice) și routere interconectate în Internet, traversând continentele și oceanele. A se vedea și **Internet**. 2. Firele care asigură transportul pentru cea mai mare parte a traficului dintr-o rețea (e.g., o magistrală în cazul unui LAN). A se vedea și **LAN**.

backdoor (*ușă din spate*). Gaură de securitate într-un sistem de calcul, realizată în mod deliberat. Motivația creării acestor „găuri” nu este întotdeauna negativă: de exemplu,

unele sisteme de operare sunt distribuite cu conturi privilegiate pentru tehnicienii din service.

backlink (*legătură înapoi*). Denumită și *incoming link*, este o legătură către site-ul propriu, putând fi plasată atât în paginile acestuia, cât și pe site-uri externe. În SEO, numărul de *backlinks* este un indicator al popularității site-ului/paginii Web (cel mai important rămânând totuși *PageRank*-ul). A se vedea și **hiperlegătură**, **inbound link**, **link building** și **SEO**.

backtracking (*revenire*). Tehnică folosită în programare pentru rezolvarea problemelor care îndeplinesc simultan următoarele condiții: (1) soluția lor poate fi scrisă sub forma unui vector $S = x_1, x_2, \dots, x_n$, cu $x_1 \in A_1, x_2 \in A_2, \dots, x_n \in A_n$; (2) A_1, A_2, \dots, A_n sunt mulțimi finite, elementele lor aflându-se într-o relație de ordine bine stabilită; (3) nu se dispune de o altă metodă, mai rapidă, de rezolvare. Cei care întâlnesc o problemă de acest tip sunt tentați să genereze toate elementele produsului cartezian $A_1 \times A_2 \times \dots \times A_n$, testând fiecare element pentru a verifica dacă este soluție. Dacă se rezolvă problema în acest mod, timpul de execuție este foarte mare, algoritmul neavând valoare practică. Tehnica *backtracking* are la bază următorul principiu: se construiește soluția pas cu pas, x_1, x_2, \dots, x_n ; dacă se constată că, pentru o valoare aleasă, nu se poate ajunge la o soluție, se renunță la valoarea respectivă și se reia căutarea din punctul în care s-a rămas.

Tehnica *backtracking* are ca rezultat găsirea tuturor soluțiilor problemei. Dacă se dorește obținerea unei singure soluții, se poate forța oprirea algoritmului atunci când aceasta a fost găsită. Termenul *backtrack* a fost inventat în anii 1950 de matematicianul american D.H. Lehmer. Limbajul de procesare a șirurilor SNOBOL (1962) a fost primul care a implementat o facilitare de tip *backtracking*.

back up (*salvare de siguranță*). Procesul prin care se realizează o copie de siguranță. A se vedea și **backup** și (a) **restaura**.

backup (*copie de siguranță*). Copia unui program, a unei baze de date, a unui disc etc., făcută cu scopul de a restaura originalul după eventuala pierdere sau degradare a datelor. Unele aplicații realizează automat copii de siguranță, în timp ce altele fac acest lucru la solicitarea utilizatorilor. A se vedea și (a) **restaura**.

Backus, John (1924–2007). Informatician american, liderul echipei care a creat limbajul de programare Fortran, pentru care a primit în 1978 premiul Turing.

Baidu. Cel mai important motor de căutare chinezesc pentru site-uri Web, imagini și fișiere multimedia (<http://www.baidu.com/>). A se vedea și **motor de căutare**.

bancă de date (*data bank*). Depozit de informații despre unul sau mai multe subiecte, organizat într-o modalitate care facilitează regăsirea informațiilor,

atât local, cât și de la distanță. În multe cazuri, conceptul este sinonim cu cel de bază de date. A se vedea și **bază de date**.

banner (*manșetă*). Denumit și *Web banner*, este o secțiune dintr-o pagină Web care conține o reclamă (de obicei plătită) realizată prin intermediul unei imagini sau a unui obiect multimedia creat folosind diverse tehnologii (e.g., *Java*, *Flash*, *Silverlight*). *Bannerul* include o legătură către site-ul al cărui proprietar a solicitat realizarea reclamei. În multe cazuri, *bannerele* sunt livrate de un server central (denumit *ad server*). Dimensiunile *bannerelor* livrate prin intermediul unui astfel de server au fost standardizate de IAB (*Interactive Advertising Bureau*).

Baran, Paul (n. 1926). Inginer american de origine poloneză, unul dintre cei trei inventatori ai rețelelor cu comutare de pachete, alături de Donald Davies și Leonard Kleinrock. Ca angajat al RAND Corporation, firmă însărcinată cu construcția unei rețele de comunicații distribuite, a publicat un memorandum intitulat *Despre comunicațiile distribuite: introducerea în rețeaua de comunicații distribuite*, în care și-a expus viziunea asupra construcției unei rețele cu comutare de pachete. În anul 1969, proiectul început de Paul Baran a devenit realitate, iar noul sistem a fost denumit ARPANET. A se vedea și ARPANET și **comutare de pachete**.

Base Class Library (BCL). Bibliotecă standard disponibilă pentru toate

limbajele care folosesc .NET Framework și care furnizează un număr mare de funcții, extrem de utile programatorilor. Este actualizată la fiecare nouă versiune a .NET Framework. A se vedea și **.NET Framework**.

Bash. Shell (interpretor de comenzi) pentru UNIX, dezvoltat în 1987 de Brian Fox. Denumirea este un acronim pentru *Bourne-again shell*, deoarece *Bash* se inspiră din *Bourne shell*. A se vedea și **Bourne shell** și **shell**.

Bartik, Jean (n. 1924). Unul dintre programatorii calculatorului ENIAC. În 2008, doamna Bartik a fost – alături de Bob Metcalfe și Linus Torvalds – unul dintre laureații *Fellow Award*, acordat de Computer History Museum. A se vedea și **ENIAC**.

bază de cunoștințe (*knowledge base*). Abreviat KB, kb sau Δ, este un tip special de baze de date pentru managementul cunoașterii, care oferă mijloacele pentru colectarea computerizată, organizarea și regăsirea cunoștințelor. Bazele de acest tip sunt clasificate în două categorii: baze de cunoștințe care pot fi citite de calculator (stochează cunoștințe în formate accesibile sistemelor de calcul; sunt utilizate în inteligența artificială, *e.g.*, ca părți ale unor sisteme expert) și baze de cunoștințe care pot fi citite de om (stochează cunoștințe în formate accesibile omului; *e.g.*, informații pentru depanare, diverse articole, manuale pentru utilizatori, liste FAQ). A se

vedea și **FAQ, inteligență artificială și sistem bazat pe cunoștințe**.

bază de date (*database*). Una sau mai multe colecții de date aflate în interdependență, împreună cu descrierea datelor și a relațiilor dintre ele. Colecția de date reprezintă un ansamblu de date organizat după anumite criterii. Modelul bazei de date cuprinde descrierea tuturor entităților care compun baza de date, precum și relațiile dintre acestea. Descrierea datelor mai este denumită și *catalog de sistem, dicționar de date sau metadate* (*i.e.*, date despre date). A se vedea și **modelul bazei de date și SGBD**.

bază de date activă (*active database*). Bază de date care are capacitatea de a reacționa la stimuli. A se vedea și **bază de date**.

bază de date distribuită (*distributed database*). Bază de date implementată într-o rețea, părțile componente ale acesteia fiind distribuite în diferite noduri ale rețelei. O astfel de bază poate fi rezidentă într-o rețea de servere în Internet, în intranetul sau extranetul unei corporații sau în alte rețele. Distribuirea bazei de date este utilizată atunci când există un trafic mare pentru actualizarea și regăsirea datelor, putând determina creșterea semnificativă a performanțelor. A se vedea și **bază de date**.

bază de date relațională (*relational database*). Bază de date alcătuită din una sau mai multe *tabele*. Un tabel include *înregistrări* sau *rânduri*. O înregistrare este alcătuită din unul sau mai multe

câmpuri, numite și *coloane* sau *atribute*. Dacă valoarea unui câmp se determină pe baza valorilor altor câmpuri, acesta se numește *câmp calculat*. Un tabel nu poate avea două rânduri identice. Între tabelele componente ale unei baze de date se pot stabili relații.

O înregistrare va fi identificată prin valoarea coloanelor sale. O coloană sau o combinație de coloane care permite identificarea înregistrărilor se numește *cheie candidată*. Dintre cheile candidate se selectează o *cheie primară*. O cheie a unui tabel poartă numele de *cheie externă* dacă este cheia primară a unui alt tabel.

În modelul relațional, implementarea unei baze de date implică parcurgerea *obligatorie* a trei etape: proiectarea bazei de date (în acest scop este utilizat un procedeu cunoscut sub numele de *modelare entitate-relație*); normalizarea bazei de date; specificarea tipului de date pentru fiecare coloană (e.g., caracter, întreg, zecimal, dată și oră și binar). A se vedea și **cheie alternativă**, **cheie candidată**, **cheie primară** și **SGBDR**.

bcc (*blind carbon copy*, *copie la indigo invizibilă*). Denumită și *blind courtesy copy* (*copie de informare invizibilă*), este o copie a unui mesaj de *e-mail* trimisă unuia sau mai multor destinatari, astfel încât destinatarii „oficiali” să nu cunoască acest lucru. A se vedea și **cc** și **mesaj de e-mail**.

BCNF (*Boyce-Codd Normal Form*, *forma normală Boyce-Codd*). A se vedea **normalizare**.

Bebo. Una dintre cele mai populare rețele sociale *online*, poate fi accesată la <http://www.bebo.com/>. A se vedea și **rețea socială online**.

Behlendorf, Brian (n. 1973). Programator american, personalitate importantă a mișcării *open source*, creatorul serverului *Web Apache* (<http://brian.behlendorf.com/>). A înființat în 1993 Organic, Inc., prima companie specializată în crearea site-urilor *Web*. Începând din anul 2003 face parte din *boardul* companiei Mozilla. A se vedea și **Apache**.

Bell, Daniel (n. 1919). Sociolog, autor și profesor emerit la Universitatea Harvard. Printre altele, a scris lucrarea *The end of ideology: On the Exhaustion of Political Ideas in the Fifties* (publicată pentru prima oară în 1960), în care sugerează că vechile ideologii umaniste care au luat naștere în secolul al XIX-lea și în prima jumătate a secolului al XX-lea sunt epuizate și că noi ideologii le vor lua locul.

Bell a fost primul sociolog care a scris despre impactul social al comunicațiilor digitale. Acesta a prezis că multe consecințe sociale majore vor proveni din două evoluții independente: miniaturizarea electronicii și inventarea circuitelor optice capabile să susțină fluxuri mari de informație în cadrul rețelelor de calculatoare. Anticipând democratizarea comunicațiilor electronice, acesta a afirmat că „organizarea socială a noii tehnologii de comunicare este cea mai importantă problemă a societății postindustriale”.

Berners-Lee, Tim (n. 1955). Fizician și informatician englez, inventatorul Web-ului. Acesta, împreună cu Robert Caillau, ambii lucrând la CERN, au propus în 1989 un sistem informatic distribuit având ca scop principal accesul rapid la documentație. Între anii 1989 și 1991, sunt puse bazele serviciului World Wide Web și create protocolul HTTP și limbajul HTML.

Tim Berners-Lee a scris primul browser (*WorldWideWeb*) și primul server Web (denumit *httpd*, acesta fiind acronimul pentru *HyperText Transfer Protocol Daemon*), elaborând totodată și specificațiile inițiale pentru URI-uri, HTTP și HTML. Primul site Web a fost realizat la CERN și plasat *online* în data de 6 august 1991. În anul 2002, lui Tim Berners-Lee i-a fost decernat *Japan Prize* pentru inventarea Web-ului, iar în 2003 i-a fost acordat titlul de cavaler al Marii Britanii. Tim Berners-Lee a contribuit la înființarea Consorțiului World Wide Web (W3C), fiind și director al acestuia. A se vedea și **CERN** și **Consorțiul World Wide Web**.

beta (*beta*). Ceea ce se referă la un produs software aflat în versiune beta. Un produs software aflat în această etapă de dezvoltare este testat în general de un număr limitat de utilizatori externi (nu sunt membri ai organizației care îl dezvoltă). A se vedea și **alfa**.

bibliotecă (*library*). În programare, este o colecție de subrutine sau clase utilizate pentru dezvoltarea software-

ului. În mod obișnuit, bibliotecile nu sunt executabile.

Bibliotecile pot fi statice și dinamice. O bibliotecă statică este copiată în aplicația-țintă de către compilator sau de către editorul de legături. În schimb, o bibliotecă dinamică este legată în aplicația-țintă numai la execuție (*runtime*). Un tip particular de bibliotecă dinamică sunt *plug-in*-urile.

bibliotecă de funcții (*function library*). A se vedea **bibliotecă**.

bibliotecă runtime (*runtime library*). În programare, fișier care conține rutine predefinite folosite în timpul execuției unui program (*runtime*). Astfel de biblioteci includ adesea funcții de intrare și ieșire sau funcții pentru managementul memoriei. Bibliotecile *runtime* sunt utilizate adesea de limbajele de nivel înalt. A se vedea și **bibliotecă** și **runtime** (1).

big endian (*big endian*). Metodă de stocare a numerelor întregi, în care primul octet se află pe prima poziție. Dintre microprocesoarele care utilizează metoda pot fi enumerate Motorola și Sun SPARC. A se vedea și **little endian**.

binary digit (*cifră binară*). Una dintre cele două cifre ale sistemului de numerație binar (*i.e.*, 0 sau 1). A se vedea și **digit**.

Bing. Fost *Live Search* (2007), *Windows Live Search* (2006) și *MSN Search* (1998), este un motor de căutare pe Web dezvoltat de Microsoft,

denumit astfel începând din luna mai a anului 2009 (<http://www.bing.com/>).

BIOS (*Basic Input/Output System*). Simultan, standard de facto care definește interfața *firmware* pentru calculatoarele compatibile IBM PC și un set de rutine software. Este primul cod rulat la pornirea PC-ului, rolul său fiind de a identifica, testa și analiza dispozitivele incluse în PC (placa grafică, hard-disk-ul etc.), în scopul de a-l aduce într-o stare în care software-ul de sistem (*i.e.*, sistemul de operare) stocat pe un mediu corespunzător (*e.g.*, hard-disk, CD) poate fi încărcat și executat, preluând controlul asupra calculatorului. Procesul descris este cunoscut ca *boot*. A se vedea și **boot** și **firmware**.

BISON (*Binary Interchange Standard and Object Notation*). Protocol proiectat pentru transmiterea de date structurate spre HTTP, dezvoltat de Karl Jäger. Poate fi utilizat pentru apelul procedurilor la distanță, pentru schimbul de informații, dar și ca format pentru serializarea obiectelor.

Unul dintre scopurile avute în vedere la proiectarea BISON a fost acela de a oferi un format binar liber, care să fie utilizat pentru codificarea datelor cu dimensiuni mari. De exemplu, pentru trimiterea unui număr pe 32 de biți, în XML sunt necesari până la 10 octeți, în vreme ce într-un format binar sunt suficienți numai patru. Al doilea scop a fost utilizarea cât mai eficientă a spațiului alocat descrierii structurii datelor. Astfel, un document XML

conține nu numai date, ci și descrierea structurii lor, aceasta din urmă ocupând un spațiu însemnat, chiar mai mare decât cel ocupat de date. BISON rezolvă parțial problema spațiului ocupat de descrierea structurii datelor, fiind un format care utilizează mai eficient spațiul, dar rămâne în esență un format text auto-descriptiv, ca și XML.

Formatul mesajelor schimbate prin BISON este numit BMF (*BISON Message Format*). Orice mesaj BMF începe cu numărul „magic” 666D62h (șirul „BMF” în notația *little-endian*), utilizat pentru a identifica un mesaj valid și pentru a recunoaște dacă acestuia i-a fost aplicată codificarea BMF.

BISON utilizează șapte tipuri de date: *boolean*, *integer*, *float*, *string*, *array*, *object* și *stream*. Fiecare tip de dată este identificat prin valoarea unui singur octet fără semn, numit *id-byte*. De exemplu, *id-byte*-ul tipului *object* este 11h.

bit (*bit*). Prescurtare de la *binary digit* (*cifră binară*), este cea mai mică unitate de informație pe care o poate manipula calculatorul. Un bit exprimă una dintre cifrele binare 0 sau 1. Un grup de 8 biți (denumit *byte* sau *octet*) poate reprezenta diverse tipuri de informații (*e.g.*, un caracter). A se vedea și **byte**.

bit rate (*rată de biți*). Denumită și *data transfer rate* (*rată de transfer a datelor*) sau *transfer rate* (*rată de transfer*), reprezintă numărul mediu de biți, caractere sau blocuri transferate în

unitatea de timp între echipamente în sistemele de transmisie a datelor. Se măsoară în multipli ai unităților *bit per second* (*biți pe secundă, bps*) și *byte per second* (*octeți pe secundă, Bps*). A se vedea și **bps**, **kbps**, **Mbps** și **Gbps**.

black hat SEO (*black hat SEO*). Metodă incorectă de optimizare a unui site/a unei pagini pentru aducerea sa pe o poziție cât mai bună în lista de rezultate returnată de motoarele de căutare. Se referă la tehnici prin care motorul de căutare „vede” un conținut, iar vizitatorii altul. Motoarele de căutare penalizează sau chiar elimină din indexul lor aceste site-uri/pagini. Astfel, în februarie 2006, *Google* a șters două site-uri din baza sa de date – cele ale companiilor BMW Germania și Ricoh Germania, pentru că au folosit practici incorecte. Ambele companii și-au cerut scuze și au remediat problemele, fiind reincluse ulterior în indexul *Google*. A se vedea și **SEO** și **white hat SEO**.

blended learning (*învățarea combinată*). A se vedea **învățarea combinată**.

blog (*blog*). Scurtătură pentru *weblog* sau *web log*. Blogul este un site Web reprezentând de cele mai multe ori un jurnal public *online* al unui autor. Majoritatea blogurilor includ mijloace de postare a comentariilor vizitatorilor. Astfel, blogul reprezintă un instrument de facilitare a interacțiunilor umane.

Termenul *weblog* a fost introdus în 1997 de Jorn Barger, al cărui site

(<http://www.robotwisdom.com/>) a constituit un model, influențând într-o măsură importantă primele bloguri. Primele servicii software de *blogging* au fost oferite utilizatorilor spre sfârșitul anilor 1990.

Prin intermediul unui blog se poate face aproape orice, de la publicarea fotografiilor personale sau de familie până la promovarea unei afaceri. Detaliat, scopurile obișnuite în care se utilizează blogurile sunt: informarea membrilor familiei și prietenilor despre experiențele personale; exprimarea opiniilor despre diverse evenimente (*e.g.*, politice, din domeniul tehnic, artistic, educațional etc.); crearea unor comunități în jurul unor subiecte de interes; promovarea unei afaceri – se realizează prin includerea regulată de conținut nou referitor la serviciile și produsele oferite, în scopul atragerii de noi clienți și păstrării celor deja câștigați; găzduirea de mesaje publicitare – unele companii cumpără spațiu publicitar pe blogurile care au un număr mare de vizitatori.

Indiferent de scopurile în care sunt utilizate, blogurile au o serie de caracteristici comune: includ legături – blogurile conțin adesea legături către alte bloguri sau site-uri Web al căror conținut este relevant pentru articolul în care sunt incluse; oferă interactivitate – în mod obișnuit, blogurile solicită *feedback* de la vizitatori, aceștia având posibilitatea de a posta comentarii (numite și *răspunsuri*) la fiecare intrare; includ conținut multimedia –

majoritatea blogurilor sunt bazate pe text, dar un număr destul de ridicat oferă posibilitatea includerii de conținut multimedia (*i.e.*, grafic, video, audio).

În componența tuturor blogurilor sunt incluse ca elemente universale: intrările (*entries*) sau *post*-urile (*posts*) – reprezintă elementele principale de conținut. Sunt actualizate frecvent și afișate în ordine inversă celei în care au fost publicate; arhivele (*archives*) – sunt colecții organizate de intrări vechi. Deoarece toate intrările sunt datate, criteriul de organizare a acestora va fi unul calendaristic (*e.g.*, pe ani și luni). Un blog va afișa câteva intrări curente (în mod obișnuit, numărul lor poate fi stabilit de utilizator) și o listă a intrărilor arhivate, plasată într-o parte a paginii Web.

Blogurile includ în mod obișnuit (dar, spre deosebire de intrări și arhive, nu obligatoriu) următoarele elemente: *blogroll* – este o listă de legături către alte bloguri; lista poate fi gestionată manual sau prin intermediul *software*-ului pentru *blogging*; categorii – reprezintă subiecte folosite pentru a organiza conținutul unui blog; categoriile sunt determinate de *blogger* și reflectă în mod obișnuit natura și scopul blogului; comentarii – sunt răspunsuri ale cititorilor la intrările incluse în blog. În cele mai multe cazuri, *software*-ul pentru *blogging* le oferă *blogger*ilor posibilitatea de gestionare a comentariilor; *permalink* – reprezintă o legătură permanentă (*permanent link*) folosită de *blogger* pentru a stabili o

legătură directă către o intrare a unui alt blog; canale RSS/Atom – denumite și canale de știri sau *feed*-uri, acestea le oferă cititorilor posibilitatea de a fi notificați de fiecare dată când pe blog este adăugat conținut.

Blogurile sunt din ce în ce mai răspândite, utilizarea acestora aducând o serie de avantaje: rezolvă o varietate de probleme ridicate de publicarea pe Web; nu necesită învățarea unor limbaje sau tehnologii; le oferă utilizatorilor posibilitatea de a publica de oriunde și oricând, singura condiție care trebuie îndeplinită fiind ca aceștia să dispună de o conexiune la Internet; oferă un potențial ridicat de colaborare; serviciile care permit crearea blogurilor sunt gratuite.

Pentru a crea un blog, utilizatorii au la dispoziție următoarele două metode: folosirea unui serviciu de *blogging* (*blogging service*); folosirea unui *software* pentru *blogging* (*blogging software*). A se vedea și **blogger**, **blogging**, **post**, **serviciu de blogging** și **software pentru blogging**.

blogcast (*blogcast*). Combinație între *blog* și *podcast*, desemnează un articol într-un *blog* având atașat un fișier audio sau video disponibil pentru descărcare într-un MP3 player, browser Web sau cititor RSS. A se vedea și **blog** și **podcast**.

blogger (*blogger*). Autorul/proprietarul unui *blog*. A se vedea și **blog**.

Blogger. Serviciu de *blogging* oferit gratuit de compania Google (<http://www.blogger.com/>). *Blogger* a

fost creat de Pyra Labs și lansat în august 1999. În februarie 2003, Pyra Labs a fost cumpărat de Google. În anul 2004, Google a achiziționat *Picasa*, integrând în *Blogger* facilitatea acestei aplicații de partajare a fotografiilor și oferindu-le astfel utilizatorilor posibilitatea de a posta imagini în bloguri. În anul 2007, *Blogger* a obținut locul 16 din 50 într-un top al celor mai vizitate domenii, în ceea ce privește numărul de utilizatori unici. A se vedea și **blog** și **blogging**.

blogging (*blogging*). Activitatea de actualizare a unui blog, adică de publicare a unui conținut nou. A se vedea și **blog**, **blogger** și **post**.

Bloglines. Una dintre cele mai cunoscute aplicații Web 2.0, este un agregator, un director de *feed*-uri și un motor de căutare în *feed*-uri (<http://www.bloglines.com/>). A se vedea și **agregator**, **director de feed-uri**, **feed**, **motor de căutare în feed-uri** și **Web 2.0**.

blogosferă (*blogosphere*). Totalitatea blogurilor și a conexiunilor dintre acestea. Blogosfera românească include blogurile scrise în limba română. A se vedea și **blog** și **blogger**.

blogroll (*blogroll*). Listă de legături către alte bloguri inclusă într-un blog. A se vedea și **blog**.

Bluetooth. Set de specificații pentru o rețea *wireless* personală (*Personal Area Network*), bazată pe undele radio. Permite schimbul de informații între

diverse dispozitive (e.g., telefoane mobile, laptopuri, desktopuri, aparate foto digitale). Tehnologia *Bluetooth* poartă numele regelui viking Harald Bluetooth, care a unit Norvegia și Danemarca. La 20 mai 1998 a fost fondată gruparea *Bluetooth Special Interest Group* (SIG), cu rolul de a licita firmelor tehnologia *Bluetooth* și de a urmări evoluția acesteia. A se vedea și **PAN**.

BMP (*bitmap*). Format de fișier, denumit uneori **DBI** (*device-independent bitmap*), utilizat pentru stocarea imaginilor digitale *raster*, în special pe platformele Windows și OS/2. Informațiile de culoare stocate pot varia de la 1 bit la 24 de biți. De asemenea, formatul suportă codificare **RLE** (*run length encoding*) în Windows sau comprimarea folosind algoritmul Huffman în OS/2. Un fișier BMP include următoarele părți: *file header*, *bitmap info header*, paleta de culori (opțional) și datele imaginilor. Extensiile de fișier utilizate sunt *.bmp* și *.dib*, iar tipul MIME este *image/x-ms-bmp*. A se vedea și **formate raster**.

bogăția rețelelor (*the wealth of networks*). Stadiu în economia informațională denumit astfel de Yochai Benkler, profesor de drept la Harvard Law School, în lucrarea sa *The Wealth of Networks: How Social Production Transforms Markets and Freedom* (*Bogăția rețelelor: cum transformă producția socială piețele și libertatea*). În opinia lui Benkler, ca rezultat al

noilor realități, indivizii pot face mult mai multe lucruri în calitate de ființe sociale decât ca actori ai pieței, prin sistemul prețului/costului. Această orientare către informația din interiorul rețelelor are o serie de efecte importante asupra societăților democratice: o autonomie mai mare (care sporește capacitatea oamenilor de a face mai multe lucruri pentru ei înșiși; crește abilitatea indivizilor de a face mai multe lucruri alături de ceilalți, într-o atmosferă informală, necondiționată de sistemul economic de piață; sporește capacitatea indivizilor de a face mai multe lucruri în organizațiile formale care operează în domeniul afacerilor); se produce o reorientare de la sfera publică mediată de mass-media către o sferă publică inter-relaționată, în care este mai dificil să atragi atenția în Internet decât în cazul canalelor mass-media; o cultură critică și relații sociale în rețea fac actele culturale mult mai transparente și mai maleabile. Acest mod de comunicare le permite indivizilor nu numai să păstreze relații familiare și de prietenie mai strânse, ci și contacte mai bogate cu cei diferiți de ei, cu care nu ar fi în legătură în lipsa noilor mijloace de comunicare. A se vedea și **rețea socială**.

BookletCreator. Miniaplicație Web 2.0 care permite transformarea oricărui document PDF într-o broșură (<http://bookletcreator.com/>). Aceasta poate fi, ulterior, printată. De menționat că aplicația reordonează paginile documentului, ceea ce permite obținerea

oricărui tip de broșură dorit. A se vedea și **Web 2.0**.

bookmark (*semn de carte*). Copie – realizată pe calculatorul utilizatorului (în folderul *Bookmarks/Favorites* al browserului) sau *online*, prin intermediul unui serviciu de *social bookmarking*, – a unei adrese Web care desemnează o locație virtuală de unde se pot descărca ulterior resurse de orice tip (*e.g.*, pagini Web, fișiere PDF, fișiere multimedia). A se vedea și **social bookmarking** și **URL**.

bookmarking (*bookmarking*). Activitatea de creare a semnelor de carte (*bookmarks*). A se vedea și **bookmark**.

bookmarking social (*social bookmarking*). A se vedea **social bookmarking**.

Boole, George (1815–1864). Matematician și filosof englez, inventatorul algebrei/logicii booleene, ramură a matematicii aflată la baza calculatoarelor electronice moderne. A se vedea și **algebra booleană**.

boot (*a încărcă*). Procesul de pornire sau de repornire a unui calculator. În cadrul acestui proces, calculatorul execută un program care încarcă și lansează în execuție sistemul de operare. A se vedea și **BIOS** și **sistem de operare**.

botmaster (*bootmaster*). Persoană care controlează un grup de calculatoare *botnet*. A se vedea și **botnet**.

botnet (*botnet*). Termen utilizat pentru a descrie o rețea de calculatoare

controlate de hackeri prin intermediul unui troian sau al altui program de tip *malware*. Aceste calculatoare sunt manipulate prin IRC pentru a derula activități nocive. Calculatoarele „bot” acționează sub comanda unui singur hacker sau a unui grup de dimensiuni reduse de hackeri, denumit *botmaster*. După ce un botmaster creează o rețea *botnet*, aceasta poate fi utilizată în diverse scopuri: lansarea de atacuri DDoS (*Distributed Denial of Service*), transmiterea de mesaje de *e-mail spam*, răspândirea de aplicații *keylogger* și de *sniffere* de rețea (aplicații care detectează și interceptează traficul de rețea) și controlul de la distanță al calculatoarelor. A se vedea și **DoS**, **hacker** și **keylogger**.

Bourne shell (*shell-ul Bourne*). Primul *shell* (interpretor de comenzi) pentru sistemul de operare UNIX. A fost creat în 1978, la AT&T Bell Laboratories, de către Steve Bourne. A se vedea și **Bash** și **shell**.

bps (*bit per second, bit pe secundă*). Unitate de măsură a vitezei cu care un dispozitiv (*e.g.*, modem) poate transfera datele (*bit rate*). A se vedea și **bit rate**.

breadcrumbs (*firimituri de pâine*). Cunoscută și ca *breadcrumb trail*, este o modalitate de a facilita navigarea într-un site, dându-le utilizatorilor posibilitatea de a cunoaște permanent pagina curentă. Aceasta este indicată printr-un lanț de legături structurate ierarhic, plasat de obicei în partea superioară a paginii curente, pornind de la pagina principală (*homepage*).

Este recomandat ca numele fiecărui nivel ierarhic să coincidă (sau să fie asemănător) cu titlul paginii corespunzătoare, iar separarea nivelurilor să se realizeze prin intermediul unui simbol (*e.g.*, >). În mod obișnuit, o astfel de structură este similară cu următoarea: *Pagina principala>Secțiune>Subsecțiune...* De exemplu: *Acasa>Produse>Cosmetice>Sapun.*

Breadcrumbs este considerat un element important atât din punctul de vedere al utilizabilității, cât și în ceea ce privește SEO. A se vedea și **SEO**.

Bricklin, Dan (n. 1951). Programator american, coautor, împreună cu Bob Frankston, al primului program de calcul tabelar, programul *VisiCalc* (<http://www.danbricklin.com/>).

browse (*a rășfoi*). A parcurge o listă de fișiere (*e.g.*, dosar), o bază de date sau un site Web, cu scopul de a găsi un anumit articol sau alte lucruri considerate interesante de către cel care efectuează operația. Nu coincide cu căutarea, care se realizează folosind aplicații specializate (*e.g.*, motoare de căutare) și nu implică modificarea informațiilor accesate, ci numai observarea (*e.g.*, citirea) acestora. A se vedea și **bază de date**, **browser Web** și **dosar**.

browser WAP (*WAP browser*). Aplicație utilizată pentru accesarea site-urilor WAP prin intermediul terminalelor mobile. Furnizează toate serviciile de bază care sunt oferite de un browser Web, dar simplificate, pentru a

opera ținând seama de restricțiile impuse de natura terminalului (e.g., dimensiune redusă a ecranului de vizualizare). A se vedea și **site WAP** și **WAP**.

browser Web (*Web browser*). Denumit, pe scurt, browser sau navigator Web, este un software care rulează pe calculatorul clientului, utilizat în principal pentru afișarea interfeței cu utilizatorul a site-ului sau aplicației Web. Comunicarea dintre clientul Web (i.e., browser) și serverul Web se realizează folosind protocolul HTTP.

Utilizatorii au posibilitatea de a alege dintr-un număr mare de browsere oferite gratuit, rulând pe diverse platforme (Windows, UNIX/Linux, MacOS). Alături de *Internet Explorer*, *Mozilla Firefox*, *Safari*, *Netscape Communicator*, *Opera* și *Mozilla*, mai pot fi amintite și browserele *Maxthon*, *Konqueror*, *Amaya*, *SeaMonkey* și *Lynx* (acesta din urmă rulând în mod text).

Principalele caracteristici ale browserele Web actuale sunt: asigură suport pentru limbaje de marcare (e.g., HTML, XHTML), pentru unele limbaje de programare utilizate în paginile Web dinamice (e.g., JavaScript, VBScript) și pentru foi de stiluri folosite în scopul obținerii unor modalități de afișare unitară a conținutului (e.g., CSS, XSLT); oferă posibilitatea realizării listelor de adrese ale site-urilor favorite (numite, în funcție de browser, *Bookmarks* sau *Favorites*), precum și acces la istoricul navigării (*History*);

asigură suport pentru aplicații care extind funcționalitățile principale. O astfel de aplicație este cunoscută adesea sub numele de *plug-in*, fiind utilizată în scopul accesării unor tipuri speciale de date (e.g., conținut grafic vectorial în cunoscutele formate *Flash* și *SVG*); asigură securitatea transmiterii datelor confidențiale; îi oferă utilizatorului posibilitatea de a configura și controla unele caracteristici ale modului de afișare și navigare. A se vedea și **server Web** și **HTTP**.

BSD (*Berkeley Software Distribution*). Versiune de UNIX dezvoltată și distribuită de *Computer Systems Research Group* (University of California, Berkeley), între 1977 și 1995, având numeroși descendenți. A jucat un rol important în conectarea la Internet a instituțiilor academice. A se vedea și **FreeBSD** și **NetBSD**.

buclă (*loop*). Secvență de instrucțiuni inclusă într-un program, care se repetă de un număr prestabilit de ori sau până când o condiție își schimbă valoarea de adevăr. A se vedea și **buclă cu contorizare**.

buclă cu contorizare (*counting loop*). Secvență de instrucțiuni inclusă într-un program, care este executată repetat, de fiecare dată fiind incrementată o variabilă cu rol de contor. Când variabila atinge o valoare prestabilită, secvența nu va mai fi executată. A se vedea și **buclă**.

buclă infinită (*infinite loop*). Într-un program, o buclă din care nu se iese

prin mijloace obișnuite, din cauza unor erori logice (e.g., nu există o condiție de terminare a buclei sau, cu toate că o astfel de condiție există, aceasta nu poate fi atinsă). A se vedea și **buclă**.

buffer de comenzi (*command buffer*). Zonă de memorie temporară în care sunt păstrate comenzile lansate de utilizator de la tastatură. Îi permite acestuia să repete comenzile fără a le mai scrie încă o dată, să corecteze comenzi scrise eronat, să editeze comenzi introduse anterior pentru a schimba argumentele acestora sau să obțină lista comenzilor lansate.

buffer de ieșire (*output buffer*). Porțiune de memorie rezervată, folosită pentru stocarea temporară a informațiilor care vor fi utilizate de un dispozitiv de ieșire (e.g., imprimantă).

buffer de tastatură (*keyboard buffer*). Zonă de memorie temporară în care sunt păstrate cele mai recente caractere introduse de utilizator de la tastatură. Rolul său este de a reține caracterele care nu au fost încă prelucrate de sistem.

buffer de tipărire (*print buffer*). Zonă în memoria calculatorului sau imprimantei în care sunt păstrate datele înainte de a putea fi tipărite. Rolul acestuia constă în a prelua rapid informațiile care urmează a fi tipărite (cu viteză relativ mică, specifică imprimantei), astfel încât calculatorul să devină disponibil pentru alte activități.

bug (*deficiență, hibă*). Eroare de proiectare sau de programare care are drept consecință funcționarea incorectă a unui produs software. De multe ori, *bug*-urile sunt descoperite după lansarea pe piață a produsului. După eliminarea *bug*-urilor (depanare), utilizatorilor le este oferită o nouă versiune a produsului. A se vedea și (a) **depana**.

bullet (*marcaj de listă*). Simbol utilizat pentru evidențierea elementelor unei liste neordonate (e.g., cerc gol, cerc plin, pătrat). În (X)HTML, *bullet*-ul se creează folosind elementul ``, utilizat împreună cu ``. Ultimul marcaj creează o listă neordonată, iar primul un element în cadrul acesteia. Pentru specificarea tipului *bullet*-ului, se folosește atributul `type` al elementului ``, acesta având următoarele valori (care trebuie plasate între ghilimele): `disc` (cerc plin; este valoarea implicită), `circle` (cerc gol), `square` (pătrat). Trebuie precizat că se recomandă înlocuirea atributului `type` în (X)HTML cu foile de stil.

bus (*magistrală*). A se vedea **magistrală**.

Bush, Vannevar (1890–1974). Inginer american care a contribuit la dezvoltarea calculatoarelor analogice. În anul 1945, Vannevar Bush scrie un articol în *Atlantic Monthly* despre un dispozitiv foto-electro-mecanic imaginar, numit *Memex*, care poate realiza legături între documente. Ideile expuse în acest articol au fost utilizate mai târziu pentru

crearea Web-ului. A se vedea și **legătură** și **Memex**.

Business.com. Motor de căutare și director Web, fondat în anul 1999 de Jake Winebaum. Are ca țintă audiența din domeniul managementului corporațiilor (<http://www.business.com/>). A se vedea și **director Web** și **motor de căutare**.

buton (*button*). Este utilizat în cadrul formularelor (X)HTML incluse în paginile Web. Pentru a crea un buton, atributul `type` al marcajului `<input>` trebuie să aibă una dintre valorile următoare (plasate între ghilimele): `radio` (afișează un buton cu interblocare, numit și *buton radio*), `button` (afișează un buton generic, folosit în special pentru prelucrarea pe client a conținutului formularului); `submit` (afișează un buton care determină expedierea conținutului formularului către server) și `reset` (afișează un buton utilizat pentru inițializarea conținutului câmpurilor formularului).

buton de inițializare (*reset*). Buton utilizat pentru inițializarea conținutului câmpurilor unui formular Web, ceea ce înseamnă, de obicei, ștergerea câmpurilor care includ text (*i.e.*, text, password). În (X)HTML este creat folosind marcajul `<input>` și valoarea `reset` a atributului `type` al acestuia, plasată între ghilimele.

Atributul `value` al marcajului `<input>` stabilește ceea ce se va afișa pe buton (*e.g.*, `value = "Sterge"`). Dacă nu

este precizată nici o valoare pentru atributul `value`, pe buton se va afișa `Reset`.

buton de expediere (*submit*). Buton utilizat pentru expedierea (trimiterea) conținutului formularelor Web către server (mai exact, către resursa precizată ca valoare a atributului `action` al marcajului `<form>`), după completarea acestora. În (X)HTML, un astfel de buton este creat folosind marcajul `<input>` și valoarea `submit` a atributului `type` al acestuia, plasată între ghilimele.

Atributul `value` al marcajului `<input>` stabilește ceea ce se va afișa pe buton (*e.g.*, `value = "Trimite"`). Dacă nu este precizată nici o valoare pentru atributul `value`, pe buton se va afișa `Submit`.

buton radio (*radio button*). Butoanele de acest tip sunt incluse în formularele Web, numai unul dintre ele putând fi selectat la un moment dat. În (X)HTML sunt create folosind marcajul `<input>` și valoarea `radio` a atributului `type`. Butoanele `radio` care au valori identice ale atributului `name` sunt grupate automat de browser. Utilizarea atributului `checked` al marcajului `input` (`checked = "checked"`) are ca efect selectarea inițială a unui buton radio.

byte (*binary term*). Denumit și *octet* în limba română, este o unitate de date formată din 8 biți. Mărimea memoriei calculatoarelor se exprimă de obicei în multipli ai *byte*-ului (*e.g.*, *kilobyte*, *megabyte*, *gigabyte* etc.). A se vedea și **bit**, **kilobyte**, **megabyte** și **gigabyte**.

byte code (*cod de octeți*). Formatul instrucțiunilor executate de o mașină virtuală Java. Codurile de octeți (generate de compilatorul Java) sunt seturi de instrucțiuni similare cu codul scris în limbaj de asamblare, independente de mediul de lucru. Spre deosebire de codul-mașină, care este

executat direct de procesor și poate fi folosit numai pe platforma pe care a fost creat, codul de octeți este interpretat de mediul Java, motiv pentru care poate fi rulat pe orice platformă pe care este instalat mediul de execuție Java. A se vedea și **mașină virtuală Java**.

C

C. Limbaj de programare structurat, de uz general, dezvoltat în 1972 de Dennis Ritchie la Bell Laboratories, pentru a fi utilizat pe platforma UNIX. Este un limbaj compilat, care conține o serie de funcții încorporate, dependente de mașină. Restul funcțiilor, independente de mașină, sunt incluse în biblioteci care pot fi utilizate de programele scrise în C.

Limbajul C este utilizat pentru dezvoltarea atât a software-ului de sistem, cât și a aplicațiilor. A influențat în mare măsură dezvoltarea altor limbaje de programare, în special C++, dar și Java și JavaScript. Limbajul este foarte lejer în privința verificării tipurilor de date, lăsând programatorului o libertate deplină, ceea ce determină adesea apariția unor erori și a unor efecte colaterale greu de depistat și de depanat. A se vedea și C++, Java și JavaScript.

C++. Citit *C plus plus*, este un limbaj de programare compilat, multi-paradigmă (*i.e.*, suportă mai multe paradigme de programare), standard ISO/IEC (1998, amendat în 2003), dezvoltat începând cu 1979 de Bjarne Stroustrup la Bell Laboratories. Inițial,

a fost văzut ca o variantă îmbunătățită a limbajului C, care implementează paradigma POO (programarea orientată pe obiect), din această cauză fiind numit inițial *C with classes*; mai târziu, în anul 1983, a fost redenumit C++.

Spre deosebire de limbajul C, C++ a introdus o verificare foarte strictă a tipurilor de date. În particular, apelul unei funcții trebuie precedat de declararea funcției respective. Astfel, folosind declarațiile, parametrii efectivi pot fi verificați prin compararea tipului lor cu cel al parametrilor formali. Dacă se constată nepotriviri între acestea, compilatorul încearcă să stabilească o corespondență prin realizarea unor conversii și generează o eroare în cazul în care nu găsește nici o posibilitate.

C++ este unul dintre cele mai populare limbaje de programare, fiind larg utilizat în crearea de software de sistem, software de aplicație, drivere, aplicații client-server, jocuri etc. A se vedea și **C**, **POO** și **Stroustrup, Bjarne**.

C++Builder. Mediu IDE RAD (*Rapid Application Development*) dezvoltat de compania Borland (achiziționată de Micro Focus în anul 2009) pentru crearea programelor în limbajul C++ pe platforma Windows. A se vedea și **IDE** și **RAD**.

C#. Denumit și C Sharp, este un limbaj dezvoltat pentru platforma .NET Framework, inspirat din Java. La fel ca și în Java, codul C# este compilat în *bytecode* și rulează într-o mașină virtuală denumită *Common Language*

Runtime (CLR) în .NET Framework. A se vedea și **byte code**, **Common Language Runtime**, **Java** și **.NET Framework**.

c-Learning (*c-Learning*). Un tip de învățare denumit și *community learning*, *communicative learning* sau *collaborative learning*, învățare realizată folosind noile medii de instruire, inclusiv pe cele *online*. Din această perspectivă, învățarea este un proces în care se întâlnesc diverse persoane și experiențe.

cablu de date (*data cable*). Cablu care conține conductoare electrice sau fibre optice, utilizat pentru transferul datelor între diverse dispozitive.

cache (*cache*). Depozit local de memorie în care sunt păstrate temporar mesajele de răspuns și un sistem de control al acestui depozit, utilizate cu scopul de a micșora timpul de răspuns și încărcarea rețelei. În mod obișnuit, browserele și serverele Web includ câte un cache. A se vedea și **cache de CPU**.

cache de CPU (*CPU cache*). Memorie rapidă situată între CPU și memoria principală, cu rolul de a stoca temporar date și instrucțiuni necesare CPU-ului pentru execuția unor comenzi viitoare, reducând astfel timpul de acces al memoriei și determinând mărirea vitezei de execuție a programelor. A se vedea și **CPU**.

CAD (*Computer-aided design*, *proiectare asistată de calculator*). Sintagma se referă la utilizarea

calculatorului în scopul proiectării obiectelor. Cu ajutorul programelor CAD pot fi desenate atât figuri 2D (bidimensionale), cât și 3D (tridimensionale). Tehnologiile CAD sunt utilizate în diverse ramuri ale industriei și arhitecturii. Una dintre cele mai cunoscute companii producătoare de software CAD este Autodesk, al cărui produs *AutoCAD* este larg utilizat (acesta a fost unul dintre primele programe CAD care a rulat pe PC, prima sa versiune fiind lansată în anul 1982).

cadru (*frame*). Zonă distinctă inclusă într-o pagină Web. Utilizarea cadrelor constituie o modalitate de structurare a paginilor, în fiecare cadru putând fi afișat câte un document (X)HTML distinct. Dacă este realizată folosind cadre, o pagină Web trebuie să conțină cel puțin două.

Pentru a crea o pagină funcțională cu cadre sunt necesare următoarele documente (X)HTML: un document pentru definirea cadrelor; cel puțin câte un document care va fi deschis în fiecare cadru. În documentul de definire a cadrelor, în locul elementului `body` (care nu mai este folosit) se utilizează elementul `frameset`. Definirea cadrelor, operație care permite împărțirea ferestrei în linii și coloane, se realizează folosind atributele `cols` și `rows` ale elementului `frameset`. Astfel, utilizarea atributului `cols` permite împărțirea ferestrei în cadre de tip coloană, iar folosirea atributului `rows` permite împărțirea ferestrei în cadre de tip linie.

Valorile celor două atribute sunt liste de elemente separate prin virgulă, acestea descriind modul în care se realizează împărțirea spațiului disponibil.

În interiorul elementului `frameset` pot fi utilizate unul sau mai multe elemente `frame`, fiecare dintre acestea introducând un cadru. Precizarea documentului care va fi afișat (încărcat) într-un cadru se realizează prin intermediul atributului `src` al elementului `frame`, acesta având ca valoare URL-ul documentului respectiv.

Ca și alte tipuri de conținut ale paginii Web, cadrele pot fi imbricate, adică pot fi incluse cadre în alte cadre, rezultând o structură complexă.

cadru de date (*data frame*). Unitate de informație transmisă prin rețea la nivelul legăturii de date al modelului de referință ISO/OSI. Include un antet adăugat la nivelul respectiv și un pachet de date. Aceste cadre sunt utilizate pentru schimbul de date dintre două noduri printr-o legătură fizică directă sau una logică. A se vedea și **ISO/OSI**.

cadru de lucru software (*software framework*). Este un proiect reutilizabil pentru dezvoltarea aplicațiilor și poate include programe, biblioteci, un limbaj de scripting, precum și un alt software. Toate componentele sunt menite să ajute la dezvoltarea și conectarea diverselor componente ale unei aplicații create. Diversele părți ale unui cadru de lucru pot fi expuse (și accesate) prin intermediul unor API-uri (*Application*

Programming Interface). Majoritatea cadrelor de lucru utilizează arhitectura MVC (*Model-View-Controller*). A se vedea și **MVC**.

cadru intern (*internal frame*). Zonă dreptunghiulară în interiorul unei pagini Web în care browserul afișează un alt document (X)HTML (intern sau extern site-ului), inclusiv marginile și barele de derulare. Un cadru intern este specificat folosind elementul `iframe`, în interiorul elementului `body`. Dacă browserul nu suportă cadre interne, va fi afișat conținutul elementului `iframe` (dacă acesta există). URL-ul documentului afișat în cadru este precizat ca valoare a atributului `src` al elementului `iframe`. Atributul `name` precizează numele cadrului intern, util în situația în care se dorește deschiderea unui document în interiorul acestuia folosind o legătură (X)HTML din exterior. În acest caz, atributul `target` al marcajului `<a>` trebuie să aibă ca valoare numele cadrului.

CakePHP. Cadru de lucru *open source* utilizat pentru crearea aplicațiilor Web, a cărui dezvoltare a început în anul 2005 (<http://cakephp.org/>). Oferit de Cake Software Foundation, Inc. și scris în PHP, este modelat după *Ruby on Rails* și distribuit sub licență MIT. A se vedea și **cadru de lucru software, licența MIT, PHP și Ruby on Rails**.

calculator (*computer*). Denumit și *sistem de calcul* (*computer system*), este o mașină capabilă să accepte date structurate (ca intrări), să le prelucreze

conform unei liste de instrucțiuni numită *program* și să furnizeze rezultatele (ca ieșiri). Calculatoarele programabile și care pot accepta în principiu orice date de intrare se numesc *calculatoare universale*. După modul de reprezentare a informațiilor, calculatoarele pot fi: analogice, numerice și hibride. În funcție de mărime, viteză și preț, calculatoarele aparțin următoarelor categorii: supercalculatoare, calculatoare medii-mari (*mainframe*), minicalculatoare, microcalculatoare și calculatoare mici. A se vedea și **calculator analogic**, **calculator hibrid**, **calculator numeric**, **calculator universal**, **microcalculator**, **program de calculator** și **supercalculator**.

calculator analogic (*analog computer*). Sistem de calcul care preia sub forma unor tensiuni valorile unor mărimi fizice cu variație continuă (*e.g.*, presiune, temperatură) și le redă utilizatorului printr-un dispozitiv indicator special. A se vedea și **calculator**.

calculator de proces (*process computer*). Controlează diverse procese tehnologice, prelucrând informații numerice sau analogice despre procesul respectiv și furnizând ieșiri cu rol de reglare a proceselor. Un astfel de sistem funcționează *în timp real*, ceea ce înseamnă că timpul de răspuns al calculatorului este compatibil cu constantele de timp ale procesului gestionat. A se vedea și **calculator**.

calculator de uz general (*general-purpose computer*). A se vedea **calculator universal**.

calculator hibrid (*hybrid computer*). Este rezultatul cuplării unui calculator numeric cu unul analogic. În mod obișnuit calculatorul numeric utilizat are o capacitate redusă, dar o memorie suficientă pentru păstrarea datelor. A se vedea și **calculator**, **calculator analogic** și **calculator numeric**.

calculator numeric (*numeric computer*). Sistem de calcul care primește, prelucrează și transmite date codificate sub formă numerică binară.

calculator personal (*personal computer*). Denumit și PC sau microcalculator, este un calculator universal a cărui mărime, performanțe și preț de vânzare îl fac accesibil pentru uz personal. Cu un astfel de calculator poate opera direct utilizatorul, nefiind necesară intervenția unui operator specializat. Primul PC a fost Apple II, lansat în 1977. Astăzi, cele mai utilizate PC-uri sunt cele compatibile IBM PC.

Un calculator personal poate fi de tip *desktop*, *laptop* sau *tablet PC*. Cele mai utilizate sisteme de operare pentru PC-uri sunt Microsoft Windows, Mac OS X și Linux. Microprocesoarele utilizate de PC-uri sunt: *compatibile x86* (produse de companiile Intel, AMD, Cyrix, VIA și altele), *PowerPC* (produs de IBM) și *ARM* (produs de ARM Limited). A se vedea și **calculator universal** și **microcalculator**.

calculator universal (*universal computer*). Denumit și *calculator de uz general* (*general-purpose computer*), este un calculator care poate fi programat astfel încât să execute un domeniu larg de sarcini, pe baza unui software adecvat. A se vedea și **calculator**.

calculatorul Atanasoff-Berry (*ABC, Atanasoff-Berry Computer*). Funcțional în 1942, a fost primul calculator electronic. A fost proiectat și realizat de John Vincent Atanasoff și Clifford Berry. ABC – controlat manual și neprogramabil – nu era un calculator de uz general, fiind limitat numai la rezolvarea sistemelor de ecuații liniare. Utiliza tuburi electronice și o memorie cu tambur rotativ. A se vedea și **Atanasoff, John Vincent**.

calculul lambda (*lambda calculus*). Utilizat în logica matematică și informatică, introdus de Alonzo Church în 1932, este un sistem formal și un sistem de notații pentru definirea și aplicarea funcțiilor. Expresiile notației sunt numite expresii lambda și fiecare asemenea expresie definește o funcție. Expresiile lambda sunt similare procedurilor anonime fără efecte secundare.

În anii 1920, Moses Schönfinkel a dezvoltat o altă teorie a funcțiilor, bazată pe ceea ce astăzi se numește teoria combinatorică. În anii 1930, Haskell Curry a redescoperit și extins teoria lui Schönfinkel și a demonstrat că aceasta era echivalentă cu calculul lambda. În anii 1950, inspirat de

calculul lambda, John McCarthy a inventat limbajul de programare LISP (unul dintre primele limbaje cu caracteristici funcționale). La începutul anilor 1960, Peter Landin a demonstrat cum pot fi specificate limbajele de programare declarativă în calculul lambda, introducând principalele notații ale programării funcționale. În anii 1970, Peter Henderson și Jim Morris au scris lucrări în care au argumentat avantajele utilizării programării funcționale pentru dezvoltatorii de software. În aceeași perioadă, David Turner a arătat că teoria combinatorică a lui Schönfinkel și Curry ar putea fi folosită pentru a crea limbaje de programare funcțională. În anii 1980, mai multe grupuri de lucru au pus în practică ideile lui Henderson și Turner prin conceperea unor arhitecturi speciale, unele având mai multe procesoare. A se vedea și **funcție, LISP și programare funcțională**.

cale (*path*). 1. În domeniul stocării fișierelor, specifică locația unui fișier pe un dispozitiv de stocare și constă dintr-un șir care include o ierarhie arborescentă de directoare separate în mod obișnuit prin caracterele *slash* (/) sau *backslash* (\), urmată de fișierul respectiv. Căile pot fi generale sau relative la directorul de lucru curent. Căile sunt folosite de sistemul de operare pentru localizarea fișierelor și pentru construirea URL-urilor. A se vedea și **cale completă, cale relativă, nume de cale și URL**. 2. În comunicații, legătura dintre două

noduri ale unei rețele. A se vedea și **nod** și **rețea de calculatoare**.

cale completă (*full path*). Numită și *cale generală*, conține toate componentele unei căi: numele unității de disc, al directorului-rădăcină, al posibilelor subdirectoare și al fișierului, e.g., `C:\User\Docs\lista.txt` în Windows sau `/home/user/docs/lista.txt` în Unix. A se vedea și **cale** și **cale relativă**.

cale relativă (*relative path*). Cale care pornește de la directorul de lucru curent, e.g., `User\Docs\lista.txt` în Windows sau `/user/docs/lista.txt` în Unix. A se vedea și **cale** și **cale completă**.

canal (*channel*). În cazul formatelor grafice de tip *raster*, include mulțimea tuturor eșantioanelor de un anumit tip dintr-o imagine, de exemplu componenta *red* a fiecărui pixel. Astfel, o imagine *truecolor* are trei canale, în timp ce o imagine *grayscale* are un singur canal. Când se vorbește despre transparență, se utilizează noțiunea de *alpha channel* (sau *alpha mask*). Acesta este un tip special de canal, care nu furnizează informații de culoare, ci un nivel al transparenței pentru fiecare pixel, sau – mai degrabă – un nivel al opacității (deoarece valoarea maximă indică faptul că pixelul este complet opac, iar valoarea minimă că pixelul este complet transparent). O imagine *truecolor* cu un *alpha channel* este numită și imagine *RGBA*. A se vedea și **formate raster**.

canal dedicat (*dedicated channel*). Legătură de comunicații rezervată

pentru un anumit scop sau pentru un anumit utilizator.

canal sigur (*secure channel*). Legătură de comunicații protejată împotriva interceptării și manipulării neautorizate a datelor.

capacitate de stocare permanentă (*permanent storage*). Dispozitiv de memorare care păstrează informațiile conținute și după întreruperea alimentării (e.g., memoria ROM, hard-disk, CD, DVD, memorie *flash*).

capacitate de stocare primară (*primary storage*). Memoria cu acces aleator (RAM), i.e., principala zonă de stocare la care microprocesorul are acces direct. A se vedea și **capacitate de stocare secundară** și **RAM**.

capacitate de stocare secundară (*secondary storage*). Orice mediu de stocare diferit de RAM (e.g., hard-disk). A se vedea și **capacitate de stocare primară** și **RAM**.

CAPTCHA (*Completely Automated Public Turing test to tell Computers and Human Apart*s). Metodă prin care se determină dacă utilizatorul unei aplicații Web este un om sau nu. A fost dezvoltată începând cu anul 1997 de Andrei Broder (de la AltaVista) și colegii săi, pentru a împiedica roboții să adauge URL-uri la motorul de căutare dezvoltat de ei. Termenul a fost introdus în anul 2000 de Luis von Ahm, Manuel Blum, Nicholas J. Hopper (toți trei de la Carnegie Mellon University) și John Langford (IBM).

Metoda este utilizată, în general, pentru a împiedica roboții să participe la sondaje *online*, să înregistreze conturi de *e-mail* gratuite, să trimită mesaje de *e-mail* nedorite (*spam*) etc. În mod obișnuit, i se prezintă utilizatorului o secvență de caractere (cifre și litere) de mărimi diferite, eventual distorsionate și, uneori, greu de descifrat, cerându-i-se să identifice secvența respectivă.

caracter (*character*). Unitate de informație care corespunde în linii mari unei grafeme (un caracter tipăribil). Caracterele tipăribile sunt literele (mici și mari), cifrele și semnele de punctuație. De asemenea, în tehnica de calcul sunt folosite și caractere netipăribile: invizibile (*e.g.*, spațiu) și de control (*e.g.*, retur de car, tab, sfârșit de paragraf, sfârșit de fișier). A se vedea și **caracter de control**, **metacaracter**, **set de caractere** și **șir de caractere**.

caracter de control (*control character*). Caracter netipăribil inclus într-un set de caractere. Toate caracterele ASCII ale căror coduri zecimale sunt cuprinse între 0 și 31, precum și cel al cărui cod este 127 sunt caractere de control (*e.g.*, *null*, *bell*, *backspace*, *line feed*, *form feed*). Majoritatea sunt utilizate de programatori și utilizatori pentru a controla un dispozitiv de tipărire, de comunicații sau de afișare pe ecran.

Fiecare dintre caracterele ASCII cu codurile zecimale cuprinse între 1 și 26 pot fi introduse din tastatură, acționând simultan tasta CTRL și o tastă literală (A, B, C, ..., Z). Caracterele de control

cu codurile ASCII cuprinse între 27 și 31 pot fi introduse folosind tasta CTRL și caracterele [, \,] , ^ și _ . A se vedea și **caracter** și **set de caractere**.

caracter de înlocuire (*wildcard character*). Caracter care poate fi utilizat pentru a înlocui unul sau mai multe caractere. De exemplu, semnul de întrebare (?) ține locul unui singur caracter, iar asteriscul (*) ține locul unui număr oarecare de caractere. A se vedea și **caracter**.

caracter rezervat (*reserved character*). Caracter care, într-un context dat, are o semnificație specială și, prin urmare, nu poate fi folosit de utilizator în alte scopuri. De exemplu, în scrierea URL-urilor, sunt utilizate următoarele caractere speciale: procent (%) are funcția de caracter escape; slash (/) este utilizat pentru delimitarea subșirurilor aflate în relație ierarhică; hash (#) este utilizat ca delimitator pentru separarea unui identificator de fragment; semnul de întrebare (*question mark*, ?) este folosit pentru delimitarea șirului de interogare (*query string*). A se vedea și **URL**.

CareerBuilder.com. Fondat în 1994 de Rob McGovern, este cel mai mare site pentru căutarea locurilor de muncă din SUA (<http://www.careerbuilder.com/>).

card de memorie (*memory card*). Dispozitiv pentru stocarea datelor bazat pe tehnologia de memorare *flash*. Cardurile de memorie sunt utilizate în principal în camere foto și video digitale, PDA-uri, telefoane mobile,

playere muzicale, console de jocuri etc. Sunt caracterizate prin așa-numitul *factor de formă* (*form factor*), care se referă la dimensiunile acestora. Iată câteva tipuri de carduri de memorie (între paranteze sunt incluse acronimele corespunzătoare): *PC Card* (PCMCIA), *SmartMedia* (SM), *Memory Stick* (MS), *Multimedia Card* (MMC), *miniSD* (miniSD), *microSD* (microSD), *Intelligent Stick* (iStick). A se vedea și **memorie flash**.

careware (*careware*). Denumit și *charityware*, *helpware* sau *goodware*, este un software distribuit gratuit, însoțit de sugestia ca, după ce l-a încercat, utilizatorul să facă o donație într-un scop caritabil nominalizat. Termenul a fost folosit pentru prima oară în anul 1988, de Al. Stevens, într-un articol publicat în revista *Dr. Dobbs' Journal*. A se vedea și **freeware** și **shareware**.

carte de adrese (*address book*). Lista tuturor adreselor și a numelor persoanelor care le dețin, salvate de un utilizator într-un program de *e-mail*. Această listă este pusă la dispoziție sub diverse nume (e.g., *Contacts* în *Yahoo! Mail*).

CASE (*Computer-aided software engineering, proiectare de software asistată de calculator*). Termen utilizat în ingineria software, desemnează utilizarea calculatorului în toate etapele de realizare a unui program. Un mediu de lucru CASE include un set de instrumente avansate care permit automatizarea proiectării și imple-

mentării unor aplicații complexe, utilizate în diverse domenii. A se vedea și **inginerie software**.

casetă de avertizare (*alert box*). În cazul interfețelor grafice cu utilizatorul (GUI), casetă afișată pe ecran de sistemul de operare sau de aplicație pentru a prezenta un mesaj de avertizare. A se vedea și **casetă de dialog** și **GUI**.

casetă de dialog (*dialog box*). În cazul interfețelor grafice cu utilizatorul (GUI), casetă afișată pe ecran de sistemul de operare sau de aplicație pentru a solicita un răspuns din partea utilizatorului. A se vedea și **casetă de avertizare** și **GUI**.

casetă de editare multilinie (*text area*). Casetă de editare inclusă într-un formular al unei pagini Web, în care utilizatorul poate introduce mai multe rânduri de text. În (X)HTML este creată folosind marcajul `<textarea>` și atributele sale. Numerele de linii (rânduri) și de coloane vizibile (i.e., afișate în casetă) sunt precizate prin intermediul atributelor obligatorii `rows` și, respectiv, `cols`. Dacă textul introdus depășește numărul de linii specificat prin intermediul atributului `rows`, va fi afișată o bară de defilare verticală (*scroll*). A se vedea și **pagină Web**.

casetă de validare (*checkbox*). Element de control interactiv utilizat în interfețele grafice cu utilizatorul (GUI). Casetele de validare permit activarea sau dezactivarea unor opțiuni sau caracteristici dintr-un set. Aceste casete sunt

incluse – de exemplu – în formularele conținute de unele pagini Web. O casetă de validare are două stări: bifată (opțiune activată) și nebifată (opțiune dezactivată). Când este bifată, în casetă apare un x sau un alt marcaj de activare.

Pentru a crea casete de validare în paginile Web folosind limbajul de marcare (X)HTML, se utilizează marcajul `<input>` și valoarea `checkbox` a atributului `type` al acestuia, plasată între ghilimele. Utilizarea atributului `checked` al marcajului `input` (`checked = "checked"`) activarea inițială a casetei. A se vedea și **GUI și pagină Web**.

catalog (*catalog*). A se vedea **dosar**.

căutare binară (*binary search*). Algoritm folosit la căutarea unui element într-o listă ordonată (tablou unidimensional), care funcționează pe baza tehnicii *divide et impera*, astfel: elementul căutat este comparat mai întâi cu cel din mijlocul listei, după care lista este împărțită în două, se stabilește în care jumătate ar trebui să se găsească elementul, repetându-se procedeul până la găsirea acestuia. Deoarece în fiecare etapă cardinalul mulțimii elementelor în care se face căutarea se înjumătățește, algoritmul are o complexitate logaritmică. În esență, este algoritmul după care se caută un cuvânt într-un dicționar sau un nume în cartea de telefon. Algoritmul – care poate fi implementat atât iterativ, cât și recursiv – este mult mai rapid decât cel de căutare

secvențială, în special în cazul tablourilor mari. Complexitatea acestui algoritm este $O(\log_2 n)$. A se vedea și **algoritm de căutare și căutare secvențială**.

căutare după text complet (*full-text search*). Metodă de căutare în documente stocate pe un computer sau într-o bază de date, în care motorul de căutare examinează pe rând toate documentele sau toate înregistrările existente căutând textul complet specificat de utilizator. Metoda este folosită de numeroase aplicații de procesare a textului (*e.g.*, Word) și de aplicații Web. Din cauza ambiguităților limbajului natural, metoda căutării după text complet produce o listă de rezultate cu precizie scăzută, unele dintre acestea fiind irelevante. A se vedea și **căutare indexată**.

căutare indexată (*indexed search*). Căutare care folosește un index pentru localizarea itemilor și pentru micșorarea timpului de căutare. Indexul este căutat secvențial, pentru a obține referințe către resursele care conțin datele căutate. A se vedea și **căutare după text complet**.

căutare secvențială (*sequential search*). Este cel mai simplu algoritm de căutare într-un tablou. Constă în parcurgerea structurii de date de la început la sfârșit, comparându-se valoarea căutată cu valoarea găsită în poziția curentă din tablou. Căutarea secvențială poate fi implementată în mai multe moduri: căutare secvențială

(propriu-zisă), căutare secvențială rapidă, căutare secvențială într-un tablou ordonat. Complexitatea algoritmului de căutare secvențială este $O(n)$. A se vedea și **algoritm de căutare** și **căutare binară**.

căutare socială (*social search*). Termenul (pus în circulație în anii 2004–2005) desemnează căutarea realizată prin intermediul motoarelor de căutare socială. Ia numeroase forme, începând de la cele simple, care implică *bookmark*-uri partajate și marcarea conținutului cu etichete descriptive, până la abordări complexe care combină inteligența umană cu diverși algoritmi de calcul. A se vedea și **motor de căutare socială**.

câmp-cheie (*key field*). Câmp al unei înregistrări sau atribut dintr-un tabel al unei baze de date relaționale inclus într-o cheie compusă. A se vedea și **cheie compusă**.

câmp de date (*data field*). Porțiune bine definită a unei înregistrări de date (e.g., o coloană într-un tabel al unei baze de date sau un câmp al unui formular Web).

câmp de editare de tip parolă (*password-edit*). Casetă cu o singură linie inclusă într-un formular al unei pagini Web, în care utilizatorul introduce o parolă. Din motive de securitate, browserul va afișa datele introduse folosind caracterul asterisc (*). În (X)HTML, o astfel de casetă este creată folosind marcajul `<input>` și valoarea `password` a atributului `type`

al acestuia, plasată între ghilimele. Valoarea atributului `maxlength` al marcajului `<input>` precizează numărul maxim de caractere care pot fi introduse (e.g., `maxlength = "30"`). A se vedea și **pagină Web**.

câmp de editare de tip text (*text-edit*). Numită și *casetă de tip text* sau *casetă de editare*, este o casetă cu o singură linie inclusă într-un formular dintr-o pagină Web, care îi oferă utilizatorului posibilitatea de a introduce un text. În (X)HTML, este creată folosind marcajul `<input>` și valoarea `text` pentru atributul `type` al acestuia, plasată între ghilimele. Este tipul implicit de câmp, ceea ce înseamnă că această casetă va fi afișată și în situația în care atributul `type` nu este utilizat. Valoarea atributului `maxlength` al marcajului `<input>` precizează numărul maxim de caractere care pot fi introduse (e.g., `maxlength = "30"`). A se vedea și **pagină Web**.

CBL (*Computer-based Learning, învățare bazată pe calculator*). A se vedea **tipuri de e-Learning**.

CBT (*Computer-based training, instruire bazată pe calculator*). A se vedea **tipuri de e-Learning**.

cc (*carbon copy, copie la indigo*). Denumită și *courtesy copy* (*copie de informare*), este o copie a unui mesaj de *e-mail* trimisă altor persoane în afara celor cărora le este destinat. Folosirea *cc* implică faptul că destinatarul nu trebuie să întreprindă nici o acțiune în urma primirii mesajului, acesta având

numai un scop informativ. Trimiterea unei copii *cc* este vizibilă pentru destinatarul de drept al mesajului. A se vedea și **bcc** și **mesaj de e-mail**.

ccLearn. Divizie a *Creative Commons* dedicată revelării întregului potențial al Internetului de a sprijini studiul deschis și resursele educaționale deschise (<http://learn.creativecommons.org/>).

Misiunea *ccLearn* este de a minimiza barierele (tehnice, legale și sociale) care privesc partajarea și reutilizarea materialelor educaționale. A se vedea și **Creative Commons**.

CDATA (*character data*). Secțiune într-un fișier XML cu rolul de a inhiba procesarea unor blocuri de text care, altfel, ar fi tratate ca marcaje. Pentru precizarea unei astfel de secțiuni se utilizează sintaxa:

```
<![CDATA[ . . . ]]>
```

Aceste secțiuni sunt utile, de exemplu, pentru includerea în document a unor secvențe de cod scrise într-un limbaj de programare (e.g., JavaScript). Nu este permisă includerea secțiunilor **CDATA** unele în altele. A se vedea și **XML**.

CeBIT (*Centrum der Büro und Informationstechnik*). Cea mai mare expoziție mondială de calculatoare (<http://www.cebit.de/>). Are loc anual, începând din 1986, în localitatea Hanovra din Germania. A se vedea și **COMDEX** și **COMPUTEX**.

cerere HTTP (*HTTP request*). Cerere adresată de un client (e.g., browser, spider) serverului Web. În acest

context, clientul este denumit și *user agent*.

O cerere HTTP conține: identificatorul resursei dorite, metoda de acces utilizată, versiunea protocolului HTTP, precum și metainformații care vor fi utilizate de server.

Metoda de acces precizează acțiunea care trebuie realizată asupra resursei specificate. În HTTP 1.1. sunt folosite opt metode: HEAD, GET, POST, PUT, DELETE, TRACE, OPTIONS și CONNECT. Există și posibilitatea utilizării unor metode nestandardizate, cu condiția ca ele să poată fi identificate de cele două părți (i.e., client și server). Metodele HEAD, GET, OPTIONS și TRACE sunt definite ca *sigure* (*safe*), ceea ce înseamnă că ele sunt utilizate numai pentru regăsirea informațiilor, neschimbând starea serverului. În schimb, metodele POST, PUT și DELETE sunt definite ca *nesigure*, deoarece pot schimba starea serverului. Cel mai des folosite metode sunt GET (pentru a obține o resursă precizată prin intermediul unui *URI), POST (e.g., pentru trimiterea pe server a datelor incluse într-un formular) și HEAD (pentru obținerea unor informații referitoare la o resursă Web, fără a o transfera în întregime).

Metainformațiile se referă la anteturile cererii (*request headers*), denumite și atribute (*request attributes*). Sunt utilizate pentru a transmite serverului Web informații suplimentare despre cererea curentă și despre client. Iată câteva anteturi: Accept, Accept-Encoding,

Accept-Language, Authorization, Content-Type, From, Host, Proxy-Authorization, Referer, User-Agent. Pentru precizarea unui antet HTTP se utilizează următoarea sintaxă: `nume_antet: valoare_antet.`

A se vedea și **HTTP** și **răspuns HTTP**.

Cerf, Vinton Gray (n. 1943). Informatician american, considerat adesea părintele Internetului, deoarece a proiectat (împreună cu Bob Kahn) suita de protocoale TCP/IP. Pentru realizările sale a primit numeroase distincții (e.g., medalia Alexander Graham Bell, premiul Turing). A se vedea și **Kahn, Robert Elliot**.

CERN (*Conseil Européen pour la Recherche Nucléaire, Consiliul European pentru Cercetări Nucleare*). Centru de cercetări în domeniul fizicii nucleare situat lângă Geneva, Elveția. Este locul în care Tim Berners-Lee a început în anul 1989 dezvoltarea Web-ului (astăzi cel mai popular serviciu al Internetului), ca modalitate de îmbunătățire a comunicațiilor între membrii comunității de cercetători. A se vedea și **World Wide Web** și **Berners-Lee, Tim**.

certificat digital (*digital certificate*). Eliberat de o autoritate de certificare, identifică și răspândește cheia publică, utilizată pentru verificarea semnăturilor digitale. Certificatul digital este folosit pentru o gamă largă de tranzacții electronice (e.g., e-mail, comerț electronic, transfer electronic de fonduri). Reprezintă un instrument

pentru stabilirea unui canal securizat utilizat în comunicarea informațiilor confidențiale. A se vedea și **autoritate de certificare** și **semnătură digitală**.

CGI (*Common Gateway Interface*). Prima metodă (din punct de vedere istoric) de generare dinamică pe server a conținutului Web. Metoda permite serverului Web să transfere unei aplicații externe (numită *script CGI*) o cerere venită de la un client Web (e.g., browser) și să îi trimită acestuia conținutul generat în urma execuției aplicației.

Aplicațiile externe utilizate prin intermediul interfeței CGI sunt numite și programe *gateway*, deoarece se comportă ca o poartă între serverul Web și sursele de date existente pe server. Specificațiile CGI au fost scrise în anul 1993, printre cei care au contribuit la crearea acestora numărându-se Rob McCool, John Franks, Ari Loutenen, Tony Sanders și George Phillips. Din punct de vedere cronologic, etapa următoare în evoluția metodelor de generare dinamică a conținutului Web a fost reprezentată de utilizarea interfețelor de programare NSAPI (Netscape Server API) și ISAPI (Microsoft Internet Services API), acestea funcționând conform modelului CGI.

cgi-bin. Prescurtare de la **CGI binaries**, director care conține programele externe (numite și *scripturi*) executate de serverul HTTP prin CGI. A se vedea și **CGI**.

CGI script (*script CGI*). Aplicație externă executată de un server HTTP ca urmare a cererii unui client (e.g., browser Web). Comunicația între script și server se desfășoară prin intermediul interfeței CGI. Scripturile CGI pot fi scrise în diverse limbaje de programare (e.g., C, C++), cel mai des utilizat fiind Perl. A se vedea și **CGI**.

CGM (*Computer Graphics Metafile*). Format grafic de tip *metafile*. A se vedea și **formate metafile**.

chart (*diagramă*). A se vedea **diagramă**.

chat (*chat*). Formă de conversație în timp real în Internet purtată prin intermediul serviciului IRC. *Chat* este un cuvânt care provine din limba engleză, însemnând *discuție, sporovăială, taifas*. A se vedea și **IRC**.

chatbot (*agent conversațional*). Denumit și *ChatterBot*, este un agent software (program de calculator) care poate purta un „dialog” cu un utilizator uman, adică are capacitatea de a răspunde în acord cu ceea ce i se comunică. Programul trebuie să fie astfel realizat încât replicile lui să pară cât mai „umane”. Termenul *ChatterBot* a fost introdus de Michael Mauldin în 1994. Două dintre *chatbot*-urile cunoscute sunt ELIZA și ALICE. A se vedea și **ALICE** și **ELIZA**.

cheie alternativă (*alternate key*). Orice cheie candidată definită pentru un tabel al unei baze de date relaționale și care nu a fost desemnată drept cheie

primară. A se vedea **bază de date relațională**, **cheie candidată** și **cheie primară**.

cheie candidată (*candidate key*). Identificator unic pentru un rând (înregistrare) dintr-un tabel al unei baze de date, alcătuit dintr-o coloană sau o combinație de coloane. Acesta trebuie să respecte următoarele reguli: unicitate (o cheie candidată identifică în mod unic o înregistrare în tabel); valori nenule (coloanele care compun cheia trebuie să aibă valori nenule); compoziție minimală (nici o coloană din cheie – când aceasta este compusă – nu poate fi eliminată fără a afecta unicitatea înregistrării în tabel). A se vedea și **bază de date relațională** și **cheie primară**.

cheie compusă (*composite key*). Cheie în care sunt incluse două sau mai multe coloane ale unui tabel conținut într-o bază de date relațională. A se vedea și **bază de date relațională**.

cheie de căutare (*search key*). Coloană inclusă într-un tabel, analizată într-un proces de căutare într-o bază de date relațională. A se vedea și **bază de date relațională**.

cheie primară (*primary key*). Una dintre cheile candidate. Pentru alegerea acesteia se pot utiliza drept criterii de selecție lungimea și ușurința în manipulare. A se vedea și **bază de date relațională** și **cheie candidată**.

chioșc informațional (*information kiosk*). Terminal cu acces liber, care

oferă publicului informații din diverse domenii. Astfel de terminale sunt instalate în primării, companii, muzee, instituții de învățământ etc. În mod obișnuit, informațiile sunt prezentate în formate multimedia, iar terminalele sunt dotate cu *touchscreen*. A se vedea și **touchscreen**.

chipset (*chipset*). Într-un calculator personal (PC), set de circuite integrate specializate situate pe placa de bază (*motherboard*). În mod obișnuit, setul include o pereche de chipuri: *northbridge* (conectează CPU-ul și dispozitivele de înaltă viteză, de exemplu memoria principală și placa grafică) și *southbridge* (conectează magistralele periferice de viteză joasă, de exemplu PCI, IDE, USB). În *chipset*-urile moderne sunt incluse și periferice integrate (e.g., dispozitive Ethernet, USB și audio). Un *chipset* este proiectat să lucreze cu o anumită familie de micro-procesoare. Producătorii cunoscuți de *chipset*-uri pentru plăcile de bază ale PC-urilor sunt NVIDIA, AMD, VIA Technologies, Intel, SiS și Broadcom. A se vedea și **placa de bază**.

ciberspațiu (*cyberspace*). Spațiul perceput pe ecranul calculatorului, în opoziție cu spațiul real. Este modelat folosind limbaje speciale: VRML (*Virtual Reality Modeling Language*) și X3D (VRML rescris în termenii XML). Ciberspațiul poate fi perceput global cu ajutorul unor dispozitive speciale (e.g., mînușa de date, casca de vizualizare). A se vedea și **hiperspațiu**, **mînușă de date** și **VRML**.

ciclu de instrucțiune (*instruction cycle*). Reprezintă prelucrările efectuate de CPU pentru execuția unei instrucțiuni. Constă din următoarele subcicluri: extragerea instrucțiunii din memorie, execuție, întrerupere. În cazul anumitor instrucțiuni, cele trei subcicluri sunt completate cu un subciclu de indirectare, plasat după cel de extragere. Orice subciclu poate fi descompus în operații elementare, numite și micro-operații. Unitatea de comandă și control (UCC) are rolul de a genera semnalele de comandă care asigură secvența corectă de execuție a micro-instrucțiunilor. A se vedea și **CPU** și **UCC**.

ciclu de viață (*life cycle*). În ingineria software, include etapele dezvoltării unui produs informatic. În aceste etape se desfășoară activități strâns legate între ele: analiza și specificarea cerințelor, proiectarea (designul), implementarea, evaluarea și testarea, întreținerea. Etapele au o anumită succesiune temporală, iar pentru realizarea fiecărei etape se folosesc tehnici specifice. A se vedea și **inginerie software**.

cifra de control (*check digit*). Cifră suplimentară adăugată la sfârșitul cifrelor existente într-un mesaj și calculată pe baza acestora, în scopul detecției erorilor. Pe baza cifrei de control pot fi detectate erori ca: introducerea eronată a unor cifre, lipsa sau permutarea altora. Cifra de control este echivalentul zecimal al sumei de control (*checksum*). A se vedea și **suma de control**.

circuite logice programabile (PLD, *Programmable Logic Device*). Circuite integrate care conțin un număr mare de porți sau celule a căror interconexiune poate fi „programată” în scopul de a implementa orice funcție combinatorică sau secvențială dorită.

circuit integrat (*integrated circuit*). În electronică, circuit electronic miniaturizat fabricat pe suprafața unui strat subțire de material semiconductor. Circuitele integrate sunt componente universale ale unui număr mare de dispozitive utilizate de omul modern (e.g., televizoare, calculatoare personale, telefoane mobile, aparate electrocasnice). Circuitul integrat a fost inventat în 1958 de Jack Kilby (laureat al premiului Nobel pentru fizică în anul 2008 pentru această descoperire). Alături de Kilby, pentru această invenție este creditat și Robert Noyce, cei doi lucrând independent. A se vedea și **Kilby, Jack și Noyce, Robert Norton**.

circuit switching (*comutare de circuite*). A se vedea **comutare de circuite**.

Cirip. Aplicație românească de *microblogging* (<http://www.cirip.ro/>). Platforma *Cirip*, dezvoltată de compania Timsoft din Timișoara (adresa companiei este <http://www.timsoft.ro/>), a fost lansată pe 17 martie 2008. Din echipa de concepție, management și dezvoltare a proiectului fac parte, printre alții, Carmen Holotescu, Cristian Armeana și Cristian Regep. A se vedea și **microblogging**.

CISC (*Complex Instruction Set Computing*, *prelucrarea datelor pe baza unui set complex de instrucțiuni*). Strategie de design și model pentru microprocesor care au ca scop implementarea unui set complex de instrucțiuni astfel încât să poată fi apelate la nivelul limbajului de asamblare. Complexitatea setului are ca preț o durată mai mare de execuție a unei singure instrucțiuni. Din categoria microprocesoarelor CISC fac parte, printre altele, familiile x86 (Intel, AMD) și VAX (DEC). A se vedea și **RISC și x86**.

Cisco Systems, Inc. Corporație multinațională cu sediul în SUA, care activează în domeniul proiectării și vânzării de tehnologie și servicii pentru comunicații și rețele de calculatoare (<http://www.cisco.com/>). Compania a fost fondată în 1984 de Len Bosack și Sandy Lerner. Lor li s-a adăugat mai târziu Richard Troiano. Cisco este cunoscută mai ales pentru switch-urile și ruterele pe care le produce și comercializează, precum și pentru pachetele software folosite de acestea.

clasă (*class*). În paradigma programării orientate pe obiect (POO), reprezintă un tip de dată. Definește aspectele abstracte ale unei entități prin caracteristicile (reprezentate de proprietăți sau atribute) și comportamentul acesteia (reprezentat de metode sau operațiuni). Din punct de vedere conceptual, clasele sunt comparabile cu categoriile folosite de oameni (e.g., animal, plantă, figură geometrică). Proprietățile și metodele

definite într-o clasă sunt denumite *membri*. A se vedea și **obiect și programare orientată pe obiect**.

clasă abstractă (*abstract class*). În paradigma programării orientate pe obiect (POO), desemnează o clasă care include una sau mai multe metode abstracte. O astfel de clasă nu prezintă funcționalități complete, motiv pentru care nu poate fi instanțiată. O clasă abstractă realizează implementarea unei noțiuni care nu poate fi concretizată, dar surprinde o caracteristică comună a claselor specializate, derivate din aceasta. Altfel spus, o clasă abstractă abstractizează comportamentul sub-claselor (claselor derivate) specializate.

Dacă metodele abstracte nu sunt definite nici în clasele derivate, acestea devin și ele clase abstracte. Clasele care nu includ metode abstracte sunt denumite și clase concrete. Într-o clasă abstractă pot fi declarate atât metode abstracte, cât și metode „obișnuite” (concrete). A se vedea și **metodă abstractă, moștenire și programare orientată pe obiect**.

clasă de adrese IP (*IP class*). Reprezintă un domeniu de adrese IP. Acestea din urmă sunt împărțite în cinci clase folosind primii biți ai identificatorului de rețea care specifică modul în care restul adresei trebuie să fie împărțită între partea de rețea și partea de gazdă: dacă primul bit este 0, adresa IP aparține clasei A. În acest caz, valoarea primului octet este cuprinsă între 0 și 127; dacă primii doi biți sunt 10, adresa aparține clasei B, iar

valoarea primului octet este cuprinsă între 128 și 191; dacă primii trei biți sunt 110, adresa aparține clasei C. În acest caz, valoarea primului octet este cuprinsă între 192 și 223; dacă primii patru biți sunt 1110, adresa aparține clasei D (folosită pentru rețele *multicast*), iar valoarea primului octet este cuprinsă între 224 și 239; clasa E, incluzând adrese prefixate cu 1111, este rezervată pentru o utilizare viitoare.

În adresele de clasă A, primul octet este utilizat de identificatorul de rețea (primul bit al acestuia fiind 1), iar ultimii trei de identificatorul de gazdă (*e.g.*, 12.00.3). Această clasă de adrese a fost proiectată pentru a satisface cerințele necesare rețelelor de mari dimensiuni. Identificarea gazdei este realizată folosind un număr mare de biți (24). În această clasă nu sunt incluse rețelele 0.0.0.0 și 127.x.y.z.

În adresele de clasă B, cei patru octeți sunt împărțiți în mod egal de cei doi identificatori (*e.g.*, 180.16.0.1). În schimb, în adresele de clasă C identificatorul de rețea ocupă primii trei octeți, iar cel de gazdă este reprezentat pe ultimul octet (*e.g.*, 192.12.7.8).

Respectând constrângerile descrise anterior, rezultă că adresele IP ale gazdelor aparținând celor cinci clase sunt incluse în următoarele domenii: adresele de clasă A: 1.0.0.0 – 127.255.255.255; adresele de clasă B: 128.0.0.1 – 191.255.255.255; adresele de clasă C: 192.0.0.0 –

223.255.255.255; adresele de clasă D: 224.0.0.0 – 239.255.255.255; adresele de clasă E: 240.0.0.0 – 247.255.255.255.

Sunt posibile 126 ($2^7 - 2$) de rețele de clasă A, fiecare cu câte 16.777.214 ($2^{24} - 2$) gazde; de asemenea, sunt posibile 16.382 ($2^{14} - 2$) rețele de clasă B, fiecare cu câte 65.534 ($2^{16} - 2$) gazde; în sfârșit, sunt posibile 2.097.150 ($2^{21} - 2$) rețele de clasă C, fiecare cu câte 254 ($2^8 - 2$) gazde.

Clasa de adrese D este folosită pentru rețele *multicast* (i.e., de trimitere multiplă). Acest spațiu de adrese este plat, toți octeții fiind folosiți pentru definirea adresei de gazdă. Fiecare adresă de clasă D identifică un grup de gazde. Deoarece primii patru biți ai primului octet sunt fixați (1110), rezultă că numărul adreselor de multicast (grupurilor) este de circa 268 de milioane. Dacă un proces trimite un pachet unei adrese de clasă D, acesta va fi livrat tuturor membrilor grupului (dar fără garanția că îl vor recepționa). Există proiecte de cercetare mari bazate pe multicast, unul dintre acestea fiind *Mbone* (*multicast backbone*).

Clasa de adrese E nu poate fi folosită în rețelele publice sau în soluții de *multicast*, fiind rezervată. A se vedea și **adresă IP**.

clasă de bază (*base class*). Denumită și *clasă-părinte* (*parent class*), reprezintă în programarea orientată pe obiect (POO) o clasă din care a fost derivată o altă clasă (denumită *clasă extinsă* sau

derivată), printr-un procedeu denumit *moștenire*. A se vedea și **clasă extinsă, moștenire și programare orientată pe obiect**.

clasă extinsă (*extended class*). Denumită și *clasă derivată* (*derived class*), reprezintă în programarea orientată pe obiect (POO) clasa obținută prin procedeu de *derivare* (sau *moștenire*) dintr-o clasă de bază. A se vedea și **clasă de bază, moștenire și programare orientată pe obiect**.

clasă virtuală (*virtual classroom*). Spațiu *online* privat, accesibil 24/7 via Internet, pe care profesorii îl pot utiliza pentru a sprijini învățarea. Învățarea într-o clasă virtuală este una dintre formele de instruire bazate pe *e-Learning*. Studentul nu este singur în procesul de învățare, aparținând unui grup (i.e., clasa) care beneficiază de materiale de studiu *online* și de îndrumarea unui tutore. O clasă virtuală le permite studenților și profesorilor – ca și clasa tradițională – să participe în timp real la lecții și discuții. Într-o lecție *online*, ca și într-o lecție desfășurată în clasa tradițională, fiecare student beneficiază nu numai de atenția profesorului, ci și de a celorlalți membri ai clasei, ceea ce înseamnă că acesta învață nu numai din materialele primite, ci și din colaborarea cu colegii săi. Avantajul unei lecții *online* este acela că, spre deosebire de lecția clasică, poate fi accesată oriunde și oricând.

O clasă virtuală poate conține: activități în clasă (acestea pot fi: activități de învățare individuală sau în grup;

discuții cu studenții, părinții sau oaspeții; teste; teme pentru acasă; cursuri complete, organizate pe module; forumuri și sesiuni de *chat*) și elemente de organizare a clasei (acestea sunt: informații importante pentru studenți și părinți; calendare și orare; pagini ale studenților).

Clasa virtuală beneficiază de coordonarea unui tutore, care efectuează următoarele activități: planifică activitatea clasei de studenți; supune spre dezbateră studenților – în conferințe asincrone (e.g., forumuri de discuții) – aspecte ale lecției și furnizează resurse auxiliare; analizează și comentează rezolvarea temelor, identificând pentru fiecare student problemele care au apărut în activitatea sa.

Tehnologia utilizată trebuie să permită constituirea claselor și crearea unui sistem de comunicare folosind *e-mail*-ul, forumurile și *chatul*. A se vedea și **e-Learning**.

click-dreapta (*right click*). Execuția unui click folosind butonul drept al *mouse*-ului. Pe platforma Windows, are ca efect deschiderea unui meniu pop-up (sau contextual) cu o serie de opțiuni corespunzătoare unor acțiuni care pot fi efectuate asupra obiectului selectat cu cursorul. A se vedea și **meniu contextual** și **meniu pop-up**.

click through (*click pe*). Procedeu prin care se efectuează un click pe o legătură inclusă în pagina de rezultate returnate de un motor de căutare, având ca efect încărcarea în browser a unei pagini

aparținând unui site indexat de motor. Legătura respectivă este foarte importantă pentru site, deoarece prin intermediul paginii către care duce sunt primiți utilizatorii motorului de căutare. A se vedea și **motor de căutare**.

client (*client*). Într-o rețea locală sau în Internet, calculator sau program care utilizează resursele de rețea oferite de alt calculator sau de alt program, conform unui protocol de comunicație. De exemplu, browserul este client al unui server Web. A se vedea și **arhitectura client-server** și **server**.

client FTP (*FTP client*). Program care îi permite utilizatorului să acceseze un site FTP, pentru a încărca sau descărca fișiere pe/de pe acesta printr-o rețea care utilizează protocolul FTP. A se vedea și **FTP**, **server FTP** și **site FTP**.

client Web (*Web client*). A se vedea **browser Web**.

clipboard (*clipboard*). Zonă de memorie temporară utilizată de sistemele de operare bazate pe ferestre (e.g., Microsoft Windows, Apple Mac OS) pentru a stoca ultimele informații (e.g., text, imagini) tăiate (*cut*) sau copiate (*copy*) de utilizator. Datele stocate pot fi transferate în programul curent sau în alt program, prin lipire (*paste*). În *clipboard* informațiile sunt păstrate temporar (în timpul unei sesiuni de lucru). Informațiile existente rămân în *clipboard* până la următoarea operație de tăiere sau copiere, când vor fi suprascrise cu cele curente.

clipping path (*cale de decupare*). În grafica 2D este un poligon sau o curbă utilizată pentru a tăia (delimita) o porțiune (suprafață) dintr-o imagine editată într-un program de editare a imaginilor. După decupare va fi păstrat numai ceea ce există în interiorul căii respective.

cloaking (*ascundere*). Metodă asociată cu utilizarea așa-numitelor *doorway pages* (*pagini de intrare*). Vizitatorului îi este livrată o versiune a paginii Web diferită de cea furnizată roboților motoarelor de căutare, utilizând scripturi care diferențiază vizitatorii de roboți pe baza adresei IP a acestora. A se vedea și **doorway page**.

clonă (*clone*). În domeniul calculatoarelor, este un sistem hardware sau software proiectat pentru a imita cât mai bine un alt sistem. Scopul declarat este obținerea compatibilității cu sistemul original, care aparține unei mărci binecunoscute și adesea mult mai scump decât clona. De exemplu, *clonă PC* (sau *compatibil IBM PC*) se referă la orice calculator personal compatibil cu PC-ul realizat de IBM; pachetul de programe *open source OpenOffice.org* oferit de Sun Microsystems, Inc. este o clonă pentru *Microsoft Office*.

Clopotel.ro. Denumit și *portalul copiilor din România*, este o comunitate a tinerilor care doresc să comunice *online* și să-și petreacă timpul liber în mediul virtual (<http://www.clopotel.ro/>).

Utilizatorii pot posta materiale, după ce și-au creat conturi. Conținutul site-ului

este protejat de *Legea dreptului de autor*, toate materialele aparținând firmei în proprietatea căreia se află acesta. Totuși, materialele oferite în acest scop pot fi descărcate sau copiate.

cluster de servere (*server cluster*). Grup de servere independente care lucrează împreună ca un sistem unic, pentru a le oferi clienților servicii cu disponibilitate ridicată. Atunci când apare o eroare de funcționare a unui calculator inclus în cluster, cererile clienților sunt redistribuite către celelalte calculatoare existente. Clusteretele de servere sunt utilizate atunci când se dorește ca utilizatorii să aibă acces constant la resurse importante, livrate într-o rețea care folosește arhitectura client-server. A se vedea și **arhitectura client-server** și **server**.

CMS (*Content Management System, sistem pentru managementul conținutului*). Aplicație utilizată pentru gestionarea fluxului de lucru necesar în crearea colaborativă, editarea, căutarea și arhivarea unui conținut media și textual divers. Conținutul poate fi reprezentat de fișiere stocate pe calculator, imagini, fișiere audio și video, documente electronice și conținut Web. Există șase categorii principale de CMS-uri: Enterprise CMS (ECMS), Web CMS (WCMS), Document Management System (DMS), Mobile CMS (MCMS), Component CMS (CCMS), Media CMS (MCMS). A se vedea și **Web CMS**.

CMYK. Model substractiv de reprezentare a culorilor. Utilizează patru culori de bază: Cyan (C), Magenta (M), Yellow (Y, galben) și Black (B, negru). Cu ajutorul acestui model se pot reproduce aproape toate culorile existente. În acest scop, culorile de bază sunt combinate în diverse proporții. Astfel, într-o culoare secundară, fiecare dintre acestea este reprezentată printr-o valoare care variază de la 0 la 100. O culoare cu saturație maximă este reprezentată prin valoarea 100, iar absența acesteia prin valoarea 0.

Modelul CMYK este utilizat, în principal, pentru tipărirea color. O imagine color este separată în cele patru culori: C, M, Y, K. Imprimarea se face în 4 etape, care corespund celor patru culori. Straturile de culoare au o anumită concentrație, tradusă prin tonurile de culoare proprii imaginii care trebuie tipărită. În general, este imposibilă reproducerea unei imagini de pe monitor (afișată folosind modelul RGB) în mod identic pe hârtie (tipărită folosind modelul CMYK). A se vedea și **RGB**.

CNED (*Centre national d'enseignement à distance, Centrul național de învățământ la distanță*). Universitate online franceză. Oferă un învățământ personalizat, adaptat cerințelor mediului social și industrial, folosind facilitățile puse la dispoziție de noile tehnologii, inclusiv din domeniul tehnicii de calcul (<http://www.cned.fr/>).

Oferta CNED include diverse domenii de formare pentru publicul larg:

instruire adaptată (adresată studenților cu cerințe educaționale speciale: handicap, maladii sau dificultăți școlare grave), instrucție școlară (oferită prin departamentele de grădiniță și învățământ primar, liceu general, liceu tehnologic și liceu profesional), instruirea adulților (numită și instruire continuă, această formă de educație oferă posibilități de formare la toate vârstele și nivelurile de pregătire). De asemenea, CNED este principalul furnizor de programe și materiale educaționale pentru instituțiile franceze care oferă cursuri *online*.

Sistemul francez de educație la distanță este unul de încredere, deoarece diplomele și certificatele obținute în urma finalizării cursurilor sunt acordate la nivel guvernamental, printr-un mod instituționalizat de examinare.

coadă (*queue*). Structură de date de tip listă liniară simplu înlănțuită în care introducerea nodurilor se face la unul dintre capete (numit capul listei), iar ștergerea, consultarea sau modificarea lor la celălalt capăt (numit coada listei). Coada se numește și listă FIFO, deoarece funcționează pe principiul *First In, First Out* (*primul intrat, primul ieșit*). O reprezentare secvențială pentru o coadă se obține folosind un tablou $C[0 \dots n-1]$, tratat ca și cum ar fi circular: după locația $C[n-1]$ urmează locația $C[0]$. A se vedea și **FIFO**, **listă înlănțuită**, **listă liniară** și **structură de date**.

coadă de tipărire (*print queue*). Zonă de memorie (*buffer*) în care sunt

păstrate documentele care urmează a fi tipărite. A se vedea și **buffer de tipărire**.

Coates, Tim (n. 1972). Expert american în *social software* și *blogger*. Blogul său este *plasticbag.org*. A se vedea și **blogger** și **software social**.

COBOL (*COmmon Business-Oriented Language*). Unul dintre cele mai vechi limbaje de programare, ale cărui specificații au fost dezvoltate inițial de programatorul american Grace Hopper (1959). Este utilizat în principal pentru crearea de aplicații folosite în domeniile afacerilor, finanțelor și administrației. A se vedea și **Hopper, Grace Murray**.

Codd, Edgar Frank (1923–2003). Informatician american de origine engleză, cunoscut pentru inventarea modelului relațional pentru gestiunea bazelor de date. Una dintre formele normale, Boyce-Codd, îi poartă numele. În anul 1981 a câștigat premiul Turing. A se vedea și **formă normală** și **modelul relațional**.

cod de bare (*bar code*). Reprezentare a unei informații alfanumerice, într-un format vizual, pe o suprafață, care poate fi citită de un dispozitiv. Pentru a realiza reprezentarea se utilizează un set de simboluri. Codurile de bare se pot întâlni în fiecare zi: pe produsele cumpărate în magazine, pe facturi și chitanțe, în corespondență etc. Prețul unui produs sau serviciu nu este afișat prin intermediul codului de bare, dar el se determină astfel: după ce *scană*erul

„citește” codul de bare, îl trimite unui computer al *punctului de vânzare* (POS, *point of sale*) central, care caută într-o bază de date și returnează prețul curent. Această abordare permite modificarea prețului de vânzare ori de câte ori se dorește, în funcție de situația de pe piață și de strategia firmei.

Necesitatea de a înlocui informațiile alfanumerice cu coduri de bare derivă din lipsa capacității sistemelor informatice de a citi (în toate situațiile) corect și rapid aceste informații. Primele coduri de bare stocau datele în linii paralele separate prin spații, dar astăzi acestea pot fi stocate în modele de puncte, cercuri concentrice sau pot fi ascunse în imagini.

De-a lungul anilor s-a dezvoltat o întreagă industrie de profil, care pune la dispoziție instrumente software pentru crearea codurilor de bare, imprimante pentru tipărirea acestora și *scannere* pentru „citirea” lor. Acestea din urmă s-au dezvoltat în diverse variante, cele mai recente fiind fără fir (*wireless*).

În timp, au fost inventate mai multe standarde pentru crearea codurilor de bare, numite *simbologii*. O simbologie este o schemă care precizează modalitatea de codificare fizică a datelor (*e.g.*, bare, spații). Cele mai utilizate sunt simbologiile liniare, iar dintre acestea cea mai populară este *EAN-13*. Din punct de vedere istoric, ideea utilizării simbolurilor pentru reprezentarea unor date de identificare asociate produselor înregistrate într-un catalog a apărut în anul 1932 (Wallace

Flint, Harvard University). Codurile de bare, în forma lor modernă, au fost dezvoltate începând din 1948 de Bernard Silver și Norman Joseph Woodland. Ele au început să fie folosite în comerț după anul 1966. În 1970 a fost propus standardul UGPIC (*Universal Grocery Product Identification Code*). În 1973 a fost adoptat standardul UPC (*Universal Product Code*) pentru identificarea produselor în Statele Unite, implementat de IBM și dezvoltat de George Laurer. Pe 26 iunie 1974, în SUA, a fost scanat și înregistrat primul produs cu cod de bare. În anul 1981, Departamentul Apărării al Statelor Unite a adoptat formatul de cod de bare *Code 39* pentru industria militară.

Codurile de bare pot fi generate și prin intermediul limbajelor de programare pentru server (e.g., PHP), fiind incluse în paginile Web. În acest scop, sunt utilizate biblioteci dedicate. A se vedea și **EAN-13**.

cod de stare (*status code*). Șiruri de caractere alfanumerice care indică succesul sau eșecul unei acțiuni inițiate de utilizator. A se vedea și **cod de stare HTTP**.

cod de stare HTTP (*HTTP status code*). Cod de stare generat de serverul Web în urma încercării de a înțelege și de a satisface cererea clientului, inclus în răspunsul HTTP. Este format din trei cifre, prima dintre acestea definind *clasa* căreia îi aparține răspunsul. În specificațiile HTTP sunt definite următoarele cinci clase, codul trebuind să

aparțină uneia dintre ele: 1xx (coduri de informare) – cererea a fost primită, iar procesul continuă; 2xx (coduri de succes) – cererea a fost îndeplinită cu succes; 3xx (coduri de redirectionare) – indică o redirectionare a cererii către o altă locație/un alt server; 4xx (coduri de eroare datorate clientului) – cererea include erori de sintaxă sau nu poate fi satisfăcută; 5xx (coduri de eroare datorate serverului) – deși cererea este aparent corectă, serverul nu o poate îndeplini (din diverse motive). Codurile de stare HTTP de tipul 4yz și 5yz identifică erori datorate clientului și, respectiv, serverului, fiind cunoscute și sub numele de *coduri de eroare*. Programatorii au posibilitatea de a gestiona aceste coduri, personalizând astfel mesajele pe care le primește utilizatorul. A se vedea și **răspuns HTTP**.

codec (*coder-decoder*). Dispozitiv sau program de calculator utilizat pentru codificarea și/sau decodificarea unui flux digital de date sau semnal.

CodeLite. IDE pentru limbajele de programare C și C++. *CodeLite* este *cross platform* și *open source*, fiind oferit sub licență GPL (<http://www.codelite.org/>). A se vedea și **IDE**.

(a) **codifica** (*encode*). 1. În programare, activitatea de transpunere a unui algoritm în cod-sursă, realizată prin utilizarea unui limbaj de nivel înalt. A se vedea și **algoritm**, **cod-sursă** și **limbaj de nivel înalt**. 2. A cripta datele transmise prin canale de comunicație,

în scopul măririi securității acestora. A se vedea și **criptare**.

cod Hamming (*Hamming code*). În telecomunicații, cod pentru detectarea și corectarea erorilor, numit după inventatorul său, Richard Hamming. Codurile Hamming pot detecta erori de până la doi biți și corecta erori de un bit. Din această cauză, comunicarea este considerată fiabilă atunci când distanța Hamming dintre modelele de biți transmise și cele recepționate este egală cu 1. A se vedea și **distanța Hamming** și **Hamming, Richard Wesley**.

codificare hard (*hard-coding*). Practică folosită în dezvoltarea software-ului în care datele sunt înglobate direct în codul-sursă al unui program, în loc să fie obținute dintr-o sursă externă.

cod-mașină (*machine code*). Codul rezultat în urma compilării unui program scris într-un limbaj de nivel înalt (e.g., C++) sau a asamblării unui program scris într-un limbaj de asamblare. Limbajul de mașină – denumit și limbaj de programare din prima generație – este alcătuit din instrucțiuni scrise în limbaj binar, care corespund unor operații elementare (aritmetice, logice, transfer de date), specifice unui anumit microprocesor. A se vedea și **asamblor** și **compilator**.

cod nativ (*native code*). Cod specific unui anumit microprocesor. A se vedea și **cod-mașină**.

cod-obiect (*object code*). Cod generat de un interpretor sau un asamblor prin translatarea codului-sursă al unui program. În mod obișnuit, termenul se referă la codul-mașină care poate fi executat direct de CPU-ul (microprocesorul) sistemului. Uneori, codul-obiect poate fi și cod în limbaj de asamblare. A se vedea și **cod-mașină** și **limbaj de asamblare**.

cod redundant (*redundant code*). Secvență de cod care execută o operație deja executată în altă parte (i.e., de către o altă secvență de cod).

cod reentrant (*reentrant code*). Cod (procedură) care se poate executa simultan în contextul mai multor procese. În memorie trebuie păstrată o singură copie a unei proceduri reentrante, care va fi utilizată de toate procesele. Condiții necesare: datele proceselor simultane care apelează procedura reentrantă nu trebuie să interfereze; fiecare proces trebuie să aibă propria zonă de date care să fie accesate separat de apelurile diferite ale aceleiași proceduri reentrante. Nucleele tuturor sistemelor de operare moderne sunt reentrante.

cod-sursă (*source code*). Instrucțiuni sau declarații de program care pot fi citite de oameni, scrise într-un limbaj de programare de nivel înalt sau într-un limbaj de asamblare, care nu pot fi înțelese direct de CPU-ul calculatorului. Codul-sursă este utilizat ca intrare pentru procesele (e.g., compilare, interpretare) care generează un program

executabil (*i.e.*, în cod-mașină). Poate fi convertit într-un fișier executabil (prin compilare) sau convertit și executat direct (*on fly*) de către un interpretor. A se vedea și **cod- mașină, compilator, interpretor, limbaj de asamblare și limbaj de nivel înalt**.

colaborare asincronă (*asynchronous collaboration*). Colaborare prin interacțiuni asincrone (*i.e.*, care nu sunt sincronizate în timp real). De exemplu, în Internet, colaborarea asincronă se face prin intermediul unor servicii/tehnologii ca *e-mail*, wiki și grupuri de știri. A se vedea și **poșta electronică și wiki**.

colectarea gunoiului (*garbage collection*). În informatică, formă de management automat al memoriei. A fost inventată de John McCarthy în 1959, pentru a depăși problema managementului manual al memoriei în LISP. Colectorul de gunoi (*garbage collector*) eliberează automat memoria utilizată de obiecte care nu mai sunt folosite de aplicații.

collate (*a interclasa*). Având și echivalentul *a colaționa*, desemnează compunerea unui set de elemente pe baza a două sau mai multe seturi cu elemente similare, păstrând ordinea existentă în seturile originale.

Colmerauer, Alain (n. 1941). Informatician francez, creatorul limbajului de programare logică PROLOG (1972), alături de Phillipe Roussel. În 1982 a câștigat premiul *la Pomme d'Or du*

Logiciel Français, acordat de Apple France, pentru PROLOG II. A se vedea și **PROLOG**.

COM (*Component Object Model*). Standard și interfață elaborată de Microsoft care permite comunicarea între componentele software folosite la crearea programelor rulate pe platforma Microsoft Windows. Componentele COM pot fi scrise utilizând diverse limbaje. Familia tehnologiilor COM include *COM+*, *Distributed COM* și *ActiveX Controls*. A se vedea și **controale ActiveX**.

comandă (*command*). Directivă lansată către un program de calculator pentru a realiza o sarcină specifică. Poate fi lansată din tastatură (după ce este scrisă de utilizator) sau selectată dintr-un meniu.

COMDEX (*Computer Dealer's Exhibition*). Serie de târguri anuale de calculatoare organizate în luna noiembrie, între anii 1979 și 2003, la Las Vegas (<http://www.comdex.com/>). Declinul COMDEX a început odată cu expansiunea unor târguri similare, precum CeBIT, în Germania, și COMPUTEX, în Taiwan. A se vedea și **CeBIT și COMPUTEX**.

(a) **comenta** (*command out*). În programare, a dezactiva temporar unul sau mai multe rânduri din codul-sursă al unui program prin includerea lor într-un comentariu. A se vedea și **comentariu**.

comentariu (*comment*). Text inclus în codul-sursă al unui program sau al unui

document marcat (e.g., scris în HTML), pentru a fi utilizat pentru documentare de către cel care l-a creat sau de alte persoane. Comentariile sunt utile mai ales atunci când programul sau documentul au dimensiuni mari sau când suferă modificări succesive.

Cele mai multe limbaje de programare și de marcare oferă o sintaxă specifică pentru introducerea comentariilor, astfel încât acestea pot fi recunoscute (și ignorate) de compilator sau interpretor (e.g., în (X)HTML acestea sunt scrise folosind construcția `<!-- comentariu -->`, iar în CSS `/* comentariu */`). A se vedea și (a) **comenta**.

comerț electronic (*e-commerce*). A se vedea **e-commerce**.

Comitetul pentru Standardizarea Tehnologiilor de Învățare (LTSC, *Learning Technology Standards Committee*). Are ca obiect de activitate dezvoltarea standardelor tehnice internaționale specifice *e-Learning*-ului, cu scopul de a facilita dezvoltarea, întreținerea și creșterea interoperabilității sistemelor de instruire și a materialelor educaționale. A se vedea și **e-Learning** și **standarde pentru e-Learning**.

Common Language Runtime (CLR). Mediu de execuție pentru aplicațiile care utilizează .NET Framework și parte a acestui cadru de lucru. Este o implementare Microsoft a standardului *Common Language Infrastructure* (CLI). CLR rulează o formă de

bytecode denumită *Common Intermediate Language (CIL)*. La compilare, compilatorul .NET Framework convertește codul existent (e.g., C#, Visual Basic) în cod CIL. La execuție, compilatorul CLR convertește codul CIL în cod nativ pentru sistemul de operare existent. A se vedea și **.NET Framework**.

compatibil PC (*PC compatible*). Referitor la calculatoarele personale (PC-uri), termenul desemnează orice calculator care respectă specificațiile IBM PC. Calculatoarele de acest tip sunt produse de un număr mare de firme. A se vedea și **calculator personal**.

compilare (*compilation*). Procesul de traducere a codului-sursă într-un format executabil. Compilarea unui program are loc în mai multe faze. În cadrul oricărei faze, are loc transformarea programului dintr-o reprezentare în alta. Principalele faze ale procesului de compilare sunt: analiza lexicală, analiza sintactică, generarea de cod intermediar, optimizarea codului, generarea codului final. De asemenea, compilatorul trebuie să efectueze gestionarea tabelii de simboluri și tratarea erorilor. A se vedea și **compilator**.

compilator (*compiler*). Program care translatează într-un format executabil (i.e., cod-mașină) codul inclus într-un program scris într-un limbaj compilat, înainte de execuția programului. A se vedea și **cod-mașină** și **limbaj compilat**.

complexitatea algoritmilor (*algorithm complexity*). Puterea de calcul necesară implementării unui algoritm. Are două componente principale: complexitatea în timp (care se referă la timpul necesar efectuării calculelor) și complexitatea în spațiu (care se referă la volumul de memorie necesar calculelor), ambele fiind exprimate ca funcții de n , unde n este mărimea datelor de intrare. Complexitatea unui algoritm se notează cu O și reține numai termenul care crește cel mai repede odată cu n . De exemplu, dacă algoritmul se execută în $2^n + n^4 + 2$ unități de timp, complexitatea acestuia în timp va fi $O(2^n)$. Iată complexitățile unor algoritmi cunoscuți: sortarea prin inserție, $O(n^2)$; *heapsort*, $O(n \log_2 n)$; turnurile din Hanoi, $O(2^n)$; căutarea binară, $O(\log_2 n)$; căutarea liniară, $O(n)$; *quick sort*, $O(n \log_2 n)$. A se vedea și **algoritm**.

(a) **comprima** (*compress*). A reduce dimensiunea unui fișier sau a unui mesaj folosind scheme de codificare specifice, implementate în diverse aplicații (de exemplu, programe de arhivare/dezarhivare).

Comprimarea (sau compresia) datelor este utilă deoarece previne consumul unor resurse (e.g., spațiul de stocare pe hard-disk, lățimea de bandă). A se vedea și **comprimare cu pierderi**, **comprimare fără pierderi** și (a) **decomprima**.

comprimarea imaginilor (*image compression*). Metodă utilizată pentru reducerea dimensiunii fișierelor în care sunt stocate imaginile în format *raster*.

Comprimarea fără pierderi conservă datele componente ale imaginii până la ultimul bit, astfel încât forma obținută după decomprimare coincide cu forma de dinaintea realizării ei. Cele mai bune metode de comprimare fără pierderi determină reducerea mărimii imaginilor de două până la trei ori. Comprimarea cu pierderi determină pierderea unor date componente ale imaginii, acestea nemaiputând fi recuperate. În mod obișnuit, metodele de comprimare cu pierderi determină o reducere a dimensiunii imaginilor de opt până la douăzeci și cinci de ori, cu o reducere mai mică sau mai mare (depinzând de imagine) a calității acestora. A se vedea și **formate raster**.

comprimare cu pierderi (*lossy compression*). Procesul de comprimare a unui fișier astfel încât unele date se pierd după comprimare și decomprimare. Algoritmul de codificare trebuie să fie ales astfel încât operația să determine pierderea unor date care nu sunt percepute de utilizator sau care nu influențează foarte mult această percepție, în funcție de natura informațiilor stocate în fișier (e.g., grafică, audio, video) și de limitările receptorilor umani (vizual și auditiv). Adesea, reducerea dimensiunii fișierului este considerabilă, aceasta ajungând la 5% din cea inițială. A se vedea și **comprimare fără pierderi** și (a) **comprima**.

comprimare fără pierderi (*lossless compression*). Procesul de comprimare a unui fișier astfel încât rezultatul

obținut după decompresare să fie identic cu fișierul original (bit cu bit). În mod obișnuit, acest tip de comprimare este folosit în cazul fișierelor care conțin text, date sau cod (deoarece nu trebuie pierdută nici o parte din informațiile incluse în ele). De obicei, dimensiunea fișierului comprimat ajunge la 40% din cea a fișierului original. A se vedea și **comprimare cu pierderi** și (a) **comprima.**

comprimare video (*video compression*). Operația de reducere a cantității de date utilizate pentru a reprezenta imaginii video digitale. A se vedea și **comprimare.**

computer vision (*vedere automată*). Disciplină cu caracter teoretic și aplicativ, sub-domeniu al inteligenței artificiale, se ocupă cu prelucrarea informațiilor vizuale de către calculator, având ca scop construirea unor sisteme artificiale care obțin informații din imaginii. Sistemele obținute sunt utilizate pentru controlul proceselor, detectarea evenimentelor, organizarea informațiilor, modelarea obiectelor și a mediului înconjurător. A se vedea și **inteligență artificială.**

COMPUTEX. Expoziția anuală de calculatoare ținută în Taipei, capitala Taiwanului, începând cu anul 1981 (numele actual l-a primit în 1984) (<http://www.computextaipei.com.tw/>). A se vedea și **CeBIT** și **COMDEX.**

comunicații de date (*data communications*). Concept general care se referă la metode, mecanisme și

dispozitive folosite în transferul de date între calculatoare printr-un mediu de comunicație.

comutare de circuite (*circuit switching*). Tehnologie utilizată în telefonia tradițională, bazată pe linii de transmisie în care se asigură un canal de comunicație între fiecare doi corespondenți. Canalul este un circuit fizic realizat din cabluri și circuite electronice. Pe durata unei convorbiri, canalul poate fi utilizat numai de corespondenții inițiali, ceea ce înseamnă că acesta este un canal dedicat. La sfârșitul convorbirii, canalul utilizat trebuie anulat explicit. A se vedea și **comutare de pachete.**

comutare de pachete (*packet switching*). Tehnologie utilizată predominant în cadrul rețelelor de calculatoare și în domeniul telecomunicațiilor. Pachetul este unitatea utilizată pentru transportul informației în rețele de acest tip. Spre deosebire de comutarea de circuite, care furnizează un canal de transmisie dedicat pentru două noduri ale unei rețele, în tehnica bazată pe comutarea de pachete un canal de transmisie poate fi utilizat de mai multe noduri.

Apariția tehnologiei care stă la baza funcționării Internetului (*packet switching – comutarea de pachete*) este legată de numele unor cercetători ca Leonard Kleinrock, Paul Baran, Donald Davies și Roger Scantlebury. În anul 1967, National Physical Laboratory (Middlesex, Anglia), sub conducerea lui Donald Davies, a experimentat

pentru prima oară o rețea cu comutare de pachete la 768 kbps (denumită *NPL Data Network*).

Cea mai cunoscută aplicație a acestei tehnologii este Internetul, în care comutarea de pachete se realizează conform protocolului IP. A se vedea și **comutare de circuite, Internet și IP.**

comutativitate (*commutativity*). În matematică, proprietate a unei operații binare. O astfel de operație este comutativă dacă prin inversarea operanzilor se obține același rezultat. Astfel, fiind dată operația $*$ definită pe mulțimea M , aceasta este comutativă dacă pentru oricare două elemente aparținând mulțimii M :

$$x * y = y * x.$$

Adunarea și înmulțirea numerelor reale este comutativă. În schimb, scăderea și împărțirea nu sunt. De asemenea, compunerea funcțiilor nu este comutativă. A se vedea și **asociativitate și operator.**

(a) **concatena** (*concatenate*). A pune cap la cap secvențial două elemente compuse. De exemplu, pot fi concatenate două șiruri de caractere (*e.g.*, prin concatenarea șirurilor „limbaj de” și „programare” se obține șirul „limbaj de programare”). În multe limbaje de programare, operatorul de concatenare are simbolul $+$ (plus).

conectivism (*connectivism*). Perspectivă emergentă asupra modului în care cunoașterea poate fi distribuită prin rețele de indivizi și dispozitive. Conform acestui punct de vedere: cunoașterea și învățarea rezidă în

diversitatea perspectivelor individuale; învățarea reprezintă un proces de conectare a surselor de informații; capacitatea de a cunoaște mai mult este mai critică decât ceea ce este cunoscut (învățarea este un proces de creare a cunoașterii); abilitatea de a vedea conexiunile dintre domenii, idei și concepte este o deprindere esențială, individul participând (ca un nod) într-o rețea care învață; acuratețea și actualizarea cunoștințelor trebuie să fie avută în vedere în toate activitățile de învățare; menținerea și îmbunătățirea conexiunilor sunt necesare pentru a facilita învățarea continuă.

Web 2.0 se „potrivește” bine într-un model conectivist al învățării, înțelegând prin aceasta că noul Web dispune de un set de instrumente care pot conduce la un tip de educație ce răspunde nevoilor societății. A se vedea și **Web 2.0.**

configurația paginii (*page setup*). Set de opțiuni care stabilește (*e.g.*, într-un editor de text, așa cum este *Microsoft Word*) modul în care informațiile sunt încadrate în pagină, precum și modul în care aceasta va fi tipărită: orientarea paginii (portret sau peisaj), mărimea marginilor paginii (sus, jos, stânga, dreapta), dimensiunile hârtiei (lățime și înălțime), antetul și subsolul paginii, alinierea verticală (sus, jos, centru, justified). În general, aceste opțiuni pot fi aplicate întregului document, secțiunii curente sau de la pagina curentă până la sfârșitul documentului.

confirmare triplă (*three-way handshake*). Procedeu prin care se realizează conexiunea dintre două procese TCP. Presupune parcurgerea a trei etape în care procesele schimbă anumite pachete de date: procesul-client trimite procesului-server o cerere pentru stabilirea conexiunii, sub forma unui pachet de sincronizare SYN (*cerere de conectare*); după primirea pachetului SYN, serverul trimite (dacă este posibilă conexiunea) un pachet SYN/ACK de acceptare a conexiunii, cerând și sincronizare din partea clientului (*acceptare conectare*); după ce primește pachetul SYN/ACK, procesul-client trimite serverului un pachet ACK (*confirmare de acceptare a conectării*), moment în care se realizează conexiunea. După ce trimite pachetul SYN/ACK, serverul intră într-o perioadă determinată de așteptare, până la primirea pachetului ACK de la client. După expirarea acestei perioade, dacă nu s-a realizat conexiunea, serverul își reia activitatea specifică.

Connection. Antet prezent atât în cererea, cât și în răspunsul HTTP. Specifică modalitatea de comunicație (*i.e.*, prin conexiuni TCP persistente sau nepersistente) aleasă de client și de server. Dacă unul dintre parteneri specifică valoarea `close` pentru acest antet, se va utiliza o conexiune nepersistentă. Dacă nici unul dintre partenerii de comunicație nu precizează o valoare pentru antet, se va utiliza o conexiune persistentă (HTTP/1.1), respectiv una

nepersistentă (HTTP/1.0). A se vedea și **cerere HTTP și răspuns HTTP**.

conexiune (*connection*). Legătură fizică stabilită între două sau mai multe dispozitive de comunicații (*e.g.*, calculatoare) prin intermediul cablurilor electrice, al cablurilor cu fibre optice sau al undelor radio.

Connexions. Mediu pentru dezvoltare colaborativă, partajare liberă și publicare rapidă a conținutului educațional pe Web (<http://cnx.org/>). Materialele educaționale incluse pot fi utilizate gratuit, fiind publicate sub licență CC Attribution. Acestea au o structură neliniară, alcătuită din module care pot fi legate și aranjate în diverse moduri. Astfel, studenții pot să vadă relațiile existente în cadrul topicurilor, ajutând astfel la demonstrarea faptului că știința este în mod natural interconectată, nefiind izolată în clase separate sau manuale. Materialele existente în *Connexions* pot fi partajate și pot constitui baza pentru construirea altor materiale, deoarece sunt reutilizabile: tehnologic (informația este stocată în formatul XML, ceea ce asigură că aceasta va putea fi utilizată pe diverse platforme, în prezent și în viitor); legal (licențele *Creative Commons* pentru conținut deschis facilitează comunicarea materialelor de către autori și le permit altora să le utilizeze/reutilizeze în mod legal, autorii beneficiind de recunoașterea eforturilor lor); educațional (autorii sunt încurajați să scrie modulele ca entități de sine stătătoare, astfel încât alții să le

poată utiliza/reutiliza în diferite cursuri sau contexte). A se vedea și **licențe Creative Commons**.

Consortiul Unicode (*Unicode Consortium*). Organizație non-profit care se ocupă de dezvoltarea, extinderea și promovarea și implementarea standardului Unicode pentru reprezentarea caracterelor în produsele software și standardele moderne (<http://www.unicode.org/>). A se vedea și **Unicode**.

Consortiul World Wide Web (*World Wide Web Consortium*). Denumit și *Consortiul Web* sau *W3C*, este format din peste 500 de instituții academice, organizații comerciale și persoane fizice, având ca scop elaborarea, dezvoltarea, întreținerea și promovarea standardelor referitoare la World Wide Web (<http://www.w3.org/>).

La 1 octombrie 1994, conform înțelegerii dintre MIT și CERN, ia naștere *World Wide Web Consortium*, prima întâlnire a membrilor săi având loc pe 14 decembrie la MIT. Consortiul funcționează sub coordonarea directă a Laboratorului de Știința Calculatoarelor și Inteligență Artificială (CSAIL, *Computer Science and Artificial Intelligence Laboratory*) al MIT, a *Consortiului European de Cercetare în Informatică și Matematică* (ERCIM, *European Research Consortium for Informatics and Mathematics*) și a Universității Keio din Japonia. A se vedea și **World Wide Web**.

constantă (*constant*). În programare, identificator pentru o valoare care nu poate fi schimbată pe parcursul execuției unui program. O convenție populară pentru reprezentarea constantelor constă în utilizarea unor nume scrise cu majuscule și, eventual, folosind liniuța de subliniere (*underscore*), e.g., `O_CONSTANTA`.

Limbajele de programare permit utilizarea a două tipuri de constante: statice (valorile lor trebuie cunoscute în momentul compilării; e.g., limbajul VisualBasic) și dinamice (valorile lor pot fi determinate numai în momentul execuției; e.g., limbajele Java, C++). A se vedea și **program** și **variabilă**.

constructivism social (*social constructivism*). Model al învățării conform căruia cunoașterea este creată de cei care învață, în contextul și ca rezultat al interacțiunii sociale. A fost propus de psihologul rus L.S. Vîgotski (1896–1934), care observă că dezvoltarea cognitivă a individului este, de fapt, o socioconstrucție. În concepția lui Vîgotski, dezvoltarea cognitivă este rezultatul unei duble formări, internă și externă, într-o mișcare orientată de la social la individual. Astfel, învățarea este un proces social, în care cel educat beneficiază de suportul educatorului și/sau al colegilor pentru a avansa.

Competențele social-cognitive pot fi dezvoltate prin utilizarea Internetului. Constructivismul social accentuează negocierea și co-determinarea sensurilor împreună cu alții. Această abordare este reflectată în particular

prin utilizarea instrumentelor Web 2.0, care acționează ca elemente de mediere între studenți și între studenți și educatori, în particular între studenții care, de multe ori, lucrează în locuri diferite și în momente diferite. A se vedea și **Web 2.0**.

constructor (*constructo^r*). În programarea orientată pe obiect (POO), metodă invocată la instanțierea unei clase. Metoda realizează inițializarea proprietăților obiectului, argumentele sale reprezentând valori atribuite acestora prin intermediul instrucțiunilor din corpul metodei. A se vedea și **destructor** și **programarea orientată pe obiect**.

construcționism (*constructionism*). Susținut de Seymour Papert, constituie o dezvoltare particulară a teoriei constructiviste în contextul Web 2.0. Această orientare împărtășește conotațiile constructiviste conform cărora învățarea construiește structurile cunoașterii, adăugând că aceasta se produce în contextul în care cel care învață este angajat conștient în construirea unei entități publice, „indiferent că este vorba de un castel de nisip pe plajă sau de o nouă teorie a universului”. Sistemele de tip software social (în particular, de tip Web 2.0) sunt utilizate pentru construirea unor entități publice, de exemplu prin intermediul unei prezentări video într-un sistem social media, al unei intrări într-un blog (în cazul unei activități individuale) sau prin intermediul unui set de pagini wiki (pentru un individ sau un grup de lucru). A se vedea și **constructivism** și **Web 2.0**.

cont implicit (*default account*). Cont de utilizator predefinit într-un sistem (e.g., program), care permite accesul imediat după punerea în funcțiune/instalarea acestuia. Uneori, numele de utilizator și parola implicite sunt aceleași în fiecare copie a sistemului. După ce sistemul este pus în funcțiune, trebuie fie să se schimbe imediat parola implicită, fie să se dezactiveze contul implicit. A se vedea și **cont de utilizator**.

cont de utilizator (*user account*). Asociat unui utilizator, permite identificarea acestuia printr-un nume de utilizator (*username*). Pentru a folosi contul, utilizatorul trebuie să se autentifice folosind numele de utilizator și o *parolă* (*password*) asociată acestuia. Alături de numele de utilizator și parolă, contul include și alte informații referitoare la utilizator, de exemplu permisiunile acestuia. A se vedea și **nume de utilizator** și **utilizator**.

Content-Type. Unul dintre anteturile cererii și răspunsului HTTP. Desemnează tipul și subtipul MIME al reprezentării resursei solicitate de un client sau transmise de un server (e.g., `Content-Type: text/html`). A se vedea și **cerere HTTP**, **MIME** și **răspuns HTTP**.

contor (*counter*). În Web, miniaplicație sau secvență de cod inclusă în paginile unui site, cu rolul de a ține evidența numărului de utilizatori ai acestuia. A se vedea și **pagină Web**.

contor de program (*program counter*). Cel mai important registru al unității centrale de prelucrare (CPU). Indică instrucțiunea următoare care va fi extrasă pentru execuție. A se vedea și **CPU**, **registru** și **registru de instrucțiuni**.

cont-rădăcină (*root account*). În sistemele de operare UNIX și UNIX-like, este contul al cărui utilizator controlează funcționarea calculatorului. Acesta este administratorul de sistem, numit și *superuser* (*superutilizator*). A se vedea și **director-rădăcină**, **superutilizator** și **UNIX**.

control ActiveX (*ActiveX control*). Componentă software reutilizabilă care utilizează tehnologia ActiveX. Controalele ActiveX pot fi utilizate pentru a include funcții speciale în paginile Web, în aplicațiile desktop (e.g., *Microsoft Office*) și în unele instrumente de dezvoltare software (e.g., *Microsoft Visual Studio*). În particular, pot servi la crearea de aplicații distribuite care pot fi utilizate în Internet prin intermediul browserelor Web. Un astfel de control este descărcat automat și executat în browserul Web (*Internet Explorer*).

Controalele ActiveX pot fi scrise folosind diverse limbaje/medii de programare (MFC, ATL, C#, C++, Visual Basic și Borland Delphi). A se vedea și **ActiveX**.

controlul accesului (*access control*). Mecanism prin care se limitează și controlează accesul în anumite zone sau resurse ale unei instalații sau la un

sistem de informații existent într-un calculator/într-o rețea de calculatoare, pe baza identității utilizatorilor și a apartenenței acestora la anumite grupuri predefinite.

convertor de format (*format converter*). Instrument care produce conversia unui format de fișier în altul. Utilizarea convertoarelor de format mărește mult gradul de audiență al unui document, deoarece (de exemplu) utilizarea unui format optimizat pentru Web facilitează citirea și descărcarea acestuia de un număr mare de persoane.

Cei care creează conținut pentru *e-Learning* pot utiliza un astfel de instrument pentru a transforma formatele uzuale (e.g., *Word*, *PowerPoint*, *Excel*) în formate care pot fi ușor afișate pe Web (e.g., HTML, Flash, Java, PDF).

Unele programe de *authoring* includ convertoare sub forma unor module, astfel încât există posibilitatea salvării (sau exportului) de conținut direct în formate optimizate pentru Web. De asemenea, unele convertoare se instalează ca *add-in-uri* în diverse programe de *authoring*. De exemplu, *iSpring* se instalează în *PowerPoint*.

Cele mai populare convertoare de conținut sunt cele care realizează transformarea fișierelor *PowerPoint* în formate potrivite pentru Web, în special HTML și *Flash*. Conversia la formatul HTML poate fi realizată utilizând și convertorul existent în *PowerPoint* (*File→Save as Web Page ...*).

Convertoarele pentru *Word* transformă fișierele create cu procesorul de text produs de Microsoft în formate optimizate pentru Web, acestea fiind adesea HTML sau PDF. Chiar aplicația *Word* include un astfel de convertor (*File→Save as Web Page...*). Unele convertoare se integrează direct în *Word* (e.g., *WordToWeb*). A se vedea și **authoring tools** și **e-Learning**.

cookie (*cookie*). Bloc de date utilizat în scopul stocării pe calculatorul utilizatorului a unor informații folosite în sesiunile de lucru pe Web.

Blocurile *cookie* sunt utilizate în special pentru păstrarea unor informații privitoare la preferințele utilizatorilor (e.g., tema cromatică preferată), precum și a unor date volatile și care nu au un caracter de confidențialitate (e.g., comenzile de produse în cadrul unui coș de cumpărături). A se vedea și **sesiune Web**.

copyleft (*copyleft*). Concept care (spre deosebire de *copyright*) elimină restricțiile referitoare la distribuția copiilor și a versiunilor modificate ale programelor. Unul dintre creatorii conceptului este Richard Stallman. Cea mai utilizată licență de tip *copyleft* este GNU Public License (GPL). Creative Commons furnizează o licență similară denumită ShareAlike. A se vedea și **Licența Publică Generală GNU** și **licențe Creative Commons**.

CORDIS. Spațiu de informare dedicat cercetării și dezvoltării (R&D, *research & development*) europene, precum

și utilizării rezultatelor R&D (<http://cordis.europa.eu/>). Scopurile principale ale CORDIS sunt următoarele: facilitatea participării în activitățile de cercetare la nivel european; îmbunătățirea exploatarei rezultatelor cercetării, cu accent pe sectoarele cruciale care privesc competitivitatea europeană; promovarea diseminării cunoașterii pentru stimularea performanței inovative.

cont de e-mail (*e-mail account*). Asigură utilizatorului unui serviciu de poștă electronică o adresă de e-mail și un spațiu de stocare (i.e., o căsuță poștală), precum și instrumente pentru gestionarea mesajelor. A se vedea și **adresă de e-mail** și **poștă electronică**.

conținut dinamic (*dynamic content*). Se referă la conținutul inclus în paginile Web generate de serverele de aplicații (e.g., PHP), pe baza unor solicitări ale utilizatorilor (e.g., cuvinte-cheie, termeni selectați dintr-o listă dată) și folosind informații stocate în mod obișnuit în baze de date (e.g., MySQL). A se vedea și **server de aplicații**.

conținut duplicat (*duplicate content*). Se referă la pagini Web distincte în ceea ce privește *design*-ul, dar care au conținuturi practic identice. Caracteristica de duplicare a conținutului poate determina penalizări din partea motoarelor de căutare. A se vedea și **pagină Web**.

coordonate carteziane (*cartesian coordinates*). Set de numere – denumite coordonate – care stabilește în mod

unic poziția unui punct într-un plan, în spațiul tridimensional sau într-un spațiu cu un număr oarecare de dimensiuni. În plan, punctele sunt specificate prin poziția lor față de două axe, notate de obicei cu x (orizontală) și y (verticală), folosind două numere, cunoscute ca abscisă și, respectiv, ordonată. În spațiul tridimensional, se adaugă încă o axă, notată cu z , pentru care este necesar un set alcătuit din trei coordonate.

Folosind coordonatele carteziene, formele geometrice pot fi descrise folosind ecuații algebrice satisfăcute de coordonatele punctelor care se găsesc pe curbele respective. De exemplu, cercul de rază egală cu trei unități cu centrul în originea sistemului de axe de coordonate este descris de ecuația $x^2 + y^2 - 3^2 = 0$.

Sistemul cartezian a fost introdus în anul 1637 de matematicianul și filosoful francez René Descartes, al cărui nume latinizat este Cartesius. Acesta a adus contribuții importante la dezvoltarea analizei matematice, geometriei analitice și cartografiei.

(a) **copia** (*copy*). **1.** A reproduce în memorie (*clipboard*) o porțiune dintr-un document editat. Ulterior, aceasta poate fi inserată (lipită) în documentul respectiv sau într-un document similar. A se vedea și (a) **lipi**. **2.** A reproduce un fișier în alt director aflat pe același disc sau pe un alt suport de stocare.

copil (*child*). În contextul marcării conținutului, se referă la un element

inclus în interiorul altui element. De exemplu, în limbajul HTML, elementul `body` este copil al elementului `html`. În construcția XML `<site><pagina /></site>`, elementul `pagina` este copil al elementului `site`. A se vedea și **element**.

Corel Corporation. Companie canadiană care produce programe de grafică și birou, fondată în 1985 de Michael Cowpland. În 1996, Corel a cumpărat aplicația *WordPerfect* de la Novell. În 2006, Corel a achiziționat InterVideo, furnizor de soluții software multimedia și, odată cu această companie, Ulead Systems. Cele mai cunoscute aplicații furnizate de Corel Corporation sunt *CorelDraw*, *WordPerfect*, *Winzip* și *Paint Shop Pro* (<http://www.corel.com/>).

CorelDraw X3 Graphics Suite. Suită de aplicații folosite în design-ul grafic și DTP (*desktop publishing*), creată de compania Corel. Include, printre alte aplicații, *CorelDraw X3* (editor grafic vectorial), *Corel Photo-Paint X3* (editor de imagine, asemănător cu *Adobe Photoshop*) și *Corel Capture X3* (utilitar folosit pentru obținerea capturilor de ecran). A se vedea și **Corel Corporation** și **DTP**.

COSL (*Center for Open Sustainable Learning*). Inclus în *Departamentul pentru tehnologie educațională* al Universității de Stat din Utah, centrul sprijină prin diverse proiecte deosebite mișcarea *Open Education* (<http://cosl.usu.edu/>).

Course Authoring Tool (CAT, instrumente pentru crearea cursurilor). Instrument care simplifică procesul de implementare a strategiilor instrucționale, prin crearea de conținuturi și scheme de navigare, fără a fi necesare cunoștințe tehnice deosebite. Concret, instrumentele CAT sunt utilizate pentru crearea de pagini, lecții și cursuri.

Instrumentele CAT sunt utilizate în mod obișnuit pentru a crea pagini în care se adaugă text, imagini și alte elemente media, furnizând și un cadru de lucru pentru organizarea paginilor și lecțiilor astfel încât utilizatorii să navigheze cu ușurință. De asemenea: unele instrumente oferă posibilitatea de a testa și obține *feedback*-ul, astfel încât studenții pot urmări propriul progres; alte instrumente pot urmări punctajul realizat de student și pot realiza rapoarte, astfel încât educatorii să monitorizeze progresul studenților.

Deși au facilități deosebite, CAT-urile trebuie utilizate împreună cu instrumente auxiliare denumite *editoare media* (ME, *media editors*). Acestea din urmă sunt necesare pentru crearea elementelor media de conținut (e.g., imagini, grafice, pictograme, animații și secvențe video). De asemenea, în mod obișnuit, instrumentele CAT se bazează pe alte aplicații software pentru a le oferi utilizatorilor materialele de învățare produse. De exemplu, aceste materiale pot fi găzduite de un server Web sau gestionate prin intermediul unui LMS.

Dintre instrumentele CAT pot fi

menționate: *Authorware*, *Macromedia Flash*, *Lectora Publisher* și *Web Course Builder*. A se vedea și **Authorware** și **media editor**.

courseware (*courseware*). Software și materiale utilizate în scop educațional și pentru instruire, distribuite gratuit.

Cox, Alan (n. 1968). Programator britanic implicat în dezvoltarea kernelului Linux. A se vedea și **Linux**.

CPAN (*Comprehensive Perl Archive Network*). Depozit care include un număr destul de important de module software *open source* scrise în Perl (<http://www.cpan.org/>). CPAN are site-uri oglindă în peste 200 de locuri din lumea întreagă. A se vedea și **Perl**.

CPU (*Central Processing Unit, unitate centrală de procesare*). Numit și *procesor* (*processor*), este un circuit electronic complex cu rolul de a executa programele de calculator păstrate în memoria principală. Funcțiile principale ale CPU sunt: extragerea instrucțiunilor din memorie, interpretarea instrucțiunilor (*i.e.*, decodificarea acestora în scopul determinării operațiilor care trebuie executate), citirea datelor (pentru execuția unor instrucțiuni poate fi necesară citirea unor date, din memorie sau de la un dispozitiv de I/E), prelucrarea datelor (pentru execuția unor instrucțiuni poate fi necesară execuția unor operații aritmetice și logice) și scrierea datelor (în memorie sau la un dispozitiv de I/E). Înglobează unitatea de comandă și control (UCC,

care extrage instrucțiunile din memoria principală și determină tipul acestora), unitatea aritmetică și logică (UAL, care realizează operațiile necesare execuției instrucțiunilor) și regiștrii (folosiți pentru depozitarea rezultatelor temporare și a unor informații de control). Componentele procesorului sunt conectate între ele printr-o magistrală internă (*bus*) pe care sunt transmise adrese, date și semnale de control. Conform principiului von Neumann, unitatea centrală de procesare a calculatoarelor clasice execută, la un moment dat, o singură instrucțiune-mașină.

Microprocesorul – introdus la începutul anilor 1970 – este un procesor realizat pe un singur circuit integrat. Dispozitivele de acest tip sunt larg utilizate astăzi nu numai în construcția calculatoarelor, ci și într-un număr mare de alte dispozitive, de la automobilele până la echipamentele medicale și telefoanele mobile. A se vedea și **registru**, **UAL** și **UCC**.

cracker (*cracker*). Persoană care obține acces neautorizat la sisteme de calcul, aplicații sau site-uri Web, trecând peste măsurile de securitate existente, pentru a obține acces la informații confidențiale sau pentru a utiliza în diverse scopuri resursele respective. A nu se confunda cu *hacker*. Țintele obișnuite ale atacurilor crackerilor sunt sistemele importante, care utilizează metode de protecție avansate și conțin informații strict secrete. A se vedea și **hacker** și **haxor**.

Cray, Seymour Roger (1925–1996). Inginer american, proiectant al unor supercomputere, cunoscut ca „tatăl” procesării rapide. A fondat compania de cercetare Cray, aceasta având ca scop proiectarea unor calculatoare foarte rapide. A se vedea și **supercalculator**.

Creative Commons. Având acronimul CC și *motto*-ul *Share, reuse and remix legally*, este o organizație nonprofit care le oferă artiștilor, autorilor, oamenilor de știință, editorilor și educatorilor opțiunea de utilizare a unui *copyright* flexibil pentru operele lor publicate pe Web, furnizând și instrumentele necesare în acest scop.

Organizația *Creative Commons* și proiectul cu același nume au fost lansate la San Francisco în 2001, cu sprijinul *Centrului pentru Domeniul Public* (*Center of Public Domain*) și *Fundației Rockefeller*, precum și al altor organizații și persoane publice. Primul set de licențe CC a fost publicat la 16 decembrie 2002, acestea fiind inspirate parțial din GNU GPL. În anul 2004, CC a primit *Golden Nica Award*, din partea *Prix Ars Electronica*, la secțiunea *Ars Vision*.

După cum afirmă Lawrence Lessig, de la Stanford Law School, unul dintre directorii *Creative Commons*, eforturile CC au ca scop eliminarea „dominantei și tot mai îngrăditoarei culturi a permisiunilor”, în care un material protejat de *copyright* poate fi utilizat numai cu acordul explicit al titularului drepturilor de autor. Acesta a criticat limitările de alegere ale legii dreptului

de autor, în care un autor care publică pe Web poate alege între a accepta drepturile prevăzute de lege sau a permite accesul la opera sa în domeniul public.

Site-ul *Creative Commons* le dă titularilor de drepturi de autor posibilitatea de a oferi publicului o parte din aceste drepturi (păstrându-și cealaltă parte), folosind o serie de mecanisme, printre care eliberarea lucrărilor în domeniul public sau sub licențe de conținut deschis. În acest scop, CC furnizează o serie de licențe libere pe care titularii drepturilor de autor le pot alege atunci când vor să-și publice lucrările pe Web. De asemenea, sunt furnizate și servicii care permit descrierea lucrărilor și licențelor corespunzătoare, facilitându-se astfel localizarea și prelucrarea automată a lucrărilor. A se vedea și **licențe Creative Commons**.

crimeware (*crimeware*). Program periculos, instalat pe un PC fără cunoștința proprietarului acestuia și folosit pentru săvârșirea infracțiunilor cibernetice. Majoritatea acestor programe sunt, de fapt, troieni, utilizați pentru a fura datele personale ale utilizatorilor. A se vedea și **malware** și **troian**.

Crimson Editor. Editor de cod-sursă și HTML gratuit pentru Windows (<http://www.crimsoneditor.com/>). A se vedea și **editor de cod-sursă**.

criptare (*encryption*). Procesul de codificare a datelor, realizată cu scopul de a preveni utilizarea neautorizată a acestora, în special în timpul transmiterii lor prin diverse canale de

comunicație. A se vedea și **criptografie** și **decriptare**.

criptografie (*cryptography*). Domeniu aflat la intersecția dintre matematică, informatică și inginerie, se ocupă cu securizarea informației (inclusiv în cursul transmiterii acesteia prin canale de comunicație) și cu restricționarea accesului într-un sistem informatic. Provine din termenii grecești *kryptos* (ascuns, secret) și *graphein* (a scrie).

Informațiile criptate trebuie transmise destinatarului împreună cu cheia de criptare. De-a lungul vremii au fost imaginate mai multe sisteme criptografice. În criptografia cu chei simetrice (criptografia tradițională), atât expeditorul, cât și destinatarul folosesc aceeași cheie. Un pas important înainte îl reprezintă criptografia cu cheie publică. A se vedea și **criptografie cu cheie publică**.

criptografie cu cheie publică (*public-key cryptography*). Sistem criptografic pe care se bazează SSL (*Secure Sockets Layer*). Rezolvă o problemă fundamentală a criptografiei, anume managementul cheii. În criptografia convențională, ambele părți aflate în comunicație partajează (schimbă) o singură valoare secretă, cunoscută drept *cheie*. În cazul HTTP, cheia este parola utilizatorului. Atât clientul (browserul), cât și serverul trebuie să cunoască parola, înainte de a continua autentificarea. Utilizarea cheilor care trebuie partajate de părțile aflate în comunicație prezintă serioase probleme pentru orice sistem de comunicație. De

exemplu, cum pot cele două părți să schimbe între ele cheile? Criptografia cu cheie publică rezolvă această problemă folosind două chei, una pentru criptarea informației, iar cealaltă pentru decriptare. Nu este posibil ca, știind una dintre chei, să fie descoperită sau calculată cealaltă. Una dintre chei poate fi publică. Astfel, serverul poate trimite o cheie publică unui client (e.g., browser) care o cere. Acesta utilizează cheia pentru criptarea informației pe care, apoi, o trimite criptată serverului. Acesta va utiliza o altă cheie, numită cheie privată, pentru a decripta informația provenită de la client. Modalitatea descrisă nu permite unei terțe părți să decripteze informația trimisă de client, chiar dacă o interceptează, deoarece aceasta nu se află în posesia cheii private. Sistemul poate fi utilizat, de asemenea, pentru autentificarea clientului, în scopul restricționării accesului la serverul Web. A se vedea și **criptografie**.

Crocker, Steve (n. 1944). Informatician american, inventatorul RFC (*Request for Comments*) și autor al primului document de acest tip (RFC 1), precum și al altor documente asemănătoare. În 2002 a primit IEEE Internet Award. A se vedea și **RFC**.

crond. Daemon utilizat în sistemele UNIX/Linux în scopul planificării proceselor. Folosește fișiere de control numite *tabele-cron* sau fișiere *crontab*. Utilizatorul *root* își poate organiza sarcinile administrative în două grupe: sarcini administrative obișnuite, care trebuie executate la intervale regulate

de timp; sarcini specializate, care trebuie executate la un anumit moment de timp. Sarcinile specializate pot fi executate ca intrări în fișierul */etc/crontab* (la care are acces numai utilizatorul *root*). Sarcinile specializate sunt organizate în directoare *cron* speciale. Fiecărui utilizator îi corespunde un fișier *crontab* personal. Pe sistemele Linux Red Hat, fișierele *crontab* ale utilizatorilor sunt localizate în directorul *cron*. Un fișier *crontab* poartă numele utilizatorului (*username*).

În esență, un fișier *crontab* este un tabel cronologic în care fiecare linie corespunde unei intrări. O intrare în fișierul *crontab* specifică informații referitoare la îndeplinirea unei sarcini (de exemplu, execuția unui script sau a unei comenzi). Aceste sarcini se numesc *cron jobs* (*sarcini cron*). Fiecare intrare trebuie specificată într-un mod acceptat de *daemonul crond*. Dacă o intrare este specificată în mod incorect, *daemonul crond* nu va executa sarcina corespunzătoare. O linie care începe cu caracterul # (diez) este tratată drept comentariu. A se vedea și **crontab**.

crontab. Comandă utilizată pentru modificarea fișierelor *crontab* pe sistemele UNIX/Linux. *Daemonul crond* examinează fișierele *crontab* ale utilizatorilor numai la inițializare. Atunci când sunt realizate schimbări în fișierul *crontab* personal folosind comanda *crontab*, un mesaj care indică schimbarea este trimis *daemonului crond*, iar acesta va reexamina fișierul respectiv.

Pentru mărirea securității, poate fi interzisă folosirea comenzii `crontab` unora dintre utilizatori sau tuturor utilizatorilor (cu excepția utilizatorului `root`). În acest scop sunt folosite fișierele `cron.deny` și `cron.allow`. Utilizatorul `root` poate crea, edita sau șterge aceste fișiere. Intrările în cele două fișiere sunt numele de utilizator (`username`), câte unul pe o linie. Cele două fișiere sunt examinate atunci când este invocată comanda `crontab`. Dacă numele utilizatorului se află în fișierul `cron.allow`, acesta poate utiliza comanda `crontab`. Dacă fișierul `cron.allow` există, numele utilizatorului `root` trebuie să apară în acest fișier.

Dacă administratorul de sistem dorește să interzică în mod explicit unui utilizator să utilizeze comanda `crontab`, o poate face introducând numele de utilizator al acestuia în fișierul `cron.deny`. Dacă fișierul `cron.allow` nu există și numele utilizatorului nu se află în fișierul `cron.deny`, comanda `crontab` poate fi folosită. Dacă există numai fișierul `cron.deny` vid, toți utilizatorii pot folosi această comandă. Dacă nici unul dintre cele două fișiere nu există, numai utilizatorul `root` poate folosi comanda `crontab`.

Comanda `crontab` poate fi invocată având ca argument un nume de fișier. În acest caz, ea înlocuiește conținutul fișierului `crontab` existent cu cel al fișierului respectiv. Conținutul acestui fișier trebuie să fie în formatul pe care daemonul `crond` îl așteaptă. Dacă fișierul `crontab` al utilizatorului nu

există, va fi creat un astfel de fișier, conținutul acestuia fiind identic cu cel al fișierului precizat. Dacă fișierul precizat nu există, comanda va invoca editorul implicit. A se vedea și **crond**.

cross-browser (*multi-browser*). Capacitatea unei construcții JavaScript, (X)HTML, CSS, XSLT etc., precum și a unui site sau a unei aplicații Web de a fi rulate pe un număr mare de browsere (e.g., *Internet Explorer*, *Firefox*, *Opera*). A se vedea și **browser Web**.

cross-platform (*multi-platformă*). Caracteristica unei aplicații de a putea fi rulată pe un număr mare de platforme.

CSS (*Cascading Style Sheets, foi de stil în cascadă*). Limbaj care furnizează o modalitate unitară de control a aspectului elementelor prin intermediul cărora sunt construite interfețele cu utilizatorul ale aplicațiilor Web. CSS poate fi utilizat pentru a controla practic orice aspect al paginii, începând cu mărimea fonturilor, culori și spațiere, până la poziționarea elementelor. Folosirea CSS permite o separare clară a prezentării documentelor de conținutul acestora, simplificându-se astfel crearea paginilor Web, precum și întreținerea site-urilor.

O foaie de stil conține declarații ale stilurilor utilizate în document. Este de reținut că, de fapt, stilurile CSS sunt proprietăți ale elementelor (X)HTML. Din acest motiv, se mai numesc și proprietăți de stil.

Consortiul Web a standardizat două niveluri ale foilor de stiluri, primul (CSS1, 1996) oferind o modalitate flexibilă de utilizare a stilurilor în paginile Web, iar al doilea (CSS2, 1998) incluzând facilități suplimentare (e.g., moștenirea stilurilor, poziționări și spațieri, efecte vizuale, integrarea unor tipuri diverse de media, fonturi descărcabile).

În prezent, sunt în curs de redactare specificațiile pentru următorul nivel – CSS3, urmărindu-se standardizarea unor module distincte (e.g., CSS Advanced Layout, CSS Backgrounds and Borders).

O foaie de stiluri CSS constă (pentru orice versiune) dintr-o listă de reguli (numite și instrucțiuni), acestea fiind de două tipuri: reguli-at (*at-rules*) și seturi de reguli (*rule-sets*). Declarațiile pot fi înconjurate de un număr oarecare de spații albe. O regulă-at începe cu cuvântul-cheie *at*, reprezentat prin intermediul caracterului @, urmat imediat de un identificator (e.g., @import, @page), ca în exemplul: @import "stiluri.css". Procesoarele CSS2 vor ignora o regulă @import inclusă într-un bloc sau care nu precede toate seturile de reguli incluse în foaia de stiluri.

O regulă set (numită, pe scurt, *regulă*) reprezintă o instrucțiune referitoare la un aspect stilistic al unuia sau mai multor elemente. Iată un exemplu de regulă CSS: `h1 { color: blue; }`. După cum se poate constata, o regulă CSS include două părți principale: un

selector (e.g., elementul `h1`) și o declarație-bloc (numită uneori, pe scurt, *bloc*) inclusă între acolade (e.g., `color: blue;`). Selectorul – însoțit întotdeauna de o declarație-bloc – reprezintă legătura între documentul XHTML și regulă, specificând care elemente sunt afectate de declarație. Declarația are două părți: proprietatea de stil (e.g., `color`) și valoarea proprietății (e.g., `blue`), stabilind efectul aplicat selectorului (în exemplul anterior, toate elementele `h1` vor fi afectate de declarație, conținutul textual al acestora fiind colorat în albastru). Proprietatea este o caracteristică a selectorului. Specificațiile CSS (inclusiv CSS2) definesc circa 120 de proprietăți, cărora li se pot atribui valori.

Acoladele (`()`) și caracterul „:” oferă browserului posibilitatea de a distinge între selector, proprietate și valoare. Între acolade poate fi prezent orice caracter, cu excepția parantezelor rotunde (`()`), parantezelor drepte (`[]`) și acoladelor (`{ }`). Ghilimelele (simple sau duble) trebuie să fie incluse în perechi (în caz contrar utilizându-se forma lor *escape*), caracterele situate între acestea fiind interpretate ca reprezentând un șir de caractere.

Există trei tipuri de reguli CSS definite de tipul selectorului. Acesta din urmă poate fi: selector (X)HTML, reprezentat de un element (X)HTML (e.g., `h1`, `td`, `p`) – în mod obișnuit, selectorul (X)HTML se folosește într-o regulă pentru a redefini modul de afișare

a conținutului elementului (ca în exemplul anterior); clasă: tipul de regulă corespunzător poate fi aplicat oricărui marcaj (X)HTML, în funcție de dorința utilizatorului – acesta este cel mai adaptabil tip de selector, deoarece poate fi aplicat mai multor marcaje (X)HTML; identificator (*id*): regulile corespunzătoare sunt asemănătoare cu cele de tip clasă, în sensul că pot fi aplicate oricărui marcaj (X)HTML. Deosebirea constă în aceea că o regulă creată folosind ca selector un identificator poate fi aplicată unui singur element al paginii.

Toate foile de stiluri sunt *case-insensitive*, cu excepția acelor părți care nu sunt sub controlul CSS (e.g., numele elementelor XHTML). În CSS2, identificatorii (i.e., numele elementelor, claselor și identificatorilor incluși în selectori) pot conține numai caracterele [A-Za-z0-9], caracterele 161 și superioare acestuia incluse în standardul ISO 10646, plus caracterul liniuță (*hyphen*). De asemenea, identificatorii nu pot începe cu o cifră sau cu caracterul liniuță.

CSCL (*Computer-supported collaborative learning, Învățarea colaborativă susținută de computer*). A se vedea **tipuri de e-Learning**.

CSV (*comma-separated values, valori separate prin virgulă*). Format de date în care câmpurile sunt separate prin caracterul virgulă și fiecare înregistrare prin caracterul *linie nouă* (CRLF). Câmpurile care conțin virgule, caracterul CRLF sau ghilimele trebuie

puse între ghilimele. Formatul CSV este utilizat adesea pentru schimbul de informații incompatibile sau diferite una de alta. Astfel, folosirea acestor fișiere face posibilă comunicarea între sistemele de baze de date care utilizează formate diferite. Formatul CSV este suportat de majoritatea programelor de calcul tabelar (e.g., *Excel, Calc*). Orice limbaj de programare care are funcționalități de intrare (*input*)/ieșire (*output*) și procesare de șiruri va putea citi și scrie fișiere CSV. Formatul CSV are ca avantaj față de XML supraîncărcarea mai redusă. Deci CSV folosește mai puțină lărgime de bandă decât XML atunci când sunt transferate cantități mari de informații, aspect important în implementarea noilor tehnici/tehnologii (e.g., *AJAX*). A se vedea și **AJAX**.

CTRL-Alt-Delete. Apăsarea simultană a celor trei taste produce (pe calculatoarele compatibile IBM PC) deschiderea unei ferestre de dialog în care utilizatorul poate alege din mai multe opțiuni (e.g., oprirea calculatorului, *reboot*-area acestuia, lansarea în execuție a programului *Task Manager*).

(a) **cuantifica** (*quantize*). În procesarea semnalelor, procedura de transformare a unui set de valori continue (e.g., numere reale) într-un set de valori discrete (e.g., numere întregi).

Cuil. Motor de căutare dezvoltat de foști angajați ai Google și IBM (<http://www.cuil.com/>). A se vedea și **motor de căutare**.

Curriki. Organizație nonprofit care a construit primul site pentru Curriculum cu Sursă Deschisă (OSC, *Open Source Curriculum*), incluzând arii curriculare și resurse educaționale pentru liceu (<http://www.curriki.org/>). Denumirea constituie un joc de cuvinte, fiind compusă din *curriculum* și *wiki*. Site-ul este rezultatul muncii depuse de GELC (*Global Education and Learning Community*), proiect online inițiat de Sun Microsystems, Inc. cu scopul de a canaliza munca educatorilor prin intermediul unui efort colaborativ și având ca misiune eliminarea decalajului educațional (*Education Divide*), care îi împiedică pe copiii din lumea întreagă să aibă acces la o educație de calitate.

Curriki este o comunitate online de educatori, dar și una de experți în educație care lucrează împreună pentru a crea materiale educaționale de calitate de care pot beneficia atât educatorii, cât și studenții din întreaga lume. Resursele și cursurile postate pe site pot fi găsite utilizând serviciul de căutare oferit de acesta. Folosind *Curriki*, utilizatorii au posibilitatea de a dezvolta, publica și accesa materiale *open source*, de a accesa gratuit planuri de lecție și cursuri de înaltă calitate, de a crea sau participa la un grup *Curriki*, de a revizui sau de a oferi *feedback* la materialele de învățare partajate cu alte persoane.

Pentru a deveni membru *Curriki*, un utilizator trebuie să se înregistreze. Această calitate îi oferă acces complet la toate resursele site-ului. Un utilizator

înregistrat va avea o secțiune personală în site, *My Curriki*. Utilizatorii pot contribui cu propriile resurse sau partaja adrese Web utile. De asemenea, site-ul oferă instrumente pentru crearea de resurse simple (e.g., un quiz, un exercițiu). Site-ul oferă posibilitatea ca utilizatorii să colaboreze la proiecte curriculare.

CursuriOnline.ro. Oferă cursuri online grupate în trei categorii principale: business, computere și hobby (<http://www.cursurionline.ro/>).

cutie poștală electronică (e-mail box). Corespunde adresei de *e-mail* a unui utilizator. În general, o cutie poștală are rezervată o zonă pe discul calculatorului care furnizează serviciul de *e-mail*.

cuvânt (word). Unitatea de organizare a informației în calculator, cu o lungime fixă. Admite un anumit număr de cifre binare (biți) determinat de considerente legate de costuri, întotdeauna multiplu al numărului 8. Altfel spus, un cuvânt este un număr întreg de octeți (*bytes*). Doi octeți (16 biți) alcătuiesc un cuvânt, iar patru octeți (32 de biți) un cuvânt dublu. Ținând seama de acești termeni, un octet se mai numește și semicuvânt. A se vedea și **octet**.

cuvânt-cheie (keyword). 1. Unul dintre cuvintele sau identificatorii care au semnificații speciale într-un limbaj de programare. În cele mai multe limbaje, un cuvânt-cheie este un cuvânt rezervat. A se vedea și **cuvânt rezervat** și **limbaj de programare**. 2. Cuvânt

inclus într-o cheie utilizată în operațiile de sortare și căutare a înregistrărilor dintr-un tabel al unei baze de date relaționale. A se vedea și **înregistrare**.

3. Cuvânt utilizat pentru căutarea în documentele incluse într-un catalog.

cuvânt rezervat (*reserved word*). Într-un limbaj de programare, cuvânt care are o semnificație specială și care nu poate fi utilizat în alte scopuri. În mod obișnuit, lista de cuvinte rezervate a

unui limbaj de programare include etichete pentru tipurile de date primitive utilizate (*e.g.*, *int*, *float*), identificatori pentru instrucțiuni (*e.g.*, *if*, *do*, *for*), cuvinte-cheie utilizate pentru definirea funcțiilor (*function*) și claselor (*class*), modificatori de acces (*public*, *protected*, *private*), o serie de operatori (*e.g.*, *this*, *new*) etc. A se vedea și **funcție**, **clasă**, **cuvânt-cheie (1)** și **limbaj de programare**.

D

Dahl, Ole-Johan (1931–2002). Programator norvegian, unul dintre părinții programării orientate pe obiect (POO). Este coautor, împreună cu Kristen Nygaard, al limbajelor de programare Simula, care au folosit pentru prima dată noțiunile de clase, subclase, moștenire și creare dinamică de obiecte. Pentru această realizare, cei doi programatori au primit în 2001 premiul Turing. A se vedea și **programare orientată pe obiect**, **Simula** și **Nygaard, Kristen**.

data de expirare (*expiration date*). Data la care expiră o versiune de încercare sau shareware a unei aplicații software. În cazul versiunilor de încercare (și, uneori, al celor shareware), aplicațiile încetează să funcționeze după data expirării. Pentru a le putea folosi în continuare, utilizatorii vor trebui să cumpere versiuni comerciale ale aplicațiilor.

datagramă (*datagram*). Entitate care conține date și informații de livrare necesare pentru trimiterea acestora de la sursă la destinație printr-o rețea cu comutare de pachete (*e.g.*, datagrame UDP, pachete/datagrame IP). A se

vedea și **comutare de pachete**, **pachet IP** și **UDP**.

data mining (*extragerea datelor*). Tehnica și procesul de extragere a unor șabloane ascunse existente în date stocate în baze de date sau în alte depozite, prin utilizarea unor instrumente statistice avansate. *Data mining* este utilizată în marketing, supraveghere, detectarea fraudelor și cercetarea științifică. De exemplu, o aplicație de tip *data mining* poate crea un model care identifică modalitățile preferate de plată (*e.g.*, *cash*, utilizare card de credit), tendințele în *shopping* etc.

date (*data*). Elemente de informație. În limba engleză, *data* (provenit din lat. *datum*) are aceeași formă, atât la plural, cât și la singular.

date dinamice (*dynamic data*). Informații accesate frecvent, actualizate asincron. A se vedea și **date persistente**.

date persistente (*persistent data*). Informații accesate rar și care nu pot fi modificate, stocate pe disc sau într-o bază de date. A se vedea și **date dinamice**.

Davies, Donald Watts (1924–2000). Fizician și informatician britanic, unul dintre cei trei inventatori ai rețelelor cu comutare de pachete, alături de americanii Paul Baran și Leonard Kleinrock. În 1967, National Physical Laboratory (Middlesex, Anglia) experimentează, sub conducerea lui Donald Davies, o rețea (*NPL Data Network*) cu comutare

de pachete la 768 kbps. A se vedea și **comutare de pachete**.

DBMS (*DataBase Management System, Sistem de gestiune a bazelor de date*). A se vedea **SGBD**.

DDL (*Data Definition Language, limbaj de definiere a datelor*). Limbaj utilizat pentru definirea structurii unei baze de date. Modelul unei baze de date (*database model*) este scris în DDL. În general, DDL se referă la orice limbaj formal utilizat pentru descrierea datelor sau a informațiilor de structură (e.g., scheme XML). A se vedea și **bază de date, modelul bazei de date și schemă XML**.

dead link (*legătură moartă*). Legătură către o pagină care nu mai există, a fost mutată sau se află pe un server nefuncțional. Unele motoare de căutare folosesc metode speciale pentru a elimina acest tip de legături.

deadlock (*punct mort*). Situație (impas) în care două sau mai multe procese nu pot rula din cauză că fiecare așteaptă pe unul dintre celelalte să facă ceva. Un exemplu tipic apare în gestionarea bazelor de date, când se produce blocarea tranzacțiilor în situația în care două sau mai multe procese (tranzacții) se așteaptă una pe alta (trebuie remarcat că există tehnici de prevenire a impasului).

debugging (*depanare*). A se vedea (a) **depana**.

decibel. Unitate de măsură logaritmică utilizată în diverse domenii (e.g.,

electronică, acustică) pentru a exprima nivelul valorii unei mărimi fizice față de o valoare de referință specificată. Nivelul relativ se calculează cu formula următoare:

$$L_{\text{db}} = 10 \log (V/V_0),$$

în care L_{db} este nivelul relativ (exprimat în decibeli), v este valoarea mărimii fizice, iar v_0 este valoarea de referință a acesteia. De exemplu, în acustică valoarea de referință a intensității sonore este 10^{-12} W/m². Dacă se utilizează o valoare de 10^{-8} W/m² se obține un nivel al intensității sonore de 40 dB ($10 \log (10^{-8}/10^{-12}) = 4 * 10 = 40$).

Denumirea unității de măsură *decibel* (având simbolul dB și reprezentând a zecea parte dintr-un *bel*) provine de la numele lui Alexander Graham Bell.

(a) **declara** (*declare*). A specifica informații despre variabilele (i.e., tipuri și nume) și funcțiile (i.e., prototipuri) care vor fi utilizate în program. A se vedea și **declarație**.

Declarația BOAI (*Budapest Open Access Initiative Declaration*). Furnizează definiția accesului deschis și fundamentează filosofia acestei mișcări (14 februarie 2002). A se vedea și **acces deschis**.

Declarația de la Berlin (*Berlin Declaration*). Având denumirea completă *Declarația cu privire la Accesul Deschis în Științe exacte și Științe umaniste* (*Berlin Declaration on Open Access in the Sciences and Humanities*), a fost adoptată de

conferința de la Berlin (20–22 octombrie 2003) referitoare la accesul deschis.

Declarația de la Berlin definește accesul liber ca „sursă completă de cunoaștere umană și a patrimoniului cultural care a fost aprobată de comunitatea științifică”. Cei care contribuie la crearea colecțiilor de resurse deschise trebuie să furnizeze rezultate de cercetare originale, date primare (brute), metadata, materiale-sursă, reprezentări digitale ale materialelor grafice, precum și materiale multimedia. A se vedea și **acces deschis**.

declarația tipului de document (*Document Type Declaration*).

Declarație inclusă la începutul unui document (X)HTML, în care este precizat tipul de document utilizat. Este cunoscută ca declarație DOCTYPE. O astfel de declarație precizează URL-ul unui fișier DTD aflat pe site-ul Consorțiului Web, care conține definiția tipului de document respectiv (DTD, *Document Type Definition*). Un DTD este un fișier text care conține reguli prin intermediul cărora sunt stabilite elementele și atributele, valorile acestora, precum și structura unui document (X)HTML (*i.e.*, modul de apariție a elementelor și atributelor, dar și relațiile dintre ele).

Declarația DOCTYPE este utilă browserului Web sau altor dispozitive pentru a identifica versiunea limbajului (X)HTML folosită în scrierea documentului, altfel spus, setul de elemente care pot fi utilizate. În situația

în care sunt folosite elemente neapartinând versiunii (X)HTML declarate, este posibil ca browserul Web să nu le proceseze în maniera în care se așteaptă proiectantul.

În specificația XHTML 1.0 sunt definite trei tipuri de documente: *XHTML 1.0 Strict*; *XHTML 1.0 Transitional* și *XHTML 1.0 Frameset*.

declarație (*declaration*). În programare, specificarea unor informații referitoare la identificatorii unor elemente de limbaj (*e.g.*, constante, variabile, funcții) utilizați într-un program. Declarațiile și definițiile funcțiilor sunt independente: Declarațiile sunt utilizate adesea pentru a putea accesa variabile sau funcții definite în diverse fișiere sau biblioteci. Asocierea efectivă dintre informații și identificatori se realizează în momentul compilării sau execuției. Declarațiile sunt obligatorii în unele limbaje de programare (*e.g.*, C++), iar în altele nu (*e.g.*, PHP). A se vedea și **constantă**, **program** și **variabilă**.

declarație a datelor (*data declaration*). În programare, specificarea caracteristicilor unor variabile care vor fi utilizate într-un program. Modalitățile în care se declară datele se schimbă de la un limbaj de programare la altul, dar, în general, trebuie specificate numele variabilelor, tipul datelor stocate, valorile lor inițiale și dimensiunile acestora. A se vedea și **tip de date** și **variabilă**.

declarație XML (*XML declaration*). Numită și *prolog* sau *preambul*, este utilizată o singură dată (opțional în versiunea 1.0 și obligatoriu în 1.1), la începutul documentului XML, în scopul precizării versiunii XML folosite și codificării documentului. Declarația începe cu `<? xml`, se termină cu `?>` și include următoarele atribute: *version* – utilizat pentru a preciza versiunea XML utilizată în document (1.0 sau 1.1), ca de exemplu `version="1.0"`, utilizarea atributului fiind obligatorie; *encoding* – precizează o submulțime Unicode (e.g., UTF-8, UTF-16), utilizată ca set de caractere în document. Dacă este folosit un caracter neinclus în setul precizat, procesorul XML generează o eroare. Utilizarea atributului este opțională, având valoarea implicită "UTF-8"; *standalone* – stabilește dacă în document este interzisă (valoarea *yes*) sau nu (valoarea *no*) utilizarea unor referințe spre alte documente XML (i.e., referințe externe) care pot conține definiții ale unor componente (i.e., elemente, atribute și entități) folosite în document. Folosirea atributului este opțională, acesta având valoarea implicită *yes*. A se vedea și **XML**.

decompilator (*decompiler*). Program care realizează operația inversă compilării, adică generează codul-sursă într-un limbaj de nivel înalt plecând de la codul în limbaj de asamblare sau de la codul- mașină. A se vedea și **cod-mașină**, **cod-sursă**, **compilator** și **limbaj de asamblare**.

(a) **decomprima** (*decompress*). A reduce la forma inițială conținutul unui fișier comprimat, utilizând o aplicație specifică. A se vedea și (a) **comprima**.

decrement (*decrement*). Valoarea cu care este micșorat un număr prin decrementare. A se vedea și (a) **decrementa** și **increment**.

(a) **decrementa** (*decrement*). A micșora un număr cu o anumită valoare (e.g., 1). A se vedea și **decrement** și (a) **incrementa**.

decriptare (*decryption*). Readucerea datelor criptate la forma inițială (necriptate), prin aplicarea unui algoritm și a unei chei corespunzătoare. A se vedea și **criptare**.

Deep Blue. Calculator programat să joace șah, creat de profesorul Fenghsiung Hsu, de la Universitatea Carnegie Mellon. Deep Blue este primul jucător de șah non-uman care a reușit să învingă un campion mondial. Această performanță a fost realizată în 1996, într-un meci între calculator și Garry Kasparov. Totuși, campionul mondial a câștigat confruntarea cu trei victorii, o remiză și o înfrângere. Ulterior, după ce a fost îmbunătățit, Deep Blue l-a învins pe Kasparov cu 3,5 puncte la 2,5 puncte.

deep link (*legătură în adâncime*). Legătură către o pagină Web care nu este pagina principală a site-ului. Se referă în special la legăturile care duc către pagini situate în „adâncimea”

site-ului. A se vedea și **legătură** și **pagină Web**.

default (*prestabilit*). Cu referire la o variabilă utilizată într-un program, reprezintă valoarea acesteia folosită în situația în care utilizatorul nu specifică alta (e.g., prin intermediul unui formular Web). A se vedea și **program**.

defilare (*scrolling*). Acțiunea de deplasare într-o fereastră, stânga-dreapta (defilare pe orizontală) și sus-jos (defilare pe verticală), care îi permite utilizatorului să vadă conținuturi aflate – la un moment dat – dincolo de marginile ferestrei folosite de un program (e.g., procesor de text, program de calcul tabelar, browser Web).

delicious. Cel mai popular serviciu de *social bookmarking* (<http://delicious.com/>). Se consideră că era Web-ului social a început odată cu lansarea acestuia, la sfârșitul anului 2003.

În 2001, analistul Joshua Schachter (<http://joshua.schachter.org/>) a scris o mică aplicație – denumită *Muxway* – care îi permitea să organizeze colecția sa de resurse *online*, ajunsă atunci la o dimensiune impresionantă (peste 20.000 de intrări), atribuind etichete scurte resurselor. J. Schachter a declarat ziarului spaniol *El Mundo* că site-ul „a apărut dintr-o nevoie personală, am scris codul care a inițiat *del.icio.us* deoarece acumulasem foarte multe link-uri care mă interesau și am dorit să le organizez și să le pot împărți cu ușurință, așa că am scris aplicația și

baza de date inițială.” Deoarece în scurt timp un număr mare de utilizatori i-au vizitat site-ul, Joshua Schachter s-a gândit să dezvolte aplicația pentru ca aceasta să devină utilă tuturor utilizatorilor de Internet. Astfel, pe 15 septembrie 2003 *del.icio.us* a luat locul aplicației *MuxWay*. În 2005, *del.icio.us* a fost achiziționat de Yahoo!. Numărul utilizatorilor a crescut rapid, în septembrie 2006 ajungând – fără nici o promovare – la un milion.

În ceea ce privește profilul demografic, un studiu realizat în 2006 de LeeAnn Prescott a arătat că predomină utilizatorii bărbați (59%), dintre care 41% au vârsta cuprinsă între 25 și 34 de ani, provin din mediul urban sau suburbii, au venituri ridicate, au o educație ridicată și sunt adepți timpurii ai tehnologiei. Conform declarațiilor creatorului serviciului denumit astăzi *delicious*, marea majoritate a utilizatorilor serviciului sunt actori educaționali (educatori, bibliotecari, studenți), *bloggeri*, programatori, persoane dornice să împărtășească și altora resursele descoperite.

Conform unui studiu *online* realizat de *Center for Learning & Performance Technologies* (<http://c4lpt.co.uk/>), *delicious* ocupa în 2008 primul loc într-un top al celor mai apreciate instrumente utilizate pentru învățare în domeniul educației și la locul de muncă (Top 100 Tools for Learning 2008). De asemenea, *delicious* ocupă primul loc în topul primelor 100 de instrumente utilizate pentru învățare la locul de

muncă (Top 100 Tools for WorkPlace Learning 2008), conform unui studiu realizat de aceeași instituție.

Pentru utilizarea serviciului *delicious*, este necesară crearea unui cont. În acest scop sunt solicitate un nume de utilizator, o parolă și o adresă de *e-mail*. La crearea contului, se oferă posibilitatea instalării (opționale) unui *add-on* (*Delicious bookmarks*) în browserul *Firefox*, care – ulterior – va îmbunătăți sistemul de creare a semnelor de carte al navigatorului. Acesta poate fi instalat și ulterior.

Dintre facilitățile oferite de *delicious* pot fi menționate următoarele: crearea unei colecții proprii de semne de carte ale unor resurse Web, marcarea semnelor de carte folosind *tag-uri* (*i.e.*, marcate sau etichete), managementul informațiilor marcate, colaborarea, căutarea și evaluarea resurselor.

Orice pagină *delicious* permite abonarea la un *feed* RSS. Un utilizator poate urmări colecțiile de semne de carte ale altor utilizatori, subcolecții ale acestora sau o listă de semne de carte caracterizate de un anumit marcaj, prin abonare la *feed-urile* corespunzătoare. De asemenea, un utilizator își poate crea propria rețea în care să adauge colecțiile unor prieteni, colegi, colaboratori, *bloggeri* favoriți etc. (despre care știe că se informează în aceleași domenii de interes).

Punerea reciprocă la dispoziție a colecțiilor de semne de carte evidențiază natura socială a aplicației *delicious* și a altor servicii similare. Se

poate spune că o astfel de aplicație reprezintă un sistem de inteligență colectivă, deoarece – permanent – sunt adăugate, modificate sau eliminate informații (semne de carte), implementând în Web o arhitectură a participării.

Pe baza *tag-urilor* *delicious* se pot genera diagrame. Acestea realizează: explorarea vizuală a conturilor și a relațiilor dintre acestea; *mashup-uri* care combină conținutul *delicious* cu cel al altor site-uri.

Dintre avantajele utilizării *delicious* pot fi menționate: independența de platforma folosită (este o aplicație *online*, accesibilă prin intermediul browserului); universalitatea (pot fi salvate semne de carte corespunzătoare oricăror resurse aflate pe orice server Web public, folosind orice calculator cu conexiune la Internet); crearea de comunități cu interese similare (resursele *delicious* pot fi organizate, marcate, partajate, împrumutate de la alții, trimise altora, pot fi urmărite prin RSS etc., rezultând o interacțiune dinamică și permanentă între utilizatori, aceștia grupându-se în funcție de interesele comune); ușurința în utilizare (interfața de administrare a contului și a semnelor de carte este facilă, aplicația fiind accesibilă practic oricărui utilizator). A se vedea și **social bookmarking** și **Social Bookmarking System**.

demon (*daemon*). Program utilizat în sistemele de operare UNIX, care funcționează continuu în background și

realizează operații specifice (e.g., organizare internă, întreținere, furnizare de servicii).

(a) **depana** (*debug*). În programare, a detecta și a corecta erorile (*bug*-urile) existente într-un program de calculator. Termenii *bug* și *debugging* sunt atribuiți lui Grace Murray Hopper, unul dintre primii programatori ai calculatorului Mark I (în anii 1940). A se vedea și **bug** și **depanator**.

depanator (*debugger*). Program de calculator utilizat pentru a testa și depana alt program. În mod obișnuit, un depanator îi oferă utilizatorului posibilitatea de a executa pas cu pas programul depanat, de a opri execuția acestuia la apariția unor evenimente sau a unor instrucțiuni specifice și de a urmări evoluția valorilor variabilelor. A se vedea și **bug** și (a) **depana**.

depășire (*overflow*). Situația în care un calcul produce un rezultat a cărui mărime depășește capacitatea registrului sau a locației în care trebuie stocat.

dependent de mașină (*machine-dependent*). Produs software sau hardware care poate fi utilizat numai pe un anumit tip de calculator. A se vedea și **independent de mașină**.

dependența de Internet (*Internet Addiction Disorder*). Termen introdus în 1996 de psihiatrul american Ivan Goldberg pentru a exprima o nouă formă de dependență a individului, cea față de Internet.

După Goldberg, comportamentele asociate dependenței de Internet

(cunoscută și sub numele de *cyberdependență*) sunt similare celor care caracterizează dependența de alcool sau de substanțe chimice. Psihiatrul a creat *Grupul de Susținere a Dependenților de Internet* – o listă de discuții în care oamenii încearcă să se vindece prin corespondență electronică.

Psihologul Mark Griffiths definește „dependența tehnologică” (cum este cea de Internet) ca o dependență non-chimică ce implică omul și mașina. *Asociația Americană a Psihologilor* a atras atenția asupra „abuzului de Internet”, care s-ar putea număra printre bolile secolului XXI. Cazurile extreme de utilizare a Internetului sunt clare (atunci când oamenii renunță la școală, își pierd locul de muncă etc.), dar nu există o posibilitate de a delimita ceea ce este „normal” de ceea ce este „anormal” în utilizarea Internetului. Unii cred că utilizarea excesivă a Internetului există, dar nu se poate vorbi totuși de o tulburare. Se pare că utilizatorii patologici înregistrează o serie de diferențe față de ceilalți: sunt singuratici, folosesc jocurile *online* mai mult decât alții și utilizează mai des unele elemente „deosebite” ale Internetului (e.g., realitatea virtuală).

Psihanalistul francez Samuel Lepastier consideră că dependența de Internet este falsă. În opinia sa, cei care se refugiază în Internet o fac pentru că aceasta este o modalitate de proiecție pe ecran a problemelor personale. Dependența de Internet este o falsă problemă pentru că lumea își con-

centrează atenția asupra efectelor, și nu asupra cauzelor.

În vreme ce încă există controverse teoretice în legătură cu existența acestei afecțiuni, unii experți au arătat că în cazul în care comportamentul există, el trebuie recunoscut și tratat. După Tonino Cantelmi, psihiatru și psihoterapeut italian, dacă navigarea durează mai mult de 5-6 ore zilnic, aceasta poate degenera într-o serie de simptome ca halucinații, confuzie mentală sau delir, care se pot rezolva numai prin internarea într-o instituție de specialitate. Un studiu efectuat la universitatea Carnegie Mellon, din Pittsburgh, Pennsylvania, SUA, a arătat că persoanele care petrec chiar și numai câteva ore pe săptămână folosind Internetul cunosc niveluri ale depresiei și singurătății mai ridicate decât cele normale. Rezultatele studiului au contrazis premisa acestuia, conform căreia Internetul facilitează comunicarea interumană, oferind posibilitatea lărgirii cercului de prieteni și reducerea singurătății. Concluzia studiului amintit este aceea că, deși Internetul poate facilita realizarea contactelor interumane, el nu trebuie să devină un substitut pentru relațiile sociale tradiționale.

Se cunosc cinci tipuri de dependență de Internet: relațiile cibernetice (manifestate prin conversații impersonale), căutarea de noutăți (*information surfing*), jocurile *online*, comerțul prin Internet și sexul cibernetic/utilizarea site-urilor pentru adulți.

Dependența de Internet poate fi cauzată de fragilitatea identității personale și de probleme în viața socială a indivizilor. De asemenea, IAD poate fi cauzată – ca și alte tipuri de dependență – de existența unor disfuncții în dezvoltarea identității personale. Slăbiciunile auto-identificării, care intră în conflict cu realitatea socială, pot fi un motiv pentru care utilizatorul să vadă lumea *online* ca o modalitate de a scăpa de problemele vieții zilnice. Dacă în unele situații lumea virtuală îi oferă individului un loc pentru a desfășura acțiuni și contacte pe care nu le poate aduce la îndeplinire în viața reală, acesta este un motiv suficient de puternic pentru ca individul să renunțe (cel puțin parțial) la viața socială.

Utilizarea spațiului virtual poate fi în același timp instructivă și distractivă, dar atunci când este practică în exces poate avea consecințe nedorite, determinând chiar o dezangajare din societate și din viața reală.

depozit de obiecte de învățare (LOR, *Learning Objects Repository*). Termen general utilizat pentru a desemna colecțiile de obiecte de învățare.

Obiectele de învățare păstrează în mod obișnuit legături cu aplicațiile care le folosesc, chiar dacă sunt centralizate într-o bibliotecă (depozit local), ceea ce contrazice – cel puțin parțial – ideea refolosirii lor. În ultimii ani se manifestă tendința de eliminare a legăturilor respective și de creare a unor depozite *online* de obiecte de învățare, unele

fiind libere. Este necesar ca materialele/obiectele de învățare utilizate în sistemele virtuale de instruire să fie separate de containerul în care sunt prezentate, deoarece este indicat ca operațiile de creare, actualizare, arhivare, regăsire și management general al acestora să fie ușor de realizat.

Pot fi identificate trei nivele de funcționalitate ale LOR, legate de gradul de modularizare al obiectelor de învățare. Acestea se referă (în ordinea creșterii modularității) la: obiecte de învățare partajabile (obiectele de învățare pot fi regăsite, dar nu poate fi modificat scopul în care sunt folosite); modificarea scopului în care este utilizat conținutul (acesta este structurat sau „spart” în module suficient de mici, astfel încât să fie posibil schimbul de componente); personalizarea metodelor de învățare și/sau a conținutului (conținutul învățării este construit independent de funcțiile de prezentare și control). A se vedea și **obiect de învățare**.

DES (*Data Encryption Standard, standard de criptare a datelor*). Metodă de criptare a datelor utilizată începând cu 1976 ca standard în SUA și, ulterior, pe plan internațional. În unele documentații, se face distincția între DES ca standard și algoritmul care se află la baza acestuia, numit DEA (*Data Encryption Algorithm*). Astăzi, cifrul DES – nesigur, deoarece utilizează o cheie de 56 de biți, considerată prea scurtă – este înlocuit cu AES (*Advanced*

Encryption Standard). A se vedea și **AES**.

(a) **deselecta** (*deselect*). Procesul de anulare a selectării unei opțiuni, unui text etc. A se vedea și (a) **selecta**.

desktop (*suprafață de lucru*). Porțiune de pe ecran care simulează suprafața unui birou prin folosirea unor pictograme și meniuri. Este caracteristică interfețelor grafice bazate pe ferestre (e.g., Microsoft Windows, Mac OS, Linux). A se vedea și **fereastră** și **GUI**.

desktop environment (*mediu desktop*). A se vedea **mediu desktop**.

destructor (*destructor*). În programarea orientată pe obiect (POO), metodă apelată automat în situația în care un obiect creat anterior este în curs de a fi distrus (e.g., nu mai există referințe la obiect). A se vedea și **constructor** și **programare orientată pe obiect**.

detecția erorilor (*error detection*). Termen utilizat în matematică, informatică și telecomunicații, se referă la tehnici care asigură transmiterea fără erori a datelor. Dacă se produc erori, acestea sunt detectate și corectate.

DevGuru. Site care oferă cursuri rapide pentru programatori, utile pentru învățarea unui număr mare de limbaje de programare și de marcare (e.g., ASP, JavaScript, PHP, VBScript, HTML, WML, XSLT), însoțite de exemple (<http://www.devguru.com/>).

DEX – Dicționar Explicativ Român. Dicționar dezvoltat sub licență GPL, oferă definiții, sinonime și antonime pentru o serie de termeni (<http://www.dexro.ro/>). De asemenea, site-ul conține un dicționar englez-român și un dicționar român-englez.

DEX online. Reprezintă transpunerea pe Internet a unor dicționare de prestigiu ale limbii române, fiind creat și întreținut de un grup de voluntari. În prezent, dicționarul este accesibil în patru locuri: *dexonline.ro* (serverul principal, aflat în România și găzduit de firma Elvsoft), *dictionare.edu.ro* (*mirror* găzduit în România de firma Siveco și Portalul SEI), *voronet.dexonline.ro* (*mirror* în San Francisco) și *dex.cartiere.ro* (*mirror*). Baza de definiții a *DEX online* este disponibilă gratuit sub licență publică generală GNU (GPL).

dezinfecție (disinfection). Referitor la un virus și la efectele acestuia, operație care se realizează prin dezactivarea capacității virusului de a opera, îndepărtarea codului virusului sau întoarcerea sistemului la o stare anterioară momentului infecției. În practică, cea mai sigură metodă de eliminare a infecției constă în ștergerea tuturor obiectelor infectate și înlocuirea acestora cu fișiere sigure salvate anterior prin *back up*. A se vedea și **back up** și **virus de calculator**.

(a) **dezinstala (uninstall).** A elimina complet o aplicație sau modul software dintr-un sistem de calcul. Unele

aplicații dispun de module de dezinstalare proprii. În celelalte cazuri, dezinstalarea se poate face folosind facilitățile oferite de sistemul de operare (e.g., *Control Panel*→*Programs*→*Uninstall a program*, în Vista).

DHCP (Dynamic Host Configuration Protocol). Protocol inclus în modelul de referință TCP/IP la nivelul aplicație, utilizat de dispozitive (numite în acest context *clienți DHCP*) pentru a obține informațiile de configurare necesare operării într-o rețea IP. Protocolul permite alocarea dinamică a unor adrese IP temporare calculatoarelor legate într-o rețea conectată la Internet, fără intervenție manuală. A se vedea și **TCP/IP**.

diacritice.com. Dicționar românesc cu două secțiuni: una oferă posibilitatea conversiei unui text fără diacritice într-unul cu diacritice (utilizatorul poate opta între varianta propusă de aplicație și cea originală, pentru toate cuvintele convertite); cealaltă secțiune permite traducerea unui text în/din mai multe limbi – engleză, franceză, română, germană (<http://srv.diacritice.com/>).

diagramă (chart). Reprezentare vizuală 2D sau 3D a unor seturi de date și a relațiilor dintre acestea, în care sunt folosite diverse elemente grafice (e.g., bare, conuri, felii de disc). Cele mai populare tipuri de diagrame sunt: *histogram*, *bar chart*, *pie chart* și *line chart*. A se vedea și **flowchart**.

diagramă de sintaxă (*syntax diagram*). Diagramă utilizată pentru descrierea sintaxei unui limbaj de programare. Este constituită din simboluri grafice: cercuri (includ simboluri speciale), elipse (includ cuvinte rezervate), dreptunghiuri (includ elemente definite prin alte diagrame de sintaxă) și săgeți (indică sensul de parcurgere a diagramei). Orice drum între intrarea și ieșirea în/din diagramă reprezintă o construcție corectă din punct de vedere sintactic. A se vedea și **diagramă și sintaxă**.

diagramă Gantt (*Gantt chart*). Diagramă cu bare care ilustrează etapele unui proiect, utilizată ca instrument de planificare a acestuia. Prima diagramă Gantt a fost realizată în 1896 de Carol Adamiecki, sub numele de *harmonogram*. Diagramele de acest tip au fost popularizate de Henry Gantt, între anii 1910 și 1915. A se vedea și **diagramă**.

dial-up (*dial-up*). Termenul se referă la procesul de conectare temporară la Internet prin intermediul unui modem, folosind linia telefonică. Ținând seama că linia telefonică nu este o legătură dedicată, se poate folosi și în alte scopuri.

Dictionary.com. Dicționar explicativ al limbii engleze, care oferă și alte resurse: sinonime (*Synonyms*), căutări similare (*Related Searches*), precum și un program de traducere a termenilor căutați în unele limbi europene (inclusiv limba română) și asiatice (<http://www.reference.com/>).

dicționar. Instrument românesc *online* extrem de util, care oferă traducerea simultană a unui termen în mai multe limbi străine (<http://www.dictionar.ro/>). Introducând un termen utilizat într-una dintre limbile română, engleză, franceză, germană, spaniolă și italiană, pot fi obținuți echivalenții săi în celelalte limbi enumerate.

dicționar explicativ. Aplicație dezvoltată și coordonată de Lucian Velea. Include un dicționar român-englez, un dicționar englez-român și dicționare explicative (român și englez) (<http://www.ro-en.ro/>).

didactic.ro. Proiect pornit ca o inițiativă a firmei Softwin (<http://www.softwin.ro/>), coordonat de Emil Onea (<http://www.didactic.ro/>). Scopul principal al proiectului este construirea unei baze de materiale educaționale care pot fi descărcate și utilizate de profesori.

Toate materialele publicate sunt protejate de Legea dreptului de autor. Utilizatorii care publică/trimit materiale sunt de acord să cedeze, pe o perioadă nelimitată, la nivelul teritorial al României, în momentul publicării/trimiterii toate drepturile de utilizare, adaptare, modificare și retransmitere a materialelor cu care au contribuit la site. Utilizatorii pot folosi conținutul site-ului numai în scopuri educaționale, fiindu-le interzise copierea, preluarea, reproducerea, publicarea, transmiterea, vânzarea, distribuția parțială, integrală sau modificată a acestuia în alte scopuri.

DIDET (*Digital Libraries for Global Distributed Innovative Design Education and Teamwork*). Depozit de obiecte de învățare utilizat în sprijinul proiectelor de grup ale studenților de la universitățile Strathclyde (UK) și Stanford (USA) (<http://www.didet.ac.uk/>). A se vedea și **obiect de învățare**.

Diffie, Bailey Whitfield (n. 1944). Matematician și criptograf american, unul dintre pionierii criptografiei cu cheie publică. A publicat în 1976, împreună cu Martin Hellman, articolul *Noi direcții în criptografie*. A se vedea și **criptografia cu cheie publică** și **Hellman, Martin Edward**.

digit (*cifră*). Simbol utilizat pentru reprezentarea unui număr într-un sistem de numerație pozițional. Cifrele formează o mulțime al cărei cardinal este egal cu baza sistemului. Exemple: sistemul zecimal (cu baza 10) utilizează zece cifre, de la 0 la 9; sistemul binar (cu baza 2) utilizează două cifre, 0 și 1; sistemul octal (cu baza 8) utilizează opt cifre, de la 0 la 7; sistemul hexazecimal (cu baza 16) utilizează șaisprezece cifre, de la 0 la 9 și de la A la F. A se vedea și **sistem de numerație pozițional**.

digital (*numeric*). În prelucrarea datelor, termen analog cu *binar*, deoarece calculatoarele prelucrează informațiile codificându-le în sistemul binar.

Dijkstra, Wybe Edsger (1930–2002). Informatician și matematician olandez,

cunoscut în special pentru algoritmul drumului minim într-un graf (1959), algoritm care îi poartă numele. Într-un articol scris în 1968 (*A Case against the Go TO Statement*) a luat poziție împotriva utilizării instrucțiunii GO TO, considerând-o dăunătoare.

dimensiune de fișier (*file size*). Exprimă mărimea unui fișier. În mod obișnuit, este exprimată folosind multiplii octetului (*e.g.*, KiloByte: 1 kB = 1024 octeți; MegaByte: 1 MB = 1024 kB sau 1048576 octeți; GigaByte: 1 GB = 1024 MB sau 1073741824 octeți).

Un fișier stocat are o dimensiune logică (indică dimensiunea reală a fișierului) și una fizică (indică spațiul ocupat efectiv pe dispozitivul de stocare). Dimensiunea fizică este adesea mai mare decât cea logică, situație explicată prin faptul că spațiul de stocare este rezervat pe unități de alocare, ultima fiind în mod obișnuit parțial ocupată. A se vedea și **fișier**.

dimensiune de font (*font size*). Mărimea caracterelor aparținând unui font, exprimată în puncte (*e.g.*, 12 pt.). A se vedea și **font**.

director (*director*) A se vedea **dosar**.

director curent (*current directory*). Directorul de lucru al utilizatorului. Este localizat la sfârșitul căii de directoare. Fișierele nou create vor fi salvate implicit în directorul curent. A se vedea și **director**.

director de feed-uri (*feed directory*). Numit și *director RSS* (*RSS directory*),

este o colecție *online* de *feed*-uri care le oferă utilizatorilor posibilitatea de a le citi și de a subscrie (de a se abona) la acestea. Orice director oferă modalități accesibile pentru înregistrarea *feed*-urilor. În cele mai multe cazuri, operația amintită este condiționată de existența unui cont, care va fi creat gratuit.

Iată câteva directoare de *feed*-uri populare: *Technorati*, *Bloglines*, *Blogdigger*; *Syndic8*, *RSSfeeds*, *Search4RSS*, *RSS Locator*. Directorul *Colecție de bloguri românești/RSS* (<http://www.timsoft.ro/ro/rss.php>), oferit de TimSoft, include o colecție de bloguri educaționale românești însoțite de *feed*-urile asociate. Se oferă posibilitatea generării documentelor OPLM. A se vedea și **feed**.

director de bază (*home directory*). Director asociat unui cont de utilizator în sistemele de operare multi-user, care conține fișierele acestuia (e.g., fișiere de date, fișiere de configurare). În Windows 2000, XP și 2003, directorul de bază are calea `<root>/Documents and Settings /<username>`, în Vista `<root>/Users /<username>`, în cele bazate pe UNIX `<root>/home/<user name>`, în Linux `/home/<username>`, iar în Mac OS `/Users/<username>`. A se vedea și **director**.

director partajat (*shared directory*). Într-o rețea locală, director sau folder accesibil mai multor utilizatori. Partajarea unor directoare este cea mai utilizată metodă pentru accesarea și utilizarea în comun a informațiilor,

într-un LAN. A se vedea și **director și LAN**.

director-părinte (*parent directory*). Directorul în care este situat un alt director. În cele mai multe sisteme de fișiere, un director are o intrare notată în mod obișnuit „.” (două puncte consecutive), care desemnează directorul-părinte al acestuia. Directorul-rădăcină este propriul său director-părinte. A se vedea și **director și director-rădăcină**.

director-rădăcină (*root directory*). Punctul de acces într-o structură ierarhică arborescentă de directoare existentă pe un disc. În sistemele de operare UNIX și UNIX-like, directorul-rădăcină este notat cu `/`. Toate intrările sistemului de fișiere, inclusiv partițiile montate, sunt „ramuri” ale acestei rădăcini. Directorul `/root` nu este directorul-rădăcină (în sensul precizat aici), ci mai degrabă directorul *home* al contului *superuser* (cunoscut și ca *root*). În DOS și Windows, fiecare partiție are atribuită o literă (e.g., `C:\`), neexistând o rădăcină comună tuturor partițiilor. A se vedea și **cont-rădăcină și director de bază**.

director Web (*Web directory*). Aplicație Web specializată în stocarea unor informații (inclusiv legături) despre un număr mare de alte site-uri, clasificate după diverse criterii.

Înscrierea site-urilor în unele directoare Web este gratuită, în vreme ce altele percep o taxă. Directoarele Web se pot clasifica în funcție de categoria în care

pot fi incluse site-urile înscrise. Astfel, una dintre categoriile importante este reprezentată de directoarele specializate în înscrierea site-urilor de comerț electronic (*shopping directory*). De asemenea, există directoare Web generale, așa cum sunt *Yahoo! Directory* și *Dmoz*. Iată și alte directoare Web cunoscute: *AboutUs.org* (director bazat pe *wiki*), *GoGuides*, *Starting Point Directory*, *World Wide Web Virtual Library* (cel mai vechi director).

Există peste 300 de directoare Web românești, cele mai multe gratuite. Iată câteva din ultima categorie: *Ponturi fierbinți*, *Smart Net Book*, *Tre.ro*, *Santamia*, *Looker*, *Link Directory*, *Smarty*, *TotalTop*, *AddSite*, *Click Link*, *Director Românesc*. A se vedea și **Dmoz**, **paid inclusion** și **Yahoo! Directory**.

dirty URL (*URL murdar*). Un URL lung, complex, dificil de citit și de ținut minte, care conține caractere neconvenționale (*e.g.*, semnul egal, semnul întrebării, ampersand). În general, spiderii motoarelor de căutare nu indexează paginile care au astfel de URL-uri. A se vedea și **TinyURL**.

dispozitiv de indicare (*pointing device*). Dispozitiv de intrare care permite unui utilizator să introducă date spațiale într-un calculator. Interfețele grafice cu utilizatorul (GUI) îi oferă acestuia posibilitatea de a introduce astfel de date folosind gesturi fizice exercitate asupra dispozitivului (*e.g.*, deplasare, click, dublu-click, tragere și

lansare). Mișcările dispozitivului de indicare se reflectă pe ecran prin producerea unor mișcări ale cursorului atașat acestuia, precum și prin alte modificări de natură vizuală. Iată câteva dispozitive de indicare: *mouse* (cel mai răspândit), tableta grafică, *touchscreen*, *trackball*, *touchpad*, *joystick*. A se vedea și **dispozitive I/E**, **GUI**, **joystick**, **mouse**, **trackball** și **trackpad**.

dispozitiv de ieșire (*output devices*). A se vedea **dispozitive I/E**.

dispozitiv de intrare (*input devices*). A se vedea **dispozitive I/E**.

dispozitive I/E (*I/O devices*). Termen general care desemnează dispozitivele (sau perifericele) prin care un calculator primește informații din lumea exterioară (dispozitive de intrare, *input devices*) și, respectiv, prin care returnează rezultatele calculelor efectuate (dispozitive de ieșire, *output devices*). Cele mai întâlnite dispozitive de introducere a datelor (informațiilor) în calculator sunt *mouse*-ul și tastatura, iar cele mai folosite dispozitive prin care acesta îi prezintă informații utilizatorului sunt monitorul, imprimanta și dispozitivele de generare a sunetului.

dispozitiv periferic (*peripheral device*). Dispozitiv conectat la calculator și controlat de microprocesor (*e.g.*, unitate de disc, imprimantă, modem). În categoria dispozitivelor periferice intră și cele de I/E. A se vedea și **dispozitive I/E**.

dispozitiv virtual (*virtual device*). Dispozitiv la care se poate face referire, dar care nu are o existență fizică. Două dintre cele mai cunoscute dispozitive virtuale sunt memoria virtuală și imprimanta virtuală. A se vedea și **imprimantă virtuală** și **memorie virtuală**.

distanța Hamming (*Hamming distance*). În teoria informației, reprezintă numărul minim de substituții necesar pentru a schimba un șir în alt șir de lungime egală sau numărul minim de erori care au schimbat un șir în celălalt. Deci distanța Hamming (inventată de matematicianul american Richard Hamming) dintre două șiruri cu lungime egală reprezintă numărul de poziții ale căror simboluri corespunzătoare sunt diferite. A se vedea și **cod Hamming** și **Hamming, Richard Wesley**.

distribuție Linux (*Linux distribution*). Termenul desemnează sistemul de operare Linux la care se adaugă o mare cantitate de software (dezvoltat sub licență GNU GPL), organizat în pachete. În prezent, în întreaga lume sunt dezvoltate câteva sute de distribuții Linux. Acestea pot fi clasificate în două mari categorii: susținute de firme (*e.g.*, *Fedora*, susținută de RedHat; *SUSE Linux*, susținută de Novell; *Ubuntu*, susținută de Canonical Ltd.; *Mandriva Linux*, susținută de Mandriva) și dezvoltate de comunități (*e.g.*, *Debian*, *Gento*). A se vedea și **Linux**.

divide et impera (*dezbină și stăpânește*). Metodă generală de

elaborare a algoritmilor. Constă în împărțirea repetată a unei probleme cu dimensiune mare în două sau mai multe subprobleme (*divide*), rezolvarea acestora (*impera*) și combinarea soluțiilor subproblemelor pentru obținerea soluției problemei inițiale. De multe ori, subproblemele sunt de același tip, și pentru fiecare se poate aplica aceeași metodă a descompunerii în subprobleme, până când – în urma descompunerilor repetate – se ajunge la probleme cu rezolvare imediată. Metoda admite o implementare recursivă, folosind un principiu general pe baza căruia se elaborează algoritmi recursivi: ce se întâmplă la un nivel se întâmplă la orice nivel (asigurându-se condițiile de terminare). Astfel, în cadrul unui algoritm elaborat folosind metoda *divide et impera*, la un anumit nivel există două posibilități: (1) s-a ajuns la o problemă care admite o rezolvare imediată, caz în care se rezolvă și se revine din apel (condiția de terminare); (2) nu s-a ajuns în situația descrisă la punctul anterior, caz în care se descompune problema în două sau mai multe subprobleme, pentru fiecare se reapelează recursiv funcția, se combină rezultatele și se revine din apel. A se vedea și **programare dinamică**.

Django. Cadru de lucru *open source*, denumit astfel după chitaristul Django Reinhardt și oferit sub licență BSD în 2005, utilizat pentru dezvoltarea aplicațiilor Web în limbajul *Python* (<http://www.djangoproject.com/>). A se vedea și **Python**.

DML (*Data Manipulation Language, limbaj de manipulare a datelor*). Denumit și *limbaj de interogare (query language)*, este utilizat în SGBD-uri pentru regăsirea, actualizarea și interogarea bazelor de date. Adesea, aceste limbaje permit efectuarea unor calcule complexe, facilitând astfel generarea rapoartelor. A se vedea și **SGBD**.

Dmoz. Denumit oficial *Open Directory Project (ODP)*, este un director Web creat și deținut de Netscape (Time Warner), dezvoltat și menținut de o comunitate de editori voluntari (<http://www.dmoz.org/>). Site-urile înscrise sunt grupate în categorii și subcategorii. *Dmoz* are și o secțiune în limba română. A se vedea și **director Web**.

DNS (*Domain Name System, sistemul numelor de domeniu*). Sistem distribuit de păstrare și interogare a unor date arbitrare într-o structură ierarhică, cea mai cunoscută aplicație a acestuia fiind gestionarea domeniilor în Internet.

DNS a fost dezvoltat de IETF ca standard pentru convertirea adreselor IP ale gazdelor existente într-o rețea TCP/IP. Sistemul atribuie fiecărei adrese IP un nume (sau identificator textual), cunoscut și ca adresă DNS. De asemenea, DNS stabilește și menține ordinea în adresare. O adresă DNS este construită din domenii și subdomenii Internet care identifică o gazdă sau o rețea particulară. Alături de adrese IP, organizația InterNIC atribuie și înregistrează adrese DNS.

Sistemul organizează adresele DNS pe baza unei scheme arborescente (ierarhice) de nume simbolice și a unui sistem de baze de date distribuite pentru implementarea acestei scheme.

Teoretic, un singur calculator server poate conține întreaga bază de date DNS. În practică, pentru a se evita încărcarea excesivă a unui singur calculator, s-a convenit ca baza de date DNS să fie împărțită în mai multe zone distincte, obținându-se o rețea de servere DNS. O astfel de zonă are un server de nume primar și unul sau mai multe servere de nume secundare, care copiază informația de pe serverul primar. În general, procesul de identificare după nume a unui calculator necesită consultarea mai multor servere DNS. În baza de date a unui server DNS, fiecărui domeniu prezent îi corespund mai multe înregistrări de resurse, cel mai des întâlnită (numită înregistrare de tipul A) fiind cea care asociază domeniului respectiv adresa IP a acestuia.

Serverele DNS au capacitatea de a delega autoritatea de rezolvare (sau rezoluție) a numelor de domeniu (adică de determinare a adresei IP corespunzătoare). Astfel, pentru adresa *www.exemplu.ro*, serverul DNS va delega autoritatea spre serverul care administrează TLD-ul *.ro*, care va delega autoritatea spre serverul DNS *exemplu.ro*, care va rezolva domeniul final *www.exemplu.ro*.

O cerere de interogare a bazei de date DNS este rezolvată de serverul cel mai apropiat. Dacă nu se poate rezolva

numele de domeniu, cererea se trimite serverului superior în ierarhie, în mod recursiv. Serverele DNS păstrează cererile într-o zonă de *cache*, folosită ulterior pentru rezolvarea mai rapidă a altor cereri.

O implementare TCP/IP conține o rutină denumită *name resolver*, cu rolul de a interoga serverul de nume în scopul obținerii translatații nume-adresă IP (*i.e.*, realizării rezoluției de nume) și invers. A se vedea și **InterNIC** și **nume de domeniu**.

document (*document*). 1. În contextul Web, un document (X)HTML este asimilat conceptului de pagină Web. Prin extensie, se utilizează termenii document XML, document RDF, document RSS, document Atom etc. 2. În contextul procesării textului, un document conține text formatat, creat cu diverse aplicații (*e.g.*, *Word*) și în variate formate.

document (X)HTML (*(X)HTML document*). Document scris folosind limbajul de marcare (X)HTML. Un astfel de document afișat de un browser poartă numele de pagină Web.

Un document (X)HTML trebuie să conțină (între elementele `<html>` și `</html>`, corespunzătoare instanței elementului-rădăcină) următoarele două secțiuni: *antet* (*head*; este precizată prin intermediul elementului *head*; oferă o serie de informații despre document; conținutul marcajelor incluse în *antet* nu este afișat în pagină); *corp* (*body*; este precizată prin

intermediul elementului *body*; include conținutul afișat al paginii Web). Elementele *head* și *body* trebuie să apară în document în ordinea enumerării lor, câte o singură dată. Marcajul de închidere `</html>` trebuie plasat la sfârșitul documentului XHTML.

Document Freedom Day (*Ziua libertății documentelor*). A se vedea **Ziua libertății documentelor**.

Dojo. Unul dintre proiectele găzduite, din anul 2004, de *Dojo Foundation* (<http://dojotoolkit.org/>), alături de *OpenRecord*, este un set de instrumente JavaScript *open source*, dezvoltat sub licențele *AFL* (*Academic Free License*) și *BSD* și utilizat pentru crearea interfețelor dinamice, precum și a aplicațiilor Web profesionale bazate pe *AJAX*. Acest set de instrumente este dezvoltat sub conducerea lui Alex Russell (<http://alex.dojotoolkit.org/>). *Dojo* este compatibil cu majoritatea browserelor utilizate astăzi. Printre altele, *Dojo* asigură: un cadru de lucru pentru crearea de către utilizatori a propriilor *widget*-uri; o bibliotecă de *widget*-uri gata realizate (*Rich Text Editor*, *Content Panel*, *Split Panel*, *Tabs*, *Fisheye List*, *Sorting Table*, *Google/Yahoo Maps*, *Tree*, *Dialog/Wizard*); suport pentru schimbul asincron de mesaje între client și server (*AJAX*); suport pentru manipularea URL-urilor în browser. A se vedea și **AJAX** și **JavaScript**.

DOM (*Document Object Model*) 1. Model care permite accesarea (*i.e.*,

consultarea și/sau modificarea) elementelor și atributelor incluse în documentele XML și (X)HTML. În mod obișnuit, prin DOM se înțelege modelul standardizat de Consorțiul Web (W3C). Înainte de a deveni specificație W3C, modelul DOM a avut o lungă istorie, intrinsec legată de evoluția navigatoarelor Web, în special a celor create de firmele Netscape și Microsoft. Pentru a caracteriza evoluția modelului DOM, atât din punct de vedere istoric, cât și din cel al posibilităților de utilizare, se folosesc termenii Dom Level 0, DOM intermediar și DOM avansat. **2.** Standard al W3C, este un API independent de platformă și limbaj utilizat pentru consultarea și/sau modificarea documentelor XML și XHTML. Acest API este numit și DOM avansat, pentru a-l deosebi de DOM-ul clasic (primul model DOM, cunoscut și sub numele de DOM de nivel 0) și de DOM-urile intermediare (implementate în versiunea a 4-a a browserelor *Navigator* și *Internet Explorer*). DOM furnizează o reprezentare structurală a unui document, oferind posibilitatea modificării dinamice a conținutului, a structurii și a stilurilor acestuia prin intermediul programelor/scripturilor. În DOM, structura documentelor este arborescentă, acestea fiind reprezentate folosind date de tip *obiect*. Deoarece modelul este independent de limbaj, interfețele ale căror definiții le include pot fi implementate practic în orice limbaj de programare (e.g., JavaScript, PHP).

Consoțiul Web precizează următoarele trei niveluri de specificare a API-ului DOM: nivelul 1 (DOM Level 1) – include un *nucleu* (DOM Core) care definește funcționalități de bază utilizate pentru procesarea documentelor XML și o extensie (DOM HTML) pentru procesarea documentelor HTML; nivelul 2 (DOM Level 2) – extinde și modularizează interfețele DOM de nivel 1; nivelul 3 (DOM Level 3) – oferă un set extins de funcționalități conținute în diverse module, unele fiind standardizate, iar altele în curs de dezvoltare.

Interfețele definite în DOM și utilizate pentru procesarea documentelor XML și (X)HTML sunt similare claselor abstracte de bază din C++ sau interfețelor Java. Acestea oferă posibilitatea unei aplicații să acceseze reprezentarea internă a unui document. Deoarece definițiile interfețelor DOM nu presupun o implementare concretă a acestora, există libertatea ca în fiecare limbaj să se folosească o reprezentare oarecare a documentelor, impunându-se o singură condiție, anume respectarea specificațiilor elaborate de W3C. Pentru definirea interfețelor DOM, este utilizat limbajul declarativ OMG IDL (*Interface Description Language*), definit în specificațiile CORBA 2.2.

Nucleul modelului DOM (DOM Core) definește un set minimal de interfețe care pot fi utilizate în scopul accesării/manipulării conținutului nodurilor arborelui DOM asociat unui document

XML sau (X)HTML. În nucleul DOM sunt definite două categorii de interfețe: fundamentale și extinse. Interfețele fundamentale trebuie implementate în mod obligatoriu, conform specificațiilor DOM.

DOM reprezintă documentele XML și (X)HTML ca pe o ierarhie de obiecte-nod (de tip `Node`) care pot implementa, de asemenea, alte interfețe, mai specializate. În ierarhia DOM, unele tipuri de noduri pot avea noduri-copii de diverse tipuri, altele fiind noduri-frunză (*i.e.*, fără copii).

DOM-ul de nivel 1 este implementat în toate browserele actuale. În schimb, modulele incluse în API-urile DOM de nivel 2 și 3 sunt implementate în proporții diferite de la un browser la altul.

3. Metodă de procesare a documentelor XML. Este una dintre metodele care utilizează procesarea într-un arbore, cu următoarele caracteristici: încarcă întregul document în memorie (astfel, este posibil accesul aleatoriu la toate nodurile acestuia); permite editarea documentului; reprezintă o procesare complexă, necesitând resurse mari de memorie.

Spre deosebire de SAX, metoda DOM încarcă în memorie întregul document XML, ceea ce determină un consum ridicat de resurse (memoria ocupată este de circa trei ori mai mare decât mărimea documentului procesat). După încărcare, arborele DOM al documentului poate fi accesat într-un mod similar celui în care se realizează

accesarea unui tablou multidimensional, fiind posibilă și editarea acestuia (*i.e.*, se pot citi, adăuga sau șterge noduri). În plus, metoda DOM oferă posibilitatea construcției unui arbore XML nou, precum și copierea unui subarbore al unui arbore în altul.

Este indicat să se utilizeze metoda DOM atunci când documentele XML sunt mici sau când se dorește să se navigheze în arborele DOM sau să se realizeze modificarea acestuia. A se vedea și **SAX**.

domeniu (*domain*). Modalitate de organizare a resurselor în Internet. Domeniile sunt dispuse pe mai multe niveluri ierarhice, cele inferioare referindu-se la resurse dintr-un domeniu de nivel superior. Corespondența dintre adresele IP ale gazdelor și numele simbolice compuse din nume de domeniu și nume de calculatoare se realizează prin intermediul sistemului de nume de domenii (DNS). Domeniile de cel mai înalt nivel sunt incluse în următoarele categorii: infrastructură (ARPA); generice (*.biz*, *.com*, *.info*, *.name*, *.net*, *.org*, *.pro*); state (*.ro*, *.us*, *.de*, *.ch*, *.fr*, *.uk*); rezervate (*.example*, *.invalid*, *.localhost*, *.test*); sponsorizate (*.aero*, *.edu*, *.gov*, *.int*, *.jobs*, *.mil*, *.tel*); pseudo-domenii (*.bitnet*, *.local*, *.root*, *.uucp*). A se vedea și **DNS** și **nume de domeniu**.

DOM intermediar (*intermediate DOM*). Modele DOM proprietate dezvoltate de Netscape și Microsoft. Spre deosebire de DOM de nivel 0, care asigură accesul la un număr limitat de

elemente HTML, DOM-urile intermediare, implementate începând cu versiunea 4 a browserelor *Navigator* și *Internet Explorer*, aveau ca scop realizarea accesului la toate elementele HTML, dându-le astfel programatorilor Web posibilitatea de a le manipula proprietățile. În continuare sunt descrise pe scurt cele două DOM-uri.

Modelul DOM intermediar propus de Netscape a fost inclus în *Navigator 4*. Pentru acest model este fundamental conceptul de *strat (layer)*. În această versiune a navigatorului este introdus elementul *layer*, care nu este suportat de alte browsere (nici chiar de *Mozilla*). Spre deosebire de alte DOM-uri, pentru a avea acces la un formular, la o imagine etc., este necesar ca mai întâi să se obțină (prin intermediul obiectului `document.layers`) accesul la stratul care include elementele respective. Acest DOM a fost abandonat în versiunile următoare ale navigatorului Netscape.

Modelul DOM intermediar propus de Microsoft a fost inclus în browserul *Internet Explorer*, începând cu versiunea 4 a acestuia. De asemenea, browserul *Opera*, începând cu versiunea 6, suportă modelul amintit. În cadrul acestuia, obiectul `document.all` oferă acces la elementele documentului. Pentru compatibilitate, acest model a fost inclus în versiunile următoare ale browserului *Internet Explorer*. Pentru a-l deosebi de noul DOM (DOM Level 1) utilizat începând cu versiunea 5 a browserului *Internet Explorer*,

Microsoft se referă la acest DOM ca la DHTML Object Model. A se vedea și **DOM**.

DOM Level 0 (*DOM de nivel 0*). Prima variantă a modelului DOM, introdusă în browserul *Netscape 2* odată cu suportul pentru limbajul JavaScript. Funcția avută în vedere la crearea DOM a fost aceea ca modelul să ofere acces la unele elemente și atribute ale lor incluse în documentele HTML. DOM de nivel 0 a fost introdus apoi în versiunea 3 a browserelor *Netscape* și *Internet Explorer*. Acest DOM mai este cunoscut sub numele de DOM clasic sau DOM JavaScript tradițional. Pentru motive legate de compatibilitate, chiar și cele mai moderne browsere (inclusiv cele care suportă *DOM Level 1*) asigură în continuare suportul pentru DOM de nivel 0.

În acest model, documentul HTML este văzut ca un arbore de noduri-obiect, structura acestuia fiind destul de simplă. Astfel, în vârful arborelui DOM se află obiectul `document`. Acesta oferă posibilitatea accesului la imagini, formulare, legături, ancore și applet-uri, prin intermediul colecțiilor de obiecte care le reprezintă.

După încărcarea completă a unui document HTML, browserul creează câteva tablouri (colecții de obiecte) în care include ca elemente obiectele asociate elementelor existente în document. De exemplu, sunt create tablourile asociative `document.forms` (include obiecte asociate tuturor formularelor existente în document) și

`document.images` (include obiecte asociate tuturor imaginilor existente în document). Pentru a avea acces la un anumit element HTML (e.g., formular, imagine) prin intermediul obiectului care îl reprezintă, trebuie să se specifice elementul corespunzător al tabloului, ținându-se cont de faptul că primul element al oricărui tablou are indexul 0. În tablourile asociative `document.images` și, respectiv, `document.forms` sunt incluse referințe la imagini și, respectiv, formulare, în ordinea apariției acestora în documentul HTML. A se vedea și **DOM**.

domeniu de valabilitate (*scope*). În programare, referitor la o variabilă, reprezintă porțiunea dintr-un program în care numele acesteia are semnificație; în acest caz se spune despre variabilă că este „vizibilă”. Domeniul de valabilitate poate fi global sau local. A se vedea și **variabilă globală** și **variabilă locală**.

doorway page (*pagină de intrare*). Pagină Web optimizată pentru un anumit motor de căutare și/sau pentru un anumit cuvânt-cheie, creată pentru *spamdexing*. Dacă utilizatorul execută click pe o legătură (e.g., o legătură inclusă în lista de rezultate returnate de un motor de căutare) care duce către o astfel de pagină, utilizatorul va fi redirectionat către pagina reală. Modalitățile de redirectionare includ folosirea metodei *meta refresh*, utilizarea fișierului *.htaccess* sau a fișierului de configurare a serverului Web. De asemenea, unele

pagini de intrare pot fi generate dinamic prin utilizarea unor limbaje de scripting (e.g., PERL, PHP). O altă modalitate utilizează așa-numita tehnică *cloaking*. A se vedea și **htaccess**, **cloaking**, **meta refresh** și **spamdexing**.

Doppler. Software de tip *podcatching client*, cu versiuni pentru sistemele de operare Windows și dispozitive mobile (<http://www.dopplerradio.net/>). A se vedea și **podcatching client**.

DoS (*Denial of Service*). Tip de atac informatic care își propune să forțeze sistemul atacat să utilizeze foarte multe resurse, astfel încât să nu mai poată răspunde la solicitări legitime sau să blocheze accesul utilizatorilor autorizați. Prin intermediul atacurilor DoS nu se fură și nu se distruge informația aflată pe serverul atacat, dar îi sunt consumate resursele de calcul. Dacă atacul este lansat simultan de pe mai multe calculatoare, el se numește DDoS (Distributed DoS).

Unul dintre atacurile DoS este *SYN Flooding*. Acesta exploatează breșele de securitate existente în stiva de protocoale TCP/IP. Conexiunea TCP dintre două procese se realizează printr-un procedeu cunoscut sub numele de *three-way handshake* (*confirmare triplă*). Dacă în acest timp serverul primește un număr foarte mare de cereri (pachete de sincronizare SYN) pentru conexiuni care nu pot fi realizate, își consumă resursele pentru memorarea lor sau chiar se poate bloca. Modalitatea prin care clientul trimite serverului cereri de conectare care nu

se pot realiza constă în utilizarea unor programe care schimbă adresa IP a sursei cu una falsă (în momentul trimiterii pachetului SYN). Serverul va trimite un pachet de acceptare de conectare SYN/ACK la această adresă, așteptând să primească răspuns (*i.e.*, un pachet de confirmare a conectării, ACK, de la client). Pentru stoparea atacurilor *SYN Flooding* bazate pe adrese IP false, este necesară utilizarea unui *firewall*.

Un alt atac DoS este *Ping Flooding*. Atacatorul trimite un număr mare de pachete ICMP (*Internet Control Message Protocol*) către sistemul pe care dorește să-l atace. Dacă atacatorul are la dispoziție o lărgime de bandă superioară sistemului atacat, acesta din urmă nu va putea trimite date în rețea. O variantă a acestui atac este *smurfing*, care face ca detectarea sursei atacului să fie mai greu de realizat. Atacul *Ping Flooding* poate fi stopat la nivelul router-ului sau printr-un *firewall*. A se vedea și **atac și confirmare triplă**.

dosar (folder). În unele sisteme de operare (*e.g.*, Windows, Mac OS), container virtual utilizat pentru organizarea pe dispozitivele de stocare a fișierelor. De asemenea, un dosar poate conține, alături de fișiere, și alte dosare.

Un dosar se mai numește și *catalog*. Este reprezentat grafic prin intermediul imaginii unui dosar. În sistemele UNIX sau bazate pe acesta, dosarul poartă numele de *director*.

dosar privat (private folder). Într-o rețea de calculatoare, dosar aflat pe calculatorul unui utilizator care nu este accesibil și celorlalți utilizatori ai rețelei. A se vedea și **dosar și rețea de calculatoare**.

double. Tip de date utilizat în unele limbaje de programare (*e.g.*, C, C++) pentru a declara variabilele care stochează numere reale în virgulă mobilă, ocupând opt octeți (64 de biți). A se vedea și **număr în virgulă mobilă**.

download (descărcare). Procesul prin care copia unui fișier este adusă de pe un calculator aflat la distanță pe calculatorul local (care l-a solicitat), prin intermediul unei rețele. A se vedea și **upload**.

drag-and-drop. În interfețele grafice cu utilizatorul (GUI), este o acțiune care constă în execuția unui click pe un obiect virtual și, în timp ce butonul este ținut apăsat, tragerea acestuia în alt loc sau pe un alt obiect virtual. Scopul constă în invocarea a diverse tipuri de acțiuni sau crearea de asociații între obiecte. A se vedea și **GUI**.

DRAM (Dynamic Random Access Memory, Memorie RAM dinamică). Tip de memorie RAM în care fiecare bit este stocat într-un condensator plasat într-un circuit integrat. Pentru a păstra datele, este necesară reîmprospătarea periodică a sarcinii electrice a condensatorilor, motiv pentru care memoria se numește dinamică. Cel mai important avantaj al memoriilor DRAM este

simplitatea lor structurală, pentru stocarea fiecărui bit fiind necesari câte un condensator și un tranzistor, comparativ cu patru tranzistori în cazul memoriilor SRAM, ceea ce permite obținerea unei densități foarte mari. A se vedea și RAM și SRAM.

DrawBot. Aplicație *open source* pentru Mac OS X care le permite utilizatorilor să scrie scripturi în limbajul Python, pentru a genera grafică bidimensională. Este utilizată mai ales în scop educațional, pentru învățarea bazelor programării. A se vedea și **Python**.

Dreamweaver (DW). Instrument comercial IDE pentru proiectarea, dezvoltarea și întreținerea site-urilor și aplicațiilor Web bazate pe standarde. DW este un instrument foarte puternic, realizând un echilibru între abordarea vizuală (*drag-and-drop*) și cea directă (scrierea codului). Este creat de Macromedia, acum în proprietatea Adobe Systems. Produsul poate fi rulat pe variate platforme software (*e.g.*, Mac, Windows).

Alături de elementele HML de bază (*e.g.*, hiperlinkuri, ancore, imagini, comentarii, text formatat), DW permite includerea în paginile Web a elementelor media (SWF, Flash, *applet*, ActiveX) și a scripturilor, crearea de tabele, formulare, cadre, precum și utilizarea HTML dinamic.

DW automatizează întregul proces de editare a paginilor, fiind asemănător în anumite privințe cu un procesor de texte. Utilizatorul poate alege să lucreze în

modul vizual, în modul HTML sau simultan în amândouă (*split*). Se poate realiza previzualizarea paginilor create în browserul instalat pe calculatorul utilizatorului. DW încorporează facilități puternice de editare a codului, inclusiv pentru autocompletare și depanare.

DW oferă și posibilitatea utilizării extensiilor. Acestea sunt componente pe care utilizatorii înregistrați le pot descărca și folosi în scopul creării de aplicații Web complexe pentru *e-Learning*, *e-commerce* etc. Include instrumente puternice pentru crearea aplicațiilor Web ale căror date sunt păstrate în baze de date, oferind suport pentru un număr mare de limbaje și platforme de programare pentru server (ASP, JSP, *ColdFusion*, AS.NET și PHP). De asemenea, DW oferă suport pentru utilizarea limbajelor JavaScript, CSS, XML și XSLT. A se vedea și **CSS, IDE, JavaScript, Web Authoring Tool, XML și XSLT**.

drepturi de acces (*access rights*). Se referă la permisiunile acordate unui utilizator de a accesa, citi (vizualiza) sau modifica un fișier sau un director. Acestea sunt verificate în momentul în care un utilizator încearcă să acceseze o resursă.

driver (*driver*). Dispozitiv hardware sau program care controlează un alt dispozitiv. A se vedea și **driver de dispozitiv**.

driver de dispozitiv (*device driver*). Componentă software care permite altor programe să comunice cu un

anumit dispozitiv hardware (e.g., imprimantă, placă grafică, placă de rețea, placă de sunet, magistrale locale de diverse tipuri, camere foto și video digitale). A se vedea și **driver**.

driver de imprimantă (*printer driver*). Program (driver) care permite aplicațiilor instalate pe un calculator să utilizeze o anumită imprimantă fără să țină cont de detaliile tehnice (de hardware-ul) acesteia. În acest scop, convertește datele care trebuie tipărite într-o formă specifică imprimantei. A se vedea și **driver** și **imprimantă**.

driver video (*video driver*). Software care permite programelor să comunice cu placa grafică. A se vedea și **driver de dispozitiv** și **placă grafică**.

dropper (*pipetă*). Program care instalează un virus într-un sistem informatic. Este folosit adesea de creatorii de viruși. A se vedea și **virus de calculator**.

Drupal. CMS *open source* care poate susține o varietate de site-uri Web, de la bloguri, broșuri colective, până la site-uri ale unor comunități largi (<http://drupal.org>).

DSA (*Digital Signature Algorithm*). Algoritm criptografic asimetric utilizat pentru crearea semnăturilor digitale. Este folosit în standardul DSS (*Digital Signature Standard*). DSA îi este atribuit lui David W. Kravitz, fost angajat la NSA.

DSL (*Digital Subscriber Line*). Tehnologie care utilizează liniile telefonice existente pentru a aduce informații pe o

lățime de bandă ridicată la organizații de dimensiuni relativ reduse, oferind o conexiune Internet permanentă (*full-time*). Cele două tipuri de bază ale tehnologiei DSL sunt asimetric (ADSL, *Asymmetric DSL*) și simetric (SDSL, *Symmetric DSL*), prima fiind cea mai utilizată. În mod obișnuit, viteza de *download* este mai mare decât cea de *upload* pentru ADSL, cele două viteze fiind egale în cazul tipului SDSL.

DTD (*Document Type Definition*). Limbaj utilizat pentru descrierea documentelor XML (*eXtensible Markup Language*). Documentele scrise în limbajul DTD servesc la validarea documentelor XML, precizând setul de elemente/atribute folosite, modul de structurare a acestora, precum și tipul conținutului. Din punct de vedere istoric, DTD a fost primul limbaj utilizat pentru descrierea documentelor XML, folosind reguli sintactice derivate din SGML.

O specificație DTD definește o clasă de documente XML care se conformează unui anumit model structural. Ea poate fi internă (*i.e.*, inclusă) sau externă (*i.e.*, referențiată) documentului XML pe care-l validează. În ultimul caz, DTD-ul este stocat într-un fișier cu extensia *dtd*. Prin intermediul unei specificații DTD, utilizând declarații (sau descrieri), pot fi precizate pentru un document XML: structura conținutului; elementele care pot apărea, modul cum apar și modul cum sunt imbricate; atributele fiecărui element și fiecărei măști pentru verificarea valorii acestora; referințele care

pot fi utilizate. Trebuie declarate toate elementele și atributele, neavând importanță ordinea în care sunt plasate declarațiile.

O declarație DTD reprezintă o definiție formală a unei componente a documentului XML (element, atribut, entitate), fiind delimitată de caracterele „<” și „>” și compusă din următoarele elemente (separate prin spații albe): un cuvânt-cheie utilizat pentru a preciza tipul componente care va fi declarată (ELEMENT, ATTLIST, ENTITY), prefixat de caracterul „!”; numele componente; descrierea componente.

DTP (*desktop publishing*). Procesul de creare și editare a unui document cu ajutorul unui calculator, având ca scop final tipărirea acestuia. În DTP sunt utilizate programe specializate (e.g., *Adobe InDesign*, *QuarkXPress*, *CorelDraw*). A se vedea și **Adobe InDesign**, **CorelDraw X3 Graphics Suite** și **QuarkXPress**.

dublă precizie (*double precision*). Denumit și *binary64* (IEEE 754-2008), este un format utilizat pentru reprezentarea numerelor în virgulă mobilă, care folosește opt octeți (64 de biți). Dubla precizie este denumită în unele limbaje de programare (C, C++, Java) și *double*. În aceste limbaje, tipurile reale sunt *float*, *double* și *long double*. A se vedea și **simplică precizie**.

Duck Duck Go. Motor de căutare dezvoltat în Valley Forge, SUA (<http://duckduckgo.com/>). Utilizează informații provenite dintr-un număr mare de surse (e.g., *Wikipedia*), pentru a crește relevanța rezultatelor furnizate. A se vedea și **motor de căutare**.

Duncan Watts. Profesor de sociologie la Universitatea Columbia și cercetător asociat al *Yahoo! Research*, este autorul cărții *Six Degrees: The Science of a Connected Age* (2004). A se vedea și **teoria celor șase grade de separare**.

E

e-business (*afaceri electronice*). Denumite și *afaceri pe Web*, reprezintă o modalitate modernă de desfășurare a afacerilor. Există mai multe modele de *e-business*: *Business-to-Business* (B2B), *Business-to-Consumer* (B2C) și *Business-to-Government* (B2G). De asemenea, există mai multe categorii de afaceri pe Web: licitații electronice (*e-auction*), tranzacții bancare electronice (*e-banking*), comerț electronic (*e-commerce*), promovarea produselor și serviciilor (*e-marketing*). A se vedea și **e-commerce**.

e-commerce (*comerț electronic*). Categorie de afaceri electronice (*e-business*) având ca obiect comercializarea de bunuri și servicii prin intermediul tehnologiilor Internet. Cele mai folosite modele de afaceri sunt: magazinul electronic (*e-shop*), magazinul electronic universal (*e-mall*), aprovizionarea electronică (*e-procurement*), piața unui terț (*third party marketplace*), comunitățile virtuale (*virtual communities*), platformele de colaborare (*collaborative environments*) și brokerajul de informații. A se vedea și **e-business**.

e-form (*formular electronic*). A se vedea **formular electronic**.

e-government (*administrare/guvernare electronică*). Se referă la utilizarea tehnologiei informațiilor și comunicațiilor pentru furnizarea unor servicii guvernamentale, afaceri și realizarea unor interacțiuni diverse cu cetățenii, inclusiv la nivel local. Iată câteva dintre modelele de *e-government* existente: *Government to Citizen* (G2C), *Government to Business* (G2B), *Government to Government* (G2G), *Government to Employees* (G2E). Administrarea electronică este realizată în special prin intermediul Internetului (formă denumită *online government* sau *Internet-based government*), dar sunt utilizate și modalități alternative: telefon, fax, PDA, mesaje SMS și MMS, *Bluetooth*, identificare biometrică, managementul traficului auto, carduri de identitate, servicii TV și radio, *e-mail* etc.

e-Learning (sau *eLearning*). Un tip de educație bazat pe tehnologie, în care mediul de instruire este reprezentat de calculator, iar conținutul învățării este livrat prin intermediul CD-urilor, Internetului sau intraneturilor, având următoarele caracteristici: utilizează conținut relevant pentru obiectivele de învățare fixate de educator; folosește metode de învățare care vin în ajutorul studentului; utilizează elemente media (e.g., text, imagini) pentru livrarea conținutului și concretizarea metodelor de învățare; permite învățarea dirijată de un educator – numit și instructor –

(*e-Learning* sincron) sau prin studiu individual (*e-Learning* asincron); oferă cunoștințe și construiește competențe în funcție de scopurile de învățare individuale sau îmbunătățește performanțele unei organizații.

În literatura de specialitate sunt disponibile și alte definiții ale *e-Learning*-ului: prin *e-Learning* se înțelege orice act sau proces virtual utilizat pentru a obține date, informații, abilități sau cunoștințe, reprezentând o modalitate de a facilita și îmbunătăți învățământul prin utilizarea dispozitivelor de calcul și a tehnologiei comunicațiilor. Definiția agreeată de Comisia Europeană este următoarea: prin *e-Learning* se înțelege utilizarea Internetului și a noilor tehnologii multimedia pentru a îmbunătăți calitatea învățării prin facilitarea accesului la resurse și servicii, schimb la distanță și colaborare.

În sens restrâns, *e-Learning*-ul reprezintă un tip de educație la distanță, ca experiență planificată de predare, învățare și evaluare organizată de o instituție care furnizează, mediat, materiale într-o ordine logică, pentru a fi asimilate de cursanți într-o manieră proprie. Mediarea se realizează folosind noile tehnologii ale informației și comunicării, în special prin Web. În multe universități, *e-Learning*-ul este utilizat pentru a defini un mod specific de participare la cursuri sau programe de studiu, în care studenții utilizează rar sau uneori deloc facilitățile educaționale tradiționale oferite în

campusuri, înlocuindu-le cu studiul *online*.

Formele de instruire bazate pe *e-Learning* sunt: învățarea individuală, învățarea individuală asistată, învățarea într-o clasă virtuală și învățarea colaborativă.

e-Learning-ul poate fi implementat atât folosind rețeaua proprie a unei instituții (intranet), cât și prin intermediul rețelei globale (Internet), numai în ultimul caz putându-se vorbi de învățare *online* sau învățare la distanță. A se vedea și **învățare la distanță și tipuri de e-Learning**.

e-mail (*poștă electronică*). A se vedea **poștă electronică**.

e-money (*bani electronici*). Se referă la banii transferați între conturile bancare ca urmare a plăților realizate prin Internet.

EAN-13. Standard implementat, inițial, în Europa de *International Article Numbering Association* sub numele de European Article Numbering, pentru marcarea produselor vândute cu amănuntul prin coduri de bare. În 1977, acest standard era utilizat în 12 state (adică în toate țările Comunității Europene). Astăzi, el este folosit în Europa, Statele Unite, Canada, Australia și Japonia.

EAN-13 este un superset al standardului *UPC-A* (simbologie utilizată în Statele Unite și Canada pentru marcarea produselor vândute cu amănuntul, care utilizează 12 cifre pentru codificarea datelor). Standardul *EAN-13* este o

simbologie liniară continuă care utilizează mai multe lățimi de bare (bara cea mai largă este de cel mult patru ori mai lată decât bara cea mai îngustă).

Fiecare țară în care se utilizează simbologia *EAN-13* are câte o autoritate *EAN* care reglementează utilizarea acesteia în țara respectivă. În România, la 1 august 1993, douăzeci de societăți comerciale au înființat *Asociația Română pentru Numerotarea Internațională a Articolelor (EAN Romania)*, cu adresa <http://www.ean.ro/>.

Standardul *EAN-13* folosește 13 caractere numerice pentru identificarea produselor. Primele două sau trei cifre reprezintă numărul de sistem (*number system*), care identifică țara sau regiunea economică în care este înregistrat producătorul (e.g., România are codul 594, iar Spania 84). Acesta este urmat de nouă sau zece cifre (depinzând de lungimea numărului de sistem) și de încă o cifră (ultima), numită *suma de control (checksum)*. Cele nouă sau zece cifre includ un cod al producătorului (*manufacturer code*) și un cod al produsului (*product code*).

Numărul de sistem este alocat de organizația *EAN* (<http://ean-int.org/>). Codul producătorului este alocat de autoritatea *EAN* din țara/regiunea economică respectivă, iar codul produsului este alocat de producător (acesta este liber să aloce fiecărui produs al său un cod, fără a consulta o organizație, cu obligația de a nu repeta un cod deja alocat). Această modalitate de

alocare determină obținerea câte unui cod unic pentru fiecare produs. Pot fi adăugate suplimentar două sau cinci cifre.

eBay, Inc. Companie americană care deține *eBay.com*, cel mai mare site din lume axat pe licitații *online* (<http://www.ebay.com/>). De asemenea, eBay a cumpărat *Skype*, *PayPal*, *StubHub* și alte companii. eBay, Inc. a fost fondată în septembrie 1995 de programatorul Pierre Omidyar.

Eckert, John Adam Presper (1919–1995). Inginer american și pionier în domeniul calculatoarelor. A inventat, împreună cu John Mauchly, ENIAC, primul calculator electronic digital universal. Cei doi au fondat Eckert-Mauchly Computer Corporation (EMCC), prima companie comercială având ca obiect de activitate producerea și vânzarea calculatoarelor. A se vedea și **ENIAC**.

Eclipse. Mediu de dezvoltare *open source* scris în Java și dezvoltat de fundația *Eclipse*, oferit sub licență EPL (*Eclipse Public License*; <http://www.eclipse.org/>). Poate fi folosit pentru a dezvolta aplicații în Java și, prin utilizarea unor *plugin*-uri, în alte limbaje de programare (e.g., C, C++, Python, COBOL, Perl, PHP). Are versiuni pentru sistemele de operare Windows, Linux și Mac OS X.

ECMAScript. Versiunea standardizată (sub numele ECMA-262) de către ECMA (*European Computer Manufacturers Association*) și Netscape a limbajelor de programare JavaScript

(Netscape) și JScript (Microsoft). Prima ediție a limbajului a apărut în iunie 1997, a doua în iunie 1998, iar a treia în decembrie 1999. Standardul ECMA-262 este standard ISO sub numele ISO-162262, începând din aprilie 1998.

Specificațiile ECMAScript, bazate pe JavaScript, includ numai nucleul acestui limbaj. Versiunile JavaScript (implementate în browserele bazate pe motorul *Gecko* și în *Opera*) și JScript (implementat în browserul *Internet Explorer*) trebuie să respecte specificațiile ECMAScript. A se vedea și **ISO** și **JavaScript**.

eContentPlus. Adoptat la 9 martie 2005 de Parlamentul European, este un program multianual având ca scop creșterea accesibilității și utilizabilității conținutului digital în Europa.

editor de cod-sursă (*source code editor*). Editor de text special conceput pentru editarea de către programatori a codului-sursă al programelor de calculator. Poate fi o aplicație independentă sau inclusă într-un mediu de dezvoltare integrat (IDE). Dintre editoarele de cod-sursă *open source* sau gratuite populare pot fi amintite *Notepad++* (Windows), *Vim* (Linux), *Programmer's Notepad* (Windows) și *PSPad* (Windows). A se vedea și **cod-sursă** și **IDE**.

editor de fonturi (*font editor*). Aplicație software utilizată în scopul modificării sau creării fonturilor. Au fost dezvoltate editoare pentru fonturi

bitmap și, respectiv, *outline*. Editoarele de fonturi pot fi *open source* (e.g., *FontForge*) și proprietare (e.g., *FontCreator*, *FontLab Studio*, *Private Character Editor*). A se vedea și **font**.

editor de legături (*link editor*). Denumit și *linker*, este un program care assemblează obiectele produse de un compilator (programe, rutine sau biblioteci) și fișierele de date, generând un singur program executabil. A se vedea și **compilator**.

editor HTML (*HTML editor*). Aplicație software folosită pentru crearea și editarea fișierelor HTML. Permite inserarea marcajelor HTML fără să fie necesară scrierea lor explicită. De asemenea, multe editoare HTML au facilități suplimentare (e.g., CSS, XML, JavaScript). Aceste aplicații au diverse grade de complexitate (e.g., text, obiect, WYSIWYG). A se vedea și **HTML** și **WYSIWYG**.

editor de imagini (*image editor*). Aplicație software folosită pentru modificarea imaginilor (e.g., aplicarea unor filtre: *blur*, *motion blur*, *emboss*, *sharpen*, *gradient*).

EditPlus. Editor comercial pentru HTML, CSS, PHP, VBScript, Perl, C/C++, Java, JScript și ASP (<http://www.editplus.com/>).

Estridge, Philip Donald (1937–1985). Cunoscut ca Don Estridge, „tatăl IBM PC”, a condus la IBM echipa de ingineri și proiectanți care a dezvoltat

calculatorul IBM Personal Computer, lansat în 1981. A se vedea și **IBM și IBM Personal Computer**.

Education and Training 2010.

Program al Comisiei Europene, integrează toate inițiativele acesteia în domeniul educației și formării la nivel european, inclusiv educația și formarea vocațională (*procesul Copenhaga*), formulând obiective strategice și priorități. Ca și *procesul Bologna*, inițiat în 1999, *procesul Copenhaga* este crucial în dezvoltarea învățământului în Europa, amândouă contribuind activ la atingerea obiectivelor stabilite în *strategia Lisabona*.

Programul este implementat printr-o metodă deschisă de colaborare: statele membre sunt responsabile pentru organizarea sistemului de educație formală și continuă, fiind sprijinite pentru realizarea unui schimb de experiență în construirea politicilor, pentru realizarea unor indicatori comuni. Sistemele de educație trebuie să îmbunătățească: pregătirea educatorilor, integrarea *Tehnologiei Informației și Comunicațiilor*, eficiența investițiilor, învățarea limbilor străine. Aceste sisteme trebuie să devină flexibile, pentru a asigura educația tuturor cetățenilor, indiferent de locul în care se află.

Programul *Education and Training 2010* sprijină dezvoltarea competențelor necesare într-o societate bazată pe cunoaștere, alături de competențele digitale fiind încurajate creativitatea,

lucrul în echipă și competențele necesare pentru rezolvarea unor probleme practice. Majoritatea statelor membre încurajează și recomandă utilizarea resurselor educaționale deschise, a sistemelor *open source* și a tehnologiilor colaborative Web 2.0. Astfel, ca parte a implementării programului, în anul 2006 a fost adoptată de Parlamentul European și Comisia Europeană o recomandare cu privire la stabilirea de competențe-cheie pentru învățarea de-a lungul întregii vieți. Prin competență-cheie se înțelege „un pachet transferabil și multifuncțional de cunoștințe, aptitudini și atitudini, necesar tuturor indivizilor în vederea dezvoltării personale și a incluziunii sociale și profesionale”. Iată cele opt competențe-cheie: lingvistice – în limba maternă; lingvistice – în limbi străine; matematice; științifice și tehnologice; de învățare (*learning to learn*); civice, interpersonale, interculturale și sociale; antreprenoriale; de deschidere/manifestare culturală. A se vedea și **Open Source Software și Web 2.0**.

educația permanentă (*lifelong learning*). Principiu de proiectare și organizare a educației în perspectiva desfășurării acesteia în diferite forme, pe toată durata vieții. Nu se reduce la educația continuă sau la formarea profesională continuă. Subliniază importanța cuprinderii în procesul educațional a integralității individului și a tuturor formelor și tipurilor de educație.

Scopurile educației permanente sunt atât de natură economică, cât și de natură individuală și socială: intensificarea progresului economic și a dezvoltării; contribuția la dezvoltarea și împlinirea personală; promovarea incluziunii sociale și participării democratice. Instrumentele Web 2.0 pot contribui la un tip de educație permanentă adaptat caracteristicilor studenților adulți și nevoilor societății: aceste tehnologii le permit studenților să participe la activități cu plăcere, iar învățarea poate să devină atunci un produs al participării. A se vedea și **Web 2.0**.

educație (*education*). Termen având ca origine latinescul *educō* (a alimenta, a crește, a îngriji – plante sau animale). Un termen cu semnificații similare este franțuzescul *éducation*, din care a fost derivat și cel românesc. Există definiții multiple care încearcă să surprindă esența fenomenului educațional. Iată două dintre acestea: „educația este acțiunea de formare a individului pentru el însuși, prin care acesta își dezvoltă o multitudine de interese” (Herbart); „educația este acea reconstrucție sau reorganizare a experienței care se adaugă la înțelesul experienței precedente și care mărește capacitatea de a dirija evoluția celei care urmează” (Dewey).

Privită din diverse unghiuri, educația poate fi înțeleasă: ca proces, ca acțiune de conducere, ca relație interumană sau ca ansamblu de influențe. Prin educare se urmăresc două scopuri principale: instrucția (*i.e.*, predarea de cunoștințe,

formarea deprinderilor și competențelor) și formarea personalității.

Fundamentele educației sunt determinări care condiționează procesul educațional: fundamente biopsihice (ereditatea generală și specifică), fundamente socioculturale (mediul socio-cultural – determinări de ordin comunitar care au impact direct asupra procesului educațional), fundamente istorice (preluarea descoperirilor altora și aplicarea lor la condițiile actuale), fundamente filosofice (în spatele educației se află o anumită filosofie), fundamente politice (practica educațională este, cel puțin parțial, un mijloc de distribuire a cunoașterii în raport cu interesele celor aflați la putere) și fundamente științifice (procesul educațional se realizează pe baza unor teorii științifice: ale învățării, predării, examinării etc.).

Educația se realizează sub trei forme: educația formală (include totalitatea influențelor intenționate și sistematice realizate în cadrul unor instituții specializate: gimnaziu, liceu, universitate), educația nonformală (totalitatea influențelor educative care se desfășoară în afara clasei sau prin intermediul unor activități facultative) și educația informală (totalitatea informațiilor neintenționate, eterogene cu care se confruntă individul în viața de toate zilele).

Educația reprezintă acțiunea simultană a numeroși factori: familia, școala, mass-media, instituțiile culturale, structurile asociative (organizații profesionale, spirituale sau disciplinare). Un rol important și o influență decisivă

în educație (sub toate cele trei forme ale acesteia) îl au astăzi mijloacele electronice utilizate în predare, învățare, divertisment și comunicație (e.g., calculatorul personal, rețelele de calculatoare, Internetul și, în particular, Web-ul, dispozitivele mobile – telefoane inteligente, PDA-uri).

edutainment (*edutainment*). Termen care provine de la *education-entertainment* (educație-amuzament), desemnează o activitate în care educația se realizează prin mijloace amuzante. În acest scop sunt utilizate diverse materiale multimedia și aplicații Web interactive. A se vedea și **aplicație Web**, **educație** și **multimedia**.

EDVAC (*Electronic Discrete Variable Automatic Computer*). Unul dintre primele calculatoare electronice, proiectat – ca și ENIAC, anterior – de John Mauchly și J. Presper Eckert, John von Neumann având rol de consultant. Spre deosebire de predecesorul său, efectua calcule în sistemul binar și utiliza programe stocate. Descrierea arhitecturii calculatorului a fost făcută de John von Neumann în articolul său *First Draft on a Report on the EDVAC* (1945). A se vedea și **ENIAC**.

EEPROM (*Electrically Erasable Programmable Read-Only Memory*). Memorie nevolatilă utilizată în calculatoare și în alte dispozitive pentru a stoca o cantitate limitată de date, inventată în 1978 de George Perlegos, de la Intel. În cazul în care trebuie stocată o cantitate mare de date, se

utilizează un tip particular (relativ recent) de EEPROM, ca de exemplu memoria *flash*, mult mai ieftină decât dispozitivele EEPROM tradiționale. Memoria EEPROM poate fi ștersă pe cale electrică. A se vedea și **EPROM**, **memorie flash** și **ROM**.

Eiffel. Limbaj de programare orientat pe obiect, standard ISO lansat în 1986, proiectat pentru a le permite programatorilor să dezvolte eficient software extensibil și reutilizabil. Este utilizat în mediul academic pentru predarea principiilor programării. Au fost create numeroase medii de dezvoltare bazate pe Eiffel, utilizate pentru crearea de aplicații folosite în finanțe, industria aerospațială, sistemul de sănătate, jocuri video. A se vedea și **limbaj de programare** și **programare orientată pe obiect**.

eJobs. Site românesc cu oferte de muncă, pentru companii și persoane particulare (<http://www.ejobs.ro/>). Utilizatorii pot crea conturi în care își postează CV-urile și ofertele de locuri de muncă.

Electronic Art (EA). Companie americană care produce și comercializează jocuri video pentru console și calculatoare (<http://www.ea.com/>). Cele mai cunoscute produse ale EA sunt simulatoarele sportive (e.g., *FIFA Soccer*), alături de *Need for Speed*, *Medal of Honor*, *The Sims* și *Battlefield*. Compania a fost înființată în 1982 de Trip Hawkins.

element (*element*). Unitatea logică de structură a unui document marcat folosind HTML, XML sau alte limbaje derivate. Datele dintr-un document trebuie marcate folosind elemente.

Într-un document sunt utilizate instanțe ale elementelor. În mod obișnuit, o instanță a unui element (e.g., *body*) include un marcaj de început (e.g., `<body>`) și unul de sfârșit (e.g., `</body>`), între ele fiind plasat conținutul textual marcat. Un element care nu are conținut textual se numește element vid (e.g., *img*, *hr*, *br*). Instanțele unui element vid sunt create cu sintaxa `<element></element>` sau `<element />` (e.g., `
`). A se vedea și **limbaj de marcare și marcaj**.

element XML (*XML element*). Unitatea logică de structură a unui document XML. Acestea pot conține diverse tipuri de elemente, fiecare cu un nume unic. Instanțierea unui element se realizează folosind un marcaj de deschidere (`<element>`) și un marcaj de închidere (`</element>`), cele două marcaje delimitând un conținut, astfel: `<element>continut</element>`.

Pentru scrierea numelui elementului (*element*) trebuie respectate următoarele reguli: între el și caracterele `<`, `</` și `>` nu trebuie să existe spații albe; numele este alcătuit din litere, cifre și alte caractere (e.g., *underscore*, punct); trebuie să înceapă cu o literă sau cu caracterul *underscore*; nu poate conține spații albe; nu trebuie să înceapă cu `xml/XML`. De asemenea, numele elementelor sunt *case-sensitive* (e.g.,

element nu este totuna cu *Element*). În plus, elementele trebuie să fie închise și imbricate corect.

Elementele XML pot fi vide (*i.e.*, au conținut vid) sau nevide (*i.e.*, au conținut nevid). Conținutul unui element nevid poate fi reprezentat de text și/sau alte elemente. Conținutul textual este format numai din caractere incluse în setul precizat prin intermediul atributului `encoding` utilizat în declarația XML. Implicit, spațiile albe multiple sunt reduse la unul singur.

Un element poate avea drept conținut alte elemente, ceea ce determină apariția unor relații de „rudenie” între acestea și a unor constrângeri suplimentare. Relațiile menționate alcătuiesc modelul structural al documentului și pot fi de tipul „frate-frate” (două elemente consecutive sunt frați dacă se află pe același nivel al structurii ierarhice a documentului XML) sau „părinte-copil” (un element inclus într-un alt element este copilul celui din urmă, acesta fiind părintele primului). Elementul fără părinte se numește element-rădăcină (*root*), acesta existând obligatoriu și fiind unic (prima constrângere). Mai multe elemente pot avea același părinte, care se constituie ca un container pentru copiii săi. De asemenea, elementele nu pot fi încrucișate, adică marcajele trebuie închise în ordinea inversă celei în care au fost deschise (a doua constrângere). Elementul-rădăcină generează o structură arborescentă a documentului XML. Frunzele arborelui generat sunt

reprezentate de elementele vide. A se vedea și **declarație XML**, **element** și **XML**.

Elearningeuropa.info. Portal educațional cu versiuni în toate limbile vorbite în țările Uniunii Europene, inclusiv în limba română (<http://www.elearningeuropa.info/>). A fost înființat de *Direcția Generală pentru Educație și Cultură* a Comisiei Europene, în scopul de a promova utilizarea tehnologiilor multimedia și Internetului în serviciul educației și formării.

Portalul oferă informații specifice, servicii și resurse pentru patru domenii de bază (școli, învățământul superior, domeniul angajaților și educația continuă), fiind o platformă deschisă unde utilizatorii și comunitățile care o folosesc pot obține informații, pot împărtăși experiențele și proiectele lor sau pot purta discuții pe diverse teme.

Serviciile oferite de portal sunt următoarele: *Directory* (furnizează o mare varietate de materiale *e-Learning* și surse de informații din întregul continent, organizate pe categorii și domenii de interes); *Community* (loc în care practicienii *e-Learning*-ului se pot întâlni pentru a dezbate probleme, a posta și a căuta informații și idei relevante pentru sarcinile și nevoile lor zilnice – include forumuri, interviuri luate de utilizatori, concursuri, sugestii ale utilizatorilor și profilurile publice ale membrilor); *eLearning Papers* (publicație *online* care oferă un sumar

al fiecărui articol, în 19 limbi și care apare de 5 ori pe an, fiind oferită gratuit tuturor cititorilor); *Newsletter* (le furnizează utilizatorilor informații despre articolele curente, evenimente și resurse *e-Learning* și este publicat în 19 limbi, lunar). A se vedea și **e-Learning**.

ELIZA. Rezultatul uneia dintre primele încercări semnificative de realizare a unui *chatbot* (agent conversațional), din care s-au inspirat ulterior și creatorii unui alt program de acest tip, ALICE. A fost creat de Joseph Weizenbaum la mijlocul anilor 1960. Agentul se bazează pe procesarea limbajului natural, iar producerea răspunsurilor se realizează folosind o serie de scripturi. A se vedea și **chatbot**.

Ellison, Lawrence Joseph (n. 1944). Antreprenor american, cofondator, alături de Ed Oates, al Oracle Corporation (1977). A se vedea și **Oracle Corporation**.

EMF (*Enhanced MetaFile*). Format grafic de tip *metafile*. A se vedea și **formate metafile**.

emoticon (*emoticon*). Șir de caractere sau pictograme care exprimă o anumită stare sufletească (emoție). Sunt utilizate în mesajele schimbate prin *e-mail* și *chat* sau postate pe unele forumuri. Iată câteva emoticoane, sub formă de șiruri de caractere: :-) sau :) (zâmbesc), ;-) (fac cu ochiul și zâmbesc), :-((imi pare rău), :D sau :-D (sunt foarte bucuros), :-O (am rămas cu gura căscată de uimire).

Engelbart, Douglas (n. 1925). Inventator american și pionier al dezvoltării calculatoarelor. Este inventatorul *mouse*-ului. A fost liderul echipei care a inventat hipertextul și promotor al utilizării calculatoarelor și rețelelor. A se vedea și **hiperlegătură**.

ENIAC (*Electronic Numerical Integrator and Computer, calculator și integrator electronic numeric*). Primul calculator electronic universal, Turing complet, capabil de a fi reprogramat în vederea rezolvării unor probleme de calcul diverse. A fost proiectat de John Mauchly și J. Presper Eckert, de la Universitatea Pennsylvania. Construcția sa a început în 1943 și s-a terminat în 1946, fiind finanțată de armata SUA. Calculatorul efectua calcule în sistemul zecimal. A se vedea și **Eckert, John Adam Presper** și **Mauchly, John**.

entități (X)HTML (*(X)HTML entities*). Porțiuni de text (șiruri de caractere sau simboluri) utilizate în documentele (X)HTML pentru a genera (în scopul afișării) simboluri speciale, caractere din diverse limbi, simboluri matematice sau caractere rezervate.

Accesarea entităților se realizează prin intermediul referințelor la entități, aceste referințe având sintaxa `&nume_entitate;` (*i.e.*, caracterul „&” urmat de numele entității și de „;”). Există posibilitatea ca o referință la o entitate-caracter să fie specificată atât prin sintaxa precizată, cât și folosind `&#cod_numeric;`, unde `cod_numeric` este codul numeric al caracterului respectiv. Iată câteva exemple de

entități (între paranteze sunt precizate referințele corespunzătoare): simbolul special „©”, *copyright* (`©` sau `©`); caracterul „α” (`α` sau `α`); simbolul matematic „∫” (`∫` sau `∫`); caracterul rezervat „>” (`>` sau `>`). A se vedea și **HTML**.

entități XML (*XML entities*). Informații utilizate în documentele XML sau DTD prin intermediul unor referințe. Specificațiile XML precizează următoarele cinci entități-caracter predefinite (între paranteze sunt indicate referințele corespunzătoare): `<` (`<`), `>` (`>`), `&` (`&`), `'` (`'`) și `”` (`"`). Toate procesoarele XML suportă referințele la entitățile predefinite, chiar dacă acestea din urmă nu sunt declarate în DTD. Entitățile-caracter numerice sunt caractere Unicode (*e.g.*, simboluri, litere, ideograme) referențiate prin intermediul codurilor numerice ale acestora (`&#cod_numeric;`), fără o declarație specială în DTD. Codul numeric (`cod_numeric`) este de forma *nnn* (codul zecimal al caracterului) sau *xhhh* (codul hexazecimal al aceluiași caracter). A se vedea și **DTD** și **XML**.

Entropia Universe (EU). Joc multiplayer și univers 3D virtual *online*, utilizat pentru divertisment, interacțiunea socială și comerț, reprezentând o nouă generație a distracției interactive (<http://www.entropiauniverse.com/>). EU a fost lansat în ianuarie 2003 de compania suedeză MingArk. Jocul beneficiază de propriul său sistem

economic, care le permite utilizatorilor să schimbe bani reali în PED (*Project Entropia Dollars*) și viceversa, cursul de schimb fiind de 10PED=\$1USD.

Utilizatorul pășește în lumea EU ca cetățean și poate să-și creeze și personalizeze propriul său avatar tridimensional, prin folosirea unui sistem de modelare care beneficiază de libertate și flexibilitate uimitoare. Acțiunea din EU nu se desfășoară pe Pământ, ci pe o colonie numită Calypso, aflată pe o planetă îndepărtată. Teritoriile de pe Calypso sunt în continuă dezvoltare, incluzând trei continente vaste cu orașe extinse, în care un proaspăt cetățean își va începe viața. Aceste orașe vor oferi oricărui cetățean un imens spațiu în care acesta își va putea desfășura activitatea. În joc sunt prezente instituții, servicii și o multitudine de posibilități de distracție virtuală. Utilizatorii români ai jocului EU au un forum de discuții propriu (<http://www.entropiaforum.ro/>).

EPROM (*Erasable Programmable Read-Only Memory*). Memorie nevolatilă care poate fi ștersă numai prin expunere la lumină ultravioletă (pentru optimizarea timpului de ștergere, lungimea de undă a acesteia trebuie să fie de 253,7 nm). A se vedea și **EEPROM** și **ROM**.

era informației (*information era*). Etapă în dezvoltarea omenirii. Cuprinde, în general, trei mari perioade: societatea informațională, societatea cunoașterii și societatea conștiinței. A

se vedea și **societatea cunoașterii și societatea informațională**.

eroare de server (*server error*). Împiedică satisfacerea unei cereri de informații realizate de client prin protocolul HTTP, din cauza unei erori a serverului Web. Astfel de erori sunt indicate de codurile de stare HTTP generate de server care încep cu cifra 5 (e.g., 500: *Internal Server Error*; 502: *Bad Gateway*; 503: *Service Unavailable*). A se vedea și **cod de stare HTTP**.

eroare de sintaxă (*syntax error*). Eroare determinată de o instrucțiune (inclusă într-un program) care nu respectă regulile sintactice ale limbajului utilizat în scrierea programului. A se vedea și **program**.

eroare hard (*hard error*). Eroare având drept cauză funcționarea incorectă sau defectarea unei componente hardware.

eroare logică (*logic error*). Eroare care, deși generează rezultate incorecte, nu determină blocarea programului. Poate fi cauzată, de exemplu, de utilizarea unui algoritm greșit.

eroare semantică (*semantic error*). Eroare determinată de o instrucțiune care, deși este corectă din punct de vedere sintactic, nu este corectă din punct de vedere funcțional.

Estridge, Philip Donald (1937–1985). Inginer american, cunoscut și ca Don Estridge, a condus echipa care a proiectat primul calculator personal produs de IBM. Este cunoscut și ca

„tatăl” IBM PC. Calculatorul, denumit IBM 1550 PC și lansat pe piață pe 12 august 1981, avea un ecran verde monocrom, un procesor Intel de 4,77 MHz și o memorie de 16 Kb. A se vedea și **IBM Personal Computer**.

eșantion (*sample*). 1. În cazul formatelor grafice de tip *raster*, este o componentă a unei culori. De exemplu, fiecare pixel al unei imagini *truecolor* este alcătuit din trei eșantioane: *red*, *green* și *blue*. Dacă imaginea are adâncimea de 24 de biți, atunci fiecare eșantion are adâncimea de 8 biți. O imagine *grayscale* utilizează, de asemenea, un eșantion cu adâncimea de 8 biți. Rezultă că adâncimea unui eșantion (exprimată în biți/eșantion) indică precizia cu care sunt specificate gradul de gri, respectiv culorile în imaginea care include eșantionul respectiv. Este important de înțeles că adâncimea unui eșantion nu trebuie confundată cu adâncimea unui pixel. Valorile lor coincid, într-adevăr, în imaginile *grayscale* și *truecolor*, dar sunt diferite în imaginile *index-color*, deoarece adâncimea unui eșantion se referă la valorile culorilor în paletă, în timp ce adâncimea unui pixel se referă la valoarea indexului culorii acestuia. De exemplu, valorile culorilor în paletă sunt reprezentate utilizând 24 de biți (8 biți pe eșantion), dar indecșii corespunzători pixelilor sunt reprezentați utilizând 8 biți sau mai puțin. A se vedea și **formate raster**. 2. Ceea ce este prelevat la intervale regulate în procesul de conversie a unui semnal

analogic la formatul digital. A se vedea și **eșantionare**.

eșantionare (*sampling*). Prelevarea, la intervale regulate, a unor eșantioane ale unui parametru (*e.g.*, semnal electric furnizat de un microfon), care apoi sunt convertite în format binar. Metoda este utilizată în scopul convertirii semnalelor analogice la formatul digital. Procesul de eșantionare este caracterizat de rata de eșantionare și de precizia eșantionării. A se vedea și **precizia eșantionării și rată de eșantionare**.

Ethernet. Familie de tehnologii pentru rețelele LAN, standardizate ca IEEE 802.3. Rețelele Ethernet utilizează o topologie de tip stea sau magistrală, cu ajutorul unei combinații de cablu bifilar torsadat (pentru rețea) și fibre optice (pentru conectarea la *backbone*). A se vedea și **backbone**, **Fast Ethernet** și **Gigabit Ethernet**.

Eudora. Client de *e-mail* utilizat cu sistemele de operare *Microsoft Windows* și *Apple Mac OS* (<http://www.eudora.com/>). A fost creat de Steve Dorner în 1988. În 1991 a fost achiziționat de Qualcomm. În 2006, această companie a oprit dezvoltarea versiunii comerciale a clientului și a finanțat crearea unei versiuni *open source*, bazată pe *Mozilla Thunderbird*. Aceasta este dezvoltată de Mozilla sub numele de cod *Penelope*. În perioada sa de glorie (în anii 1990), *Eudora* a fost folosit de zeci de milioane de utilizatori, însă a fost eliminat de pe

piață, în principal din cauza clientului gratuit de *e-mail* distribuit odată cu sistemul de operare Windows. A se vedea și **Qualcomm**.

Europeana. Biblioteca multimedia *online* deschisă pentru public pe 20 noiembrie 2008 și inaugurată la Palais de Charles de Lorraine, în Biblioteca Regală Națională din Bruxelles. La adresa www.europeana.eu utilizatorii pot accesa un portal care la momentul lansării includea peste două milioane de cărți, înregistrări, fotografii, documente de arhivă, lucrări de artă plastică și filme care provin din bibliotecile naționale și instituțiile culturale din toate țările membre ale Uniunii Europene.

Având ca scop crearea unor mijloace moderne de explorare a patrimoniului cultural european, *Europeana* oferă oricărei persoane interesate de opere literare, artă, știință, politică, istorie, arhitectură, muzică și cinema posibilitatea de a avea acces rapid la cele mai cunoscute opere de artă și colecții europene în cadrul unui singur portal disponibil în toate limbile UE. În anul 2010, *Europeana* va oferi acces la milioane de lucrări care reprezintă diversitatea culturală a Europei, având secțiuni interactive, de exemplu comunități dedicate anumitor subiecte.

Proiectul *Europeana* a fost inițiat în 2005 de Comisia Europeană, evoluând prin colaborarea cu bibliotecile naționale și cu alte instituții culturale ale statelor membre, precum și cu sprijinul Parlamentului European. Este un

proiect coordonat de *European Digital Library Foundation* și sprijinit de asociații de bibliotecari, arhive, muzee, arhive audiovizuale și instituții culturale, fiind găzduit de Biblioteca Națională a Olandei (Koninklijke Bibliotheek). În septembrie 2007, Parlamentul European a sprijinit acest proiect prin adoptarea unei rezoluții care prevedea crearea unei biblioteci digitale europene. Astăzi, *Europeana* este finanțată prin programul *eContent Plus*, urmărind atingerea obiectivelor incluse în proiectul *i2010*.

Comisarul european pentru Societatea Informațională și Media Viviane Reding spunea la festivitatea de lansare a portalului: „în acest moment fac apel la instituțiile culturale europene, la edituri și companii pentru a contribui la *Europeana* cu mai mult conținut în formă digitală. Ar trebui să facem din *Europeana* căminul pentru participarea creativă interactivă la îndemâna oamenilor care doresc să modeleze propria zonă de cultură europeană și să le-o împărtășească celorlalți. Obiectivul meu este ca în 2010 *Europeana* să includă cel puțin 10 milioane de obiecte.” De asemenea, Elisabeth Niggemann, director al Bibliotecii Naționale a Germaniei și președinte al *European Digital Library Foundation*, afirmă: „*Europeana* evidențiază mai mult instituțiile culturale pentru generația Web 2.0, care așteaptă să poată citi text, viziona filme și imagini și asculta într-un singur spațiu. Prin oferirea unei experiențe multimedia complete tinerilor interesați, aceștia vor

fi conectați la cultura europeană, la trecutul și prezentul acesteia.” A se vedea și **eContentPlus** și **i2010**.

European Schoolnet (EUN). Consorțiu nonprofit format la începutul anului 2009 din 31 de ministere ale educației din țări europene. Portalul asociat include mai multe site-uri europene ale unor proiecte care promovează dimensiunea europeană a educației și utilizarea noilor tehnologii în predare și învățare (<http://www.europeanschoolnet.org/>). Scopul EUN este educația prin proiecte, competiții, activități și schimburi de informații la toate nivelurile de învățământ preuniversitar, folosind tehnologii inovative.

evaluare (evaluation). În programare, determinarea valorii unei variabile, a unei expresii sau a acțiunii specificate de o instrucțiune. Se poate face la compilare sau la execuție. A se vedea și **compilare**.

eveniment (event). În Web, acțiune care se produce în cadrul paginilor HTML, în mod obișnuit în urma unor operațiuni executate de utilizator sau de browser.

Din punct de vedere istoric, modelele de eveniment au cunoscut modificări și variații de la un browser la altul, aspect care a determinat apariția unor probleme de compatibilitate între acestea. Totuși, un model al evenimentelor a fost standardizat de Consorțiul Web în DOM Events nivelul 2. De asemenea, DOM Events nivelul 3 este în curs de

standardizare. Majoritatea evenimentelor definite în acest model sunt comune tuturor browserelor.

Primul model pentru tratarea evenimentelor a fost introdus de Netscape în browserul *Navigator*, fiind utilizat apoi în toate browserele. Se disting două tipuri ale acestui model: *inline* și *tradițional*, prezentate în continuare. În tipul *inline* al modelului *Navigator* (numit și *model inline*), *handler*ele de eveniment sunt adăugate ca atribute ale elementelor HTML. Unui eveniment nu i se poate asocia (înregistra) decât un singur *handler*. Nu există posibilitatea de a elimina prin cod un *handler* de eveniment. În tipul *tradițional* al modelului *Navigator* (numit și *model tradițional*), *handler*ele de eveniment pot fi adăugate sau eliminate prin cod. La fel ca și în cazul tipului *inline*, fiecărui eveniment i se poate asocia (înregistra) un singur *handler*. Acesta este adăugat prin atribuirea metodei de tratare asociate proprietății cu același nume a obiectului element. Pentru a elimina un *handler*, proprietatea respectivă trebuie setată la valoarea `null`.

Consoțiul Web (W3C) a propus un model pentru tratarea evenimentelor reprezentat de interfața DOM `EventTarget`, utilizată pentru înregistrarea sau eliminarea procedurilor de tratare a evenimentelor. Modelul W3C este suportat de browserele *Mozilla Firefox*, *Safari*, *Opera* și *Konqueror*.

Evenimentele comune tuturor browserelor actuale și standardizate de W3C sunt: evenimente de mouse, evenimente

de tastatură, evenimente declanșate în formularele HTML și evenimente legate de cadre și ferestre. Majoritatea acestora sunt incluse în modulele DOM Events nivelurile 2 și 3. Evenimentele DOM permit unor limbaje de programare (e.g., JavaScript, JScript, ECMAScript, VBScript, Java) să înregistreze *handlers* de eveniment (i.e., funcții însărcinate cu responsabilitatea răspunsului automat în momentul în care se produce evenimentul respectiv) asociate nodurilor incluse în arborii DOM ai diverselor tipuri de documente (e.g., XML, HTML, SVG). A se vedea și **DOM** și **JavaScript**.

evenimente declanșate în formularele HTML (*events triggered in HTML forms*). Evenimentele declanșate în formularele HTML, incluse în *DOM Events*, sunt: `select`, `change`, `submit`, `reset`, `focus` și `blur`. Aceste evenimente sunt prezentate în continuare.

Evenimentul `select` este declanșat atunci când utilizatorul selectează o porțiune de text inclusă într-o casetă de tip *text* sau *textarea*. *Handlerul* de eveniment asociat este `onselect`.

Evenimentul `change` este declanșat atunci când datele incluse într-un câmp al unui formular sunt modificate de către utilizator. Acest eveniment este utilizat frecvent pentru a valida datele introduse de utilizator în formulare. *Handlerul* de eveniment asociat este `onchange`.

Evenimentul `submit` este declanșat înainte de trimiterea conținutului unui formular către server, în mod obișnuit prin execuția unui *click* asupra butonului *submit* inclus în acesta. Trimiterea conținutului nu se va produce dacă *handlerul* de eveniment `onsubmit` va returna valoarea `false`. Valoarea `true` sau orice altă valoare va determina trimiterea conținutului formularului. *Handlerul* de eveniment `onsubmit` este locul cel mai potrivit pentru plasarea validării datelor la nivel de formular, înainte ca acestea să fie trimise serverului sau să suporte alte prelucrări.

Evenimentul `reset` determină reinițializarea conținutului unui formular (la valorile implicite/inițiale ale câmpurilor acestuia) și este declanșat de obicei în urma execuției unui *click* pe butonul *reset*. *Handlerul* de eveniment asociat este `onreset`, inclus în marcajul `<form>`.

Evenimentul `focus` este declanșat atunci când o fereastră, un cadru sau un element al unui formular este activat (prin intermediul tastelor, executând *click* sau în urma apelului metodei `focus()`). La un moment dat poate să fie activ un singur obiect. *Handlerul* de eveniment este `onfocus`.

Evenimentul `blur` (inversul evenimentului `focus`) este declanșat atunci când o fereastră, un cadru sau un element al unui formular nu mai este activă/activ. Dezactivarea poate fi determinată de utilizator prin execuția unui *click* în afara ferestrei curente, a

cadrlui sau a elementului, prin utilizarea tastei Tab pentru a parcurge diversele elemente afișate pe ecran sau prin apelul metodei `blur()`. *Handlerul* de eveniment asociat este `onblur`. Acesta trebuie plasat în interiorul marcajului `<body>` al obiectelor `Window`, `Frame`, `Frameset` sau al marcajelor prin intermediul cărora sunt introduse câmpurile unui formular (e.g. de tip *text*).

Sucesiunea evenimentelor în cazul unui câmp al unui formular este următoarea: `focus`, `change`, `blur`. Ținând seama de această succesiune, se pot scrie funcții adecvate pentru tratarea complexă a întregului lanț de evenimente. A se vedea și **DOM și eveniment**.

evenimente de mouse (*mouse events*). Evenimentele de *mouse* precizate în DOM Events nivelul 2 sunt `click`, `mousedown`, `mouseup`, `mouseover`, `mousemove` și `mouseout`, analizate în continuare.

Evenimentul `click` este declanșat atunci când, folosind indicatorul *mouse*-ului, este executat un click asupra unui element al paginii Web. Acesta este definit ca o combinație dintre un eveniment `mouseup` și unul `mousedown`, generate unul după celălalt, în același loc de pe ecran. *Handlerul* de eveniment asociat este `onclick`. Pentru casete de validare (*checkbox*), legături, butoane *radio*, *submit* și *reset*, `onclick` poate returna valoarea `false`, ceea ce anulează acțiunea implicită asociată evenimentului

`click`. Dacă *handlerul* de eveniment întoarce valoarea `false`, butoanele *radio* și casetele de validare (*checkbox*) nu sunt setate, conținutul formularului nu este trimis (butonul *submit*) sau resetat (butonul *reset*).

Evenimentul `mousedown` este declanșat atunci când utilizatorul apasă butonul *mouse*-ului. *Handlerul* de eveniment asociat este `onmousedown`. Dacă acesta returnează `false`, acțiunea implicită este anulată.

Evenimentul `mouseup` este declanșat atunci când utilizatorul eliberează butonul *mouse*-ului. *Handlerul* de eveniment asociat este `onmouseup`. Dacă acesta returnează `false`, acțiunea implicită este anulată. De exemplu, dacă `onmouseup` returnează `false` după o legătură acționată (i.e., a fost declanșat anterior evenimentului `mousedown`), pagina către care indică legătura nu va fi încărcată în browser.

Evenimentul `mouseover` este declanșat de fiecare dată când utilizatorul plasează indicatorul *mouse*-ului deasupra unui obiect sau a unei zone, venind din afara acestora. *Handlerul* de eveniment asociat este `onmouseover`. Este necesar ca acesta să returneze `true` pentru a seta proprietatea `status`. Implicit, valoarea atributului `href` al elementului a este afișată în bara de stare a browserului atunci când utilizatorul plasează indicatorul *mouse*-ului asupra legăturii corespunzătoare. Este posibil să se afișeze un anumit mesaj.

Evenimentul `mouseout` este declanșat de fiecare dată când utilizatorul

deplasează indicatorul *mouse*-ului în exteriorul unei zone sau al unei legături, din interiorul acestora. *Handlerul* de eveniment asociat este `onmouseout`. Este necesar ca acesta să returneze `true` pentru a seta proprietatea `status`.

Evenimentul `mousemove` este declanșat de fiecare dată când utilizatorul deplasează *mouse*-ul. Deoarece este declanșat foarte des, acesta nu reprezintă evenimentul implicit al vreunui obiect. *Handlerul* de eveniment asociat este `onmousemove`. Pentru a utiliza evenimentul `mousemove` cu un obiect, este necesar ca, în mod explicit, obiectul să fie setat pentru a-l captura. A se vedea și **DOM și eveniment**.

evenimente de tastatură (*key events*). Evenimentele de tastatură definite în DOM Events nivelul 3 sunt `keydown` și `keyup`. Acestea sunt utilizate în toate browserele actuale. De asemenea, este utilizat și evenimentul `keypress`. Cele trei evenimente amintite sunt prezentate în continuare.

Evenimentul `keydown` este declanșat de fiecare dată când utilizatorul apasă o tastă. *Handlerul* de eveniment asociat este `onkeydown`.

Evenimentul `keyup` este declanșat de fiecare dată când utilizatorul eliberează o tastă. *Handlerul* de eveniment asociat este `onkeyup`.

Evenimentul `keypress` este declanșat de fiecare dată când utilizatorul apasă sau ține apăsată o tastă. *Handlerul* de eveniment asociat este `onkeypress`. A se vedea și **DOM și eveniment**.

evenimente legate de cadre și ferestre (*events related to frames and windows*). Evenimentele legate de ferestre și cadre incluse în DOM Events sunt `load`, `unload`, `abort`, `error`, `resize` și `scroll`, prezentate în continuare.

Evenimentul `load` este declanșat atunci când browserul a terminat de încărcat documentul HTML. *Handlerul* de eveniment asociat este `onload`.

Evenimentul `unload` este declanșat de fiecare dată când utilizatorul părăsește un document. *Handlerul* de eveniment este `onunload`, acesta fiind utilizat în interiorul marcajelor `<body>` sau `<frameset>`.

Evenimentul `abort` este declanșat atunci când utilizatorul oprește încărcarea unei imagini executând *click* pe butonul *Stop* al browserului sau pe o altă legătură, înainte ca imaginea să fie complet încărcată în browser. *Handlerul* de eveniment asociat este `onabort`, acesta fiind utilizat în interiorul marcajului ``.

Evenimentul `error` este declanșat de fiecare dată când o imagine sau un document determină o eroare în cursul încărcării. Trebuie făcută distincția între erorile de browser (*e.g.*, utilizatorul execută *click* asupra unei legături către un URL inexistent) și erorile JavaScript (de execuție sau de sintaxă). *Handlerul* de eveniment `onerror` va fi declanșat numai la apariția unei erori JavaScript. Dacă acesta primește valoarea `null` (`onerror=null`), caseta de eroare standard JavaScript va fi eliminată. Pentru a înlocui caseta de eroare

standard, la apelul unei funcții JavaScript utilizând handlerul `onerror`, această funcție trebuie să returneze valoarea `true`.

Evenimentul `resize` este declanșat de fiecare dată când utilizatorul sau scriptul executat redimensionează o fereastră sau un cadru. *Handlerul* de eveniment asociat este `onresize`. Utilizarea acestui *handler* permite determinarea mărimii și pozițiilor elementelor ferestrei, modificarea dinamică a atributului `src` etc. A se vedea și **DOM** și **eveniment**.

evidențierea sintaxei (*syntax highlighting*). Caracteristică a unor editoare de text care permite afișarea acestuia (în special a codului-sursă) în diverse culori și fonturi, în funcție de categoria căreia îi aparține textul (*e.g.*, date, instrucțiuni). Evidențierea nu afectează semnificația textului, fiind destinată numai utilizatorului uman. A se vedea și **cod-sursă** și **sintaxă**.

excepție (*exception*). Eveniment care se produce în timpul execuției unui program provocând întreruperea cursului normal al acestuia. Excepțiile pot să apară din diverse cauze și pot să aibă niveluri diferite de gravitate, pornind de la erori fatale cauzate de echipamentul hardware și până la erori care sunt determinate de codul inclus în program. Atunci când se produce o astfel de eroare, sistemul generează automat un obiect de tip excepție care conține informații despre aceasta și despre starea programului în momentul producerii sale. Crearea unui obiect de

tip excepție se numește *aruncarea unei excepții* (*throwing an exception*). Atunci când o metodă aruncă o excepție, sistemul de execuție este responsabil cu găsirea unei secvențe de cod dintr-o metodă care să o trateze. Căutarea se face recursiv, începând cu metoda care a generat excepția și mergând înapoi pe linia apelurilor către metoda respectivă. Secvența găsită se numește *analizor de excepție* (*exception handler*), iar interceptarea și tratarea excepției sunt cunoscute ca *prinderea excepției* (*catch the exception*). În principiu, variantele de tratare a excepțiilor pot fi: abandonarea programului; informarea utilizatorului și terminarea programului; informarea utilizatorului pentru ca acesta să ia măsuri pentru continuarea programului; luarea automată a unor măsuri pentru corectarea erorii, fără informarea utilizatorului.

excludere mutuală (*mutual exclusion*). Excluderea de către un *task* aflat într-o secțiune critică referitoare la o resursă a posibilității altor *task*-uri de a pătrunde în secțiuni critice care au ca obiect aceeași resursă. Pentru înțelegerea definiției anterioare, în continuare sunt prezentate câteva noțiuni utile. Resursa este un element necesar unui *task* pentru a putea fi rulat. Resursele sunt materiale (*e.g.*, procesorul, memoria internă, dispozitivele de I/E) și logice (*e.g.*, subrutine, variabile, fișiere). Resursa locală poate fi accesată de un singur *task*, în vreme ce resursa comună poate fi accesată de cel puțin două

task-uri. Resursa critică este o resursă comună care admite ca la un moment dat numai o sesiune de lucru asupra ei să fie în derulare. Secțiunea critică este o porțiune a unui program prin care se realizează o sesiune de lucru asupra unei resurse critice (*i.e.*, cel puțin două *task-uri* necesită aceeași resursă critică în același interval de timp).

Excluderea mutuală poate fi realizată prin utilizarea semafoarelor, a fani-oanelor de excludere, a blocurilor-resursă și prin dezactivarea întreprinderilor.

(a) **executa** (*execute*). A duce la îndeplinire o instrucțiune exprimată în cod-mașină. În acest scop, codul-mașină (inclus într-un fișier executabil sau obținut prin interpretarea pe rând a instrucțiunilor conținute în programul-sursă) este încărcat în memoria calculatorului și executat de micro-procesor. A se vedea și **cod-mașină**.

execuție concurentă (*concurrent execution*). Numită și *execuție paralelă*, desemnează execuția aparent simultană a două sau mai multe programe. Pentru a o realiza sunt utilizate tehnici de partajare a timpului.

Expat. Bibliotecă *open source* pentru procesarea orientată pe flux a documentelor XML, scrisă în C. Este inclusă într-un număr mare de proiecte *open source* (*e.g.*, *Apache HTTP Server*, *Mozilla*, *Perl*, *Python*, *PHP*). Biblioteca este dezvoltată de James Clark și alții începând cu 1998 și are versiuni pentru majoritatea sistemelor

de operare. Proiectul *Expat* este postat la adresa <http://expat.sourceforge.net/>. A se vedea și **XML**.

explozie informațională (*information explosion*). Sinonim cu *revoluție informațională* (*information revolution*), termenul se referă la creșterea rapidă (explozivă) a cantității de informații publicate prin diverse mijloace (presă, radio, televiziune, Internet). Termenul a fost utilizat pentru prima oară într-un supliment al ziarului *New York Times* (care conținea o reclamă plătită de compania IBM) din data de 30 aprilie 1961.

expresie (*expression*). Combinație de valori, variabile, operatori și funcții care poate fi evaluată în acord cu regulile de precedență și asociere care funcționează în cadrul unui limbaj de programare. Valoarea obținută în urma evaluării poate fi atribuită unei variabile, transmisă ca argument unei funcții, folosită în altă expresie etc.

expresie regulată (*regular expression*). Șir de caractere care descrie o mulțime (chiar infinită) de cuvinte posibile, reprezentând un șablon sau tipar (*pattern*, în limba engleză).

Expresiile regulate sunt utilizate pentru recunoașterea și manipularea șirurilor de caractere. Puterea expresiilor regulate derivă din posibilitatea de a include alternative și repetiții. De asemenea, expresiile regulate sunt utilizate pentru validarea documentelor XML, prin intermediul DTD și al schemelor XML.

O expresie regulată definește un limbaj (*i.e.*, o mulțime de șiruri), iar toate cuvintele limbajului respectă o regulă impusă de expresia regulată respectivă. Limbajele care pot fi descrise cu expresii regulate se numesc *limbaje regulate*, iar celelalte *limbaje neregulate*.

Ca orice limbaj, și cel definit prin intermediul unei expresii regulate utilizează un alfabet. Cele mai simple expresii regulate sunt cele care descriu limbaje al căror alfabet este reprezentat de un singur caracter (acesta poate fi orice caracter, inclusiv orice literă și orice cifră). Unele caractere (numite *metacaractere*) au semnificații speciale. Pentru a fi folosite, acestea trebuie citate (*quote*, în limba engleză), prin plasarea caracterului *backslash* („\”) înaintea fiecăruia. Metacaracterele sunt utilizate pentru alcătuirea unor construcții cu rolul de a identifica secvențe de caractere într-un text. În continuare sunt prezentate câteva reguli utilizate în sintaxa PCRE (*Perl-Compatible Regular Expressions*).

O expresie regulată trebuie delimitată de două caractere (aceleași). Ca delimitatori se pot utiliza orice caractere (*e.g.*, *slash*), cu excepția celor alfanumerice și a caracterului *backslash*. Dacă delimitatorul trebuie folosit în expresia regulată, este necesar ca acesta să fie securizat prin alăturarea caracterului *backslash*. Delimitatorul de sfârșit poate fi urmat de modificatori (*e.g.*, *i*, care determină efectuarea unei căutări insensitive).

Metacaracterul „|” reprezintă *operatorul de alternare*, fiind utilizat pentru a specifica o listă de opțiuni alternative, ca în exemplul următor: expresia regulată `/marca(nt|je)/` desemnează unul dintre cuvintele `marca`, `marcant` și `marcaje`. Între parantezele rotunde sunt incluse subtipare (*subpatterns*).

Metacaracterele „*”, „+” și „?” reprezintă operatori pentru specificarea repetițiilor. Ei se aplică în dreapta unei expresii regulate, făcând-o să se potrivească repetitiv. Fiecare operator implică un număr minim și unul maxim de repetiții. În general, pentru a specifica limitele minimă și maximă ale numărului de repetiții se folosesc acoladele (`{` și `}`), utilizându-se sintaxa `{m,n}`, în care *m* și *n* sunt numere întregi pozitive. Aceste numere respectă următoarele reguli: *m* trebuie să fie mai mic sau cel mult egal cu *n* și amândouă trebuie să fie mai mici decât 65536. De exemplu, `{2,5}` precizează un număr minim de 2 și unul maxim de 5 repetiții. Astfel, expresia regulată `/a{2,5}/` desemnează oricare dintre cuvintele `aa`, `aaa`, `aaaa` și `aaaaa`.

Pentru a grupa expresiile regulate se utilizează parantezele rotunde, ca în exemplul prezentat în continuare. Expresia `/mn*/` desemnează orice șir format din litera *m* urmată de un număr oarecare de ori (inclusiv niciodată) de litera *n*. În schimb, expresia `/(mn)*/` desemnează orice șir reprezentat de grupul *mn* repetat de un număr oarecare de ori (inclusiv niciodată). Astfel,

operatorul „*” se aplică grupului mn, nu numai literei n.

expresie relațională (*relational expression*). Expresie în care se utilizează un operator relațional pentru a compara două sau mai multe expresii. O astfel de expresie are ca rezultat o valoare booleană (*i.e.*, adevărat sau fals). A se vedea și **operator relațional**.

extensie de fișier (*file extension*). Set de caractere (de obicei trei) adăugat numelui de fișier pentru a reflecta conținutul acestuia. Este adăugat automat de programul cu care a fost creat fișierul (*e.g.*, *doc*, *xls*) sau manual, de către utilizator. A se vedea și **fișier**.

extranet (*extranet*). Rețea privată a unei companii sau organizații folosită în scopul partajării informațiilor sau operațiunilor cu partenerii de afaceri sau cu colaboratorii. Furnizează diverse niveluri de accesibilitate în funcție de identitatea utilizatorului (partenerului). A se vedea și **Internet** și **extranet**.

eZ Publish. CMS și mediu de dezvoltare având ca funcționalitate crearea site-urilor de comerț electronic (<http://ez.no/ezpublish>). Poate fi utilizat sub licență profesională (*eZ Publish Pro* sau *Enterprise*) sau sub licență *open source* GPL. A se vedea și **CMS**.

F

Facebook. Cea mai populară rețea socială online creată în anul 2009 (<http://www.facebook.com/>). A se vedea și **rețea socială online**.

familie de fonturi (font family). Colecție de fonturi cu același aspect global, concepute pentru a fi utilizate împreună, dar care pot să difere prin grosime și stil, dar nu și prin design (e.g., *Times* este o familie de fonturi, în timp ce *Times Bold* și *Times Italic* sunt fonturi individuale incluse în familia *Times*). A se vedea și **font**.

FAQ (Frequently Asked Questions, întrebări frecvente). Listă de întrebări (și răspunsuri la acestea) presupuse a fi formulate cel mai des de utilizatori în legătură cu un anumit subiect.

FastFlip. Aplicație Google (<http://fastflip.googlelabs.com/>) pentru citirea rapidă a articolelor publicate, printre altele, pe BBC, *The New York Times*, *The Washington Post*, *BusinessWeek*, *Elle*, *Cosmopolitan*, *Marie Claire*, *Newsweek* și *Popular Mechanics*. Totuși, articolele nu sunt disponibile în întregime prin *Fast Flip*, pentru citirea lor integrală fiind necesară vizitarea site-urilor publi-

cațiilor de pe care au fost luate și, la nevoie, logarea pe aceste site-uri. Avantajul principal al aplicației este viteza pe care o oferă. A se vedea și **Google**.

Fast Ethernet. Termen utilizat pentru a desemna o serie de standarde Ethernet care asigură o rată de transfer de 100 Mbps. A se vedea și **Ethernet** și **Gigabit Ethernet**.

favicon (Favorites icon). Cunoscut și ca *shortcut icon*, *url icon* sau *bookmark icon*, este un icon asociat unui site Web, afișat în bara de adrese a browserului, în stânga URL-ului. De asemenea, favicon-ul este alăturat adresei Web în lista *Favorites/Bookmarks*, însoțind și titlul paginii în browserele bazate pe taburi. Favicon-ul trebuie să fie bine ales, deoarece utilizarea lui are ca scop furnizarea unui indiciu cu privire la natura site-ului sau a conținutului acestuia.

Dimensiunile obișnuite ale unui favicon sunt 16x16 pixeli, dar pot fi utilizate și 32x32 și 48x48, toate având adâncimea de culoare de 8 sau 24 biți. Pentru a fi acceptat de toate browserele, favicon-ul trebuie să fie în format *ico*. Acesta a fost înregistrat în anul 2003 la IANA cu tipul MIME `image/vnd.microsoft.icon`, dar poate fi utilizat în continuare și tipul mai vechi `image/x-icon`. Browserul *Mozilla Firefox* permite specificarea unor favicon-uri în formatele *ico*, *PNG* și *GIF*, fiind capabil să afișeze și gif-uri animate ca favicon-uri.

Specificarea unui favicon se realizează prin plasarea marcajului link în secțiunea *head* a documentului (X)HTML. Iată exemple de utilizare:

```
<link rel="shortcut icon"
      type="image/x-icon"
      href="/imagini/myicon.ico">
```

(cod acceptat de browserele *Internet Explorer* și *Mozilla Firefox*);

```
<link rel="icon"
      type="image/
      vnd.microsoft.icon"
      href="/imagini/myicon.ico">
```

(cod acceptat de browserul *Mozilla Firefox*). A se vedea și **icon**, **MIME** și **IANA**.

fax-modem (*fax-modem*). Dispozitiv care poate fi atașat unui calculator permițând transmiterea documentelor electronice sub formă de faxuri. Acesta este însoțit de programe de comunicație care oferă facilități indisponibile în cazul faxurilor obișnuite (e.g., printr-un fax-modem se poate difuza simultan un document fax mai multor destinatari).

FDM (*Frequency Division Multiplexing, multiplexare cu divizarea frecvenței*). Formă de multiplexare a semnalelor care presupune atribuirea unei benzi de frecvență fiecărui semnal care provine de la un utilizator al unui mediu de transmisie. Atunci când FDM este utilizată pentru a permite mai multor utilizatori să partajeze un canal fizic de comunicație, este denumită „acces multiplu cu divizarea frecvenței” (FDMA, *frequency-division multiple access*). A se vedea și **multiplexare**.

feed (*feed*). Denumit și *flux de știri* sau *canal de știri*, este un fișier realizat într-un format special, bazat pe XML (i.e., RSS sau Atom), care furnizează informații de sumar despre articolele noi publicate într-un site sau un blog. Distribuitorii de conținut le oferă utilizatorilor posibilitatea de a se abona la *feed*-uri prin intermediul unor instrumente dedicate denumite *agregatoare* (e.g., *Bloglines*, *Feed Demon*, *FeedReader*, *Google Reader*).

După crearea unui *feed* RSS, se recomandă validarea acestuia, folosind un validator (de fapt, un verificator de sintaxă). Acesta verifică dacă în alcătuirea *feed*-ului a fost respectată sintaxa RSS. Dintre validatoarele *online* pot fi amintite: *RSS Validator*, *FEED Validator*, *Feed Validation Service*, *Weblogs.com Validator*, *RSS and Atom Validator*, *RSS 1.0 (RDF) Validator*. A se vedea **Atom** și **RSS**.

FeedBurner. Sistem *online* utilizat pentru managementul *feed*-urilor, lansat în 2004 (<http://www.feedburner.com/>). Aplicația corectează un *feed* (dacă este necesar), îi adaugă diverse facilități și îl publică din nou la o adresă de pe site-ul propriu.

Pentru a putea utiliza aplicația *FeedBurner* este necesar să se creeze un cont și să se adauge (înregistreze) unul sau mai multe *feed*-uri. Iată câteva dintre facilitățile oferite utilizatorilor de această aplicație: contorizarea abonaților, optimizarea și publicarea *feed*-urilor.

Contorizarea avansată a abonaților este una dintre cele mai cunoscute facilități, reprezentând motivul cel mai important pentru care mulți utilizatori își transferă *feed*-ul pe *FeedBurner*. Contorizarea efectuată de aplicație permite generarea unui raport care include: numărul utilizatorilor care s-au abonat la *feed*, aplicațiile folosite pentru abonare, un grafic care permite efectuarea unei analize a site-ului/blogului.

Aplicația le furnizează utilizatorilor o serie întreagă de unelte folosite în scopul optimizării *feed*-ului. Iată câteva dintre acestea: *SmartFeed* (transformă oarecum structura *feed*-ului, făcându-l să fie compatibil cu aproape orice cititor de *feed*-uri); *FeedFlare* (permite afișarea la sfârșitul fiecărui articol linkuri care l-ar putea ajuta pe cititor la *sharing* sau *bookmarking*; pot fi afișate numărul de comentarii la articol, linkuri către site-uri sociale etc); *Sumarry Burner* (crează un sumar – primele *n* caractere ale articolului – pe care îl afișează cititorului); *Convert Format Burner* (permite conversia *feed*-ului la un alt format); *SmartCast* (permite adăugarea în *feed* de legături către conținut multimedia; acestea pot fi atât elemente *iTunes*, cât și *Yahoo! Media RSS*).

Dintre instrumentele oferite de *FeedBurner* și utilizate pentru publicarea *feed*-urilor pot fi amintite: *FeedCount* (implementată prin intermediul unui buton care indică numărul de abonați la *feed*, acesta putând fi

afișat în site/blog prin includerea unei secvențe XHTML; în cazul în care numărul de abonați este mic, se recomandă ca această facilitate să nu fie utilizată); *E-mail Subscription* (facilitate foarte des utilizată; aceasta generează un *newsletter* care include ultimele articole publicate și pe care abonații îl primesc în mod regulat prin *e-mail*); *Headline Animator* (permite generarea unei imagini animate care afișează titlurile ultimelor articole de pe site/blog); *PingShot* (realizează trimiterea unei notificări către motoarele de căutare de *feed*-uri la fiecare apariție a unui nou articol pe site/blog); *Awareness API* (este un API dedicat dezvoltatorilor, care permite crearea unor aplicații externe *FeedBurner* pentru afișarea, analizarea și promovarea traficului de date al *feed*-ului).

FeedBurner poate fi utilizată și pentru managementul *feed*-urilor Podcast. Aplicația furnizează butonul *Play Now*, care oferă posibilitatea ascultării/vizualizării conținutului media în browserul Web.

Există două motive pentru care proprietarii de *feed*-uri ar trebui să utilizeze *FeedBurner*: dacă numărul abonaților la *feed* este mare, crește posibilitatea ca aceștia să realizeze un consum de bandă ridicat, ceea ce va pune în dificultate gazda site-ului/ blogului. În schimb, dacă *feed*-ul este „ars” folosind *FeedBurner*, consumul este transferat către serverul care găzduiește această

aplicație; dacă site-ul/blogul este *offline*, *feed*-ul poate fi totuși consultat, acesta aflându-se pe serverul *FeedBurner*.

Există și alte aplicații de analiză a *feed*-urilor. Una dintre acestea este *BlogPerfume*, care oferă o serie de statistici proprii asupra *feed*-urilor „arse” cu *FeedBurner*. În acest scop este necesară activarea pe *FeedBurner* a serviciului *Awareness API* pentru *feed*-urile care vor fi analizate de *BlogPerfume*. A se vedea și **feed**.

FeedDemon. Agregator pentru platforma Windows, care se poate sincroniza cu aplicațiile *Bloglines* și *NewsGator Online*. Poate importa liste OPML (<http://www.feeddemon.com/>). A se vedea și **agregator** și **feed**.

feed reader (*cititor de feed-uri*). A se vedea **agregator**.

FeedReader. Agregator de *feed*-uri gratuit pentru platforma Windows, care asigură un mediu de lucru robust și intuitiv. Oferă posibilitatea organizării *feed*-urilor (<http://www.feedreader.com/>). A se vedea și **agregator** și **feed**.

fereastră (*window*). Zonă de pe ecranul calculatorului, de formă în general dreptunghiulară, folosită în cadrul unei interfețe grafice (GUI) pentru interacțiunea cu utilizatorul. O fereastră poate fi deplasată, redimensionată, minimizată, maximizată, restaurată și închisă.

Orice program de aplicație este rulat în cadrul unei ferestre principale (numită

fereastră de aplicație). În cursul interacțiunii cu utilizatorul, în funcție de necesități, pot fi create ferestre suplimentare (e.g., de dialog, de proprietăți), care sunt modale (utilizatorul este solicitat să interacționeze numai cu aceasta) sau nemodale (oferă posibilitatea interacțiunii și cu alte ferestre). A se vedea și **fereastră activă**, **fereastră inactivă**, **GUI** și **interfață grafică**.

fereastră activă (*active window*). Într-un mediu care poate afișa simultan mai multe ferestre pe ecran (prin intermediul *managerului de ferestre*), este fereastra curentă (i.e., cea al cărei conținut afișat va fi afectat de următoarea comandă introdusă de la tastatură, de mișcarea cursorului, de introducerea unui text etc.). Managerul de ferestre permite schimbarea ferestrei active. A se vedea și **fereastră** și **fereastră inactivă**.

fereastră inactivă (*inactive window*). Într-un mediu care poate afișa simultan mai multe ferestre pe ecran (prin intermediul *managerului de ferestre*), este o fereastră neutilizată în mod curent. Poate fi ascunsă parțial sau total de alte ferestre. O fereastră inactivă poate deveni activă prin selectarea ei de către utilizator. A se vedea și **fereastră** și **fereastră activă**.

fermă de wiki (*wiki farm*). Un server sau o colecție de servere care găzduiește mai multe site-uri wiki. Printre cele mai cunoscute ferme de wiki sunt *PeanutButterWiki*, *Socialtext*, *Wetpaint*, *PTTWiki* și *Wikia*. A se vedea și **wiki**.

fibră optică (*optical fiber*). Fibră de sticlă cu un grad foarte ridicat de puritate, de grosimea unui fir de păr, utilizată pentru transmiterea luminii (având în mod obișnuit frecvența în domeniul infraroșu) cu pierderi minime, prin fenomenul de reflexie totală. Transmiterea datelor prin fibra optică necesită conversia impulsurilor electrice în lumină (cu ajutorul unor dispozitive laser), aceasta fiind transmisă apoi printr-un mănunchi de fibre optice (numit cablu optic). La destinație, lumina este reconvertită în impulsuri electrice. Fibrele optice sunt utilizate astăzi pe scară largă în telecomunicații și rețele de calculatoare.

O fibră optică are următoarele părți componente: miez (prin care este transmisă lumina), înveliș optic (acoperă miezul și reflectă total lumina) și înveliș protector (realizat din material plastic, are rolul de a proteja primele două componente împotriva umezelii și zgârieturilor).

Dintre avantajele utilizării fibrelor optice pot fi amintite rata de transfer foarte mare în raport cu celelalte tipuri de conexiune utilizate și posibilitatea de instalare rapidă, favorizată de greutatea redusă.

FIFO. Acronim pentru *First In, First Out* (primul intrat, primul ieșit), reprezintă o modalitate de organizare și manipulare a datelor. În particular, expresia descrie un principiu de procesare a datelor organizate într-o structură numită *coadă*. A se vedea și *coadă*.

filtrarea datelor (*data filtering*). Mecanism prin care se validează datele introduse într-o aplicație. Acesta trebuie realizat astfel încât să nu poată fi ocolit (voluntar sau involuntar) și să nu permită ca date invalide să treacă drept date valide și invers.

Filtrarea datelor presupune: determinarea surselor, validarea (se verifică dacă datele respectă un anumit format, e.g., cel al unei adrese de *e-mail*), utilizarea unei convenții de nume strictă pentru date, folosirea unei abordări de tip *white list* pentru date (aplicația va utiliza numai datele conținute în această listă, iar datele vor fi incluse în listă numai după validarea lor).

filtru de pachete (*packet filter*). Una dintre cele mai simple și mai vechi forme de *firewall*. Acceptă sau respinge traficul pe baza adreselor sursă și destinație și, eventual, a tipului de trafic. A se vedea și **firewall**.

FinAid. Înființat în anul 1994 ca un serviciu public, a devenit odată cu trecerea timpului cea mai importantă sursă de informații legate de ajutorul financiar de care pot beneficia studenții sau cei care doresc să studieze în SUA (<http://www.finaid.org/>).

Site-ul *FinAid* are patru secțiuni principale: *Scholarship* (oferă posibilitatea de a căuta bursele care se potrivesc profilului personal); *Loans* (oferă informații despre posibilitățile de obținere a unor împrumuturi financiare, în calitate de student); *Savings* (sunt prezentate informații despre ofertele de

cazare și masă puse la dispoziție de universități studenților lor); *Military Aid* (prezintă informații – destinate cetățenilor americani – despre ajutoarele financiare acordate studenților care doresc să urmeze o carieră în domeniul militar, precum și veteranilor sau familiilor acestora).

De asemenea, site-ul oferă informații despre alte tipuri de ajutoare pentru studiu (*Other Types of Aid*) și despre redactarea aplicațiilor pentru obținerea de resurse financiare (*Financial Aid applications*). Pe site pot fi obținute răspunsuri la întrebările personale (*Answering Your Questions*) și pot fi calculate costurile presupuse de fiecare dintre programele de studiu (*Calculators*).

Find Satoshi (*Caută-l pe Satoshi*). Joc online dezvoltat de compania de jocuri multiplayer Mind Candy, pentru a afla în cât timp îl poate găsi comunitatea utilizatorilor de Internet pe Satoshi (un bărbat japonez) numai pe baza unei imagini (<http://www.findsatoshi.com/>). A se vedea și **teoria celor șase grade de separare**.

fir de execuție (*thread*). În programare, flux de instrucțiuni care se execută în interiorul unui proces. Un proces poate să fie format din mai multe asemenea fire, care se execută în paralel, având însă în comun toate resursele principale caracteristice procesului. Deci în interiorul unui proces firele de execuție împart între ele zona de date și execută porțiuni distincte din același cod. Deoarece zona de date este comună,

toate variabilele procesului vor fi văzute la fel de toate firele de execuție, orice modificare realizată de un fir devenind vizibilă pentru toate celelalte fire ale procesului. A se vedea și **proces**.

Firefox. A se vedea **Mozilla Firefox**.

firewall (*firewall*). Program instalat pe serverul unei rețele de calculatoare, asigură securitatea acesteia împotriva atacurilor venite din exterior, având rolul de a monitoriza, controla și, dacă este necesar, de a respinge pachetele de date incluse în traficul rețelei protejate. *Firewall*-urile pot fi clasificate în următoarele categorii: *firewall* filtru de pachete (în funcție de protocolul de comunicare utilizat, de adresa IP și de portul sursă sau destinație, se stabilesc reguli care să permită sau nu trecerea unui pachet de date); *firewall server proxy* (se utilizează următoarele două modele: *Circuit Level Gateway*, care face o filtrare sumară a pachetelor și *Application Level Gateway*, care ține cont de aplicațiile care schimbă pachete); *firewall* bazat pe controlul stării conexiunii și pe istoricul acesteia (dacă o conexiune a fost considerată la un moment dat sigură, se verifică dacă starea actuală a conexiunii este în concordanță cu cea anterioară).

Un *firewall* eficient trebuie să includă și un sistem de detectare a posibilelor atacuri (*Intrusion Detection System*), astfel încât atacurile *SYN Flooding* să nu blocheze accesul celorlalți utilizatori în sistem. A se vedea și **DoS** și **filtru de pachete**.

FireWire. Interfață serială pentru comunicații de mare viteză și transfer de date în timp real (până la 800 Mbps), utilizată de PC-uri și alte dispozitive digitale audio și video, dezvoltată inițial de Apple și standardizată de IEEE (standard 1394).

firmware (*firmware*). Întâlnit și sub denumirea de *microprogram* sau *microcod* (*microcode*), este un set de rutine software (programe) stocate în memoria internă ROM a unui dispozitiv (*e.g.*, camere foto digitale, telefoane mobile, playere MP3, console de jocuri, mașini de spălat). A se vedea și **BIOS**.

fișier (*file*). Unitatea fundamentală de stocare a datelor, care permite sistemului de operare și aplicațiilor să deosebească între ele seturile de date. Natura datelor conținute în fișiere diferă mult de la un tip de fișier la altul (*e.g.*, text, imagini, animații, filme, sunete sau combinații între acestea). În funcție de semnificația datelor pe care le conțin și de organizarea lor internă, fișierele au diverse formate predefinite. În mod obișnuit, fișierele sunt identificate folosind un nume și o extensie, separate printr-un punct. Extensia unui fișier coincide cu formatul său.

fișier ascuns (*hidden file*). Fișier care, în mod obișnuit, nu este inclus în lista afișată (și văzută de utilizator) a fișierelor care fac parte dintr-un director. Un fișier de acest tip include, de obicei, date și coduri importante pentru sistemul de operare. Ascunderea

este realizată cu scopul de a feri fișierul de ștergere sau modificări intenționate sau accidentale. În sistemele UNIX sau similare cu acesta, numele fișierelor ascunse sunt prefixate cu un punct (*e.g.*, */home/user/.config*). A se vedea și **fișier**.

fișier comprimat (*compressed file*). Fișier obținut după comprimarea conținutului unui alt fișier (*original*). Fișierul comprimat ocupă pe dispozitivul de stocare (*e.g.*, hard-disk) un spațiu de stocare mai mic decât cel original. Iată câteva extensii ale fișierelor comprimate: *.zip*, *.rar*, *.gz*, *.tar.gz*, *.lzh*, *.tgz*, *.lha*. A se vedea și (a) **comprima**.

fișier de date (*data file*). Fișier care conține date sub diverse forme (*e.g.*, text). Se deosebește de fișierul de program, care conține declarații, instrucțiuni și comenzi. A se vedea și **fișier de program**.

fișier de program (*program file*). Fișier care conține versiunea executabilă a unui program de calculator. Se deosebește de fișierul de date. A se vedea și **fișier de date**.

fișier de text (*text file*). Fișier cu extensia *.txt*, care poate fi citit și editat de orice editor simplu de text. Din acest motiv, fișierele de text sunt considerate independente de platformă (universale). Formatul de fișier implicit este ASCII. În cazul în care sunt utilizate caractere non-ASCII, este necesar să se aleagă o codificare corespunzătoare (*e.g.*, Unicode, UTF-8). A se vedea și **fișier**.

fișier temporar (*temporary file*). Fișier creat de sistemul de operare sau de un program de aplicație (e.g., *Word*), în diverse scopuri. Fișierul este șters automat după terminarea sesiunii de lucru. A se vedea și **fișier**.

Five o'Clock. Oferă două dicționare *online*, română-englez și englez-română (<http://www.fiveoclock.ro/>). Pot fi traduse cuvinte sau expresii.

Flash. Metodă des utilizată pentru adăugarea de animație și interactivitate paginilor Web. În mod obișnuit, *Flash* este utilizat pentru: crearea animațiilor și reclamelor, integrarea materialelor video în paginile Web, dezvoltarea aplicațiilor RIA. Formatul utilizat este SWF, fișierele corespunzătoare având în mod obișnuit extensia *.swf*. Acestea pot fi incluse ca obiecte într-o pagină Web sau rulate de sine-stătător într-un *Flash Player*.

Animațiile *Flash* incluse ca obiecte în paginile Web sunt vizualizate cu *Adobe Flash Player*, disponibil gratuit pentru instalare ca *plug-in* în browser. Acest *player* poate fi utilizat de asemenea și pe dispozitive mobile (e.g., telefoane). A se vedea și **Adobe Flash Player**, **codec**, **Flash Video**, **RIA** și **SWF**.

Flash Video. Format utilizat pentru livrarea în Internet a materialelor video vizualizate cu *Adobe Flash Player*. De asemenea, conținutul *Flash Video* poate fi inclus în fișierele SWF. Fișierele *Flash Video* (cu extensia *.flv*) pot fi rulate și de sine-stătător folosind un *player* FLV (e.g., *VLC*, *QuickTime*,

Windows Media Player care conțin codecuri externe).

Formatul este utilizat de furnizori importanți de materiale video în Internet, ca: *YouTube*, *Google Video*, *Yahoo! Video* etc. A se vedea și **Flash**.

Flex. A se vedea **Adobe Flex**.

Flickr. Instrument utilizat pentru încărcarea, marcarea și partajarea conținutului grafic (<http://www.flickr.com/>). *Flickr* (una dintre primele aplicații Web 2.0) este atât un site pentru găzduirea de imagini și video-uri, precum și o suită de servicii Web și o comunitate *online*. Site-ul este folosit de utilizatori pentru partajarea de fotografii, dar și de *bloggeri*, ca depozit de imagini.

Flickr a fost dezvoltat sub licență Creative Commons Licensing de Ludico (<http://www.ludicorp.com/>), o companie din Vancouver, care l-a lansat în februarie 2004. În martie 2005, Yahoo! a achiziționat Ludico și *Flickr*. În iunie 2005, conținutul *Flickr* a fost transferat de pe serverele din Canada pe servere din SUA. În decembrie 2006, spațiul disponibil în cadrul contului *Free* a fost mărit la 100 MB/lună, iar cel corespunzător contului *Pro* a devenit nelimitat. Pe 20 septembrie 2007, Yahoo! a închis propriul serviciu de *photo sharing* (*Yahoo! Photos*). Toți utilizatorii care au migrat către *Flickr* au primit gratuit (timp de 3 luni) un cont *Pro*. Începând cu aprilie 2008, *Flickr* permite încărcarea de clipuri video, contra cost (i.e., folosind un cont *Flickr Pro*).

Deși încărcarea clipurilor video este rezervată numai celor care dețin conturi *Pro*, acestea pot fi vizualizate de toți utilizatorii *Flickr*. Sunt acceptate formatele AVI, WMV, MOV, MPEG 1, 2, 3, precum și 3GP. Durata clipurilor video încărcate este limitată la 90 de secunde, iar dimensiunea fișierelor la 150 MB.

Flickr le oferă utilizatorilor posibilitatea de a-și organiza imaginile încărcate folosind tag-uri, acestea permițând celor care realizează căutări să găsească imagini pe baza unui subiect (e.g., numele unei localități). De asemenea, *Flickr* implementează conceptul *TagCloud*, pentru a furniza acces rapid la imaginile marcate cu cele mai populare tag-uri. *Flickr* oferă un serviciu Web (API) care le dă programatorilor posibilitatea să creeze aplicații, acestea punând la dispoziție aproape toate funcțiile la care au acces utilizatorii înregistrați pe site-ul *Flickr*. Crearea unui cont gratuit pe *Flickr* se face rapid, utilizând ID-ul Yahoo!. Implicit, numele contului creat coincide cu ID-ul amintit, dar poate fi și diferit de acesta, în funcție de dorința utilizatorului.

Partajarea de fotografii este o activitate inherent socială, *Flickr* fiind primul serviciu bazat pe Web care a implementat această activitate în spațiul virtual. Astfel, se poate considera că *Flickr* este o rețea socială *online* bazată pe text, imagini și clipuri video. A se vedea și **rețea socială online** și **Web 2.0**.

float. Tip de date utilizat în unele limbaje de programare (e.g., C, C++) pentru a declara variabilele care stochează numere reale în virgulă mobilă, ocupând patru octeți (32 de biți). A se vedea și **număr în virgulă mobilă**.

FLOPS (*floating-point operations per second*, operații în virgulă mobilă pe secundă). Măsură a vitezei cu care calculatoarele realizează operații în virgulă mobilă (e.g., calcule științifice). A se vedea și **operație în virgulă mobilă**.

FLOSS (*Free/Libre Open Source Software*). Termenul a fost utilizat pentru prima oară în 2001 de Ryshab Ayier Ghosh, pentru ca în anul următor să fie folosit într-un studiu al Comisiei Europene. Spre deosebire de *Libre Software*, care a fost introdus în scopul eliminării unei confuzii (dintre libertatea de utilizare și gratuitate), FLOSS este utilizat pentru a preveni o posibilă concurență neproductivă în ceea ce privește utilizarea uneia sau alteia dintre expresiile *Free Software* și *Open Source Software*.

Mișcările *Free Software* și *Open Source* reprezintă cele două fețe ale aceleiași monede. Astfel, mișcarea pentru *software* liber pune accentul pe importanța FLOSS în ceea ce privește libertățile civile și drepturile omului, în vreme ce mișcarea *open source* se concentrează pe promovarea aspectelor pragmatice și tehnice ale dezvoltării și utilizării acestui tip de aplicații.

Un pas important către crearea unei comunități *Open Source* în România a fost făcut de ROSI (<http://www.rosi.ro/>) și Agora Media (<http://www.agora.ro/>), prin organizarea periodică (anual, începând cu 2007) a conferinței internaționale numite *eLiberatica* (<http://www.eliberatica.ro>), la care participă personalități importante ale mișcării FLOSS. A se vedea și **Free Software** și **Open Source Software**.

flowchart (*organigramă*). Diagramă utilizată pentru reprezentarea unui algoritm sau proces. Operațiile sunt reprezentate folosind simboluri grafice (e.g., pătrate, romburi, elipse) conectate prin săgeți care indică fluxul datelor sau al controlului. Prima metodă grafică structurată pentru ilustrarea fluxului unui proces a fost introdusă în 1921 de Frank Gilbreth, membru ASME (*American Society of Mechanical Engineers*). John von Neumann și Herman Goldstine au dezvoltat în 1947 o metodă grafică (denumită inițial *diagram* și apoi *flow chart*) pentru descrierea programelor de calculator. A se vedea și **diagramă**.

Flowgram. Instrument *online* similar cu *VoiceThread*, dar combinat cu *screencasting*-ul. Altfel, se poate spune că *Flowgram* este o aplicație pentru *screencasting* multimedia. Conform site-ului *Flowgram*, acesta este un instrument de tip *Show and Tell* (arată și spune).

Aplicația *Flowgram* transformă orice conținut Web (e.g., pagini, imagini, *feed*-uri RSS/Atom, prezentări *Power*

Point) într-un flux digital personalizabil cu propriile note, voce și porțiuni de text evidențiate (*highlights*). Astfel, sunt create prezentări multimedia interactive complexe, cei care le vizualizează având posibilitatea de a controla paginile, executând scroll, click-uri pe legături, pot vizualiza secvențe video etc. Fluxurile create sunt denumite *Flowgrams*. Accesul la comunitate este liber. Instrumentul este complet gratuit, oferindu-se fără costuri găzduire gratuită pentru toate *Flowgram*-urile create. Acestea pot fi incluse de autor în site-ul/blogul propriu prin intermediul unui *widget*, cu un cod oferit de aplicație.

Pentru a putea folosi aplicația *Flowgram*, este necesară crearea unui cont de utilizator (*Register*). În acest scop sunt solicitate informații obligatorii (un nume de utilizator, adresa de *e-mail* și o parolă), precum și informații opționale. Utilizatorul are posibilitatea de a alege un avatar dintre cele oferite de aplicație sau unul pe care-l poate încărca de pe propriul calculator.

Instrumentul *Flowgram* poate fi folosit în scopuri diverse, inclusiv în domeniul educațional. În continuare sunt enumerate câteva modalități în care poate fi utilizat: realizarea de comentarii (folosind vocea sau textul) asupra propriilor fotografii postate pe *Flickr* sau *Facebook*; crearea de prezentări complexe, adăugând voce și note propriilor prezentări *PowerPoint*; crearea de cursuri care vor fi partajate cu studenții, aceștia având astfel posibilitatea de a posta comentarii; realizarea unor proiecte/

prezentări care implică includerea unor tipuri variate de conținut; crearea de materiale demonstrative în care sunt prezentate/promovate concepte, strategii, produse, servicii etc. A se vedea și **VoiceThread**.

folder (*dosar*). A se vedea **dosar**.

FOLDOC – Computing Dictionary.

Dicționar dezvoltat de Department of Computing al Imperial College (<http://foldoc.org/>). Oferă explicații pentru un număr mare de termeni (circa 15 000) utilizați în tehnica de calcul. Pot fi căutate explicații atât pentru cuvinte, cât și pentru expresii. Sunt oferite extensii FOLDOC pentru browserele populare (e.g., *Mozilla Firefox*, *Internet Explorer*).

folksonomy (*folksonomie*). Cunoscută și ca *marcare colaborativă*, *clasificare socială*, *indexare socială* sau *marcare socială*, este o practică și o metodă de creare colaborativă a *tag*-urilor și de management al acestora, în scopul adnotării și clasificării resurselor Web.

Termenul *folksonomy* provine din *folk* (sau *folks*; popular) și *taxonomy* (taxonomie), fiind introdus în 2007 de Thomas Vander Wal.

În contrast cu metodele de indexare tradițională a conținutului, *tag*-urile nu sunt create de experți, ci de consumatorii de conținut (acest aspect constituie, de fapt, esența unei folksonomii). Astfel, folksonomia reprezintă o taxonomie populară a Web-ului, adică o clasificare socială informală a acestuia.

Folksonomia (fără să fie denumită încă așa) a devenit populară începând cu anul 2004, odată cu apariția unor aplicații Web 2.0 de tip *social software*, cum ar fi cele de *social bookmarking* sau cele de adnotare a fotografiilor. O folksonomie are următoarele caracteristici: marcarea resurselor se face în mod independent; constituie un spațiu de nume (vocabular) plat (*flat namespace*); relațiile dintre *tag*-uri sunt obținute prin inferență; orice metodă de inferență este validă (deși unele sunt mai bune decât altele).

O clasificare este înțeleasă în general ca fiind rezultatul unui efort de ordonare într-un sistem ierarhic predefinit, în total dezacord cu abordarea non-ierarhică și non-exclusivă a unei folksonomii, ceea ce face ca utilizarea acestui termen în contextul marcării libere a resurselor să fie oarecum controversată (utilizatorii furnizează în legătură cu o resursă, prin intermediul *tag*-urilor folosite, propriile semnificații, conform propriei lor înțelegeri).

În jurul termenului s-a creat o dispută legată de semnificația sa. Astfel, unii autori fac distincția între o folksonomie (având sensul de colecție de *tag*-uri create de un individ cu scopul de a fi folosită personal) și un *colabullary* (un vocabular colectiv). Alți autori (majoritatea) utilizează termenul folksonomie cu sensul de vocabular colectiv.

Folksonomia poate reprezenta o cheie – disponibilă la un cost scăzut – către Web-ul semantic și ar putea fi baza noului Web educațional, în care fiecare

pagină conține metadate (*tag-uri*) pentru descrierea conținutului care pot fi citite cu diverse aplicații.

Marcarea socială a resurselor poate fi descrisă ca o activitate altruistă prin care oamenii contribuie la împărtășirea bazelor de cunoștințe, permițând dezvoltarea unor comunități virtuale în jurul unor interese comune. De asemenea, marcarea socială oferă informații furnizorilor de forță de muncă și managerilor despre ariile de interes, nevoile și obiceiurile utilizatorilor.

flux de știri (*feed*). A se vedea *feed*.

font (*font*). Set de caractere de același tip (*e.g.*, *Garamond*), același stil (*e.g.*, drept, italic) și aceeași grosime (*e.g.*, aldin). Altfel spus, un font este alcătuit din toate caracterele cu un anumit stil și o anumită grosime ale unui model (sau tip) de caracter. Iată exemple de fonturi: Arial bold, Arial italic aldin.

Fonturile sunt stocate în fișiere cu formate de tip *bitmap*, *outline* sau *stroke*. Un fișier în format *bitmap* conține o serie de puncte reprezentând imaginea unui caracter. Un fișier în format *outline* (numit și vectorial) conține instrucțiuni și formule matematice pentru descrierea unui caracter, astfel încât caracterele *outline* sunt scalabile la orice dimensiune (*e.g.*, *TrueType*, *PostScript*, *OpenType*). Fonturile *stroke* utilizează o serie de linii și informații adiționale pentru a defini profilul unui caracter.

Fonturile *bitmap* sunt ușor de utilizat, dar inflexibile, cerând câte un fișier

pentru fiecare dimensiune. În schimb, fonturile *outline* și *stroke* pot fi redimensionate, dar (spre deosebire de fonturile *bitmap*) necesită un software suplimentar pentru a fi interpretate (în format *bitmap*) în scopul afișării pe ecran sau printării. A se vedea și **editor de fonturi și familie de fonturi**.

font PostScript (*PostScript font*). Font de tip *outline* (vectorial) descris prin intermediul unor primitive grafice PS (*i.e.*, stabilite în limbajul PostScript) și destinat tipăririi la o imprimantă compatibilă PostScript. A se vedea și **font și PostScript**.

font monospațiat (*monospace font*). Font care include caractere de aceeași lățime (*e.g.*, litera *i* are alocat pe orizontală același spațiu ca și litera *m*). Fontul este similar celui folosit de mașinile de scris. A se vedea și **font și font proporțional**.

font proporțional (*proportional font*). Font care include caractere de diverse lățimi (*e.g.*, litera *i* are alocat pe orizontală un spațiu mai mic decât litera *m*). A se vedea și **font și font monospațiat**.

font scalabil (*scalable font*). Font care poate fi scalat pentru a obține caractere de diferite dimensiuni (*e.g.*, *TrueType*, create pentru utilizatorii de Windows și *PostScript*, realizate de Adobe Systems). A se vedea și **font**.

format de fișier (*file format*). Stabilește structura unui fișier, definind natura datelor incluse în acesta, modalitatea în care sunt stocate, precum

și modul în care sunt afișate pe ecran. A se vedea și **fișier**.

format deschis (*Open Format*). Specificație publică folosită pentru stocarea digitală a datelor, menținută în mod obișnuit de o organizație de standardizare și care poate fi utilizată de oricine. Spre deosebire de formatele deschise, formatele proprietare sunt definite și controlate de firme private (e.g., Microsoft).

Formatele deschise pot fi incluse în următoarele trei categorii: *text* (TXT, HTML/XHTML, OpenDocument, Open Office XML, PDF); *multimedia* (JPEG 2000, PNG, SVG, VRML/X3D, Ogg); *altele* (XML).

forma rețelelor sociale (*social networks form*). Factor-cheie al utilității unei rețele sociale pentru membrii lor. Rețelele dense sunt mai puțin utile decât cele cu multe spații libere și cu numeroase conexiuni slabe cu indivizi din afara rețelei principale. Rețelele deschise, cu legături slabe și spații sociale goale, introduc idei noi și oportunități pentru membrii lor într-o măsură mai mare decât rețelele închise, care au multe legături redundante. A se vedea și **rețea socială**.

formatare (*formatting*). 1. Adăugarea unor elemente de stil și de prezentare (e.g., margini, indentări, fonturi) într-un document (e.g., Word). 2. Pregătirea unui disc (e.g., hard-disk) în vederea stocării informațiilor.

formate metafile (*metafile formats*). Formate grafice (e.g., EMF, WMF,

CGM) care oferă posibilitatea de a stoca în fișier ambele tipuri de reprezentări ale unei imagini: *raster* și *vectorial*. A se vedea și **formate raster** și **formate vectoriale**.

formate raster (*raster formats*). Numite și *bitmap*, sunt formate grafice (e.g., BMP, GIF, JPEG, PNG). O reprezentare *raster* stochează o imagine într-un tablou bidimensional de pixeli. În fișierul corespunzător sunt stocate date despre fiecare pixel al imaginii. A se vedea și **BMP**, **GIF** și **JPEG**.

formate vectoriale (*vector formats*). Formate grafice (e.g., EPS, SWF, SVG, VML, XAML). În fișier sunt stocate numai elementele matematice necesare pentru generarea imaginii. Elementele de grafică vectorială includ diverse obiecte ca segmente de dreaptă și curbe, aspect care determină o flexibilitate sporită în raport cu formatele grafice *raster*, deoarece acestea din urmă trebuie să stocheze informații referitoare la fiecare pixel. În plus, formatele vectoriale permit integrarea imaginilor de tip *raster* și le pot combina cu informație vectorială pentru a realiza imagini complexe.

În ceea ce privește modul de stocare a informațiilor, acesta este destul de divers. Astfel, SVG și VML sunt formate textuale. În plus, formatul SVG poate fi comprimat. În schimb, SWF este un format binar, care poate fi comprimat. A se vedea și **SVG**, **SWF**, **VML**, **XAML**.

format nativ (*native format*). Format de fișier pe care îl utilizează intern o aplicație (e.g., formatul intern al aplicației *Word* diferă de cel al aplicației *WordPerfect*). Alături de cele interne, aplicațiile suportă și alte formate. Formatul oricărui fișier recunoscut este convertit de aplicație în formatul intern, înainte de utilizare.

formular electronic (*electronic form*). Formularele electronice sunt incluse în paginile Web, oferindu-le utilizatorilor posibilitatea de a furniza (introduce) date care, ulterior, vor fi transmise serverului în vederea prelucrării și stocării. Acolo vor fi procesate de scripturi scrise într-un limbaj de programare pentru server. Un formular include trei categorii de *câmpuri* (numite și *controale*): câmpuri cu scop multiplu (e.g., casetă de editare, casetă de validare, butoane radio, buton de expediere, buton de inițializare), listă de selecție și casetă de text multilinie.

În (X)HTML, pentru crearea unui formular se utilizează marcajul `<form>`. Acesta are o serie de atribute specifice, dintre care prezentăm în continuare două (indicăm și tipurile de valori posibile): `action="URL"` și `method="get|post"`. Atributul `action` are ca valoare URL-ul scriptului de prelucrare a datelor conținute în formular. Atributul `method` indică metoda HTTP prin care datele conținute în câmpurile formularului vor fi trimise scriptului. Cele două metode sunt `get` (metoda implicită) și `post`.

formă normală (*normal form*). Referitor la o bază de date relațională, set de criterii care furnizează o modalitate pentru determinarea gradului de vulnerabilitate la incoerențe logice și anomalii ale tabelelor componente. Formele normale sunt aplicabile tabelelor individuale. Dacă o bază de date este în forma normală n (acesta fiind un număr întreg cu valori cuprinse între 1 și 6), înseamnă că toate tabelatele din care este compusă sunt în forma normală respectivă. Cu cât forma normală este mai înaltă (i.e., n este mai mare), cu atât gradul de vulnerabilitate este mai redus. Fiecare tabel dat are o formă normală cea mai înaltă (HNF, *highest normal form*). Automat, aceasta va satisface și cerințele incluse în formele normale cu grad mai redus. A se vedea și **normalizare**.

Forth. Limbaj de programare imperativă, creat de Charles Moore în anii 1970. Este un limbaj interpretat, structurat, care utilizează execuția pe fire. Folosește notația postfix pentru expresiile matematice, iar programatorul trebuie să lucreze direct cu stiva programului. O implementare Forth este menținută de GNU Project. A se vedea și **notație postfix, limbaj interpretat și programare imperativă**.

forum (*forum*). Denumit și *tablă de mesaje* (*message board*), este un site de discuții *online*, echivalentul modern al tradiționalului *avizier* (*bulletin board*) oferit de sistemele BBS. Din punct de

vedere tehnic, un forum este o aplicație Web care gestionează conținutul creat de utilizatori. În mod obișnuit, un forum necesită înregistrarea utilizatorilor. După confirmarea înregistrării, aceștia devin membri, calitate care le conferă dreptul de a posta mesaje pe forum.

Orice forum are o structură arborescentă, conținând la nivelul de jos așa-numitele *topics* (subiecte), denumite și *threads* (fire). Un subiect include unul sau mai multe *posturi* (sau *mesaje*).

Un forum este condus de o serie de persoane denumite administratori și moderatori. Moderatorul este un utilizator cu dreptul suplimentar de a edita (inclusiv de a șterge) subiectele și mesajele tuturor celorlalți utilizatori, cu scopul de a păstra decența discuțiilor sau de a elimina mesajele spam. Administratorul are rolul de a gestiona aspectele tehnice care vizează buna funcționare a aplicației. A se vedea și **aplicație Web**.

FPDF. Bibliotecă scrisă în PHP, dezvoltată de Olivier Plathey, utilizată pentru generarea dinamică a documentelor PDF (<http://www.fpdf.org/>).

fractal (*fractal*). Termen introdus în 1975 de Benoît Mandelbrot pentru a desemna o serie de forme neregulate având proprietatea de autosimilaritate. Aceasta înseamnă că un fractal poate fi divizat în părți mai mici similare, fiecare, cu figura inițială. Fractalii sunt utilizați în programele de grafică pe

calculator pentru a crea imagini cu aspect natural (arbori, ferigi, nori, fulgi de zăpadă, cristale, lanțuri muntoase, țărnul mării, conopida, broccoli, sistemul de vase sanguine și vase pulmonare). Teoria fractalilor este expusă pe larg de B. Mandelbrot în cartea *The Fractal Geometry of Nature* (1982).

freeBSD. Versiune de BSD UNIX distribuită gratuit, destinată calculatoarelor personale IBM și celor compatibile. A se vedea și **BSD**.

Free Software. Expresie și concept introduse de Richard Stallman în 1983 pentru a desemna *software*-ul liber, ai cărui utilizatori au posibilitatea de a-l folosi, studia, modifica și redistribui. Odată cu introducerea expresiei amintite a fost lansată și mișcarea cu același nume (*free software movement*). În 1985, același Richard Stallman a pus bazele fundației non-profit *Free Software Foundation* (FSF), în scopul susținerii mișcării create anterior. FSF a publicat *Free Software Definition* (FSD), o definiție a ceea ce înseamnă *Free Software*. Expresia este definită prin patru *libertăți* – nu gratuități – de care beneficiază utilizatorul unui program: să îl ruleze indiferent de scop; să îl studieze și să-l adapteze propriilor nevoi (fiind permis, evident, accesul la codul-sursă); să distribuie copii ale acestuia; să-l îmbunătățească și să-l facă public. A se vedea și **Stallman, Richard**.

freeware (*freeware*). Software cu drepturi de autor înregistrate, distribuit spre a fi utilizat gratuit pentru o perioadă nelimitată de timp. Autorul poate impune restricții de utilizare: personală, academică, necomercială sau combinații între acestea. Termenul a fost introdus de Andrew Fluegelman, odată cu distribuirea programului său *PC-Talk*. A se vedea și **careware** și **shareware**.

Friendster. Una dintre cele mai populare rețele sociale *online*, lansată în 2002 (<http://www.friendster.com/>). A se vedea și **rețea socială online**.

FSD (*Free Software Definition*). Definiție a ceea ce înseamnă *Free Software*, scrisă de Richard Stallman și publicată de FSF (*Free Software Foundation*) în ediția din februarie 1986 a buletinului GNU. A se vedea și **Free Software** și **Stallman, Richard**.

FSF (*Free Software Foundation*). Fundație non-profit creată de Richard Stallman în 1985 cu scopul de a susține proiectul GNU și de a promova conceptul *Free Software*. A se vedea și **Free Software** și **Stallman, Richard**.

FTP (*File Transfer Protocol*). Protocol standard de rețea utilizat pentru schimbul și manipularea fișierelor într-o rețea IP (e.g., Internet), funcționând la nivelul transport în modelul de referință TCP/IP.

FTP este implementat folosind arhitectura client-server. În mod obișnuit, serverele FTP „ascultă” portul 21

pentru a satisface eventualele cereri de conectare venite de la clienți. O gazdă care furnizează un serviciu FTP poate oferi suplimentar acces anonim. A se vedea și **anonymous**, **client FTP**, **IP**, **server FTP**, **site FTP** și **TCP/IP**.

fullscreen (*pe tot ecranul*). În mediile bazate pe ferestre, termenul se referă la o aplicație care utilizează o fereastră ce poate ocupa întreaga suprafață a ecranului (*desktop*). A se vedea și **fereastră**.

function library (*bibliotecă de funcții*). A se vedea **bibliotecă de funcții**.

funcție (*function*). Concept utilizat în programarea funcțională, reprezintă o secvență de cod care realizează o anumită operație, putând fi apelată în mod repetat, de câte ori este necesar. Altfel spus, o funcție constă dintr-un set de instrucțiuni care realizează o sarcină specifică. În general, funcțiile pot fi predefinite sau definite de utilizator.

O funcție este apelată folosind o sintaxă similară cu `nume_f(arg1, arg2, ...)`. Se observă că apelul unei funcții constă în invocarea numelui său, urmat imediat de o listă de argumente actuale (sau parametri actuali), numărul acestora variind de la o funcție la alta (`arg1, arg2, ...`). Fiecare argument trebuie să fie o expresie validă, care poate include și alte apeluri de funcții.

O funcție predefinită este caracterizată printr-un prototip, care include tipul de dată returnat de funcție, numele funcției și lista argumentelor (inclusiv tipul de dată al fiecăruia dintre ele).

În general, pentru definirea unei funcții se folosește următoarea sintaxă:

```
function nume_f(arg1, arg2, ...)
{
    // corpul functiei
}
```

Se poate observa că definiția unei funcții include următoarele părți: numele funcției (*nume_f*), precedat de cuvântul-cheie *function*; lista de argumente formale (care poate să fie și vidă), inclusă între două paranteze rotunde (*arg1, arg2, ...*); corpul funcției, care constă din orice secvență validă de cod, conținută în mod obligatoriu între acolade, inclusiv definițiile altor funcții sau ale unor clase. Termenul este înrudit cu alți termeni ca procedură, rutină și subrutină. A se vedea și **prototip**, **procedură**, **rutină** și **subrutină**.

funcție de agregare (*aggregate function*). Funcție utilizată pentru a agrega un set de valori și a furniza, pe baza acestora, o singură valoare. A se vedea și **funcție**.

funcție principală (*main function*). Una dintre funcțiile utilizate într-un program. Reprezintă corpul principal al acestuia și punctul de început al execuției sale. Într-un program scris în limbajul C/C++, funcția principală (a cărei prezență este obligatorie) are numele *main*. A se vedea și **funcție**.

funcție recursivă (*recursive function*). Funcție care se apelează pe ea însăși, acest procedeu numindu-se

recursivitate. Dacă o funcție se autoapelează direct, se spune că se produce o *recursivitate directă*. În schimb, dacă o funcție se apelează pe ea însăși prin intermediul altei funcții, se spune că se produce o *recursivitate indirectă*. În mod obișnuit, este utilizată recursivitatea directă.

În cazul funcțiilor recursive se pune problema terminării calculelor. Apelul unei funcții este terminat doar dacă determină un număr finit de apeluri către ea însăși sau către alte funcții. Pentru aceasta, trebuie să se garanteze terminarea apelurilor recursive prin inserarea unor condiții de terminare (*e.g.*, asocierea unei variabile booleene cu funcția recursivă, apelul făcându-se până când variabila își schimbă valoarea). Iată un exemplu (scris în PHP) de funcție recursivă care calculează factorialul argumentului:

```
function factorial($n) {
    if($n == 1)
        return 1;
    else
        return $n*factorial($n-1);
}
```

Dacă se apelează funcția *factorial()* cu valoarea 3 (*i.e.*, *factorial(3)*), rezultatul returnat va fi 6. În acest caz, funcția se autoapelează de două ori. A se vedea și **funcție** și **recursivitate**.

funcție standard (*standard function*). Funcție disponibilă întotdeauna într-un anumit limbaj de programare. A se vedea și **funcție**.

funcție hash (*hash function*). Utilizate în informatică (dar nu numai), constituie o categorie de funcții neinvertibile definite pe o mulțime cu un număr mare de elemente (posibil infinită), cu valori într-o mulțime cu un

număr fix (și mai redus) de elemente. Valoarea unei funcții *hash* poartă numele de *valoare hash*, *sumă hash* sau doar *hash*. A se vedea și **funcție** și **MD5**.

G

Gates, William Henry III (n. 1955).

Cunoscut și ca Bill Gates, este un magnat, filantrop și autor american. A fondat Microsoft Corporation (împreună cu Paul Allen), aceasta devenind cea mai importantă companie de software de pe planetă (compania a fost denumită inițial Micro-Soft). A înființat împreună cu soția sa, Melinda, fundația *Bill & Melinda Gates*. Treptat, a transferat atribuțiile pe care le avea în cadrul companiei *Microsoft* către alți angajați (rămânând la sfârșitul lunii iunie 2008 *non-executive chairman*), pentru a se ocupa de administrarea fundației. În 2007, Bill și Melinda Gates au ocupat locul secund în topul celor mai generoși donatori, cu suma de 28 de miliarde de dolari.

Revista *Time* l-a desemnat ca una dintre cele mai influente 100 de personalități ale secolului al XX-lea, dar și ca unul dintre cei mai influenți 100 de oameni ai anilor 2004, 2005 și 2006. A primit numeroase distincții academice, precum și (împreună cu soția sa) recunoaștere internațională pentru actele de caritate. A se vedea și **Microsoft Corporation**.

gazdă (*host*). În rețelistică, desemnează un calculator conectat la Internet sau, în general, la orice tip de rețea. Un astfel de calculator poate găzdui diverse informații și software client sau server. Fiecare gazdă Internet (cunoscută și sub denumirea de nod Internet) are o adresă IP unică, aceasta incluzând o parte denumită adresă de rețea. A se vedea și **Internet**, **nume de gazdă** și **rețea de calculatoare**.

Gbps (sau *Gbit/s*, *Gb/s*). Prescurtare de la *gigabits per second* (gigabiți pe secundă). Reprezintă un miliard de biți pe secundă (*i.e.*, 10^9 bps). A se vedea și **bit rate**.

GD (*Graphics Draw*). Biblioteca grafică a serverului de aplicații PHP. Este utilizată pentru generarea dinamică a imaginilor în diverse formate *raster*. A se vedea și **formate raster** și **PHP**.

generator de rapoarte (*report generator*). Aplicație inclusă în mod obișnuit într-un sistem de gestionare a bazelor de date, care generează rapoarte folosind conținutul acestora. Un raport poate include toate câmpurile unui tabel sau ale unei baze de date sau numai o parte dintre acestea. De asemenea, raportul poate conține toate înregistrările sau numai o parte dintre ele (*i.e.*, cele care îndeplinesc anumite condiții, specificate de utilizator). În plus, raportului îi pot fi adăugate alte elemente: titluri, numere de pagini, elemente grafice etc. A se vedea și **bază de date** și **raport**.

generația Internet (*Internet generation*). Numită și *generația Z*, urmează generației Y, incluzând persoanele născute începând cu anul 1990. O serie de alte denumiri au fost utilizate pentru a se face referire la acest grup de populație: generația V (V de la virtual), generația C (C de la community, content, computer, click, creative, celebrity etc.), generația Google, noua generație tăcută etc. De multe ori, termenii „generația C” sau „generația Internet” se referă atât la generația Y, cât și la generația Z. Membrii generației Internet au în comun în primul rând utilizarea tehnologiilor de comunicare și a celor media (e.g., telefoane mobile, *iPod*-uri, Web-ul, mesageria instantanee, DVD-uri etc.).

Cu toate că membrii generației Internet își canalizează atenția pentru intervale mai mici de timp și au o abilitate mai redusă de a reflecta asupra unui subiect, acestea sunt înlocuite de însușiri vizuale dezvoltate, de abilitatea de a se concentra simultan asupra mai multor medii și de abilitatea de a monitoriza schimbările și de a face descoperiri prin inducție. După părerea lui Marc Prensky, „cu toate că aceste însușiri cognitive pot să nu fie noi, combinarea și intensitatea lor sunt noi. Avem o nouă generație, cu o combinație de însușiri cognitive foarte diferită de cea a predecesorilor – *copiii născuți în epoca digitală (Digital Natives)*.” A se vedea și **generația X** și **generația Y**.

generația X (*generation X*). Include persoanele născute între anii 1965, și

1980. Aceste persoane sunt ambițioase, optimiste, caută permanent echilibrul, apreciind foarte mult valorile tradiționale și siguranța familială. A se vedea și **generația Internet** și **generația X**.

generația Y (*generation Y*). Include persoanele născute între anii 1980 și 1990, urmând generației X. Aceste persoane împărtășesc unele dintre valorile generației X, dar se deosebesc de aceasta prin faptul că investesc mai mult în educație. De asemenea, Internetul reprezintă pentru generația Y o modalitate importantă de comunicare, alături de telefoanele mobile. A se vedea și **generația Internet** și **generația X**.

geo-mashup. Aplicație *mashup* care are în componența sa un serviciu geografic (i.e., de cartografie). A se vedea și **mashup**.

GeoRSS (*Geographically Encoded Objects for RSS feeds*). Modul de extensie RSS utilizat pentru codificarea zonelor geografice în *feed*-urile RSS 1.0, RSS 2.0 și Atom, dar poate fi folosit și în alte codificări XML (<http://georss.org/>). Promovat de Yahoo! și Microsoft, folosește ubicuitatea RSS și schema GML pentru a crea cea mai bună metodă de geodistribuire.

Este oferit în două variante: *GeoRSS-Simple* și *GeoRSS-GML*. *GeoRSS-Simple* este un format simplu care le permite dezvoltatorilor și utilizatorilor să-l folosească rapid și fără dificultăți, acoperind cazurile tipice de codificare a locației. În schimb, varianta

GeoRSS-GML oferă un număr sporit de facilități. A se vedea și **feed**, **GML**, **RSS** și **XML**.

ghost site (*site-fantomă*). Site care nu a mai fost actualizat de mult timp. Site-urile de acest tip nu sunt site-uri abandonate! Ele includ explicații în care se precizează că nu au mai fost actualizate de mult timp, dar nu au fost șterse deoarece conțin informații importante sau cu valoare istorică. A se vedea și **site Web**.

GIF (*Graphic Interchange Format*). Format *raster* dezvoltat în anii 1980 de CompuServe. Este folosit datorită dimensiunii relativ mici a fișierelor utilizate, fiind unul dintre cele mai populare formate grafice în Internet. La începutul anilor 1990, dezvoltatorii Web-ului au adoptat formatul GIF pentru includerea conținutului grafic în paginile Web, datorită eficienței și răspândirii sale.

GIF este un format fără pierderi. Aceasta înseamnă că se poate salva de oricâte ori se dorește o imagine în acest format, după modificări succesive, fără a pierde informații, adică fără scăderea calității imaginii respective.

Formatul GIF utilizează pentru comprimare algoritmul LZW (*Lempel-Ziv Welch*), acesta fiind o variantă a algoritmului *Huffman*. Schema LZW este eficientă pentru comprimarea imaginilor care utilizează un număr mic de culori și includ zone omogene de culoare mari, fiind mai puțin eficientă

dacă imaginile utilizează multe culori și conțin texturi complexe.

Formatul GIF are două versiuni: GIF87a (format non-transparent) și GIF89a (format transparent). Extensia pentru fișierele în care sunt salvate imaginile în acest format este *gif*. Tipul MIME pentru GIF este *image/gif*.

Fiecare fișier cu extensia *gif* are propria sa paletă de culori, al căror număr nu poate depăși 256 (adâncime de culoare de 8 biți). Imaginile GIF89a pot include zone transparente (transparența este pe un bit, ceea ce înseamnă că într-o imagine poate exista o singură culoare transparentă, aceasta fiind, de regulă, culoarea fundalului). Se poate utiliza un software de editare a imaginilor (*e.g.*, *Photoshop*, dar și multe aplicații *shareware*) pentru selectarea culorii care va deveni transparentă din paleta de culori a unei imagini GIF.

Se pot realiza animații GIF (la nivel elementar, fiecare cadru fiind stocat separat ca o imagine), formatul reprezentând singura soluție ieftină pentru crearea animațiilor *online*, cu excepția limbajului SVG. Realizarea animațiilor GIF necesită transparența fundalului cadrelor utilizate.

Formatul GIF suportă întreteserea, utilă mai ales pentru imagini cu dimensiuni mari (*e.g.*, ilustrațiile) și mai puțin utilă pentru cele cu dimensiuni mici (*e.g.*, butoane).

În ceea ce privește posibilitățile de utilizare, formatul grafic GIF este destinat realizării unor imagini care includ un număr mic de culori (*e.g.*,

mici pictograme pentru navigare sau diagrame simple, *logo-uri*, imagini clip-art, grafice în care există zone mari monocrome sau imagini care conțin text). De asemenea, cele mai multe *bannere* animate sunt realizate în format GIF. Problemele formatului GIF sunt determinate de faptul că utilizează o paletă de 256 de culori, ceea ce înseamnă că nu este capabil să evidențieze variațiile fine de culoare ale imaginilor. A se vedea și **algoritmul LZW** și **formate raster**.

Gigabit Ethernet. Termen utilizat pentru a desemna tehnologii Ethernet care asigură o rată de transfer a datelor în rețelele de calculatoare de 1 Gbps (*i.e.*, 1000 Mbps). Astăzi sunt folosite și tehnologii cu rate de transfer superioare (10-gigabit Ethernet și 100-gigabit Ethernet). A se vedea și **Ethernet** și **Fast Ethernet**.

gigabyte (*gigaoctet*). Notat GB, are 2^{30} (sau 1024^3) *bytes* (sau *octeți*), adică 1.073.741.824 *bytes*. A se vedea și **byte**.

GIMP (*GNU Image Manipulation Program*). Editor de grafică *raster* gratuit (<http://www.gimp.org/>). GIMP este utilizat mai ales pentru tăierea, redimensionarea și combinarea imaginilor. De asemenea, este folosit pentru conversia între diverse formate *raster*, precum și pentru crearea de animații GIF simple. A se vedea și **formate raster**.

GIS (*Geographic Information System, sistem informațional geografic*). Sistem software de captare, stocare, analiză,

prelucrare și gestionare a datelor și atributelor asociate care se referă la Pământ (date geografice); *e.g.*, *Google Maps*. Un astfel de sistem dispune de hărți și imagini obținute prin intermediul sateliților, permițând adăugarea datelor proprii deținute de utilizator, alături de localizarea lor, pentru a crea reprezentări personalizate ale acestora. A se vedea și **Google Maps**.

Gliffy. Aplicație Web 2.0 utilizată pentru crearea de diagrame UML (*Unified Modelling Language*), diagrame de rețea, diagrame SWOT, desene tehnice, organigrame etc. *Gliffy* face parte din categoria de aplicații *Mind Mapping*.

Facilitățile oferite de aplicația *Gliffy* sunt următoarele: publicare rapidă a diagramelor; import în documente; colaborare instantanee; independență de platformă; urmărirea schimbărilor.

Utilizatorul are posibilitatea de a folosi în diagramele realizate imagini pre-existente (*My Images*). Acestea vor fi încărcate, în prealabil, pe serverul aplicației *Gliffy*. Ulterior, pot fi utilizate în orice diagramă pe care o va crea. La încărcare, utilizatorul va asocia etichete imaginilor. Unei imagini îi pot fi asociate una sau mai multe etichete (în ultimul caz, acestea vor fi separate prin virgulă). Etichetele vor fi folosite ulterior pentru căutarea imaginilor. De asemenea, în diagramele create de utilizator pot fi folosite imagini găsite în Web prin intermediul modulului de

căutare inclus în *Gliffy* (creat pe baza unui API oferit de Yahoo!).

Înainte de utilizarea aplicația *Gliffy*, este necesară crearea unui cont folosind o adresă de *e-mail*. În varianta gratuită a aplicației, este posibil să se stocheze pe serverul acesteia cel mult cinci diagrame publice (*i.e.*, care pot fi vizualizate și de alte persoane). A se vedea și **Mind Mapping** și **Web 2.0**.

GML (*Geography Markup Language*). Formate de date bazat pe XML definit de OGC (*Open Geospatial Consortium*) pentru a exprima caracteristici geografice (*e.g.*, coordonatele unui punct). Servește ca limbaj de modelare pentru sistemele de tip GIS și ca format deschis de schimb în tranzacțiile cu date geografice în Internet. Este utilizat în aplicațiile *geo-mashup*. A se vedea și **geo-mashup** și **XML**.

GNOME. Mediu *desktop (desktop environment) open source* pentru sisteme compatibile UNIX, cu licență GPL (<http://www.gnome.org/>). Este mediul de lucru standard pentru numeroase distribuții Linux și pentru Solaris, disponibil și în limba română. Proiectul GNOME a început să fie dezvoltat în anul 2007 de către Miguel de Icaza. A se vedea și **KDE** și **Licența Publică Generală GNU**.

GNU GPL (*GNU General Public License*). A se vedea **Licența Publică Generală GNU**.

Goodfellow, James (n. 1937). Inventator scoțian, a patentat tehnologia PIN

(*Personal Identification Number*). Este recunoscut și ca inventatorul ATM-ului (*Automatic Teller Machine*); în dezvoltarea acestuia au avut un rol deosebit de important și John Shepherd-Barron. A se vedea și **ATM** și **PIN**.

GoGuides. Director Web în care site-urile înscrise sunt clasificate în categorii și în subcategorii (<http://www.goguides.org/>). Oferă și un serviciu de căutare. A se vedea și **director Web**.

Google, Inc. Corporație publică americană care oferă numeroase servicii *online*, dintre care pot fi amintite cele de căutare, *e-mail*, *social networking*, publicitate, partajare video, hărți și *Office*. Google oferă produse și servicii atât pentru publicul larg, cât și pentru mediul de afaceri. Compania a fost fondată în septembrie 1998 de Larry Page și Sergey Brin, doi doctoranzi – în acea vreme – la Stanford University.

Google este cunoscută mai ales pentru motorul de căutare cu același nume (<http://www.google.com/>). Iată și alte servicii și aplicații populare ale companiei: *AdWords* (publicitate *online*), *YouTube* (partajare video), *Blogger* (*blogging*), *Gmail* (*e-mail*), *Google Docs* (*Office online*), *Google Earth* (planeta Pământ virtuală), *Google Chrome* (browser Web), *Google Reader* (agregator RSS *online*), *Google Translate* (translator Web).

Google bombing. Metodă de creare a unui număr mare de legături (*backlinks*) către o pagină Web, bazată pe utilizarea în textul acestora a unei expresii-cheie; are ca efect regăsirea paginii respective în rezultatele returnate de motoarele de căutare. Cel mai cunoscut exemplu se referă la utilizarea expresiei „miserable failure”.

Google Code Search. Serviciu oferit de Google, lansat în octombrie 2006, care le dă utilizatorilor posibilitatea de a căuta un cod-sursă în Internet (<http://www.google.com/codesearch>). În căutare pot fi folosiți operatorii `lang:`, `package:`, `license:` și `file:`. A se vedea și **Google, Inc.**

Google dance (*Google dance*). Actualizarea bazei de date utilizate de motorul de căutare *Google*. Este realizată lunar, de obicei între a 20-a și a 28-a zi. A se vedea și **Google, Inc.**

Google Docs. Cel mai popular pachet de aplicații *Office 2.0*, oferit gratuit de Google, care include *Google Document*, *Google Spreadsheet* și *Google Presentation*. Pachetul utilizează formatul deschis ODF (*OpenDocument Format*). Conform politicii companiei Google, deținerea unui cont de *e-mail Gmail* este suficientă pentru utilizarea pachetului, precum și a aplicației *Google Calendar*. A se vedea și **Office 2.0**, **OpenDocument Format** și **Web 2.0**.

Google juice (*Google juice*). Valoarea și avantajul dobândite de un site odată cu poziționarea sa în primele rezultate

returnate de motorul de căutare *Google*. A se vedea și **link juice** și **SEO**.

Google Labs. Site demonstrativ pentru proiectele companiei Google aflate în dezvoltare, o modalitate de obținere a *feedback*-ului de la utilizatori înaintea lansării versiunilor finale ale acestora (<http://labs.google.com/>). A se vedea și **Google, Inc.**

Google Maps. Serviciu care pune la dispoziția utilizatorilor hărți terestre prin intermediul unei interfețe-utilizator prietenoase, încărcarea informației realizându-se în manieră asincronă (AJAX) (<http://maps.google.com/>). Face parte din categoria aplicațiilor GIS. A se vedea **AJAX** și **GIS**.

Google Maps API. API pus la dispoziție de serviciul *Google Maps*. Face posibilă includerea unor hărți personalizate în cadrul paginilor Web, folosind un set de obiecte și metode JavaScript care pot fi utilizate de orice persoană înregistrată. API-ul oferă posibilitatea realizării de *geo-mashups*.

Datele adăugate în cadrul unei aplicații *geo-mashup* realizate folosind API-ul *Google Maps* pot fi specificate în diverse formate: XML, CSV, JSON și fișiere text. De asemenea, există mai multe tipuri de formate de date care pot fi utilizate pentru crearea sau găsirea unui conținut cu relevanță geografică: GML, KML, GPX și GeoRSS. A se vedea și **geo-mashup**, **Google Maps**, **mashup** și **API**.

Google News. Agregator automat de știri furnizat de Google, cu versiuni în multe limbi (<http://news.google.com/>). Oferă și un serviciu de căutare.

Google Reader. Unul dintre cele mai populare agregatoare bazate pe Web, creat de compania Google (<http://www.google.com/reader>), folosit pentru abonarea la *feed*-urile RSS și Atom și pentru gestionarea acestora.

Versiunea beta a produsului *Google Reader* a fost lansată în octombrie 2005 prin intermediul site-ului *Google Labs*. În septembrie 2007, a fost eliminat statutul de versiune beta.

Pentru a putea utiliza *Google Reader*, este necesar ca utilizatorul să dețină un cont de *e-mail* pe *Google Mail* (cunoscut și ca *GMail*, serviciul poate fi accesat la <http://gmail.google.com>). Autentificarea acestuia în *Google Reader* se va face folosind numele de utilizator și parola *Gmail*.

Dacă utilizatorul a folosit un anumit agregator și, la un moment dat, se hotărăște să utilizeze *Google Reader*, are posibilitatea de a importa subscripțiile din prima aplicație (*i.e.*, *feed*-urile la care este abonat). În acest scop, este necesar ca, în prealabil, utilizatorul să le exporte din vechiul agregator în format OPML, fișierul obținut fiind apoi importat în *Google Reader*. De asemenea, agregatorul oferit de compania Google poate exporta subscripțiile existente sub forma unui fișier OPML. A se vedea și **feed** și **OPML**.

Google Search. Cel mai popular motor de căutare, dezvoltat de compania Google, creat inițial de Larry Page și Sergey Brin (cofondatori) în 1997 (<http://www.google.com/>). Algoritmul de căutare Google utilizează un sistem patentat denumit *PageRank*, folosit pentru ordonarea rezultatelor care se potrivesc cu un șir de căutare dat. A se vedea și **Bing**, **GYM**, **PageRank** și **Yahoo! Search**.

Google Translate. Serviciu de traducere în/din peste 40 de limbi oferit de Google (<http://translate.google.com/>). Pot fi traduse texte introduse de utilizatori, pagini Web ale căror URL-uri trebuie specificate sau documente încărcate de utilizatori.

Gopher sau **gopher**. Protocol și instrument creat în 1991 la University of Minnesota de către o echipă de dezvoltatori; utilizat pentru distribuirea, căutarea și regăsirea documentelor în Internet. Structura arborescentă (ierarhică) a meniurilor este esențială, ajutându-i pe utilizatori în regăsirea documentelor. Acestea pot fi descărcate și afișate. Un client *Gopher* are acces la toate serverele *Gopher* disponibile.

Serverele *Gopher* au devenit atât de populare, încât până în 1993 a apărut un număr foarte mare de servere conținând peste un milion de documente. Pentru a găsi un astfel de document a fost creat un instrument de căutare specializat, numit *Veronica* (*very easy rodent-oriented netwide index to computerized archives*). A se vedea și **Veronica**.

Gopherspace (*spațiu Gopher*). Totalitatea informațiilor oferite în Internet pe serverele *Gopher*. A se vedea și **Gopher**.

Gosling, James (n. 1955). Dezvoltator software canadian, cunoscut în special pentru crearea limbajului de programare Java (1991), inclus ulterior (1995) în platforma Java (<http://blogs.sun.com/jag/>). A se vedea și **Java**.

GPRS (*General packet radio service*). Serviciu mobil de date orientat pe pachete și metodă de transfer utilizată în sistemele 2G și 3G, permite ca informația să fie trimisă și primită de-a lungul unei rețele de telefonie mobilă. A fost dezvoltat ca un răspuns GSM la tehnologiile CDPD (*Cellular Digital Packet Data*) și *i-Mode*.

Vitezele maxime teoretice GPRS de până la 171,2 kbps sunt obținute utilizând simultan opt căi de acces, ceea ce face ca viteza oferită de acest serviciu să fie de trei ori mai mare decât cea posibilă în rețelele de comunicații de astăzi și de zece ori mai mare decât cea a serviciilor de transfer de date rapid (CSD, *Circuit Switched Data*) în rețelele GSM.

graf (*graph*). Structură de date formată dintr-o pereche ordonată de mulțimi notată $G=(X,\Gamma)$, unde X este o mulțime finită și nevidă de elemente numite *noduri* sau *vârfuri* și Γ o mulțime de perechi (ordonate sau neordonate) de elemente din X , numite *muchii* (dacă sunt perechi neordonate) sau *arce* (dacă

sunt perechi ordonate). În primul caz, graful se numește *neorientat*, iar în al doilea caz *orientat*. Dacă $\Gamma=\emptyset$ (mulțimea vidă), graful G se numește *graf nul* și reprezentarea lui în plan se reduce la noduri izolate.

Numărul de noduri ale grafului se numește *ordinul grafului*. Dacă ordinul este 0 sau 1, graful se numește *trivial*. Două noduri se numesc *adiacente* dacă există o muchie care le unește. Două muchii sunt adiacente dacă există un nod care să fie incident cu ambele muchii. Numărul de muchii conectate la un nod se numește *gradul nodului*. Suma gradelor nodurilor unui graf este egală cu dublul numărului muchiilor acestuia.

Sucesiunea de muchii adiacente și distincte care conectează două vârfuri ale unui graf se numește *drum* în graf, cele două noduri fiind *capetele* (sau *extremitățile*) drumului. Un drum este *simplic* dacă muchiile care îl compun sunt distincte. Se numește *ciclu* (sau *circuit*) un drum care are drept capete același nod. Un *ciclu* se numește *hamiltonian* dacă este simplic și trece prin toate nodurile grafului câte o singură dată. Altfel spus, un ciclu elementar care trece prin toate nodurile grafului se numește *circuit hamiltonian*. Un ciclu se numește *eulerian* dacă trece prin toate nodurile câte o singură dată. Se spune că SG este un *subgraf* al grafului G dacă acesta conține o parte din nodurile lui G și numai acele muchii care le conectează. Un subgraf SG al unui graf G se obține din G prin

suprimarea anumitor noduri și a tuturor muchiilor adiacente cu acestea. Se spune că un graf este *conex* dacă între oricare două noduri ale acestuia există cel puțin un drum. Graful trivial este considerat conex. A se vedea și **structură de date**.

GraphicConverter. Aplicație shareware utilizată pentru vizualizarea, editarea și organizarea fișierelor grafice, precum și pentru conversia formatelor, creată de compania Lemke Software GmbH și având versiuni pentru Mac OS X și Mac OS Classic (<http://www.lemkesoft.com/>). Aplicația poate roti și redimensiona imaginile. De asemenea, le poate aplica diverse efecte și filtre. *GraphicConverter* poate importa aproximativ 200 de formate de fișier și exportă circa 80.

greedy (*lacom*). Una dintre cele mai directe tehnici de proiectare a algoritmilor, care se aplică la o varietate largă de probleme. În general, această tehnică se aplică problemelor de optimizare. Specificul său constă în aceea că soluția optimă se construiește pas cu pas, la fiecare pas fiind selectat (sau „înghițit”) în soluție elementul care pare cel mai bun la momentul respectiv, în speranța că va duce la o soluție optimă globală.

Fiind dată o mulțime A cu n elemente, se cere să se determine o submulțime a sa, fie aceasta B, care satisface anumite restricții (soluție posibilă). În plus, se cere să se determine o soluție posibilă (numită soluție optimă) care minimizează sau maximizează o anumită

funcție *obiectiv* impusă. Aplicarea tehnicii *greedy* la problema de optimizare respectivă presupune parcurgerea pașilor următori: (1) mulțimea B este inițial vidă; (2) se alege un element din mulțimea A care ar putea reprezenta soluția optimă la pasul i; (3) se verifică dacă elementul ales poate fi adăugat la mulțimea soluțiilor, fiind adăugat în caz afirmativ; (4) procedeul continuă, repetitiv, până când au fost determinate toate elementele din mulțimea soluțiilor. Algoritmul se termină fie când a fost găsită soluția cerută, fie când s-a constatat inexistența acesteia.

În general, numărul de operații de bază efectuate de un algoritm *greedy* este o expresie polinomială (*i.e.*, algoritmul este performant). Trebuie remarcat că, fiind dată o problemă, nu există întotdeauna un algoritm *greedy* care găsește soluția optimă. Mai mult, pentru majoritatea problemelor nu se cunosc algoritmi *greedy*. A se vedea și **programare dinamică**.

grep (*global regular expression print*). Comandă UNIX/Linux care caută fișiere sau intrări standard globale pentru liniile corespunzătoare unei expresii regulate date și le afișează la intrarea standard.

grup de utilizatori (*user group*). Grup de persoane interesate de utilizarea unei anumite tehnologii sau aplicații. În mod obișnuit, acesta oferă un forum în care membrii grupului dezbat probleme de interes comun și fac schimb de idei și informații.

GPX (*Global Positioning System eXchange Format*). Format bazat pe XML, utilizat pentru schimbul datelor GPS între aplicații și serviciile Web.

GUI (*Graphical User Interface*). A se vedea **interfață grafică**.

GYM. Acronim pentru Google-Yahoo!-Microsoft, cei trei „mari” ai motoarelor de căutare (*Google Search* – 60%, *Yahoo! Search* – 10% și *MSN Search* – 16%). A se vedea și

Bing, Google Search și Yahoo! Search.

gzip. Acronim pentru **GNU zip**, un utilitar UNIX folosit pentru compresia fișierelor, dezvoltat de Jean-Loup Gailly și Mark Adler în cadrul proiectului GNU. Versiunea 1.0 a fost lansată în februarie 1993. Programul *gzip* este bazat pe algoritmul DEFLATE, o combinație LZ77 și Huffman. A se vedea și (a) **comprima și proiectul GNU**.

H

hacker (*hacker*). Persoană care deține cunoștințe foarte avansate în domeniul tehnicii de calcul și al programării, reușind să scrie sau să înțeleagă fără un efort aparent programe complexe, precum și să proiecteze, să dezvolte sau să îmbunătățească diverse tehnologii informatice. A se vedea și **cracker** și **haxor**.

Hamming, Richard Wesley (1915–1998). Matematician și informatician american, a adus contribuții majore în informatică, teoria informației și telecomunicații (e.g., coduri Hamming, distanța Hamming). A fost fondator și președinte al ACM (*Association for Computing Machinery*). În 1968 a obținut premiul Turing. A se vedea și **ACM**, **cod Hamming** și **distanța Hamming**.

handler (*handler*). În programare, secvență de cod (e.g., rutină) care reacționează la un eveniment sau care controlează accesul la anumite zone de memorie.

handler de întrerupere (*interrupt handler*). Rutină care se execută la apariția unei întreruperi. Poate avea

diverse funcții, care depind de natura întreruperii. A se vedea și **întrerupere**.

hardware (*hardware*). Componentele fizice ale unui sistem de calcul, inclusiv perifericele. Acestea pot fi incluse în următoarele categorii: placa de bază (părțile atașate direct acesteia sunt CPU, RAM, BIOS, magistralele interne, controlerele pentru magistrala externă), sursa de putere (convertește curentul alternativ în curent continuu, furnizând diverse tensiuni pentru diverse părți ale calculatorului), unitatea de afișare video, dispozitive de stocare detașabile (e.g., CD-ROM Drive, CD Writer, DVD-ROM Drive, DVD Writer, BD-ROM Drive, BD Writer, USB Flash Drive), dispozitive interne de stocare (e.g., hard-disk, RAID array controller), dispozitive de intrare (e.g., tastatură, mouse, trackball, joystick, scanner, webcam, microfon) și dispozitive de ieșire (e.g., monitor, imprimantă). A se vedea și **software**.

hard-disk (HD, *disc dur*). Dispozitiv magnetic de stocare compus din mai multe discuri (sau platane), fiecare fiind prevăzut cu un cap propriu de citire/scriere pentru fiecare față. La efectuarea operațiunii de citire/scriere, capetele funcționează aproape simultan, asigurând o viteză ridicată de transfer a informației. Discurile au același număr de piste, toate pistele cu același număr de ordine formând un cilindru. Numărul de cilindri este egal cu numărul de piste conținute pe o față a discului. Înscrierea unui fișier pe același cilindru conduce la economisire

de timp, deoarece nu mai este necesară deplasarea repetată a capului de citire/scriere. În general, HD-urile stochează sistemul de operare, aplicațiile și fișierele de date. Există două categorii de discuri dure: pack discuri și discurile hard (discurile Winchester).

harta site-ului (*site map* sau *sitemap*). Listă structurată a paginilor Web ale unui site cu legături către fiecare dintre acestea. Este utilă atât utilizatorilor, cât și roboților motoarelor de căutare și foarte importantă pentru indexarea corectă a paginilor site-ului.

În 2005, Google a lansat *Sitemap*, proiect menit să vină în sprijinul dezvoltatorilor Web, astfel încât aceștia să aibă posibilitatea de a publica în manieră standardizată o listă a legăturilor incluse în site-urile lor. În noiembrie 2006, Google, MSN (*Microsoft Network*) și Yahoo! au anunțat sprijinul lor pentru protocolul *Sitemap*. Formatul *Sitemap* constă dintr-o serie de marcaje XML care vor fi folosite pentru crearea fișierelor *Sitemap*.

Un fișier *Sitemap*, cu extensia *.xml*, trebuie să fie codificat UTF-8, nu poate să fie mai mare de 10 MB și nu trebuie să conțină mai mult de 50 000 de legături, dar poate fi comprimat *gzip*. Fișierele *Sitemap* pot fi trimise direct motoarelor de căutare (*pinged*). De asemenea, locația acestora poate fi precizată în fișierele *robots.txt* adăugând linia *Sitemap*: `<locatie_sitemap>`, unde `locatie_sitemap` este URL-ul

fișierului *Sitemap* (de exemplu, <http://www.exemplu.ro/sitemap.xml>).

hartă de imagini (*image map*). Denumită și *hartă pentru click* (*clickable map*), este o imagine inclusă într-o pagină Web care conține o serie de hiperlegături către alte pagini. Pentru aceasta, se adaugă marcajului `` următoarele atribute: `src` (are ca valoare URL-ul fișierului imagine), `usemap` (are ca valoare o expresie de genul `#nume`, unde `nume` este numele hărții; acesta va fi utilizat în interiorul etichetei `<map>`). Definierea hărții se realizează folosind marcajele `<map>` și `</map>` și atributul obligatoriu `name`, cu aceeași valoare ca și `usemap`. Zona sau zonele conținute de hartă sunt introduse folosind eticheta `<area>` cu atributele obligatorii: `shape` (precizează forma ariei declarate, cu valorile: `rect`, `circle`, `poly`, `default`), `coords` (precizează coordonatele zonei), `href` (indică URL-ul paginii-destinație a legăturii). A se vedea și **hiperlegătură**.

Haskell. Limbaj avansat de programare funcțională, succesori al limbajelor LISP și Scheme, bazat pe calculul lambda și creat de Curry Haskell (<http://www.haskell.org/>). Utilizarea acestui limbaj permite creșterea productivității programatorilor, scrierea unui cod curat și cu dimensiuni reduse, mărirea fiabilității și reducerea numărului de erori. A se vedea și **calculul lambda**, **Haskell**, **Brooks Curry** și **programare funcțională**.

Haskell, Brooks Curry (1900–1982). Matematician și logician american. Este cunoscut mai ales pentru inventarea a două limbaje de programare funcțională, Haskell și Curry (al doilea fiind experimental și bazat pe primul). A se vedea și **Haskell și programare funcțională**.

haxor (*haxor*). Cracker amator, fără cunoștințe de programare. Utilizează în activitatea sa ilegală programe realizate de alții. Sunt desconsiderați atât de crackerii cu experiență, cât și de hackeri. A se vedea și **cracker și hacker**.

HCI (*Human Computer Interaction*). Disciplină a științei și tehnologiei informației având ca obiect de studiu proiectarea, evaluarea și implementarea sistemelor interactive în scopul utilizării lor de către om, precum și fenomenele majore implicate.

Scopul de bază al HCI este de a îmbunătăți interacțiunea dintre utilizatori și calculatoare și de a facilita, astfel, producerea unor sisteme utilizabile, funcționale și sigure. Pentru aceasta este necesar ca dezvoltatorii să înțeleagă factorii care determină modul în care oamenii utilizează tehnologia și să dezvolte instrumente și tehnici care să permită crearea unor sisteme corespunzătoare. Astfel, preocupările HCI sunt legate de: dezvoltarea unor metodologii și procese utilizate pentru proiectarea interfețelor; crearea unor metode pentru implementarea acestora; crearea unor tehnici pentru evaluarea și compararea interfețelor; dezvoltarea unor noi tipuri de interfețe și tehnici de interacțiune

om-calculator; crearea unor modele și teorii descriptive și predictive referitoare la aceasta.

HCI are un caracter multidisciplinar, situându-se la zona de contact dintre numeroase domenii de cercetare cu preocupări legate de dezvoltarea calculatoarelor, dintre care pot fi amintite: tehnologia informației – preocupată de proiectarea și implementarea interfețelor cu utilizatorul; sociologia și antropologia – interesate de interacțiunea dintre tehnologii și de modalitățile prin care sistemele umane și tehnic se adaptează unul la celălalt; psihologia cognitivă – interesată de procesele cognitive umane și de comportamentul utilizatorilor; lingvistica, interesată atât de dezvoltarea limbajului uman și de cel al mașinilor, cât și de relația dintre acestea.

Cercetările actuale în domeniul HCI urmăresc dezvoltarea unor noi metodologii de proiectare a interfețelor, experimentând dispozitive hardware moderne, creând prototipuri ale unor sisteme software noi, explorând noi paradigme ale interacțiunii om-computer și dezvoltând modele și teorii ale acesteia.

Metodologiile de proiectare HCI au în centrul lor utilizatorul. Astfel, proiectarea centrată pe utilizator (UCD, *user-centered design*), paradigmă folosită astăzi la scară largă, impune ca proiectanții și dezvoltatorii să lucreze împreună, definind nevoile și limitele utilizatorului final și creând – în funcție de acestea – sisteme

corespunzătoare. În cadrul UCD, se evidențiază proiectarea sub conducerea/cu participarea utilizatorului (USD, *user-driven design*/PD, *participatory design*), paradigmă în care unii dintre utilizatori devin chiar membri ai echipei de proiectare.

În HIC sunt implicați doi agenți: omul (utilizatorul) și calculatorul. Deși aceștia sunt fizic separați, ei sunt capabili să schimbe informații prin intermediul unor canale de comunicație.

Din perspectiva omului, în schimbul de informații sunt implicate două procese de bază: percepția și controlul. Primul proces se realizează prin intermediul unor canale de intrare umane (HIC, *Human Input Channels*), iar al doilea folosește canale de ieșire umane (HOC, *Human Output Channels*). În ceea ce privește calculatorul, percepției îi corespund modalități (canale) de intrare (CIM, *Computer Input Modalities*), iar controlului medii/dispozitive (canale) de ieșire (CMO, *Computer Media Output*).

Canalele de intrare (sau de percepție) umane HIC sunt vizual, auditiv și tactil. Dispozitivele de ieșire CMO sunt utilizate pentru a-i furniza utilizatorului uman informații, fiind în directă corespondență cu HIC. Astfel, există dispozitive pentru ieșirea vizuală (e.g., monitor, proiector, imprimantă, plotter), dispozitive pentru ieșirea acustică (e.g., boxe, căști) și dispozitive pentru ieșirea tactilă (e.g., display Braille).

Canalele de ieșire (sau de control) umane HOC sunt vorbirea, mișcățile

faciale, scrierea de mână, gestică și manipularea obiectelor. Corespunzător HOC, dispozitivele de intrare CIM (utilizate de om pentru a furniza informații calculatorului) sunt: dispozitive de indicare (e.g., *mouse/trackball/joystick*, *touch screen*, tabletă grafică), tastatura, microfonul, aparatul foto și camera digitală, precum și diverși senzori.

heap (*grămadă*). Arbore binar complet cu următoarea proprietate (numită *proprietate de heap*): valoarea fiecărui vârf este mai mare sau egală cu valoarea fiecărui copil al său. Cea mai importantă caracteristică a acestei structuri de date este aceea că modificarea valorii unui vârf se va face foarte eficient, păstrându-se proprietatea de heap. A se vedea și **arbore binar și structură de date**.

Hellman, Martin Edward (n. 1945). Criptograf și matematician american. Este cunoscut pentru dezvoltarea (împreună cu Withfield Diffie) a *schimbului de chei Diffie-Hellman*, o tehnică de distribuție a cheilor în sistemele criptografice cu cheie publică. A se vedea și **criptografia cu cheie publică și Diffie, Bailey Withfield**.

Hewlett-Packard. Cunoscută ca HP, este o companie americană specializată în produse hardware (PC-uri *desktop*, *laptop* și *notebook*; servere; stații de lucru; imprimante; scannere; aparate foto digitale; PDA-uri; calculatoare de buzunar) și software (<http://www.hp.com/>).

Compania a fost fondată în 1939 de William Hewlett și David Packard, absolvenți în domeniul ingineriei electrice la Stanford University. În anul 2002, HP a cumpărat compania Compaq, producător cunoscut de PC-uri *desktop* și *laptop* (<http://www.compaq.com/>).

Hewlett, William Reddington (1913–2001). Inginer american și cofondator în 1939, alături de David Packard, al companiei Hewlett-Packard (HP). În 1966 a fondat împreună cu soția sa fundația *William and Flora Hewlett*. A se vedea și **Hewlett-Packard** și **Packard, David**.

Hi5. Cea mai utilizată rețea socială *online* în România (<http://hi5.com/>). A se vedea și **rețea socială online**.

hiperlegătură (*hyperlink*). Denumită și *link* (*legătură*), este un element de navigare inclus într-o pagină Web care conduce – atunci când utilizatorul execută un click pe el – către alte elemente din pagina respectivă sau dintr-o altă pagină. Este important de reținut că, deși sunt elemente de bază ale Internetului și Web-ului, utilizarea hiperlegăturilor nu se limitează la acestea, fiind folosite practic în orice alt document electronic (*e.g.*, *Office*, PDF).

În anul 1945, Vannevar Bush scrie în articolul *As We May Think*, publicat în *Atlantic Monthly*, despre un dispozitiv foto-electro-mecanic imaginar numit *Memex*, care poate realiza legături între documente. Începând cu anul 1962, Douglas Engelbart lucrează la un

prototip pentru *oN-Line System* (NLS), care utilizează legăturile pentru parcurgerea documentelor.

Pentru crearea unei legături în limbajul (X)HTML se utilizează elementul `a` și atributele sale: `href` (destinația legăturii, precizată printr-un URL), `title` (titlul legăturii), `class` (clasa CSS), `target` (comportamentul legăturii). În plus, se folosește textul-ancoră (*anchor text*), vizualizat de utilizator. Obligatoriu, trebuie precizate destinația legăturii și textul-ancoră. O legătură pornește de la o ancoră și îl trimite pe utilizator la destinația specificată. Iată un exemplu:

```
<a  
href="http://www.exemplu.ro/"  
title="exemplu"  
target="_blank"  
class="clasa">
```

Text ancora

A se vedea și **CSS**, **legătură**, **text-ancoră**, și **URL**.

hipermedia (*hypermedia*). Termen utilizat ca o extensie a celui de hipertext, în care elementele multimedia, împreună cu textul și hiperlegăturile, creează un mediu neliniar. Acest mediu este diferit de cel liniar, creat de elementele multimedia (*e.g.*, video). Web-ul este un exemplu clasic de sistem hipermedia.

În 1975, la Carnegie-Mellon, debutează primul sistem hipermedia distribuit, numit inițial ZOG și ulterior KMS. Primul videodisc hipermedia a fost *Aspen Movie Map*. Acesta, dezvoltat în 1978 de o echipă de la MIT (*Architecture*

Machine Group) condusă de Andrew Lippman, permite efectuarea unui tur virtual al orașului Aspen, Colorado. Compania Telos introduce în anul 1984 pe calculatoarele *Macintosh* sistemul hipermedia denumit *Filevision*. În 1985 iau naștere *Symbolic Document Examiner* (produs de Janet Walker) și *Intermedia*, ultimul fiind un sistem hipermedia conceput de o echipă condusă de Norman Meyrowitz la Brown University (SUA). În 1987, firma *Apple Computers* lansează *HyperCard*, primul sistem hipermedia larg răspândit. A se vedea și **multimedia**.

hiperspațiu (*hyperspace*). Ansamblul tuturor documentelor accesibile prin hiperlegături în World Wide Web. A se vedea și **ciberspațiu**, **hiperlegătură** și **Web**.

hipertext (*hypertext*). Mod de construcție a unui sistem de management și reprezentare a informației folosind legături pentru a crea o rețea de noduri. Reprezintă o tehnică pentru organizarea informației textuale printr-o metodă neliniară, în vederea facilitării explorării rapide a unei mari cantități de cunoștințe. Conform unei definiții a lui Paul Gisler, hipertextul este în același timp un proces mental și o unealtă digitală.

Conceptul de bază în definirea hipertextului este legătura (*link*-ul). Un document hipertext îi permite autorului său să creeze noduri și să le lege între ele, iar unui cititor să navigheze de la un nod la altul. Un nod poate conține orice fel de informație: text, grafică,

imagini, animații etc. Nodul-sursă al unei legături se numește *referință*, iar cel destinație *referent*.

Termenul *hypertext* a fost propus de Ted Nelson în articolul *Complex information processing: a file structure for the complex, the changing and the indeterminate*, publicat în anul 1965. Conform acestuia, hipertextul este un „material scris sau grafic interconectat într-o manieră complexă, care în mod convențional nu poate fi reprezentat pe hârtie”. Doi ani mai târziu, în 1967, Andy van Dam și alți colaboratori construiesc primul sistem de editare de hipertexte. În decembrie 1968, Douglas Engelbart face prima demonstrație publică a unei interfețe hipertext. Cel mai cunoscut exemplu de mediu hipertext este World Wide Web. A se vedea și **hiperlegătură** și **Nelson, Theodor Holm**.

Hollerith, Herman (1860–1929). Inventator, statistician și om de afaceri american, fondator al companiei IBM. În 1889 a construit prima mașină cu cartele perforate (*tabulating machine*). Hollerith a folosit această mașină la recensământul din 1890, pentru a prelucra datele provenite de la 63 056 000 de oameni într-o treime din timpul necesar în 1880 pentru a procesa datele furnizate de 50 262 000 de oameni. A se vedea și **IBM**.

home directory (*director de bază*). A se vedea **director de bază**.

Hopper, Grace Murray (1906–1992). Informatician și ofițer american,

probabil cea mai cunoscută programatoare din lume. Este prima femeie care și-a luat masteratul în matematică la Universitatea Yale. A proiectat și a condus dezvoltarea unuia dintre primele limbaje de programare moderne, COBOL. De asemenea, a făcut parte din echipa care a dezvoltat calculatorul UNIVAC I. A se vedea și **COBOL** și **UNIVAC**.

host (*gazdă*). A se vedea **gazdă**.

Host. Unul dintre anteturile cererii HTTP. Este utilizat pentru specificarea adresei Internet și a portului, în vederea stabilirii exacte a locului unde se află resursa respectivă (e.g., `Host: www.exemplu.ro`). Antetul este obligatoriu în HTTP/1.1, chiar dacă are o valoare vidă. În absența sa în cererile HTTP/1.1, serverul Web va răspunde întotdeauna cu codul de stare 400 (Bad Request). A se vedea și **cerere HTTP**.

HP-UX. Acronim pentru Hewlett-Packard UNIX. Este o versiune a sistemului de operare UNIX special creată pentru stațiile de lucru Hewlett-Packard. A se vedea și **Hewlett-Packard**.

href. Atribut al elementului HTML ancoră a prin intermediul căruia este creată într-o pagină Web o legătură hipertext. Atributul are ca valoare URL-ul resursei care constituie ținta legăturii. A se vedea și **hipertext**, **HTML**, **legătură** și **URL**.

.htaccess (*hypertext access*). În cazul unor servere Web (e.g., *Apache*) este

numele implicit al fișierului de configurare la nivel de director, utilizat pentru managementul descentralizat al configurației acestora. Fișierul începe cu un punct deoarece aceasta este convenția de denumire a fișierelor ascunse în sistemul de operare UNIX.

În mod obișnuit, fișierul *.htaccess* este utilizat pentru: autorizare și autentificare, personalizarea mesajelor serverului în cazul apariției unor erori, rescrierea URL-urilor (*URL rewriting*), controlul *cache*-ului. Deoarece fișierele *.htaccess* sunt citite la fiecare cerere HTTP, rezultă că modificările care se produc în acestea au efect imediat, spre deosebire de fișierul principal de configurare a serverului Web, în cazul căruia este necesară restartarea serverului pentru ca modificările efectuate să fie luate în considerare. În general, fișierele *.htaccess* sunt folosite de utilizatorii care nu au acces la fișierul principal de configurare (e.g., *httpd.conf*, pentru Apache). A se vedea și **cerere HTTP** și **rescrierea URL-urilor**.

.html. Extensie de fișier care identifică fișierele HTML afișate în browser ca pagini Web. Pentru același tip de fișiere se poate utiliza și extensia *.htm*. A se vedea și **HTML**.

HTML (*HyperText Markup Language*). Limbaj de marcare, *lingua franca* pentru publicarea hipertextului pe Web. Este un limbaj non-proprietar, bazat pe SGML. Specificațiile HTML sunt dezvoltate de Consorțiul Web. Cea mai recentă specificație este HTML 5 (schiță de lucru publicată pe 12 august 2009).

Documentele HTML, afișate în browser ca pagini Web, pot fi create utilizând o mare varietate de instrumente, de la editoarele de text (e.g., *Notepad* pe platforma Windows sau *TextEdit* pe Mac OS) până la instrumente complexe WYSIWYG (e.g., *Dreamweaver*). A se vedea și **Dreamweaver, limbaj de marcare, marcaj și XHTML.**

HTTP (*HyperText Transfer Protocol*). Set de reguli care guvernează comunicarea dintre clientul și serverul Web. Dezvoltarea protocolului este asigurată în prezent de Consorțiul Web, versiunea curentă – lansată în 1999 – fiind 1.1. Dintre caracteristicile protocolului HTTP pot fi amintite: este independent de platformă (calculatoarele care îl utilizează pentru comunicare pot folosi diverse sisteme de operare și aplicații hipertext); se bazează pe stiva de protocoale TCP/IP, funcționând la nivelul aplicație al acesteia; este protocolul implicit de comunicare între browserul și serverul Web; este suportat de toate browserele și serverele Web.

La fel ca în cazul majorității protocoalelor utilizate în Internet, HTTP are la bază o cerere (*request*) a unui client Web (e.g., browser) prin care se solicită o resursă stocată pe un server Web cunoscut, urmată de un răspuns (*response*) al acestuia. Cererea și răspunsul sunt denumite *mesaje* HTTP. Serverul Web pe care este stocată resursa solicitată de client este numit și *server de origine*. Motivul este acela că

între comunicarea dintre server și client pot să apară și intermediari, de exemplu un *proxy* (numit și *server proxy*), o poartă (*gateway*) sau un tunel (*tunnel*). Serverul *proxy* retrimite un mesaj HTTP, modificând – eventual – o parte a acestuia. *Poarta* este un intermediar situat înaintea unui server de origine, identificându-se drept acesta (clientul nu cunoaște acest lucru). Tunelul nu schimbă conținutul mesajului, având numai rol de retransmitere.

Pe scurt, comunicarea dintre clientul și serverul Web se desfășoară astfel: procesul server Web „ascultă” (implicit) portul 80 al mașinii-gază (server) pentru a răspunde cererilor venite de la clienți; clientul inițiază o conexiune HTTP cu serverul Web; după stabilirea conexiunii TCP, clientul care a inițiat conexiunea realizează o cerere HTTP pentru o resursă situată pe server; serverul trimite un răspuns HTTP, acesta incluzând un cod de stare care indică starea serverului după interpretarea cererii, un mesaj explicativ asociat codului respectiv, *meta*informații care vor fi procesate de client și, dacă aceasta există, resursa cerută.

hub (*distribuitor*). Dispozitiv utilizat de regulă pentru conectarea rețelelor locale de mici dimensiuni (în tehnologia stea). Nu conține componente electronice active și nu poate fi utilizat pentru extinderea unei rețele locale.

hyperlink (*hiperlegătură*). A se vedea **hiperlegătură.**

I

i2010. Inițiativă lansată de Comisia Europeană în iunie 2005, este o strategie pe cinci ani a Uniunii Europene menită să conducă la dezvoltarea economiei digitale. *i2010* reprezintă elementul fundamental al strategiei Lisabona în vederea construirii unei economii competitive, bazată pe cunoaștere și care să ofere creștere economică și locuri de muncă. Strategiile și acțiunile prezentate în *i2010 Communication* (2005) sunt revăzute și actualizate prin rapoarte anuale (*i2010 Annual Reports*). De asemenea, aceste rapoarte analizează dezvoltarea sectorului TIC și evaluează progresul statelor membre în implementarea propriilor obiective TIC.

Promotoarea inițiativei *i2010* este Vivian Reding, doctor în științe umaniste, membru (în 2005) al Comisiei Europene, responsabil cu societatea informațională și media. În opinia acesteia, „tehnologiile informatice și de comunicații reprezintă temelia unei economii moderne, o sursă a inovării și un sector economic a cărui importanță este în creștere”, având în vedere realizarea „convergenței digitale” între

TIC, conținutul media și alte echipamente.

Strategia *i2010* prevede: crearea unei piețe unice și competitive pentru societatea informațională (pentru a sprijini convergența tehnologică printr-o convergență politică, se au în vedere: o politică eficientă de management al spectrelor de frecvențe, modernizarea regulilor privind audiovizualul, actualizarea cadrului de reglementare privind comerțul electronic, elaborarea unei strategii eficiente pentru o societate informațională sigură și o abordare cuprinzătoare privind gestionarea eficiente a drepturilor privind produsele digitale); creșterea cu 80% a investițiilor în cercetarea privind TIC, plecând de la constatarea că în UE se investește puțin în acest domeniu, comparativ cu Japonia și SUA; promovarea unei societăți informaționale cuprinzătoare prin: implementarea unor servicii de e-guvernare orientate către cetățean, trei inițiative privind calitatea vieții (tehnologii specifice pentru o societate îmbătrânită, vehicule inteligente, mai sigure și mai nepoluante, biblioteci virtuale digitale), acțiuni menite să atenueze „prăpăstiile digitale” cu caracter geografic și sociale, culminând cu o inițiativă europeană pentru „e-incluziune”.

IAB (*Internet Architecture Board*). Parte a ISOC, se ocupă de problemele de arhitectură globală privind Internetul. IAB reglementează eventualele dispute care pot să apară în cursul

elaborării standardelor. A se vedea și **ISOC**.

IANA (*Internet Assigned Numbers Authority*). Organizație responsabilă pentru coordonarea globală a DNS-urilor și a adreselor IP, precum și pentru menținerea unor coduri incluse în protocoalele Internet.

Realizează operarea și menținerea unor elemente esențiale ale DNS, incluzând zona-rădăcină (*root zone*) și domeniile *.int* și *.arpa* (<http://www.iana.org/>). În ceea ce privește adresele IP, rolul IANA este acela de a le aloca pe cele disponibile către registrele regionale, RIR (*Regional Internet Registry*), în acord cu nevoile acestora. De asemenea, IANA este responsabilă pentru menținerea unor coduri și numere incluse într-o varietate de protocoale Internet, în coordonare cu IETF. A se vedea și **IETF**.

IBM (*International Business Machines Corporation*). Corporație multinațională cu sediul în SUA (denumită pe scurt *Big Blue*, datorită culorii albastre utilizate în logo-ul său oficial), care activează în domeniul tehnologiei computerelor și consultanței IT, având o existență neîntreruptă încă din secolul al XIX-lea.

IBM a fost fondată în anul 1896 sub denumirea *Tabulating Machine Company*, de către Herman Hollerith. În 1911, TMC a fuzionat cu alte trei corporații, formând CRT (*Computing Tabulating Recording Corporation*). Compania a fost redenumită în 1924,

când a primit numele actual. În 1981, compania a lansat pe piață calculatorul IBM PC. De asemenea, compania dezvoltă numeroase produse software (*e.g.*, *WebSphere*, *Lotus*, *Tivoli*, *DB2*). A se vedea și **Hollerith**, **Herman** și **IBM Personal Computer**.

IBM Personal Computer. Cunoscut ca IBM PC, este versiunea originală și „părintele” actualelor calculatoare personale compatibile IBM PC. A fost creat de o echipă de ingineri și proiectanți ai IBM condusă de Don Estridge și lansat pe piață în august 1981. A se vedea și **Estridge**, **Philip Donald** și **IBM**.

ICMP (*Internet Control Message Protocol*). Protocol Internet utilizat la nivelul rețea (stratul 3 în modelul ISO/OSI). Asigură detectarea problemelor în rețea (*e.g.*, noduri inaccesibile, congestie de pachete). Protocolul IP utilizează informațiile pentru a lua măsurile care se impun (*e.g.*, actualizarea tabelelor de dirijare, *internal routing tables*). A se vedea și **IP** și **ISO/OSI**.

icon (*icon*). Derivat din cuvântul grecesc *eikon*, este o imagine de dimensiuni mici asociată unui program, unei aplicații, unui site Web sau unei comenzi. Icon-ul este folosit pentru a deschide rapid un document sau o aplicație, pentru a executa o comandă și pentru a preciza natura/conținutul unui site Web. De asemenea, icon-ul permite recunoașterea tipului unui fișier dintr-o listă de fișiere, cele cu aceeași

extensie având același icon. Pentru a executa o comandă prin intermediul unui icon (e.g., deschiderea sau editarea unui document, unei imagini etc.), se execută dublu click pe acesta. Spre deosebire de imaginile obișnuite, iconurile au dimensiuni standard (e.g., 16x16, 32x32, 48x48).

IDE (*Integrated Development Environment, mediu integrat de dezvoltare*). Software complex care oferă posibilitatea dezvoltării altor programe, conducându-l pe programator prin toți pașii necesari (e.g., editarea codului-sursă, compilare, depanare, testare, generarea documentației). De obicei, un mediu de dezvoltare este specific unui limbaj de programare, dar există și medii care pot utiliza mai multe limbaje. A se vedea și **limbaj de programare**.

identificator (*identifier*). În limbajele de programare, șir folosit pentru a denumi o entitate (e.g., variabilă, funcție, clasă). A se vedea și **limbaj de programare**.

identitatea 2.0 (*identity 2.0*). Este, în mod similar celei reale, o expresie a reputației individului, supusă unor rigori de verificabilitate și încredere socială. Focalizează atenția către utilizator, devenit în cadrul noului Web elementul principal al procesului de interacțiune din mediul virtual.

Identitatea 2.0 se constituie ca o îmbinare a mai multor fațete, incompletă totuși, din cauza procesului permanent de construcție/deconstrucție și

dezvoltare. Dintre dimensiunile sale pot fi evidențiate următoarele: existența virtuală reală (activitățile virtuale desfășurate de un utilizator), dimensiunea personală (persoanele agreeate), dimensiunea profesională (ocupația efectivă), reputația (contactele pe care le stabilește *online*), cunoștințele, resursele partajate, *hobby*-urile, comportamentul comunicațional. A se vedea și **identitatea digitală**.

identitatea digitală (*digital identity*). Cunoscută și ca *identitate virtuală* (*virtual identity*), este suma reprezentărilor despre sine, „afișate” sau declarate, ale unei persoane care comunică *online*, definind roluri, acțiuni, informații deținute și calități ale acesteia.

Rolul constă în funcția declarată (adeverată sau nu) a utilizatorului (e.g., profesor, actor, avocat, star de cinema), implicând sexul, rasa și vârsta acestuia. *Acțiunile* reprezintă fapte săvârșite sau nu, reale sau imaginare ale acestuia (e.g., primul om care a pășit pe Marte). *Informațiile deținute* exprimă anumite achiziții cognitive ale utilizatorului, reale sau imaginare, reprezentative într-un anumit context (e.g., persoana care știe numele asasinului lui JFK).

Calitățile se referă la relația de proprietate a utilizatorului asupra anumitor bunuri, reale sau simbolice (e.g., cel care are în proprietate un castel, cel care deține un autoturism foarte scump, cel care deține 51% dintre acțiunile unei companii).

Identitatea virtuală este un construct pur imaginar al individului și poate reflecta total, parțial sau deloc viața lui reală. Abilitatea de a asuma un *alias*, o nouă identitate, este o experiență liberă, nesupusă vreunei constrângerii directe.

În ciber spațiu este ușor pentru utilizatori să-și construiască o identitate creativă, o personalitate virtuală sau chiar o mască în relația cu alți membri ai comunității. Pe de altă parte, Internetul, ca mediu interacțional prin excelență, permite exprimarea sub diverse forme a unei largi palete de trăsături umane. În acest sens, comunitățile virtuale (în special) reprezintă un cadru de construcție a unei imagini de sine favorizând exhibarea unor caracteristici și preferințe care în lumea reală (*i.e.*, mediul social, politic, cultural în care trăiesc utilizatorii) ar putea fi considerate inacceptabile, reprobabile sau greu de afișat.

Identitatea virtuală – ca o personalitate secundară a individului – poate fi considerată o modalitate de defulare, de proiecție a unor elemente de personalitate care nu își găsesc o manieră potrivită de obiectivare în lumea reală. În spațiul virtual, elementele indezirabile social găsesc un canal de manifestare, sub acoperirea oferită de fenomenul dezindividualizării.

Multiplele identități virtuale sugerează că fiecare individ este compus din multiple fațete. Sherry Turkle califică identitățile pe care le adoptă utilizatorii Internetului drept *măști*, caracterizate

de manifestarea alternantă și totuși simultană, pe diferite paliere pe care este vizată autoprezentarea și exprimarea sinelui. Astfel, idealul omniprezenței este atins în era digitală, fiind permisă prezența simultană în mai multe planuri: cel real și unul sau mai multe planuri virtuale, imaginea de sine configurându-se ca un sistem complex, distribuit. În opinia lui Turkle, „identitatea pe computer a unei persoane este suma prezențelor sale distribuite”.

În spațiul virtual, eul apare ca esență, fiind eliminate barierele corpului material, considerat adesea ca o frână în calea unei existențe dezirabile, desfășurată în absența unor constrângeri impuse de dimensiunile spațială și temporală ale existenței reale. Astfel, Internetul facilitează o multiplicare practic nelimitată a posibilităților de configurare a sinelui în cadrul unui spațiu virtual imens, care permite crearea – în funcție de context – a unor identități diverse – uneori lipsite de substanță și efemere –, dincolo de sinele unitar.

Aplicațiile Web 2.0 le oferă utilizatorilor posibilitatea unor experiențe sociale complexe. Aceștia sunt investiți cu posibilitatea de a-și construi ei înșiși spațiul de relaționare în mediul virtual (*e.g.*, rețele sociale), care – astfel – devine din ce în ce mai umanizat. În cadrul acestor schimbări radicale, identitatea virtuală evoluează și ea către o nouă etapă: *identitatea 2.0*.

Una dintre cele mai revoluționare idei din domeniul identității virtuale constă

în dezvoltarea unui mecanism prin care se urmărește eliminarea necesității folosirii unor nume de utilizator multiple (identități diverse) pe site-uri diferite, simplificând considerabil experiența *online* a utilizatorilor. În acest sens s-au dezvoltat proiecte ca OpenID și Sxip. Sistemele create conform acestor specificații, *MyOpenId* (<https://www.myopenid.com/>), *Sxipper* (<http://www.sxipper.com/>) și *PIP Verisign* (<https://pip.verisignlabs.com/>), prefigurează – în contextul experiențelor sociale inedite oferite de aplicațiile Web 2.0 – posibilitatea unei identificări unice, implicite a inter-nauților. A se vedea și **identitatea 2.0**.

idle. Neutilizat, cu activitate redusă, dar în stare de funcționare. Acesta poate fi, de exemplu, statusul unui utilizator pe *Yahoo! Messenger*.

IEEE (*Institute of Electrical and Electronics Engineers*). Organizație profesională internațională nonprofit, reprezentând autoritatea tehnică cea mai importantă în domenii ale ingineriei electrice și electronice tehnologia informației, comunicațiile, electroenergetica, electronica aerospațială, electronica biomedicală și electronica bunurilor de consum (<http://www.ieee.org/>).

IESG (*Internet Engineering Steering Group*). Grup inclus în ISOC împreună cu IAB, revizuieste standardele propuse de IETF. A se vedea și **IAB**, **IETF** și **ISOC**.

IETF (*Internet Engineering Task Force*). Comunitate deschisă de arhitecți de rețele, operatori, comercianți și cercetători având ca interese comune evoluția arhitecturii Internetului și funcționarea corectă a acestuia. Activitatea comunității se desfășoară formal sub tutela *Internet Society* (<http://www.ietf.org/>).

IETF elaborează și promovează standarde pentru Internet legate în special de suita de protocoale TCP/IP, în colaborare cu W3C și ISO/IEC. Propunerile sunt transmise către IAB. IETF este condusă de IESG. Lucrările IETF și coordonarea acestora sunt realizate exclusiv de voluntari, efortul acestora fiind susținut material de instituțiile la care aceștia lucrează.

IETF este organizată într-un număr mare de grupuri de lucru și grupuri informale de discuții, fiecare ocupându-se de un subiect specific. Grupurile de lucru sunt incluse în arii de activitate (*Applications, General, Internet, Operations and Management, Real-time Applications and Infrastructure, Routing, Security and Transport*). A se vedea și **IANA**, **IAB**, **IESG** și **Internet Society**.

iGoogle. Serviciu pentru personalizarea paginii de start oferit de compania Google (<http://www.google.com/ig/>). A se vedea și **serviciu pentru personalizarea paginii de start**.

image inline (*inline image*). Imagine înglobată în textul unui document (e.g., HTML). A se vedea și **HTML**.

IMAP4 (*Internet Message Access Protocol*). Versiunea 4 a protocolului IMAP (care funcționează la nivelul aplicație al modelului de referință TCP/IP) pe baza căruia un client de *e-mail* are acces la mesajele stocate pe serverul de poștă electronică, fără a le descărca pe calculatorul utilizatorului. IMAP – proiectat de Mark Crispin în 1986 – este mai recent și mai avantajos decât POP, oferindu-i utilizatorului posibilitatea de a avea acces la contul său de *e-mail* din diverse locații. A se vedea și **POP3** și **poșta electronică**.

(a) **imbrica** (*nest*). A include o structură în alta, ca în exemplele următoare: o înregistrare poate include un câmp care, la rândul lui, este tot o înregistrare; o instrucțiune de control (*e.g.*, *do...while, for*) poate îngloba o altă instrucțiune similară; definiția unei funcții poate include definiția unei alte funcții.

Imooty.eu. Primul agregator realizat cu scopul de a concentra știri publicate în diverse surse aflate pe continentul european (*e.g.*, bloguri independente), incluzând *feed*-uri publicate pe site-uri din Uniunea Europeană și din alte state (<http://www.imooty.eu/>). A se vedea și **agregator** și **feed**.

imprimantă (*printer*). Echipament periferic de ieșire care reproduce pe hârtie sau pe un alt suport (*e.g.*, folie transparentă) textul sau imaginile generate de calculator. Suportul este secvențial, adică poate fi parcurs într-un singur sens, fără posibilitate de revenire

în scopul rescrierii unor porțiuni. O imprimantă are în componență blocul de imprimare (care imprimă conținutul pe suport prin diverse procedee), sistemul de avans/antrenare a hârtiei, sistemul logic de comandă (procesorul), panoul cu butoane și/sau led-uri, interfața.

Pentru caracterizarea performanțelor unei imprimante sunt utilizați următorii parametri: rezoluția, viteza, dimensiunea hârtiei, capacitatea memoriei proprii, extinderea setului de caractere.

La clasificarea imprimantelor sunt utilizate mai multe criterii: tehnologia utilizată la tipărire (cu cap rotitor, matriceale, cu jet de cerneală, cu laser și termice), metoda de transmisie (paralele sau seriale), modul de formare a caracterelor (imprimante care produc caractere integrale, imprimante care produc caractere din puncte), metoda de tipărire (caracter cu caracter, rând cu rând, pagină cu pagină), calitatea tipăririi (scăzută, medie, înaltă).

imprimantă cu jet de cerneală (*ink-jet printer*). Cel mai utilizat tip de imprimantă pentru calculator. Utilizează un cartuș cu jet de cerneală fluidă, având o funcționare ușoară și eficientă. Aceste imprimante sunt mai lente, dar și mai ieftine decât imprimantele laser. Totodată, imprimantele cu jet de cerneală oferă o soluție relativ ieftină pentru printarea documentelor color. A se vedea și **imprimantă**.

imprimantă cu laser (*laser printer*). Tip de imprimantă care tipărește rapid text și imagini de calitate superioară pe hârtie obișnuită, utilizând tehnologia fotocopiatoarelor. A fost inventată în 1969 de Gary Starkweather, pe atunci cercetător la Xerox. A se vedea și **imprimantă**.

imprimantă partajată (*shared printer*). Imprimantă utilizată în comun de mai multe calculatoare. A se vedea și **imprimantă**.

imprimantă virtuală (*virtual printer*). Software ale cărui interfețe utilizator și API sunt similare cu cele ale unui driver de imprimantă, dar care nu este conectat cu o imprimantă fizică. Dacă un utilizator solicită „tipărirea” unui document de către o imprimantă virtuală, software-ul respectiv va procesa conținutul acestuia într-o modalitate specifică, având ca rezultat crearea unui fișier sau transmiterea informațiilor către diverse dispozitive. Iată câteva dintre posibilitățile de utilizare a imprimantelor virtuale: producerea fișierelor PDF, trimiterea documentelor către un server de fax, conversia documentelor în imagini (e.g., JPEG). A se vedea și **dispozitiv virtual** și **imprimantă**.

IMS Content Packaging. Cel mai folosit standard și format pentru împachetarea conținutului obiectelor de învățare.

Recomandările standardului IMS Content Packaging permit o modalitate de specificare a obiectelor de învățare

care pot fi schimbate între diverse programe de instruire, facilitând astfel furnizarea, reutilizarea și partajarea materialelor didactice. Concret, acest standard permite: asamblarea (agregarea) într-un singur pachet a unui conținut existent, pentru livrare; transmiterea conținutului de la un mediu virtual de instruire (VLE) la altul; importul conținutului într-un VLE (dintr-un depozit sau de la producător); exportul conținutului astfel încât acesta să poată fi utilizat de alte persoane, departamente sau instituții; transferul conținutului de la un instrument de tip *authoring* la un VLE sau depozit.

Unul dintre beneficiile-cheie ale IMS Content Packaging este acela că diverse materiale-suport pentru o piesă de conținut (e.g., foi de stil, clipuri video și imagini) pot fi puse împreună, oferind certitudinea că acestea nu se vor deteriora din cauza ruperii piesei în mai multe părți. De asemenea, ele pot fi dezagregate și reutilizate ca obiecte de învățare independente. Un alt beneficiu este acela că pachetele de conținut utilizează metadate, care permit filtrarea și căutarea/regăsirea lor în depozitele de obiecte de învățare.

Un pachet de conținut IMS constă dintr-o arhivă care include o colecție de fișiere (resurse) și un fișier manifest. Pentru crearea manifestului este utilizat metalimbajul XML. Acest fișier descrie conținutul și modul de organizare a pachetului, incluzând în acest scop următoarele trei secțiuni: *metadata* (metadata) – oferă o descriere generală

a pachetului; *organizations* (organiza-re) – descrie structura resurselor incluse în pachet; *resources* (resurse) – este o listă a resurselor incluse în pachet (pre-cizate prin intermediul unor hiperle-gături la resursele Web sau folosind căi relative către fișierele din arhivă) și metadata individuale. De asemenea, în preambulul fișierului manifest se declară spațiile de nume folosite și numele fișierului respectiv (exprimat ca valoare a atributului *identifier*) dat de creatoful cursului. Fișierul manifest poate fi scris „manual”, utilizând un editor de text oarecare (*e.g.*, *Notepad*), sau poate fi generat automat de diverse instrumente software.

Resursele din arhivă pot avea orice tip: imagini GIF sau JPEG, documente PDF, fișiere HTML, documente Word. Evident, este necesar ca receptorul pachetului IMS să includă o aplicație care permite vizualizarea conținutului fișierelor. Cerințele legate de existența acestor aplicații pot fi incluse în metadatale pachetului sau ale resurselor individuale (*e.g.*, o versiune particulară a unui browser, necesară pentru a vizualiza o pagină Web; o placă de sunet, necesară pentru a asculta un fișier audio).

Pentru crearea unui pachet de conținut IMS sunt necesare materiale de învățare în format electronic și un instrument de împachetare a conținutului. Acesta din urmă poate fi inclus chiar în mediul virtual de învățare (VLE) sau în instrumentul de *authoring*. Pentru vizualizarea unui pachet IMS trebuie să

se realizeze importul acestuia într-un VLE sau într-o altă aplicație care implementează specificațiile IMS Content Packaging.

Pachetele de conținut IMS sunt distribuite în mod obișnuit ca arhive ZIP, putând fi trimise chiar și prin *e-mail*, ca fișiere atașate și apoi încărcate în VLE de pe calculatorul pe care au fost descărcate. În cazul unor aplicații, pachetele de conținut pot fi trimise direct, folosind unul dintre protocoalele HTTP, FTP sau SOAP.

Împachetarea conținutului educațional într-un pachet IMS permite folosirea în comun a acestuia între sisteme precum instrumentele de *authoring*, depozitele digitale și mediile virtuale de instruire. A se vedea și **FTP**, **HTTP**, **mediu virtual de instruire**, **obiect de învățare** și **SOAP**.

inbound link (*legătură de intrare*). Se referă la legăturile incluse în alte site-uri și care duc către site-ul propriu. A se vedea și **legătură**.

incoming link (*legătură de sosire*). A se vedea **backlink**.

increment (*increment*). Valoarea cu care este mărit un număr prin incrementare. A se vedea și (a) **incrementa** și **decrement**.

(a) **incrementa** (*increment*). A mări un număr cu o anumită valoare (*e.g.*, 1). A se vedea și **increment** și (a) **decrementa**.

independent de mașină (*machine-independent*). Produs software sau

hardware care poate fi utilizat fără modificări pe mai multe tipuri de calculatoare. A se vedea și **dependent de mașină**.

index (*index*). Bază de date folosită de motoarele de căutare, în care sunt păstrate informații despre paginile Web vizitate și indexate de acestea. A se vedea și **indexer** și **motor de căutare**.

indexare automată (*automatic indexing*). Procesul de examinare automată a unui ansamblu de itemi pentru a construi o structură de date în care se pot realiza căutări rapide.

indexer (*indexer*). Parte a unui motor de căutare, primește de la *spider* informații despre paginile găsite. Creează înregistrări ale acestora într-o bază de date numită *index*, păstrând și URL-urile. Motoarele puternice (*e.g.*, *Google*) folosesc un număr mare de copii ale indexului, situate pe servere multiple, deoarece unul singur nu ar putea face față unui număr foarte mare de cereri ale utilizatorilor. A se vedea și **index**, **motor de căutare** și **spider**.

INET. Conferință anuală organizată de *Internet Society* (ISOC). A se vedea și **Internet Society**.

informatică (*computer science*). Disciplină vastă care studiază fundamentele teoretice ale informației și procesării acesteia, precum și modalitățile practice pentru implementarea lor în sistemele de calcul. Combină cunoștințe preluate din inginerie, electronică, teoria informației, logică și matematică.

Subdomeniile informaticii sunt (între paranteze au fost incluse exemple de câmpuri de studiu care aparțin sub-domeniului): informatica teoretică (*e.g.*, logică matematică, teoria automatelor, teoria grafurilor, geometrie computațională), algoritmi și structuri de date (*e.g.*, analiza algoritmilor, algoritmi, structuri de date), metodologia programării și limbaje de programare (*e.g.*, compilatoare, limbaje de programare), elemente de computer și arhitectură (*e.g.*, logică digitală, microarhitecturi, multiprocesare), calcul numeric și simbolic (*e.g.*, algebră computațională, chimie computațională, științe cognitive, fizică computațională, bioinformatică, algoritmi numerici), aplicații (*e.g.*, sisteme de operare, rețele de calculatoare, inteligență artificială, securitatea calculatoarelor, robotică, grafică pe calculator, baze de date, interacțiunea om-calculator).

inginerie software (*software engineering*). Disciplină care stabilește și utilizează principiile ingineriei pentru a realiza produse software de calitate, ținând seama de limitările datorate resurselor. Termenul a fost folosit prima dată în anul 1968, într-o conferință a NATO pe tema crizei software-ului.

Dezvoltarea sistemelor informatice include mai multe activități strâns legate între ele, reprezentând etape ale ciclului de viață (*life cycle*) al produsului informatic.

În anii 1960–1980, când s-au dezvoltat diverse modalități de a răspunde crizei software, majoritatea sistemelor

informatică erau dezvoltate în scopul prelucrării unor cantități mari de date, motiv pentru care multe metode tradiționale dezvoltate în cadrul ingineriei software se ocupă în special cu modelarea datelor, abordând mai puțin problemele de interactivitate.

Demersurile tradiționale utilizate pentru implementarea etapelor enumerate anterior, demersuri numite și *paradigmele dezvoltării software-ului*, sunt: modelul în cascadă (*waterfall model*), modelul de dezvoltare exploratorie, modelul de dezvoltare evolutivă, modelul în spirală (*spiral model*). Dintre modelele enumerate, primul și ultimul sunt cele mai cunoscute și mai des utilizate.

Modelul în cascadă propune o abordare sistematică în care fiecare etapă este bine definită și produce o ieșire care devine intrare pentru etapa următoare. Conform acestui model, cele patru etape care trebuie parcurse în dezvoltarea unui produs software sunt: analiza și specificarea cerințelor, proiectarea, implementarea, testarea. Modelul nu constituie o analogie perfectă cu practica dezvoltării sistemelor informatice, aceasta din urmă având în realitate un caracter iterativ. Astfel, după fiecare etapă urmează o evaluare a rezultatelor, care poate avea drept consecință o reluare a etapelor precedente. În cadrul modelului în cascadă, se acordă o importanță secundară aspectelor legate de interactivitate, deși – în cadrul dezvoltării sistemelor informatice – ponderea realizării interfeței este ridicată. De asemenea, în

prima etapă sunt implicați numai analiștii de sistem, deși este necesară și o analiză a cerințelor utilizatorilor.

Modelul în spirală (cunoscut și sub numele de modelul bazat pe risc), dezvoltat de Barry W. Boehm în 1988, reprezintă o abordare alternativă la modelul în cascadă, în care proiectul este realizat într-o serie de cicluri succesive, fiecare încheindu-se cu o versiune de software executabil. Această metodă are avantajul că permite obținerea unui *feedback* periodic din partea clientului și, implicit, identificarea problemelor înainte de a ajunge la versiunea finală a produsului. Astfel, se pot schimba parțial chiar și specificațiile. De asemenea, metoda permite încorporarea mai rapidă în produs a schimbărilor produse în tehnologiile utilizate.

Alături de metodele tradiționale utilizate pentru dezvoltarea *software-ului*, s-au dezvoltat o serie de abordări moderne. Acestea folosesc același principiu ca și modelul în spirală (*i.e.*, realizarea proiectului în cicluri succesive), dar propun abordări mai formalizate. Dintre metodele moderne, menționăm procesul unificat (*Unified Process*). În cadrul său, procesul de dezvoltare a *software-ului* este împărțit în cicluri – similar procesului în spirală –, fiecare ciclu referindu-se la o nouă generație a *software-ului*. Un ciclu de dezvoltare este împărțit în patru faze succesive: început (*inception*), elaborare (*elaboration*), construcție (*construction*) și tranziție (*transition*).

Pentru dezvoltarea site-urilor/aplicațiilor Web poate fi utilizat cu succes modelul în spirală. Astfel, dezvoltarea acestora se va desfășura iterativ, rezolvându-se pe parcurs problemele generate de specificarea incompletă a cerințelor sau de apariția unor noi. În procesul de creare a unui site/a unei aplicații, colaborarea cu beneficiarii poate fi fructuoasă, rezultând detalieri ale cerințelor și o specificare cât mai exactă a acestora. A se vedea și **ciclu de viață**.

inline linking (*legare inline*). Metodă utilizată pentru legarea unui obiect (e.g., imagine, clip audio) aparținând unui site într-o pagină Web inclusă într-un alt site. Se spune că al doilea site are o legătură *inline* către primul site. În acest scop, se utilizează URL-uri absolute pentru atributele marcajelor (X)HTML folosite pentru includerea obiectelor în pagină. De exemplu, pentru atributul `src` al marcajului `img` se va utiliza un URL de genul: `http://www.exemplu.ro/imagini/iarna.jpg`. Metoda este posibilă datorită faptului că protocolul HTTP permite includerea într-o pagină Web a unor obiecte/resurse situate pe servere multiple. A se vedea și **HTML**, **HTTP**, **pagină Web** și **URL**.

inițializare (*initialization*). 1. Procesul de alocare a unor valori inițiale variabilelor și structurilor de date declarate și utilizate într-un program. A se vedea **program** și **variabilă**. 2. Pornirea unui calculator.

Inițiativa de la Budapesta. Având denumirea completă *Inițiativa de la Budapesta cu privire la accesul deschis* (BOAI, *Budapest Open Access Initiative*), este primul moment important din istoria mișcării *Open Access* (februarie 2002). A se vedea și **acces deschis**.

Inkscape. Aplicație *open source* de editare grafică vectorială, scrisă în C++, cu versiuni pentru sistemele de operare Linux, FreeBSD, Mac OS X și Windows (<http://www.inkscape.org/>).

instanță (*instance*). Obiect creat în momentul execuției (*runtime*). Operația de creare a unui obiect – pornind de la o clasă dată – se numește *instanțiere*. A se vedea și **clasă**, **obiect** și **programare orientată pe obiect**.

instrucțiune (*statement*). Într-un limbaj de programare imperativă, cea mai mică unitate executabilă. Una sau mai multe instrucțiuni alcătuiesc un program (sau script). A se vedea și **program** și **programare imperativă**.

instrucțiune condițională (*conditional statement*). În limbajele de programare, instrucțiune care selectează calea de execuție în funcție de îndeplinirea unei condiții specificate de programator. În acest scop, condiția este evaluată la o valoare de adevăr (*true*, dacă este îndeplinită, și *false*, dacă nu este îndeplinită). În cele mai multe limbaje de programare, instrucțiunile condiționale utilizate sunt *if-else* (instrucțiunea alternativă) și *switch* (instrucțiunea selectivă). A se vedea și

instrucțiune și instrucțiune de control.

instrucțiune de control (*control statement*). Instrucțiune care afectează fluxul execuției unui program sau determină execuția repetată a unor secvențe de cod. Implementează în limbajele de programare o structură de control. Sunt utilizate instrucțiuni condiționale (e.g., *if-else*, *switch*) și instrucțiuni repetitive (*while*, *do..while*, *for*). A se vedea și **instrucțiune, instrucțiune condițională, instrucțiune repetitivă și structură de control.**

instrucțiune-mașină (*machine instruction*). Instrucțiune în cod-mașină care poate fi executată direct (i.e., fără a fi translatată într-un alt format) de microprocesor. Fiecare instrucțiune trebuie să includă elementele necesare pentru execuția acesteia: codul operației (cod binar care specifică operația care trebuie executată), adresa operandului-sursă, adresa rezultatului (dacă operația produce un rezultat), adresa instrucțiunii următoare.

Instrucțiunile-mașină variază de la un microprocesor la altul, în majoritatea calculatoarelor fiind întâlnite următoarele tipuri: transferuri de date, instrucțiuni aritmetice, instrucțiuni logice, instrucțiuni de salt și de apel, instrucțiuni de control și instrucțiuni de I/E. A se vedea și **cod-mașină și set de instrucțiuni.**

instrucțiune de procesare XML (*XML PI, XML Processing Instruction*).

Precizează informații referitoare la anumite aplicații externe care urmează a fi executate în vederea procesării conținutului XML. Instrucțiunile conțin, la început, secvența „<?”, apoi numele aplicației și date specifice acesteia și, la sfârșit, secvența „>”. A se vedea și **XML.**

instrucțiune repetitivă (*repetitive instruction*). În limbajele de programare, instrucțiune care determină execuția repetată a unui grup de instrucțiuni. În cele mai multe limbaje de programare instrucțiunile repetitive sunt *for*, *do* și *do..while*. A se vedea și **instrucțiune de control.**

instrument de evaluare (*assessment tool*). Metodă utilizată pentru obținerea unor informații utile despre activitatea de învățare a unui educabil (e.g., elev, student). Scopul utilizării unui instrument de evaluare este de a oferi educatorului informațiile care îi permit să aleagă acele strategii educaționale potrivite pentru îmbunătățirea performanțelor educabilului într-un anumit domeniu.

integritatea bazei de date (*database integrity*). Condiția unei baze de date în care toate datele existente respectă o serie de constrângeri impuse de natura acestora sau de relațiile dintre ele. Menținerea integrității bazei de date presupune verificarea integrității datelor la fiecare operațiune de actualizare (e.g., ștergere, modificare, inserare). În teoria relațională se utilizează următoarele trei principii de

integritate: principiul integrității domeniului (valorile unui atribut trebuie să aibă un tip, o lungime și o valoare implicită precizate), principiul integrității entităților (într-o relație de bază nici un atribut al unei chei primare nu poate fi *null*) și principiul integrității referențiale (dacă într-o relație există o cheie străină, valoarea acesteia trebuie să coincidă cu valoarea unei chei-candidat a unei înregistrări în relația sa de bază sau să fie în întregime *null*). A se vedea și **bază de date relațională**.

integritatea datelor (*data integrity*). Se referă la păstrarea datelor în forma inițială sau așteptată. Integritatea poate fi compromisă în diverse moduri: producerea unor erori umane la introducerea lor (e.g., într-o bază de date), apariția unor erori la transmisia datelor de la un calculator la altul, funcționarea defectuoasă a dispozitivelor (e.g., deteriorarea hard-disk-ului), existența unor probleme ale software-ului (e.g., *bug*-uri, viruși), prelucrare defectuoasă.

integritate estetică (*aesthetic integrity*). Principiu conform căruia un design trebuie să fie atrăgător din punct de vedere vizual și să aibă o identitate clară, o ierarhie vizuală potrivită, o bună aliniere și proporții corecte.

Intel Corporation (*Integrated Electronics Corporation*). Cel mai mare producător mondial de materiale semiconductoare (<http://www.intel.com/>). Compania a fost înființată în anul 1968 de Robert Noyce (fizician, coinventator

al circuitelor integrate, alături de Jack Kilby) și Gordon Moore (chimist și fizician), cărora li s-a alăturat ulterior Andrew Grove. Acesta din urmă a avut un rol-cheie în succesul companiei.

Intel a inventat arhitectura x86, utilizată în microprocesoarele incluse în majoritatea calculatoarelor personale. Microprocesoarele produse de Intel (e.g., Pentium, Intel Core și Intel Atom, Intel Core i7) sunt larg utilizate în PC-uri. A se vedea și **x86** și **Moore, Gordon Earle**.

intelență artificială (*artificial intelligence, AI*). Domeniu de cercetare pentru construirea sistemelor inteligente. În opinia profesorului John McCarthy (inventatorul conceptului, pe care l-a prezentat în vara anului 1956 la întrunirea *Dartmouth Summer Research Project on Artificial Intelligence*), un astfel de sistem este „o mașină cu un comportament care ar putea fi considerat inteligent, dacă ar fi vorba de un om”. Sistemele inteligente sunt capabile să învețe, în scopul autoperfecționării, fără ajutor extern.

Un sistem (mașină) este inteligent dacă respectă definiția lui John McCarthy și trece *testul Turing*: un arbitru (om) se angajează într-o conversație în limbaj natural cu două entități, un om și o mașină; dacă arbitrul nu poate preciza cu certitudine cine este omul și cine este mașina, sistemul a trecut testul.

La baza inteligenței artificiale stau numeroase domenii ale cunoașterii umane: matematica, informatica, automatica și cibernetica, lingvistica,

psihologia, biologia. În mod tradițional, AI s-a bazat pe calculul logic și pe manipularea simbolurilor de către calculatoare, tehnici folosite astăzi în sistemele-expert. O alternativă la modelele simbolice sunt modelele conexioniste, care folosesc rețele neuronale artificiale. Acestea din urmă au la bază principii derivate din modul de funcționare a creierului uman, folosind un număr mare de elemente simple (numite noduri sau celule) interconectate. Totuși, unii oameni de știință afirmă că inteligența umană este imposibil de egalat de către o mașină. În 1989, matematicianul britanic Roger Penrose susținea că mecanismul de funcționare specific creierului uman nu poate fi replicat de mașină, nici măcar în principiu.

Iată câteva dintre principalele aplicații ale AI: sistemele-expert, logica și sistemele *fuzzy*, algoritmi genetici, rețele neuronale, agenții inteligenți, sintetizarea vorbirii, recunoașterea automată a formelor (*e.g.*, scrisul) și a sunetelor (*e.g.*, vocea). A se vedea și **McCarthy, John**.

interfața om-calculator (*user interface*). Reprezintă ansamblul elementelor fizice reale (*e.g.*, tastatură, mouse, monitor) sau virtuale (*e.g.*, ferestre, meniuri, butoane), precum și a programelor implicate în interacțiunea dintre utilizator și calculator. În definiție sunt incluși trei termeni distincți: omul (în calitate de utilizator), calculatorul (care rulează aplicații, aflate deseori la distanță) și

interacțiunea. Aceasta din urmă are două aspecte, unul fizic și altul informațional, al doilea fiind predominant. În mod obișnuit, interacțiunea fizică (*e.g.*, apăsarea unei taste, execuția unui click cu *mouse*-ul) este orientată către asigurarea interacțiunii informaționale.

În general, interfața cu utilizatorul furnizează modalități de interacțiune între om și calculator, incluse în următoarele două categorii: modalități de intrare (*input*), care îi permit utilizatorului să manipuleze sistemul de calcul, adică interpretează intențiile utilizatorului și le traduc în acțiuni adecvate sistemului; modalități de ieșire (*output*), care permit sistemului să producă efecte în urma manipulării sale de către utilizator, adică filtrează răspunsul sistemului și îi prezintă utilizatorului informații relevante privind rezultatele serviciilor cerute.

Modul este o metodă distinctă de operare în care aceleași informații de intrare pot produce rezultate percepute de utilizator ca fiind diferite în funcție de starea interfeței (programului). Dificultățile de folosire a modurilor determină reducerea utilizabilității interfeței, fiind necesar ca utilizatorul să consume efort pentru reamintirea stării modului curent și pentru a comuta între stări (dacă este necesar). Rezultă că o modalitate definește tipul de date schimbate între utilizator și calculator, în vreme ce modul determină contextul în care aceste date sunt interpretate.

În funcție de modul de interacțiune dintre om și calculator, pot fi

identificate următoarele categorii de interfețe: textual (este utilizată o interfață în linie de comandă, cu un limbaj de comandă, e.g., *shell*-ul Unix); vocal (în limbaj natural sau prin intermediul unui set restrâns de comenzi); cu manipulare directă (interfețe grafice, interfețe cu realitate virtuală); hipertext și hipermedia (textului i se adaugă manipularea prin intermediul legăturilor stabilite între fragmente de informație textuală sau multimedia); prin gesturi ale utilizatorului (captate prin camera video sau scanate).

Unele interfețe sunt sisteme multimedia, prin intermediul lor fiind furnizate informații care – din punctul de vedere al utilizatorului – pot fi reprezentate ca sunete și imagini, statice sau în mișcare.

Interfețele cu realitate virtuală constituie un tip de interfețe cu manipulare directă. Totuși, acestea prezintă caracteristici care le singularizează în raport cu cele din categoria menționată, două dintre ele fiind „scufundarea” multisenzorială a utilizatorului în mediul virtual al interfeței și posibilitatea navigării în interiorul său.

Este de remarcat că există tipuri de interfețe care pot fi incluse în mai multe dintre categoriile enumerate anterior. Astfel, interfețele Web pot fi incluse atât în categoria celor cu manipulare directă (fiind interfețe grafice), cât și în categoria hipertext și hipermedia.

Metodologiile de proiectare a interfețelor (inclusiv a interfețelor Web) își propun crearea unor interfețe

utilizabile, care – altfel spus – pot fi folosite cu ușurință și eficacitate, operațiunile realizate de acestea fiind cât mai ușor de învățat. Termenul „prietenos” (*user friendly*) – referitor la interfețe – este utilizat adesea ca sinonim pentru „utilizabilitate” (*usability*).

Crearea unor interacțiuni complexe, asigurarea dialogului multimodal și obținerea aceluiași funcționalități chiar în situația în care aplicația este accesată prin intermediul dispozitivelor mobile sunt dificil de realizat folosind limbajele HTML și XHTML.

Soluția la problemele enumerate constă în utilizarea unui format universal, independent de platformă și – în funcție de modul de interacțiune cu dispozitivul de redare – în transformarea automată a informațiilor în documente marcate într-unul dintre limbajele implementate de acesta (e.g., HTML, XHTML, WML).

Formatul care îndeplinește cele două criterii este XML (eXtensible Markup Language), acesta permițând implementarea unei meta-interfețe Web, care include descrierea interfeței propriuzise și a interacțiunilor dintre utilizator și aplicație. Concret, pentru proiectarea și dezvoltarea meta-interfețelor este utilizat XUL (eXtensible User-interface Language), un limbaj bazat pe XML și folosit pentru descrierea interfeței browserului Mozilla. A se vedea și **interfață grafică** și **interfață Web**.

interfață activă (*active interface*). Interfață care monitorizează și urmărește interacțiunea sa cu utilizatorul.

interfață cu linie de comandă (CLI, *command line interface*). Tip de interfață dintre utilizator și sistemul de operare în care utilizatorul scrie comenzile folosind un limbaj de comenzi, specific interfeței. Confirmarea unei comenzi și lansarea ei în execuție se face prin apăsarea tastei ENTER. Sistemele bazate pe astfel de interfețe sunt mai dificil de învățat. Totuși, ele le oferă utilizatorilor o flexibilitate mai ridicată în raport cu sistemele care au interfață grafică, deoarece sunt, de obicei, programabile. A se vedea și **linie de comandă**.

interfață grafică (GUI, *Graphical User Interface*). Interfață cu utilizatorul care include următoarele patru elemente: fereastra (*window*), pictograma (*icon*), meniul (*menu*) și dispozitivul de selecție (*pointing device*), folosit pentru manevrarea ferestrei. La această interfață se face referire și prin intermediul acronimului WIMP (*Window, Icon, Menu, Pointing device*), fiind inventată de Douglas Engelbart și de echipa sa de cercetători de la Stanford Research Institute și folosită ulterior de Xerox PARC (PUI, *PARC User Interface*) și Apple MacIntosh.

Microsoft Windows, Mac OS X sau interfețele X WindowSystem sunt GUI deosebit de familiare astăzi. Acestea se bazează pe specificațiile CUA (*Common User Access*), dezvoltate de IBM și Microsoft.

Ferestrele sunt zone de pe ecranul calculatorului, având de cele mai multe ori formă dreptunghiulară, folosite

pentru interacțiunea cu utilizatorul. Acestea pot fi redimensionate, deplasate, minimizate, maximizate, restaurate și închise. Pictogramele reprezintă simboluri asociate acțiunilor utilizatorului (e.g., imprimarea unui text). Meniurile sunt alcătuite din liste de acțiuni (organizate după diverse criterii, în funcție de tipul aplicației) care pot fi realizate de utilizator.

Interfețele grafice pot fi implementate independent de sistemul de operare (în cazul sistemului de ferestre X Window System folosit în UNIX) sau chiar în cadrul nucleului acestuia (în cazul Microsoft Windows). Implementarea unor interfețe grafice se bazează pe fragmente de cod reutilizabil denumite *componente, controale* sau *widget-uri*. Tehnologiile utilizate pentru dezvoltarea lor sunt cunoscute sub numele de *ActiveX* (sau OLE, *Object Linking and Embedding – automation*) în sistemul de operare Microsoft Windows sau *Motif* și *Atena* (de exemplu) pentru X WindowSystem.

În cadrul modelului client-server folosit de X WindowSystem, *managerul de ferestre* este un client care permite lansarea aplicațiilor și manipularea ferestrelor (e.g., KDE, *The K Desktop Environment*; GNOME, *The GNU Network Object Modeling Environment*, ambii utilizați de sistemul de operare Linux). A se vedea și **interfața om-calculator**, **interfață Web**, **manager de ferestre** și **X WindowSystem**.

interfață paralelă (*parallel interface*). Canal (sau port) capabil să transfere simultan mai mult de un bit, folosind fire conectate în paralel. Cele mai multe calculatoare personale *desktop* sunt dotate cu cel puțin un port paralel. Acestea sunt folosite în principal pentru conectarea imprimantelor. Cea mai răspândită interfață paralelă este *Centronics*. A se vedea și **interfață serială**.

interfață serială (*serial interface*). Canal (sau port) capabil să transmită la un moment dat un singur bit. Cele mai multe porturi seriale existente pe calculatoarele personale *desktop* corespund standardelor RS-232C sau RS-422. Un port serial este o interfață de uz general care poate fi utilizată pentru a conecta dispozitive precum modemul sau *mouse*-ul. A se vedea și **interfață paralelă**.

interfață Web (*Web interface*). Componentă a site-urilor și aplicațiilor Web care permite realizarea interacțiunii dintre om și calculator. Deoarece interfețele Web sunt concepute cu scopul declarat de a fi utilizate de oameni, caracteristicile lor sunt practic aceleași cu cele ale interfețelor tradiționale, dintre care pot fi enumerate: funcționalitatea (precizează *ce* realizează interfața); utilizabilitatea (arată *cum* realizează o interfață funcționalitatea pentru care a fost proiectată, adică modul în care răspunde așteptărilor utilizatorilor în ceea ce privește facilitarea muncii acestora); adaptabilitatea (interfața trebuie să

ofere suport pentru interacțiuni multimodale); consistența conținuturilor, structurilor de navigare, a dispunerii spațiale etc. (crearea site-urilor Web consistente este dificilă); separarea interfeței de aplicația propriu-zisă (se știe că funcționalitatea site-ului/aplicației Web – programele/ scripturile care rulează pe server sau client – alternează cu interfața (e.g., elemente XHTML, ceea ce înseamnă că izolarea datelor de logica aplicației este aproape imposibilă); accesibilitatea (persoanele cu dizabilități pot percepe, înțelege, naviga și interacționa cu site-ul/aplicația Web; accesibilitatea este importantă și pentru alte categorii de persoane, inclusiv pentru oamenii în vârstă, dar și pentru persoanele care nu au caracteristicile menționate, deoarece un principiu de bază al acestuia este proiectarea care permite ca site-urile/aplicațiile Web să fie flexibile în raport cu preferințele și nevoile utilizatorilor).

Cele mai multe site-uri/aplicații Web prezintă bariere de accesibilitate care fac dificilă sau chiar imposibilă utilizarea lor de către persoanele cu dizabilități înăscute, dobândite sau datorate vârstei înaintate. A se vedea și **interfață grafică**, **interfață om-calculator**.

Internet. Rețea globală alcătuită din alte rețele, prin intermediul căreia calculatoarele pot comunica folosind suita de protocoale denumită TCP/IP. Fiecare rețea este alcătuită din diverse calculatoare conectate prin diverse modalități (cablu electric, fibre optice,

wireless). Pe la mijlocul anilor '80, colecția de rețele interconectate existentă atunci a fost văzută ca o rețea globală. Nu a existat, totuși, o zi oficială de naștere a Internetului. Tehnologia care stă la baza funcționării acestuia este comutarea de pachete (*packet switching*). Internetul le oferă utilizatorilor o serie de servicii, dintre care pot fi amintite: poșta electronică (*e-mail*), transferul de fișiere (FTP), conectarea la distanță (*telnet*) și World Wide Web (sau Web). A se vedea și **comutare de pachete și TCP/IP**.

Internet2. Organizație non-profit cu sediul în Statele Unite, condusă de o comunitate de cercetători și educatori și având ca membri universități, companii, organizații din domeniul educațional. Această organizație și-a propus dezvoltarea de aplicații și tehnologii de rețea avansate. Proiectul a luat naștere în 1996 sub auspiciile EDUCAUSE și a primit numele actual în 1997. Organizația operează rețeaua cu același nume. Internet2 este o rețea optică bazată pe următoarea generație a protocolului Internet (*i.e.*, IPv6). Ceea ce avantajează această rețea în raport cu Internetul este viteza de transfer a datelor, net superioară, precum și numărul redus de utilizatori. A se vedea și **Internet**.

Internet Society. Prima organizație non-profit neguvernamentală care s-a axat exclusiv pe probleme legate de Internet (ISOC). Fondată în anul 1992 cu scopul de a reprezenta punctul

central la nivel internațional în ceea ce privește cooperarea și coordonarea globală a dezvoltării Internetului, are sedii în Washington D.C și Geneva (<http://www.isoc.org/>).

ISOC reprezintă centrul organizațional al unor organisme care au un rol important în stabilirea standardelor Internet (*e.g.*, IETF). Organizația are două tipuri de membri: persoane fizice și persoane juridice. Astăzi, ISOC are peste 10 000 de membri persoane fizice, în special persoane interesate de dezvoltarea Internetului. De asemenea, peste 150 de organizații reprezentând entități comerciale și non-comerciale (companii, fundații și organisme guvernamentale) au statut de membri ISOC. A se vedea și **IAB** și **IETF**.

Internet banking. A se vedea **online banking**.

Internet Explorer. Abreviat IE, este un browser Web dezvoltat de compania Microsoft și inclus începând cu anul 1995 în sistemele de operare Microsoft Windows. Este cel mai popular browser Web. Au fost dezvoltate versiuni și pentru alte sisteme de operare: *Internet Explorer Mobile* (pentru Windows CE și Windows Mobile), *Internet Explorer* pentru Mac OS și *Internet Explorer* pentru UNIX (Solaris și HP-UX).

InterNIC (*Internet Network Information Center*). Organizație fondată în 1993 care are rolul de a înregistra adresele de tip nume de domeniu și IP, precum și de a distribui informații referitoare la Internet

(<http://www.internic.net/>). Organizează conferințe anuale INET. A se vedea și **INET** și **NIC**.

interogare (*query*). 1. Set de instrucțiuni care permit extragerea unor date dintr-o bază de date. 2. Procesul de extragere a datelor dintr-o bază de date. A se vedea și **bază de date**.

interogări de agregare (*aggregation queries*). Interogări executate frecvent de sistemele de suport decizional. Aceste interogări însumează și grupează mari cantități de date, operatorii de agregare utilizați de obicei fiind **SUM**, **COUNT**, **AVG**. A se vedea și **agregare**.

interpretor (*interpreter*). Program care translatează în format executabil și execută pe rând instrucțiunile unui program scris într-un limbaj interpretat. A se vedea și **limbaj interpretat**.

interpretor de comenzi (*command interpreter*). Program care preia comenzile introduse de la tastatură, le analizează și le execută. A se vedea și **linie de comandă**, **limbaj de comenzi** și **shell**.

intranet (*intranet*). Rețea privată a unei organizații care utilizează tehnologiile Internet. Altfel spus, este o parte a Internetului utilizată exclusiv în cadrul unei companii sau organizații. A se vedea și **extranet** și **Internet**.

intrare (*input*). Informația introdusă într-un program în vederea prelucrării, prin intermediul tastaturii sau al unui fișier stocat pe disc. A se vedea și **program**.

IP (*Internet Protocol*). Funcționând la nivelul rețea în modelul de referință TCP/IP, este un protocol care asigură trimiterea datelor în Internet de la un calculator la altul.

Protocolul IP adaugă datelor primite de la nivelul transport propriu antet și le trimite către mașina-destinație, sub forma unor pachete IP (sau *datagramme IP* sau, încă, *pachete/datagramme internet*). Rolul IP-ului este acela de a găsi o rută (denumită și *cale*) pentru pachete. Pachetele pot fi trimise pe rute diferite, circulând independent până la destinație. Antetul IP conține sursa și destinația (ca adrese IP), numărul de protocol și suma de control (pentru acest nivel). Numărul de protocol comunică mașinii-destinație că trebuie să transmită pachetul nivelului TCP sau UDP. Suma de control dă posibilitatea nivelului IP (*internet*) de pe mașina-destinație să verifice dacă pachetul a ajuns nealterat. Pachetele pot să sosească într-o ordine diferită de cea în care au fost trimise. În acest caz – dacă se dorește –, reordonarea lor cade în sarcina nivelului superior. Versiunea curentă a protocolului este IPv4. Următoarea, în curs de implementare, este IPv6. A se vedea și **TCP/IP**.

IP delivery (*livrare în funcție de adresa IP*). Tehnică *spam* prin care se prezintă un conținut diferit al paginii Web în funcție de adresa IP a clientului. Astfel, *spider*-ul poate accesa un anumit conținut, iar utilizatorii umani alt conținut. A se vedea și **adresă IP**.

Ipe. Editor grafic vectorial *cross-platform*, utilizat pentru crearea figurilor în formatele EPS și PDF (<http://tclab.kaist.ac.kr/ipe/>). Poate procesa codul-sursă LaTeX și importa figuri PDF. A se vedea și **LaTeX** și **PDF**.

iPhone. Telefon inteligent (*smart-phone*), proiectat și dezvoltat de compania americană Apple Inc. și lansat în ianuarie 2007. Utilizează sistemul de operare iPhone OS. Simultan, *iPhone* este și cameră foto/video, media player și client pentru Internet (asigurând acces la Web, la *e-mail* și interactivitate Wi-fi). Telefonul iPhone 3G (iulie 2008) suportă tehnologia 3G, iar iPhone 3GS (iunie 2009) implementează versiunea 3.0 a *iPhone OS*. A se vedea și **Apple, Inc.** și **iPhone OS**.

iPhone OS. Sistem de operare derivat din Mac OS X, dezvoltat de Apple, Inc. pentru două dintre dispozitivele sale, iPhone și iPod Touch. În martie 2009, a fost lansată versiunea 3.0 iPhone OS, suportată de iPhone 3GS. A se vedea și **iPhone** și **iPod**.

iPod. Brand al companiei Apple, cel mai bine vândut media player digital și, totodată, primul dispozitiv pentru care a fost dezvoltat un agregator (de către Adam Curry) în scopul consumării *feed*-urilor Podcast. A fost lansat în octombrie 2001. Linia de producție include iPod Classic (bazat pe hard-disk), iPod Touch (cu *touchscreen*; utilizează sistemul de operare iPhone

OS), iPod Nano (poate rula materiale video) și iPod Shuffle (dispozitiv compact). A se vedea și **Apple**.

IRC (*Internet Relay Chat*). Serviciu Internet care le oferă utilizatorilor posibilitatea de a participa la conversații în timp real. Serviciul (creat în Finlanda, în august 1988, de Jarkko Oikarinen) este proiectat în principal pentru comunicare în grup folosind canale de discuții, dar permite și schimbul de mesaje private. De asemenea, oferă și posibilitatea schimbului de fișiere via protocolul *Direct Client-to-Client*.

Pentru a comunica cu alți utilizatori IRC, este necesar un client IRC pentru conectare la un server IRC. Cel mai popular client pentru Windows este *mIRC*. După realizarea conexiunii, se alege unul sau altul dintre canalele de discuții. Cele mai mari rețele IRC sunt: *QuakeNet*, *Undernet*, *IRCnet*, *EFnet* și *DALnet*.

IRC este „rudă” cu serviciile moderne de mesagerie instantanee (IM, *Instant Messaging*). Dintre acestea pot fi amintite ICQ, *Yahoo! Messenger*, *Windows Live Messenger*, *Skype*, *GoogleTalk*, AIM, *Jabber*, *Meebo*. În vreme ce IRC permite numai comunicarea în mod text, noile servicii oferă posibilitatea utilizării unor pictograme (*emoticons*), incorporând și video conferințe, Voice Over IP (VoIP) și conferințe Web. A se vedea și **chat**.

IrDA (*Infrared Data Association*). Protocol de comunicație pentru

schimbul de date în rețele personale *wireless*, care folosește lumină infraroșie (<http://www.irda.org/>). Permite telefoanelor mobile, calculatoarelor personale, PDA-urilor și altor dispozitive să se conecteze între ele cu diverse scopuri. A se vedea și **PAN**.

IrfanView. Aplicație care poate vizualiza, edita și converti fișiere grafice (având totuși facilități reduse în raport cu programele de editare grafică „clasice”, e.g., *Adobe Photoshop*, *GIMP*); aplicația poate reda, de asemenea, fișiere audio și video (<http://www.irfanview.com/>). *IrfanView* este pusă la dispoziția utilizatorilor începând cu anul 1996, sub dublă licență: proprietară (pentru uz comercial) și gratuită (pentru uz non-comercial).

ISBN (*International Standard Book Number*). Standard utilizat în scopuri comerciale, dezvoltat în Marea Britanie în 1966 și bazat pe standardul EAN-13, prin care fiecărui titlu de carte i se asociază un identificator unic (numit ISBN). A fost adoptat ca standard ISO 2108 în 1970. Un identificator similar, numit ISSN, este utilizat pentru publicații periodice (e.g., reviste).

La data de 1 ianuarie 2005 au început să fie folosite ISBN-urile exprimate cu 13 cifre, proces care s-a finalizat la data de 1 ianuarie 2007. Sistemul obținut este identic cu formatul de coduri de bare EAN.

Fiecare identificator ISBN este alcătuit din 13 cifre grupate în cinci părți: un

prefix EAN (978 sau 979); codul țării în care este publicată cartea sau codul limbii utilizate de aceasta (până la 5 cifre; România are codul 973); codul editorului; un număr asociat cărții; suma de control (*checksum*, reprezentată de o cifră). A se vedea și **EAN-13**.

ISDN (*integrated services digital network*). Set de protocoale de comunicație care permite rețelelor de telefonie să transporte date, voce și video. ISDN a fost proiectat pentru a permite un acces mai rapid prin sistemele de telefonie existente, fără timp suplimentar de configurare a apelului. Suportă rate de transfer de până la 64 kbps. ISDN oferă două tipuri de servicii: **BRI** (*Basic Rate Interface*), care furnizează o lățime de bandă totală de 144 kbps, într-o linie cu trei canale separate (două de tip B, având 64 kbps fiecare, și unul de tip D, cu 16 kbps); **PRI** (*Primary Rate Interface*), cu o rată de transfer de până la 2,048 Mbps (30 de canale de tip B și unul de tip D). Pentru ISDN PRI, ambele tipuri de canale (B și D) au 64 kbps. Canalele B sunt utilizate pentru transmisiile de date (pot include și voce), iar canalele D sunt folosite pentru semnalizare și control (dar nu este exclus să fie folosite și pentru date).

ISO (*International Standard Organization, Organizația internațională de standardizare*). Organizație internațională pentru elaborarea și stabilirea standardelor în toate domeniile de activitate, cu excepția celui electric,

electronicii și telecomunicațiilor. Are peste 150 de membri (țările înscrise în organizație). Standardele ISO sunt numerotate, având un format de tipul „ISO nnnnn:aaaa: Titlu”, unde nnnnn este numărul standardului, aaaa este anul publicării, iar Titlu descrie standardul.

Standardele ISO sunt adoptate, traduse și difuzate în România de ASRO (*Asociația de Standardizare din România*, <http://www.asro.ro/>). Această asociație participă prin specialiști în cadrul comitetelor tehnice internaționale ale ISO.

ISO/OSI (*Open Systems Interconnection, Interconectarea sistemelor deschise*). Model de referință bazat pe o propunere a ISO, este o arhitectură de rețea cu șapte niveluri (model *stivă*): nivelul fizic – asigură serviciile fizice (transmite biții printr-un canal de comunicație) și include specificațiile mediului (cabluri și semnale electromagnetice), conectorii, tehnicile de modulară etc.; nivelul legăturii de date – se ocupă cu adresarea fizică, topologia rețelei, accesul la rețea, sesizarea erorilor, divizarea datelor în cadre (*frames*) și controlul fluxului; nivelul rețea – asigură dirijarea și controlul fluxului de date, fragmentarea și reasamblarea pachetelor, interconectarea rețelelor; nivelul transport – stabilește circuite virtuale între noduri, asigură detecția și recuperarea erorilor și menține nivelurile superioare departe de problemele de implementare, reprezentând un nivel

de graniță între nivelurile-gazdă (ultimele patru în prezentarea curentă, inclusiv acesta) și nivelurile-media (primele trei); nivelul sesiune – stabilește, gestionează și finalizează conexiunile (sesiunile) dintre aplicații locale și la distanță, oferind date nivelului următor (nivelul prezentare); nivelul prezentare – furnizează forma de prezentare (sintaxa) datelor care vor fi transmise (dacă acestea au formate diferite, le convertește într-un format comun, înțeles de ambele părți); nivelul aplicație – nu oferă servicii altor niveluri (fiind în vârful stivei), dar furnizează servicii de rețea aplicațiilor utilizatorilor (e.g., browser, aplicație pentru *e-mail*, emularea terminalului, aplicație pentru transferul fișierelor). La acest nivel se verifică partenerii de comunicație, integritatea și sincronizarea datelor.

Dintre echipamentele folosite la unele dintre nivelurile enumerate, amintim: repetor (nivel fizic), punte (*bridge*; nivel legătură de date), router (nivel rețea) și poartă (*gateway*; nivel rețea sau superior). De menționat că punțile, router-ele și porțile pot fi și componente software.

Dintre avantajele modelului de referință descris (cunoscut și sub numele de modelul ISO/OSI) amintim: este un sistem deschis, folosește protocoale standardizate. Pentru rețelele locale (LAN) se folosește un subset al modelului ISO/OSI. A se vedea și **ISO**, **LAN** și **TCP/IP**.

ISSN (*International Standard Serial Number*). A se vedea **ISBN**.

issuu. Serviciu *online* care convertește fișierele PDF, PowerPoint și Word în cărți electronice (sau în alte tipuri de produse, *e.g.*, cataloage, manuale) în format *Flash* (<http://issuu.com/>).

Fiecare produs creat și găzduit folosind *issuu* are o pagină proprie. Aceasta oferă facilități pentru: partajare (*Share*), printare (*Print*), descărcare (*Download*), introducerea comentariilor (*Comments*), marcarea (*Flag*) și evaluare (*Rate it!*).

Produsul creat poate fi vizualizat și în modul *Fullscreen*. Sunt posibile schimbarea modului de vizualizare (*Magazine view* – implicit, *Presentation view* sau *Paper view*) și obținerea unor informații despre document (*e.g.*, titlu, descriere). Produsul poate fi parcurs pagină cu pagină (înainte și înapoi). De asemenea, sunt posibile afișarea simultană a tuturor paginilor produsului, adăugarea paginii curente într-un director existent sau creat pe loc

în pagina utilizatorului, căutarea unui cuvânt/unei expresii în paginile produsului și saltul la o pagină specificată.

istorie (*history*). Lista acțiunilor pe care le realizează un utilizator într-un program (*e.g.*, browser Web, *shell*-ul sistemului de operare).

iTunes. Aplicație software gratuită lansată în ianuarie 2001 de compania Apple, utilizată pentru organizarea și redarea fișierelor audio și video. Are versiuni pentru platformele Mac OS X și Microsoft Windows. Aplicația este folosită și pentru administrarea conținutului stocat pe iPod-uri. Dacă se dispune de o conexiune Internet, aplicația se poate conecta la magazinul virtual al companiei Apple (*iTunes Store*), permițând descărcarea de melodii, materiale video, podcasturi și filme. *iTunes* este cel mai cunoscut software de tip *podcatching client*. A se vedea și **Apple, Inc.**, **iPod** și **podcatching client**.



înregistrare (*record*). 1. Una dintre cele mai simple structuri de date utilizate în limbajele de programare (e.g., C++). Constă din două sau mai multe valori (ale unor variabile) stocate în poziții consecutive din memorie. Tipul înregistrare este un tip de date care descrie astfel de structuri. Limbajele moderne de programare le permit programatorilor să definească noi tipuri înregistrare. O definiție include specificarea tipului de date al fiecărui câmp, numele acestuia și un identificator al înregistrării. Iată un exemplu care arată cum se definește o înregistrare în C++:

```
struct elev
{ char nume[ 20] , prenume[ 20] ;
  int varsta;
};
```

În exemplul oferit este definită înregistrarea `elev`, aceasta conținând trei variabile, două de tipul `char` (`nume` și `prenume`) și una de tipul `int` (`varsta`). A se vedea și **structură de date** și **tip de date**. 2. În contextul unei baze de date relaționale, reprezintă un set de date corelate inclus într-un tabel. Este cunoscut și sub numele de *rând*. Toate rândurile unui tabel au aceeași

structură. A se vedea și **bază de date relațională**.

în timp real (*real time*). Operații executate de calculator care se desfășoară în condiții restrictive de timp. Reprezintă răspunsul imediat al calculatorului la unele evenimente, fiind caracteristice unor sisteme complexe (e.g., dirijarea traficului aerian, prelucrarea tranzacțiilor).

întreg (*integer*). Tip de date simplu având ca valori numere întregi, negative sau pozitive. Este, alături de tipul real, unul dintre tipurile numerice.

Tipurile întregi uzuale sunt: întreg fără semn (*unsigned int*, ocupă 16 octeți și ia valori cuprinse între 0 și 65 535), întreg scurt (*short int*, ocupă 16 biți și ia valori cuprinse între -32 768 și 32 767), întreg cu semn (*int*, ocupă de regulă 16 biți și ia valori cuprinse între -32 768 și 32 767), întreg lung fără semn (*unsigned long*, ocupă 32 de biți și ia valori cuprinse între 0 și 4 294 967 295) și întreg lung cu semn (*long*, ocupă 32 de biți și ia valori cuprinse între -2 147 483 648 și 2 147 483 647). În C++, tipul caracter este asimilat tipurilor întregi: caracter fără semn (*unsigned char*, ocupă 8 biți și ia valori cuprinse între 0 și 255) și caracter (*char*, ocupă 8 biți și ia valori cuprinse între -128 și 127).

Numerele întregi pot fi specificate în trei formate de citire, care corespund sistemelor de numerație octal, zecimal și hexazecimal. Formatul zecimal este implicit, iar cele octal și hexazecimal

sunt precizate utilizând ca prefix cifra 0 (zero) și, respectiv, secvența 0x (zero urmat imediat de litera x). Orice format poate fi precedat de semnul „-” (minus), care arată că numărul corespunzător este negativ. A se vedea și **real** și **tip de date**.

întrerupere (*interrupt*). Solicitarea intervenției microprocesorului. Reprezintă modalitatea prin care acesta comunică cu celelalte componente ale sistemului de calcul. Când survine o întrerupere (e.g., apăsarea unei taste, primirea unor date pe placa de rețea), microprocesorul poate să oprească temporar o sarcină pe care o execută și să comute pe o alta, ca răspuns la întreruperea respectivă. Mecanismul descris face ca sistemul de calcul să fie foarte flexibil, permițând formularea unui răspuns imediat la un eveniment extern a cărui tratare poate fi foarte urgentă, prin oprirea sarcinii curente și reluarea acesteia după rezolvarea cerințelor impuse de întrerupere. Există o ierarhie a priorităților care stabilește ordinea de tratare atunci când apar mai multe întreruperi simultan. Este posibil ca un program să dezactiveze temporar anumite întreruperi pentru a beneficia exclusiv de activitatea microprocesorului.

Apariția unei întreruperi determină suspendarea programului în curs de execuție și transferul controlului către o rutină de tratare a întreruperii numită *handler de întrerupere*. Mecanismul prin care se face acest transfer este în esență de tip apel de procedură, ceea ce înseamnă că după ce s-a terminat

execuția rutinei de tratare a întreruperii se realizează revenirea în programul întrerupt, din locul în care acesta a rămas.

Întreruperile pot fi hardware și software. Întreruperile hardware sunt interne și externe (dezactivabile și nedezactivabile), iar cele software sunt provocate de sistem sau de utilizator (prin program). Diferența dintre întreruperile hardware și cele software este determinată de entitatea care generează întreruperea. Astfel, dacă aceasta este determinată de un dispozitiv hardware, atunci este o întrerupere hardware, iar dacă este determinată de un program, este o întrerupere software. Într-un sistem pot exista maximum 256 de întreruperi distincte. Fiecare dintre acestea poate avea asociată o procedură numită rutină de tratare. Adresele acestor rutine figurează într-o așa numită tabelă de întreruperi. A se vedea și **handler de întrerupere**, **întrerupere hardware** și **întrerupere software**.

întrerupere hardware (*hardware interrupt*). Întrerupere generată de hardware-ul unui sistem de calcul. Poate fi internă sau externă. Întreruperile hardware interne apar ca urmare a unor situații speciale în care se poate afla procesorul (e.g., accesul la o adresă inexistentă, împărțirea la zero). Întreruperile hardware externe sunt provocate de semnale electrice aplicate, în ultimă instanță, de dispozitivele periferice pe intrările INT și NMI ale microprocesorului.

Înteruperile externe dezactivabile se produc la aplicarea unui semnal pe intrarea INT și sunt luate în considerare numai dacă bistabilul IF (*Interrupt Flag*) din registrul indicatorilor de condiții are valoarea 1. Înteruperile externe nedezactivabile se produc la aplicarea unui semnal corespunzător pe intrarea NMI (*Non Maskable Interrupt*) și sunt luate în considerare întotdeauna (e.g., semnalizarea căderii tensiunii de alimentare). A se vedea și **handler de înteruperere** și **înteruperere**.

înteruperere mascabilă (*maskable interrupt*). Înteruperere hardware care poate fi dezactivată (mascată) temporar de un program. A se vedea și **înteruperere hardware**.

înteruperere nemascabilă (*nonmaskable interrupt*). Înteruperere hardware (cu prioritate mai mare în raport cu înteruperile software) care nu poate fi dezactivată (mascată). Este trimisă microprocesorului numai în situația apariției unor probleme foarte grave (e.g., căderea sursei de alimentare). A se vedea și **înteruperere hardware**.

înteruperere software (*software interrupt*). Înteruperere generată de un program. Este rezultatul instrucțiunilor INT dintr-un program în curs de execuție, care pot fi văzute ca evenimente lansate de programator. Acestea opresc execuția programului și determină apelarea unor rutine de tratare a înteruperii. Rutinele respective sunt incluse de obicei în sistemul de operare. A se vedea și **handler de înteruperere** și **înteruperere**.

întrețesere (*interlacing*). Cu referire la formatele grafice *raster*, este o modalitate de salvare a imaginilor astfel încât acestea vor fi încărcate progresiv (mai întâi o versiune cu detalii de mică importanță, apoi – succesiv – straturile care conțin celelalte detalii). Modalitatea creează iluzia încărcării rapide a imaginilor în browser, dar cel mai important beneficiu al întrețeserii este acela că oferă posibilitatea de a realiza o previzualizare a imaginii de către utilizator (ceea ce determină captarea atenției sale), în timp ce aceasta este încărcată în browser. A se vedea și **formate raster**.

întreținere (*maintenance*). Una dintre etapele dezvoltării unui produs software, în care se pun în practică măsuri necesare pentru funcționarea corectă a acestuia. A se vedea și **ciclu de viață**.

învățarea centrată pe student (*student-centered learning*). Tip de instruire care le oferă celor instruiți o mai mare autonomie și un control sporit cu privire la disciplinele de studiu, metodele de învățare și ritmul de studiu, promovând ideea conform căreia trebuie să li se lase acestora un control sporit asupra învățării prin asumarea de responsabilități cu privire la ceea ce învață, la modul în care învață și la scopul învățării, precum și la momentul în care învață.

În contextul învățării centrate pe student (sau elev), cei instruiți trebuie să își aleagă în mod activ scopurile și să își administreze învățarea. În acest sens, nu se mai pot baza pe faptul că

profesorul le va spune ce, cum și când trebuie să gândească, devenind ei înșiși căutători activi de cunoaștere.

Învățarea centrată pe student este o abordare extinsă care presupune înlocuirea prelegerilor cu învățarea activă, integrarea unor programe de învățare după un ritm propriu și/sau a unor situații de cooperare în grup, care îi oferă studentului responsabilitatea pentru propriul progres în educație.

La baza învățării eficiente centrate pe student stau următoarele principii: în activitatea de învățare accentul trebuie să fie pus pe student (*i.e.*, persoana care învață), și nu pe educator; procesul de predare tradițional este numai unul dintre instrumentele care pot fi utilizate pentru a-i ajuta pe studenți să învețe; rolul profesorului este acela de a dirija procesul de învățare al studenților; într-o măsură destul de importantă, procesul de învățare nu are loc în sala de clasă și nici în prezența cadrului didactic; educatorul trebuie să încurajeze și să faciliteze implicarea activă a studenților în administrarea și planificarea propriului lor proces de învățare, prin proiectarea oportunităților de învățare, atât în sala de clasă, cât și în afara ei; studenți diferiți pot învăța eficient în moduri foarte diferite.

Dintre avantajele învățării centrate pe student pot fi amintite: sprijină învățarea eficientă; determină o retenție îmbunătățită; creează o motivație mai puternică; sprijină integrarea abilităților și competențelor-cheie; oferă o pregătire mai bună pentru exercitarea unei profesii; sprijină

formarea și dezvoltarea abilităților pentru învățarea pe parcursul întregii vieți.

Instruirea bazată pe *e-Learning* potențează într-o măsură importantă învățarea centrată pe student, oferind noi oportunități pentru implementarea acesteia. A se vedea și **e-Learning**.

învățarea colaborativă (*collaborative learning*). Termen-umbrelă pentru o varietate de abordări în educație care implică un efort intelectual comun al studenților sau al educatorilor și studenților. Acest termen se referă la metodologiile și mediile în care studenții se angajează în sarcini comune. Grupele de studenți lucrează împreună, căutând semnificații și soluții sau creând diverse produse.

Pentru a indica munca în echipă, alături de învățarea colaborativă au fost create diverse alte denumiri: învățarea prin cooperare, învățarea colectivă, învățarea comunitară, învățarea reciprocă, învățarea în echipă, studiu circular, studiu de grup etc.

La baza învățării colaborative se află o serie de premise despre cei care învață și despre procesul de învățare. Iată câteva dintre acestea: învățarea este un proces activ de construcție (pentru a achiziționa noi informații, idei sau competențe, studenții trebuie să le utilizeze activ, trebuie să coreleze noile materiale cu ceea ce ei știu deja sau să le utilizeze pentru a reorganiza vechile cunoștințe; prin învățarea colaborativă, studenții nu primesc pur și simplu noi idei sau informații, ci – folosindu-le – creează ceva nou; procesele intelectuale

de construire a semnificațiilor sau a unor idei/produse noi sunt cruciale pentru învățare); învățarea depinde de context (învățarea este puternic influențată de contextul în care se desfășoară; activitățile de învățare colaborativă introduc studenții într-un mediu în care trebuie să facă față unor sarcini provocatoare din punct de vedere intelectual; aceste activități încep frecvent cu probleme pentru care studenții trebuie să prezinte fapte și idei; în loc să rămână observatori indiferenți ai întrebărilor și răspunsurilor sau ai problemelor și soluțiilor, studenții devin practicieni); studenții sunt diferiți (aceștia aduc cu ei în sala de clasă diverse perspective în ceea ce privește pregătirea generală, stilul de învățare, experiența și aspirațiile, care sunt valorificate în cadrul activităților colaborative); învățarea este o activitate socială (învățarea colaborativă produce o sinergie intelectuală pentru indivizi aflați în căutarea soluțiilor unor probleme, precum și angajarea într-un efort comun; explorarea mutuală, crearea de semnificații și *feedback*-ul imediat conduc de cele mai multe ori la o mai bună înțelegere a problemelor pentru fiecare student în parte).

Învățarea colaborativă este bazată pe următoarele principii: interdependența pozitivă (succesul grupului depinde de efortul depus în realizarea sarcinii de toți membrii acestuia; astfel, studenții sunt dirijați către un scop comun, fiind stimulați de o apreciere colectivă); responsabilitatea individuală (fiecare membru al grupului își asumă respon-

sabilitatea sarcinii de rezolvat sau a unei secvențe incluse în ea); formarea și dezvoltarea capacităților sociale, stimularea inteligenței interpersonale (învățarea colaborativă solicită abilitatea studentului de a comunica, de a primi sprijin atunci când are nevoie, de a oferi – la rândul lui – ajutor altor studenți); interacțiunea față în față (această formă de învățare presupune contactul direct cu ceilalți membri ai grupului, atât în clasa tradițională, cât și în cadrul învățării combinate); împărțirea sarcinilor în grup (învățarea colaborativă este o activitate bazată pe principiul egalitarist, care presupune lipsa ierarhiei și autorității).

Tehnologia aleasă pentru susținerea învățării colaborative trebuie să asigure un cadru potrivit pentru desfășurarea unor activități specifice, acesta reprezentând un spațiu adecvat comunicării, colaborării, cercetării și găsirii unor noi resurse. În acest sens, cele mai potrivite instrumente software pentru susținerea învățării colaborative sunt oferite de noul Web (*e.g.*, bloguri, wiki-uri, servicii pentru marcarea colaborativă a resurselor, servicii pentru publicarea și partajarea conținutului). A se vedea și **blog**, **Web 2.0** și **wiki**.

învățarea combinată (*blended learning*). Combinație de abordări multiple ale învățării, prin utilizarea atât a resurselor fizice, cât și a celor virtuale. În sens strict, acest tip de învățare se produce atunci când un educator combină două categorii de activități pentru realizarea instruirii.

Este cunoscută și ca *învățare hibridă* (*hybrid learning*) – termen folosit în special în America de Nord – sau *învățare mixtă* (*mixed learning*).

Un exemplu tipic de învățare combinată este reprezentat de acele cursuri (oferite atât în sistemul oficial de învățământ, cât și persoanelor particulare și angajaților firmelor) care au atât o componentă tradițională/clasică, cât și una bazată pe tehnologia informatică (*e-Learning*).

Învățarea combinată este un rezultat al procesului de convergență a două paradigme ale mediilor de instruire. Una este reprezentată de sistemul clasic, iar cealaltă de mediile distribuite de învățare. Acestea din urmă s-au dezvoltat practic exponențial pe măsură ce noile tehnologii au permis apariția unor noi modalități de comunicare și interacțiune. În trecut, cele două tipuri de medii de instruire au rămas în cea mai mare parte separate, deoarece utilizau combinații diferite de metode și se adresau nevoilor unor audiențe diferite. Astfel, învățarea tradițională este controlată în întregime de educator, suportând interacțiunea de la persoană la persoană, într-un mediu sincron. În schimb, sistemele de instruire la distanță se bazează și, în același timp, stimulează învățarea autoreglată (adaptată la posibilitățile individului), interacțiunea dintre actorii educaționali producându-se în mod obișnuit în medii asincrone.

Combinată cu metodele tradiționale de învățare și substituind unele dintre acestea, învățarea de tip *e-Learning*

permite o nouă relație dintre educator și persoana instruită, colaborarea dintre aceștia extinzându-se dincolo de cadrul restrâns al sălii de clasă. Totodată, i se oferă educatorului posibilitatea de a utiliza noi resurse digitale în scopul creșterii calității activității de instruire.

Contrar unei percepții neavizate, învățarea combinată nu este neapărat o variantă de instruire ieftină, ea necesitând investiții în infrastructură (sau închirierea unor spații existente), în crearea suportului electronic de curs și în modalitatea de realizare a instruirii (*e.g.*, prin intermediul unui site Web).

Se constată că într-o abordare combinată a învățării se ține cont de mai multe elemente: modalitățile de livrare a cursurilor (acestea vor fi susținute atât prin activități față-în-față, cât și prin activități *online*, proporția lor variind de la un curs la altul); locul de desfășurare (în unele situații activitățile *online* se desfășoară într-o încăpere, în prezența unui educator; în alte situații aceste activități pot fi susținute de studenți răspândiți într-un campus universitar; activitățile față-în-față au loc într-o sală de clasă; se pot desfășura simultan atât activități *online*, cât și activități față-în-față, în aceeași încăpere); resursele (în ceea ce privește sursa, se pot utiliza atât resurse *online*, cât și umane; în funcție de forma de prezentare, resursele pot fi electronice sau scrise).

În general, combinarea elementelor se produce la unul dintre următoarele patru niveluri: activitate, curs, program, instituțional. Adesea, combinarea la

nivelul instituțional și la cel al programului este lăsată la latitudinea persoanelor care vor fi instruite, în vreme ce educatorii se implică mai ales la nivelul cursurilor și al activităților.

Astăzi există o serie de motive extrem de puternice pentru care un educator poate opta pentru utilizarea unui sistem de învățare combinată: dezvoltare pedagogică, acces la cunoaștere, interacțiune socială, cost mai redus al instruirii, utilizarea unor materiale educaționale personale, ușurința în revizuirea structurii și conținuturilor. A se vedea și **e-Learning**.

Învățarea contextuală (*contextual learning*). Formă de învățare bazată pe fapte și evenimente reale, realizată în afara sălii de clasă, într-un context specific care determină studentul să utilizeze cunoștințele formale dobândite anterior. Structurile care favorizează învățarea contextuală includ *internship*-ul (angajare temporară), *service-learning* (învățare la locul de muncă) și *study abroad* (studii în străinătate). Dintre teoreticienii care au studiat învățarea contextuală pot fi amintiți John Dewey, Kurt Lewin, David Kolb și Peter Ewell.

Învățare activă (*active learning*). Formă de învățare în care elevii sau studenții nu se rezumă numai la a asculta lecția, ci se și implică activ în cadrul acesteia. Învățarea activă are la bază două ipoteze de bază: învățarea este prin natura sa un efort activ; persoane diferite învață în moduri diferite.

Învățarea individuală (*individual learning*). Formă de instruire bazată pe *e-Learning*, în cadrul căreia studentul se află în contact numai cu materialele didactice. Când o persoană învață individual, corelează informațiile pe care le deține deja cu ceea ce experimentează la momentul respectiv, generând astfel noi cunoștințe. Cunoașterea este deconstrucție și reconstrucție, fiind rezultatul interacțiunii dintre ceea ce este cunoscut deja și ceea ce urmează a fi cunoscut, într-o nouă formă. Deși există diverse definiții ale învățării individuale, toate au la bază ideea că acest tip de învățare se produce atunci când are loc o schimbare în cunoaștere sau în comportament.

Învățarea individuală prezintă o serie de aspecte pozitive. Iată câteva dintre acestea: este potrivită pentru studenții introvertiți sau timizi; este sigură, liberă de presiunea celorlalți studenți sau a educatorului; oferă motivații intrinseci: competiție cu sine însuși, curiozitate, control; dezvoltă autodisciplina; poate fi adaptată propriului stil de învățare; permite studentului să muncească în funcție de propriul său ritm sau de timpul disponibil; oferă satisfacții personale; reprezintă o oportunitate pentru învățarea autentică, bazată pe stilul și preferințele individuale; este lipsită de costuri (deoarece se desfășoară în absența educatorilor).

Dintre aspectele negative ale învățării individuale pot fi amintite: oferă o interacțiune insuficientă cu ceilalți studenți sau cu educatorii; determină

neglijarea unor norme și valori; generează singurătate și poate deveni plictisitoare; solicită foarte multă autodisciplină; studentul poate întâmpina dificultăți în sintetizarea cunoștințelor dobândite; dacă este permanentizată, poate conduce la perspective unilaterale; se concentrează pe succesul și interesele personale, ignorând succesul sau eșecul altor persoane.

Materialele didactice utilizate în învățarea individuală sunt oferite *online*. Este necesar ca acestea să prezinte următoarele caracteristici: să fie interactive; să permită desfășurarea unor activități de monitorizare și *feedback*; să-i ofere studentului sprijinul necesar pentru învățare. A se vedea și **e-Learning**.

Învățarea individuală asistată (*individual learning assisted*). Formă de instruire bazată pe *e-Learning* în cadrul căreia studentul beneficiază de materiale de învățare furnizate *online* și de sprijinul unui îndrumător (denumit adesea tutore). Acesta din urmă poate avea un rol mai mult sau mai puțin important: îl îndrumă pe student numai în situații speciale; îl îndrumă pe student numai în faza finală de evaluare; îl îndrumă și îl sprijină pe student permanent (pe tot parcursul procesului de învățare).

Mediile de învățare și tehnologia utilizate pentru furnizarea acestei forme de instruire trebuie să permită comunicarea prin mijloace electronice între student și îndrumător (*e.g.*, *e-mail*, conferințe Web).

Dintre dezavantajele pe care le implică învățarea individuală asistată pot fi amintite: absența comunicării și interacțiunii cu ceilalți studenți care sunt incluși în același grup/aceeași clasă; poate necesita costuri ridicate, mai ales atunci când relația cu îndrumătorul este permanentă. A se vedea și **e-Learning**.

Învățarea într-o clasă virtuală (*learning in a virtual classroom*). Formă de instruire bazată pe *e-Learning*, realizată într-o clasă virtuală. A se vedea și **clasa virtuală**.

Învățarea mobilă (*mobile learning* sau *m-learning*). Orice formă de învățare care se produce prin intermediul unui dispozitiv mobil (telefon mobil, audio-player, PDA).

O serie de universități au adoptat deja tehnologiile *m-Learning*. De exemplu, unii profesori din Marea Britanie utilizează SMS-ul pentru a transmite cerințele cursurilor, datele de desfășurare și rezultatele unor examene etc. Folosirea telefonului mobil în scopul de a le reaminti studenților unele informații (care sunt/pot fi furnizate și pe alte căi) este considerată o practică pentru ghidarea și managementul învățării și mai puțin o modalitate de dezvoltare a gândirii. Un număr mare de profesori utilizează PDA-urile (*Personal Digital Assistant*) pentru stocarea și regăsirea informațiilor (*e.g.*, *e-books*, cursuri). De asemenea, audio-player-ele (*e.g.*, Apple iPod) sunt folosite de studenți pentru a descărca *podcast*-urile furnizate de educatori. Acestea pot fi ascultate ulterior de câte ori se dorește.

Practica utilizării *podcasting*-urilor pentru audierea cursurilor este în dezvoltare, atât educatorii, cât și studenții fiind atrași de această tehnologie. Posibilitatea ca studenții să asculte de mai multe ori cursurile și, eventual, să noteze unele părți considerate importante este considerat un avantaj foarte important, atât pentru studenți, cât și pentru profesori. Strategia de predare prin transmiterea informațiilor și cea de învățare prin repetiție se potrivesc bine paradigmei behavioriste. A se vedea și **iPod** și **podcast**.

Învățământ la distanță (*distance learning*). Domeniu bazat pe tehnologie și pe proiectarea unor sisteme de instrucție cu scopul de a oferi educație unor studenți care nu se află (din punct de vedere fizic) în același loc. Studenții și educatorii comunică în momentele alese de comun acord, schimbând documente tipărite sau în format electronic sau folosind tehnologii care le permit să comunice în timp real. Cursurile de educație la distanță care solicită prezența fizică a studenților într-un anumit loc și în anumite momente, din diverse motive (*e.g.*, susținerea unor examene), sunt considerate ca fiind de tip mixt (învățare combinată).

Sistemul modern de educație la distanță își are originea în instrucția prin corespondență, organizată în Anglia, în jurul anului 1840, de Sir Isaac Pitman, această formă de educație fiind

încurajată și sprijinită de dezvoltarea serviciilor poștale în secolul al XIX-lea. University of London a fost prima universitate care a oferit cursuri de licență prin intermediul educației la distanță, în anul 1859. În 1873, la Boston, a fost înființată *Societatea pentru încurajarea studiului la domiciliu*. În 1911, în Australia, la University of Queensland, a fost înființat un *Departament pentru studii prin corespondență*. În anul 1939, în Franța, se înființează *Centre national d'enseignement à distance* (CNED). În 1969, în Marea Britanie ia naștere prima universitate deschisă (*Open University*). Astăzi, există în lume un număr mare de astfel de universități, unele devenind megauniversități (acest termen desemnează universitățile cu peste 100.000 de studenți).

Principalele tipuri de cursuri organizate prin intermediul sistemului de educație la distanță sunt: corespondență transmisă regulat folosind *e-mail*-ul; cursuri transmise prin Internet, sincron sau asincron; tele-cursuri/radio-cursuri, acestea fiind livrate prin intermediul unor rețele de televiziune sau radio, în general private; CD-ROM, studentul accesând – prin intermediul unui calculator – cursuri stocate pe CD-uri; învățare mobilă (*Mobile Learning*), prin care studentul accesează cursuri stocate pe dispozitive mobile (*e.g.*, PocketPC) sau pe un server *wireless*; educație la distanță integrată.

J

Jaiku. Serviciu de rețea socială, *microblogging* și *lifestreaming*, cel mai serios concurent pentru *Twitter*, înființat în februarie 2006 în Finlanda și lansat în luna iulie a anului respectiv. (<http://www.jaiku.com/>) A fost achiziționat de Google în octombrie 2007. A se vedea și **microblogging**.

Java. Limbaj de programare imperativă, orientat pe obiecte, compilat, dezvoltat inițial în 1991 de James Gosling la Sun Microsystems și inclus în 1995 ca parte a platformei Java.

Un program Java compilat poate fi rulat fără modificări pe orice platformă (*e.g.*, Windows, Linux, Mac OS, Solaris) pe care este instalată o mașină virtuală Java. Acest lucru este posibil deoarece programele Java sunt compilate într-un format standard denumit cod de octeți (*byte-code*), intermediat între codul-sursă și codul-mașină (specific calculatorului). Mediul în care se execută programele Java este denumit mașină virtuală Java.

Iată câteva dintre caracteristicile limbajului: simplitate (elimină supraîncărcarea operatorilor și moștenirea multiplă); ușurință în crearea de

aplicații complexe; complet orientat pe obiecte; portabilitate; este modelat după C și C++, trecerea de la aceste limbaje la Java făcându-se ușor.

Limbajul Java a fost utilizat pentru dezvoltarea unor tehnologii grupate în câteva platforme de lucru (instrumente de tip SDK), care includ biblioteci și utilitare folosite pentru crearea de aplicații destinate unor categorii specifice de utilizatori: J2SE (Standard Edition – oferă suport pentru dezvoltarea unor aplicații independente și *applet*-uri), J2ME (Micro Edition – oferă suport pentru dezvoltarea de aplicații care vor fi rulate pe dispozitive mobile), J2EE (Enterprise Edition – oferă suportul necesar dezvoltării de aplicații complexe, *e.g.*, aplicații și servicii Web). A se vedea și **byte code**, **JDK**, **mașină virtuală Java**, **SDK** și **Gosling, James**.

JavaScript. Limbaj de scripting creat în scopul extinderii posibilităților browserelor Web adăugându-le interactivitate. Una dintre cele mai importante utilizări ale limbajului constă în scrierea unor scripturi înglobate sau incluse în documentele (X)HTML, care interacționează cu modelul DOM al acestora pentru a realiza diverse sarcini, dintre care pot fi amintite: preprocesarea datelor pe client, înainte ca acestea să fie trimise serverului; schimbarea dinamică a conținutului și a stilurilor utilizate în paginile Web, ca răspuns la acțiunile utilizatorului; distribuirea unor colecții de date de dimensiuni mici și furnizarea

unor interfețe prietenoase pentru utilizarea acestora; controlul navigării în paginile Web alcătuite din cadre multiple; crearea condițiilor pentru ca pagina Web să „reacționeze” la interacțiunea utilizatorului cu câmpurile formularelor. Spre deosebire de alte limbaje (e.g., C++, Java), JavaScript este un limbaj cu tipuri dinamice, ceea ce înseamnă că nu este necesar să se specifice tipul unei variabile atunci când aceasta este declarată și că tipul datelor este convertit automat pe durata execuției scripturilor.

Numit la origine Mocha și apoi LiveScript, JavaScript este un limbaj independent de platformă, creat de Brendan Eich, de la Netscape, și utilizat pentru prima oară în browserul *Navigator 2.0B3* al acestei companii (decembrie 1995). Actualmente, JavaScript este marcă înregistrată Sun Microsystems, Inc.

JavaScript are la bază conceptul de *prototip*. Sintaxa sa este similară cu a limbajului de programare C. Ca și acesta, nu are construcții de intrare sau ieșire. În timp ce limbajul C se bazează pe bibliotecile I/O, motorul JavaScript se bazează pe mediul-gazdă (*host environment*) în care este înglobat. Cel mai cunoscut mediu-gazdă pentru acest limbaj este browserul Web, care creează obiecte responsabile pentru reflectarea DOM în JavaScript.

Limbajul JavaScript extins include: Core JavaScript (nucleul limbajului JavaScript, care conține elementele de bază ale limbajului: operatori, expresii,

instrucțiuni și obiecte predefinite); Client-Side JavaScript (CSJS: extinde limbajul de bază – Core JavaScript –, introducând obiecte utilizate pentru controlul browserului și al modelului obiectului document al acestuia); Server-Side JavaScript (SSJS: extinde limbajul de bază, introducând obiecte relevante pentru utilizarea limbajului pe server, e.g., pentru comunicarea cu baze de date relaționale și manipularea fișierelor). Extensiile CSJS și SSJS depind de nucleul limbajului și nu pot fi utilizate fără acesta.

JavaScript este utilizat pe un număr mare de platforme, inclusiv în afara Web-ului. A fost licențiat de un număr mare de companii pentru includere în instrumentele/tehnologiile create de acestea: *Adobe Acrobat* și *Adobe Reader* suportă JavaScript în fișierele PDF; instrumentele incluse în *Adobe Creative Suite* (inclusiv *Photoshop*) permit *scripting* prin JavaScript; tehnologia Active Scripting (de la Microsoft), cunoscută și ca ActiveX Scripting (bazată pe ActiveX/com), suportă limbajul JScript (compatibil cu JavaScript) ca limbaj de *scripting*. Fiecare dintre aceste instrumente furnizează propriul model DOM, care permite accesul la mediul-gazdă, utilizând de fiecare dată JavaScript pentru a manipula acest model. A se vedea și **DOM** și **ECMAScript**.

JavaScript.com. Site care include un număr mare de resurse utile pentru învățarea limbajului de programare

JavaScript: tutoriale, scripturi, coduri-sursă etc. (<http://www.javascript.com/>).

A se vedea și **JavaScript**.

jDeveloper. IDE *cross-platform*, scris în Java, oferit gratuit de Oracle (<http://www.oracle.com/>). Oferă facilități pentru dezvoltare folosind Java, XML, SQL și PL/SQL, HTML, JavaScript și PHP. A se vedea și **IDE**.

jEdit. Editor de cod-sursă oferit sub licență GPL, scris în Java, cu versiuni pentru cele mai multe platforme (<http://www.jedit.org/>). jEdit suportă evidențierea sintaxei pentru circa 130 de tipuri de fișiere (e.g., JavaScript, Perl, PHP, HTML, XML, XSL). Necesită pentru rulare o mașină virtuală Java. A se vedea și **editor de cod-sursă** și **JVM**.

JDK (*Java Development Kit*). A se vedea **set de instrumente de dezvoltare Java**.

Jobs, Steven Paul (n. 1955). Om de afaceri american, cofondator – împreună cu Steve Wozniak – al companiei Apple Computer (astăzi Apple, Inc.). Împreună cu Wozniak, a avut o contribuție importantă la popularizarea calculatorului personal spre sfârșitul anilor 1970. La începutul anilor 1980, Jobs a fost unul dintre primii care au sesizat succesul comercial pe care îl pot avea interfețele grafice cu utilizatorul. A se vedea și **Apple II**, **Apple, Inc.**, **calculator personal**, **interfață grafică** și **Wozniak, Stephen Gary**.

Joomla. Cadru de lucru (*framework*) *open source* utilizat pentru dezvoltarea unor soluții integrate de comerț electronic, sisteme de control al inventarului, instrumente de raportare a datelor, cataloage personalizate cu produse, sisteme de rezervări etc. (<http://www.joomla.org/>). Este scris în PHP și utilizează SGBDR MySQL pentru stocarea informațiilor. A se vedea și **MySQL** și **PHP**.

Jorum. Depozit național britanic de obiecte de învățare, care are ca scop sprijinirea comunităților pentru educația la nivel universitar și învățare continuă (<http://www.jorum.ac.uk/>). A se vedea și **obiect de învățare**.

Joy, William Nelson (n. 1954). Informatician american, cunoscut și ca Bill Joy, cofondator al Sun Microsystems (1982). A creat familia de editoare de text *vi*, precum și *shell C* (*csh*) pentru BSD. A se vedea și **Sun Microsystems, Inc**.

joystick (*joystick*). Dispozitiv de indicare utilizat de obicei pentru controlul jocurilor video. De asemenea, *joystick*-ul este folosit în aplicațiile de control industriale. Dispozitivul constă din butoane de control și o manetă. Butoanele sunt utilizate pentru a determina producerea unor evenimente pe ecran, iar maneta pivotează pentru a deplasa un obiect pe ecran. A se vedea și **dispozitiv de indicare**.

JPEG (*Joint Photographic Experts Group*). Standard, acronim pentru numele comitetului care l-a elaborat,

scris uneori fără E. JPEG este, alături de GIF, unul dintre cele mai des folosite formate *raster* pentru imaginile utilizate pe Web. Extensiile pentru fișierele în care sunt salvate imaginile în acest format sunt *.jpg*, *.jpeg* sau *.jpe*. Tipul MIME pentru JPEG este `image/jpeg`.

Formatul JPEG a fost proiectat pentru a manipula imaginile *full-color*, cunoscute sub numele de imagini naturale sau imagini fotografice (care oferă suport pentru 16,7 milioane de culori). De asemenea, formatul permite stocarea imaginilor *grayscale*.

Pentru comprimare, JPEG divizează o imagine în blocuri de 8x8 pixeli și apoi utilizează o tehnică numită DCT (*Discrete Cosine Transform*) pentru a calcula un set de coeficienți, fiecare asociat unui bloc. Urmează o aproximare (rotunjire) a acestor coeficienți, în acord cu o tabelă de cuantizare. Ultima etapă determină pierderi de date, dar permite rate mari de comprimare. Rezultă că JPEG este un format cu pierderi, ceea ce înseamnă că imaginile care utilizează acest format nu au aceeași dimensiune ca originalele. În plus, la fiecare salvare – după modificări corespunzătoare – calitatea acestora scade, datorită comprimărilor succesive. Din acest motiv, atunci când se modifică o imagine JPEG, trebuie să se utilizeze originalul!

Gradul de comprimare a imaginilor în format JPEG este variabil. Acest lucru permite celor care creează imaginile

JPEG să aleagă un grad de comprimare care să corespundă calității dorite a imaginilor, dar și unei viteze de descărcare acceptabile. Astfel, pot fi obținute fișiere JPEG mai mici, calitatea imaginilor stocate scăzând în mod proporțional.

De multe ori, în practică, se utilizează un grad de comprimare de 60%, rezultând astfel un optim al balanței dintre calitatea imaginii și mărirea fișierului care o stochează. Formatul JPEG 2000 realizează o comprimare complexă, care include grade diferite de comprimare în cadrul unei imagini date.

Formatul JPEG este utilizat pentru stocarea imaginilor complexe, așa cum sunt, de exemplu, scenele din lumea reală (e.g., figuri umane, animale, diverse peisaje etc.), imagini care nu ar fi plăcute ochiului dacă ar utiliza pentru stocare formatul GIF (care, după cum se știe, suportă cel mult 256 de culori). JPEG suportă întreșerea, dar, spre deosebire de formatul GIF, nu suportă transparența și animația. A se vedea și **formate raster**.

JpGraph. Bibliotecă grafică scrisă în PHP, utilizând metoda programării orientate pe obiect (POO). A se vedea și **programare orientată pe obiect**.

jQuery. Bibliotecă JavaScript *open source* realizată în 2006 de John Resig sub o dublă licență, MIT și GPL (<http://jquery.com/>). A fost dezvoltată în scopul de a facilita procese ca traversarea arborelui DOM,

managementul evenimentelor, crearea unor efecte și animații, manipularea stilurilor CSS, managementul interacțiunii AJAX. A se vedea și **AJAX** și **DOM**.

JSON (*JavaScript Object Notation*). Format textual utilizat pentru serializarea datelor structurate. Sistemele de calcul pot genera/procesa ușor date în acest format. JSON este derivat din elementul literal de tip obiect al limbajului JavaScript, așa cum este el definit în ECMAScript, ediția a 3-a, fiind un format nativ al acestuia. Este utilizat (printre altele), ca și XML, pentru descrierea datelor schimbate între server și client în cadrul aplicațiilor Web bazate pe AJAX.

Reprezentările JSON ale datelor structurate – obținute prin utilizarea unui set de reguli care vor fi prezentate în această secțiune – sunt simple șiruri de caractere care pot fi convertite ușor în tablouri de obiecte JavaScript. Acest format este des utilizat pentru scrierea argumentelor funcțiilor, a valorilor returnate de acestea, a răspunsurilor serverului etc. Tipul MIME pentru JSON este `application/json`.

Cu toate că formatul JSON este complet independent de limbaj, utilizează convenții familiare programatorilor, întâlnite într-un număr mare de limbaje (e.g., C, C++, C#, Java, JavaScript, Perl, Tcl). Astfel, formatul amintit are la bază două structuri universale de date, implementate toate în limbajele de programare actuale: o colecție de perechi nume–valoare

(implementată ca obiect, structură, înregistrare, tablou asociativ etc.); o listă ordonată de valori (implementată ca tablou numeric, listă, vector etc.). În JSON, structurile enumerate anterior iau următoarele forme: 1) obiect (*object*) – este un set neordonat de perechi de forma nume–valoare, inclus între acolade; numele (de tip șir) este urmat de caracterul două puncte („:”) și de valoarea sa, iar perechile sunt separate între ele prin caracterul virgulă („,”); pot fi introduse spații albe între oricare două perechi succesive; 2) tablou (*array*) – reprezintă o colecție ordonată de valori separate prin virgulă, plasată între paranteze drepte; 3) valoare (*value*) – poate fi un șir (inclus între ghilimele duble), un număr, un tablou, un obiect sau valorile `true`, `false`, `null`; structurile menționate pot fi imbricate; 4) șir (*string*): este o colecție plasată între ghilimele duble, alcătuită din niciunul, unul sau mai multe caractere Unicode, cu excepția caracterelor „” (ghilimele duble), „\” (*slash*) și a caracterelor de control; caracterele exceptate pot fi utilizate numai în secvențe *escape* (e.g., „\”, „\\”, „\n”); 5) număr (*number*) – este asemănător cu tipul similar utilizat în limbajul JavaScript (dar formatele octal și hexazecimal nu sunt folosite).

După cum se poate observa în enumerarea anterioară, utilizând JSON pot fi reprezentate patru tipuri de date simple (șiruri, numere, valori *booleene* și tipul `null`) și două structurate (obiecte și tablouri). Este de remarcat că JSON nu poate reprezenta funcții sau

expresii, fiind un format orientat exclusiv pe date.

Iată o comparație între JSON și XML: utilizare pentru transfer de date http – atât XML, cât și JSON sunt formate care descriu un obiect sub formă de text simplu, fiind potrivite pentru transferul datelor prin HTTP (e.g., intrări și ieșiri de date pentru serviciile Web); asigurarea suportului în limbajele de programare – amândouă formatele sunt suportate prin intermediul a numeroase biblioteci, scrise în diverse limbaje de programare (inclusiv JavaScript), care transformă un obiect nativ în aceste formate și invers; simplitate – XML este un format relativ simplu, dar JSON este și mai simplu, deoarece utilizează o gramatică mult mai redusă, care, în plus, se potrivește mai bine cu structurile de date utilizate în multe limbaje de programare; extensibilitate – XML este extensibil, dar JSON nu este (această caracteristică nu este neapărat un dezavantaj pentru JSON, deoarece acesta nu este un limbaj de marcare); interoperabilitate – JSON are aceeași potențială interoperabilitate ca și XML; deschidere – JSON este cel puțin la fel de deschis ca și XML, unul dintre motive fiind acela că, deocamdată, nu este obiectul unor politici de standardizare; orientare – XML este orientat pe document, iar JSON este orientat pe date; ușurință în utilizare – JSON este mai ușor de citit și de scris, atât de oameni, cât și de sistemele de calcul; ușurință în procesare – JSON este mult mai simplu de procesat decât XML; structuri de date utilizate –

structurile de date JSON sunt bazate pe tablouri și înregistrări, în timp ce structurile de date XML sunt bazate pe elemente, atribute, conținut textual, entități, DTD-uri și alte metastructuri; dimensiunea datelor – în mod obișnuit, reprezentarea JSON a unei date are dimensiuni mai mici decât reprezentarea XML a aceleiași date; autodescriere a datelor – atât JSON, cât și XML prezintă această caracteristică; internaționalizare – atât JSON, cât și XML suportă standardul Unicode; tipuri de date utilizate – documentele XML pot conține orice tip de date, inclusiv componente ActiveX și *applet*-uri Java, caracteristică văzută uneori nu ca un avantaj, ci ca un potențial pericol pentru securitatea sistemului (deoarece JSON nu utilizează secțiuni `<[CDATA[]>`, nu poate transporta imagini, sunete sau alte date binare); familiaritate – XML este mult mai familiar comunității IT decât JSON; suport în browser – browserele moderne asigură suport aproape complet pentru XML (DOM, XPath și XSLT), dar și pentru JSON, acesta fiind un format bazat pe JavaScript; compatibilitate cu alte standarde – JSON este compatibil cu YAML (o altă alternativă la XML).

Cu toate că JSON este considerat adesea un concurent al formatului XML, ele sunt utilizate de multe ori în aceeași aplicație sau ca formate complementare. Un număr mare de limbaje de programare oferă suport pentru formatul JSON. Dintre acestea, pot fi amintite ActionScript, C, C#,

Java, JavaScript, Perl, PHP, Python și Ruby. Este de remarcat că serviciile Web oferite de Yahoo! suportă acest format. A se vedea și **XML**.

json. Extensie PHP PECL utilizată pentru codificarea valorilor PHP în obiecte JSON, precum și pentru decodificarea obiectelor JSON în valori PHP (<http://pecl.php.net/package/json>). A se vedea și **JSON**, **PHP** și **PECL**.

JSP (*JavaServer Pages*). Platformă de programare utilizată pentru crearea paginilor Web dinamice, bazată pe limbajul Java. Paginile JSP sunt fișiere text în care sunt incluse cod HTML și Java. Acestea sunt accesibile printr-un server de aplicații, care primește cereri

HTTP de la un browser prin intermediul serverului Web. Serverul de aplicații include și o mașină virtuală Java în care rulează atât codul Java inclus în paginile JSP, cât și obiectele instanțiate de acesta. A se vedea și **HTML** și **Java**.

JSVectorEditor. Editor grafic SVG *open source online*, bazat pe JavaScript, care rulează în browserele *Firefox*, *IE*, *Safari*, *Chrome* și *Opera* (<http://code.google.com/p/jsvectoreditor/>). Este furnizat sub licență MIT. A se vedea și **licența MIT** și **SVG**.

JVM (*Java Virtual Machine*). A se vedea **mașină virtuală Java**.

K

Kahn, Robert Elliot (n. 1938). Informatician american, cunoscut sub numele de Bob Kahn. A creat – împreună cu Vinton Cerf – suita de protocoale TCP/IP. Cei doi au primit premiul Turing și medalia Alexander Graham Bell. A se vedea și **Cerf**, **Vinton Gray**.

Karbon14. Editor vectorial pentru Linux (<http://www.koffice.org/karbon/>). *Karbon14* utilizează formatul nativ SVG și este inclus în *KOffice*, suita de programe de birou pentru KDE. A se vedea și **KDE**, **KOffice** și **SVG**.

KartOO. Motor de căutare care utilizează o interfață vizuală (<http://www.kartoo.com/>). A se vedea și **motor de căutare**.

kbps (sau *kbit/s*, *kb/s*). Prescurtare de la *kilobits per second* (kilobiți pe secundă). Reprezintă o mie de biți pe secundă (*i.e.*, 10^3 bps). A se vedea și **bit rate**.

KDE (*K Desktop Environment*). Mediu *desktop* (*desktop environment*) *open source* pentru sisteme compatibile UNIX, dezvoltat sub licență GPL începând cu anul 1996 de o echipă coordonată de Matthias Ettrich

(<http://www.kde.org/>). Este mediul *desktop* standard pentru numeroase distribuții Linux (*e.g.*, Mandriva, Kubuntu, SUSE), disponibil și în limba română. A se vedea și **GNOME**, **Licența Publică Generală GNU** și **Linux**.

KDevelop. IDE gratuit pentru mediul *desktop* KDE, oferit sub licență GPL (<http://www.kdevelop.org/>). Suportă peste 30 de limbaje de programare (*e.g.*, C, C++, Java, Pascal, Perl, PHP, Python, Ruby). A se vedea și **IDE** și **KDE**.

Kerberos. Protocol de autentificare în rețea elaborat de MIT în cadrul proiectului Athena, în anul 1984 (<http://web.mit.edu/kerberos/www/>). Are rolul de a permite unui client să-și demonstreze identitatea unui server aflat la distanță, într-o rețea complet nesigură. De asemenea, Kerberos îi garantează clientului că nu conversează cu un calculator care se dă drept server, deoarece autentificarea se face în ambele direcții. Protocolul indică o serie de mesaje care trebuie schimbate între părțile care doresc să comunice, unele dintre acestea fiind criptate. Implementările curente folosesc un algoritm standard de criptare, numit DES (*Data Encryption Standard*). MIT oferă o implementare gratuită a protocolului, dar există și un număr mare de produse comerciale. A se vedea și **DES**.

kernel (*nucleu*). Componenta centrală a unui sistem de operare, având ca

responsabilitate gestionarea (alocarea) resurselor sistemului de calcul și comunicarea cu acestea, pentru a permite și altor programe să ruleze și să le utilizeze. Resursele respective sunt unitatea centrală de procesare (CPU), memoria calculatorului și dispozitivele de intrare/ieșire (e.g., tastatură, mouse, unitățile de disc). De asemenea, nucleul furnizează metode pentru managementul proceselor (i.e., lansarea și planifi-carea acestora, precum și comunicarea între ele). A se vedea și **sistem de operare**.

Kernighan, Brian (n. 1942). Programator canadian, a avut o contribuție importantă la dezvoltarea sistemului de operare UNIX și a creat numeroase programe UNIX (e.g., *cron*). Împreună cu D. Ritchie, a scris cartea *The C Programming Language*, cea mai cunoscută carte de programare (<http://cm.bell-labs.com/cm/cs/cbook/>). A se vedea și **C**, **UNIX** și **Ritchie, Dennis**.

keylogger (*înregistrator de taste*). Program care înregistrează tastele apășate de utilizator, folosit de hackeri pentru a obține informații confidențiale (e.g., date de înregistrare, parole, numere de carduri, PIN-uri etc.). Troienii *backdoor* au integrat – de multe ori – un asemenea program. A se vedea și **hacker**, **spyware** și **troian**.

keypad (*minitastatură*). Set de butoane (taste) aranjate într-un bloc, care includ, de obicei, cifre și alte simboluri, dar nu și un set complet de caractere

alfabetice. În cazul în care cea mai mare parte a butoanelor conțin numere, aceasta este o tastatură numerică.

keyword density (*densitatea cuvântului-cheie*). Se referă la numărul de repetări ale unei expresii-cheie în corpul paginii Web. Motoarele de căutare vor poziționa mai bine în rezultatele returnate o pagină cu o valoare ridicată a *keyword density* pentru o anumită expresie-cheie. A se vedea și **pagină Web**.

Kilby, Jack (1923–2005). Fizician american, laureat al premiului Nobel în anul 2000, pentru inventarea circuitului integrat (în 1958, în timp ce lucra la Texas Instruments). De asemenea, este inventatorul imprimantei termice și al calculatorului de buzunar. A se vedea și **circuit integrat** și **Noyce, Robert Norton**.

kilobyte. Notat kB sau KB și denumit *kilooctet* în limba română, are 2^{10} bytes (sau octeți), adică 1.024 bytes. A se vedea și **byte**.

Kleinrock, Leonard (n. 1934). Inginer și informatician american, profesor la UCLA (University of California, Los Angeles). A avut contribuții majore la dezvoltarea rețelelor de calculatoare, inclusiv la teoria acestora, fiind unul dintre inventatorii tehnicii comutării de pachete. A jucat un rol important în dezvoltarea ARPANET la UCLA. Pentru realizările sale, Kleinrock a primit numeroase premii. În 2008 a fost selectat pentru a primi prestigioasa *National Medal of Science*, cea mai

înalță distincție științifică americană. A se vedea și **ARPANET**.

KML (*Keyhole Markup Language*). Limbaj bazat pe XML, dezvoltat de *Google* și utilizat pentru exprimarea datelor geografice în aplicațiile *Google Earth* și *Google Maps*. A se vedea și **XML**.

Knuth, Donald Erwin (n. 1938). Profesor emerit la Universitatea Stanford, autorul cărții *The Art of Computer Programming* (<http://www-cs-faculty.stanford.edu/~knuth/>). Este creatorul sistemului de culegere a documentelor TeX. Activitatea sa a fost recompensată cu numeroase premii, printre care medalia John von Neumann, premiul Turing, Medalia națională în științe și premiul Kyoto. A se vedea și **TeX**.

KOffice. Suită de programe de birou *open source* pentru KDE (<http://www.koffice.org/>). Utilizează

formatul deschis ODF (*OpenDocument Format*). A se vedea și **KDE** și **OpenDocument Format**.

Korn shell (*shell-ul Korn*). Shell UNIX dezvoltat de David Korn la începutul anilor 1980. Este compatibil cu *shell-ul Bourne* și include multe caracteristici ale *shell-ului C*. A se vedea și **Bourne shell**.

Krugle. Lansat în iunie 2006, este un motor de căutare care le oferă programatorilor și dezvoltatorilor posibilitatea de a căuta coduri *open source* (<http://www.krugle.com/>). A se vedea și **motor de căutare**.

Kurzweil, Raymond (n. 1948). Inventator american, unul dintre cei mai importanți experți în domeniul inteligenței artificiale, pionier în domeniul recunoașterii optice a caracterelor, al sintezei vocale (<http://www.kurzweiltech.com/>). A se vedea și **inteligență artificială**.

L

LAN (*Local Area Network*). Rețea de calculatoare sau alte dispozitive răspândite pe o suprafață relativ limitată și conectate astfel încât un sistem poate interacționa cu oricare altul inclus în rețea.

Aceste rețele includ, alături de calculatoare, și resurse partajate (*e.g.*, imprimante, hard-discuri de capacitate mare). Dispozitivele care alcătuiesc o rețea locală – denumite noduri – sunt legate între ele prin cabluri utilizate pentru transmiterea mesajelor. A se vedea și **LAN fără fir și rețea de calculatoare**.

LAN fără fir (*WLAN, wireless LAN*). Rețea locală (LAN) fără fir în care comunicația se realizează prin intermediul undelor radio, ceea ce le oferă utilizatorilor posibilitatea de a se deplasa împreună cu dispozitivele, acestea fiind conectate în permanență. Componentele unui WLAN se numesc *stații*, acestea putând fi puncte de acces (*e.g.*, routere) și clienți (*e.g.*, laptopuri, PDA-uri, telefoane IP, dispozitive fixe). Toate stațiile sunt echipate cu adaptoare *wireless*. A se vedea și **adaptor wireless, LAN, punct de acces fără fir și router**.

(a) **lansa** (*launch*). A determina punerea în execuție a unui program, în special prin intermediul interfeței grafice cu utilizatorul. A se vedea și **GUI**.

landing page (*pagină de primire*). Termen folosit în *marketing-ul online*, desemnează pagina la care ajunge un utilizator după ce execută click pe un banner publicitar, pe una dintre legăturile incluse în rezultatele returnate de un motor de căutare, pe o reclamă contextuală, pe o legătură dintr-un mesaj de *e-mail* etc. Astfel de pagini – extrem de importante pentru companiile de comerț electronic – sunt de tip referință și tranzacțional. Paginile de tip referință prezintă informații relevante pentru utilizator și date despre compania căreia îi aparține site-ul: text, imagini, legături etc. În schimb, o pagină de tip tranzacțional include un formular pe care vizitatorul este invitat să-l completeze sau un obiect cu care acesta va interacționa, scopul fiind eventuala vânzare a unui produs sau serviciu. Eficacitatea unei pagini de acest tip este măsurată prin rata de conversie, aceasta reprezentând procentul vizitatorilor care realizează acțiunea dorită.

LAN virtual (*virtual LAN*). Rețea locală (LAN) alcătuită din grupuri de gazde aflate pe segmente diferite, dar care comunică între ele ca și cum ar fi conectate fizic prin cablu. A se vedea și **LAN**.

latență (*latency*). Într-o rețea cu comutare de pachete, reprezintă timpul

necesar unui pachet pentru a se propaga dintr-un punct în altul al acesteia. Se măsoară în două moduri: timpul necesar propagării într-un singur sens (*one-way*, i.e., de la sursă la destinație) și, respectiv, timpul necesar propagării dus-întors (*round-trip*, i.e., de la sursă la destinație și înapoi).

LaTeX. Sistem de programare a documentelor bazat pe TeX, creat de Leslie Lamport în 1984. A devenit principala metodă pentru programarea în TeX, permițând precizarea aspectelor utilizate în *desktop publishing* (imagini, tabele, figuri, referințe încrucișate, note, bibliografie). Sistemul LaTeX îi permite utilizatorului să se ocupe mai mult de conținutul documentului decât de aspectul său. Este folosit în special în mediul academic, dar și în mediul comercial, datorită costurilor reduse de utilizare (LaTeX și TeX sunt gratuite). A se vedea și **TeX**.

LCMS (*Learning Content Management System*, sistem pentru managementul conținutului învățării). Software utilizat pentru managementul și publicarea conținutului care, în mod obișnuit, va fi furnizat prin intermediul unui LMS (*Learning Management System*).

Un LCMS permite – prin integrarea funcționalităților unei baze de date – stocarea, gestionarea și reutilizarea conținutului educațional, fiind o aplicație complexă care etichetează obiectele învățării, le organizează și le furnizează într-un număr mare de combinații.

LCMS-urile pot include instrumente pentru crearea conținutului, în special pentru realizarea site-urilor Web, precum și editoare media. Aceste instrumente pot furniza cursuri către un LMS și pot urmări studenții și cursurile la care aceștia sunt înscriși. Într-un LCMS, cursurile pot fi accesate și se poate naviga în interiorul acestora folosind un browser Web. Dacă un LCMS nu poate efectua și operații de testare, poate livra totuși teste create și administrate de un instrument pentru crearea testelor.

Un LCMS combină funcțiile de gestionare și administrative ale unui LMS cu cele de creare a conținutului și de asamblare personalizată. Acestea din urmă sunt caracteristice unui sistem de management al conținutului (CMS, *Content Management System*). În fond, LCMS-urile sunt CMS-uri în care sunt luate în considerare parțial unele aspecte psihopedagogice ale procesului de învățare și care implementează un set de instrumente pentru gestionarea acestuia.

Dintre caracteristicile LCMS-urilor pot fi amintite: oferă posibilitatea creării/editării conținutului; pot include articole de învățare dezvoltate extern; oferă posibilitatea dezvoltării unor teste de evaluare; asigură suport pentru publicarea CD-urilor, tipăriturilor, prezentărilor și pentru publicarea pe Web; le oferă utilizatorilor posibilitatea auto-înregistrării la cursuri *online*; permite urmărirea evoluției studenților; oferă posibilitatea grupării cursurilor și

studentilor, în orice moment; urmăresc accesarea cursurilor la nivelul modulelor individuale sau al obiectelor de învățare; asigură instrumente pentru administrare.

Un LCMS are în structură următoarele componente principale: o interfață pentru lansarea conținutului; un instrument de creare a conținutului; o componentă pentru înregistrarea studenților, lansarea cursurilor și urmărirea progresului; un depozit (bază de date) pentru obiectele de învățare.

Deoarece un LCMS este utilizat în mod obișnuit împreună cu un LMS, este necesară stocarea cursurilor în formate standardizate (SCORM sau IMS Content Packaging), care să permită folosirea acestora.

O organizație educațională care dorește să implementeze un sistem de *e-Learning* poate să utilizeze o soluție comercială (având de ales dintre numeroasele soluții de acest tip existente pe piață) sau una *open source* (în acest caz trebuie avută în vedere și necesitatea/posibilitatea de a achiziționa servicii și programe comerciale, asociate cu platformele educaționale gratuite). De asemenea, numeroase universități sunt implicate astăzi în dezvoltarea unor soluții *open source* proprii, sub licență GPL. A se vedea și **IMS Content Packaging, LMS, obiect de învățare și SCORM.**

LDAP (*Lightweight Directory Access Protocol*). Protocol de rețea TCP/IP utilizat pentru interogarea și actualizarea serviciilor de director. Un

director este un set de obiecte cu atribute similare organizate logic și ierarhic (e.g., clasic: cartea de telefoane).

legătură (*link*). A se vedea **hiperlegătură**.

legătură de comunicații (*communications link*). Termen utilizat în rețelistică pentru a desemna conexiunea dintre două sau mai multe calculatoare care permite transferul datelor. A se vedea și **rețelistică**.

legea lui Moore (*Moore's law*). Descrie o tendință importantă în domeniul hardware-ului: numărul tranzistoarelor care pot fi plasate ieftin într-un circuit integrat crește rapid, dublându-se aproximativ la fiecare 2 ani. Observația a fost făcută în 1965 de Gordon Moore, chimist și fizician, cofondator al Intel Corporation. Tendința a continuat mai mult de o jumătate de secol și se așteaptă să continue cel puțin încă un deceniu sau mai mult. A se vedea și **Moore, Gordon Earle**.

legendă (*legend*). Text care explică sau descrie structura și/sau conținutul unui grafic, ale unei hărți sau ale unui document.

Lempel-Ziv. A se vedea **algoritmul Lempel-Ziv**.

library (*biblioteca*). A se vedea **biblioteca**.

Libre Software. Termen utilizat pentru prima oară în anul 2000, în documentele Comisiei Europene, pentru a

desemna software-ul oferit fără costuri. Spre deosebire de *free* (folosit în expresia *Free Software*), cuvântul *libre* (împrumutat din limbile franceză și spaniolă) nu permite confuzia între libertatea de utilizare și gratuitate. A se vedea și **FLOSS**, **Free Software** și **Open Source Software**.

libxml. Bibliotecă gratuită de procesare a documentelor XML, implementată în C pentru întreg modelul DOM, scrisă de Daniel Veillard, de la Consorțiul Web (<http://xmlsoft.org/>). Permite asocierea și procesarea spațiilor de nume XML, reguli de validare a documentelor etc. Biblioteca este inclusă și în mediul de dezvoltare GNOME. Versiunea *libxml2* include implementări complete pentru XPath, XPointer și XInclude, fiind utilizată pe mai multe platforme (e.g., UNIX/Linux, Windows, MacOS X și OS/2) și putând fi legată la diverse limbaje de programare (e.g., C, C++, C#, Python, Delphi, Pascal, Ruby, PHP 5). A se vedea și **DOM**.

libxslt. Bibliotecă XSLT scrisă în C, bazată pe *libxml2*, dezvoltată sub licență MIT pentru proiectul GNOME (<http://xmlsoft.org/>). Este procesorul XSLT standard pentru diverse limbaje de programare (e.g., Perl, PHP). A se vedea și **GNOME**, **libxml** și **XSLT**.

licența Apache (*AL*, *Apache License*). Licență pentru software creată de ASF (*Apache Software Foundation*). Versiunile AL (1.0, 1.1 și 2.0) solicită păstrarea informațiilor legate de

copyright și a exonerării de răspundere, dar nu este o licență *copyleft*, adică permite utilizarea codului-sursă pentru dezvoltarea de software proprietar sau *open source*. A se vedea și **Apache Software Foundation** și **copyleft**.

licența BSD (*BSD license*). Una dintre cele mai des utilizate licențe pentru software-ul liber (*free software*). Denumirea sa constituie acronimul pentru *Berkeley Software Distribution*, un sistem de operare derivat din UNIX (distribuit de University of California, Berkeley) pentru care a fost creată, inițial, licența respectivă.

licența LGP (*LGPL*, *Lesser General Public License*). Licență *free software* publicată de FSF (*Free Software Foundation*), proiectată cu scopul de a reprezenta un compromis între licența *copyleft* GNU GPL (*GNU General Public License*) și alte licențe mai permissive (e.g., licențele BSD și MIT). LGPL a fost scrisă de Richard Stallman în 1991 și actualizată în 1999 și 2007.

LGPL este utilizată în special pentru bibliotecile software, dar și pentru o serie de aplicații de sine stătătoare (e.g., *Mozilla*, *OpenOffice.org*). A se vedea și **copyleft**.

licența MIT (*MIT License*). Licență *free software* elaborată la MIT (Massachusetts Institute of Technology). Este o licență permisivă, ceea ce înseamnă că acceptă reutilizarea software-ului liber în cel proprietar, cu condiția distribuirii licenței împreună cu cel din urmă. Iată câteva dintre

aplicațiile și bibliotecile cunoscute care utilizează licența MIT: *CakePHP*, *jQuery*, *libxml2*, *Paint.NET*, *Prototype*, *Symfony*.

licența Open Software (OSL, *Open Software License*). Licență software creată de Lawrence Rosen. Conform OSI (*Open Source Initiative*), OSL este o licență *open source*. De asemenea, OSL este și o licență *copyleft*. A se vedea și **copyleft**.

Licența Publică Generală GNU (*GNU GPL*, *GNU General Public License*). Având acronimul GPL, este o licență software de tip *copyleft*, elaborată de Fundația pentru Software Deschis (*Open Software Foundation*). Oferă oricărui utilizator dreptul de a copia, modifica și redistribui programe și codurile-sursă ale acestora, provenind de la programatorii care își licențiază programele folosind GPL. Prima versiune a acestei licențe a fost scrisă de Richard Stallman. Versiunea GPLv3 a fost publicată pe 27 iunie 2007. A se vedea și **copyleft**.

licențe Creative Commons (*Creative Commons Licenses*). Cunoscute și ca licențe CC (*CC licenses*), permit schimbarea termenilor *copyright*-ului, înlocuind clasicul *All rights reserved* (toate drepturile rezervate) cu *Some rights reserved* (unele drepturi rezervate). Ținta licențelor CC este de a lărgi domeniul de creații disponibile tuturor în mod legal, atât pentru utilizare, cât și ca părți ale unor noi creații. Sub licența CC pot fi distribuite text, muzică,

video, imagini etc. Licențele CC sunt utilizate de un număr foarte mare de site-uri, deși nu toate materialele postate pe acestea sunt disponibile sub astfel de licențe. Iată câteva dintre site-urile respective: *Flickr*, *Internet Archive*, *OpenCourseWare*, *Public Library of Science*, *Wikimedia Commons*. De asemenea, există proiecte naționale care sprijină publicarea operelor sub licențe *Creative Commons*. Dintre acestea pot fi amintite *Images for the future* (proiect guvernamental olandez susținut cu 137 de milioane de euro, pentru a încărca în Web filme, emisiuni TV și înregistrări radio, precum și fotografiile din arhivele olandeze) și *Overmundo* (site colaborativ pentru promovarea culturii braziliene).

Licențele *Creative Commons* variază de la o libertate aproape totală până la permisiuni foarte limitate. Aceste permisiuni incluse în licențele CC sunt: atribuire (attribution – autorul le oferă utilizatorilor posibilitatea de a copia, distribui, afișa și interpreta o operă, inclusiv o operă derivată, cu condiția de a menționa autorul); necomercial (*noncommercial* – autorul le oferă utilizatorilor posibilitatea de a copia, distribui, afișa și interpreta o operă, dar și operele derivate, exclusiv în scopuri necomerciale); fără opere derivate (*no derivative works* – autorul le oferă utilizatorilor posibilitatea de a copia, distribui, afișa și interpreta o operă, fără a crea opere derivate pe baza acesteia); distribuire în condiții identice (*share alike* – autorul le oferă utilizatorilor

posibilitatea de a distribui opere derivate numai sub o licență identică cu cea a operei originale).

Pentru orice refolosire sau distribuție, utilizatorul trebuie să prezinte termenii licenței pentru opera respectivă, cea mai bună modalitate de a realiza acest lucru fiind de a plasa un link spre pagina CC dedicată licenței. Dacă se obține permisiunea din partea titularului drepturilor de autor, se poate renunța la oricare din condițiile impuse de licență. Linkurile pot fi incluse prin intermediul unor pictograme furnizate de *Creative Commons* (prezentate anterior, pentru fiecare licență), această organizație oferind totodată și codul XHTML necesar. Fiecare pictogramă include elemente grafice care precizează condițiile specificate în licență.

Site-ul *Creative Commons* pune la dispoziție un formular (*Choose a License*) care îi permite autorului să aleagă cu ușurință licența optimă sub care dorește să-și publice opera (răspunzând la câteva întrebări), un logo de identificare și un text descriptiv.

Licențele *Creative Commons* au fost scrise ținând seama de sistemul legal din Statele Unite. Astfel, există posibilitatea ca, folosind modelul SUA indiferent de legile locale, licențele CC să devină neaplicabile. Proiectul CCI (*Creative Commons International*) a fost creat pentru a evita astfel de probleme, scopul acestuia fiind de a ajusta formularea licențelor CC în acord cu legislația din fiecare țară. În

data de 2 septembrie 2008, la Centrul pentru Jurnalism Independent din București, a avut loc lansarea licențelor *Creative Commons* în România, la inițiativa lui Bogdan Manolea. Adaptarea celor șase licențe, începând de la *atribuire (by)* până la *atribuire-necomercial-fără opere derivate (by-nc-nd)* în limba română reprezintă al 26-lea proiect al *Creative Commons* în Europa și al 47-lea în întreaga lume. A se vedea și **Creative Commons**.

Licklider, Joseph Carl Robnett (1915–1990). Unul dintre cei mai cunoscuți informaticieni americani. În anul 1962, Licklider a introdus conceptul unei așa-numite *rețele galactice* (*galactic network*) de calculatoare interconectate, în care oricine poate avea acces la date și programe de oriunde în lume. A se vedea și **rețea galactică**.

LIFO. Acronim pentru *Last In, First Out*, *ultimul intrat, primul ieșit*, reprezintă o modalitate de organizare și manipulare a datelor. În particular, expresia descrie un principiu de procesare a datelor organizate într-o structură denumită *stivă* (*stack*). A se vedea și **stivă**.

limbaj compilat (*compiled language*). Limbaj de programare de nivel înalt utilizat pentru scrierea programelor translatate într-un format executabil înainte de execuție (compilate; e.g., C++), spre deosebire de programele scrise într-un limbaj interpretat. A se vedea și **limbaj interpretat**.

limbaj de asamblare (*assembly language*). Limbaj de programare de nivel jos cu care se scriu coduri folosind abrevieri – numite *mnemonice* – ale instrucțiunilor din limbajul-mașină al CPU (microprocesorului). Iată câteva exemple generale de mnemonice: ADD (adunare), SUB (scădere), MUL (înmulțire), DIV (împărțire), LOAD (încărcarea datelor din memorie), STORE (memorarea unei date). Un astfel de limbaj este denumit și limbaj din generația a treia, furnizând o reprezentare simbolică a instrucțiunilor-mașină. Programul scris într-un limbaj de asamblare specific unui anumit CPU este tradus în limbaj-mașină folosind un asamblor. A se vedea și **asamblor, cod- mașină și instrucțiune mașină**.

limbaj de comenzi (*command language*). Limbaj format din cuvinte-cheie și expresii, utilizat pentru scrierea comenzilor acceptate de interpretorul de comenzi. A se vedea și **linie de comandă**.

limbaj de descriere a paginii (PDL, *page description language*). Limbaj de programare (e.g., PostScript) utilizat pentru a descrie informațiile trimise unei imprimante sau unui dispozitiv de afișare. A se vedea și **PostScript**.

limbaj de interogare (*query language*). Subset al unui limbaj de manipulare a datelor, utilizat pentru regăsirea și afișarea datelor incluse într-o bază de date. A se vedea și **SQL**.

limbaj de marcare (*markup language*). Sistem (limbaj) care permite interpretarea explicită a unui fragment

de dată (e.g., text). Dintre limbajele de marcare cunoscute pot fi amintite SGML, HTML, XML, XHTML, RSS și LaTeX. Un limbaj de marcare trebuie să specifice mulțimea de marcaje obligatorii, maniera de identificare a marcajelor și semantica fiecărui marcaj disponibil. Sunt utilizate trei categorii de marcaje electronice: prezentaționale, procedurale și descriptive.

Marcajele prezentaționale sunt folosite de aplicațiile tradiționale de procesare a textului și înglobate în format binar în documente text pentru a produce efecte WYSIWYG. Aceste marcaje sunt, în mod obișnuit, ascunse utilizatorilor.

Marcajele procedurale sunt incluse în text și furnizează instrucțiuni pentru aplicațiile care îl procesează (e.g., `move(left_margin, 3 cm)`). Cele mai cunoscute limbaje de marcare procedurale sunt LaTeX și PostScript.

Marcajele descriptive (e.g., HTML, XML) sunt folosite ca etichete în document, obiectivul fiind acela de a decupla structura acestuia de orice interpretare a sa (adică de informația de prezentare – obiectivul este atins mai mult sau mai puțin, în grade care variază mult de la un limbaj la altul, e.g., de la HTML la XML). A se vedea și **HTML, marcaj, XHTML și XML**.

limbaj de nivel înalt (*high-level language*). Limbaj de programare care oferă un nivel înalt de abstractizare în raport cu limbajul de asamblare (e.g., PHP, C++, C#, Python, Ruby). Un astfel de limbaj este numit și limbaj din generația a treia.

Un limbaj de programare de nivel înalt utilizează instrucțiuni în structura cărora sunt incluse cuvinte-cheie preluate din limbajul natural (limba engleză). Limbajele de acest tip sunt ușor de învățat și prezintă un grad înalt de independență în raport cu platforma pe care sunt utilizate. Primul limbaj de nivel înalt creat de om a fost Plankalkül. A se vedea și **cod-mașină** și **Plankalkül**.

limbaj de programare (*programming language*). Limbaj artificial creat cu scopul de a exprima operații care pot fi efectuate de un calculator, furnizând instrucțiuni care pot fi prelucrate și executate de acesta. Prin utilizarea unui limbaj de programare, se pot scrie programe care sunt folosite pentru controlul comportamentului unui calculator.

Limbajele de programare au cunoscut o evoluție complexă, dar rapidă, sintetizată în continuare: cod-mașină; limbaj de asamblare; limbaj de nivel înalt (compilatoare, interpretoare); programare modulară (biblioteci de funcții, proceduri, linkeditorul), programarea structurată; programarea vizuală; programarea imperativă (procedurală) *versus* programarea declarativă; programarea orientată pe obiecte și pe evenimente; limbaje pentru baze de date, pentru simulare, conducere în timp real, concurențe; paradigma client-server. A se vedea și **program**.

limbaj de programare la nivel mașină (*machine-level programming language*). A se vedea **cod-mașină**.

limbaj de scripting (*scripting language*). Limbaj de programare interpretat, utilizat adesea în paginile Web dinamice. Limbajele de scripting pentru Web se clasifică în două categorii: limbaje pentru client (*e.g.*, JavaScript) și limbaje/platforme de programare pentru server (*e.g.*, PHP, JSP, ASP, Perl). A se vedea și **limbaj de programare**.

limbaj de uz general (*general-purpose language*). Limbaj de programare care poate fi utilizat pentru a crea diverse tipuri de aplicații (*e.g.*, C, C++, Pascal, Java). A se vedea și **limbaj de programare**.

limbaj din generația a doua (*second-generation programming language*). A se vedea **limbaj de asamblare**.

limbaj din generația a treia (*third-generation programming language*). A se vedea **limbaj de nivel înalt**.

limbaj din generația a patra (*fourth-generation language*). Limbaj de programare conceput să imite (parțial) limbajul uman. Termenul desemnează în special unele limbaje utilizate în scopuri speciale (*e.g.*, gestionarea bazelor de date relaționale, dezvoltarea de software pentru afaceri, crearea interfețelor GUI, dezvoltarea aplicațiilor Web). Se consideră că aceste limbaje sunt superioare celor de nivel înalt. Termenul a fost utilizat prima dată în 1982 de James Martin în cartea sa *Applications Development Without Programmers*.

limbaj interpretat (*interpreted language*). Limbaj de programare de nivel înalt utilizat pentru scrierea programelor ale căror instrucțiuni sunt translatate într-un format executabil și executate una câte una (e.g., PHP), spre deosebire de programele scrise într-un limbaj compilat. A se vedea și **interpretor și limbaj compilat**.

limbaj natural (*natural language*). Limbaj utilizat de oameni pentru vorbire, scriere sau comunicare prin semne. Este diferit de limbajele formale (e.g., limbajele de programare). A se vedea și **limbaj de programare**.

limbaj simbolic (*symbolic language*). Limbaj de programare ale cărui instrucțiuni sunt alcătuite pe baza unor simboluri (e.g., variabile, operatori, cuvinte-cheie). Cu excepția limbajului-mașină, toate limbajele de programare sunt simbolice. A se vedea și **limbaj de programare și limbaj de programare la nivel mașină**.

lighttpd. Server realizat sub licența BSD revizuită (<http://www.lighttpd.net/>), este utilizat de numeroase aplicații Web (inclusiv Web 2.0), de exemplu *YouTube*, *Wikipedia* sau *meebo*. A se vedea și **server Web și Web 2.0**.

LimSee. Editor SMIL, proiect al echipei WAM (*Web Adaptation Multimedia*) de la INRIA (*Institut National de Recherche en Informatique et en Automatique*), Grenoble, Franța (<http://limsee3.gforge.inria.fr/public-site/>). Printre facilitățile oferite de *LimSee*, pot fi menționate vizualizarea

șablonului spațial și a scenariului temporal, execuția prezentării, căutarea în document, validarea acestuia și stabilirea codificării utilizate. A se vedea și **SMIL**.

lingro. Instrument *online* foarte util celor care caută explicații pentru termenii incluși în paginile unui site Web sau traduceri ale acestora (<http://lingro.com/>). Utilizarea aplicației presupune parcurgerea a două etape: în prima se introduce (în caseta existentă în pagina principală a *Lingro*) adresa unei pagini Web. După apăsarea tastei *Enter*, conținutul paginii respective va fi inclus în vizualizatorul *Lingro* (*web viewer*). În a doua etapă, utilizatorul va executa click pe orice termen din pagina încărcată în *Lingro*, ceea ce va determina afișarea unei ferestre care include explicații referitoare la termenul selectat sau traducerea acestuia (în funcție de opțiunea utilizatorului). Sunt oferite traduceri în diverse limbi. De asemenea, aplicația poate oferi explicații sau traduceri ale unor termeni incluși în documente aflate pe calculatorul utilizatorului (sunt suportate numai formatele TXT, DOC și PDF). Acestea trebuie încărcate în prealabil în vizualizatorul *Lingro* (*file viewer*). Este pusă la dispoziția utilizatorilor o extensie pentru browserul *Firefox*, care permite folosirea rapidă a facilităților oferite de aplicație.

linie de comandă (*command line*). Șir text scris într-un limbaj de comenzi și transferat interpretorului de comenzi.

Confirmarea comenzii și transferul ei către interpretor se fac prin apăsarea tastei ENTER. Interpretorul primește, analizează și execută comanda respectivă. A se vedea și **interpretor de comenzi și limbaj de comenzi**.

linii de cod-sursă (SLOC, *source lines of code*). Modalitate utilizată pentru a măsura mărimea unui program prin numărarea rândurilor incluse în codul-sursă al acestuia. Este utilizată pentru a face predicții cu privire la efortul necesar dezvoltării unui program, ca și pentru a estima productivitatea unui programator sau efortul depus în dezvoltarea unui produs software. A se vedea și **program**.

link-uri nereciproce (*non-reciprocal links*). Legătură (*link*) situată pe site-ul A către site-ul B, acesta din urmă neavând legături către site-ul A. Acest tip de legături are o relevanță mai mare pentru motoarele de căutare decât cele reciproce. A se vedea și **link-uri reciproce**.

link-uri reciproce (*reciprocal links*). Două legături, una situată pe site-ul A către site-ul B, iar alta situată pe site-ul B către site-ul A. De multe ori, relevanța lor este pusă la îndoială de motoarele de căutare. A se vedea și **link-uri nereciproce** și **link exchange**.

link checker (*verificator de legături*). Program care scanează un site Web cu scopul de a identifica legăturile nefuncționale (*dead links*). În mod obișnuit, un astfel de program

generează un raport care include o listă cu toate legăturile moarte existente în site.

LinkedIn. Cea mai mare rețea socială *online* destinată profesioniștilor, folosită de peste 40 de milioane de utilizatori (în 2009) pentru schimb de informații, idei și oportunități (<http://www.linkedin.com/>).

Rețeaua a fost lansată în 2003 de Reid Hoffman și Konstantin Guericke, ca un mecanism de recomandare profesională bazat pe încrederea personală pe care utilizatorii o acordă altor utilizatori înregistrați. Până în 2005 circa două milioane de utilizatori își creaseră conturi pe *LinkedIn*, consolidând astfel una dintre cele mai active comunități de afaceri din lume. A se vedea și **rețea socială online**.

link building (*construirea legăturilor*). Plasarea pe site-uri externe a unor *link-uri* către site-ul propriu (numite *backlinks*), în scopul măririi vizibilității acestuia prin creșterea *PageRank*-ului. Există multe metode de a realiza *link building*, una dintre acestea fiind cumpărarea de legături. Metoda este penalizată de *Google*, utilizarea acesteia fiind depistată atunci când *PageRank*-ul are variații rapide (creșteri urmate de descreșteri). Iată și alte metode de mărire a vizibilității: înscrierea gratuită a site-ului în directoare Web, plasarea legăturilor pe site-urile de socializare, trimiterea regulată a unui newsletter, publicarea de anunțuri gratuite pe site-uri dedicate,

utilizarea *Twitter*, crearea de bloguri-sateți, promovarea fluxurilor RSS/Atom, publicarea unor comunicate de presă, comentarii pe forumuri, înscrierea în programele de afiliere, utilizarea unor legături în contul propriu *YouTube*, folosirea serviciilor de *social bookmarking*. A se vedea și **backlink**, **hiperlegătură**, **link exchange** și **PageRank**.

Link Directory. Director Web românesc în care site-urile sunt organizate pe categorii (cele mai populare dintre acestea fiind *Afaceri*, *Cumpărături*, *Acasă*, *Internet*, *Jocuri*, *Turism*, *Construcții și întreținere*) și pe subcategorii (<http://www.linkdirectory.ro/>). A se vedea și **director Web**.

link exchange (*schimb de legături*). Una dintre cele mai folosite metode de *link building*, utilizată în scopul promovării site-urilor. Varianta simplă (*link exchange* simplu) implică schimbul de legături între două site-uri: site-ul A postează o legătură către site-ul B ($A \rightarrow B$), care – la rândul lui – postează o legătură către site-ul A ($B \rightarrow A$). Acest schimb are avantajul că este ușor de realizat. De asemenea, se utilizează variantele *3-way linking* și *4-way linking*. Prima dintre ele se realizează astfel: site-ul A postează o legătură către site-ul B ($A \rightarrow B$), iar B către C ($B \rightarrow C$), A și C fiind în proprietatea aceleiași persoane sau companii. A doua metodă (numită și *diamond*) poate lua două forme (se presupune că site-urile A și C aparțin unei persoane/companii, iar B și D

alteia): un dublu *link exchange* simplu ($A \rightarrow B$, $C \rightarrow D$ și $B \rightarrow A$, $D \rightarrow C$) sau un dublu *3-way linking* ($A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$, $D \rightarrow A$). Excesul în utilizarea metodei *link exchange* este dezaprobat de *Google*, în special excesul utilizării variantei simple. Majoritatea optimizatorilor experimentați o vor evita ca metodă viabilă de *link building*, ea fiind totuși utilizată în anumite situații de nișă. A se vedea și **link building**.

link juice (*link juice*). Exprimă puterea, greutatea, valoarea pe care le transmite o legătură (*link*) documentului la care trimite. Este proporțional cu importanța paginii în care este inclusă legătura și invers proporțional cu numărul de legături publicate în pagină. A se vedea și **Google juice**.

link popularity (*popularitatea legăturii*). Expresia cantitativă și calitativă a legăturilor incluse în alte site-uri și care fac referire la site-ul propriu. Reprezintă un criteriu folosit de motoarele de căutare în poziționarea site-urilor. *Google Toolbar* oferă informații legate de această caracteristică, precum și *PageRank*-ul paginii curente.

Linux. Se referă la nucleul Linux, dar, în mod obișnuit, termenul este folosit pentru a denumi un întreg sistem de operare (*i.e.*, nucleu, biblioteci și diverse instrumente). Linux este unul dintre cele mai cunoscute proiecte de software *open source*. Nucleul Linux a fost dezvoltat de Linus Torvalds. O distribuție Linux este alcătuită din componentele de bază (enumerare mai

sus) și o mare cantitate de aplicații organizate în pachete. A se vedea și **distribuție Linux și Torvalds, Linus.**

(a) **lipi (paste).** A insera o porțiune dintr-un document, păstrată în memorie (*clipboard*), în documentul curent (din care a fost tăiată) sau în alt document similar. A se vedea și (a) **tăia.**

LISP (List Processing). Limbaj de programare funcțional folosit pentru crearea aplicațiilor de inteligență artificială, mai ales între anii 1970 și 1980. A fost inventat în 1958, la MIT, de John McCarthy. Este cel de-al doilea limbaj de nivel înalt în ceea ce privește vechimea, după Fortran. Principalele două dialecte utilizate astăzi sunt *Common Lisp* (abreviat CL) și *Scheme*. A se vedea și **inteligență artificială și McCarthy, John.**

listă (list). Colecție de elemente de informație (noduri) aranjate într-o anumită ordine. Lungimea listei este egală cu numărul nodurilor pe care le conține. Structura de date de tip listă trebuie să permită determinarea primului/ultimului nod din structură și a predecesorului/succesorului (dacă există) unui nod dat. Cea mai simplă listă este cea liniară, care are un cap și o coadă. O listă circulară este o listă în care după ultimul nod urmează primul, fiecare nod având astfel un predecesor și un succesor. Reprezentarea cea mai comodă pentru această structură este cea secvențială, bazată pe tablouri. A se vedea și **listă liniară, structură de date și tablou.**

listă de selecție (select list). Listă de opțiuni inclusă într-un formular (X)HTML existent într-o pagină Web.

Lista este creată folosind marcajul (X)HTML `<select>` și atributele sale: *name* (valoarea sa este un text care stabilește numele listei), *size* (valoarea sa este un număr care stabilește câte rânduri ale listei vor fi afișate simultan) și *multiple* (valoarea sa este "multiple"; dacă este setat, atributul permite selectarea simultană a mai multor opțiuni).

O listă de selecție poate fi de următoarele tipuri: listă derulantă (este prezent, opțional, numai atributul *name*); casetă de selecție (sunt prezente numai atributele *name* și *size*); listă derulantă cu selectări multiple (are precizate numai atributele *name* și *multiple* și afișează *toate* opțiunile listei); casetă de selecție cu selectări multiple (sunt prezente simultan toate cele trei atribute).

O listă de selecție trebuie să conțină una sau mai multe opțiuni, fiecare dintre acestea fiind creată folosind marcajul `<option>`. Iată câteva dintre atributele marcajului: *selected* (dacă este folosit, atributul precizează că opțiunea este preselectată); *value* (stabilește un text asociat opțiunii); *disabled* (dezactivează pentru selecție opțiunea respectivă); *label* (stabilește o etichetă asociată opțiunii). A se vedea și **HTML.**

listă înlănțuită (linked list). În programare, este o structură de date de tip listă liniară, alcătuită dintr-o

secvență de noduri care includ, fiecare, unul sau două câmpuri cu referințe (pointeri, adică informații de adresă) la nodurile învecinate. Există două tipuri de liste înlănțuite: listă simplă și listă dublă.

Într-o listă simplu înlănțuită (cunoscută și sub numele de *listă legată*, *linked list*), fiecare nod (cu excepția ultimului) include câte un singur câmp care conține o referință (pointer) către următorul nod. Alocarea simplă permite parcurgerea listei într-un singur sens (de la stânga la dreapta). Listele legate sunt cele mai utilizate structuri de date și sunt folosite pentru a implementa o serie de structuri de date abstracte cunoscute (e.g., stive, cozi). De asemenea, dintre aplicațiile acestui tip de listă liniară pot fi menționate sortarea prin inserție și sortarea topologică.

Într-o listă dublu înlănțuită (cunoscută și sub numele de *listă dublu legată*, *doubly linked list*), fiecare nod include câte două câmpuri cu câte o referință (pointer) la nodul predecesor, respectiv succesor.

Listele înlănțuite au avantajul că adăugarea și eliminarea nodurilor se fac rapid. Dintre dezavantaje pot fi menționate: accesul la un nod se face după parcurgerea tuturor nodurilor care îl preced, operație care necesită efort de calcul; referințele conținute în cadrul fiecărui nod ocupă memorie. A se vedea și **coadă**, **listă liniară**, **listă secvențială** și **stivă**.

listă liniară (*linear list*). În programare, colecție de noduri aflate într-o relație de ordine. Este denumită și *listă deschisă* (*open list*), deoarece ultimul nod al listei conține o referință *null* (nici o adresă). Într-o listă circulară, toate nodurile sunt legate într-un cerc continuu (nu există nici o referință *null*).

Operațiile permise asupra unei liste liniare sunt: accesul la oricare dintre noduri în scopul citirii sau modificării informațiilor conținute de acesta; adăugarea unui nod, indiferent de poziția pe care o va ocupa în listă; ștergerea unui nod, indiferent de poziția pe care acesta o are în listă; schimbarea poziției unui nod în listă. Există două modalități de alocare a unei liste liniare: alocarea secvențială (listă secvențială) și alocarea înlănțuită (listă înlănțuită). A se vedea și **listă înlănțuită** și **listă secvențială**.

listă secvențială (*sequential list*). În programare, structură de date de tip listă liniară alcătuită dintr-o secvență de noduri care ocupă poziții succesive în memorie. Avantajul utilizării unei astfel de liste este acela că există acces direct la oricare dintre nodurile sale, la fel ca la elementele unui vector. Dezavantajul utilizării acestui tip de listă liniară este acela că adăugarea, ștergerea sau schimbarea poziției unui nod necesită un efort mare de calcul. A se vedea și **listă liniară** și **listă înlănțuită**.

little endian (*little endian*). Numită și *ordonare inversă a octeților* (*reverse byte ordering*), este o metodă de stocare

a numerelor întregi în care cel mai puțin semnificativ octet este poziționat primul. Este utilizată de microprocesoarele produse de Intel. A se vedea și **big endian**.

LiveJournal. Abreviat adesea LJ, este un serviciu gratuit de *blogging*, creat în 1999 (<http://www.livejournal.com/>). A se vedea și **serviciu de blogging**.

LMS (*Learning Management System, sistem pentru managementul învățării*). Software utilizat pentru furnizarea, urmărirea și managementul instruirii *online*. Utilizate în educație, dar și în alte sectoare (*e.g.*, administrație, industrie) în scopul pregătirii angajaților, LMS-urile realizează, în principal, funcțiile de administrare a participanților și cursurilor unei instituții *online*. Gestiunea cursurilor se face la nivel ridicat, ceea ce înseamnă că – în mod obișnuit – cursurile sunt importate din alte aplicații (*e.g.*, instrumente pentru crearea cursurilor).

Utilizatorii unui LMS aparțin următoarelor categorii: cursanți (studenți), instructori (educatori), administratori și dezvoltatori (designeri). Administratorul are posibilitatea ca prin intermediul unei interfețe Web să realizeze diferite operațiuni, ca de exemplu: înregistrarea studenților, adăugarea cursurilor, înregistrarea rezultatelor/notelor și generarea rapoartelor. Pentru stocarea informațiilor despre cursanți și cursuri, un LMS utilizează o bază de date al cărei conținut poate fi modificat de administratori și,

automat, de diferite subsisteme ale acestuia.

Deși au o serie de caracteristici comune, LMS-urile se diferențiază prin implementarea unor servicii specifice care răspund unor nevoi particulare, caracteristice sectorului de activitate în care sunt utilizate. Dintre caracteristicile comune ale LMS-urilor pot fi amintite următoarele: gestionarea utilizatorilor, rolurilor, cursurilor, educatorilor; oferirea de cataloage de curs și înregistrare *online*; lansarea și urmărirea conținutului educațional; managementul resurselor educaționale; generarea de rapoarte personalizabile; furnizarea calendarului cursurilor; transmiterea de mesaje și notificări studenților; testarea și evaluarea studenților; oferirea unor instrumente de învățare; capacitatea de a integra și alte aplicații consacrate. A se vedea și **LCMS**.

load balancing (*echilibrarea încărcării*). Tehnică prin care se împarte efortul de calcul între mai multe calculatoare, discuri etc., în scopul obținerii unei utilizări optime a resurselor și minimizării timpului necesar procesării. Această tehnică oferă numeroase avantaje: flexibilitate, disponibilitate maximă, scalabilitate, timp de răspuns minim. Tehnica are patru tipuri de funcționalități majore: *server load balancing*, *global server load balancing*, *firewall load balancing*, *transparent cache switching*.

localhost (*gază locală*). Nume de gazdă folosit pentru un calculator

atunci când acesta este simultan sursă și destinație pentru mesajele TCP/IP. Totodată, acesta este un nume de domeniu rezervat. De fapt, pachetele IP trimise de/către gazda locală nu ajung în Internet. Numele gazdei locale este translatat în adresa IP de buclă locală, anume 127.0.0.1. Numele gazdei locale este utilizat în special pentru testarea software-ului (e.g., site-uri și aplicații Web). Astfel, *http://localhost* este URL-ul paginii principale a unui site Web găzduit de un server HTTP local. A se vedea și **adresă IP**.

localizare (*localization*). Adaptarea la zona geografică (e.g., România) a unui produs software (de sistem sau de aplicație). În mod obișnuit, procedura necesită cel puțin traducerea produsului (e.g., în limba română).

Location. Unul dintre anteturile răspunsului HTTP. Este utilizat pentru a redirecționa clientul Web către o locație a resursei diferită de cea specificată în cerere, împreună cu coduri de stare 3xx sau 201 (Created), fiind precizat sub forma unui URL. A se vedea și **cod de stare HTTP** și **răspuns HTTP**.

logic (*logical*). Numit și *boolean*, este un tip de date primitiv cu două valori posibile: adevărat (*true*) sau fals (*false*). Unele limbaje de programare nu includ tipul logic ca tip predefinit de date. Primul limbaj care a inclus explicit acest tip este Algol 60 (1960). A se vedea și **tip de date**.

logica fuzzy (*fuzzy logic*). Logica vagă (*fuzzy*), spre deosebire de cea clasică

(sau binară), utilizează o plajă continuă de valori logice (numite grade de adevăr sau probabilități) cuprinse în intervalul 0-1, unde 0 indică *fals*, iar 1 *adevărat*. A fost definită în anul 1965 de profesorul Lotfi Zadeh, de la Berkeley University.

Logica *fuzzy* este utilizată în sistemele de inteligență artificială pentru reprezentarea unor concepte imprecise (variabile *fuzzy*). Sistemele *fuzzy* au fost incluse în diverse bunuri de larg consum (e.g., aparate foto, mașini de spălat, cuptoare cu microunde). A se vedea și **inteligență artificială**.

LOGO (*Logic Oriented Graphic Oriented*). Limbaj de programare funcțională, adaptare și dialect al Lisp (uneori, este numit *Lisp fără paranteze*). Oferă facilități pentru manipularea listelor și fișierelor, recursivitate etc. A fost creat în 1967 de o echipă formată din Seymour Papert, Daniel G. Bobrow și Wally Feurzeig, cu scopul de a fi utilizat în învățământ, în special pentru predarea programării în manieră constructivistă. A se vedea și **constructivism social** și **Papert, Seymour**.

log off (*a închide o sesiune de lucru*). A încheia o sesiune de lucru cu un calculator aflat la distanță. Operația se mai numește și *log out* (scris și *logout*), *sign out* sau *sign off*. A se vedea și **log on**.

log on (*a deschide o sesiune de lucru*). A deschide o sesiune de lucru cu un calculator aflat la distanță. În mod obișnuit, pentru a realiza această

operație sunt necesare un nume de utilizator și o parolă. Operația se mai numește și *sign on* sau *log in*. A se vedea și **log off**.

LOM (*Learning Object Metadata*). Cel mai răspândit standard pentru specificarea metadatelor asociate obiectelor de învățare, aprobat de *IEEE-Standards Association* pe 12 iunie 2002. LOM se află la baza standardelor pentru învățare electronică SCORM și IMS Learning Design.

Scopul LOM este de a facilita căutarea, evaluarea, utilizarea și reutilizarea obiectelor de învățare de către educatori, studenți sau medii de învățare *online*. Alegerea metadatelor folosite de dezvoltatori este adesea dificilă. Totuși, respectarea unui standard (e.g., LOM) a devenit obligatorie din punct de vedere comercial, deși nu este întotdeauna potrivită cu interesele comunităților educaționale. A se vedea și **obiect de învățare**.

LOMS (*Learning Object Management Systems, sistem pentru managementul obiectelor de învățare*). Sistem de management care trebuie să le permită utilizatorilor să caute nu numai după metadatele asociate obiectelor de învățare, ci și după conținutul acestora. În plus, un LOMS nu trebuie să impună limite cu privire la tipul conținutului, respectând astfel definiția principală a obiectului de învățare adoptată în literatura de specialitate, conform căreia acesta este reprezentat de orice conținut digital utilizat într-un context

de învățare. A se vedea și **obiect de învățare**.

long double. Tip de date utilizat în unele limbaje de programare (e.g., C, C++) pentru a declara variabilele care stochează numere reale în virgulă mobilă, ocupând în mod obișnuit 80 de biți. A se vedea și **număr în virgulă mobilă**.

long tail (*coadă lungă*). Termen introdus de Chris Anderson în octombrie 2004. Se referă la faptul că un comerciant de produse de nișă pe Internet poate realiza un profit ridicat atâta timp cât numărul clienților săi este suficient de mare. Această teorie poate fi aplicată la SEO astfel: cuvintele derivate sau nu din cele de bază vor genera individual un trafic neglijabil, dar luate împreună vor aduce un aport substanțial la traficul organic. A se vedea și **SEO**.

Looker. Director Web românesc organizat pe categorii (e.g., Afaceri și catalog firme, Auto, Calculatoare, Comunități *online*, Cultură și artă) și subcategorii (<http://www.looker.ro/>). A se vedea și **director Web**.

LORE (*Learning Object Repository for Edinburgh University*). Depozit de obiecte de învățare creat de universitatea din Edinburgh (<http://www.lore.ed.ac.uk/>). A se vedea și **obiect de învățare**.

Lovelace, Ada (1815–1852). Matematician englez, fiica lordului Byron. A fost singura persoană dintre

contemporanii lui Charles Babbage care a înțeles ideile acestuia referitoare la dispozitivele de calcul. A scris primul program pentru mașina analitică a lui Babbage, rămânând astfel în istorie ca „primul programator”. În onoarea sa, limbajul Ada îi poartă numele. A se vedea și **Ada și Babbage, Charles**.

lowercase (*litere mici*). Ceea ce ține de utilizarea literelor mici. A se vedea și **uppercase**.

Lucene. Software de căutare dezvoltat de *Apache Software Foundation* (<http://www.apache.org/>), este ceea ce

se numește o *full-featured text search engine library* (*biblioteca-motor de căutare după text complet*). Poate fi utilizată atât pentru aplicații Web, cât și pentru aplicații *desktop*.

lume virtuală (*virtual world*). Mediu simulat pe computer care le oferă utilizatorilor săi posibilitatea de a interacționa prin intermediul unor avataruri. În mod obișnuit, avatarurile sunt reprezentări grafice bi- sau tridimensionale, fiind posibile și formate alternative (auditive sau tactile). A se vedea și **Second Life**.

M

MAC address (*adresă MAC*). A se vedea **adresă fizică**.

Macintosh. Denumită și Mac, este o linie de calculatoare personale produse de compania Apple începând cu anul 1984. Calculatoarele Mac au fost printre primele PC-uri dotate cu interfață grafică și *mouse*. A se vedea și **calculator personal**.

Mac OS X. Linie de sisteme de operare bazate pe UNIX, dezvoltată și comercializată de Apple, Inc. Include versiuni pentru *desktop* și server (Mac OS X Server). Platforma hardware standard este întreaga linie de computere MacIntosh (*laptop*, *desktop* și server). De asemenea, compania Apple a dezvoltat versiuni specializate pentru alte trei dispozitive proprii: iPhone OS pentru iPhone și iPod Touch și o versiune (fără nume) pentru Apple TV.

machetă de tastatură (*keyboard layout*). Aranjament mecanic, spațial sau funcțional al tastelor de pe o anumită tastatură. Unele sisteme îi permit utilizatorului să-și configureze (personalizeze) tastatura în funcție de propriile preferințe.

magistrală (*bus*). Linie electrică folosită în comun de mai multe unități funcționale pentru realizarea unei sarcini. Magistralele pot fi de date, de adrese și de comenzi.

Magistrala de date (bidirecțională) este utilizată pentru transferul datelor între microprocesor, memorie și dispozitivele de intrare-ieșire. Magistrala de adrese (unidirecțională) este utilizată pentru transmiterea adreselor de la microprocesor către memorie și dispozitivele de intrare-ieșire. Magistrala de comenzi reunește semnalele de la dispozitivele de intrare-ieșire și de la microprocesor; acestea pot fi semnale de control sau de stare.

magistrală locală PCI (*PCI, Peripheral Component Interconnect*). Magistrală locală utilizată pentru atașarea la calculator a unor dispozitive. Acestea pot fi circuite integrate incluse pe placa de bază sau plăci de extensie.

mailto. Identificator de serviciu utilizat în atributul `href` al unei hiperlegături pentru a trimite mesaje de *e-mail* unei persoane, ca în exemplul următor:

```
<a href="mailto:george@exemplu.ro">e-mail George</a>
```

A se vedea și **e-mail** și **hiperlegătură**.

malware (*malware*). Termen generic care se referă la orice software creat în mod deliberat pentru acțiuni neautorizate și care, în mod obișnuit, produce pagube sistemelor de calcul în care este executat. În categoria desemnată prin intermediul acestui termen sunt incluse

diverse tipuri de programe: virușii de calculator (*computer viruses*), viermii (*worms*), troienii (*trojan horses*), precum și programele *adware* și *spyware*. A se vedea și **adware**, **spyware**, **troian**, **vierme** și **virus de calculator**.

Mambo. CMS *open source* foarte popular, bazat pe PHP și MySQL, folosit pentru a crea și administra site-uri printr-o interfață grafică simplă; oferă o mulțime de facilități, dintre care pot fi amintite: salvarea paginilor în *cache* (pentru a îmbunătăți performanțele site-urilor cu trafic ridicat), tehnici avansate de realizare a șabloanelor (*templates*), versiuni printabile ale paginilor, forum-uri, știri, calendare și *feed*-uri RSS (<http://www.mambo-server.com/>). A se vedea și **CMS**.

MAN (*Metropolitan Area Network*). Rețea de calculatoare de mare viteză care poate transporta date pe distanțe și cu viteze mari. Poate conține una sau mai multe rețele LAN și echipamente de telecomunicații. A se vedea și **LAN**, **rețea de calculatoare** și **WAN**.

manager de ferestre (*window manager*). Software de sistem care controlează afișarea și aspectul ferestrelor în cadrul unui sistem de ferestre (*window system*) într-o interfață-utilizator grafică. Elementele asociate în mod uzual cu un manager de ferestre sunt cele care îi permit utilizatorului să deschidă, să închidă, să maximizeze, să minimizeze, să redimensioneze și să mute ferestrele.

Utilizatorii sistemului de ferestre X Window System (denumit, pe scurt, X) pot folosi diverși manageri de ferestre, cum ar fi cei incluși în mediile desktop (*desktop environments*) GNOME, KDE etc.; dintre aceștia pot fi amintiți *twm* (*Tab Window Manager*), *gwm* (*Generic Window Manager*) și *olwm* (*OpenLook Window Manager*). A se vedea și **GNOME**, **KDE**, **sistem de ferestre** și **X Window System**.

manipularea erorilor (*error handling*). Anticiparea, detectarea și rezolvarea unor erori de programare sau de comunicație.

marcaj (*markup*). Informație structurală stocată într-un document. În mod obișnuit, informația respectivă este separată de conținutul propriu-zis al documentului. Marcajele utilizate sunt de trei tipuri: prezentaționale, procedurale și descriptive. A se vedea și **limbaj de marcare**.

marcaj HTML (*HTML markup*). Marcaj utilizat într-un document HTML. A se vedea și **HTML** și **marcaj**.

marcarea resurselor (*tagging*). Operația de etichetare de către utilizatori a resurselor Web descoperite sau create de ei înșiși, folosind marcaje (*tag-uri*) proprii și sisteme sau servicii care suportă această operație.

Persoanele care realizează operația de marcare a resurselor sunt denumite utilizatori (*users*). Ocazional, acestea sunt denumite și *taggers*. Utilizatorii pot avea o varietate de interese, nevoi,

scopuri și motivații: aceștia marchează resursele pentru ca, ulterior, să le poată regăsi cu ușurință. O resursă poate fi orice: o carte, o fotografie, o pagină Web, o intrare a unui blog, o secvență video sau chiar o locație. Astfel, în *LibraryThing*, resursele sunt cărți, în *Flickr* sunt fotografiile, iar în *delicious* resursele sunt URL-uri. Cuvintele-cheie sau etichetele asociate de utilizatori resurselor se numesc marcaje (*tags*).

O caracteristică importantă a activității de marcare a resurselor este aceea că *tag*-urile pot fi reutilizate de câte ori se dorește, furnizând astfel informații despre propria lor popularitate, dar și despre interesele utilizatorilor. În ceea ce privește accesibilitatea resurselor marcate, acestea pot fi încadrate în următoarele categorii: resurse deschise tuturor utilizatorilor (publice); resurse deschise unui grup de utilizatori; resurse personale (private); resurse partajate selectiv.

Deoarece pentru marcarea unei resurse pot fi utilizate *tag*-uri folosite anterior de alte persoane pentru resurse similare și – de asemenea – pot fi marcate resurse ale altor utilizatori, operația de marcare mai poartă numele de *social tagging*. În contextul existenței unei audiențe (*i.e.*, resursele și *tag*-urile pot fi vizualizate nu numai de proprietarul lor, ci și de ceilalți utilizatori), marcarea resurselor nu este numai o activitate individuală, desfășurată în scopul organizării resurselor, ci este prin excelență un *act social* cu impact direct

asupra întregii comunități de vizitatori/utilizatori. În contextul folosirii unui sistem de marcare socială, datele partajate (resurse și *tag*-uri) pot fi denumite *date sociale* (*social data*).

Marcarea socială a resurselor pune în valoare inteligența colectivă, deoarece percepția resurselor nu mai este una personală, nici colectivă (fiecare utilizator își marchează numai propriile resurse), ci una colaborativă (o resursă dată poate fi marcată de mai mulți utilizatori). A se vedea și **sistem de marcare a resurselor și tag**.

Mark I. Primul calculator electronic de dimensiuni mari din SUA, văzut și ca primul calculator universal. A fost proiectat de Howard Aiken și produs de IBM (1944). Mark I era programat prin bandă perforată și utiliza relee. Pentru construcția sa au fost utilizate peste 765 000 de componente și câteva sute de kilometri de cablu. A se vedea și **Aiken, Mark**.

mashup (*aplicație hibridă*). Aplicație Web care combină în mod inteligent date provenind din surse externe diverse într-o formă unitară și complexă.

Datele utilizate pentru crearea *mashup*-urilor sunt puse la dispoziție pe Web prin intermediul interfețelor (publice) de programare a aplicațiilor (API), prin servicii Web sau *feed*-uri. Aceste tehnologii realizează divizarea informației în părți de dimensiuni reduse și slab conectate, reducând complexitatea aplicațiilor care le utilizează

(e.g., *mashup-uri*), inclusiv efortul de integrare.

În funcție de tipul informației conținute, *mashup-urile* pot fi clasificate în mai multe categorii: cartografiere, distribuție de fotografii, cumpărături *online*, distribuție video, rețele sociale și căutare de informație.

Cele mai populare *mashup-uri* sunt cele de cartografiere (*mapping*). În mod obișnuit, un astfel de *mashup* localizează și organizează datele în cadrul unei hărți, folosind o interfață-utilizator personalizată. O aplicație *mashup* care are în componența sa un serviciu geografic (i.e., de cartografiere) se mai numește și *geo-mashup*. Astăzi, există peste 40 de API-uri care oferă diverse servicii legate de cartografiere, dintre care pot fi menționate (în ordinea descrescătoare a popularității): *Google Maps API*; *Yahoo! Maps API*; *Microsoft Virtual Earth API*; *AOL MapQuest*. A se vedea și **API, feed, Google Maps API și serviciu Web**.

mașină (*machine*). Termen echivalent cu calculator, gazdă sau sistem.

mașina Turing (*Turing machine*). Mecanism imaginar de dispozitive de prelucrare a simbolurilor descris în 1936 de Alan Turing. Acest mecanism poate fi adaptat pentru a simula logica oricărui calculator. Studiul proprietăților abstracte ale mașinilor Turing este utile în informatică și teoria complexității. Conform conjecturii Church-Turing, orice problemă de

calcul bazată pe o procedură algoritmică poate fi rezolvată de o mașină Turing.

O mașină Turing capabilă să simuleze orice altă mașină Turing poartă numele de *mașină Turing universală* (UTM, *Universal Turing Machine*). A se vedea și **Turing, Alan Mathison**.

mașină virtuală Java (JVM, *Java Virtual Machine*). Componentă esențială a platformei Java, constă într-un set de programe și structuri de date având drept scop traducerea (i.e., interpretarea) instrucțiunilor unui *byte code* (cod de octeți) generat în mod obișnuit din codul-sursă Java, în instrucțiuni-mașină pentru sistemul de operare curent și execuția acestora. Astfel, fișierele intermediare *byte code* pot fi copiate și executate pe orice platformă (e.g., Windows, UNIX, Mac OS). JVM este o instanță a JRE (*Java Runtime Environment*) și intră în acțiune atunci când un program Java este executat. A se vedea și **byte code, Java și mediu de execuție Java**.

Matsumoto, Yukihiro (n. 1965). Programator japonez, creatorul limbajului de programare Ruby. Este autorul cărții *The Ruby Programming Language*. A se vedea și **Ruby**.

Mauchly, John (1907–1980). Fizician american, a proiectat – împreună cu inginerul Presper Eckert – ENIAC, primul calculator electronic digital de uz general. Cei doi au înființat Eckert-Mauchly Computer Corporation (EMCC), prima companie comercială

de producere și distribuție a calculatoarelor. A se vedea și **Eckert, John Adam Presper** și **ENIAC**.

(a) **maximiza** (*maximize*). Referitor la interfețele grafice cu utilizatorul, este procesul prin care se mărește dimensiunea unei ferestre astfel încât aceasta să ocupe întregul spațiu pe care îl are la dispoziție (*e.g.*, întregul ecran). În acest scop este utilizat un buton dedicat. De asemenea, operația poate fi realizată prin execuția unui dublu click pe bara de titlu a ferestrei. A se vedea și **fereastră** și (a) **minimiza**.

mănușă de date (*data glove*). Dispozitiv de intrare pentru mediile de realitate virtuală. Este prevăzut cu senzori care convertesc mișcărilor degetelor și ale mâinilor în comenzi. Prima mănușă de date – denumită *Sayre Glove* – a fost creată în 1977 de Electronic Visualization Laboratory (EVL).

Mbps (sau *Mbit/s*, *Mb/s*). Prescurtare de la *megabits per second* (megabiți pe secundă). Reprezintă un milion de biți pe secundă (*i.e.*, 10^6 bps). A se vedea și **bit rate**.

McCarthy, John (n. 1927). Informatician american, inventatorul conceptului de *inteligentă artificială* (*artificial intelligence*, AI). Este laureat al premiului Turing (1971), pentru contribuțiile aduse în domeniul AI. A inventat limbajul de programare LISP. A se vedea și **inteligentă artificială** și **LISP**.

MD5. În criptografie, funcție de tip *hash* unidirecțional care generează o valoare cu lungimea fixă de 128 de biți, reprezentată în mod obișnuit ca o secvență de 32 de cifre hexazecimale. Funcția a fost creată în 1991 de profesorul Ronald L. Rivest, de la MIT, fiind publicată în 1992 în RFC 1321 și înlocuind funcția MD4, care prezenta vulnerabilități de securitate. A se vedea și **md5checksum** și **funcție hash**.

md5sum. Program de calculator care calculează și verifică valoarea funcției MD5 pentru un fișier, aceasta putând fi utilizată drept sumă de control. Deoarece orice schimbare a unui fișier va determina modificarea sumei de control asociate, aceasta poate fi utilizată pentru a verifica integritatea fișierului (*e.g.*, se verifică dacă fișierul a fost sau nu alterat ca urmare a transferului prin *upload* sau *download*). Programul *md5sum* este instalat automat de sistemele de operare UNIX și Linux. De asemenea, variantele BSD (inclusiv Mac OS X) instalează un utilitar similar, denumit *md5*. A se vedea și **MD5**.

Media Convert. Convertor *online* general gratuit, suportă conversia din/într-un număr foarte mare de formate (text, documente, arhive, video, audio, grafice). Poate fi descărcat la (<http://media-convert.com/>).

Alături de conversia formatelor, *Media Convert* mai permite: redimensionarea și compresia imaginilor (pe baza nivelului furnizat de utilizator); împărțirea (splitarea) fișierelor audio

(pot fi alese punctele în care va fi realizată splitarea); alegerea metodei de codificare.

Dimensiunea fișierelor care pot fi convertite folosind serviciul *online Media Convert* este de până la 150 MB. Aceste fișiere trebuie încărcate de pe calculatorul utilizatorului sau precizate prin indicarea URL-ului. În ultimul caz sunt acceptate protocoalele *http*, *ftp*, *mms*, *mmsh*, *mmst* și *rtsp*. Conversia unui fișier se realizează în trei etape: selectarea fișierului (se utilizează butonul *Browse*); selectarea formatului de intrare (dacă autodetecția acestuia a eșuat) și a formatului de ieșire; trimiterea fișierului (se utilizează butonul *Ok*).

media editor (ME, *editor media*). Instrument software utilizat pentru crearea/modificarea conținutului media existent în diverse materiale, inclusiv în cele folosite în *e-Learning*.

ME sunt folosite drept aplicații externe, adăugându-se atunci când este necesar componentelor de editare media incluse deja în diverse instrumente utilizate în *e-Learning* (e.g., CAT, LCMS). Este de notat că rularea unor tipuri de conținuturi media poate necesita folosirea unor instrumente specifice (e.g., media player).

În lista următoare este prezentată o clasificare a categoriilor și a celor mai populare tipuri media existente: multimedia (categoria include animații interactive); imagini (categoria include următoarele tipuri: imagini în formate vectoriale, imagini în formate *raster*,

imagini capturate cu camere digitale și transferate pe calculator, unde sunt editate folosind aplicații specifice); animații (sunt imagini afișate secvențial, într-o anumită perioadă de timp); audio (categoria include următoarele tipuri: sunete, muzică); video (sunt secvențe video capturate de camere video digitale, care pot fi transferate pe computer și editate pentru a produce videoclipuri).

Alegerea formatului în care va fi creat conținutul media este foarte importantă, deoarece este posibil ca un format puțin răspândit sau a cărui utilizare necesită playere media speciale care să determine imposibilitatea accesării conținutului respectiv. Ținând seama de această observație, se poate realiza încă o clasificare a formatelor media: formate native pentru browser (e.g., GIF, JPEG, *Flash*), formate larg răspândite (e.g., SVG, PNG, MP3, RealAudio, QuickTime, RealMedia) și formate proprietare (e.g., WMA, ASF, WMV). A se vedea și **Course Authoring Tool** și **e-Learning**.

MediaWiki. Unul dintre cele mai populare software pentru *wiki*, pe baza sa fiind creată enciclopedia *online Wikipedia*. Este scris în PHP și poate utiliza baze de date MySQL sau PostgreSQL. Dintre caracteristicile sale pot fi amintite: suportă legături interne și spații de nume; permite crearea subpaginilor, categoriilor și secțiunilor; oferă posibilitatea utilizării șabloanelor. A se vedea și **wiki** și **Wikipedia**.

mediu de execuție Java (JRE, *Java Runtime Environment*). Mediul în care sunt executate aplicațiile Java (compilate în *byte code*). Include o mașină virtuală Java și un set de biblioteci utile dezvoltatorilor. A se vedea și **byte code** și **mașină virtuală Java**.

mediu desktop (DI, *desktop environment*). Se referă la interfața grafică cu utilizatorul (GUI), utilizată astăzi de toate sistemele de operare. Un DE tipic constă din pictograme, ferestre, bare de instrumente, *wallpapers* (fundaluri pentru *desktop*) și *widget*-uri pentru *desktop*. Cele mai cunoscute DE sunt GNOME și KDE (Linux), precum și cele existente pe platformele Microsoft Windows și Mac OS X. A se vedea și **GNOME** și **KDE**.

mediu RAD (*RAD environment*). Mediu de dezvoltare integrat (IDE) care utilizează o metodă rapidă de dezvoltare a aplicațiilor cunoscută sub numele de RAD (*Rapid Application Development*). Un IDE de acest tip este C++Builder. A se vedea și **C++Builder**, **IDE** și **RAD**.

mediu virtual de învățare (VLE, *Virtual Learning Environment*). Sistem software proiectat pentru a sprijini predarea, învățarea și evaluarea. În mod obișnuit, un VLE este utilizat în Internet, furnizând un set complex de instrumente pentru livrarea conținutului, comunicare, evaluare, administrarea grupurilor de studenți, urmărirea progresului școlar etc. Dintre caracteristicile versiunilor recente ale acestor

medii se remarcă includerea wiki-urilor, blogurilor, fluxurilor RSS și spațiilor virtuale de învățare 3D.

Sistemele VLE sunt cunoscute și sub alte denumiri, ca de exemplu: sisteme de management al învățării (LMS, *Learning Management System*), sisteme de management al conținutului (CMS, *Content Management System*), sisteme de management al conținutului învățării (LCMS, *Learning Content Management System*), medii pentru managementul învățării (MLE, *Managed Learning Environment*), sisteme pentru suportul învățării (LSS, *Learning Support System*), centre *online* pentru învățare (OLC, *Online Learning Center*) sau platforme de învățare (LP, *Learning Platform*).

Sistemele/mediile enumerate anterior permit realizarea educației prin comunicare mediată de computer (CMC, *Computer-Mediated Communication*) sau așa-numita educație *online* (*Online Education*). Trebuie remarcat totuși că scopurile în care au fost proiectate, caracteristicile și modalitățile de utilizare ale sistemelor enumerate nu se suprapun în totalitate.

Meebo. Aplicație Web de mesagerie instantanee (IM) bazată pe AJAX (<http://www.meebo.com/>). Suportă servicii IM multiple, incluzând (dar nu numai) *Yahoo! Messenger*, *Windows Live Messenger*, *AOL Instant Messenger* (AIM), *MySpaceIM* și *ICQ*.

megabyte (*megaoctet* sau *megabit*). Notat MB, are 2^{20} (sau 1024^2) *bytes*

(sau octeți), adică 1.048.576 *bytes*. A se vedea și **byte**.

memcached. Sistem de *caching* dezvoltat de Danga Interactive și distribuit sub o licență de tipul PFSL (*Permissive Free Software Licence*). Este folosit de un număr mare de site-uri binecunoscute (e.g., *YouTube*, *LiveJournal*, *SourceForge*, *Wikipedia* și *Facebook*). Pentru utilizare în PHP, sistemul este implementat sub forma unui modul PECL. A se vedea și **PECL** și **PHP**.

Memex. Dispozitiv foto-electro-mecanic imaginar propus de Vannevar Bush în articolul „As We May Think”, publicat în anul 1945 în revista cu apariție lunară *The Atlantic*. Acesta era un sistem de memorare a informațiilor în care utilizatorii aveau posibilitatea să creeze linii (trasee) informaționale, legături spre documente sau ilustrații corelate, care să poată fi stocate și utilizate ulterior. A se vedea și **Bush**, **Vannevar**.

Memletic. Sistem de învățare accelerată (*Memletic Accelerated Learning System*) care dezvoltă teoria lui Colin Rose, identificând un număr mai mare de stiluri de învățare: vizual (spațial) – studentul preferă imagini, scheme și aranjamente spațiale; aural (auditiv-muzical) – studentul preferă să învețe utilizând diverse materiale de învățare bazate pe sunete (discursuri, muzică); verbal (lingvistic) – sunt preferate cuvintele, atât sub formă vorbită, cât și scrisă; fizic (kinestezic) –

studentul preferă să-și utilizeze corpul, mâinile și simțul tactil; logic (matematic) – studentul utilizează logica, demonstrațiile și sistemele conceptuale; social (interpersonal) – este preferată învățarea în grup, bazată pe interacțiune cu alții; solitar (intrapersonal) – studentul lucrează singur, preferând studiul individual.

Sistemul *Memletic* îi ajută pe oameni să învețe mai repede și să-și amintească mai mult, fiind adoptat de un număr mare de instituții care activează în domeniul învățământului *online*. A se vedea și **accelerated learning**.

memorie (*memory*). Partea unui sistem de calcul în care se păstrează programele și datele. Cele mai importante caracteristici ale memoriei sunt: amplasarea (în CPU, memorii externe, memorii interne), capacitatea (dimensiunea cuvântului, numărul de cuvinte), unitatea de transfer (cuvântul, blocul), metoda de acces (acces secvențial, direct, aleator, asociativ), performanțe (timpul de acces, durata ciclului, rata de transfer), tipul memoriei (memorie cu semiconductori, memorie magnetică), caracteristici fizice (volatile sau nevolatile, cu sau fără posibilitate de ștergere).

Memoria internă principală păstrează temporar programele și datele. Pentru operațiile interne ale CPU sunt utilizați regiștrii. Memoria internă principală este extinsă cu o memorie-tampon rapidă, cu dimensiuni mici, numită memorie *cache*.

Pentru păstrarea datelor și a fișierelor de programe pe termen lung sunt utilizate memoriile externe, cele mai folosite fiind discurile magnetice. De asemenea, discurile asigură o extensie a memoriei principale, denumită memorie virtuală.

Cele mai rapide memorii sunt regiștrii CPU, urmate de memoria *cache*, memoria principală și discul magnetic. Cu cât memoria este mai lentă (*i.e.*, crește timpul de acces), cu atât scade costul pe bit, crește capacitatea și scade frecvența cu care CPU o accesează. A se vedea și **CPU, memorie cache și memorie virtuală**.

memorie cache (*cache memory*). A se vedea **cache de CPU**.

memorie fizică (*physical memory*). Cantitatea de memorie RAM din sistem, diferită de memoria virtuală. Denumită și memorie primară, este utilizată pentru păstrarea programelor și a datelor în vederea utilizării lor de unitatea centrală de prelucrare (CPU). A se vedea și **CPU, memorie virtuală și RAM**.

memorie flash (*flash memory*). Memorie non-volatilă care poate fi ștearsă și rescrisă, inventată în 1980 de Fujio Masuoka (de la Toshiba). Este un tip particular de memorie EEPROM. Memoria flash este utilizată pentru stocarea datelor și transferul acestora între calculator și alte dispozitive (*e.g.*, cameră foto digitală, telefon mobil). Tehnologia este folosită în cardurile de memorie și în unitățile flash USB. A se

vedea și **card de memorie, EEPROM și unitate flash USB**.

memorie semiconductoare (*semiconductor memory*). Memorii realizate pe baza materialelor semiconductoare. Principalele tipuri de memorii semiconductoare sunt: RAM (memorie volatilă, permite ștergerea și scrierea pe cale electrică), ROM (memorie nevolatilă, nu permite ștergerea, permite scrierea prin măști), PROM (memorie nevolatilă, nu permite ștergerea, permite scrierea pe cale electrică), EPROM (memorie nevolatilă, permite ștergerea folosind lumină ultravioletă, permite scrierea pe cale electrică) și EEPROM (memorie nevolatilă, permite ștergerea și scrierea pe cale electrică). A se vedea și **EEPROM, EPROM, PROM și RAM**.

memorie video (*video memory*). Memorie situată în mod obișnuit pe placa grafică, utilizată pentru afișarea imaginii pe ecran. A se vedea și **placă grafică**.

memorie virtuală (*virtual memory*). Memoria „văzută” de o aplicație. Poate fi simulată parțial de un dispozitiv de stocare (*e.g.*, hard-disk). A se vedea și **dispozitiv virtual și memorie fizică**.

memorie volatilă (*volatile memory*). Memorie care își pierde conținutul (date și programe) odată cu oprirea alimentării (oprirea calculatorului). Cel mai cunoscut tip de memorie volatilă este RAM. A se vedea și **memorie și RAM**.

meniu (*menu*). Listă de opțiuni oferită de o aplicație, din care utilizatorul poate selecta la un moment dat, folosind *mouse*-ul, una singură, în vederea efectuării unei acțiuni. În multe aplicații, meniul reprezintă o alternativă grafică la comenzi, mult mai ușor de învățat de utilizatori. A se vedea și **meniu contextual**, **meniu în cascadă** și **submeniu**.

meniu contextual (*context-sensitive menu*). Meniu în care sunt incluse opțiunile disponibile într-un anumit context. Cele mai multe interfețe grafice utilizează butonul drept al *mouse*-ului pentru a afișa aceste meniuri, în timp ce butonul stâng este utilizat pentru interacțiunea cu obiectele deja vizibile în interfață. Meniul oferă adesea funcții disponibile în bara de meniuri, rolul lui fiind acela de a furniza acces rapid la un subset de funcții relevante pentru zona în care a fost executat click cu butonul drept al *mouse*-lui. A se vedea și **meniu**.

meniu în cascadă (*cascading menu*). Sistem ierarhizat de meniuri grafice în care, atunci când indicatorul *mouse*-ului este plasat pe o categorie principală, este afișat un meniu lateral incluzând subcategoriile. Un exemplu sugestiv este meniul *Start* utilizat de sistemul de operare Windows. De asemenea, acest tip de meniu este utilizat în numeroase aplicații pentru *desktop*, precum și de unele site-uri Web. A se vedea și **meniu**.

meniu pop-up (*pop-up menu*). În interfețele grafice GNOME și

Microsoft Windows, termenul este echivalent cu cel de meniu contextual. În Mac OS, termenul se referă la un *widget* similar listelor *drop-down*. A se vedea și **meniu contextual**.

MERLOT (*Multimedia Educational Resource for Learning and Teaching*). Site care oferă o colecție de materiale de învățare *online*, scopul său fiind acela de a deveni o comunitate în care facultățile, educatorii și studenții din întreaga lume își împărtășesc materialele de învățare și experiența pedagogică (<http://www.merlot.org/merlot/>).

Activitățile MERLOT sunt bazate pe colaborarea și sprijinul membrilor individuali, al instituțiilor partenere, al partenerilor din mediul de afaceri și al consiliului editorial. Membrii MERLOT formează comunități în funcție de disciplinele de studiu (e.g., biologie, chimie, tehnologia informației, matematică, fizică). Conținutul este oferit sub licențe *Creative Commons*. A se vedea și licențe **Creative Commons**.

mesaj (*message*). 1. În comunicații, unitatea de comunicare între două dispozitive. În general, un mesaj include un bloc de date și un antet, cel din urmă fiind necesar pentru trimiterea mesajului de la sursă la destinație, dar și pentru a transmite alte informații. Un mesaj poate fi transmis integral sau poate fi fragmentat. A se vedea și **antet de mesaj**. 2. În sistemele de operare bazate pe mesaje (e.g., Windows), informație transmisă între programe, dispozitive și sistemul de operare.

mesaj de e-mail (*e-mail message*). Folosit pentru comunicare în cadrul serviciului de poștă electronică, mesajul constă din două părți: antet (*header*), structurat în mai multe câmpuri, și corp (*body*), acesta fiind mesajul propriu-zis; mesajul conține uneori la sfârșit un bloc-semnătură. Conținutul poate fi atât text simplu (*plain text*), cât și text formatat. Mesajului îi pot fi atașate unul sau mai multe fișiere, în diverse formate.

Iată câteva dintre câmpurile incluse în antetul unui mesaj de *e-mail*: *From* (conține adresa de *e-mail* a expeditorului); *To* (conține adresa de *e-mail* a destinatarului, care trebuie să fie cunoscută de expeditor pentru a putea trimite mesajul); *Subject* (descriere scurtă a conținutului mesajului); *Date* (data la care a fost trimis mesajul, completată automat de programul-client); *Reply-to* (adresa la care expeditorul dorește să primească răspunsul la mesajul respectiv); *Organization* (numele organizației, companiei etc. care deține calculatorul de pe care s-a trimis mesajul); *Message-ID* (șir de identificare generat de agentul de transport la trimiterea mesajului); *Cc* (*Carbon copy* – copie la indigo; utilizatorii ale căror adrese de *e-mail* figurează în acest câmp vor primi și ei câte o copie a mesajului); *Bcc* (*Blind carbon copy* – ca și pentru *Cc*, utilizatorii ale căror adrese de *e-mail* figurează în acest câmp vor primi și ei câte o copie a mesajului; totuși, niciunul dintre destinatari nu va ști cine a mai primit o astfel de copie); *Fișiere-*

semnătură (cale de adăugare a unor informații suplimentare; de cele mai multe ori sunt folosite pentru a include informații despre utilizator și despre modul în care poate fi contactat). A se vedea și **adresă de e-mail și poșta electronică**.

metacarakter (*metacharacter*). Caracter cu semnificație specială. Pentru a fi folosite, metacaracterele trebuie citate (*quote*, în limba engleză), prin plasarea caracterului *backslash* („\”) înaintea fiecăruia dintre ele. Metacaracterele pot fi utilizate în diverse scopuri; astfel, în expresiile regulate sunt folosite pentru alcătuirea unor construcții, cu rolul de a identifica anumite secvențe de caractere într-un text. De exemplu, caracterul „|” reprezintă *operatorul de alternare*, fiind utilizat pentru a specifica o listă de opțiuni alternative, ca în acest exemplu: expresia `/marca(nt|je)/` desemnează unul dintre cuvintele *marca*, *marcant* și *marcaje*. Între parantezele rotunde sunt incluse subtipare (*subpatterns*). A se vedea și **caracter**.

metalimbaj (*metalanguage*). Limbaj utilizat pentru crearea sau descrierea altor limbaje. În contextul Web-ului, XML este un metalimbaj din care sunt derivate alte limbaje, precum XSL, XSLT, RSS, Atom. A se vedea și **XML**.

meta refresh (*meta refresh*). Metodă utilizată pentru a instrui browserul să facă un *refresh* automat al paginii Web încărcate în browser, după un interval precizat. În acest scop se utilizează

marcajul (X)HTML meta și atributele `http-equiv` și `content` ale acestuia. Valoarea primului atribut trebuie să fie "refresh", în timp ce valoarea celui de-al doilea, exprimată în secunde, reprezintă intervalul menționat mai sus. Este posibil ca browserul să fie instruit ca, după scurgerea intervalului precizat, să încarce o pagină diferită de cea curentă, prin includerea URL-ului acesteia în valoarea atributului `content`. Iată un exemplu în care este utilizată această metodă:

```
<meta http-equiv="refresh"
      content="3;
      url=http://www.exemplu.ro/"
/>
```

A se vedea și **doorway page** și **spamdexing**.

Metcalf, Robert Melancton (n. 1946). Inginer american, coinventator al standardului Ethernet. În anul 1979 a fondat 3Com, companie producătoare de dispozitive pentru rețele de calculatoare (<http://www.3com.com/>). A se vedea și **Ethernet**.

metodă abstractă (*abstract method*). Denumită și *metodă virtuală* (*virtual method*) sau *pură* (*pure method*), este un concept utilizat în paradigma programării orientate pe obiect (POO), desemnând o metodă care nu este implementată (*i.e.*, nu este definită) în clasa din care face parte, ci într-o clasă derivată. Metodele abstracte nu pot fi apelate, rolul lor fiind doar acela de a declara tipul, numele și parametrii unor metode care urmează să fie redefinite concret în fiecare clasă specializată

derivată din clasa de bază. A se vedea și **clasă abstractă**, **moștenire** și **programare orientată pe obiect**.

Microblogging (*microblogging*). Formă de *blogging* care le permite utilizatorilor să publice *online* texte scurte, de până la 140-200 de caractere (numite *actualizări* sau *tweets*), însoțite uneori și de imagini. Informațiile pot fi editate/accesate *online* sau trimise telefonic prin SMS (*Short Message Service*), prin *e-mail* sau prin IM (mesageria instantanee). În mod obișnuit, însemnările de pe microbloguri pot fi încorporate prin intermediul unor *widget*-uri pe bloguri sau site-uri. Cel mai important avantaj al aplicațiilor de *microblogging* constă în asigurarea unei prezențe sociale pentru utilizatori. Aceștia folosesc aplicațiile de acest tip pentru a comunica, a pune întrebări, a cere recomandări etc., dar și pentru crearea unor note personale. Fiecare utilizator poate monitoriza însemnările altor utilizatori, care apar apoi listați în profilul său.

Iată câteva aplicații de *microblogging*: *Twitter*, *Jaiku*, *Tumblr*. Dintre aplicațiile românești pot fi amintite *Cirip* și *Logoree*. Cele mai multe aplicații de *microblogging* au apărut datorită succesului înregistrat de *Twitter*. Acestea au adăugat servicii specifice în funcție de țară sau combinând facilități ale *microblogging*-ului cu alte servicii, cum ar fi cele de *filesharing* (*e.g.*, *Pownce*, *Jaiku*). Site-urile populare de genul *Facebook* sau *MySpace* au, de asemenea, o

caracteristică *microblogging* denumită *Status Update*.

Deși *microblogging*-ul este o tehnologie Web 2.0 colaborativă recent apărută, aceasta poate avea numeroase utilizări în domeniul educațional. Posibilele utilizări pedagogice ale acestei tehnologii sunt: crearea de comunități educaționale bazate pe principiile metodologiei comunicative; exprimarea reacției studenților; suport pentru conferințe și alte manifestări; crearea de experiențe de învățare în grup; facilitarea claselor virtuale de discuții. A se vedea și **Web 2.0**.

microcalculator (*microcalculator*). Calculator a cărui unitate centrală este implementată ca un microprocesor, având următoarele caracteristici: are un preț relativ redus, fiind accesibil utilizatorilor obișnuiți; utilizatorul final îl poate utiliza fără a avea o pregătire de specialitate; este universal, putând fi utilizat într-un număr mare de domenii de activitate; are o structură modulară, organizată în jurul unei magistrale; poate fi utilizat independent de alte microcalculatoare sau cuplat cu acestea în rețele. După forma unității centrale, microcalculatoarele pot fi: *desktop*, *mini-*, *midi-* sau *big-tower* și *laptop* (denumit și *notebook*). A se vedea și **calculator** și **calculator personal**.

microcod (*microcode*). Set de instrucțiuni având cel mai jos nivel, implicate în punerea în execuție a instrucțiunilor în cod-mașină. Microcodul traduce instrucțiunile de mașină în operațiuni detaliate la nivel de

circuit. Totodată, face posibilă separarea instrucțiunilor-mașină de operațiunile electronice de bază, astfel că instrucțiunile pot fi proiectate și modificate mai liber. De asemenea, permite construirea unor instrucțiuni complexe, reducând în același timp complexitatea circuitelor electronice.

Scrierea de microcod se numește *microprogramare*, iar microcodul pentru un procesor dat este numit adesea *microprogram*. A se vedea și **cod-mașină**.

microformate (*microformats*). Sunt componente Web 2.0 utilizate pentru specificarea elementelor de semantică. Standardizarea microformatelor este în lucru, unele fiind complet specificate (*e.g.*, *hCard*, *hCalendar*, *XFN*, *XOFO*), iar altele se află într-un stadiu intermediar (*e.g.*, *adr*, *geo*, *hAtom*, *rel-license*, *rel-tag*, *rel-enclosure*).

Microformatele („deschise” și disponibile în mod liber, indiferent de scopul în care sunt utilizate) sunt convenții simple pentru încapsularea elementelor de semantică în cadrul elementelor (X)HTML (folosind clase și atribute), urmărind utilizarea sub formă de metadata a conținutului existent pe Web. Microformatele utilizează aceste limbaje, atributele *class*, *rel* și *rev* indicând prezența metadatelor.

Microformatele sunt utilizate pentru a face documentele Web mai „inteligente”, având ca scop furnizarea de informații (*e.g.*, coordonate geografice, descriere de evenimente, detalii de contact) care să poată fi procesate

automat prin intermediul unor aplicații. Acestea nu vor afecta modul în care paginile Web sunt procesate și afișate de browser. Se poate spune că microformatele sunt „felii” de informație „injecate” în documente utilizate în Web. Pe măsură ce acestea vor fi adăugate unui număr din ce în ce mai mare de documente, vor fi disponibile pe Web informații valoroase care pot fi utilizate pentru căutare, indexare, extragere, salvare sau combinare. Astfel, folosirea și procesarea microformatelor permit ca informațiile să poată fi reutilizate și combinate.

Principiile de proiectare care au stat la baza dezvoltării microformatelor pot fi sintetizate sub forma „*reduce, reuse and recycle*” („*reducere, refolosire, reciclare*”). Microformatele nu reprezintă singura soluție pentru a furniza date mai inteligente pe Web, altele (mai sofisticate) fiind utilizarea limbajelor XML și RDF, precum și a Web-ului semantic, care însă nu sunt compatibile cu toate cele trei principii amintite. Aceste soluții se impun totuși atunci când tipul de date care trebuie încorporat nu corespunde niciunui microformat existent.

Microformatele sunt deja folosite pe multe site-uri/aplicații, inclusiv sociale (e.g., *facebook*, *Twitter*, *flickr*, *WordPress*, *Yahoo!*, *Linkedin*, *Google Maps*, *digg*, *BBC*, *W3C*, *Wikipedia*, *AOL*), existând implementări de clase/aplicații pentru procesarea/generarea formatelor uzuale

(<http://microformats.org/wiki/implementations>). Un exemplu de aplicație care operează cu microformate constă în transferarea automată a datelor personale de contact și a listei de prieteni de pe o rețea socială pe alta.

microinstrucțiune (*microinstruction*). Instrucțiune inclusă în microcod. A se vedea și **microcod**.

microprocesor (*microprocessor*). A se vedea **CPU**.

Microsoft Corporation. Corporație multinațională cu sediul în SUA (denumită inițial Micro-Soft), fondată în 1975 de Bill Gates și Paul Allen, activează în domeniul calculatoarelor (<http://www.microsoft.com/>). Microsoft dezvoltă, produce și licențiază un evantai larg de produse software pentru dispozitivele de calcul (e.g., sisteme de operare, pachete *Office*). A se vedea și **Microsoft Office**, **Windows** și **Gates, William Henry III**.

Microsoft Office. Suită de programe de birou creată de compania Microsoft, cunoscută și ca *MS Office*. Are, începând cu versiunea *Office 97*, și o variantă în limba română.

MS Office 2003 și versiunile anterioare utilizează formate binare de fișiere, proprietare, cu extensiile *.doc* (create cu procesorul de texte *Word*), *.xls* (create cu programul de calcul tabelar *Excel*), *.ppt* (create cu aplicația pentru prezentări *PowerPoint*) etc. Începând cu versiunea *Office 2007*, este utilizat OOXML (*Office Open XML*), format bazat pe XML. A se vedea și **Office Open XML**.

Microsoft Photo Editor. Editor grafic inclus în suita de programe de birou produsă de Microsoft, începând cu versiunea *Office 97*, până la *Office XP*, inclusiv. A se vedea și **Microsoft Office**.

Microsoft Visio. Program pentru realizarea diagramelor în format vectorial, creat de Microsoft pentru platforma Windows. Visio este oferit în două ediții: *Standard* și *Professional*. Microsoft a achiziționat compania Visio – cea care a dezvoltat aplicația *Visio* – în anul 2000. A se vedea și **Microsoft**.

Microsoft Visual Studio. IDE dezvoltat de compania Microsoft, utilizat pentru crearea aplicațiilor cu interfață grafică, precum și a site-urilor, aplicațiilor și serviciilor Web. A se vedea și **IDE**.

Microsoft Visual Studio Express. Set de medii de dezvoltare integrate (IDE) oferit gratuit de Microsoft. Include: *Visual Basic Express*, *Visual Web Developer Express*, *Visual C++ Express*, *Visual C# Express* și *SQL Server Express*. A se vedea și **IDE**.

MIDI (*Musical Instrument Digital Interface*). Standard de interfață serială care oferă posibilitatea conectării la calculatoare a instrumentelor muzicale electronice (e.g., claviaturi). MIDI nu transmite semnale sau informații audio, ci așa-numitele *mesaje de eveniment* numite și *mesaje MIDI* (e.g., înălțimea notei de reprodus, intensitatea acesteia,

semnale de control al unor parametri). Dispozitivele conectate la calculator prin această interfață, numite dispozitive MIDI, pot fi utilizate la crearea, înregistrarea și redarea muzicii.

migrarea datelor (*data migration*). Procesul de transfer automat al datelor între diverse medii de stocare, formate, baze de date, aplicații sau calculatoare. Este utilizat în variate situații (e.g., schimbarea sau îmbunătățirea calculatoarelor, schimbarea sau actualizarea aplicațiilor). Procesul implică extragerea și încărcarea datelor, urmate de verificarea acestora.

Milgram, Stanley (1933–1984). A efectuat în perioada 1967–1969 o serie de experimente pentru a examina lungimea medie a căilor de interconectare în rețelele sociale din Statele Unite. Rezultatele cercetărilor au sugerat că societatea umană este o lume/rețea caracterizată prin lungimi mici ale căilor de interconectare. Experimentele sale sunt adesea asociate cu teoria celor șase grade de separare, deși Milgram nu a utilizat niciodată această expresie.

Experimentele au constatat în următoarele: cetățeni americani, aleși întâmplător, au fost rugați să trimită pachete unui străin căruia îi cunoșteau doar numele, ocupația și orașul, prin acea cunoștință despre care au presupus că ar fi cel mai posibil să-l cunoască pe destinatar. Noii expeditori aveau aceeași sarcină, procesul continuând până când pachetul ajungea la

destinatar. Experimentele au demonstrat că, în ciuda părerilor participanților, care credeau că este nevoie de minimum 100 de persoane, sunt necesare – în medie – circa 6,2 persoane pentru ca pachetul să ajungă la destinatar. A se vedea și **teoria celor șase grade de separare**.

MIME (*Multipurpose Internet Mail Extensions*). Standard Internet (RFC 1049) creat în scopul de a extinde formatul mesajelor de *e-mail* astfel încât să permită mesaje textuale non-ASCII, atașamente non-textuale, mesaje non-textuale și includerea unor informații non-ASCII în anteturile mesajelor. Dincolo de scopul inițial, MIME este utilizat astăzi pentru descrierea tipului oricărui conținut livrat în Internet, inclusiv în Web.

Standardul MIME (inclus în protocolul HTTP) definește câmpul `Content-Type` al antetului HTTP al cererii clientului și al răspunsului serverului, câmp care descrie natura și formatul datelor solicitate/trimise. Utilizarea standardului permite transmiterea, pe lângă textul simplu (*plain text*), și a altor informații (audio, video, imagini etc.).

Datele MIME aparțin unui tip și unui subtip. În general, tipul este utilizat pentru a declara natura datei, iar subtipul specifică un format pentru tipul de date utilizat. Tipul conținutului poate fi: `application`, `audio`, `image`, `text`, `video`, `message` și `multipart` (primele cinci sunt tipuri discrete, iar ultimele două sunt tipuri compuse).

Fiecare tip conține mai multe subtipuri. De exemplu tipul `text` conține subtipurile `plain` și `html`. Un subtip poate fi însoțit de parametri (fiecare parametru este precizat ca *nume_parametru = valoare_parametru*). Numele tipurilor, subtipurilor și parametrilor nu sunt case-sensitive, iar valorile parametrilor sunt sau nu case-sensitive, de la un caz la altul. Precizarea unui tip, a unui subtip și a unui parametru MIME se face – ca în exemplele următoare – folosind caracterul *slash* (/) între numele tipului și cel al subtipului și caracterul punct și virgulă (;) între numele subtipului și cel al parametrului:

```
Content-type: text/plain;
charset=us-ascii
Content-type: text/html
Content-type: image/gif
Content-type: image/jpeg
Content-type:
application/msword
Content-type: application/x-tar
Content-type: audio/wave
Content-type: video/mpeg
```

Mind Mapping. Metodă/tehnică de optimizare a capacităților de a înțelege conectarea elementelor incluse în structuri complexe. Tehnica poate crește eficiența învățării/studiului cu până la 15% în raport cu metodele convenționale.

miniatură (*thumbnail*). Versiune cu dimensiuni reduse a unei imagini, care permite răsfoirea rapidă a unui catalog. Sunt utilizate pe scară largă în site-urile Web și în motoarele de căutare vizuale. De asemenea, miniaturile sunt utilizate

în programele folosite pentru organizarea imaginilor în sistemele de operare moderne și în mediile *desktop*.

(a) **minimiza** (*minimize*). Referitor la interfețele grafice cu utilizatorul, este procesul prin care se ascunde o fereastră fără a fi oprită rularea programului care o folosește. În acest scop este utilizat un buton dedicat. Prezența unei ferestre minimizezate este indicată în continuare în bara de task-uri, sub forma unei pictograme. Atunci când utilizatorul execută click pe aceasta, fereastra este readusă la dimensiunea inițială, de dinainte de minimizare. A se vedea și **fereastră** și (a) **maximiza**.

Minix. Sistem de operare *open source* creat de profesorul Andrew Tanenbaum în scopuri educaționale. Numele său provine din cuvintele *minimal* și *UNIX*. Minix a constituit o sursă de inspirație pentru Linux Torvalds în crearea sistemului de operare Linux. A se vedea și **Linux**, **Tanenbaum**, **Andrew** și **UNIX**.

Misspelling (*greșeală de ortografie*). În Web, se referă la ortografierea greșită de către utilizatori a unor expresii-cheie folosite în interogările realizate cu motoarele de căutare. Unii dezvoltatori Web își optimizează site-urile pentru unele cuvinte scrise greșit, cu scopul de a atrage un număr mai mare de utilizatori.

MIT (*Massachusetts Institute of Technology*). Universitate tehnică aflată în localitatea Cambridge din statul american Massachusetts, fondată în

1861 de William Barton Rogers (<http://mit.edu>). Este alcătuită din cinci școli tehnice și un colegiu, care conțin 32 de departamente științifice. Universitatea este orientată pe cercetarea științifică și tehnologică, teoretică și aplicativă, deviza urmată încă de la înființare fiind *learning by doing* („a învăța prin practică”). *Motto*-ul MIT este *Mens et Manus* („minte și mâini”). Un număr mare de absolvenți și profesori ai universității au câștigat de-a lungul anilor premii Nobel în diverse domenii. Majoritatea cursurilor predate la MIT sunt publicate *online* gratuit (*OpenCourseWare*, <http://ocw.mit.edu>). Există și o asociație a absolvenților MIT (*MIT Alumni Association*, <http://alum.mit.edu>). Site-ul pentru admiterea la MIT este *MIT Admissions* (<http://www.mitadmissions.org/>). Harta campusului MIT se găsește la adresa <http://whereis.mit.edu/map-jpg>.

model non-interferențial (*non-interference model*). Model de sistem de calcul care nu permite ca un utilizator să interfereze cu resursele sau procesele unui alt utilizator. Cele mai multe sisteme de calcul moderne urmează într-o anumită măsură acest model.

modelul bazei de date (*database model*). Numită și *schemă* (*database schema*) sau *model conceptual*, reprezintă o specificație referitoare la structura (organizarea) unei baze de date. Cele mai cunoscute modele sunt: ierarhic, rețea, relațional, obiectual și hibrid (obiectual-relațional).

modelul client-server. A se vedea **arhitectura client-server.**

modelul de referință ISO/OSI (*reference model ISO/OSI*). A se vedea **ISO/OSI.**

modelul entitate-relație (*ERM, Entity-relationship model*). Model utilizat în etapa de proiectare a unei baze de date. Presupune parcurgerea următorilor pași: identificarea coloanelor, luând în considerare datele care trebuie stocate în baza de date; gruparea coloanelor în tabele, astfel încât coloanele corelate să fie incluse în același tabel (este posibil ca o coloană să se regăsească în mai multe tabele); identificarea cheilor primare, oricare dintre acestea fiind reprezentată de una sau mai multe coloane existente sau de o nouă coloană creată special în acest scop; identificarea cheilor externe.

modelul relațional (*relational model*). Model de organizare a datelor sub formă de relații (sau tabele). A se vedea și **bază de date relațională.**

modem (*modulator-demodulator*). Dispozitiv care modulează un semnal analog purtător pentru a codifica informații în format digital și, de asemenea, demodulează un semnal purtător pentru a decodifica informațiile transmise. Scopul în care au fost dezvoltate modemurile este de a produce un semnal care poate fi transmis ușor și decodificat, pentru a reproduce informațiile inițiale.

modem de cablu (*cable modem*). Tip

de modem care asigură o comunicație de date bidirecțională pe o infrastructură de cablu TV (CATV). Modemurile de cablu sunt utilizate în primul rând pentru accesul în bandă largă la Internet (*broadband*). A se vedea și **modem.**

modulare (*modulation*). Procesul prin care amplitudinea, frecvența sau faza unei unde electromagnetice este schimbată pentru a transmite date. Modularea combină un semnal de date și o undă purtătoare pentru transmisie prin rețea.

Cele mai utilizate metode de modulare sunt: modularea în amplitudine (AM, *amplitude modulation*), modularea în frecvență (*frequency modulation*) și modularea în fază (*phase modulation*).

Moisil, Constantin Grigore (1906–1973). Matematician român, membru al Academiei Române, al Academiei din Bologna și al Institutului Internațional de Filozofie. A publicat lucrări în domeniile mecanicii, analizei matematice, geometriei, algebrei și logicii matematice. De asemenea, a elaborat metode noi de analiză și sinteză a automatelor finite și a avut contribuții valoroase în domeniul teoriei algebrice a mecanismelor automate. Este considerat părintele informaticii românești.

Moodle. CMS și pachet software cu sursă deschisă, furnizat sub licența GNU GPL, proiectat cu scopul de a-i ajuta pe educatori în crearea unor comunități de studiu *online* eficiente

(<http://moodle.org/>). CMS-ul poate fi instalat pe un computer care rulează PHP și care suportă baze de date SQL (e.g., MySQL), pe platformele Windows, Mac și pe un mare număr de versiuni Linux (e.g., RedHat, Debian GNU). A se vedea și **CMS**.

Moore, Gordon Earle (n. 1929). Fizician și chimist american, cofondator al Intel Corporation (1968). Într-un articol publicat în *Electronics Magazine* (19 aprilie 1965) a formulat așa-numita lege a lui Moore, referitoare la ritmul de creștere a numărului de tranzistoare care pot fi plasate fără costuri mari în circuitele integrate. A se vedea și **Intel Corporation** și **legea lui Moore**.

Mosaic. Primul browser Web grafic utilizat pe scară largă, creditat pentru răspândirea Web-ului. De asemenea, este client și pentru alte servicii timpurii (e.g., FTP, Usenet, Gopher). A fost creat între 1992 și 1993 la NCSA (*National Center for Supercomputing Applications*) de Mark Andreessen și Eric Bina, dezvoltarea sa fiind oprită în 1997. Totuși, browserul poate fi descărcat și astăzi de pe site-ul NCSA (<ftp://ftp.ncsa.uiuc.edu/Web/Mosaic/>), fiind oferite versiuni pentru Mac, UNIX și Windows. A se vedea și **Andreessen, Marc**.

moștenire (*inheritance*). În programarea orientată pe obiect (POO), reprezintă un procedeu de derivare a unei clase dintr-una sau mai multe clase de bază. În ultimul caz, se vorbește

despre moștenire multiplă. A se vedea și **clasă de bază** și **programare orientată pe obiect**.

motherboard (*placă de bază*). A se vedea **placa de bază**.

motor (*engine*). În informatică, componenta principală a unui program, care stabilește modalitățile de gestionare a datelor și funcționalitățile acestuia. Este diferită de aspectele periferice ale programului, de exemplu interfața cu utilizatorul. A se vedea și **interfața om-calculator**.

motor de căutare (*search engine*). Cel mai popular instrument de găsimă a informațiilor pe Web. Astăzi, piața motoarelor de căutare este dominată de *Google*, *Yahoo! Search* și *MSN/Windows Live*. În mod obișnuit, un motor de căutare este alcătuit din trei părți: un *spider*, un *indexer* și un procesor de interogare (*query processor*).

Deși la începuturile utilizării lor motoarele de căutare erau similare în ceea ce privește structura, modul de funcționare și natura informațiilor furnizate, odată cu trecerea timpului au apărut diferențieri. Iată o clasificare a motoarelor de căutare valabilă astăzi: motoare de căutare dedicate (e.g., *Google*, *Yahoo! Search*, *MSN/Windows Live*); metamotoare de căutare (e.g., *DogPile*, *MetaCrawler*, *Mamma*, *WebCrawler*, *Sputtr*); motoare de căutare cu limbaj natural (e.g., *Excite*, *AskJeeves*); motoare de căutare vizuale (e.g., *Kartoo*, *Quintura*, *SearchCrystal*);

motoare de căutare specializate (e.g., căutare în biblioteci, imagini, muzică, bloguri, wiki-uri, grupuri de discuții); motoare de căutare destinate găsirii informațiilor educaționale (e.g., *Infomine*, *Google Scholar*, *Education SearchEngine*, *Tutorial SearchEngine*, *NorthernLight*). A se vedea și **spider**, **indexer** și **procesor de interogare**.

motor de căutare în feed-uri (*feed search engine*). Motor specializat care realizează căutarea în *feed*-uri (*search feed*), implementat în mod obișnuit ca modul într-o aplicație (e.g., agregator, director de *feed*-uri). Unele motoare creează pentru fiecare căutare câte un *feed* RSS, numit *Search Feed*. Astfel, nu mai este necesară accesarea periodică a motorului pentru a relua căutarea, fiind suficientă abonarea la *feed*-ul creat anterior. De asemenea, pe lângă termeni, se pot căuta și legături. Dintre sistemele/aplicațiile care oferă facilitatea *Search Feed*, pot fi menționate: *Bloglines*, *delicious*, *Technorati* și *Moreover*. A se vedea și **agregator**, **director de feed-uri** și **feed**.

motor de căutare socială (*social search engine*). Tip de motor de căutare care determină relevanța rezultatelor căutării prin cuantificarea interacțiunilor și contribuțiilor utilizatorilor. Metoda și algoritmul utilizate sunt diferite de cele folosite de motoarele de căutare clasice, în care relevanța este determinată prin analiza textului paginilor și a structurii de legături incluse în acestea.

Deși motoarele de căutare socială nu au demonstrat că algoritmul utilizat este superior celui folosit în motoarele clasice, motoarele de acest tip aduc o serie de beneficii: creșterea relevanței rezultatelor returnate, deoarece acestea au fost selectate de utilizatori; paginile Web sunt considerate relevante din perspectiva cititorilor, și nu a creatorilor acestora; furnizarea unor rezultate actuale, acestea fiind constant îmbunătățite de utilizatori. A se vedea și **căutare socială** și **motor de căutare**.

mouse (*mouse*). Dispozitiv de intrare utilizat pentru sistemele de operare cu interfață grafică. Este folosit pentru selectarea unor opțiuni din meniuri, pentru manipularea unor obiecte sau pentru controlul mișcării cursorului pe ecranul de afișare. *Mouse*-ul – inventat în 1963 de Douglas Engelbart – este considerat una dintre marile descoperiri ergonomice, datorită posibilității de a evita folosirea exagerată a tastaturii. Din punct de vedere constructiv, dispozitivele *mouse* pot fi: mecanice (inclusiv o bilă care se poate roti în toate direcțiile; senzorii conținutăți determină direcția și deplasează corespunzător cursorul pe ecran); optice (pentru detectarea mișcării folosesc un laser și o suprafață specială prevăzută cu o grilă; deoarece nu includ componente mobile, au o acuratețe mai mare în ceea ce privește detectarea și interpretarea mișcărilor; sunt mai rapide și mai precise decât cele mecanice); opto-mecanice (folosesc o combinație de tehnologii mecanice și optice). Rotița

de *scroll* asigură funcții similare cu cele ale barelor de derulare dintr-o fereastră Windows. A se vedea și **dispozitiv de intrare**.

Movable Type. Software pentru *blogging*, valabil pentru uz personal, comercial, educațional și nonprofit (<http://www.movabletype.com/>). A se vedea și **blog** și **software pentru blogging**.

Mozilla Firefox. Browser Web *open source* dezvoltat de Mozilla Corporation (<http://www.firefox.com/>). Ocupă locul al doilea în ceea ce privește popularitatea printre utilizatori, după *Internet Explorer*, fiind folosit (în 2009) de peste 20% dintre aceștia. Pentru afișarea paginilor Web, *Firefox* utilizează motorul Gecko (*Gecko engine*), acesta implementând cele mai recente standarde Web. Rulează pe un număr mare de sisteme de operare, printre care Microsoft Windows, Mac OS X, Linux. A se vedea și **browser Web**.

MPEG (*Moving Pictures Experts Group*). Set de standarde utilizate pentru comprimarea și transmisia conținutului audio și video, elaborat de grupul ISO cu același nume. Metodologia MPEG este considerată asimetrică, deoarece codificarea este mai complexă decât decodificarea.

MPEG a standardizat următoarele formate de compresie și standarde auxiliare: MPEG-1 (standardul MPEG original, dezvoltat pentru stocarea și redarea conținutului audio și video pe

CD-ROM), MPEG-2 (extindere a MPEG-1, destinată transmisiunilor de televiziune), MPEG-3 (nu mai este utilizat) și MPEG-4 (utilizează modalități de codificare complexe, prin care se obțin factori de compresie mai mari decât în cazul MPEG-2; în plus, este eficient pentru utilizarea în aplicațiile software grafice).

.mpg. Extensie de fișier pentru formatul specificat de MPEG. A se vedea și **MPEG**.

MSN (*Microsoft Network*). A se vedea **Windows Live**.

multi-browser (*cross-browser*). A se vedea **cross-browser**.

multi-platformă (*cross-platform*). A se vedea **cross-platform**.

multimedia (*multimedia*). Cu referire la o aplicație, atunci când aceasta include cel puțin un element media continuu (*e.g.*, sunet, video, animație) și cel puțin unul discret (*e.g.*, text, imagine). Astăzi, din ce în ce mai multe site-uri Web constituie exemple de aplicații multimedia. Creșterea importantă din ultimii ani a numărului de tipuri de aplicații Web se datorează într-o măsură însemnată și utilizării unui conținut informațional bazat pe elemente multimedia, care atrage un număr important de utilizatori. Aplicațiile dezvoltate pentru învățământ (*e-Learning*), pentru video-conferințe, dar și pentru divertisment (*e.g.*, jocuri) utilizează cele mai noi tehnologii multimedia. A se vedea și **hipermedia**.

multimodalitate (*multimodality*). Reprezintă capacitatea unui sistem (e.g., calculator) de a comunica cu utilizatorul (cvasi)simultan prin mai multe canale de comunicație (cel puțin două) și de a extrage și transmite automat date cu semnificație. Multimodalitatea unui sistem implică o interacțiune multimodală. În acord cu definiția anterioară, practic oricare interacțiune om-calculator poate fi considerată ca multimodală, deoarece practic simultan utilizatorul privește monitorul, tastează comenzi sau deplasează cursorul mouse-ului, aude reacția (e.g., *beep*-uri, *click*-uri) etc.

multiplexare (*multiplexing*). Procedeu prin care semnalele aferente mai multor canale de transmisie sunt combinate într-un singur semnal compozit, în vederea traversării unui mediu de transmisie dat. Iată două exemple: multiplexarea cu diviziune în frecvență (FDM, *Frequency Division Multiplexing*) și multiplexarea cu diviziune în timp (TDM, *Time Division Multiplexing*). Dispozitivul utilizat pentru combinarea canalelor se numește multiplexor. A se vedea și FDM și TDM.

multitasking (*multitasking*). Proprietatea unui sistem de calcul de a executa simultan (*i.e.*, paralel) multiple sarcini (cunoscute și ca procese), prin partajarea unor resurse (e.g., unitatea centrală de calcul, CPU). Este important de precizat că în sistemele în care există un singur CPU, simultaneitatea este numai aparentă, deoarece

aceasta poate executa la un moment dat numai o singură sarcină, celelalte fiind în așteptare. Chiar și în sistemele care includ mai multe CPU-uri, numărul proceselor executate „simultan” este mult mai mare. Actul de schimbare a alocării CPU-ului de la un proces la altul se numește *schimbare de context* (*context switch*). Dacă această schimbare survine frecvent, se creează iluzia paralelismului proceselor, adică a *multitasking*-ului. Sistemele de operare folosesc mai multe strategii de planificare (*scheduling strategies*). Iată câteva dintre acestea: multiprogramarea, multitasking cooperativ, multitasking preemptiv. A se vedea și **multitasking cooperativ** și **multitasking preemptiv**.

multitasking cooperativ (*cooperative multitasking*). Strategie de planificare pentru *multitasking* în care procesele din fundal beneficiază de timp de lucru în perioadele de inactivitate ale proceselor din prim-plan numai atunci când un proces de prim-plan permite acest lucru. A se vedea și **multitasking** și **multitasking preemptiv**.

multitasking preemptiv (*preemptive multitasking*). Strategie de planificare pentru *multitasking* în care sistemul de operare întrerupe periodic execuția unui proces și transferă controlul sistemului către un alt proces, aflat în așteptare. Această strategie elimină posibilitatea ca un proces să monopolizeze controlul sistemului, spre deosebire de *multitasking*-ul cooperativ. A se vedea

și **multitasking** și **multitasking cooperativ**.

multiutilizator (*multiuser*). Caracteristică a unui sistem de calcul care poate fi folosit simultan de mai mulți utilizatori, accesul acestora fiind permis prin diverse sisteme de comunicații sau terminale de rețea (e.g., UNIX: HP-UX, Solaris, FreeBSD; Windows 2000; Windows XP, Windows Vista). A se compara cu *single user system*, acest termen desemnând un sistem care poate fi folosit la un moment dat de un singur utilizator (e.g., MS-DOS, Windows 95, Windows 98).

MVC (*Model-View-Controller*). Arhitectură utilizată de cadrele de lucru software (*software framework*) cu scopul separării datelor (*Model*) de interfața cu utilizatorul (*View*), astfel încât schimbările în interfață să nu afecteze datele, iar acestea să poată fi reorganizate fără schimbări în interfață. Arhitectura MVC rezolvă această problemă prin decuplarea accesului la date și a logicii aplicației de prezentarea datelor și interacțiunea cu utilizatorul, introducând o componentă intermediară numită *Controller*. Rezultă că arhitectura MVC este constituită din trei componente principale: 1) componenta *Model*, care constituie o reprezentare specifică (sub formă de date) a informațiilor cu care operează aplicația; multe aplicații utilizează stocarea persistentă a datelor (e.g., baze de date); 2) componenta *View*, care interpretează modelul într-o formă potrivită pentru interacțiune (în mod obișnuit, într-un element de

interfață); pentru un singur model pot fi dezvoltate mai multe *views*, în diverse scopuri; 3) componenta *Controller*, reprezentând clasele care realizează comunicarea între cele două componente anterioare, procesează evenimentele din interfață (în mod obișnuit, acțiunile utilizatorilor) și răspunde la acestea. Ultima componentă poate invoca schimbări în model.

My Documents. În sistemele de operare Windows (cu excepția Windows NT) este numele unui folder special, utilizat în mod obișnuit pentru stocarea fișierelor utilizatorilor (e.g., documente, imagini, muzică, fișiere descărcate de pe Internet). A se vedea și **folder**.

MySpace. Una dintre cele mai populare rețele sociale *online* (<http://www.myspace.com/>). A se vedea și **rețea socială online**.

MySQL. Dezvoltat de MySQL AB (<http://www.mysql.com/>), este cel mai popular sistem de gestiune a bazelor de date relaționale cu sursă deschisă utilizat în Internet, având versiuni pentru majoritatea platformelor (e.g., UNIX/Linux, Windows, Mac OS X Server). Serverul MySQL este multi-fir și multi-utilizator. Dintre caracteristicile sale pot fi menționate stabilitatea și rapiditatea.

O *distribuție MySQL* conține numai o parte sau toate componentele următoare: serverul *mysqld* (sau diverse variante; e.g., distribuția standard client-server pentru Windows oferă

următoarele servere: *mysqld*, *mysqld-opt*, *mysqld-nt*), care îndeplinește următoarele funcții principale: gestionează bazele de date și conturile de acces, primește cererile clienților, obține accesul la bazele de date, le furnizează clienților informațiile solicitate; programe-client (e.g., *mysql*), utilizate în principal pentru conectarea la server, transmiterea comenzilor și afișarea rezultatelor primite; programe utilitare (e.g., *mysqladmin*, *mysqldump*, *myisamchk*), folosite pentru administrarea sistemului MySQL; fișiere-antet și fișiere-bibliotecă; documentație sub forma unui manual MySQL; baza de date *mysql*, care conține, printre altele, tabelele de acordare a privilegiilor pentru utilizatori. Clientul *mysql* lucrează în modul text, fără interfață grafică. Dacă se dorește utilizarea unor clienți cu interfață grafică, ei trebuie procurați separat (e.g., *SQL-Front*).

Distribuțiile MySQL sunt de două tipuri: distribuții-client, care nu conțin nici un server, dar conțin programele-client necesare pentru transmiterea comenzilor către un server MySQL instalat pe un alt calculator; distribuții-server, care conțin atât programele-server, cât și programele-client și utilitare.

Alături de server, clienți și utilitare, MySQL AB pune la dispoziția utilizatorilor o serie de instrumente cu interfață grafică folosite pentru administrarea serverului și lucrul cu bazele de date, dintre care poate fi amintit *MySQL GUI Tools*, pachet care conține aplicații GUI (*MySQL Administrator*, *MySQL QueryBrowser* și *MySQL MigrationToolkit*). De asemenea, sunt oferite drivere utilizate pentru conectarea la serverul MySQL prin programe scrise în diverse limbaje de programare (e.g., *MySQL Connector/J*, *MySQL Connector/NET*, *Connector/ODBC*).

N

NAT (*Network Address Translation, translatarea adresei de rețea*). Definit în RFC 1631, este procesul de schimbare a unei adrese IP în antetul pachetului IP. În practică, NAT este folosit pentru accesul la Internet al gazdelor cu adrese IP private, înlocuindu-le cu adrese IP publice și invers. Tehnica face posibilă utilizarea unui număr mare de adrese IP private în cadrul unui singur domeniu. A se vedea și **adresă IP**.

navigare vocală (*voice navigation*). Utilizarea comenzilor vocale pentru a controla un browser Web (e.g., *Opera*). Caracteristică a unor aplicații *plugin* care, instalate în browser, îi permit utilizatorului să navigheze prin Web folosind vocea. A se vedea și **browser Web** și **Opera**.

navigator Web (*Web navigator*). A se vedea **browser Web**.

Nelson, Theodor Holm (n. 1937). Cunoscut ca Ted Nelson, este un sociolog și filosof american, pionier al tehnologiei informației. A creat o serie de termeni consacrați astăzi, care stau la baza funcționării Internetului. În 1965, a introdus termenul *hypertext*, un

tip de scriitură non-secvențială ale cărei componente nu sunt create pentru a fi citite într-o ordine prestabilită. În același an, savantul a inventat termenul *hypermedia*, pentru a desemna sisteme hipertextuale care includ elemente media combinate: text, imagini, sunet și video. A se vedea și **hipertext** și **hipermedia**.

nelivrabil (*undeliverable*). Care nu poate fi trimis destinatarului specificat, din diverse motive. De exemplu, în cazul poștei electronice, este posibil ca mesajul să-i fie returnat expeditorului (împreună cu o explicație atașată din partea serverului de *e-mail*) deoarece adresa destinatarului a fost specificată greșit, contul său a fost șters sau (mai rar) căsuța poștală este plină. A se vedea și **poștă electronică**.

NetBIOS (*Network Basic Input/Output System*). API care poate fi utilizat într-o rețea locală, destinat în special programatorilor. Oferă un set de comenzi pentru solicitarea unor servicii, funcționând la nivelul sesiune al modelului OSI și permițând programelor care rulează pe calculatoare diferite să comunice între ele. Cu toate că NetBIOS este o interfață software, și nu un protocol, oferă servicii similare unui protocol. A fost dezvoltat în 1983 de Sytek, Inc. și, ulterior, de IBM.

Există implementări NetBIOS care folosesc TCP sau UDP ca suport de protocol de transport. Problema în acest caz este că numele din NetBIOS nu corespund adreselor utilizate de TCP/IP, ceea ce înseamnă un efort

suplimentar de programare. Standardele RFC 1001 și 1002 descriu utilizarea NetBIOS cu TCP/IP (orice stație adresabilă TCP/IP este adresabilă și NetBIOS).

Serviciul NetBIOS folosește portul 139. În general, acest port trebuie închis (e.g., prin dezactivarea *File Sharing* și *Print Sharing*), pentru a mări securitatea sistemului. A se vedea și **API**, **LAN** și **ISO/OSI**.

NetBSD. Versiune *open source* a sistemului de operare UNIX BSD (<http://www.netbsd.org/>), dezvoltată de NetBSD Foundation și lansată în anul 1993 sub licență BSD. NetBSD este cunoscut pentru portabilitatea sa ridicată, motiv pentru care este utilizat în unele dispozitive mobile (e.g., Pocket PC) și în sistemele încorporate (*embedded systems*). *Motto-ul* NetBSD este *Of course it runs NetBSD*. A se vedea și **BSD** și **Pocket PC**.

.NET Framework. Cadru de lucru software dezvoltat de Microsoft. Poate fi utilizat pe calculatoarele care rulează sisteme de operare Microsoft Windows. Include o mașină virtuală (CLR, *Common Language Runtime*) care gestionează execuția programelor scrise pentru acest cadru de lucru, precum și o bibliotecă (*Base Class Library*) de soluții pentru probleme comune de programare (e.g., acces la date, interacțiune cu bazele de date, criptografie, procesarea documentelor XML, algoritmi numerici, comunicare în rețea) și folosită de programatori (împreună cu propriul cod) pentru

crearea unor aplicații diverse. A se vedea și **Base Class Library** și **Common Language Runtime**.

netichetă (*netiquette*). Cuvânt care provine din expresia *eticheta (Inter)netului*. Reprezintă un set de convenții de comunicare prin mesaje electronice (e.g., postate pe forumuri și bloguri sau trimise prin *e-mail*, *chat* și *messenger*). Regulile de netichetă depind de mijloacele de comunicare utilizate. Cu toate că unele dintre reguli se repetă, ele încearcă totuși să sublinieze diverse aspecte specifice fiecărui mijloc. A se vedea și **netizen**.

netizen (*netățean*). Cetățean al (Inter)netului. Este orice persoană care participă la comunicații *online* efectuate prin intermediul numeroaselor servicii și aplicații Internet, în cele mai multe cazuri oferite gratuit. A se vedea și **netichetă**.

Netvibes. Serviciu pentru personalizarea paginii de start, bazat pe AJAX, având și o interfață în limba română (<http://www.netvibes.com/>). Serviciul a fost fondat în anul 2005 de Tariq Krim și Florent Frémont și reprezintă un instrument de pionierat, alternativă la tradiționalele portaluri Web.

NetVibes aduce laolaltă sursele media și serviciile favorite, oferindu-le utilizatorilor posibilitatea de a asambla site-urile Web și blogurile preferate, conturile de *e-mail*, rețelele sociale folosite, motoarele de căutare favorite, aplicațiile de mesagerie instantanee, fotografiile, clipuri video, *podcast*-uri și

orice alte elemente care îi facilitează activitatea de navigare și comunicare. Acestea sunt incluse în pagina personală prin intermediul *widget*-urilor (numite și *module*). Modulele sunt organizate în taburi (numite și *file*). Filele și modulele pot fi partajate cu alți utilizatori particulari sau via *NetVibes Ecosystem*. Acesta din urmă este o colecție de *widget*-uri trimise de utilizatori, construite folosind *NetVibes Universal Widget API (UWA)*. Rezultă că *NetVibes* este o aplicație Web de tip *all in one* (mai multe servicii într-o singură aplicație). Informațiile incluse în module vor fi automat actualizate ori de câte ori utilizatorul își deschide propria pagină *Netvibes*.

Serviciul *NetVibes* poate fi accesat de pe telefonul mobil la adresa <http://m.netvibes.com/> și de pe *iPhone* la adresa <http://iphone.netvibes.com/>.

Astăzi, *Netvibes* este o comunitate globală de utilizatori care dețin controlul asupra propriilor vieți digitale prin personalizarea experienței Web. Conținutul favorit poate fi partajat cu prietenii și cu alți utilizatori. De asemenea, *NetVibes* este o platformă de *widget*-uri utilizată de un număr foarte mare de editori din lumea întreagă.

Crearea unui cont *NetVibes* presupune furnizarea unei adrese de *e-mail* și a unei parole. La adresa precizată va fi trimis automat un mesaj care confirmă crearea contului. Pagina personală va putea fi accesată ulterior de pe orice calculator, folosind adresa de *e-mail* și parola furnizate la crearea contului.

Un utilizator deține pe *NetVibes* două pagini: una privată și alta publică. La pagina privată are acces numai utilizatorul, după autentificare. În schimb, la pagina privată au acces toți utilizatorii/vizitatorii serviciului *NetVibes*. Utilizatorul își poate partaja (*i.e.*, face public) orice *widget* inclus în pagina sa privată (*Send to my public page*).

Serviciul *NetVibes* și cele similare acestuia pot fi utilizate în educație în diverse scopuri: crearea unui panou de informare rapidă (atât pentru educatori, cât și pentru studenți) în care vor fi incluse *widget*-uri create pe baza celor mai utilizate *feed*-uri sau site-uri Web (de asemenea, există posibilitatea verificării *e-mail*-ului direct în pagina personală); partajarea informațiilor între actorii educaționali; punerea la dispoziția studenților a unor surse de informare care furnizează date și cunoștințe menite să completeze/dezvolte unele cursuri; crearea unui panou de *widget*-uri, utilizat ca sursă de informare pentru dezvoltarea și implementarea unui proiect școlar; crearea unor notițe care includ informații necesare pe termen lung (*e.g.*, orar școlar) sau scurt (*e.g.*, activități diverse, teme). A se vedea și **serviciu pentru personalizarea paginii de start**.

von Neumann, John (1903–1957). Matematician american de origine maghiară. A adus contribuții importante în numeroase domenii, printre care

fizica cuantică, informatică, statistică și teoria mulțimilor.

În 1945, von Neumann a scris articolul *First Draft of a Report on the EDVAC*. Acesta – având 101 de pagini, dar neterminat – conține prima descriere publicată a conceptului de calculator digital care utilizează programe stocate, cunoscut ulterior sub numele de *arhitectură von Neumann*. A se vedea **arhitectura von Neumann** și **EDVAC**.

newsletter (*newsletter*). Publicație distribuită periodic de companii, instituții de presă, edituri, instituții de învățământ, asociații, organizații guvernamentale etc., în scopul promovării propriilor produse, servicii, acțiuni și evenimente. Un tip particular foarte utilizat este așa-numitul *e-newsletter*, livrat prin intermediul serviciului de poștă electronică.

NewsLife. Agregator de *feed*-uri RSS pentru Mac OS, ușor de utilizat (<http://www.thinkmac.co.uk/newslife/>). A se vedea și **agregator** și **feed**.

news reader (*cititor de știri*). A se vedea **agregator**.

NIC (*Network Information Center*). Organizație care oferă informații despre o rețea și le furnizează asistență utilizatorilor rețelei respective. Principala organizație NIC din Internet este InterNIC (<http://www.internic.com/>). Uneori, intraneturile și alte rețele private au propria lor organizație NIC. A se vedea și **InterNIC**.

Ning („pace” în limba chineză). Platformă cu ajutorul căreia utilizatorii își pot crea propriile site-uri și rețele sociale. *Ning*, fondată de Marc Andreessen (coautor al primului browser grafic, *Mosaic*, și cofondator al Netscape Communications Corporation) și Gina Bianchi, a fost lansată în octombrie 2005, în Palo Alto, California.

De la început, creatorii *Ning* au sperat ca această platformă să intre în competiție cu binecunoscutele servicii *Facebook* și *MySpace*, adresându-se utilizatorilor care doreau să creeze rețele în jurul unor interese specifice sau celor care dispuneau de abilități tehnice reduse (conform *Wikipedia*). Cea mai importantă caracteristică a *Ning* este aceea că oricine își poate crea și personaliza propria rețea socială pentru un anumit subiect sau o nevoie particulară, adresându-se unei audiențe specifice.

În septembrie 2006, *Ning* a oferit trei șabloane: pentru site-uri de grup, pentru site-uri de fotografii și pentru site-uri de clipuri video, utilizatorii având posibilitatea de a include conținut pe care, ulterior, îl puteau folosi în orice scop. Mai târziu, aceste trei șabloane au fost incluse într-unul singur, cu scopul de a le oferi non-dezvoltatorilor posibilitatea de a-și personaliza mult mai rapid și mai facil propria copie a unui site social. *Ning* oferă suport pentru API-ul *OpenSocial*, lansat de Google în 2007. Astfel, dezvoltatorii au posibilitatea de a include *widget*-urile *OpenSocial* în cadrul propriilor rețele.

Utilizatorii își pot crea pe platforma *Ning* propria rețea socială (numită și *spațiu*), beneficiind de găzduire gratuită. În acest caz, *Ning* adaugă și conținut propriu în site-ul creat (*i.e.*, reclame). Opțiunea „*Ning for Business*” le oferă utilizatorilor posibilitatea – în schimbul unei taxe lunare – de a-și crea rețele sociale în care vor adăuga numai conținut propriu. De asemenea, aceștia pot beneficia de spațiu de stocare și de lățime de bandă suplimentare, precum și de URL-uri non-*Ning*.

Mulți educatori utilizează *Ning* pentru dezvoltarea resurselor educaționale. Participanții (membrii) au posibilitatea să încarce imagini și video-uri create de studenți în cadrul unor proiecte, precum și să includă legături către bloguri și către *Flickr*. Rețeaua *Ning in Education*, creată de Steve Hargadon (<http://education.ning.com/>), le oferă un real sprijin celor care doresc să exploateze potențialul educațional al rețelelor sociale create folosind platforma *Ning*. O listă deschisă care include un număr important de rețele sociale utilizate în domeniul educațional poate fi consultată pe *wiki-ul Social Networks in Education*. A se vedea și **rețea socială online**.

nod (*node*). În rețelele locale, dispozitiv conectat în rețea, capabil să comunice cu alte dispozitive incluse în aceeași rețea. A se vedea și **LAN**.

NodeBox. Aplicație *open source* pentru crearea de grafică și animații 2D, scris în Python (<http://www.nodebox.net/>).

Este bazat pe *DrawBot*, un alt proiect *open source*. A se vedea și **DrawBot** și **Python**.

(a) **normaliza** (*normalize*). Referitor la o bază de date relațională, înseamnă aplicarea unui set de proceduri în scopul eliminării anomaliilor logice și funcționale. Astfel, prin normalizare vor fi simplificate operațiile de actualizare și interogare. În schimb, după normalizare se va mări numărul de tabele componente ale bazei de date. A se vedea și **normalizare**.

normalizare (*normalization*). Ansamblu de transformări succesive ale unei baze de date având drept scop obținerea unei organizări optimizate a acesteia. Odată cu fiecare transformare, baza de date este adusă într-o formă standard, cunoscută sub denumirea de *formă normală* (FN). Deși există șase forme normale, este acceptat că o bază de date este suficient de bine optimizată dacă a fost adusă în cea de-a treia formă normală.

Prima formă normală (FN1, E.F. Codd, 1970) elimină câmpurile compuse și pe cele repetitive. A doua formă normală (FN2, E.F. Codd, 1971) elimină dependențele funcționale parțiale (trebuie ca toate câmpurile unui tabel să depindă numai de cheia sa primară, nu de părți ale acesteia). A treia formă normală (FN3, E.F. Codd, 1971) elimină dependențele funcționale tranzitive (dependențele de acest tip iau naștere când un câmp C1 depinde de un alt câmp C2, acesta depinzând de un alt câmp C3, astfel încât C1 depinde de

C3, situație care poate determina o serie de anomalii nedorite). Alături de cele șase forme normale, au fost definite încă două: BCNF (*Boyce-Codd normal form*; R.F. Boyce și E.F. Codd, 1974) și DKNF (*Domain/key normal form*; Ronald Fagin, 1981).

În procesul de normalizare trebuie respectate următoarele reguli: se utilizează numai valori atomice ale coloanelor (nici o coloană nu trebuie să conțină valori compuse; de exemplu, este contraindicat să se utilizeze o singură coloană care să conțină numele și prenumele, în locul ei trebuind utilizate două coloane distincte – una care conține numele și alta care conține prenumele); cheia primară este utilizată în totalitate (sunt interzise tabelele în care o coloană depinde numai de o porțiune a cheii primare); se utilizează numai cheia primară (adică un tabel cu informații despre mai multe categorii de obiecte trebuie divizat în tabele separate, unite printr-o cheie externă). Etapele de proiectare și de normalizare a unei baze de date se pot suprapune, deoarece aducerea la o nouă formă normală presupune construirea unor noi tabele prin gruparea unor coloane existente, dar și a altora noi, precum și crearea unor noi chei primare. A se vedea și **formă normală**.

notație BNF (*BNF notation*). În informatică, notația Backus-Naur Form (BNF) este o metasintaxă utilizată pentru descrierea gramaticii unui limbaj, elaborată de John Backus (1959) și Peter Naur (1963). Este

practic identică cu așa-numitele mecanisme de generare a șirurilor de limbaj elaborate de Chomsky între anii 1956 și 1959. BNF folosește abstracțiile pentru a descrie structurile sintactice. O descriere BNF sau o gramatică (independentă de context) este o colecție de reguli. A se vedea și **sintaxă**.

notație postfix (*postfix notation*). Cunoscută și ca *notația poloneză inversă* (RPN, *Reverse Polish notation*), este o notație utilizată pentru expresiile matematice în care fiecare operator urmează tuturor operanzilor săi. Prin folosirea notației RPN, este eliminată necesitatea utilizării parantezelor.

Notația postfix a fost propusă în 1954 de Burks, Warren și Wright, fiind reinventată la începutul anilor 1960 de F.L. Bauer și E.W. Dijkstra. În informatică, notația este utilizată în limbajele de programare bazate pe stivă (*e.g.*, Forth, PostScript). A se vedea și **Forth, notație prefix și PostScript**.

notație prefix (*prefix notation*). Cunoscută și ca *notația poloneză* (PN, *Polish Notation*), este o notație pentru expresiile matematice în care operatorii sunt plasați în stânga operanzilor lor. Ca rezultat, este eliminată necesitatea utilizării parantezelor. Notația a fost introdusă în anii 1920 de matematicianul polonez Jan Lucasiewicz și este utilizată în unele limbaje de programare (*e.g.*, Ambi, LISP). A se vedea și **notație postfix**.

notație științifică (*scientific notation*). Modalitate de reprezentare a numerelor în virgulă mobilă, utilizată mai ales pentru numerele foarte mici sau foarte mari. În această notație, o valoare este reprezentată ca un produs între un număr (a) și o putere întregă a lui 10 (b): $a \times 10^b$. Numărul a se numește *mantisă*, iar numărul b se numește *exponent*.

Notepad++. Editor de cod-sursă *open source* pentru un număr mare de limbaje; de exemplu C++, Java, XML, HTML, PHP, CSS, Perl, Python, Ruby, JavaScript, ASP, SQL (<http://notepadplusplus.sourceforge.net/uk/site.htm>). Suportă evidențierea sintaxei, oferind și alte facilități, dintre care pot fi amintite: stiluri personalizate, editarea simultană a mai multor fișiere, autocompletare, căutarea și înlocuirea șirurilor folosind expresii regulate, *drag&drop*. A se vedea și **editor de cod-sursă**.

Noyce, Robert Norton (1927–1990). Fizician american, poreclit „primarul din Silicon Valley”, cofondator al Fairchild Semiconductor (1957) și al Intel Corporation (1968). Este creditat ca fiind coinventator al circuitului integrat (microcip), alături de Jack Kilby. A se vedea și **circuit integrat**, **Intel Corporation** și **Kilby, Jack**.

NSFNET. Rețea creată de NSF (*National Science Foundation*) pentru a conecta centrele cu supercalculatoare printr-o magistrală cu viteză mare (56 kbps). Ca organizație necomercială,

NSFNET permitea conectarea la Internet fără restricțiile impuse de ARPANET. În 1990 ARPANET este desființată, după ce toate organizațiile care erau conectate au trecut la NSFNET. La rândul său, NSFNET își încetează activitatea în anul 1995, când accesul la Internet ajunge să fie asigurat de firme comerciale pentru întreaga lume. A se vedea și **ARPANET**.

număr binar (*binary number*). Număr exprimat în sistemul de numerație binar. În acest scop se folosesc puterile lui 2 ($1=2^0$, $2=2^1$, $4=2^2$, $8=2^3$, $16=2^4$, $32=2^5$, $64=2^6$, $128=2^7$, $256=2^8$, $512=2^9$, $1024=2^{10}$ etc.). Pentru a realiza conversia unui număr din sistemul zecimal în cel binar, se procedează ca în exemplul următor (în care se convertește numărul 89): $89 = 1*64 + 0*32 + 1*16 + 1*8 + 0*4 + 0*2 + 1*1 = 1*2^6 + 0*2^5 + 1*2^4 + 1*2^3 + 0*2^2 + 0*2^1 + 1*2^0$. Forma binară a numărului se obține reținând factorii care înmulțesc puterile lui 2, în ordinea descrescătoare a puterilor. În exemplul anterior, rezultatul va fi 1011001.

număr în virgulă fixă (*fixed-point number*). Format intern utilizat pentru reprezentarea numerelor întregi și subunitare. Este dedus în mod direct din forma zecimală ordinară a numerelor, în conformitate cu care numerele prezintă o virgulă zecimală, în stânga acesteia aflându-se partea întregă a numărului, iar în dreapta partea zecimală.

Pentru reprezentarea numerelor binare cu semn, din cei n biți disponibili, unul

este utilizat pentru semn, acesta fiind prin convenție situat în partea stângă (bitul cel mai semnificativ) și având valoarea „0” pentru semnul „+” și „1” pentru semnul „-”. Biții pot fi numerotați de la dreapta la stânga (cum se întâmplă la SPARC), reprezentare care se numește *Little Endian*, sau de la stânga la dreapta (cum se întâmplă la familia Intel), reprezentare care se numește *Big Endian*. În practică se utilizează trei reprezentări ale numerelor binare cu virgulă fixă: semn-mărime, în complement față de 1 și în complement față de 2. A se vedea și **număr în virgulă mobilă**.

număr în virgulă mobilă (*floating-point number*). Format intern utilizat pentru reprezentarea numerelor reale. În acest scop, sunt folosite o mantisă și un exponent pentru o anumită bază (de obicei 2, în cazul calculatoarelor). Mantisă este un număr pozitiv în virgulă fixă, reprezentat în cod complementar. Exponentul este un număr întreg cu semn. Termenul de virgulă mobilă (sau flotantă) se referă la faptul că virgula care separă partea întreagă de cea fracționară se poate deplasa, adică poate fi plasată oriunde relativ la cifrele semnificative ale numărului. Poziția virgulei este indicată separat în cadrul reprezentării interne. Valoarea unui număr reprezentat în virgulă mobilă se determină ridicând baza la o putere egală cu exponentul, rezultatul înmulțindu-se cu mantisă. A se vedea și **număr în virgulă fixă**.

nume de cale (*pathname*). Denumită și *cale de directoare* (*directory path*),

reprezintă, într-un sistem de fișiere organizat ierarhic, lista directoarelor care trebuie parcurse pentru a se ajunge din directorul curent la un anumit fișier. A se vedea și **cale** și **nume de cale complet**.

nume de cale complet (*full pathname*). Într-un sistem de fișiere organizat ierarhic, reprezintă lista directoarelor care trebuie parcurse pentru a ajunge la un anumit fișier pornind de la directorul rădăcină. A se vedea și **cale** și **nume de cale**.

nume de domeniu (*domain name*). Nume convențional (simbolic) utilizat în Internet pentru identificarea univocă a unei rețele sau a unui calculator. Este constituit dintr-o succesiune de două sau mai multe grupuri de caractere separate între ele prin puncte, ultimul fiind denumit nume de domeniu de nivel superior (TLD, *Top Level Domain*), cunoscut și ca domeniu primar sau principal.

Fiecare domeniu este împărțit, la rândul lui, în mai multe subdomenii, asociate unor companii, instituții sau persoane (cu lungimea maximă de 63 de caractere). La rândul lui, un subdomeniu poate fi compus din sub-subdomenii sau din nume simbolice de calculatoare (gazde) conectate la Internet. Un calculator poate avea asociate mai multe nume simbolice. Se obține astfel o structură de tip arbore, în care frunzele reprezintă (sub)domenii care nu mai au, la rândul lor, subdomenii. A se vedea și **domeniu**.

nume de fișier (*filename*). Șir de caractere utilizat pentru a identifica un fișier aflat pe un dispozitiv de stocare. Lungimea maximă, caracterele permise, precum și alte restricții impuse numelui de fișier depind de sistemul de operare. A se vedea și **fișier**.

nume de gazdă (*hostname*). Nume unic sub care un dispozitiv conectat într-o rețea este cunoscut în rețeaua respectivă. Acesta este utilizat, de exemplu, pentru identificarea unei gazde care furnizează diverse tipuri de servicii (*e.g.*, WWW, *e-mail*). În Internet, un nume de gazdă este în mod obișnuit o combinație între numele local al acesteia și numele domeniului-părinte. A se vedea și **gazdă** și **Internet**.

nume de utilizator (*username*). Șir de caractere format din litere, cifre și alte caractere, care identifică un cont de utilizator. A se vedea și **cont de utilizator** și **utilizator**.

Nvidia Corporation. Este o corporație multinațională cu sediul în SUA, specializată în producerea de proesoare grafice și *chipset*-uri pentru calculatoare, stații de lucru și dispozitive mobile (<http://www.nvidia.com/>). A fost înființată în 1993 de Jen-Hsun Huang, Curtis Priem și Chris Malachowski.

Nygaard, Kristen (1926–2002). Programator norvegian, unul dintre părinții programării orientate pe obiect (POO). Este coautor, împreună cu Ole-Johan Dahl, al limbajelor de programare Simula, care au folosit pentru prima oară noțiunile de clase, subclase, moștenire și creare dinamică de obiecte. Pentru această realizare, ambii programatori au primit în 2001 premiul Turing. A se vedea și **programare orientată pe obiect**, **Simula** și **Dahl, Ole-Johan**.

O

obiect (*object*). În programarea orientată pe obiect (POO), reprezintă un exemplar (sau instanță) al (a) unei clase. Setul de valori ale proprietăților unui obiect particular este denumit *starea* acestuia. Obiectul constă din starea și din comportamentul său, cel din urmă fiind definit de metode. Dintr-o clasă dată, prin instanțiere, pot fi obținute mai multe obiecte. A se vedea și **clasă**, **instanță** și **programare orientată pe obiect**.

obiect de învățare (LO, *Learning Object*). Entitate digitală sau non-digitală care poate fi utilizată, reutilizată sau referențiată în procesele de învățare, de educație sau de pregătire susținute de tehnologie. Definiția anterioară aparține *Comitetului pentru Standardizarea Tehnologiilor de Învățare* (LTSC) al IEEE, care oferă ca exemple de LO: conținut multimedia, materiale tipărite, lucrări de studiu, exerciții, cursuri, programe de studiu, obiectivele învățării, software educațional, persoane, organizații sau evenimente la care se face referire în cursul procesului de învățare. Din definiția amintită poate fi reținută ideea că LO pot fi reprezentate și de

componente software reutilizabile în construirea arhitecturii sistemelor-suport pentru instruire.

Obiectele de învățare (denumite și obiecte educaționale) constituie un concept relativ nou referitor la organizarea și managementul conținutului educațional, oferind o nouă abordare a procesului de învățare, diferită de cea tradițională a împărțirii în module (lecții) predate/învățate (fiecare) în cadrul unei ore de curs. Spre deosebire de aceasta, obiectele de învățare reprezintă componente (unități de învățare) relativ mici, create/întreținute independent și reutilizabile, conceptul fiind adesea comparat cu jocul de LEGO.

Într-o abordare diferită de cea a LTSC, se face o paralelă între arhitectura pe componente a software-ului și structura unui curriculum. Astfel, Robby Robson consideră că „resursele de instruire” sunt similare cu obiectele utilizate în programarea orientată pe obiect (POO) și – în consecință – acestea „au metode și proprietăți”. Conform lui Robson, metodele tipice sunt cele de interpretare și evaluare, iar proprietățile sunt conținutul și relațiile cu alte resurse. Modelul utilizat este în conformitate cu cel folosit pentru obiecte în POO, ceea ce implică prezența principiilor de încapsulare, abstractizare și moștenire, caracteristice acestora.

Andrés Chiappe definește LO ca „o entitate digitală independentă, reutilizabilă, având un scop educațional clar și incluzând cel puțin trei componente

editabile: conținut, activități de învățare și elemente care o încadrează în context. De asemenea, LO trebuie să aibă o structură externă de informații care să faciliteze identificarea, stocarea și regăsirea metadatelor.”

Adaptând o definiție provenită de la *Wisconsin Online Resource Center*, Robert J. Beck sugerează că OI trebuie să aibă următoarele caracteristici: să ofere o nouă modalitate de a gândi conținutul învățării (în mod tradițional, conținutul învățării/unitatea de învățare este fragmentat în câteva părți care sunt predate/învățate într-un număr corespunzător de ore de curs; LO sunt componente ale învățării mult mai mici, având ca durată obișnuită 2-15 minute); să poată fi utilizate independent; să fie reutilizabile (un LO trebuie să fie utilizat în multiple contexte, pentru multiple scopuri); să poată fi agregate (LO trebuie să poată fi grupate în colecții mari de conținut educațional, incluzând aici și structurile tradiționale de curs); să fie etichetate folosind metadata (fiecare LO trebuie să aibă asociate informații descriptive care oferă posibilitatea de a fi ușor de găsit prin căutare).

Analizând definițiile anterioare, se poate spune că obiectele de învățare sunt: de sine stătătoare (pot fi folosite independent); interactive (studentul interacționează cu conținutul); reutilizabile (pot fi utilizate în contexte diferite, pentru scopuri diferite); interoperabile (pot fi folosite în sisteme diferite); agregabile (pot fi incluse în colecții

largi de obiecte, inclusiv în cursuri tradiționale); adaptabile diferitelor stiluri de învățare; marcate (au asociate informații – numite metadata – care facilitează operațiile de căutare și regăsire).

Obiectele de învățare sunt cunoscute și sub numele de RLO (*Reusable Learning Objects*), definit ca cea mai mică unitate instrucțională independentă care conține un obiectiv, o unitate de învățare și un element pentru evaluare. Compania CISCO utilizează termenii RLO și RIO (*Reusable Information Object*), primul echivalând cu o lecție, iar al doilea cu o secțiune din lecție. A se vedea **Comitetul pentru Standardizarea Tehnologiilor de Învățare**.

obiect de învățare deschis (*OpenLO*, *Open Learning Object*). Resursă digitală care poate fi reutilizată în sprijinul învățării. Cuvântul *Open* precizează caracterul deschis al conținutului obiectului, ceea ce înseamnă că acesta este dezvoltat într-un format deschis (*e.g.*, OpenDocument Format) sau că sunt disponibile fișierele sursă, chiar dacă formatul este închis (*e.g.*, Adobe *Flash*).

Modelul *OpenLO* le permite utilizatorilor să editeze obiectele de învățare create de diverși autori și să le modifice în conformitate cu propriile nevoi pedagogice. În plus, comunitățile de profesioniști din domeniul educației pot lucra la același obiect de învățare, contribuind la evoluția colaborativă (la nivelul conținutului) a acestuia.

Implementarea modelului *OpenLO* și definirea metodologiilor educaționale bazate pe acest model presupune următoarele aspecte: schimbarea ciclului de viață al obiectelor de învățare și, în consecință, a metodologiilor pentru producerea acestor resurse; atribuirea unui rol dinamic metadatelor, care trebuie să evolueze în paralel cu obiectul de învățare; trecerea de la depozitele de obiecte de învățare actuale (LOR, *Learning Object Repositories*) la noile sisteme de management al obiectelor de învățare (LOMS, *Learning Object Management Systems*). A se vedea și **depozit de obiecte de învățare** și **obiect de învățare**.

object-oriented programming (POO, *programare orientată pe obiect*). A se vedea **programare orientată pe obiect**.

octet (*byte*). A se vedea **byte**.

octogonul lui Kahn (*Khan's octogonal framework*). Ghid pentru planificarea, dezvoltarea, managementul, evaluarea și livrarea unui program de instruire pentru învățarea combinată, cu următoarele dimensiuni: instituțională (include probleme referitoare la serviciile organizatorice și administrative); pedagogică (se referă la adaptarea conținutului oferit celor instruiți la nevoile acestora și la obiectivele instruirii); tehnologică (include aspecte referitoare la crearea unui mediu de instruire și a unui set de instrumente care vor fi utilizate pentru livrarea programului de instruire); proiectarea interfeței (se referă la factorii aflați în

legătură cu interfața cu utilizatorul, asigurând că aceasta este suficient de complexă pentru a suporta toate elementele sistemului/ programului de instruire); de evaluare (se referă la capacitatea sistemului de instruire de a evalua performanța fiecărei persoane instruite); de management (se referă la aspectele legate de managementul unui sistem de învățare combinată, așa cum sunt infrastructura și logistica, pentru a gestiona livrarea unor tipuri multiple de conținut); resurse-suport (se referă la crearea unor tipuri variate de resurse – *offline* și *online* – care le vor fi oferite celor instruiți); etică (identifică aspectele etice care trebuie avute în vedere atunci când este dezvoltat un sistem de învățare combinată).

În ceea ce privește domeniul educațional, cursurile bazate pe utilizarea sistemelor de învățare combinată se desfășoară în special (dar nu exclusiv) în mediul universitar. Componentele *online* ale acestora integrează utilizarea unor servicii Web 2.0: *feed-uri* RSS/Atom, bloguri, site-uri wiki și sisteme de *bookmarking* colaborativ. A se vedea și **e-Learning** și **învățarea combinată**.

ODBC (*Open DataBase Connectivity*). API creat de Microsoft în 1992 pentru accesarea surselor de date de către aplicații. Sursa de date este un ansamblu de informații care include: o bază de date de un anumit tip, în care sunt stocate datele utilizate de aplicație; un *driver ODBC*, care furnizează un set de comenzi specific sistemului de baze de

date utilizat; parametrii conexiunii care va fi stabilită de driver cu serverul de baze de date.

ODBC este o interfață de acces universal la date (*Universal Data Access*) având drept caracteristică interoperabilitatea, adică posibilitatea de a accesa diverse tipuri de baze de date utilizând același cod-sursă. Din acest motiv, ODBC este una dintre interfețele standard utilizate pentru dezvoltarea aplicațiilor independente de motorul de baze de date. Au fost dezvoltate drivere ODBC pentru un număr mare de tipuri de baze de date (e.g., *Microsoft SQL Server, Oracle, MySQL, PostgreSQL*).

ODF (*OpenDocument Format*). A se vedea **OpenDocument Format**.

OER Commons. Rețea pentru educația deschisă, care folosește tehnici specifice Web 2.0 (*tag-uri, comentarii și lucrul colaborativ*) pentru a crea o experiență de lucru *online* care să le ofere profesorilor prilejul de a contribui cu cele mai bune practici educaționale (<http://www.oercommons.org/>).

Rețeaua a fost creată ISKME (*Institute for the Study of Knowledge in Education*) și este sprijinit de Fundația Flora și de William Hewlett (<http://www.hewlett.org/>).

Misiunea *OER Commons* este de a crește oportunitățile educaționale, prin oferirea de acces la resurse educaționale de calitate și prin facilitarea creării, utilizării și reutilizării unor

astfel de resurse pentru educatori, studenți și autodidacți.

Obiectivele *OER Commons* sunt: utilizarea resurselor educaționale (furnizarea unui singur punct de acces prin care educatorii, studenții și toți cei interesați să poată căuta, naviga, evalua, descărca și discuta asupra resurselor educaționale deschise care sunt disponibile liber *online*); reutilizarea resurselor educaționale (creșterea oportunităților pentru cei care utilizează resurse educaționale deschise pentru a dezvolta și propune conținut de înaltă calitate pentru uzul celorlalți); dezvoltarea comunității (mărirea oportunităților pentru educatori, studenți și autodidacți de a schimba informații și de a crea standarde, cu alte cuvinte de a îmbunătăți și evalua calitatea resurselor educaționale deschise disponibile în Internet). A se vedea și **resurse educaționale deschise și Web 2.0**.

Office 2.0. Categorie de aplicații Web 2.0 care permit crearea *online* a documentelor text (*Word Processor*), a foilor de calcul (*Spredsheet*) și a prezentărilor (*Presentations*), oferindu-le utilizatorilor facilități similare cu cele ale pachetelor *Microsoft Office* sau *Open Office*. Liderii acestei categorii de aplicații sunt *Google Docs* și *Zoho*. Pentru a produce ușor și rapid documente fără a învăța modul de utilizare a celor două pachete (relativ complexe), pot fi folosite – de exemplu – aplicațiile *online Thumbstacks* și *ajaxWrite*. De asemenea, în categoria aplicațiilor *online* de birou mai pot fi incluse și cele

de tip calendar, una dintre acestea fiind *Google Calendar*.

Aplicațiile *Office 2.0* le oferă utilizatorilor o serie de avantaje, dintre care pot fi menționate: sunt 100% gratuite; pot fi utilizate folosind orice calculator, iar documentele create pot fi accesate de oriunde (de acasă, de la birou, în deplasare); pot fi accesate de pe orice platformă (Windows, Linux, Mac), lucrând la fel pe oricare dintre acestea; documentele realizate folosind aceste aplicații pot fi împărtășite altor utilizatori, ceea ce presupune și posibilitatea editării lor colaborative; nu este necesar ca utilizatorii să instaleze și să actualizeze software-ul, orice caracteristică nou creată fiind adăugată automat aplicației; creatorii aplicațiilor își oferă gratuit propria expertiză, ajutându-i pe utilizatori, care – din acest motiv – nu mai consumă timp pentru a găsi soluții la problemele întâmpinate în folosirea lor. A se vedea și **Google Docs** și **Zoho**.

Office Open XML. Cunoscut și ca *OOXML*, *OpenXML* sau *Open XML*, este un format gratuit deschis pentru reprezentarea documentelor *Office*, dezvoltat de Microsoft și aprobat ca standard internațional de ISO/IEC în anul 2008.

Microsoft a dezvoltat formatul *Office Open XML* ca succesori al formatelor sale binare și *Office XML 2003*. Ulterior, specificațiile dezvoltate de Microsoft au devenit standard ECMA

(decembrie, 2006), apoi ISO/IEC (noiembrie, 2008). Începând cu *Microsoft Office 2007*, *Office Open XML* a devenit format de fișier implicit pentru suita *Microsoft Office*. *Service Pack 2* pentru *Microsoft Office 7* (mai, 2008) adaugă suport nativ pentru ODF, oferind posibilitatea de a alege ODF 1.1 sau PDF ca format implicit. A se vedea și **format deschis**.

oglină de date (*data mirror*). Sistem în care datele sau informațiile despre diverse activități sunt scrise și stocate simultan pe două sau mai multe discuri, astfel încât dacă unul dintre acestea nu mai poate fi utilizat, datele sunt încă disponibile.

OHA (*Open Handset Alliance*). Alianță formată în 2007 din peste 50 de companii, creată și condusă de Google, Inc. cu scopul dezvoltării de standarde deschise pentru dispozitivele mobile (<http://www.openhandsetalliance.com/>)

OHA a lansat în anul 2008 sistemul de operare *open source* Android, ca un competitor al altor platforme mobile, furnizate de companiile Microsoft, Nokia, Apple, Palm și Symbian.

OLE (*Object Linking and Embedding, legarea și înglobarea obiectelor*). Tehnologie dezvoltată de Microsoft începând cu anul 1990, care permite legarea sau înglobarea în documente (e.g., create cu procesorul de text *Word*) a obiectelor (e.g., fișiere care conțin imagini, diagrame). Când obiectul este

legat, documentul conține numai o referință la el, orice modificare a obiectului fiind reflectată în document. În schimb, dacă obiectul este legat, documentul conține numai o copie a sa, iar modificările efectuate asupra obiectului original nu se vor reflecta în document.

Olimpiade.ro. Site care include o serie de materiale educaționale la nivel olimpic, din toate materiile predate în sistemul de învățământ românesc la care se organizează olimpiade (<http://www.olimpiade.ro/>). Termenii legali în care sunt oferite aceste materiale pot fi consultați în secțiunea *Termeni legali*.

OMA (Open Mobile Alliance). Organizație de standardizare creată în anul 2002 de cei mai importanți producători de echipamente și sisteme mobile de telecomunicații (<http://www.openmobilealliance.org/>).

online banking (tranzacții bancare online). Denumită și *Internet banking*, este o modalitate de realizare a tranzacțiilor financiare și de administrare a conturilor bancare prin intermediul unui site Web operat de o instituție financiară.

oNLine System. Sistem al cărui prototip a fost propus de Douglas Engelbart în anul 1960. Într-un astfel de sistem, pentru parcurgerea documentelor sunt utilizate legături. A se vedea și **legătură**.

Open Access (acces deschis). A se vedea **acces deschis**.

OpenCourseWare. Publicație bazată pe Web care include conținutul educațional deschis creat de MIT (*Massachusetts Institute of Technology*).

Dezvoltarea publicației *OpenCourseWare* a început în 2002 cu postarea a 50 de cursuri (având atașate traduceri în spaniolă și portugheză). În 2007, numărul acestora a ajuns la 1800, site-ul înregistrând lunar peste 2 milioane de vizite. În 2008 au fost postate sau actualizate peste 200 de cursuri.

OpenCourseWare Consortium (Consortiul OpenCourseWare). Organizație reprezentând colaborarea a peste 100 de instituții de învățământ superior și organizații asociate din întreaga lume care creează împreună după un model comun un corp larg de conținut educațional (<http://www.ocwconsortium.org/>).

Scopurile Consortiului *OpenCourseWare* sunt următoarele: extinderea bogăției și impactului *opencourseware* prin încurajarea adoptării și adaptării de material educațional deschis din întreaga lume; sprijinirea dezvoltării de proiecte *opencourseware* adiționale; asigurarea durabilității proiectelor *opencourseware* prin identificarea de noi modalități de îmbunătățire a eficacității și de reducere a costurilor.

Open Directory Project (ODP). Director Web. A se vedea și **Dmoz**.

OpenDocument Format (ODF, *Open Document Format for Office Applications*). Format deschis pentru documentele *Office*, derivat din XML, creat inițial de Sun Microsystems.

Dezvoltat de Comitetul Tehnic (TC, *Technical Committee*) al OASIS (*Organization for the Advancement of Structured Information Standards*) și propus ca Standard Internațional de această organizație, a fost adoptat de ISO/IEC în 30 noiembrie 2006, primind indicativul ISO 26300:2006. Specificațiile *OpenDocument 1.0*, publicate de TC-OASIS, corespund standardului amintit. Versiunea *OpenDocument 1.1* include caracteristici suplimentare, fiind adoptată ca standard OASIS în februarie 2007.

Fiind derivat din XML, ODF are și avantajele acestui metalimbaj de marcare: este extensibil, specificațiile ODF indicând explicit acest lucru. Astfel, utilizatorul poate adăuga extensii formatului, în funcție de nevoile sale; documentele ODF pot fi manipulate cu procesoarele XML existente (numărul acestora este destul de mare, XML fiind o tehnologie matură – „vârsta” sa este mai mare de 20 de ani); favorizează conversia la alte formate XML, această operațiune fiind mult mai ușor de realizat față de conversia între formatele binare.

ODF este proiectat pentru a fi folosit ca format de fișier pentru aplicațiile *Office*, având următoarele caracteristici principale: reduce dimensiunea fișierelor; păstrează integritatea datelor;

permite utilizatorilor să salveze, să editeze și să partajeze documente *Office* (documente text, foi de calcul, prezentări, baze de date), indiferent de aplicația și platforma pe care au fost create.

Cele mai obișnuite extensii de fișier pentru documentele *OpenDocument* sunt: *.odt* (pentru documentele text), *.ods* (pentru foile de calcul), *.odp* (pentru prezentări), *.odg* (pentru desene și grafice) și *.odf* (pentru formule matematice).

ODF este formatul nativ pentru câteva pachete *Office*, dintre care pot fi menționate: *OpenOffice.org* (începând cu versiunea 2.0); *KOffice* (include aplicații pentru procesarea textului, crearea foilor de calcul și prezentărilor, manipularea imaginilor etc.; suita poate fi utilizată pe platformele Unix, Linux); *StarOffice* (suită *Office* proprietară deținută de Sun Microsystems, Inc., care poate fi utilizată pe platformele Windows și Linux); *NeoOffice* (suită *Office* pentru Mac OS X, bazată pe *OpenOffice.org 2.1*); *EuroOffice* (suită *Office open source* derivată din *OpenOffice.org*; poate fi utilizată pe platformele Windows, Linux și Solaris); *IBM Workplace* (suită furnizată de IBM care poate fi utilizată pe platformele Windows și Linux).

De asemenea, există aplicații care suportă formatul ODF, chiar dacă acesta nu este nativ: *Oxygen XML Editor*, *Lotus Notes 8*, *Corel WordPerfect Office*, *Microsoft Office* etc. Dintre suitele *Office* bazate pe Web,

Peepel are ODF ca format nativ. Serviciile *Google Docs* și *Zoho* suportă formatul ODF.

Au fost create numeroase programe pentru citirea/vizualizarea documentelor ODF. Dintre acestea, pot fi menționate: *Slideshare* (aplicație bazată pe Web pentru încărcarea și partajarea prezentărilor în formatele PDF, PowerPoint, ODF etc.); *Firefox ODF Reader* (extensie pentru browserul *Mozilla Firefox* care permite vizualizarea documentelor ODF); *jodviewer* (vizualizator pentru prezentările în format ODF).

ODF Validator este un serviciu *online* oferit gratuit de *OpenDocument Fellowship*, utilizat pentru a verifica dacă documentele OpenDocument sunt conforme cu specificațiile ODF. A se vedea și **format deschis și ODF**.

open format (*format deschis*). A se vedea **format deschis**.

OpenLearn. Site care oferă acces gratuit oricărei persoane, indiferent de locul în care se află, la materialele de curs ale *OpenUniversity* (<http://openlearn.open.ac.uk/>). Orice utilizator înregistrat va avea pe site un spațiu propriu (*myLearningSpace*), pe care îl va putea gestiona conform necesităților sale. Conținutul site-ului este oferit sub licență *Creative Commons Attribution-NonCommercial-ShareAlike 2.0*. A se vedea și **licențe Creative Commons**.

OpenOffice.org. Cunoscut și ca *OpenOffice*, *Ooo* sau *OO.o*, este un

pachet *Office* dezvoltat de Sun Microsystems, Inc., ajuns în octombrie 2008 la versiunea 3. Implementează *OpenDocument Format* (ODF). Pachetul include aplicații (numite și module sau componente) pentru: procesarea textului (*Writer*); calcul tabelar (*Calc*); desen și grafică vectorială (*Draw*); crearea de prezentări (*Impress*); editare de formule (*Math*); crearea bazelor de date (*Base*).

Pachetul, care poate fi utilizat pe majoritatea platformelor (Windows, Linux, Solaris, BSD, OS/2, Mac OS X), fiind valabil într-un număr mare de limbi (peste 70 în noiembrie 2008), este disponibil pentru descărcare gratuită și utilizare fără restricții în orice domeniu (privat, educațional, comercial, administrație publică) și în orice scop. *OpenOffice.org* este compatibil cu *Microsoft Office*, având disponibilă și opțiunea pentru export în format PDF. Pachetul este ușor de instalat și de învățat, fiind potrivit atât pentru începători, cât și pentru cei care au experiență în utilizarea altor pachete similare.

OpenOffice.org folosește o licență dublă pentru codul-sursă (GNU GPL și SISSL – *Sun Industry Standards Source License*), precum și o licență separată pentru documentele publicate pe site și pentru cele care nu sunt incluse în produs (*Public Document License*).

Deși formatul nativ suportat de aplicațiile incluse în *OpenOffice.org* este ODF, acestea pot fi utilizate și pentru citirea (și în unele cazuri

editarea) documentelor create cu alte pachete software de tip *Office* (*WordPerfect*, *StarOffice*, *Lotus*, *MS Works* și *Rich Text Format*). Astfel, pot fi editate majoritatea documentelor create cu *MS Office* (*Word*, *Excel*, *PowerPoint*), care, ulterior, pot fi salvate în formatul original sau în cel nativ.

OpenOffice.org Basic este un limbaj de programare similar cu Microsoft VBA și bazat pe *StarOffice Basic*. Acest limbaj poate fi utilizat în aplicațiile *Writer* și *Calc*, pentru scrierea, prin sarcini specifice, a funcțiilor (denumite subrutine sau macroui).

Un document ODF creat în particular cu *OpenOffice.org* (dar nu obligatoriu), este, de fapt, o arhivă. Pentru ca aceasta să fie recunoscută de instrumentul software utilizat pentru dezarhivare, este necesar să se schimbe extensia fișierului în *.zip*.

Utilizat adesea în mediul educațional, pachetul *OpenOffice.org* vine în întâmpinarea nevoilor educatorilor și studenților. Fiind creat sub o licență *open source*, poate fi distribuit și utilizat liber. A se vedea și **OpenDocument Format**.

Open Source Software (*software cu sursă deschisă*). Termenul, propus de Bruce Perens în 1998, înseamnă nu numai acces la libera utilizare a codului-sursă, ci implică și alte prerogative: libertatea redistribuirii fără restricții suplimentare (incluzând licența de utilizare), obligativitatea furnizării codului-sursă, libertatea de creare a unor produse derivate,

inexistența discriminărilor față de persoane sau grupuri. Eric Raymond și Bruce Perens au fondat în același an OSI (*Open Source Initiative*), cu scopul de a promova termenul amintit ca parte a unui program de marketing. Definiția software-ului *Open Source* agreeată de OSI se referă la aceleași libertăți incluse în FSD (*Free Software Definition*). A se vedea și **Free Software** și **FSD**.

Open Standard (*standard deschis*). A se vedea **standard deschis**.

Open University of The United Kingdom (OU, *Universitatea Deschisă a Marii Britanii*). Este una dintre cele mai prestigioase instituții de învățământ la distanță din Europa și din lume (<http://www.open.ac.uk/>). OU este deschisă oamenilor, locurilor, metodelor și ideilor, promovând oportunitățile educaționale și justiția socială prin furnizarea unei educații universitare de calitate tuturor celor care doresc să-și îndeplinească propriile ambiții și să-și folosească întregul potențial. Universitatea oferă un învățământ asistat, fiecare student beneficiind de sprijinul unui tutore.

Anual, la OU sunt circa 150 000 de studenți înscriși la cursuri universitare și mai mult de 30 000 la cursuri postuniversitare. Dintre studenții înscriși, circa 10 000 prezintă dizabilități. De asemenea, mai mult de 25 000 de studenți locuiesc în afara Marii Britanii. O treime dintre studenții înscriși la cursuri universitare au o pregătire inițială mai scăzută decât cea obișnuită, solicitată de alte universități britanice. Singura

condiție impusă unui candidat la OU este aceea de a avea vârsta minimă de 18 ani, neexistând o limită superioară. Materialele de curs sunt oferite folosind diverse suporturi: cursuri tipărite, seturi de cărți, casete audio și video, programe TV, CD-uri, site-uri Web și kituri experimentale pentru acasă. Pentru studenții cu dizabilități sunt utilizate formate alternative. Cursurile pot include un număr de sarcini și o examinare. Fiecare sarcină acoperă materialele de curs, ajutându-l pe student să-și consolideze ceea ce a învățat. Unele cursuri includ și sarcini scrise (e.g., proiecte, dizertații).

Opera. Browser Web gratuit cu versiuni pentru numeroase sisteme de operare, dezvoltat de compania norvegiană Opera Software (<http://www.opera.com/>). Are integrați clienți IRC și BitTorrent. Suportă XSLT, XPath, WebForms, CSS și SVG. Este utilizat pe calculatoare personale, telefoane mobile și PDA-uri, folosind în implementările pentru dispozitivele mobile (*Opera Mini*) o tehnologie proprie (*Small Screen Rendering*).

Opera poate fi controlat prin voce (începând cu versiunea 8), această facilitate fiind rezultatul unui parteneriat cu IBM. Comenzile vocale pot fi modificate. De asemenea, *Opera* poate citi pagini sau textul selectat. Browserul oferă facilități avansate care pot fi folosite de dezvoltatorii web pentru crearea unor site-uri interactive,

multi-modale, care acceptă comenzi vocale și răspund prin sinteză vocală.

Opera a fost primul browser Web care a integrat *mouse gestures* ca o metodă de navigare pe Internet. De asemenea, se poate controla orice aspect al browserului utilizând numai tastatura. Comenzile din tastatură și *mouse gestures* pot fi modificate pentru a răspunde cerințelor diferiților utilizatori. A se vedea și **browser Web**.

operator (*operator*). În programare și în matematică, simbol care precizează o operație efectuată asupra unuia sau mai multor operanzi.

Operatorii se clasifică după: numărul operanzilor (unari, binari, ternari), după ordinea de succedare a operatorilor și operanzilor (prefixați, infixati, postfixați), după tipul operanzilor și al prelucrării (aritmetici, relaționali, logici, la nivel de bit). Operatorii se împart în clase de precedență, fiecare clasă având o regulă de asociativitate. Aceasta din urmă indică ordinea aplicării operatorilor consecutivi de aceeași precedență. A se vedea și **asociativitatea operatorilor** și **precedența operatorilor**.

operator aritmetic (*arithmetic operator*). Operator utilizat în programare și matematică pentru a realiza o operație aritmetică. Operatorii de acest tip sunt unari și binari, respectiv aditivi și multiplicativi. Operatorii aritmetici sunt următorii: adunare (+), scădere (-), înmulțire (*) și împărțire (/). A se vedea și **operator**.

operator de adresare (*address operator*). Operator utilizat pentru a obține adresa unei entități. Operatorii de adresare uzuali sunt: indexare ([]), selecție directă (.), selecție indirectă (->), determinare adresă (&) și adresare indirectă (*). A se vedea și **operator**.

operator de atribuire (*assignment operator*). Operatorul de atribuire este „=”. Alături de acesta, sunt utilizați operatori de atribuire compuși, care sunt prescurtări pentru o serie de expresii de atribuire: „+=”, „-=”, „*=”, „/=”, „%=”, „<<=”, „>>=”, „&=”, „^=” și „|=”. A se vedea și **operator**.

operator logic (*logic operator*). Numit și boolean, un astfel de operator este similar cu un operator relațional: compară două expresii și returnează o valoare logică (true sau false). Operatorii logici sunt: and (&&), or (||) și not (!). Primii doi sunt operatori binari, iar ultimul unar. A se vedea și **operator relațional**.

operator pe biți (*bitwise operator*). Operator la nivel de bit. Se aplică fiecărui bit din reprezentarea operandilor întregi, spre deosebire de ceilalți operatori, care se aplică valorilor operatorilor. Iată operatorii la nivel de bit, enumerați în ordinea descrescătoare a priorității: complementare (~), deplasare la dreapta (>>), deplasare la stânga (<<), și (&), sau exclusiv (^), sau (!). A se vedea și **operator și precedența operatorilor**.

operator relațional (*relational operator*). Numit și *operator de comparație*

(*comparison operator*), este utilizat în programare pentru compararea valorilor sau expresiilor. Cei mai utilizați operatori relaționali sunt: mai mare (>), egal cu (=), mai mic decât (<), diferit de (<>), mai mare sau egal (>=), mai mic sau egal (<=). A se vedea și **expresie relațională și operator**.

operatorul de conversie explicită (*cast*). Operator unar care se utilizează atunci când se dorește ca tipul unui operand (expresie) să fie altul decât cel implicit. Are sintaxa (tip) expresie, în care tip este tipul explicit, iar expresie este operandul (expresia) al cărui tip trebuie modificat. A se vedea și **operator**.

operatorul condițional (*conditional operator*). Operatorul atribuie o valoare pe baza unei condiții. Sintaxa de utilizare a acestui operator este:

```
variabila = cond ? val1 : val2
```

Modul de operare este următorul: se evaluează condiția cond; dacă este adevărată, atunci variabila ia valoarea val1, altfel ia valoarea val2. Este singurul operator ternar. Cei trei operandi sunt cond, val1 și val2. A se vedea și **operator**.

operatorul secvențial (*sequential operator*). Operatorul virgulă are cea mai scăzută prioritate. Se folosește atunci când sintaxa limbajului necesită prezența unei singure expresii, dar prelucrarea presupune evaluarea a cel puțin două expresii. Acestea se evaluează de la stânga la dreapta, iar rezultatul întregii expresii este cel al ultimei

expresii. Iată un exemplu de utilizare:

```
int x, y, aux, max;  
max=x>y?x:(aux=y, y=x, x=aux);
```

În exemplul prezentat, operatorul secvențial este utilizat pentru a avea o singură expresie după „:”. A se vedea și **operator**.

operație în virgulă mobilă (*floating-point operation*). Operație aritmetică efectuată asupra unor numere în virgulă mobilă. A se vedea și **FLOPS** și **număr în virgulă mobilă**.

OPML (*Outline Processor Markup Language*). Format XML creat de Dave Winer, permite specificarea unei colecții de elemente arbitrare (numită și *listă* sau *schită – outline*), organizată ierarhic. Prin extensie, un document OPML poartă numele de *schită*. Formatul OPML poate fi folosit pentru a realiza *play-list*-uri, specificații, prezentări, agende, liste de *bookmark*-uri etc. Totuși, cea mai comună utilizare a acestuia constă în crearea listelor de *feed*-uri RSS, eventual organizate pe categorii, care pot fi schimbate (importate/exportate) între agregatoarele de *feed*-uri. A se vedea și **agregator** și **feed**.

Oracle Corporation. Companie americană de software înființată în 1977 de Larry Ellison și Ed Oates (<http://www.oracle.com/>). Oracle este specializată în dezvoltarea și comercializarea de produse pentru întreprinderi, fiind cunoscută mai ales pentru SGBDR *Oracle Database*. În 2009, Oracle a achiziționat Sun

Microsystems. A se vedea și **Ellison**, **Lawrence Joseph** și **SGBDR**.

Orkut. Rețea socială *online* și site de discuții operat de Google (<http://www.orkut.com>). A se vedea și **rețea socială online**.

osCommerce. Cea mai des utilizată soluție *open source* pentru dezvoltarea site-urilor de comerț electronic, bazată pe PHP și MySQL, cu versiuni pentru un număr mare de platforme: Windows, Linux, Solaris, BSD, Mac OS (<http://www.oscommerce.com/>). A se vedea și **comerț electronic**.

output buffering (*output buffering*). Mecanism care oferă posibilitatea trimiterii anteturilor HTTP și a variabilelor *cookie* chiar și după ce scriptul a început să genereze date de ieșire (e.g., prin intermediul instrucțiunii PHP `echo`). Acest lucru este posibil deoarece datele sunt păstrate în *buffer*-ul de ieșire și trimise ulterior. Activarea sau dezactivarea implicită a mecanismului se realizează în PHP prin intermediul directivei `output_buffering`, în *php.ini* (fișierul de configurare al PHP). Pentru activare, se atribuie directivei o valoare care fixează (în octeți) o limită superioară pentru *buffer*-ul de ieșire, de exemplu: `output_buffering=4096`. Dacă nu se dorește fixarea unei limite superioare, este necesar să se atribuie directivei valoarea `On`. Pentru dezactivarea implicită a mecanismului, se va atribui directivei valoarea `Off`.

Este posibil să se activeze mecanismul *output buffering* numai temporar, pe durata execuției unui script, prin utilizarea funcțiilor de control al ieșirii. Acestea sunt extrem de utile atunci când este necesar să se trimită anteturi HTTP browserului, *după* ce scriptul a început să genereze date de ieșire.

<oXygen/> XML Editor. Editor XML creat de compania *SyncRo soft*, din Craiova (<http://www.oxygenxml.com/>).

Poate fi utilizat pentru a lucra cu toate tehnologiile bazate pe XML, inclusiv cu bazele de date XML și cu serviciile Web. Suportul pentru XSLT și XQuery este îmbunătățit cu depanatoare avansate. Este disponibil pentru cele mai cunoscute sisteme de operare: Windows, Mac OS X, Linux și Solaris. Editorul poate fi folosit independent sau ca plugin în *Eclipse*. A se vedea și **Eclipse și XML**.

P-Q

pachet IP (*packet IP*). Numit și *pachet Internet* sau *datagramă Internet*, este o entitate care conține date și informații de livrare necesare pentru trimiterea lor într-o rețea IP, acestea din urmă fiind incluse într-un antet IP. Rolul IP-ului este acela de a găsi o rută (denumită și *cale*) pentru pachete. Antetul IP conține sursa și destinația (ca adrese IP), numărul de protocol și suma de control (pentru acest nivel). Numărul de protocol comunică mașinii-destinație că trebuie să transmită pachetul nivelului TCP sau UDP, la nivelul transport. Suma de control dă posibilitatea nivelului IP (*internet*) de pe mașina-destinație să verifice dacă pachetul a ajuns nealterat. A se vedea și **adresă IP**, **antet de mesaj** și **payload**.

Packard, David (1912–1996). Inginer american, cofondator, împreună cu William Hewlett, al companiei Hewlett-Packard (HP). În 1988 i-a fost acordată distincția *Presidential Medal of Freedom* de către președintele de atunci al SUA, Ronald Reagan. A se vedea și **Hewlett-Packard** și **Hewlett, William Reddington**.

packet switching (*comutare de pachete*). A se vedea **comutare de pachete**.

PageFlakes. Serviciu pentru personalizarea paginii de start (<http://www.pageflakes.com/>). A se vedea și **serviciu pentru personalizarea paginii de start**.

PageRank (*rangul paginii*). Nume și algoritm utilizate de motorul de căutare *Google Search*. Numele este marcă înregistrată a companiei, iar algoritmul a fost patentat, acesta aparținând Universității Stanford, unde a fost dezvoltat de Larry Page (care, împreună cu Serghei Brin, este cofondator Google). Compania Google deține o licență exclusivă de utilizare, acordată de Universitatea Stanford. A se vedea și **algoritm de căutare** și **Google Search**.

pagină de bază implicită (*default home page*). Fișierul furnizat de serverul Web atunci când este indicat un director, dar nu și un nume de fișier. În mod obișnuit, acesta este *index.htm*, *index.html*, *index.php*, *index.jsp* etc. Numele fișierului respectiv poate fi precizat/schimbat în fișierul de configurare (e.g., *httpd.conf* în cazul serverului *Web Apache*).

pagină HTML (*HTML page*). A se vedea **pagină Web**.

pagină orfană (*orphan page*). Pagină Web către care nu duce nici o legătură. A se vedea și **legătură** și **pagină Web**.

pagină Web (*Web page*). Document din World Wide Web vizualizat într-un browser. Poate fi statică sau dinamică. A se vedea și **pagină Web dinamică** și **pagină Web statică**.

pagină Web dinamică (*dynamic Web page*). 1. Pagină Web în care sunt incluse animații GIF, *applet*-uri Java sau controale ActiveX. A se vedea și **applet**, **controale ActiveX**, **GIF** și **pagină Web**. 2. Pagină Web cu o structură fixă și un conținut variabil, stabilit în funcție de opțiunile utilizatorului. Este generată folosind scripturi în limbaje de programare pentru server (*e.g.*, PHP). A se vedea și **pagină Web** și **PHP**.

pagină Web statică (*static Web page*). Pagină Web constând dintr-un document (X)HTML în care sunt incluse diverse resurse (imagini, animații etc.). A se vedea și **HTML**, **pagină Web** și **XHTML**.

paid inclusion (*includere plătită*). Taxă cerută de unele directoare Web pentru înscrierea site-urilor în baza lor de date (în special a celor comerciale). A se vedea și **director Web**.

Paint.NET. Program de editare grafică *raster* pentru Microsoft Windows, dezvoltat de *dotPDN* pe platforma *.NET Framework* (<http://www.getpaint.net/>). Dezvoltarea *Paint.NET* a început în anul 2004, sub licență MIT. A se vedea și **.NET Framework** și **licența MIT**.

Paint Shop Pro. Editor comercial de grafică *raster*, iar în versiunile recente, și de grafică vectorială pentru platforma Windows. A fost dezvoltat inițial de Jasc Software, care a fost achiziționată în 2004 de Corel Corporation. Funcționalitățile *Paint Shop Pro* pot fi extinse printr-o serie de plugin-uri *Photoshop*. A se vedea și **Corel Corporation** și **formate raster**.

PAN (*Personal Area Network*). Rețea de mici dimensiuni formată din calculatoare personale și alte dispozitive. O rețea de acest fel poate fi cu fir (utilizând porturile USB și *FireWire*) sau *wireless* (utilizând tehnologii ca *IrDA* și *Bluetooth*). A se vedea și **Bluetooth**, **FireWare**, **IrDA** și **USB**.

pană (*failure*). Întreruperea funcționării corecte a unui calculator sau a unui dispozitiv asociat acestuia (*e.g.*, imprimantă).

panou de control (*control panel*). Parte a interfețelor grafice ale sistemelor de operare Windows și Mac OS care îi permite utilizatorului să vadă și să manipuleze o serie de setări de bază prin intermediul unor *applet*-uri (*e.g.*, opțiuni de accesibilitate, adăugarea sau eliminarea unor dispozitive hardware și aplicații software, controlul conturilor de utilizator). A se vedea și **applet**.

Papert, Seymour (n. 1928). Matematician, informatician și profesor american de origine sud-africană. Este unul dintre pionierii inteligenței artificiale și inventatorul limbajului de

programare LOGO. A se vedea și **inteligență artificială** și **LOGO**.

paradigmă de programare (*programming paradigm*). Stil fundamental de programare. Diversele paradigme de programare diferă atât în ceea ce privește modalitățile de reprezentare a elementelor unui program (e.g., funcții, obiecte, variabile), cât și în ceea ce privește pașii care alcătuiesc un proces de calcul (e.g., atribuire, evaluare, fluxuri de date). Cele mai cunoscute paradigme de programare sunt programarea imperativă, programarea orientată pe obiect și programarea declarativă (un subtip al acesteia fiind programarea funcțională). În multe cazuri, un limbaj de programare suportă mai multe paradigme. A se vedea și **programare declarativă**, **programare funcțională**, **programare imperativă** și **programare orientată pe obiect**.

paralel (*parallel*). Ceea ce este transmis sau prelucrat în grupuri. De exemplu, în transmisia paralelă a datelor, între dispozitivele de comunicație sunt trimiși simultan mai mulți biți, pentru fiecare bit fiind utilizat câte un fir. A se vedea și **serial**.

parametru de tip referință (*reference parameter*). Parametru prin intermediul căruia se transferă funcției apelate adresa unei variabile, și nu valoarea acesteia. A se vedea și **funcție**.

parolă implicită (*default password*). Parola (e.g., de administrator) existentă atunci când un sistem este livrat de producător. Păstrarea parolei implicite

după instalarea sistemului este o neglijență foarte gravă, reprezentând un risc major de securitate. A se vedea și **cont implicit**.

partajarea datelor (*data sharing*). Punerea la dispoziție a datelor către mai mulți utilizatori. Operația este realizată de autorii sau de creatorii fișierelor care conțin datele partajate. Operația de partajare se face în general prin intermediul unei rețele de calculatoare, publică (e.g., Internet) sau privată. A se vedea și **Internet** și **rețea de calculatoare**.

partajare grafică (*photo sharing*). Operația de încărcare, etichetare și partajare a conținutului grafic folosind o aplicație Web. Cel mai cunoscut serviciu de partajare grafică este *Flickr*. A se vedea și **Flickr**.

partajare video (*video sharing*). Operație prin intermediul căreia se realizează încărcarea, marcarea și partajarea conținutului video folosind o aplicație Web. Cel mai cunoscut serviciu de *video sharing* este *YouTube*. A se vedea și **TeacherTube** și **YouTube**.

Pascal. Limbaj procedural, compilat și structurat, creat de Niklaus Wirth între anii 1968 și 1969 și publicat în 1970. Are la bază limbajul ALGOL, dar cu o sintaxă mai simplă și cu mai multe tipuri și structuri de date. Limbajul a fost dezvoltat cu scopul de a promova bunele practici în programare prin utilizarea programării structurate și a structurilor de date. *Object Pascal* a

fost derivat din *Pascal* cu scopul de a promova programarea orientată pe obiect. A se vedea și **ALGOL**, **programarea orientată pe obiect** și **Wirth, Niklaus**.

password sniffing (*detectarea parolilor*). Procesul de capturare (furt) a parolilor transmise într-o rețea (locală sau Internet), folosind un program de tip *sniffer*. A se vedea și **sniffer**.

payload (*sarcină utilă*). Datele propriu-zise recepționate de destinatar. Sunt obținute după înlăturarea din unitatea transmisă a informațiilor necesare pentru livrarea acesteia, incluse într-o secvență cunoscută adesea sub numele de antet (*header*). A se vedea și **antet de mesaj** și **pachet IP**.

pay-per-click (PPC). Model de publicitate pe Internet utilizat de motoarele de căutare, de rețelele de publicitate și de site-urile de conținut (e.g., bloguri) în care proprietarii reclamelor plătesc gazdele numai atunci când vizitatorii execută click pe acestea.

Există două tipuri de modele PPC: unul bazat pe licitație și altul în care se plătește un preț fix. Primul tip este folosit în cazul motoarelor de căutare, cei care doresc publicarea unor reclame licitând pentru a obține o poziție cât mai ridicată în ceea ce privește un cuvânt-cheie caracteristic țintei lor de piață. În schimb, în mod obișnuit, site-urile de conținut fixează un preț pentru fiecare click.

Cei mai mari operatori de PPC sunt *Google AdWords*, *Yahoo! Search Marketing* și *Microsoft adCenter*, toți operând cu tipul PPC bazat pe licitație. Costul pe click (CPC) variază în funcție de motorul de căutare și de nivelul, scăzut sau ridicat, al competiției pentru un anumit cuvânt-cheie.

PC. 1. Calculator personal realizat de IBM, numit și IBM PC. **2.** Calculator personal compatibil cu standardul elaborat de compania IBM pentru PC. **3.** Calculator personal (e.g., IBM PC, compatibil IBM PC, Apple). A se vedea și **calculator personal**.

PC de mână (H/PC, *handheld PC*). Calculator de dimensiuni mici, care permit ca acesta să fie ținut în mână. O serie de H/PC-uri rulează sistemul de operare Windows CE. Microsoft a oprit producția propriilor H/PC-uri în 2000, concentrându-se pe Pocket PC și Smartphone. A se vedea și **Pocket PC** și **smartphone**.

PCL (PCL, *Printer Control Language*). Dezvoltat de Hewlett-Packard, este un protocol pentru controlul imprimantei devenit standard industrial *de facto*. Propus inițial în 1984 pentru imprimantele cu jet de cerneală, a fost dezvoltat ulterior pentru diverse alte tipuri (e.g., matriciale, termice). A se vedea și **imprimantă**.

PCRE (*Perl Compatible Regular Expressions*). Bibliotecă PHP care implementează suportul pentru expresii regulate. A se vedea și **expresie regulată**.

PDA (*Personal Digital Assistant, asistent personal digital*). Calculator de dimensiuni mici, care poate fi ținut în mână (*palmtop*), cu numeroase funcții legate de organizarea activității (*e.g.*, agendă electronică, calculator, redarea fișierelor multimedia, editarea foilor de calcul, ceas cu alarme) și de comunicații (*e.g.*, accesarea serviciilor de *e-mail* și *Web*, receptare radio, GPRS). Unele PDA-uri pot fi folosite și ca telefoane inteligente sau pot comunica între ele folosind IrDA și Bluetooth. Termenul PDA a fost folosit pentru prima dată pe 7 ianuarie 1992 la târgul *Consumer Electronics Show*, pentru a desemna produsul Apple Newton. A se vedea și **Bluetooth**, **IrDA** și **telefon inteligent**.

PDF (*Portable Document Format*). Format de fișier cu extensia de fișier *.pdf*, creat de Adobe Systems în 1993. Tipul MIME principal este *application/pdf*. Formatul este utilizat pentru reprezentarea documentelor bidimensionale independent de aplicație, sistem de operare și hardware. Documentele în format PDF pot fi vizualizate folosind programul gratuit *Adobe Reader*, creat de Adobe Systems. A se vedea și **Adobe Reader** și **Adobe Systems**.

PDF Online. Serviciu *online* gratuit (<http://www.pdfonline.com/>) oferit de *BCL Technologies*, utilizat pentru conversia între formatele PDF, HTML și DOC și incluzând următoarele componente: *Doc2PDF* (convertește fișierele *Word* în format PDF);

PDF2Word (convertește fișierele PDF în DOC); *Web2PDF* (transformă paginile Web în fișiere PDF; această componentă se poate apela prin intermediul unui buton *widget* inclus în site-ul sau blogul utilizatorului, permițându-le vizitatorilor să salveze pagina respectivă ca fișier PDF; de asemenea, componenta oferă și informații referitoare la paginile pentru care s-a realizat conversia în PDF); *PDF Service for iPhone* (fișierele atașate unui mesaj de *e-mail* trimis la adresa *iPhone@pdfonline.com* vor fi convertite în PDF-uri și trimise înapoi prin *e-mail*; fișierele respective pot fi accesate ulterior de pe iPhone, laptop sau de pe orice alt dispozitiv de tip *handheld*). A se vedea și **PDF**.

PEAR (*The PHP Extension and Application Repository*). Cadru de lucru și sistem de distribuție pentru componente PHP reutilizabile (<http://pear.php.net/>). A se vedea și **PECL** și **PHP**.

PEAR::Text_CAPTCHA. Pachet PEAR dezvoltat de Christian Wenz sub licență PHP. Este utilizat pentru producerea imaginilor *antispam*. A se vedea și **CAPTCHA** și **PEAR**.

PEAR::Image_Graph. Pachet PEAR dezvoltat de Stefan Neufeind, Tobias Schlitt și Jesper Veggerby Hansen sub licență LGPL. Furnizează un set de clase utilizate pentru generarea diagramelor folosite în scopul reprezentării datelor numerice. Pachetul are la bază biblioteca *GraphPite*,

dezvoltată de Jesper Veggerby Hansen. A se vedea și **PEAR**.

PEAR::Image_Text. Pachet PEAR dezvoltat de Tobias Schlitt sub licență PHP. Este utilizat pentru manipularea textului în imaginile de tip *raster*. A se vedea și **formate raster** și **PEAR**.

PEAR::MP3_Playlist. Pachet PEAR utilizat pentru generarea dinamică a *playlist*-urilor MP3 în diverse formate, ca XML, RSS, XHTML și SMIL. A se vedea și **PEAR**.

PECL (*PHP Extension Community Library*, biblioteca de extensii a comunității PHP). Depozit de extensii PHP, furnizează un director al tuturor extensiilor cunoscute și găzduiește facilități pentru descărcarea și dezvoltarea acestora (<http://pecl.php.net/>). Sistemul de împachetare și distribuție utilizat de PECL este partajat cu PEAR. A se vedea și **PEAR** și **PHP**.

PeekYou. Motor de căutare fondat în aprilie 2006 de Michael Hussey. Indexează persoane și legăturile care se referă la acestea (<http://www.peakyou.com/>). A se vedea și **motor de căutare**.

penetrare (*penetration*). Ocolirea cu succes a mecanismelor de securitate ale unui sistem. A se vedea și **semnătură de penetrare** și **sistem de detectare a intruziunilor**.

Pentium. Microprocesor pe 32 de biți lansat în 1993 de compania Intel ca succesorul lui i486. Pentium este un microprocesor superscalar de tip x86. Arhitectura sa este o extensie directă a

arhitecturii 80486. În 1996 a fost introdus microprocesorul Pentium MMX, având aceeași arhitectură, completată cu instrucțiuni MMX (*MultiMedia eXtension*). Ulterior, au fost produse Pentium Pro (1995), Pentium II (1997), Pentium III (1999), Pentium 4 (2000), Pentium M (2003) și Pentium D (2005–2006). A se vedea și **microprocesor**.

Perception Authoring Manager. Produs al companiei QuestionMark, oferă posibilitatea creării, modificării și publicării testelor de evaluare (<http://www.questionmark.com/>).

Pentru stocarea datelor utilizează depozite create folosind baze de date *Microsoft Access*, *Oracle* sau *SQL Server*. Datele pot fi stocate într-un depozit local (pe calculatorul autorului, în baze de date *Access*) sau într-un depozit *online* partajat (la care autorul are acces prin intermediul unei rețele, în baze de date *Oracle* sau *SQL Server*). Depozitele partajate nu pot fi create folosind *Authoring Manager*. În acest scop trebuie utilizat instrumentul *Shared Repository Manager*, instalat odată cu *Perception Server*.

Înainte de a crea întrebările, trebuie creată o *temă* (*topic*) în a cărei structură vor fi stocate. O temă conține un grup de întrebări legate de un subiect anume. De asemenea, o temă poate include subteme. De exemplu, tema *Matematica* poate include două subteme, una denumită *Analiza* și alta *Algebra*. Dacă autorul dispune de întrebări într-un format extern, acestea

pot fi importate într-un depozit *Perception*. Metoda utilizată va fi aleasă în funcție de formatul conținutului care urmează a fi importat. Astfel, pot fi incluse (importate) întrebări din versiunile primare ale *Perception*, fișiere *Microsoft Word*, *Blackboard 6*, zip sau ASCII. A se vedea și **Testing and Assessment Tool**.

Perl (*Practical Extraction and Report Language*). Limbaj de programare procedural, interpretat, creat de Larry Wall în 1987. Împrumută caracteristici ale multor limbaje (e.g., C, AWK, Lisp). Perl beneficiază de un depozit (CPAN) care include numeroase module *open source*. A se vedea și **CPAN** și **Wall, Larry**.

PerlySMIL. Modul Perl specializat în generarea dinamică de prezentări SMIL (<http://www.webiphany.com/perlysmil/>). Componenta principală a modulului este obiectul `Smil`, care oferă diverse metode corespunzătoare elementelor care alcătuiesc structura generală a unui document SMIL. A se vedea și **SMIL**.

permisiuni (*permissions*). A se vedea **drepturi de acces**.

personal computer (PC, *calculator personal*). A se vedea **calculator personal**.

(a) **personaliza** (*customize*). Referitor la software, înseamnă: **1.** modificarea de către utilizator a unor aspecte ale interfeței grafice a acestuia, în limitele impuse de sistemul de operare sau de aplicație; **2.** modificarea unei aplicații

pentru un anumit client, ceea ce poate determina creșterea prețului acesteia.

(a) **petici** (*patch*). În programare, a rezolva temporar o problemă prin introducerea într-un program a unei porțiuni suplimentare de cod. În mod obișnuit, programul „peticit” este înlocuit într-un timp relativ scurt cu o nouă versiune (îmbunătățită).

phishing (*pescuit*). În comunicațiile electronice, activitate infracțională prin care se încearcă obținerea frauduloasă a unor informații confidențiale, folosind tehnici de manipulare a unei persoane sau a unei instituții. În mod obișnuit, un atac de tip *phishing* constă în trimiterea unui mesaj de *e-mail* în care se cere utilizatorului să-și divulge datele confidențiale (motivându-se că datele originale au fost pierdute din cauza unor probleme tehnice sau că acesta a câștigat un premiu) prin intermediul unui site Web care este o clonă a site-ului unei instituții financiare sau de comerț electronic. Informațiile solicitate sunt de obicei: numărul cardului de debit/credit, codul PIN pentru ATM, informații despre contul bancar, codul numeric personal, parole, adrese de *e-mail*, alte date personale. Conform unui raport Symantec, 5% din totalul site-urilor de tip *phishing* au fost detectate în România, ceea ce face ca țara noastră să se claseze – în acest domeniu – pe locul trei în lume, după China și Statele Unite, și pe primul loc în Europa.

photo sharing (*partajare grafică*). A se vedea **partajare grafică**.

phonecast. Combinație între *phone* și *podcast*, desemnează un conținut audio sau video distribuit pentru telefoanele mobile. A se vedea și **podcast**.

PHP (acronim recursiv pentru *PHP: Hypertext Preprocessor*). Limbaj de programare utilizat în special pentru dezvoltarea aplicațiilor Web dinamice. Ocupă poziția a treia în ceea ce privește popularitatea printre limbajele de programare, după C/C++ și Java.

Conform Netcraft, serverul de aplicații/limbajul PHP era utilizat în iulie 2009 de aproximativ 22 milioane de site-uri Web și de 1,3 milioane de adrese IP. Astfel, PHP este lider, având o cotă de peste 30 de procente și depășind ASP, Perl, Python, Ruby etc. Dintre aplicațiile (incluse în „sfera” Web 2.0) care utilizează PHP, pot fi menționate *Yahoo!*, *Flickr*, *Technorati*, *Facebook*, *Friendster* și *Tagged.com*.

PHP este un limbaj de *scripting* pentru server utilizat la scară largă, realizat și distribuit în sistem *open source* și destinat în special dezvoltării Web, codul PHP putând fi încapsulat în (X)HTML. Prin PHP se înțelege atât limbajul de scripting, cât și serverul de aplicații, cel din urmă având rolul de a procesa și interpreta scripturile în limbajul respectiv. Spre deosebire de JavaScript, codul PHP este interpretat pe server, utilizatorul primind numai rezultatul obținut, în diverse formate (e.g., (X)HTML).

Rasmus Lerdorf a creat în anul 1994 un *wrapper* CGI numit PHP (*Personal Home Page*), pentru a număra vizitatorii site-ului său. Ulterior (1995), el a adăugat acestuia elemente de interfață între utilizatori și bazele de date, pe care le-a denumit *Form Interpreters* (FI), rezultând astfel PHP/FI. În anul 1997, Zeev Surasky și Andi Gutmans (cofondatori ai companiei Zend Technologies (<http://www.zend.com>) au inițiat – ca pe un proiect universitar – dezvoltarea unui limbaj de programare Web pentru server bazat pe utilitarul PHP/FI și pe sintaxa C, pe care l-au denumit PHP (de data aceasta, acronim recursiv pentru *PHP: Hypertext Preprocessor*). În anul următor, cei doi (împreună cu Rasmus Lerdorf) le-au oferit utilizatorilor versiunea 3.0 a limbajului, care asigură – printre altele – suport limitat pentru programarea orientată pe obiect (POO). Versiunea 4.0 a PHP, bazată pe *motorul Zend* (*Zend engine*; denumirea provine de la prenumele autorilor, **Zeev** și **Andi**), a fost lansată oficial în anul 2002. PHP 5.0 (2004) se bazează pe versiunea 2 a motorului Zend și reprezintă un progres important în evoluția limbajului.

Printre particularitățile limbajului PHP care îl recomandă ca fiind cel mai indicat pentru dezvoltarea aplicațiilor Web în general și a celor Web 2.0 în particular pot fi amintite: este ușor de învățat și de utilizat, oferă funcționalități complete și este simplu de extins. Diverse module și biblioteci asigură lucrul PHP cu un număr mare de API-uri și alte tehnologii Web 2.0.

De exemplu, există un număr mare de instrumente PHP care facilitează crearea aplicațiilor Web bazate pe AJAX. Un alt factor care face din PHP un limbaj deosebit de atractiv pentru dezvoltarea aplicațiilor Web 2.0 este acela că au fost create medii de dezvoltare profesionale scrise în PHP, așa cum sunt *Zend Core* (proiectat în scopul dezvoltării de aplicații pentru bazele de date IBM și Oracle) și *Zend Framework* (cadru de lucru *open source* care simplifică dezvoltarea aplicațiilor).

Motorul Zend 2, „inima” celei de-a 5-a versiuni a serverului de aplicații PHP, asigură: implementarea structurilor de date utilizate în limbajul PHP; analiza sintactică, compilarea în memorie și execuția scripturilor; interfața cu modulele de extensie; o serie de servicii standard, dintre care se poate menționa managementul memoriei și al altor resurse.

Dintre multiplele caracteristici ale serverului de aplicații PHP, care se constituie în tot atâtea argumente în favoarea utilizării sale, pot fi menționate: dispune de interfețe pentru un număr mare de sisteme de gestiune a bazelor de date (e.g., Oracle, Microsoft SQL Server, IBM DB2, MySQL, Postgresql, Informix, Sybase), precum și pentru ODBC (care îi permite conectarea la alte servere de baze de date, prin intermediul *driver*-ului ODBC); asigură suport pentru comunicația cu servicii care utilizează diverse protocoale, dintre care amintim

HTTP, FTP (*File Transfer Protocol*), IMAP (*Internet Message Access Protocol*), POP3 (*Post Office Protocol version 3*), COM (*Component Object Model*), LDAP (*Lightweight Directory Access*), SOAP (*Simple Object Access Protocol*); implementează standardul POSIX Extins pentru prelucrarea textului folosind expresii regulate; implementează diverse metode de procesare a documentelor XML, dintre care pot fi amintite SAX (*Simple API for XML*), DOM (*Document Object Model*), XMLReader și SimpleXML, oferind și posibilitatea transformării acestora prin intermediul foilor de stil XSLT (*XSL Transformations*); asigură suport pentru generarea și prelucrarea imaginilor, gestiunea sesiunilor (utilizând variabile *cookie* și, respectiv, sesiune), instanțierea obiectelor Java și utilizarea lor în mod transparent ca obiecte PHP, comerț electronic (funcții de plată Cybercash, VeriSign Pay Flo etc.), crearea animațiilor *flash* etc.

Serverul de aplicații PHP poate fi utilizat pe cele mai multe platforme (e.g., UNIX/Linux, Windows, Mac OS X), împreună cu un număr mare de servere Web (e.g., Apache, Microsoft IIS, iPlanet, Xitami, OmniHTTPd). De obicei, PHP este folosit ca modul inclus în procesul server HTTP, iar în cazul în care serverul amintit suportă standardul CGI (*Common Gateway Interface*), poate fi utilizat ca procesor CGI.

Atunci când browserul cere un fișier PHP serverului Web, acesta trimite

cererea procesorului PHP, care efectuează interpretarea scriptului și produce la ieșirea standard conținut în diverse formate (e.g., XHTML, XML, PDF, GIF, JPEG, PNG). Codul PHP este delimitat de marcate de start (de exemplu `<?php`) și de sfârșit (de exemplu `?>`), acestea determinând intrarea și, respectiv, ieșirea procesorului în/din „modul PHP”. Se pot obține o mulțime de informații utile despre serverul de aplicații PHP și despre extensiile instalate în sistem odată cu acesta, prin intermediul funcției `phpinfo()`, utilizată astfel: `<?php phpinfo(); ?>`.

PHP-Fusion. CMS *open source* de categorie „ușoară”, scris în PHP. Folosește o singură bază de date MySQL pentru stocarea întregului conținut al site-ului (<http://php-fusion.co.uk/news.php>). A se vedea și **CMS** și **PHP**.

PHP-Nuke. CMS *open source* scris în PHP, flexibil în exploatare (<http://phpnuke.org/>). Dispune de instrumente de actualizare a conținutului, fiind una dintre cele mai accesibile soluții de tip portal din Internet. A se vedea și **CMS** și **PHP**.

php.ini. Numele fișierului de configurare PHP. Fișierul este citit la fiecare pornire a interpretorului PHP, astfel: dacă PHP rulează ca modul în procesul server Web, este citit la fiecare pornire a serverului Web; dacă PHP funcționează ca proces CGI, fișierul va fi citit la fiecare invocare a interpretorului (i.e. de fiecare dată când serverul Web

primește o cerere pentru execuția unui script PHP).

Fișierul *php.ini* conține, în ordine, următoarele secțiuni (sunt indicate numai o parte dintre cele existente): opțiuni privind limbajul (*Language Options*); limite de resurse (*Resource Limits*); manipularea erorilor și înregistrarea acestora (*Errors handling and logging*); manevrarea datelor (*Data Handling*); căi și directoare (*Paths and Directories*); încărcarea fișierelor (*File Uploads*); extensii dinamice (*Dynamic Extensions*); setări ale modulelor (*Module Settings*). Fiecare secțiune include o serie de directive de configurare. Dacă se dorește ca directivele să aibă efectul scontat, este necesar ca acestea să fie comentate prin eliminarea caracterului de comentare (`:`) care le precede. A se vedea și **PHP**.

PHP Classes. Site care include un număr mare de clase scrise în limbajul PHP (<http://www.phpclasses.org/>). Un cont gratuit le asigură utilizatorilor posibilitatea de a încărca și descărca materiale. A se vedea și **PHP**.

PHP DOM. API scris în limbajul de programare PHP și utilizat pentru procesarea documentelor XML. Are următoarele caracteristici: este orientat pe obiect; suportă limbajul XPath, validări de documente și spații de nume; asigură interoperabilitate cu SimpleXML și limbajul XSLT; permite acces rapid prin XPath la arborele XML, precum și manipularea acestuia. API-ul implementează DOM de nivel 3, incluzând o serie de clase predefinite,

care pot fi clasificate în două categorii: clase de bază și clase extinse.

PhpEd. IDE/editor comercial complet personalizabil, care „știe” HTML PHP, Perl, Python. Este oferit de *NuSphere* (<http://www.nusphere.com/>). A se vedea și **IDE**.

PHPEdit. IDE PHP comercial pentru profesioniști care include o serie de instrumente puternice (<http://www.waterproof.fr>). A se vedea și **IDE** și **PHP**.

phpHtmlLib. Cadru de lucru scris în PHP bazat pe programarea orientată pe obiect (POO), utilizat pentru dezvoltarea aplicațiilor Web (<http://phphtml.lib.newsblob.com/>). Suportă modelele MVC și AJAX, oferind posibilitatea dezvoltării aplicațiilor RIA. A se vedea și **MVC**, **PHP**, **programare orientată pe obiect**, **RIA** și **SVG**.

phpMyAdmin. Instrument scris în PHP, utilizat pentru administrarea bazelor de date MySQL prin intermediul unei interfețe Web (<http://www.phpmyadmin.net>). A se vedea și **MySQL** și **PHP**.

Picasa. Aplicație gratuită pentru editarea și organizarea fotografiilor digitale (<http://picasa.google.com/>). A fost creată de Idealab și este deținută începând cu iulie 2004 de Google. Are versiuni pentru sistemele de operare Windows XP, Windows Vista și Mac OS X. A se vedea și **Google**, **Inc**.

PIN (*Personal Identification Number, număr personal de identificare*). Cod numeric cu caracter confidențial, format de obicei din patru cifre, folosit de un utilizator autorizat pentru accesarea unui serviciu, sistem (*e.g.*, bancomat). Majoritatea sistemelor accesate pe baza unui PIN au și un contor de încercări, astfel că după un număr de tentative nereușite de autentificare a unui utilizator, sistemul blochează contul și – uneori – reține chiar cardul de acces, considerându-se că a avut loc o tentativă de fraudă. PIN-ul a fost inventat de James Goodfellow. A se vedea și **Goodfellow**, **James**.

ping. Program care folosește protocolul ICMP pentru a determina dacă un nod este sau nu accesibil prin intermediul unei rețele IP. În acest scop, trimite mesaje ICMP *echo request* (solicitare de răspuns) prin pachete adresate nodului vizat și așteaptă răspunsul ca ICMP *echo response*. Pachetele sunt transmise periodic, *ping* calculând (pe baza rezultatelor obținute) timpul de *round-trip* (dus-întors) al pachetelor și rata de pierdere a acestora. A se vedea și **ICMP**, **IP** și **pachet IP**.

piraterie (*piracy*). Referitor la produsele informatice, reprezintă utilizarea și/sau distribuirea neautorizată a unui program de calculator.

pixel (*pixel*). Cea mai mică parte a unei imagini digitale afișate (*picture elements*).

pixlr. Editor *online* pentru imagini (<http://pixlr.com/>). Imaginile create pot

fi salvate pe calculatorul utilizatorului, în format JPEG, PNG sau BMP.

placa de bază (*motherboard*). Circuit central cablat existent în diverse dispozitive electronice, cel mai cunoscut dintre acestea fiind calculatorul personal.

Placa de bază a unui PC include: *socket*-uri (sau sloturi) în care sunt instalate unul sau mai multe CPU-uri (*i.e.*, microprocesoare), sloturile în care sunt instalate modulele memoriei principale (RAM), *chipset*-ul (circuit integrat complex, interfață între FSB, memoria principală și magistralele periferice), memoria ROM (care conține *firmware*-ul sau BIOS-ul), generatorul de ceas (produce un semnal de ceas pentru sincronizarea diverselor componente ale calculatorului), slot-uri pentru plăcile de extensie, conectori pentru dispozitivele de intrare (*e.g.*, mouse, tastatură), controlere de disc, porturi seriale etc. A se vedea și **chipset**.

placă grafică (*graphics card*). Numită și placă video, adaptor video sau placă de accelerare grafică, este un dispozitiv utilizat pentru a genera date (imagini) care vor fi afișate pe ecran. Majoritatea plăcilor grafice moderne oferă funcții avansate (*e.g.*, redarea scenelor 3D, captură video, ieșire TV, tuner TV). Pentru rularea pe calculator a jocurilor complexe sunt necesare plăci grafice cu performanțe ridicate. Dintre producătorii cunoscuți, pot fi amintiți NVIDIA și ATI, iar dintre plăcile grafice populare, GeForce (NVIDIA) și Radeon (ATI).

planificator (*scheduler*). Proces al sistemului de operare cu rolul de a lansa, gestiona și încheia programe, precum și de a aloca resursele sistemului, pe baza unui algoritm. A se vedea și **algoritm de planificare**.

Plankalkül. Primul limbaj de programare de nivel înalt, dezvoltat între 1943 și 1945 de Konrad Zuse (pentru o arhitectură non-von Neumann). A se vedea și **limbaj de nivel înalt**.

platformă (*platform*). Tip de arhitectură hardware sau de cadru de lucru software care permite unei aplicații să ruleze. În mod obișnuit, o platformă include arhitectura calculatorului, sistemul de operare, limbajele de programare și bibliotecile legate de acestea. Adesea, se vorbește despre platformă hardware (*e.g.*, supercalculator, Macintosh, x86, RISC), platformă sistem de operare (*e.g.*, Linux, Mac OS, Microsoft Windows, Solaris) și platformă software (*e.g.*, .NET Framework, Java, Adobe AIR). A se vedea și **cadru de lucru software** și **sistem de operare**.

Plone. Unul dintre cele mai avansate CMS-uri, identificat de CMSReport și eWeek drept una dintre cele trei variante *open source* din gama CMS *enterprise* (<http://plone.org/>). A se vedea și CMS.

plotter (*plotter*). Numit și trasator, este un dispozitiv de ieșire care realizează desene de mare precizie (*e.g.*, hărți, desene tehnice) pe hârtie sau pe un alt material, în funcție de comenzile

transmise de un calculator. Dispozitivul include un modul de trasare, un bloc de control al trasării, o unitate logică și o interfață cu calculatorul. A se vedea și **dispozitiv de ieșire**.

plug and play (*conectează și folosește*). Caracteristică a unui calculator sau a unui dispozitiv care permite configurarea automată a unei componente hardware atașate sistemului în timpul funcționării sale, fără să fie necesară repornirea acestuia sau intervenția utilizatorului pentru rezolvarea unor conflicte de resurse. Pentru realizarea configurării automate, este necesar ca și dispozitivul atașat sistemului să fie de tip *plug and play*.

PNG (*Portable Network Graphics*). Format *raster* gratuit, dezvoltat de un consorțiu de firme specializate în software grafic. Deoarece algoritmul de comprimare LZW al formatului GI dezvoltat de CompuServe era proprietar Unisys Corporation, orice dezvoltator al unui software grafic care salva în formatul GIF trebuia să plătească acestei corporații drepturi de autor (începând cu anul 1995). Pentru a ocoli situația, devenită piedică în calea utilizării libere a formatului, consorțiul a dezvoltat PNG.

Prima versiune (cunoscută ca PNG-8) este similară formatului GIF. Poate utiliza cel mult 256 de culori și suportă transparența pe un bit (adică un pixel poate fi opac sau transparent). Fișierele PNG-8 sunt mai mici decât echivalentele lor GIF. A doua versiune (cunoscută ca PNG-24) utilizează

culori pe 24 de biți. Cu toate acestea, nu este indicat să se înlocuiască formatul JPEG cu PNG, deoarece acesta din urmă suportă numai comprimarea fără pierderi, fișierele PNG-24 fiind mai mari în raport cu fișierele JPEG comprimate. Unul dintre cele mai mari avantaje ale formatului PNG-24 este suportul pentru transparența pe 8 biți, ceea ce înseamnă că o imagine poate avea transparența variabilă (graduală) pe întinderea sa, opacitatea unui pixel fiind cuprinsă între 0 (transparență totală) și 255 (opacitate totală).

Formatul PNG nu are suport pentru animație. Totuși, extensia MNG (*Multiple-image Network Graphics*) asigură posibilitatea realizării animației. Cele mai multe browsere nu includ suportul pentru MNG. A se vedea și **formate raster**.

Pocket PC. Specificație pentru PDA-urile (*Personal Digital Assistant*) care rulează sistemul de operare Microsoft Windows Mobile. De asemenea, Pocket PC – având multe dintre caracteristicile PC-urilor actuale – poate rula un sistem de operare alternativ (*e.g.*, NetBSD, Linux, Android). PDA-urile Pocket PC sunt produse de diverse companii, cele mai cunoscute dintre acestea fiind HP (Hewlett-Packard), Toshiba, Acer, ASUS, Dell și Fujitsu-Siemens.

podcast (*podcast*). Colecție de fișiere media (audio și/sau video) distribuite în Internet folosind tehnologia *feed*-urilor RSS, în scopul redării lor pe dispozitive media portabile și pe calculatoare. Termenul, utilizat pentru prima oară în

februarie 2004 de Ben Hammersley într-un articol apărut în cotidianul *Guardian*, a fost declarat cuvântul anului 2005 de *New Oxford American Dictionary*.

Deși *podcast* este o combinație între termenii *broadcasting* și *iPod*, o astfel de înregistrare nu necesită obligatoriu un dispozitiv iPod pentru a fi audiată, utilizatorul putând să o descarce și să o asculte oricând și oriunde dorește, folosind propriul său dispozitiv de redare (audio/MP3 *player*).

Apple a inclus o secțiune dedicată *podcast*-urilor în programul său *iTunes*, care poate fi instalat gratuit pe calculatoarele Mac și PC. Acesta este un *player* audio/video, dar poate fi utilizat și pentru a descărca legal muzică din *Apple iTunes Music Store*. Apple este un promotor major al distribuirii de conținut media digital via *podcast* prin intermediul magazinului *online* amintit.

Termenul *podcast* desemnează fișierul/fișierele media, dar și metoda de distribuire (denumită, ulterior, *podcasting*). Autorul unui *podcast* este denumit *podcaster*. Uneori, însuși *feed*-ul RSS poartă numele de *podcast*. În acest caz, un fișier media inclus în *feed* se numește *episod* (*episode*). Deși numărul de episoade care pot fi incluse într-un *feed* *Podcast* nu este limitat, se obișnuiește ca acesta să fie egal cu unu. Un *podcast* se deosebește de alte formate digitale media prin posibilitatea de a fi distribuit, de a permite subscrierea (abonarea) la *feed*-ul RSS

corespunzător și de a fi descărcat automat, folosind un agregator/cititor de *feed*-uri. Multe *podcast*-uri permit descărcarea directă a conținutului media de către utilizatori, prin intermediul unor linkuri către fișierele audio/video corespunzătoare, incluse în *feed*-urile RSS sau în paginile Web.

Exploatarea la întregul potențial a tehnologiei *podcast* necesită utilizarea pentru redare a unui software adecvat, numit adesea *podcatching client*. Cel mai cunoscut dintre acestea este *playerul iTunes* al firmei Apple. Dintre alternative pot fi menționate: *Zune Marketplace* (*Microsoft*), *Juice* (agregator multiplatformă, fost *iPodder*), *Doppler* (cu versiuni pentru Windows și pentru dispozitive mobile), *Podget* (pentru Linux), *Podracer* (pentru Linux), *RSSRadio*.

Utilizatorii pot urmări conținutul media oferit prin intermediul *podcast*-urilor folosind oricare dintre următoarele trei modalități: un dispozitiv de tip MP3 *player*, un *player* software instalat pe un calculator, tehnologia VoIP (*Voice over IP*) – prin apelarea unui număr de telefon virtual.

Deși, inițial, *podcast*-urile au fost utilizate pentru distribuirea de emisiuni radio, acestea au început să fie folosite rapid într-o varietate de moduri. Astfel, *podcast*-urile sunt utilizate pentru distribuirea de spectacole audio/video, lecții (în sistemul educațional), tururi virtuale ale muzeelor, *talk-show*-uri, interviuri, tutoriale, instrucțiuni, comentarii. Alături de *podcast* sunt

utilizați o serie de termeni înrudiți care se referă la tehnologii similare: *vodcast* (sau *vidcast*), *blogcast*, *phonecast*, *screencast* și *sketchcast*. A se vedea și **feed**, **iTunes** și **RSS**.

podcatching client (*client pentru podcast-uri*). Client utilizat pentru citirea *feed*-urilor *podcast* și pentru redarea conținutului media. A se vedea și **podcast**.

pointer (*pointer*). În programare, variabilă care conține adresa de memorie a unor date, și nu datele propriu-zise.

pointer înainte (*forward pointer*). Într-o listă înlănțuită, conține adresa următorului element din listă. A se vedea și **listă înlănțuită** și **pointer înapoi**.

pointer înapoi (*backward pointer*). Într-o listă dublu înlănțuită, conține adresa elementului anterior din listă. A se vedea și **listă înlănțuită** și **pointer înainte**.

polimorfism (*polymorphism*). Concept utilizat în paradigma programării orientate pe obiect (POO). Reprezintă abilitatea obiectelor aparținând unor tipuri diferite de a răspunde apelurilor unor metode cu același nume, fiecare obiect având un comportament specific. Astfel, prin polimorfism se execută acțiuni diferite prin mesaje cu semnături (nume) identice asupra unor obiecte de tip diferit (*i.e.*, obiecte obținute din clase diferite răspund la același mesaj). A se vedea și **programare orientată pe obiect**.

Ponturi fierbinți. Director Web românesc, este un catalog al site-urilor din domeniul afacerilor. Este organizat pe categorii (*e.g.*, *Afaceri imobiliare*, *Afaceri pe Internet*, *Auto-Moto*, *Locuri de muncă*, *Investiții și credite*, *Magazine online*) și subcategorii (<http://ponturifierbinti.com/>). A se vedea și **director Web**.

POO (*programare orientată pe obiect*). A se vedea **programare orientată pe obiect**.

pop (*a scoate din stivă*). A elimina ultimul element introdus în stivă (cel mai de sus). A se vedea și **push** și **stivă**.

POP3 (*Post Office Protocol*). Versiunea 3 a protocolului POP, care funcționează la nivelul aplicație al modelului de referință TCP/IP, utilizat de un client local de *e-mail* pentru regăsirea mesajelor stocate pe un server de poștă electronică. Mesajele sunt descărcate mai întâi pe calculatorul personal și apoi citite. A se vedea și **IMAP4** și **poșta electronică**.

port (*port*). Număr alcătuit din doi octeți (16 biți), cuprins între 0 și 65535. În modelul client-server, utilizat pentru comunicații în Internet, identifică un program server. Pe un calculator pot fi instalate simultan mai multe programe server care primesc apeluri de la clienți. Pentru ca apelurile să ajungă la programul server corect, fiecare dintre acestea „ascultă” un anumit port. Cele mai multe porturi sunt neatribuite. Valorile mai mici de 1024 sunt rezervate pentru servicii predefinite

(*well known ports*, porturi general cunoscute). De exemplu, orice proces care dorește să stabilească o conexiune cu o mașină pentru a transfera un fișier utilizând serviciul FTP, se va conecta la portul 21 al mașinii-destinație pentru a contacta serverul FTP. Alte aplicații utilizează numere de porturi mai mari decât 1024 (e.g., serverul MySQL folosește implicit portul 3306). De asemenea, programatorii pot utiliza pentru propriile lor aplicații porturi mai mari de 1024. A se vedea și modelul client-server.

Portalul Ministerului Educației și Cercetării. Pus la dispoziția publicului începând cu anul 1998, prezintă activitatea Ministerului Educației și Cercetării, publicând informații și documente oficiale utile cadrelor didactice, elevilor și părinților (<http://www.edu.ro>). Acestea sunt incluse în următoarele secțiuni: Învățământ preuniversitar, Învățământ superior, Învățământ pentru minorități, Cercetare, Resurse umane, Afaceri europene și Relații internaționale. Pe acest site sunt descrise metodologiile de organizare și desfășurare a examenelor naționale, sunt oferite programele de examen și modele de subiecte.

portal Web (*Web portal*). Prezintă informații din diverse surse și oferă diferite servicii, într-o modalitate unitară. Separat de motoarele de căutare, portalurile Web oferă alte servicii, ca: *e-mail*, știri, petrecerea timpului liber, diverse informații. Iată câteva exemple de portaluri Web: MSN, Yahoo!, AOL.

portabil (*portable*). Software capabil să ruleze pe mai multe sisteme de calcul sau sisteme de operare.

port de imprimantă (*printer port*). Port utilizat pentru conectarea unei imprimante la un calculator. În cazul PC-urilor, în mod obișnuit porturile de imprimantă sunt cele paralele, identificate prin numele logic LPT. De asemenea, și porturile seriale, identificate prin numele logic COM, pot fi folosite pentru conectarea imprimantelor. A se vedea și **imprimantă, interfață paralelă și interfață serială**.

POS (*point of sale, punct de vânzare*).

1. Punct în care se plătește marfa achiziționată într-un magazin, dotat cu un scanner pentru citirea codurilor de bare, conectat la un sistem electronic de tranzacții. 2. Modul destinat înregistrării și procesării tranzacțiilor efectuate în cadrul vânzărilor cu amănuntul (vânzări prin bonuri fiscale, rapoarte zilnice și periodice, evidența stocurilor).

POSIX (*Portable Operating System Interface for UNIX*). Standard IEEE care definește un set de proceduri de bibliotecă pe care trebuie să le suporte orice sistem UNIX. Majoritatea acestor proceduri invocă un apel sistem, dar numai câteva dintre ele pot fi implementate în afara nucleului. Astfel, producătorul unui software care utilizează numai proceduri POSIX este sigur că acel program va rula pe orice sistem UNIX. A se vedea și **UNIX**.

post (*post*). Articol adăugat de un *blogger* în blogul său. Posturile mai sunt denumite și *intrări* (*entries*). Cele mai noi posturi sunt afișate primele, conform regulii ultimul intrat, primul listat. Pentru fiecare post este specificată data și ora publicării. A se vedea și **blog**, **blogger** și **blogging**.

Postei, Jonathan Bruce (1943–1998). Inginer și informatician american, a adus contribuții deosebite în dezvoltarea standardelor pentru Internet. Este cunoscut ca editor al RFC (*Request for Comments*) și administrator al IANA (*Internet Assigned Numbers Authority*). În onoarea sa au fost create premiul Postel (*Postel Award*), acordat de Internet Society, și Centrul Postel (în cadrul *Information Sciences Institute*). A se vedea și **IANA**, **Internet Society** și **RFC**.

PostNuke. CMS *Open Source*, reprezentând un sistem de administrare colaborativ, pentru comunități și conținuturi (<http://www.postnuke.com/>). A se vedea și **PHP**.

PostScript. Cel mai cunoscut limbaj de descriere a paginii, dezvoltat începând cu anul 1982 de John Warnock și Charles Geschke, ambii de la Adobe Systems. Limbajul folosește comenzi alcătuite din cuvinte în limba engleză, pentru a realiza diverse operațiuni asupra paginii (*e.g.*, machetare, încărcarea și scalarea fonturilor). Au fost create trei versiuni: PostScript Level 1 (1984), PostScript Level 2 (1991) și PostScript Level 3 (1997). A se vedea

și **font PostScript** și **limbaj de descriere a paginii**.

poșta electronică (*electronic mail*). Unul dintre cele mai populare servicii ale Internetului. Abreviată *e-mail*, *email* sau pur și simplu *mail*, oferă posibilitatea de a compune, trimite, recepta și stoca mesajele în sistemele electronice de comunicație. Serviciul permite transmiterea de mesaje între utilizatori, indiferent de locul unde se află aceștia, cu o viteză foarte mare.

Primul program de poștă electronică a fost creat în anul 1971 de Ray Tomlinson, programator la compania BBN. În anul următor, Larry Roberts, cercetător la MIT, a scris primul său utilitar de management al poștei electronice pentru listarea, citirea, înregistrarea, expedierea și răspunderea la mesaje.

Din punct de vedere formal, serviciul de poștă electronică seamănă cu poșta clasică, rolul celei din urmă fiind preluat de un calculator denumit *server de e-mail*. Pentru a fi clientul acestuia, utilizatorul trebuie să dispună de un cont de *e-mail*, care îi asigură o adresă și un spațiu de stocare (*i.e.*, o casuță poștală), precum și instrumente pentru gestionarea mesajelor.

Pentru trimiterea mesajelor de *e-mail* de la expeditor la server și între servere este folosit protocolul SMTP (*Server Mail Transfer Protocol*). În schimb, pentru primirea mesajelor este utilizat protocolul POP3 sau IMAP (*Internet Message Access Protocol*), o variantă îmbunătățită a POP3. Unele servicii de

poștă electronică oferă accesul la conturi prin intermediul unor clienți cu interfață Web. Există destul de multe site-uri care acceptă găzduirea unor conturi gratuite de *e-mail*, dintre care cele mai populare sunt *Yahoo! Mail* (<https://mail.yahoo.com/>) și *Google Mail* (<https://www.gmail.com>). A se vedea și **adresă de e-mail, IMAP, POP3, SMTP și Tomlinson, Ray.**

potrivire (*matching*). Procesul prin care se caută un articol de date (aflat, de exemplu, într-un document sau într-o bază de date) identic cu o cheie folosită drept criteriu de căutare.

PowerPC. Arhitectură de microprocesor elaborată începând cu 1991 de firmele Motorola, IBM și Apple. Un microprocesor PowerPC este de tip RISC, cu o magistrală de adrese pe 32 de biți și o magistrală de date pe 64 de biți. Acest tip de microprocesor este utilizat în calculatoarele produse de cele trei firme, în sistemele de jocuri Nintendo GameCube și Wii, în Playstation 3 (produs de Sony) și Xbox 360 (produs de Microsoft), precum și în diverse sisteme înglobate. A se vedea și **CPU.**

PPP (*Point-to-Point Protocol*). Protocol la nivelul legăturii de date al modelului de referință ISO/OSI, utilizat pentru a stabili o legătură directă între două noduri ale unei rețele (RFC 1661). PPP are facilități pentru autentificare, compresie și criptare. Protocolul este utilizat în multe tipuri de rețele fizice, de exemplu în cele realizate prin cablu

serial, linie telefonică, telefoane celulare, legături radio specializate, legături prin fibră optică. Furnizorii de servicii Internet utilizează două forme de încapsulare a cadrelor PPP pentru a oferi conectare la Internet folosind DSL (PPoE și PPPoE). A se vedea și **PPPoE.**

PPPoE (*Point-to-Point Protocol over Ethernet*). Protocol de rețea pentru încapsularea cadrelor PPP în cadre Ethernet (RFC 2516). PPPoE oferă simularea unei conexiuni *dial-up* peste o conexiune Ethernet, fiind utilizat mai ales cu serviciile DSL. Protocolul prezintă facilități pentru autentificare, compresie și criptare.

Protocolul PPPoE are două faze distincte: *discovery* și *session*. În prima fază, clientul inițializează sesiunea de autentificare printr-o cerere *broadcast*. În cea de-a doua fază, serverul îi răspunde cu adresa MAC proprie și îi prezintă oferta de conectare, clientul se autentifică cu un nume de utilizator și o parolă, sesiunea fiind astfel stabilită. A se vedea și **PPP.**

Prado (*PHP Rapid Application Development Object-oriented*). Cadru de lucru bazat pe componente și pe programarea condusă pe evenimente (<http://www.pradosoft.com/>).

precedența operatorilor (*operator precedence*). Ordinea în care sunt efectuate operațiile (cărora le corespund anumiți operatori) într-o expresie utilizată în matematică și programare. Ordinea standard (în sensul scăderii precedenței) este următoarea: ridicare

la putere și extragere rădăcină, înmulțire și împărțire, adunare și scădere. În absența parantezelor, se execută operațiile cu precedență mai mare. Această ordine poate fi modificată prin folosirea parantezelor. A se vedea și **operator**.

precizia eșantionării (*sampling precision*). Măsurată în biți, reprezintă precizia cu care un eșantion exprimă mărimea fizică eșantionată. Astfel, un eșantion pe 8 biți exprimă un semnal cu o precizie de $1/256$ (i.e., $1/2^8$) din intervalul de variație al acestuia. A se vedea și **eșantionare** și **rată de eșantionare**.

prelucrare distribuită (*distributed processing*). Modalitate de prelucrare a datelor în care sunt implicate mai multe calculatoare distincte, legate într-o rețea de comunicații.

Prensky, Marc (n. 1946). Autor american, proiectant și consultant în domeniul educațional. A introdus expresia „copiii născuți în epoca digitală” (*digital natives*).

premiul Turing (*Turing Award*). Distingție decernată în fiecare an de ACM (*Association for Computing Machinery*) unor personalități care au adus contribuții deosebite de natură tehnică în domeniul informaticii. Este numit și „premiul Nobel în informatică”. Premiul – sponsorizat și de companiile Google și Intel – poartă numele lui Alan Turing, considerat „părintele” informaticii teoretice și al inteligenței artificiale. Primul laureat

al premiului Turing a fost Alan Pernis. Prima femeie care a primit acest premiu (în 2006) a fost Frances E. Allen (de la IBM). A se vedea și **Turing, Alan Mathison**.

Preview Blum. Previzualizator de feed-uri oferit de Grazr (<http://www.grazr.com>), care se poate instala ca *add-on* în browserele IE 6, IE 7, Firefox 2.0, Opera și Safari. Aplicația permite detectarea automată și afișarea sub forma unui meniu vertical a *feed*-urilor RSS și Atom conținute într-o pagină Web, precum și vizualizarea acestora. A se vedea și **feed**.

previzualizare înainte de tipărire (*print preview*). Operație specifică procesoarelor de texte și altor aplicații, care permite afișarea pe ecran a documentului în forma în care va fi tipărit, de obicei înainte de trimiterea acestuia către imprimantă.

prezentare multimedia interactivă (*interactive multimedia presentation*). Prezentare multimedia cu care utilizatorul poate interacționa și care include imagini, documente și secvențe video. Permite adăugarea comentariilor de către membrii unui grup, folosind vocea (prin intermediul unui microfon), textul, telefonul, fișiere audio și fișiere video (prin intermediul unui webcam). A se vedea și **VoiceThread**.

prietenos cu utilizatorul (*user-friendly*). Referitor la un software (e.g., aplicație Web), ușor de învățat și de utilizat.

primitivă grafică (*graphics primitive*). Un bloc de bază utilizat în programele de grafică (e.g., un punct, o linie, o curbă). Primitivele sunt combinate într-un astfel de program pentru a obține imagini.

principii de integritate (*integrity principles*). Reguli care trebuie respectate de un SGBDR. Acestea sunt: principiul integrității domeniului (stabilește constrângeri de domeniu care precizează restricții asupra valorilor permise ale atributelor), principiul integrității entităților (într-o relație de bază nici un atribut al unei chei primare nu poate fi *null*) și principiul integrității referențiale (dacă într-o relație există o cheie străină, valoarea acesteia trebuie să coincidă cu valoarea unei chei-candidat a unei înregistrări în relația sa de bază sau să fie în întregime *null*). Un SGBDR care îndeplinește cele trei principii de integritate este un sistem total relațional. A se vedea și **SGBDR**.

privilegii de acces (*access privileges*). Numite pe scurt *privilegii* sau *permisiuni*, reprezintă tipul operațiilor pe care un utilizator le poate efectua asupra unei resurse din sistem sau din rețea.

Pe platforma UNIX/Linux privilegiile sunt organizate în trei grupuri: primul grup conține permisiunile acordate proprietarului fișierului; al doilea grup include permisiunile acordate membrilor grupului în proprietatea căruia se află fișierul; al treilea grup conține privilegiile acordate celorlalți utilizatori. Un grup conține trei secvențe

de permisiuni. O permisiune este reprezentată prin intermediul unui caracter care poate fi una dintre literele *r*, *w* și *x* sau cratima (-), cu semnificațiile următoare: *r* semnifică permisiunea de citire (*read*); *w* semnifică permisiunea de scriere (*write*); *x* semnifică permisiunea de execuție a fișierului; cratima semnifică absența uneia dintre permisiunile *r*, *w* sau *x*. Primele trei permisiuni amintite sunt plasate în ordinea *rwX*, existând posibilitatea ca, într-un grup, una, două sau chiar toate să fie înlocuite cu cratimă.

procedură (*procedure*). Modul de program scris independent de programul principal (*main program*), având un rol de sine stătător. Procedura este legată de programul principal printr-un proces de transfer al execuției codului și apoi de revenire la acesta. Avantajele utilizării procedurilor (i.e., programării modulare) sunt: reutilizarea simplă a codului compilat, întreținere ușoară, permite lucrul în echipă, oferă o bună lizibilitate a codului. Procedura recursivă este cea care se autoapelează. Procedura reentrantă poate fi întreruptă de un proces și apoi reapelată de alt proces (este specifică *multitasking*-ului). Termenul este înrudit cu funcție, rutină și subrutină. A se vedea și **funcție, rutină și subrutină**.

proces (*process*). Instanță a unui program de calculator, constând din unul sau mai multe fire de execuție (*threads*). Procesul este executat secvențial de un sistem de calcul care are capacitatea de a rula concurent mai

multe programe și include în plus față de program informațiile de stare legate de execuția programului respectiv (e.g., stiva, valorile regiștrilor). De asemenea, este posibil ca un program să fie format din mai multe procese care pot să ruleze în paralel. În mod obișnuit, acest tip de execuție se realizează alocând pe rând procesorul câte unui proces.

Un proces se află în execuție atunci când instrucțiunile sale sunt executate de microprocesor. De asemenea, un proces este pregătit de execuție atunci când este lăsat în așteptare deoarece un alt proces este în execuție în momentul respectiv. În sfârșit, un proces poate fi blocat (deoarece își suspendă singur execuția sau efectuează o operație mare consumatoare de timp). A se vedea și **fir de execuție**, **multitasking** și **program**.

proces-copil (*child process*). Proces inițiat de un alt proces denumit *părinte* (*parent process*), printr-o acțiune denumită *bifurcare* (*fork*). Procesul-copil moștenește cele mai multe atribute ale procesului-părinte. În UNIX, procesul-copil este creat ca o copie a procesului-părinte. Dacă un proces nu are părinte, în mod obișnuit el este creat direct de nucleul (*kernel*) sistemului de operare. Adesea, după inițiere, procesul-părinte devine inactiv (este suspendat), rămânând așa până la terminarea execuției procesului-copil. A se vedea și **proces**.

procesor de interogare (*query processor*). Parte a unui motor de căutare, caută în indexul creat de

indexer și găsește contextele în care apare expresia de căutare introdusă de utilizator, trimițând informațiile obținute către un server de document; acesta formatează informația astfel încât browserul să o poată înțelege. Rezultatul obținut este trimis browserului. A se vedea și **indexer** și **motor de căutare**.

procesor de text (*word processor*). Aplicație utilizată pentru crearea, editarea și formatarea unui material (document) destinat adesea tipăririi. Liderul comercial în această categorie de aplicații este *Word*, inclus în suita *Office* dezvoltată de compania Microsoft. Dintre procesoarele de text *open source*, cel mai cunoscut este *Writer*, având versiuni pentru numeroase platforme (e.g., Windows, Mac OS X, Linux, FreeBSD, Solaris). Procesorul este inclus în pachetul *OpenOffice.org*, creat de Sun Microsystems, Inc.

procesor dublu (*dual processor*). În arhitectura calculatoarelor, termenul se referă la un calculator cu două CPU-uri distincte sau la două microprocesoare aflate în interiorul unui singur CPU (dual-core CPU). A se vedea și **CPU**.

program (*program*). Șir de declarații și instrucțiuni care pot fi înțelese și executate de calculator. În esență, un program este exprimarea unui algoritm într-un limbaj de programare. Termenul se referă atât la codul-sursă (scris într-un limbaj de programare), cât și la codul executabil (în limbaj-mașină) obținut prin compilarea codului-sursă. A se

vedea și **cod-mașină**, **cod-sursă** și **compilator**.

programare (*programming*). Procesul de scriere, testare, depanare și întreținere a codului-sursă al unui program de calculator. În cadrul ingineriei software, este una dintre fazele procesului de dezvoltare a software-ului (*i.e.*, implementarea). A se vedea și **inginerie software** și **program**.

programare condusă de evenimente (*event-driven programming*). Denumită și *programare bazată pe evenimente* (*event-based programming*), este o paradigmă de programare în care fluxul unui program este determinat de evenimente (*e.g.*, apăsarea tastelor, mișcarea mouse-ului) sau mesaje provenite de la alte programe. Astfel de programe sunt tipice pentru platformele cu interfață grafică. A se vedea și **program**.

programare declarativă (*declarative programming*). Paradigmă de programare care, referitor la calcul, exprimă logica acestuia fără a descrie controlul fluxului. În acest context, programatorul trebuie să găsească o formulare exactă a problemei, și nu algoritmul ei de rezolvare. Într-un limbaj declarativ, problema este specificată prin descrierea universului problemei și a relațiilor sau funcțiilor existente între obiectele din acest univers. Este opusă paradigmei imperative, care necesită furnizarea

unui algoritm. Iată câteva exemple de limbaje de programare care respectă această paradigmă: GPSS, SYMUB, DINAMO, APL, LISP, Scheme. Are ca subparadigme programarea funcțională și programarea logică. A se vedea și **paradigmă de programare, programare funcțională** și **programare logică**.

programare dinamică (DP, *dynamic programming*). Tehnică de rezolvare a problemelor de optimizare care, ca și *greedy*, conduce de cele mai multe ori la un timp de calcul polinomial. Deși, în principiu, tehnica DP poate fi utilizată pentru rezolvarea oricărei probleme de optimizare, aceasta oferă rezultate foarte bune numai pentru anumite clase de probleme (*e.g.*, probleme care modelează evoluția în timp a unui proces sau probleme de alocare). Creatorul tehnicii DP este Richard Bellman, lucrarea sa fundamentală fiind *Dynamic programming*.

Ca și *divide et impera*, DP rezolvă problemele prin combinarea soluțiilor subproblemelor. Algoritmii de tip *divide et impera* partiționează problemele în subprobleme independente, rezolvă subproblemele în mod recursiv și apoi combină soluțiile găsite pentru a rezolva problema inițială. Dacă subproblemele conțin sub-subprobleme comune, în locul metodei *divide et impera* este mai avantajos să se aplice tehnica programării dinamice. A se vedea și **divide et impera** și **greedy**.

programare funcțională (*functional programming*). Subparadigmă declarativă care tratează calculul ca evaluare a unor funcții și evită schimbarea stării programului și a datelor. Modelul matematic al programării funcționale este calculul lambda. Un limbaj de programare care respectă această paradigmă (cel mai reprezentativ fiind LISP) folosește funcții elementare pe baza cărora programatorul construiește funcții complexe necesare pentru rezolvarea unei anumite probleme. Față de modelul procedural, programarea funcțională prezintă avantajul că încurajează abordarea modulară a programelor, văzute ca funcții care trebuie construite din alte funcții. Astfel, programele nu sunt create de la zero, ci pe baza unor funcții definite anterior în cadrul limbajului de programare. Această tehnică este general acceptată astăzi ca fiind cea mai eficientă metodă pentru dezvoltarea unor produse software complexe.

Limbajele de programare funcțională, mai ales cele pur funcționale, sunt utilizate mai ales în mediile academice, fiind rar folosite în dezvoltarea de software comercial. Cu toate acestea, există limbaje de programare funcționale utilizate și în aplicații comerciale (e.g., Haskell, Scheme) sau specifice unor domenii (e.g., Mathematica). A se vedea și **calculul lambda**, **funcție** și **programare declarativă**.

programare imperativă (*imperative programming*). Paradigmă de programare care descrie calculul ca o suită de

instrucțiuni care modifică starea unui program. Definește procesul de prelucrare ca o succesiune de comenzi (instrucțiuni) care, atunci când sunt urmate, conduc la rezultatul dorit. Rezolvarea unei probleme date presupune identificarea metodei (i.e., a algoritmului) de rezolvare. Programarea procedurală ilustrează cel mai bine această paradigmă, cei doi termeni fiind utilizați adesea ca sinonime. A se vedea și **paradigmă de programare**.

programare logică (*logic programming*). Subparadigmă declarativă bazată pe logica matematică. Într-un limbaj care respectă paradigma logică (e.g., PROLOG), programatorul definește obiecte și relații între acestea. A se vedea și **programare declarativă** și **PROLOG**.

programare orientată pe obiect (POO, *object-oriented programming*). Paradigmă de programare care permite dezvoltarea unor aplicații mai bine structurate și mai ușor de depanat, în comparație cu cele bazate pe abordarea clasică, cunoscută sub numele de programare procedurală. Metoda – care își are rădăcinile în anii 1960, în special în limbajul de programare Simula – a început să fie utilizată la scară largă pentru dezvoltarea aplicațiilor software în anii 1990. Cele mai multe limbaje de programare moderne suportă POO.

Cei mai utilizați termeni în POO sunt cei de *clasă* și *obiect*. Obiectul este o instanță a unei clase. Crearea unui obiect pe baza unei clase definite anterior sau predefinite se realizează

prin intermediul operației de instanțiere. Clasa (extensie a conceptului de tip de date) reprezintă un șablon care include două categorii de componente: date membre (cunoscute și ca proprietăți, attribute sau variabile membre) și funcții membre (denumite și *metode*). În principiu, funcțiile membre efectuează operații asupra datelor membre. Utilizarea metodei POO, adică a claselor (ca tip de date) și obiectelor (ca date), are numeroase avantaje. Iată câteva dintre acestea: încapsularea (componentele obiectului – proprietăți și metode – sunt grupate într-o singură entitate, bine organizată și delimitată de celelalte date; reutilizarea codului (clasele create și utilizate într-o aplicație pot fi folosite în orice altă aplicație, atunci când este necesar); creșterea abstracției (utilizarea obiectelor face posibilă ignorarea temporară a detaliilor legate de implementarea acestora, ceea ce îi permite programatorului să se ocupe numai de structura logică a aplicației pe care o dezvoltă); protejarea accesului (este posibil să se restricționeze accesul la anumite proprietăți și metode ale obiectului, folosindu-se cuvintele-cheie sau etichetele *public*, *protected* și *private*); ascunderea informației (programul poate trata oricare obiect ca pe un element cu conținut practic necunoscut, similar unei „cutii negre”, fiind necesar să se cunoască detalii numai despre metodele publice – cunoscute sub numele de *funcții de interfață* – care permit „comunicarea” cu acesta); utilizarea moștenirii (pot fi definite noi clase prin derivarea din clase

existente, numite *de bază*; clasele noi, numite *derivate*, moștenesc proprietățile și metodele definite în clasele de bază).

Proprietățile private (*private*) vor putea fi accesate numai de metodele definite în aceeași clasă în care au fost declarate. Proprietățile protejate (*protected*) vor putea fi accesate de metodele definite în clasa în care au fost declarate, dar și în clasele derivate din aceasta. În schimb, proprietățile publice (*public*) vor putea fi accesate în orice punct al programului. Metodele private vor putea fi utilizate numai în clasa în care au fost definite, cele protejate în clasa respectivă, dar și în cele derivate, iar metodele publice în orice punct al programului. A se vedea și **paradigmă de programare**.

programator (*programmer*). Persoană care scrie programe pentru calculator. A se vedea și **program**.

program executabil (*executable program*). Numit și *program binar* (*binary program*), este un program care poate fi rulat (*i.e.*, încărcat în memoria calculatorului și executat de microprocesorul acestuia). Termenul este utilizat în mod obișnuit pentru a desemna rezultatul obținut prin compilarea unui program-sursă (scris într-un limbaj de nivel înalt). Un program executabil este stocat într-un fișier executabil. În cazul limbajelor interpretate, programul executabil poate fi și codul-sursă în formatul corespunzător. A se vedea și **limbaj de nivel înalt și program**.

proiectul GNU (*The GNU Project*). Acronim pentru GNU's Not UNIX, este un proiect având ca obiectiv dezvoltarea software-ului liber, inițiat la MIT în 27 septembrie 1983, de Richard Stallman. Primul scop al proiectului GNU a fost dezvoltarea unui sistem de operare denumit GNU. Software-ul elaborat în cadrul proiectului GNU este distribuit sub o licență proprie (GNU GPL). A se vedea și **Licența Publică Generală GNU** și **Stallman, Richard**.

PROLOG (*Programming in logic*). Limbaj de programare logică. Deși a fost conceput inițial în 1972 de Alain Colmerauer și Philippe Roussel pentru dezvoltarea unui interpretor de limbaj natural, s-a impus ca o soluție practică de construire a unui demonstrator automat de teoreme folosind metoda rezoluției. PROLOG a devenit cu timpul un limbaj de uz general, fiind o unealtă importantă în aplicațiile de inteligență artificială. Pentru multe probleme, un program PROLOG are de circa zece ori mai puține rânduri decât echivalentul lui în limbajul Pascal. Astăzi există mai multe implementări PROLOG, atât *open source* (e.g., swi-prolog), cât și proprietare. A se vedea și **Colmerauer, Alain** și **programare logică**.

PROM (*Programmable read-only memory*). Memorie similară cu ROM. Spre deosebire de aceasta, procesul de scriere este electric și poate fi realizat de furnizor sau de utilizator (după încheierea procesului de fabricație), cu

ajutorul unui echipament specializat. A se vedea și **ROM**.

proprietar de date (*data owner*). Autoritate, persoană sau organizație care are în responsabilitatea sa unele date, această responsabilitate fiind conferită prin statut/lege, ordin sau directivă.

protecția datelor (*data security*). Protejarea datelor împotriva modificării, distrugerii sau divulgării, aceste operații/ activități putând fi accidentale sau intenționate.

protecție la copiere (*copy protection*). Tehnologie utilizată pentru prevenirea copierii neautorizate a software-ului, muzicii, filmelor și a altor produse media. Este utilizată în realizarea DVD-urilor, CD-urilor audio, discurilor cu software și jocurilor video etc.

protocol de comunicații (*communications protocol*). În telecomunicații, reprezintă un set de reguli (incluse în standarde) utilizate în scopul reprezentării datelor, semnalizării (i.e., controlului comunicației), autentificării și detecției erorilor, operații necesare atunci când se realizează trimiterea informațiilor printr-un canal de comunicație. Cele mai recente protocoale sunt atribuite de IEEE pentru comunicații în Internet. A se vedea și **IEEE**.

protocol de rețea (*network protocol*). Set de reguli utilizate de calculatoare pentru a comunica într-o rețea. Un astfel de protocol poate fi implementat ca hardware, ca software sau ca o

combinație între acestea. A se vedea și **rețea de calculatoare**.

protocol rutabil (*routable protocol*). Protocol de comunicații utilizat pentru rutarea datelor într-o rețea de calculatoare. În acest scop sunt utilizate o adresă de rețea și o adresă de dispozitiv. Un exemplu de protocol rutabil este suita TCP/IP. A se vedea și **TCP/IP**.

prototip (*prototype*). În programare, reprezintă declarația unei funcții. Aceasta omite corpul funcției, dar specifică numele său, tipurile și numele argumentelor și tipul valorii returnate. Declarația funcției poate fi înlocuită cu definiția acesteia. Iată un exemplu în care se declară o funcție (în C++):

```
int produs (int n, int m)
```

În exemplul prezentat, numele funcției este `produs`; aceasta are două argumente (`n` și `m`) de tip `int` (*integer*). Valoarea returnată este, de asemenea, un număr întreg. A se vedea și **funcție**.

Prototype. Bibliotecă JavaScript care facilitează dezvoltarea aplicațiilor Web dinamice, scris de Sam Stephenson (<http://prototype.conio.net/>). O porțiune a bibliotecii acoperă funcționalitățile oferite de AJAX. Pentru a o utiliza, la pagina Web trebuie legat fișierul *prototype.js*. A se vedea și **AJAX** și **JavaScript**.

pseudocod (*pseudocode*). Limbaj utilizat pentru descrierea algoritmilor. Are o sintaxă și o semantică asemănătoare limbajelor de programare moderne și o anumită flexibilitate în ceea ce privește sintaxa, cu scopul ca

operația de codificare a unui algoritm într-un limbaj de programare să fie cât mai comodă. În mod obișnuit, în pseudocod sunt omise detaliile neesențiale pentru înțelegerea algoritmului.

Un algoritm reprezentat în pseudocod este constituit dintr-o secțiune în care se declară variabilele și tipul de date asociat acestora și se definesc procedurile sau funcțiile apelate de algoritm și din corpul algoritmului, acesta constând într-o succesiune finită de instrucțiuni executabile, astfel:

```
algoritm <nume_algoritm>  
  <declarare_variabile>  
  { tipul variabilelor }  
  <declarare_proceduri>  
  { definire_proceduri/  
    functii }  
begin  
  <proces_de_calcul>  
  { instructiuni }  
end
```

După scrierea unui program în pseudocod, este necesară transpunerea acestuia în limbajul de programare utilizat. A se vedea și **algoritm** și **Rodin**.

PSPad. Editor gratuit de cod-sursă, scris în Delphi și produs începând cu anul 2001 (<http://www.pspad.com>). PSPad suportă evidențierea sintaxei pentru un număr de peste 120 de tipuri de fișiere, e.g., CSS, HTML, XHTML, Java, JavaScript, Perl, PHP, Python, RSS, SQL. Încorporează diverse tipuri de instrumente, dintre care pot fi menționate un editor hexazecimal și un client FTP. A se vedea și **editor de cod-sursă**.

punct de acces fără fir (WAP, *wireless access point*). În rețelele de calculatoare, dispozitiv (e.g., router) care permite dispozitivelor de comunicație fără fir să se conecteze la o rețea *wireless*, utilizând diverse standarde (e.g., *Wi-Fi*, *Bluetooth*). În mod obișnuit, un WAP se conectează la o rețea cu fir. A se vedea și **LAN fără fir** și **router**.

punte (*bridge*). Dispozitiv care conectează multiple segmente de rețea, operând la nivelul 2 (legătură de date) în modelul de referință ISO/OSI. Puntea sunt similare cu *hub*-urile și repetoarele, dispozitive care conectează segmente de rețea la nivelul 1 (fizic) în modelul ISO/OSI. A se vedea și **hub** și **repetor**.

push (*a scrie în stivă*). A adăuga un element în stivă. A se vedea și **pop** și **stivă**.

put (*a pune*). În programare, a scrie date într-un fișier. A se vedea și **fișier**.

Python. Limbaj de programare de nivel înalt, cu o putere remarcabilă și o sintaxă foarte clară, creat de programatorul olandez Guido van Rossum și dezvoltat de Python Software Foundation sub o licență proprie (<http://www.python.org/>). Python suportă multiple paradigme de programare și managementul automat al memoriei. Este utilizat adesea ca limbaj de scripting. Extensiile uzuale de fișier folosite în acest limbaj sunt *.py*, *.pyw* și *.pyo*. A se vedea și **Django** și **van Rossum, Guido**.

Qualcomm. Companie americană de cercetare și dezvoltare în domeniul telecomunicațiilor *wireless* (<http://www.qualcomm.com/>), fondată în 1985 de Irvin Jacobs, Andrew Viterbi și alții. Până în 2006 a produs clientul de *e-mail Eudora*.

Qualcomm a inventat standardele CDMA pentru comunicații în rețelele mobile. De asemenea, a participat la dezvoltarea sistemului de satelit *Globalstar*, împreună cu Loral Space & Communications. Acest sistem folosește o constelație de 44 de sateliți activi care gravitează pe o orbită terestră joasă (LEO, *low-earth orbit*). Sistemul este utilizat pentru servicii de telefonie prin satelit și pentru transfer de date prin satelit folosind modemuri mobile. A se vedea și **Eudora** și **telefon prin satelit**.

QBE (*Query by example, interogare prin exemplu*). Limbaj de interogare pentru baze de date relaționale, conceput de Moshé M. Zloof, de la IBM Research, la mijlocul anilor 1970, în paralel cu dezvoltarea SQL. QBE este primul limbaj grafic de interogare prin care utilizatorul poate specifica câmpurile care vor fi afișate, legăturile dintre tabele și criteriile de regăsire a datelor. Multe *frontend*-uri grafice pentru bazele de date utilizează astăzi idei care provin din QBE. A se vedea și **bază de date relațională**.

Quanta Plus. IDE pentru platforma Linux, stabil, foarte bogat în facilități, eficient și extensibil (<http://quanta.kdewebdev.org/>).

QuarkXPress. Aplicație pentru DTP (*desktop publishing*), unul dintre liderii mondiali în domeniu, creată de Quark, Inc. (<http://www.quark.com/>). Varianta pentru calculatoarele MacIntosh a fost lansată în 1987, iar cea pentru PC în 1992. Are drept competitor aplicația *Adobe InDesign*. A se vedea și **Adobe InDesign** și **DTP**.

query (*interogare*). A se vedea **interogare**.

quicksort (*sortare rapidă*). A se vedea **sortare rapidă**.

QuickTime Format. Format proprietar de tip container multimedia, dezvoltat de Apple Inc., având extensia de fișier *.mov*. Un fișier cu acest format include una sau mai multe piste (*tracks*), fiecare stocând un tip particular de date (audio, video, efecte, text pentru subtitrare).

QuickTime Player. Player multimedia dezvoltat de Apple, Inc. pentru platformele Mac OS X și Windows și distribuit gratuit. A se vedea și **QuickTime Format**.

QuirksMode. Site în care este specificat gradul de implementare a standardelor Web (e.g., CSS, DOM) în browserele *Internet Explorer*, *Firefox*, *Safari*, *Chrome*, *Opera* și *Konqueror* (<http://www.quirksmode.org/>). Site-ul este întreținut de Peter-Paul Koch, programator olandez, specialist în HTML, CSS și JavaScript. A se vedea și **browser Web**.

quiz. Set de întrebări utilizat de obicei în domeniul educațional, prezentat de multe ori *online* sub forma unui joc interactiv simplu.

R

RAD (*Rapid Application Development, dezvoltarea rapidă a aplicațiilor*). Metodă de dezvoltare a aplicațiilor care utilizează o planificare minimală în favoarea creării rapide a unor prototipuri, lansată de James Martin în 1991. În cadrul RAD, nu se așteaptă finalizarea întregului proiect pentru implementare, aplicația fiind implementată pe segmente. Lipsa unei planificări extensive permite, în general, ca software-ul să fie scris mult mai rapid și să fie schimbat mai ușor la cerința clienților. Un exemplu de produs în care este implementată metoda RAD este IDE-ul C++Builder, creat de Borland. A se vedea și **C++Builder**.

radiație electromagnetică (ER, *electromagnetic radiation*). Numit și *undă electromagnetică*, este un câmp electromagnetic care se propagă. Undele radio, lumina și radiațiile X sunt exemple de unde electromagnetice. Undele electromagnetice au aceeași viteză în vid: circa 300 000 km/s (*i.e.*, $3 \cdot 10^8$ m/s). Aceste unde prezintă fenomene specifice: interferența, difracția și polarizarea. ER sunt utilizate în telecomunicații.

RAID (*Redundant Array of Independent Disk, vector redundant de discuri independente*). Arhitectură de discuri multiple utilizată pentru a oferi toleranță la erori și o îmbunătățire a timpilor de acces la date. Termenul de RAID a fost definit în 1987 de David A. Petterson, Garth A. Gibson și Randy Katz (University of California).

Scopul principal în care este folosită arhitectura RAID este mărirea siguranței datelor (importantă pentru protejarea informațiilor critice utilizate în domeniul afacerilor) și creșterea vitezei de furnizare a acestora. Sunt folosite trei tipuri (sau niveluri) principale de arhitectură RAID: *mirroring* (ogindire; utilizează copierea datelor pe mai multe discuri), *striping* (întrețesere; împărțirea datelor pe mai multe discuri) și *error correction* (corectarea erorilor; sunt folosite discuri de verificare redundante care stochează datele cu scopul de a detecta și corecta eventualele erori).

RAM (*Random Access Memory, memorie cu acces aleator*). Memorie volatilă (datele stocate se pierd dacă alimentarea cu energie electrică este întreruptă) implementată pe cipuri electronice, care poate fi scrisă și citită de microprocesor sau de alt dispozitiv. Memoria RAM poate fi accesată aleator, oferind acces direct la orice locație, în orice ordine. A se vedea și **DRAM** și **SRAM**.

raport (*report*). Modalitate de prezentare a informațiilor bazată pe utilizarea unui software special. Un

raport include text, diagrame și, adesea, elemente grafice. Aplicațiile pentru gestionarea bazelor de date dispun de funcții pentru generarea rapoartelor. A se vedea și **bază de date**.

RARP (*Reverse Address Resolution Protocol, protocolul de rezoluție inversă a adresei*). Protocol la nivelul rețea în modelul de referință TCP/IP, determină adresa IP pe baza adresei fizice, fiind astfel complementar protocolului ARP. Funcționează pe baza unui server RARP, care deține o tabelă de corespondențe între adrese fizice și adrese IP, pentru nodurile client pe care le deservește. A se vedea și **adresă IP, ARP, IP și TCP/IP**.

rasterizare (*rasterization*). Convertirea unei imagini vectoriale într-o imagine echivalentă alcătuită din pixeli. A se vedea și **formate raster și formate vectoriale**.

rată de eșantionare (*sampling rate*). Frecvența de prelevare a eșantioanelor unei mărimi fizice (semnal), măsurată în eșantioane/secundă (*samples/second, S/s*) și notată (în mod obișnuit) cu f_s . Cu cât aceasta este mai mare, cu atât ieșirea digitizată va reda mai fidel semnalul original. Teorema de eșantionare Nyquist-Shannon (1924) afirmă că este posibilă reconstrucția exactă a unui semnal în banda de bază dacă acesta este limitat ca bandă și dacă rata de eșantionare este mai mare decât dublul benzii semnalului (frecvența Nyquist). De exemplu, dacă un semnal are o limită superioară a frecvenței de 200

Hz, cu o rată de eșantionare de cel puțin 400 Hz, semnalul original va putea fi refăcut complet. A se vedea și **eșantionare**.

rată de reîmprospătare (*refresh rate*). Cu referire la placa video, reprezintă frecvența cu care trebuie redesenat ecranul pentru a menține o imagine stabilă, fără pâlpâiri. Valorile uzuale pentru rata de reîmprospătare sunt 60 Hz, 70 Hz, 75 Hz și 86 Hz.

rată de transfer (*transfer rate*). A se vedea **bit rate**.

ray tracing (*trasare de raze*). În grafica pe calculator, metodă utilizată pentru generarea imaginii unui obiect prin trasarea unor raze individuale de lumină care pornesc de la sursa care iluminează obiectul, ținând seama de modul cum sunt afectate aceste raze de diverse fenomene optice (*e.g.*, reflexie, refracție, împrăștiere și absorbție), de poziția observatorului și de aberațiile optice. Tehnica permite obținerea unor imagini de înaltă calitate, cu un grad ridicat de fotorealism, dar necesită un efort de calcul ridicat.

răspuns HTTP (*HTTP response*). Generat de serverul Web (după primirea și interpretarea unei cereri HTTP) și trimis clientului. Este alcătuit dintr-o linie de stare și din anteturile (sau atributele) răspunsului.

Linia de stare include, în ordine, următoarele trei componente: versiunea protocolului HTTP utilizat de server (*e.g.*, HTTP/1.1), un cod de stare (*e.g.*, 200) și un text explicativ asociat

codului de stare (folositor utilizatorilor umani; e.g., Ok, asociat codului de stare 200).

Anteturile răspunsului permit transmiterea unor informații adiționale despre server și despre resursa precizată în cerere prin identificatorul său, informații care nu au putut fi incluse în linia de stare. Iată câteva: Location, Proxy-Authenticate, Server și WWW-Authenticate. A se vedea și cerere HTTP, cod de stare HTTP și HTTP.

RDBMS (*Relational DataBase Management System, sistem de gestiune a bazelor de date relaționale*). A se vedea **SGBDR**.

RDF (*Resource Description Framework, cadru de lucru pentru descrierea resurselor*). Limbaj universal standardizat de Consorțiul Web (W3C), utilizat în Web pentru reprezentarea datelor și procesarea metadatelor (date referitoare la date). Oferă interoperabilitate unor aplicații care fac schimb inteligent de informații. RDF folosește XML pentru reprezentarea sintactică a metadatelor.

RDF are utilizări precum: inspectarea resurselor (e.g., oferă noi funcții motoarelor de căutare); agenți inteligenți (facilitează schimbul și împărtășirea cunoștințelor); descrierea dreptului de proprietate intelectuală asupra paginilor Web; catalogarea datelor (e.g., este folosit pentru descrierea/evaluarea conținutului și relațiilor dintre informații aflate în site-urile Web);

securitatea generală sau personală a datelor (e.g., oferă suport pentru semnături digitale).

Modelul de bază al RDF se construiește cu ajutorul a trei tipuri de obiecte: resurse, proprietăți și declarații. Resursele reprezintă datele descrise de expresiile RDF: pagini Web, fragmente de document (e.g., o imagine, o secvență video), dar și obiecte care nu sunt accesibile prin Internet; resursele sunt specificate folosind URI-uri și, opțional, un identificator de legătură). Proprietățile reprezintă caracteristici, atribute sau relații care descriu resursele. Orice proprietate are asociate o semantică, un set de valori permise, o mulțime de tipuri de resurse pe care le descrie și un set de relații cu alte proprietăți.

O declarație poate fi considerată un triplet subiect-predicat-obiect în care subiectul este resursa (specificată folosind un URI), iar predicatul precizează o relație (dată de un URI) dintre subiect și obiect. Acesta din urmă poate fi o altă resursă (precizată de un URI), un literal (tip primitiv de dată sau șir de caractere, conform specificațiilor XML) sau o colecție de literalii. Un literal poate conține marcaje XML, însă acestea nu vor fi analizate de procesorul RDF. Declarațiile se pot reprezenta astfel: graf orientat (nodurile sunt subiecte sau obiecte, iar arcele semnifică un predicat); marcaje (<subject> HAS <predicate> <object>); RDF/XML. A se vedea și **Web semantic**.

README. Fișier care însoțește o mare parte din aplicațiile software. Include sub formă de text simplu informații folositoare utilizatorului, dar care nu sunt conținute în documentație sau în sistemul Help.

read-only (*numai pentru citire*). Mediu (*e.g.*, fișier, memorie, dispozitiv/mediu de stocare) care poate fi citit, dar nu și modificat (*i.e.*, scris).

read/write (*cu citire/scriere*). Mediu (*e.g.*, fișier, memorie) care poate fi citit și scris.

real (*real*). Tip de dată care reține numere. Spre deosebire de întreg, tipul *real* reține numere cu zecimale (*e.g.*, 5,75; -3,81), în locul virgulei utilizându-se punctul (*e.g.*, 5.75; -3.81). Tipurile reale sunt *float* (ocupă 32 de biți), *double* (ocupă 64 de biți) și *long double* (ocupă 80 de biți). Pentru memorarea numerelor reale se folosește reprezentarea în virgulă mobilă. Numerele reale pot fi specificate utilizând și notația științifică, adică adăugând formatului anterior litera „e” urmată de un întreg, care semnifică o putere a numărului 10 (*e.g.*, numărul 2.5×10^{-2} se poate scrie 0.025, dar și 2.5e-2). A se vedea și **întreg**.

RealAudio. Format audio proprietar, având extensia de fișier *.ra*, dezvoltat de RealNetworks începând cu anul 1995. Este potrivit pentru *streaming* audio, fiind utilizat de numeroase stații radio în Internet. A se vedea și **RealNetworks**, **RealVideo** și **streaming media**.

RealNetworks. Companie americană, dezvoltă instrumente software și protocoale pentru livrarea conținutului media în Internet. Este cunoscută în special pentru formatele *RealAudio* și *RealVideo* și pentru *player*-ul *RealPlayer*. A se vedea și **RealAudio** și **RealVideo**.

realitate virtuală (*virtual reality*). A se vedea **ciberspațiu**.

RealVideo. Format video proprietar, având extensia de fișier *.rv*, dezvoltat de compania RealNetworks începând cu anul 1997 și suportat de numeroase platforme (Windows, Mac OS, Linux). Este utilizat împreună cu formatul *RealAudio*, conținutul audio și video fiind împachetat într-un container *RealMedia* (*.rmi*). Acesta este utilizat ca format pentru *streaming media*. Un astfel de format este potrivit pentru vizionarea programelor TV în Internet. *Player*-ul oficial pentru formatul *RealVideo* este *RealPlayer*, dezvoltat de RealNetworks, cu versiuni pentru numeroase platforme. A se vedea și **RealAudio**, **RealNetworks** și **streaming media**.

recunoașterea optică a caracterelor (OCR, *optical character recognition*). Translatarea mecanică sau electronică a imaginilor cu scris de mână, tipărit sau printat (de obicei scanate) în text editabil. Un program OCR cunoscut este *ABBYY FineReader* (<http://www.abbyy.com/>). A se vedea și **caracter**.

recursivitate (*recursion*). Utilizarea în corpul unei funcții a unei referințe la ea însăși. Permite implementarea unor

funcții care includ o cantitate mică de cod, dar eficiente. Utilizarea eronată a recursivității poate crea diverse probleme (e.g., consumă o cantitate de memorie însemnată din stivă), scăzând viteza și ducând chiar la blocarea programului (apelurile recursive pot conduce la epuizarea stivei, respectiv la mesaje de eroare de tipul „stack overflow”). A se vedea și **funcție recursivă**.

Recycle Bin. În sistemele de operare Windows, dosar reprezentat pe *desktop* prin intermediul unei pictograme care simbolizează un coș de deșeuri. Conține fișierele șterse de utilizator. Fiecare dintre acestea poate fi recuperat (i.e., readus în locația ocupată înainte de ștergere) folosind opțiunea *Restore* din meniul contextual, după deschiderea dosarului și selectarea fișierului. Recuperarea simultană a tuturor fișierelor conținute de *Recycle Bin* poate fi realizată folosind opțiunea *Restore All Items*.

redirectarea URL-urilor (*URL redirection*). Denumită și *URL forwarding*, este o tehnică utilizată pentru a face o pagină Web accesibilă, folosind mai multe URL-uri. Tehnica este utilizată din mai multe motive: mutarea unui site pe un domeniu nou, utilizarea unor *alias*-uri scurte pentru URL-uri lungi, schimbarea anterioară a URL-urilor unor pagini Web, manipularea vizitatorilor (e.g., în atacurile de tip *phishing*). În cazul unei redirectări, mesajul HTTP de răspuns al serverului Web include un cod de stare care începe

cu cifra 3, acesta determinând browserul să adreseze cererea către o nouă locație. A se vedea și **pagină Web** și **URL**.

redo (*a reface*). A relua ultima operațiune anulată (*undo*) în programele de aplicație. A se vedea și **undo**.

referer. URL-ul paginii care include legătura folosită de utilizator pentru a ajunge la pagina curentă. A se vedea și **Referer** și **URL**.

Referer. Unul dintre anteturile cererii HTTP. Precizează, în beneficiul serverului, URL-ul de la care a fost obținut identificatorul resursei solicitate. A se vedea și **cerere HTTP**, **referer** și **URL**.

regăsirea informațiilor (*information retrieval*). Găsirea, organizarea și afișarea informațiilor căutate în documente, baze de date relaționale și Web. A se vedea și **bază de date relațională** și **Web**.

regedit (*Registry Editor*). Aplicație inclusă în sistemele de operare Windows, care îi permite utilizatorului să editeze intrările din baza de date *Registry*. A se vedea și **Registry**.

registru (*register*). Zonă de memorie de mare viteză, inclusă într-un microprocesor. Regiștrii microprocesorului, folosiți de unitatea de execuție a acestuia pentru diverse operații, se pot clasifica în: regiștri generali (utilizați pentru calcule, operații cu șiruri de biți, operații de intrare-ieșire sau ca regiștri de bază), regiștri de segment (conțin:

adresa segmentului de cod curent unde se află instrucțiunile programului; adresa segmentului de stivă curent; adresa segmentului de date curent unde se află zona de date a programului; adresa segmentului de date externe), registrul pointerului de instrucțiuni (conține adresa următoare a programului-mașină, relativ la începutul segmentului de cod) și registrul indicatorilor de stare și control (conține biți care reflectă starea microprocesorului, precum și informații referitoare la modul de lucru al acestuia). A se vedea și **CPU**.

registru de instrucțiuni (*instruction register*). Unul dintre cei mai importanți regiștri ai unității centrale de prelucrare (CPU). Păstrează instrucțiunea în curs de execuție. A se vedea și **contor de program, CPU și registru**.

Registry. Numită și *System Registry*, este o bază de date ierarhică utilizată în sistemele de operare Windows pentru stocarea informațiilor necesare în scopul configurării sistemului pentru utilizatori, aplicații și dispozitive hardware. A se vedea și **regedit**.

RELOAD (*Reusable eLearning Object Editor – Authoring&Delivery*). Scris în limbajul Java, editorul furnizează suport pentru standardele IMS Metadata, IMS Content Packaging 1.1.4, SCORM 1.2, SCORM 2004 și IMS Learning Design. Folosind editorul RELOAD se pot genera fișiere-manifest în format XML, compatibil cu standardele amintite. Prin utilizarea interfeței

vizuale se pot crea pachete de conținut și metadatele asociate. De asemenea, editorul validează definițiile introduse. A se vedea și **IMS Content Packaging și SCORM**.

repetor (*repeater*). Denumit și *repeater*, este un dispozitiv care preia semnalul de la calculatoare și alte dispozitive din rețea și îl regenerează (amplifică), astfel încât acesta își păstrează integritatea pe o lungime mai mare a mediului de rețea decât este posibil în mod normal.

reproducere (*replication*). În sistemele de gestionare a bazelor de date distribuite, reprezintă procesul de copiere (parțială sau integrală) a unei baze de date în alt loc din rețea, în scopul sincronizării sistemului. Replicarea poate să îmbunătățească performanțele aplicațiilor și să protejeze disponibilitatea acestora, oferind alternative de acces la date.

rescrierea URL-urilor (*URL rewriting*). Tehnică utilizată pentru modificarea URL-urilor (afișate în bara de adrese a browserului), obținându-se astfel URL-uri scurte și relevante pentru paginile Web. În felul acesta fișierele utilizate pentru generarea dinamică a paginilor Web sunt separate de acestea din urmă (prezentate vizitatorilor). Pentru rescrierea URL-urilor se utilizează un software denumit *rewrite engine* (*motor de rescriere*).

Dintre avantajele utilizării acestei tehnici pot fi menționate: crearea unor URL-uri puternic descriptive, ascunderea

detaliilor legate de fișierele folosite pentru generarea dinamică a paginilor Web și de structura bazelor de date folosite, prevenirea utilizării tehnicii *inline linking*.

Iată un exemplu de utilizare a tehnicii de rescriere a URL-urilor:

```
http://www.exemplu.ro/articole.php?an=2009&luna=10&ziua=20
```

este înlocuit cu:

```
http://www.exemplu.ro/articole.php/2009/10/20/
```

Rezultatul obținut îi va oferi utilizatorului o idee mult mai clară despre conținutul paginii. Cadrele de lucru apărute relativ recent oferă posibilitatea utilizării tehnicii de rescriere a URL-urilor (e.g., *Ruby on Rails*, *Django*). A se vedea și **Django**, **inline linking** și **Ruby on Rails**.

REST (*Representational State Transfer*). Termen introdus de Roy Fielding pentru a descrie arhitectura Web, desemnând un model pentru calculul distribuit. Atunci când este aplicat serviciilor Web, modelul implică trei tehnologii: HTTP, URI și XML. În continuare este explicată noțiunea de *serviciu Web REST*.

Ca și alte tipuri de servicii Web, arhitectura REST se bazează pe conceptul de *resursă*. Unui serviciu Web REST îi sunt impuse următoarele două constrângeri: 1) interfețele sale sunt limitate la protocolul HTTP (astfel, în REST, sunt definite următoarele interfețe: a) HTTP GET, utilizată pentru a obține o reprezentare a unei resurse;

un consumator utilizează metoda pentru a regăsi o reprezentare folosind un URI; serviciile furnizate prin această interfață nu presupun nici o obligație din partea consumatorilor; b) HTTP DELETE, utilizată pentru a elimina reprezentările unei resurse; c) HTTP POST, utilizată pentru actualizarea sau crearea reprezentărilor unei resurse; d) HTTP PUT, utilizată pentru crearea reprezentărilor unei resurse); 2) mesajele utilizate folosesc formatul XML, putând fi validate via specificații XML Schema sau RELAX NG.

Serviciile Web REST necesită o infrastructură software minimală, adică una care să implementeze protocolul HTTP și tehnologii de procesare XML, suportate de cele mai multe platforme și limbaje de programare. A se vedea și **serviciu Web**.

(a) **restaura** (*restore*). A copie pe hard-disk a fișierelor salvate anterior pe un dispozitiv de stocare de siguranță, înlocuindu-se astfel fișierele existente, care s-au deteriorat sau au fost șterse din greșeală. A se vedea și **backup**.

resursă (*resource*). 1. Tip de date special utilizat în unele limbaje de programare, asociat rezultatelor returnate de unele funcții. De exemplu, în PHP, în cazul extensiei *mysql* aceste funcții pot realiza operații ca: deschiderea fișierelor, deschiderea conexiunii cu un server de baze de date și interogarea bazelor de date. 2. Parte a unui sistem de calcul (e.g., imprimantă, zonă de memorie) utilizată de un program, proces sau dispozitiv. 3. În

Web, un fișier situat într-o locație cunoscută sub numele de nod și utilizat prin intermediul unui sistem hipertext, folosind un sistem de identificare numit URL. Resursele Web-ului sunt organizate în pagini Web și livrate utilizatorilor de un *server Web*. Utilizatorii folosesc pentru vizualizarea paginilor Web un program client cunoscut sub denumirea de *browser* sau *navigator Web*. A se vedea și **browser Web**, **PHP**, **server Web** și **URL**.

resursă partajată (*shared resource*). Resursă care poate fi utilizată de mai multe programe sau dispozitive. A se vedea și **resursă** (2).

resurse educaționale deschise (OER, *Open Educational Resources*). Materiale educaționale și resurse oferite gratuit și deschis oricui dorește să le folosească, utilizând o serie de licențe care permit remixarea, îmbunătățirea și redistribuirea acestora. Într-o lume în care accesul la cunoaștere este inegal distribuit, potențialul depozitelor libere/deschise de cunoștințe și resurse educaționale deschise de a redresa această balanță este enorm.

Termenul OER a fost adoptat la Forumul UNESCO de la Paris, desfășurat în perioada 1–3 iulie 2002, în cadrul căruia s-a analizat impactul pe care proiectele *OpenCourseWare* îl au asupra învățământului superior în țările în curs de dezvoltare. Forumul UNESCO amintit a adoptat următoarea definiție pentru termenul OER: resursele educaționale deschise se referă la

accesul liber la materialele educaționale, facilitat de tehnologiile comunicaționale și informaționale, pentru consultare, utilizare și adaptare de către comunitatea utilizatorilor, în scopuri necomerciale.

Dezvoltarea resurselor educaționale deschise include trei arii de activitate: 1) crearea de *software open source* și de utilitare în scopul realizării, furnizării și utilizării conținutului deschis folosit în educație; acesta include: instrumente pentru căutarea și organizarea conținutului, sisteme pentru managementul conținutului și al învățării, instrumente pentru dezvoltarea conținutului, comunități *online* de învățare și de practică; 2) crearea de conținut deschis pentru învățare: cursuri complete, materiale pentru cursuri, planuri de lecție și obiecte pentru învățare (acest ultim concept va fi explicat detaliat în articolul următor); 3) dezvoltarea de standarde și licențe care promovează publicarea liberă a materialelor și oferă principii de proiectare a conținutului acestora.

În septembrie 2007 a fost publicată declarația de la Cape Town privind educația deschisă (*Cape Town Open Education Declaration*). A se vedea și **acces deschis**.

retur de car (*carriage return*). Caracter de control care comandă calculatorului sau imprimantei să revină la începutul rândului curent. În setul de caractere ASCII, codul zecimal al acestui caracter este 13. În mod obișnuit, este utilizat împreună cu caracterul „salt la

un rând nou” (*linefeed*), amândouă determinând avansul cursorului sau capului de imprimare la începutul unui rând nou. A se vedea și **salt la rând nou**.

(a) **returna** (*return*). **1.** A transfera controlul sistemului de la rutina (funcția) apelată la programul apelant. **2.** A întoarce rezultatele generate de rutina (funcția) apelată către programul apelant. A se vedea și **funcție și rutină**.

rețea (*network*). Colecție de relații care leagă între ei oameni, *status*-uri sociale sau alte unități de analiză (*e.g.*, grupuri, organizații formale).

rețea de calculatoare (*computer network*). Ansamblu de calculatoare autonome interconectate, capabile să comunice unele cu altele folosind canale de comunicație (medii de transmisie).

Componentele unei rețele se pot conecta în variate moduri și pot avea funcții diferite. Dintre scopurile creării și utilizării rețelilor de calculatoare pot fi menționate: partajarea resurselor (fizice, logice, informaționale); accesul la informații aflate la distanță; comunicare între oameni; divertisment interactiv.

Este de remarcat deosebirea dintre un sistem distribuit și o rețea: într-un sistem distribuit, existența mai multor calculatoare autonome este transparentă pentru utilizator, operațiile efectuate de acesta neimplicând alegerea explicită a unui anumit calculator; în schimb, într-o rețea, utilizatorii se conectează

explicit la un anumit calculator și, la fel, solicită execuția anumitor procese.

Un criteriu de clasificare a rețelilor de calculatoare des utilizat este disponerea spațială. Din acest punct de vedere, se deosebesc: rețele locale (LAN, *Local Area Network*) – rețele private de mică întindere, localizate într-o singură clădire sau într-un campus (CAN, *Campus Area Network*), pe distanțe de până la câțiva kilometri, și recomandate pentru aplicații de business și educaționale; rețele metropolitane (MAN, *Metropolitan Area Network*) – versiuni extinse de LAN-uri (atât private, cât și publice), care se pot întinde pe o zonă de clădiri învecinate sau pe suprafața unui întreg oraș; rețele pentru zone geografice mari (WAN, *Wide Area Network*), care acoperă o arie geografică extinsă adesea la nivelul unei țări sau chiar al unui continent.

Din punctul de vedere al disponerii spațiale, majoritatea rețelilor sunt de tip LAN sau WAN. Un tip special de rețea este Internetul (supranumit *rețeaua globală*), constituit prin interconectarea unui număr foarte mare de rețele și extins la scara întregii planete.

După modul de transmisie a datelor, care constituie un alt criteriu de clasificare, rețelele de calculatoare pot fi: rețele cu difuzare (cu un singur canal de comunicație) și rețele *peer-to-peer*, adică rețele punct la punct (sunt realizate conexiuni multiple între perechi de mașini).

Evoluția rețelisticii a determinat o redefinire a naturii și limitelor unui calculator. În acest sens, pot fi menționate cuvintele atribuite specialiștilor John Gage și Bill Joy, ambii de la *Sun Microsystems*: „the network is the computer” („rețeaua este calculatorul”). A se vedea și **LAN**, **MAN** și **WAN**.

rețea de clasă A (*class A network*). A se vedea **clasă de adrese IP**.

rețea de clasă B (*class B network*). A se vedea **clasă de adrese IP**.

rețea de clasă C (*class C network*). A se vedea **clasă de adrese IP**.

rețea de întreprindere (*enterprise network*). A se vedea **intranet**.

rețea de mare suprafață (**WAN**, *Wide Area Network*). A se vedea **WAN**.

rețea de socializare (*social networking*). A se vedea **rețea socială online**.

rețea galactică (*Galactic Network*). Concept introdus (într-o serie de note) în anul 1962 de J.C.R. Licklider de la MIT. Într-o astfel de rețea, calculatoarele sunt interconectate astfel încât oricine poate avea acces la date și programe din orice loc.

rețea locală (**LAN**, *Local Area Network*). A se vedea **LAN**.

rețea locală fără fir (*wireless LAN*). Rețea locală în care comunicarea între calculatoare și distribuitor se realizează prin intermediul undelor electromagnetice. A se vedea și **LAN** și **wireless**.

rețea metropolitană (**MAN**, *Metropolitan Area Network*). A se vedea **MAN**.

rețea neuronală artificială (**ANN**, *Artificial Neural Network*). Denumită pe scurt **rețea neuronală**, este o mașină proiectată pentru a modela felul în care creierul rezolvă o anumită problemă, fiind implementată cu ajutorul unor componente electronice sau simulată printr-un program. Este alcătuită din elemente de procesare simple denumite noduri sau neuroni artificiali, dar inferioare corespondentului uman, puternic interconectate prin intermediul unor legături numite interconexiuni, prin care se propagă informația numerică și operând în paralel. Simon Haykin (profesor la McMaster University, Hamilton, Ontario, Canada și autor al unor cursuri recunoscute în domeniu) consideră că o rețea neuronală este un procesor masiv paralel, distribuit, care are o tendință naturală de a înmagazina cunoștințe experimentale și de a le face disponibile pentru utilizare. O rețea neuronală se aseamănă cu creierul uman în două privințe: cunoștințele sunt achiziționate de rețea printr-un proces de învățare; cunoștințele sunt depozitate în conexiunile interneuronale (denumite ponderi sinaptice), și nu în elementele de procesare (neuroni).

Caracteristicile cele mai importante ale rețelelor neuronale sunt: capacitatea de a învăța (au un algoritm de învățare care determină ajustarea ponderilor sinaptice pe baza unor modele prezentate), capacitatea de generalizare

(sunt capabile să dea răspunsuri corecte și pentru intrări diferite de cele pentru care au fost antrenate, cu două condiții – diferențele să nu fie prea mari și rețeaua să fie instruită adecvat) și capacitatea de sinteză (pot lua decizii când dispun de informații incomplete sau irelevante).

Rețelele neuronale se pot analiza în funcție de următoarele elemente: modelul utilizat pentru elementul de procesare (modelul neuronului artificial), structura de interconexiuni (arhitectura) și algoritmi de învățare. Cele mai cunoscute tipuri de rețele neuronale sunt: Perceptronul (F. Rosenblatt, 1958), Perceptronul multi-strat (*Multi-Layer Perceptron*, Minsky, S. Papert, 1969), rețeaua cu propagare înapoi (*Backpropagation Net*, G.E. Hinton, E. Rumelhart, R.J. Williams, 1986), rețeaua *Hopfield* și rețeaua *Kohonen* (*Kohonen Feature Map*, Teuvo Kohonen, 1982). A se vedea și **inteligență artificială**.

rețea socială (*social network*). Structură socială alcătuită din noduri (indivizi sau organizații) legate prin unul sau mai multe tipuri particulare de interdependențe (e.g., valori, idei, prietenie, antipatie, asemănare, conflicte). A se vedea și **analiza rețelelor sociale, bogăția rețelelor, forma rețelelor sociale, rețea socială online și software pentru analiza rețelelor sociale**.

rețea socială online (*online social network*). Tip de serviciu bazat pe Web

care le permite utilizatorilor să-și construiască un profil public sau semi-public, să creeze o listă cu alți utilizatori cu care partajează o conexiune și să vadă listele similare ale acestora din urmă. Natura conexiunilor poate varia de la un site la altul, iar rezultatul este o structură de tip graf, adesea foarte complexă. Conform unui studiu comScore, Inc., serviciile de socializare care au luat naștere în America de Nord au început să se extindă treptat și pe alte piețe, devenind un fenomen global. Astfel, popularitatea acestora a crescut în anul 2008 cu 25% .

La sfârșitul anului 2007, Google a intrat în *social networking*, dar nu prin crearea unei rețele sociale noi, ci prin crearea unui *layer* (strat) intermediar între API-urile mai multor rețele sociale (*Orkut*, *Salesforce*, *LinkedIn*, *Ning*, *Hi5* și *Friendster*, pentru început), denumit *Open Social* (<http://code.google.com/apis/opensocial/>). Prin analogie cu domeniul bazelor de date, *OpenSocial* este un *database abstraction layer*, care – folosind un API unic – oferă acces la mai multe tipuri de baze de date (i.e., rețele sociale). A se vedea și **rețea socială**.

rețea virtuală privată (VPN, *Virtual Private Network*). Colecție de noduri într-o rețea publică (e.g., Internet) care comunică unele cu altele utilizând o tehnologie de criptare, ceea ce înseamnă că mesajele schimbate între ele nu pot fi înțelese de utilizatorii neautorizați. Rezultă că această colecție de noduri poate fi privită ca o rețea în care

nodurile ar fi legate prin linii private. A se vedea și **nod**.

rețelistică (*networking*). Disciplină care se ocupă cu comunicația dintre calculatoare sau dintre alte dispozitive. Disciplina are ca scop proiectarea și implementarea rețelelor de comunicații și a dispozitivelor necesare pentru implementarea acestora (*e.g.*, routere), precum și cu elaborarea protocoalelor de comunicație. Rețelistica utilizează adesea rezultatele obținute în telecomunicații, tehnologia informației, informatică și ingineria calculatoarelor. A se vedea și **rețea de calculatoare**.

reutilizare (*reusability*). Utilizarea unei secvențe de cod sau a unui program în alte programe sau aplicații. În particular, posibilitatea de reutilizare este o caracteristică a codului scris folosind paradigma programării orientate pe obiect (POO). A se vedea și **programare orientată pe obiect**.

rezoluție (*resolution*). 1. Finețea detaliilor redată de un dispozitiv de afișare/prelucrare a imaginilor: scanner, imprimantă sau monitor. Se măsoară în puncte pe inch (*dpi*, *dots per inch*). 2. Dimensiunile imaginii afișate de un monitor, date de numărul total de pixeli afișați pe orizontală și pe verticală (*e.g.*, 1440x900). 3. Procesul de traducere a unei adrese de tip nume de domeniu într-o adresă IP și invers. A se vedea și **adresă IP** și **nume de domeniu**.

RFC (*Request for Comments*). Document publicat de IETF în care sunt descrise propuneri de metode, cercetări sau inovații aplicabile sistemelor conectate în Internet. Unele dintre propunerile publicate ca RFC-uri sunt adoptate de IETF ca standarde Internet. Primul RFC (intitulat *Host Software*) a fost scris de Steve Crocker și publicat pe 7 aprilie 1969 (RFC 1). Din 1969 până în 1998, editor al RFC-urilor a fost John Postel. Sursa oficială a acestor documente este *RFC Editor* (<http://www.rfc-editor.org/rfc.html>). A se vedea și **IETF**, **Crocker**, **Steve** și **Postel**, **Jonathan Bruce**.

RGB (*Red Green Blue*). Model aditiv de descriere a culorilor în care acestea sunt obținute prin amestecul în diverse proporții (care variază între 0 și 255) a trei culori primare (*red* – roșu, *green* – verde și *blue* – albastru). Utilizează sensibilitatea la cele trei culori a celulelor receptoare din retina ochiului uman. Modelul este utilizat pentru reprezentarea imaginilor în sistemele electronice de afișaj (*e.g.*, televizoare, monitoare, ecranele telefoanelor mobile, proiectoare video). A se vedea și **CMYK**.

RIA (*Rich Internet Applications*). Aplicații Web ale căror caracteristici sunt similare celor pentru *desktop*. Sunt furnizate în mod obișnuit folosind *plugin*-uri instalate în browser sau independent, prin intermediul unor mașini virtuale sau al unor medii securizate (*sandbox*). Termenul a fost introdus în martie 2002 de *Macromedia*

(cumpărat ulterior de *Adobe*). Iată câteva exemple de cadre de lucru RIA: *Curl*, *Adobe Flash/Flex/AIR*, *Java/JavaFX*, *Microsoft Silverlight*. A se vedea și **sandbox**.

Rico. Bibliotecă JavaScript *open source* dezvoltată sub licență Apache, utilizată pentru crearea aplicațiilor Web (<http://www.openrico.org/>). Asigură suport **AJAX**, *drag-and-drop*, efecte cinematice, precum și diverse funcționalități (e.g., *LiveGrid*). A se vedea și **AJAX** și **JavaScript**.

RISC (*Reduced Instruction Set Computing*, *prelucrarea datelor pe baza unui set redus de instrucțiuni*). Strategie de design și model pentru microprocesor care au ca scop prelucrarea rapidă a unui set alcătuit dintr-un număr relativ mic de instrucțiuni simple, astfel încât, prin optimizarea acestora, să fie executate într-un timp mic (în mod obișnuit, într-o perioadă de ceas). În categoria microprocesoarelor RISC sunt incluse, printre altele, cipurile SPARC (Sun Microsystems), PowerPC (create de IBM, Apple și Motorola), SuperH (Hitachi) și Alpha (DEC). A se vedea și **CISC**.

Ritchie, Dennis (n. 1941). Programator american, unul dintre pionierii informaticii. A dezvoltat limbajul de programare C. Împreună cu Brian Kernighan a scris lucrarea *The C Programming Language*. În 1983 a câștigat premiul Turing, împreună cu Kenneth Thomson. A se vedea și **C**, **Kernighan, Brian** și **Thomson, Kenneth**.

Roberts, Lawrence (n. 1937). Cunoscut ca Larry Roberts, este un informatician american, considerat creatorul Internetului, alături de Lenard Kleinrock, Vinton Cerf și Bob Kahn. A fost șeful proiectului ARPA, în cadrul căruia a fost creată tehnologia comutării de pachete și rețeaua ARPANET, predecesorul Internetului. Pentru contribuțiile sale a primit numeroase distincții (e.g., *IEEE Internet Award*, *Draper Prize*).

robots.txt. Fișier plasat pe serverul Web, cu rolul de a controla indexarea site-ului de către roboții motoarelor de căutare. De exemplu, este posibil ca – folosind fișierul *robots.txt* – unor roboți să li se permită indexarea site-ului, în vreme ce altora nu. De asemenea, este posibil să se interzică indexarea anumitor pagini (tuturor roboților sau numai unora) sau indexarea conținutului grafic.

Conform standardului pentru excluderea roboților, orice robot, înainte de a începe să transfere fișiere de pe un server, va accesa în primul rând fișierul *robots.txt*. Acesta trebuie stocat în directorul-rădăcină al serverului și să precizeze care zone (directoare sau fișiere) vor fi evitate în cursul parcurgerii automate a acestuia. Dacă este găsit acest fișier, el trebuie interpretat, iar robotul trebuie să acționeze conform indicațiilor și restricțiilor pe care le conține. A se vedea și **SRE**.

Rodin. Limbaj de programare imperativ, nou implementat (2009), în stil C, destinat rulării programelor în pseudocod scrise în limba română. A se vedea și **algoritm** și **pseudocod**.

ROM (*read-only memory, memorie care poate fi numai citită*). Memorie nevolatilă implementată pe cipuri electronice, care poate fi numai citită. Datele stocate în memoriile ROM sunt scrise din fabricație și nu pot fi modificate niciodată. A se vedea și **PROM**, **EPROM** și **EEPROM**.

rootkit (*rootkit*). Colecție de programe utilizate de un *hacker* pentru a evita detectarea sa atunci când încearcă să acceseze în mod neautorizat un calculator. Odată instalate într-un sistem de calcul, aceste programe sunt invizibile pentru utilizatori și pot evita să fie detectate de programele de securitate. Deoarece mulți utilizatori își accesează calculatorul folosind un cont de administrator în loc să creeze un cont diferit, cu acces restricționat, facilitează instalarea *rootkit*-urilor de către *hackeri*. A se vedea și **hacker**.

ROSI (*Romanian Open Source and Free Software Initiative*). Grup informal inițiat de Lucian Savluc, informatician român stabilit în Canada, în scopul promovării FLOSS în România. ROSI și Agora Media organizează anual în țară conferința eLiberatica (<http://www.eliberatica.ro>). A se vedea și **FLOSS**.

RoTLD (*Romania Top Level Domain*). Agenție a ICI (Institutul Național de

Cercetare-Dezvoltare în Informatică, <http://www.ici.ro/>), având ca obiect de activitate înregistrarea unui domeniu .ro (<http://www.rotld.ro/>). Operația respectivă se poate realiza și la unul dintre partenerii autorizați. A se vedea și **domeniu**.

round robin (*alocare prin rotație*). Unul dintre cei mai simpli algoritmi utilizați pentru planificarea proceselor într-un sistem de operare. Algoritmul alocă ciclic, secvențial, durate de timp egale fiecărui proces, fără a acorda prioritate vreunui dintre ele. A se vedea și **algoritm de planificare** și **sistem de operare**.

RSS. Format dezvoltat inițial de firma Netscape (în martie 1999) ca RSS 0.90. Astăzi este numele unei familii de formate XML utilizate de editorii Web în scopul distribuirii știrilor, notărilor sau sumarurilor. Activitatea este cunoscută sub numele de *Web syndication* (punerea la dispoziție a informațiilor actualizabile din Web). Prin utilizarea RSS, informațiile de pe Web devin mult mai ușor de găsit, iar programatorii le pot distribui mult mai ușor grupurilor de utilizatori interesați. Deoarece datele RSS sunt de mici dimensiuni, ele se încarcă rapid. Din acest motiv, acestea pot fi utilizate inclusiv în dispozitive ca telefoane mobile și PDA-uri. În particular, RSS este folosit de site-urile de știri, site-urile unor companii care vând diverse produse, site-urile de tip weblog etc.

RSS este deosebit de important, deoarece permite informarea utilizatorilor

despre actualizări ale resurselor *online* fără a utiliza serviciul de poștă electronică. Se reduce astfel posibilitatea răspândirii virușilor prin intermediul acestui serviciu. Din multe puncte de vedere, RSS este similar serviciului *Newsletter*, pe care multe site-uri îl oferă prin intermediul poștei electronice în scopul informării utilizatorilor despre actualizările efectuate pe aceste site-uri.

Acronimul RSS se poate referi la: *Rich Site Summary* (RSS 0.9x; RSS 0.90 nu se mai utilizează; RSS 0.91 permite o flexibilitate scăzută; versiunile 0.92, 0.93 și 0.94 permit o flexibilitate ridicată; este indicat ca în locul versiunilor 0.91, 0.92, 0.93 și 0.94 să se utilizeze RSS 2.0); *RDF Site Summary* (RSS 1.0, bazat pe RDF, este extensibil prin module și utilizat pentru aplicațiile bazate pe RDF sau care necesită module RDF specifice); *Really Simple Syndication* (RSS 2.0; versiunea este utilizată în scopuri generale, pentru punerea la dispoziție a unor tipuri diverse de informații actualizabile). A se vedea și **feed** și **RDF**.

Ruby. Limbaj de scripting interpretat, integral orientat pe obiecte, având o sintaxă simplă (parțial inspirată din Eiffel și Ada), care îl face ușor de utilizat. A fost creat în Japonia la mijlocul anilor 1990, de Yukihiro Matsumoto, dar a devenit popular printre programatorii Web numai odată cu apariția documentației în limba engleză. Suportă multiple paradigme de programare și managementul automat al

memoriei. Este oferit sub dublă licență, Ruby License și GNU General Public License.

Ruby combină diverse caracteristici ale altor limbaje de programare, dintre care pot fi menționate: eleganța conceptuală (Smalltalk), ușurința în învățare (Python) și pragmatismul (Perl). Este potrivit pentru procesarea fișierelor text, precum și pentru managementul proceselor (ca și Perl). Iată câteva caracteristici ale limbajului: manipularea erorilor este realizată într-un mod similar celui din Java și Python; utilizează conceptul de *modul*, acesta reprezentând o colecție de metode. Fiecare clasă Ruby poate importa un modul, obținând pentru folosire toate modulele acestuia; nu este necesară declararea variabilelor, utilizându-se numai o convenție de denumire care denotă scopul în care acestea sunt folosite; poate încărca dinamic module de extensie, dacă sistemul de operare permite acest lucru; poate fi portat pe un număr mare de platforme (*e.g.*, UNIX/Linux, Windows, MacOS, BeOS, OS/2). A se vedea și **Ruby on Rails**.

Ruby on Rails. Denumit adesea *Rails* sau *RoR*, este un cadru de lucru *open source* utilizat în scopul dezvoltării aplicațiilor Web folosind limbajul de programare Ruby (<http://rubyonrails.org/>). A fost creat de David Heinemeier Hansson și este dezvoltat de *Rails Core Team* sub licență MIT.

Cel mai important element adus de *Ruby on Rails* este creșterea impresionantă a vitezei de lucru, obținută fără sacrificii în ceea ce privește calitatea aplicațiilor rezultate. Acest lucru este posibil – în primul rând – datorită limbajului Ruby, în care multe lucruri sunt simplu de făcut în raport cu alte limbaje. În plus, funcționează două principii importante care au ghidat realizarea acestui cadru de lucru: *less software* (mai puțin software) și *convention over configuration* (convenție dincolo de configurație). Primul principiu presupune posibilitatea de a scrie aplicații complexe utilizând numai câteva rânduri de cod, fapt care determină creșterea importantă a vitezei de programare și face codul ușor de înțeles, de depanat și de dezvoltat. Al doilea principiu presupune utilizarea, în locul unor fișiere de configurare complexe (e.g., XML), a câteva convenții de notare.

Pe baza *Ruby on Rails*, aplicațiile Web sunt create în acord cu modelul MVC (*Model-View-Control*). Vederea (numită și *prezentare*) constă în șabloane care sunt responsabile, în primul rând, pentru inserția datelor între marcasele HTML. Modelul conține obiecte „inteligente” (e.g., *Account*, *Person*, *Product*, *Post*) care „țin” toată logica aplicației. Controlerul administrează cererile primite (e.g., `Save New Account`, `Update Product`, `Show Post`), utilizând modelul și direcționând datele către prezentare. Modelul este administrat de *Active Record*, care permite prezentarea

datelor extrase din bazele de date ca obiecte. Controlerul și prezentarea sunt administrate de *Action Pack*, care comandă ambele straturi utilizând cele două componente ale sale: *Action Controller* și *Action View*.

Ruby on Rails utilizează implicit un server Web inclus denumit *WEBrick*, nefiind necesară instalarea altuia. *Ruby on Rails* poate fi rulat și pe alte servere Web, de exemplu *Apache*. În acest caz – pentru creșterea performanțelor – este necesar să se instaleze FastCGI (`mod_fastcgi` pentru *Apache 1.3* și `mod_fcgi` pentru *Apache 2.0+*). Pentru producție, este recomandat serverul *lighttpd*, considerabil mai rapid decât *WEBrick*, dar care necesită instalări adiționale și, curent, este utilizat numai pe sistemele de operare UNIX (utilizatorii Windows sunt încurajați să folosească *WEBrick*). Serverul Web *Mongrel* (server bazat pe Ruby, dar cu o componentă C) lucrează foarte bine pe platforma Windows.

Ruby on Rails are un model simplu pentru implementarea tehnicii AJAX. Odată ce browserul a interpretat și a afișat pagina Web, diferitele acțiuni ale utilizatorului pot determina afișarea unei noi pagini (ca în aplicațiile Web clasice) sau pot declanșa interacțiuni bazate pe AJAX. A se vedea și **Ruby**.

runtime (*timpul execuției*). **1.** Termenul, scris și *run-time* sau *run time*, se referă la timpul în care un program este în curs de execuție. În acest timp au loc diverse evenimente, precum evaluarea variabilelor și alocarea dinamică a

memoriei. **2.** Durata necesară execuției unui program.

runtime error (*eroare produsă în timpul execuției*). Eroare software produsă în timp ce un program este în curs de execuție. O eroare de acest fel apare atunci când, de exemplu, se încearcă împărțirea la zero (*division by zero*). A se vedea și **runtime**.

router (*router*). Dispozitiv hardware sau software care operează la nivelul 3 (rețea) al modelului de referință ISO/OSI. Se ocupă de livrarea pachetelor pe care, folosind adresele lor IP (de rețea), le trimite către destinație

pe traseul cel mai eficient. În general, routerele conțin un sistem de operare specializat (*e.g.*, IOS de la Cisco), RAM, memorie *flash* și cel puțin două interfețe de rețea. A se vedea și **ISO/OSI**.

rutină (*routine*). Secvență de cod care poate fi invocată și executată în cadrul unui program. În mod obișnuit, unei rutine îi este atribuit un nume, acesta fiind utilizat pentru invocarea (apelarea) sa. Termenul este înrudit cu alți termeni precum funcție, procedură și subrutină. A se vedea și **funcție**, **procedură** și **subrutină**.

S-S

Safari. Browser Web dezvoltat de Apple, Inc. începând cu anul 2003. Este navigatorul implicit utilizat pe platforma Mac OS. A fost dezvoltată o versiune *Safari* și pentru iPhone OS, sistem de operare pentru două dintre dispozitivele mobile produse și comercializate de Apple (iPhone și iPod Touch). De asemenea, în 2007 a fost lansată o versiune *Safari* pentru platforma Microsoft Windows, atât pentru Windows XP, cât și pentru Windows Vista. A se vedea și **Apple, Inc.**

safe mode (*modul de siguranță*). Mod de încărcare a sistemului de operare Windows, utilizat atunci când există probleme care determină funcționarea anormală a acestuia. În *safe mode*, cele mai multe drivere și periferice sunt deconectate. Scopul utilizării *safe mode* este de a permite depanarea Windows-ului. După depistarea și rezolvarea problemelor existente, se face *reboot* și sistemul de operare se va încărca în mod normal. A se vedea și **sistem de operare și Windows.**

Sage. Agregator pentru *feed*-uri RSS și Atom care oferă facilitatea de auto-descoperire a acestora. Poate

importa/exporta liste OPML (<http://sage.mozdev.org/>). A se vedea și **agregator și feed.**

Sajax (*Simple AJAX*). Bibliotecă JavaScript *open source* creată cu intenția de a fi utilizată de programatorii Web pentru dezvoltarea aplicațiilor bazate pe tehnica AJAX, permițându-le acestora să apeleze funcții PHP, Perl sau Python din paginile Web via JavaScript, fără ca acestea să fie reîmprospătate (*refresh*) (<http://www.modernmethod.com/sajax/>). Detaliile legate de obiectul XMLHttpRequest, utilizat în scopul realizării aplicațiilor Web bazate pe AJAX, sunt abstractizate, utilizând fișierul *Sajax.php*. Acest lucru face mult mai simplă dezvoltarea aplicațiilor de tipul amintit, reducându-se totodată și șansele apariției unor erori. Legăturile utilizate în codul HTML vor conține numai apeluri de funcții JavaScript. *Sajax* oferă o cale modulară de dezvoltare a aplicațiilor Web bazate pe AJAX, prin utilizarea dinamică a apelurilor de funcții. A se vedea și **AJAX și JavaScript.**

salt la rând nou (*linefeed*). Caracter de control care determină saltul cursorului pe ecran la un rând nou. În setul de caractere ASCII, codul zecimal al acestuia este 10. În cazul imprimantelor, caracterul produce saltul hârtiei cu un rând. Ambele salturi se produc fără schimbarea poziției orizontale a cursorului sau a capului de tipărire. A se vedea și **retur de car.**

(a) **salva** (*save*). A scrie date sub forma unui fișier pe un suport de stocare. A se vedea și **fișier**.

Samba. Grup de aplicații Linux care folosește protocolul SMB (*Server Message Block*). Sistemele de operare Microsoft Windows și OS/2 folosesc protocolul SMB pentru aplicații client-server de partajare a fișierelor și imprimare în rețea. Samba permite sistemelor Linux să comunice folosind același protocol (*i.e.*, SMB) cu sistemele Windows. Astfel, sistemele Linux pot fi accesate de utilizatorii rețelei ca orice sistem Windows. De asemenea, Samba oferă aplicațiile necesare utilizatorilor de Linux pentru a accesa resursele partajate de utilizatorii sistemelor Microsoft Windows.

Serverul Samba le oferă următoarele servicii clienților unei rețele: partajare de fișiere, păstrarea de spații publice sau protejate prin nume de utilizator și parolă; partajare de imprimante și imprimare în rețea; autentificarea clienților din rețele Windows; WINS (*Windows Internet Name Service*).

Samba poate fi implementat în rețelele locale ale unităților de învățământ, pentru a îmbunătăți managementul și fluxul intern al documentelor.

Sandbox (*cutie de nisip*). Mecanism de securitate utilizat pentru execuția separată a unor programe. În mod obișnuit, mecanismul este folosit pentru execuția programelor netestate, a programelor care nu prezintă încredere, precum și a celor provenite de la diverși utilizatori (*e.g.*, *applet-uri Adobe Flash, Java*

Applets și Silverlight). A se vedea și **applet**.

saturație (*saturation*). În teoria culorilor, reprezintă cantitatea de culoare, specificată în procente. Este una dintre cele trei coordonate asociate unei culori în reprezentarea HSL: *hue* (H, nuanță), *saturation* (S, saturație), *lightness* (L, strălucire).

SAX (*Simple API for XML*). Metodă utilizată pentru procesarea documentelor XML. Este una dintre metodele care utilizează procesarea orientată pe flux, acestea având următoarele caracteristici: nu încarcă întregul document în memorie (din acest motiv, numai nodul curent este accesibil pentru procesare); nu permit editarea documentului; necesită timp scurt pentru procesare și resurse minime de memorie. În SAX, fiecare nod al arborelui XML asociat documentului procesat declanșează un eveniment XML (care trebuie tratate de funcții definite de programator), reprezentând apariția: marcajului de început al unui element; marcajului de sfârșit al unui element; unui element de tip `CDATA`; unei instrucțiuni de procesare.

Metoda SAX furnizează acces serial la documentul XML, citind fișierul în care este stocat documentul și apelând funcțiile definite de programator de fiecare dată când sunt declanșate evenimentele XML. Nu sunt posibile navigarea în documentul XML și modificarea acestuia. De asemenea, utilizarea metodei SAX necesită scrierea unei cantități relativ mari de cod. Este indicat ca această metodă să fie

folosită atunci când documentele XML care urmează a fi procesate sunt foarte mari, respectiv când se dorește obținerea unor cantități mici de date sau oprirea procesării înainte de terminarea acesteia. A se vedea și **DOM**.

SAX Project. Site dedicat metodei SAX de procesare a documentelor XML, întreținut de David Megginson (<http://www.saxproject.org/>). A se vedea și **SAX**.

scalabil (*scalable*). Caracteristică a unui produs software sau hardware care permite dezvoltarea acestuia fără modificări majore, în scopul de a satisface noi necesități.

scală de gri (*grayscale*). Set de diverse grade de gri (de la alb la negru) folosit în grafica pe calculator pentru a reda detaliile unei imagini alb-negru sau pentru a reprezenta o imagine color pe un dispozitiv de ieșire monocrom (*e.g.*, imprimantă). Pentru reprezentarea intensității unui pixel (*i.e.*, gradul de gri) se utilizează un număr de biți (*e.g.*, 8, 16). De exemplu, formatele grafice TIFF și GIF suportă nativ o scală de gri pe 16 biți. Valoarea minimă (0) este asociată cu negru, iar cea maximă (255 sau 65535) cu alb, valorile intermediare fiind asociate unor grade de gri.

scanner (*scanner*). Dispozitiv optic utilizat pentru a captura imagini de pe un suport (*e.g.*, hârtie) pe care le convertește în format digital.

schemă logică (*logical scheme*). Modalitate grafică utilizată pentru

descrierea algoritmilor. Prelucrările incluse într-un algoritm se reprezintă grafic prin blocuri, iar ordinea de execuție este indicată de săgeți. Sunt reprezentate grafic: operațiile (de început/sfârșit, de citire, de scriere, de atribuire, de decizie), structurile de control de bază (secvența, selecția, ciclul cu test inițial) și structurile de control derivate (ciclul cu test final, ciclul cu contor). A se vedea și **algoritm** și **pseudocod**.

schemă XML (*XML schema*). Descriere a structurii unui document XML exprimată în termeni de constrângeri referitoare la setul de elemente și atribute folosite, la modul de structurare al acestora, precum și la tipul conținutului său. Altfel spus, o schemă XML definește o clasă de documente XML care se conformează unui anumit model structural. O astfel de schemă este utilizată pentru validarea documentelor XML corespunzătoare (care respectă schema). Pentru crearea schemelor XML sunt utilizate două limbaje: DTD și XML Schema. A se vedea și **DTD** și **XML Schema**.

schimb de link-uri (*link exchange*). A se vedea **link exchange**.

SCO (*Sharable Content Objects, obiecte de conținut partajabile*). Obiecte de conținut reutilizabile specificate în standardul SCORM pentru administrarea obiectelor de învățare. Includ conținut și metadate. Pot fi utilizate ca blocuri constructive, prin asamblarea

lor obținându-se pachete de SCO-uri. Pentru a realiza agregarea conținutului (*i.e.*, asamblarea SCO-urilor în pachete), trebuie îndeplinite următoarele condiții: SCO-urile trebuie fundamentate și organizate pe baza unei structuri; instrucțiunile de asamblare trebuie scrise astfel încât un LMS să cunoască ordinea SCO-urilor; SCO-urile și instrucțiunile trebuie înglobate într-un pachet portabil. A se vedea și **LMS, obiect de învățare și SCORM.**

scope (domeniu de valabilitate). A se vedea **domeniu de valabilitate.**

SCORM (Shareable Content Object Reference Model). Standard pentru administrarea obiectelor de învățare dezvoltat de ADL. Standardul este definit printr-un set de specificații tehnice care formează un model de referință ce trebuie implementat atât de producătorii de conținut educațional, cât și de dezvoltatorii de LMS-uri, pe baza conceptului de *obiect de conținut (SCO)* care poate fi partajat/distribuit.

SCORM a fost creat de ADL ca urmare a unui mandat al Departamentului pentru Apărare al SUA din 1997, cu scopul de a dezvolta strategii pentru tehnologii educaționale și de instruire. În 2004 a fost publicată versiunea 1.3, denumită și SCORM 2004. Standardul SCORM este bazat pe rezultatele obținute de AICC (<http://www.aicc.org/>), IMS Global Learning Consortium, IEEE și Ariadne.

Modelul SCORM definește interrelaționarea componentelor cursurilor, modelelor de date și protocoalelor, astfel încât obiectele de învățare să poată fi utilizate de orice mediu de învățare local sau bazat pe Internet/intranet care se conformează modelului. Mai exact, SCORM definește: comunicarea între obiectul de învățare și serverul-gazdă, acesta din urmă fiind denumit și *run-time environment*; împachetarea datelor într-un format care să fie recunoscut de toate sistemele de instruire asistată de calculator, de programele de editare de conținut educațional și de produsele software de administrare a conținutului.

În versiunea SCORM 2004 este introdus conceptul de *secvențiere*, aceasta fiind un set de reguli care specifică ordinea în care studentul poate interacționa cu obiectele de învățare.

Cerințele impuse în SCORM pentru obiectele de învățare sunt: durabilitate, interoperabilitate, accesibilitate și reutilizabilitate. Domeniile de aplicabilitate a modelului sunt: oferirea de instruire bazată pe Web pentru cursanții individuali; crearea de obiecte de învățare electronice care pot fi reutilizate în diverse cursuri; arhivarea materialului educațional și a metadatelor, în scopul transferării lor (import/export) între diferite medii virtuale de învățare; urmărirea și memorarea înregistrărilor privind progresul școlar al unui student care evoluează în cadrul unei unități de învățare.

Modelul SCORM este alcătuit din trei specificații separate: 1) modelul de agregare a conținutului (CAM, *Content Aggregation Model*) – descrie modul în care sunt structurate și descrise pachetele de conținut, inclusiv metadatele asociate și informații despre secvențializare, fiind furnizată o schemă consistentă prin: definirea terminologiei legate de modelul de conținut; descrierea cerințelor pentru agregarea și împachetarea conținutului învățării; descrierea atributelor pentru metadate; descrierile și cerințele pentru definirea secvențierii și navigării; 2) modelul de secvențializare și navigare (SN, *Sequencing and Navigation*) – descrie modul în care pachetele declară secvențializarea între activități, modul în care utilizatorii pot formula cereri de navigare și modul în care acestea sunt interpretate de sistemul de management al învățării (LMS), ceea ce înseamnă că acest modul controlează și urmărește interacțiunea dintre utilizatori și LMS; aceste specificații se bazează pe specificațiile secvențierii simple ale consorțiului IMS; 3) mediul de execuție (RTE, *Run-Time Environment*) – descrie modul în care un sistem de management al învățării pune la dispoziția cursantului pachetele (lansare, *launch*), incluzând interfața de comunicație între pachete și sistem în timpul execuției și modelul de date folosit de sistem pentru a înregistra rezultatul interacțiunii între cursant și obiectele de învățare (model de date, *Data Model*).

În SCORM sunt utilizate următoarele categorii (sau niveluri) de obiecte de

conținut: articole (*assets*) – reprezintă conținut de bază (text, imagini, sunete, pagini web, obiecte pentru evaluare) sau alte informații care pot fi livrate clientului Web; SCO-uri (*Sharable Content Objects*) – desemnează o colecție formată din unul sau mai multe articole; un SCO reprezintă cel mai fin nivel de granularitate pentru materialele de curs care poate fi urmărit de un LMS în mediul de execuție SCORM; activități de învățare (*Learning Activities*) – reprezintă un eveniment instrucțional (e.g., un test inițial, un test final) sau un eveniment inclus într-un obiect de învățare sau prezentat ca un agregat de obiecte de învățare; structură de conținut (*Content Organization*) – este o hartă care poate fi utilizată pentru a agrega obiecte de învățare într-o unitate de instruire încheagată (e.g., curs, capitol, modul).

SCORM nu este acceptat de toți producătorii de platforme pentru *e-Learning*, motivul principal al acestui refuz fiind complexitatea standardului, care trebuie să fie ascunsă pentru utilizatorii finali prin intermediul unor soluții și instrumente software relativ complexe. Prin folosirea standardului SCORM, dezvoltatorii de conținut de învățare au posibilitatea de a proiecta și realiza materiale reutilizabile, care vor putea fi schimbate rapid și ușor între platforme pentru *e-Learning* interoperabile, în beneficiul studenților și educatorilor. A se vedea și **ADL, e-Learning, mediu virtual de învățare, SCO și obiect de învățare.**

screencast (*screencast*). Combinație între *screen* și *podcast*, desemnează o înregistrare video digitală cunoscută sub numele de *video screen capture*, utilizată adesea pentru a prezenta caracteristicile și modul de utilizare ale unor produse software. A se vedea și **podcast**.

Screencast-O-Matic. Aplicație *online* care oferă o modalitate gratuită și rapidă pentru realizarea unor înregistrări video ale ecranului calculatorului utilizatorului (denumite videocapturi sau *screencast-uri*) și pentru încărcarea acestora pe serverul aplicației, găzduirea fiind gratuită (<http://www.screencast-o-matic.com/>).

Înainte de crearea unui *screencast* trebuie precizate: limba utilizată; dimensiunea capturii (*Capture Size*) – poate fi aleasă una dintre valorile 640x480, 800x600, 1024x768 sau *Full Screen* (trebuie să fie 1024x768 sau mai puțin); microfonul utilizat: fără microfon (*No Audio*) sau unul dintre microfoanele existente.

Pentru a putea să încarce *screencast-uri* pe server și să le administreze, utilizatorul trebuie să dețină un cont. Acesta se poate crea gratuit, precizând o adresă de *e-mail* și o parolă.

screen saver (*program de protecție a ecranului*). Program care afișează o anumită imagine dacă utilizatorul nu folosește tastatura sau *mouse-ul* un anumit timp. Deși, inițial, acest tip de programe au fost create pentru a proteja

ecranul, astăzi ele au numai un rol decorativ, fiind încă destul de populare.

screenshot (*fotografie de ecran*). Imagine care include întregul ecran sau numai o parte a acestuia. Astfel de imagini sunt obținute folosind programe specializate, adesea gratuite (*e.g.*, *Jing*).

Scribd. Aplicație gratuită *online* care le permite utilizatorilor să posteze și să partajeze documente în multe formate (<http://www.scribd.net/>). Site-ul *Scribd* include un număr foarte mare de documente publicate de numeroșii săi membri.

Pentru a posta documente pe *Scribd* este necesar ca utilizatorul să-și creeze un cont. În acest scop, trebuie să precizeze adresa de *e-mail* (obligatorie), parola (obligatorie) și un nume de utilizator (opțional). Posesia unui cont de utilizator permite – în afară de postare – partajarea documentelor, gestionarea și actualizarea acestora.

script.aculo.us. Bibliotecă realizată sub licență MIT de Thomas Fuchs, pe baza bibliotecii *Prototype*, căreia îi sunt adăugate o serie de funcționalități suplimentare (*e.g.*, efecte vizuale, *drag-and-drop*, controale; <http://script.aculo.us/>). La implementarea bibliotecii au fost vizate următoarele scopuri: asigurarea posibilității folosirii marcajelor HTML, utilizarea unor standarde ca DOM, XHTML și CSS, compatibilitate cu majoritatea browserelor, orientare pe obiect completă, scrierea unei cantități reduse de cod pentru dezvoltarea

aplicațiilor bazate pe AJAX. A se vedea și **AJAX**, **licență MIT** și **Prototype**.

scrolling (*defilare*). A se vedea **defilare**.

SCSI (*Small Computer System Interface*, pronunțat *scuzzy*). Creată de fondatorii companiei Adaptec în 1979, SCSI reprezintă un set de standarde pentru conectarea și transferul datelor între periferice și calculatoare care asigură performanțe ridicate, conectivitate și expandabilitate.

Magistrala SCSI poate conecta o varietate de periferice inteligente la un calculator prin intermediul unui adaptor, acesta fiind în mod obișnuit o placă de extensie care se cuplează pe unul din *slot*-urile libere ISA, EISA, VL sau PCI ale calculatorului. Adaptorul asigură de obicei cuplarea atât a dispozitivelor interne, cât și a celor externe. Acestea sunt incluse în diverse categorii (*e.g.*, unități CD-ROM, scannere, unități magnetice, imprimante), cele mai răspândite fiind hard-disk-urile.

scurtarea URL-urilor (*URL shortening*). Tehnică utilizată în Web pentru crearea unui URL foarte scurt, care se poate folosi alături de cel original (presupus lung) sau în locul acestuia. A se vedea și **TinyURL**.

scurtătură (*shortcut*). Element (*e.g.*, pictogramă situată pe *desktop*) care permite accesul rapid la un program sau la un fișier. A se vedea și **desktop**.

SDK (*Software Development Kit*). A se vedea **set de instrumente de dezvoltare pentru software**.

Seagate Technology. Companie americană specializată în producerea de hard-disk-uri și soluții de stocare, înființată în 1979 de Alan Shugart sub numele de Shugart Technology, schimbat repede în Seagate Technology (<http://www.seagate.com/>). În 2006 a achiziționat Maxtor, companie producătoare de hard-disk-uri înființată în 1982 de James McCoy, Jack Swartz, Bob Teal și Raymond Niedzwiecki.

search engine friendly. Termenul descrie *design*-ul, meniurile și conținutul unui site optimizat pentru motoarele de căutare. Astfel, site-ul va obține o poziție cât mai bună în listele cu site-uri indexate de motoarele de căutare. A se vedea și **SEO**.

SearchMedia. Motoare de căutare gratuite realizate de medici pentru medici și alte categorii profesionale din domeniul medical, cu versiuni localizate în Marea Britanie, SUA, Franța și Spania. Prima versiune (cea britanică) a fost lansată în 2006. A se vedea și **motor de căutare**.

Second Life (*a doua viață*). Lume virtuală tridimensională despre care se vorbește din ce în ce mai mult, inclusiv în legătură cu posibilitatea de utilizare a acesteia în educație (<http://secondlife.com/>). Deși este un joc (creat de *Linden Research, Inc.*), *Second Life* se dezvoltă rapid pentru a permite interacțiuni sociale și folosirea în exteriorul jocului

a conținutului creat de utilizatori. Această lume virtuală are o economie proprie, bazată pe dolari Linden care pot fi schimbați pe dolari SUA (rata de schimb este L\$270/US\$1). Motivele pentru care oamenii își creează conturi și folosesc *Second Life* sunt variate: înscrierea în cluburi de dans, desfășurarea unor afaceri care permit conversia câștigurilor în bani reali în afara jocului, utilizare în educație, cercetare academică, exprimare artistică.

Utilizatorii sunt reprezentați prin avataruri în întregime personalizabile. Dialogurile dintre avataruri se fac prin tastarea mesajelor, iar unele acțiuni care presupun mișcări speciale (e.g., dansul) sunt oferite în cluburi virtuale în care avatarului propriu i se transmit mișcările respective, pe cale digitală, de la proprietarul virtual al clubului. Astfel, la cumpărarea unui pian virtual, avatarul care se așează la pian va ști imediat cum se cântă piesa respectivă! În general, orice obiect virtual achiziționat îi conferă utilizatorului virtual capacitatea și cunoștințele pentru a-l folosi în mod corespunzător.

Multe firme importante au cumpărat terenuri virtuale în *Second Life* și au construit clădiri virtuale în care au deschis filiale virtuale pentru a-și promova produsele. Un avatar care dorește să cumpere un telefon celular de un anumit tip îl poate achiziționa, folosind dolari Linden, de la magazinul virtual al firmei care îl comercializează. Practic la orice filială virtuală poate fi accesată pagina Web reală a firmei,

denumită *Old fashion Web site* (site-ul Web de modă veche).

În ediția din 8 iulie a ziarului *The Wall Street Journal* a fost publicat articolul *Avatarurile se teleportează afară din Second Life*, în care s-a relatat momentul în care câteva avataruri din *Second Life* s-au transferat într-o lume virtuală creată pe serverele IBM, acest eveniment fiind comparat cu primul pas pe Lună al omului („A small step for an avatar could turn out to be a giant leap for avatar-kind”, „un mic pas pentru un avatar, un salt uriaș pentru lumea avatarurilor”). A se vedea și **lume virtuală**.

sector (sector). Cea mai mică unitate de stocare a unui disc, putând păstra 512 octeți. Un hard-disk este format din mai multe discuri. Fiecare disc are câte două fețe. Fiecare față include piste care, la rândul lor, sunt împărțite în sectoare. A se vedea și **hard-disk**.

sector de boot (boot sector). Sector existent la începutul fiecărui disc, cu rolul de a identifica arhitectura discului (e.g., mărimea sectorului, mărimea cluster-ului). În cazul discurilor de *startup*, sectorul de *boot* conține un program care încarcă sistemul de operare.

securitate informatică (IT security). Desemnează protecția datelor într-un sistem informatic, implicând comunicarea, stocarea și procesarea acestora. Asigurarea securității unui sistem informatic se referă la următoarele aspecte: integritatea (asigură persoanele

care utilizează sistemul că datele sunt corecte); confidențialitatea (informațiile sunt folosite numai de către persoanele autorizate); valabilitatea (datele stocate sunt disponibile oricând este nevoie de ele). Securitatea unui sistem informatic trebuie să fie o preocupare constantă pentru dezvoltatorii acestuia. Pe de altă parte, posibilitatea de utilizare a unui sistem se poate reduce odată cu creșterea securității acestuia. Din acest motiv, trebuie întotdeauna să se asigure un echilibru între securitate și posibilitatea de utilizare a unui sistem.

Atunci când se vorbește despre securitatea sistemelor informatice se utilizează termeni ca riscuri (*risks*), amenințări (*threats*), vulnerabilități (*vulnerabilities*). Riscul reprezintă posibilitatea ca sistemul să sufere pagube sau pierderi. Amenințarea reprezintă o expresie a intenției de a produce diverse pagube. Vulnerabilitatea reprezintă un gol de securitate al aplicației, adică o slăbiciune a acesteia, care le permite utilizatorilor neautorizați să aibă acces asupra sa. Vulnerabilitățile unei aplicații pot avea drept cauză un design prost, o implementare greșită, dar și o administrare incorectă a acesteia. O amenințare poate exploata o vulnerabilitate.

Seesmic. Serviciu Web 2.0 de *video microblogging* (<http://www.seesmic.com/>). Având o interfață realizată în *Flash*, a fost dezvoltat de francezul Loic Le Meur, susținut de un număr de investitori (e.g., Atomico, un grup care îi include pe Niklas Zennstrom și Janus

Friis, cel din urmă fiind codezvoltator al aplicațiilor *KaZa* și *Skype*, cea de-a doua cumpărată în 2005 către eBay).

Conversațiile video postate pe *Seesmic* pot fi difuzate simultan pe *Twitter* și (în viitor) pe alte platforme. *Seesmic* oferă un plugin pentru platforma *WordPress*, care, odată instalat, le permite cititorilor blogului să posteze videoclipuri. Se preconizează ca utilizatorii *Seesmic* să aibă posibilitatea de a înregistra conversații *Skype* și *chat*. A se vedea și **video microblogging**.

SEI (*Sistem educațional informatizat*).

1. Program complex inițiat de Ministerul Educației și Cercetării în anul 2001, al cărui obiectiv de bază este susținerea procesului de predare-învățare în învățământul preuniversitar prin tehnologii moderne, materializare a strategiei guvernului român în domeniul informatizării și educației asistate de calculator. Programul este implementat prin intermediul unui parteneriat public-privat, principalele companii implicate fiind SIVCO România, HP și IBM. 2. Portalul SEI (<http://portal.edu.ro/>) reprezintă centrul de comunicare pe Internet între Ministerul Educației și cercetării și publicul larg din România (profesori, elevi, părinți). Pe acest portal sunt afișate începând cu anul 2001 rezultatele repartiției computerizate a elevilor în licee, școli de arte și meserii, precum și rezultatele examenului de titularizare. Din anul 2002 portalul are un forum de discuții foarte animat, cu un număr mare de utilizatori înregistrați.

(a) **selecta** (*select*). Operațiunea de alegere a unor opțiuni (e.g., într-o pagină Web), a unui text (e.g., într-un document *Office*) etc. A se vedea și (a) **deselecta**.

semantică (*semantics*). Cu referire la un limbaj de programare, reprezintă semnificația construcțiilor sintactice corecte (ce anume realizează instrucțiunile). A se vedea și **sintaxă**.

semnătură digitală (*digital signature*). Informație care identifică expeditorul unui document. Semnătura digitală este creată prin criptarea conținutului documentului, folosind o cheie criptografică a expeditorului. Rezultă că semnătura obținută este unică atât pentru fișier, cât și pentru deținătorul cheii. De asemenea, eventualele modificări ale documentului afectează semnătura, oferindu-se astfel atât integritate, cât și autentificare.

Semnăturile digitale utilizează criptarea asimetrică, în care se folosesc două chei: una pentru a crea semnătura și alta, legată de prima, pentru a o verifica. Într-un sistem de securitate a cheii publice, toți participanții au nevoie de propria cheie de semnare (e.g., cheie privată). Cheia publică este răspândită și identificată de certificatele digitale. Acestea sunt emise de terți de încredere, cunoscuți sub numele de autorități de certificare. A se vedea și **autoritate de certificare**.

semnătură de penetrare (*penetration signature*). Urmă lăsată la penetrarea unui sistem. Poate fi utilizată în

sistemele de detectare a intruziunilor. A se vedea și **penetrare** și **sistem de detectare a intruziunilor**.

semn de carte (*bookmark*). A se vedea **bookmark**.

send (*a trimite*). A transmite un mesaj sau un fișier (e.g., de *e-mail*).

sendmail. Program de poștă electronică, unul dintre cele mai vechi din această categorie, creat de Eric Allman. A se vedea și **poșta electronică** și **Allman, Eric**.

semnătură de virus (*virus signature*). Tiparul binar al unui virus. Este utilizat de un program antivirus pentru detectarea și eliminarea virusului. A se vedea și **virus**.

SEO (*Search Engine Optimization, optimizare pentru motoarele de căutare*). Strategie de *marketing* apărută în anii 1990, reunind totalitatea tehnicilor prin care un site Web este adus la o poziție cât mai bună în lista de rezultate returnată de motoarele de căutare pentru diverse cuvinte-cheie. Există numeroase companii care oferă servicii SEO. Tehnicile SEO se grupează în două mari categorii: *white hat SEO* (tehnici permise) și *black hat SEO* (tehnici dezaprobrate). Încadrarea unei tehnici într-o categorie sau alta se face analizând dacă aceasta respectă sau nu norme specifice furnizate de dezvoltatorii motoarelor de căutare. A se vedea și **black hat SEO** și **white hat SEO**.

serial (*serial*). Transmis sau prelucrat pe rând, unul câte unul. De exemplu, într-o transmisie serială datele sunt transmise bit cu bit. A se vedea și **paralel**.

serif (*serif*). În cazul unor fonturi (numite *cu serife*), reprezintă ornamentele aflate la capetele liniilor care alcătuiesc un caracter. Se crede că serifele își au originea în antichitatea romană, când literele erau sculptate în piatră. A se vedea și **font**.

SERP (*Search Engine Result Page*). Pagina cu rezultate furnizată de un motor de căutare. A se vedea și **motor de căutare**.

server (*server*). Într-o rețea locală sau în Internet, calculator sau program care răspunde comenzilor trimise de un client (*e.g.*, server de fișiere, server de *e-mail*, server Web, server FTP). A se vedea și **arhitectura client-server** și **client**.

Server. Unul dintre anteturile răspunsului HTTP. Conține informații despre serverul Web (*e.g.*, nume, versiune, producător), precum și despre alte aplicații care sunt utilizate de acesta în scopul îndeplinirii cererilor clientului (*e.g.*, module suplimentare, servere de aplicații). A se vedea și **răspuns HTTP**.

server de aplicații (*application server*). Denumit și *aplicație server* (*server application*), este o platformă software care asigură generarea conținutului dinamic solicitat de

utilizator, funcționând de obicei ca extensie a serverului Web.

Dintre avantajele utilizării unui server de aplicații pot fi menționate: asigură suport pentru sesiuni; permite utilizarea unor conexiuni persistente cu bazele de date; asigură *load-balancing*-ul; determină creșterea securității aplicațiilor create.

Serverele de aplicații pot fi utilizate în diverse scopuri: generarea dinamică și afișarea conținutului site-urilor Web; utilizarea poștei electronice (*Web-mail* și trimiterea de mesaje din aplicațiile Web); asigurarea suportului tehnic *online* pentru diverse produse (hardware sau software); crearea de aplicații pentru comunitate (*e.g.*, forumuri, *newsletters*, jocuri); realizarea de cataloage, broșuri și site-uri de informații și publicitate; crearea unor aplicații care necesită conectarea serverului Web la alt server, numit *backend server* (*e.g.*, server de baze de date); crearea unor aplicații personalizate. A se vedea și **tehnologii pentru server**.

server FTP (*FTP server*). Program server care utilizează protocolul FTP pentru a le oferi utilizatorilor posibilitatea de a încărca sau descărca fișiere pe/de pe un site FTP, prin Internet sau printr-o altă rețea TCP/IP. A se vedea și **client FTP**, **FTP**, **site FTP** și **TCP/IP**.

server de imprimare (*print server*). Calculator sau dispozitiv conectat la una sau mai multe imprimante și la calculatoarele-client printr-o rețea, care poate să accepte sarcini de imprimare

de la calculatoarele respective și să le trimită către imprimantele corespunzătoare.

server de nume (*name server*). În DNS este folosit în procesele de adresare, înlocuind sistemul inițial (neperformant) al fișierelor de *host*-uri, reținute de toate calculatoarele-gază. Fiecare server de nume va stoca adresele unor zone specifice (e.g., într-o rețea locală) și adresele altor servere de nume. Spațiul de nume DNS este împărțit în zone care nu se suprapun, într-o structură arborescentă. Fiecare zonă conține câte o parte a arborelui respectiv, împreună cu numele serverelor (în mod obișnuit se folosesc un server primar și, eventual, mai multe servere secundare) care păstrează informația despre acea zonă. A se vedea și **DNS**.

Server NCSA. Cunoscut și ca *NCSA HTTPd*, este un server Web creat la NCSA (*National Center for Supercomputing Applications*), Universitatea Illinois, SUA. Serverul NCSA și serverul creat la CERN au fost primele două servere Web. NCSA a reprezentat o sursă de inspirație pentru *Apache*, astăzi cel mai popular server Web. A se vedea și **server Web**.

server Web (*Web server*). Software care rulează pe calculatorul server, utilizat pentru livrarea conținutului Web. Cele mai cunoscute servere Web sunt: *Apache*, *Microsoft IIS*, *Sun Web Server*, *lighttpd* și *Zeus*. Conform datelor Netcraft, Inc., în iunie 2009 *Apache* era utilizat de circa 47,12% din

totalul domeniilor, urmat la distanță de *Microsoft IIS* (24,80%). Comunicarea dintre clientul Web (*i.e.*, browser) și serverul Web se realizează folosind protocolul HTTP.

Conținutul livrat browserului de serverul Web poate fi static (acesta este stocat în fișiere de pe calculatorul server) și/sau dinamic (acesta este generat de programe/scripturi sau API-uri apelate de serverul Web). Este de remarcat că livrarea conținutului static este mai rapidă decât cea a conținutului dinamic, mai ales în situația în care – în cazul ultimului tip – datele sunt extrase din baze de date.

Principalele caracteristici ale serverelor Web sunt: livrează atât conținut static, cât și conținut dinamic; utilizează module (legate sau încărcate dinamic) care oferă posibilitatea extinderii capacităților de bază oferite; asigură suport pentru conexiuni securizate prin criptarea fluxurilor de date; gestionează simultan mai multe site-uri Web, asigurând așa-numita găzduire virtuală (*virtual hosting*) pe baza aceleiași adrese IP (modalitate utilizată de toți furnizorii de servicii Internet); asigură compresia conținutului pentru a reduce mărimea răspunsului; pot fi configurate prin intermediul unui fișier text sau al unei interfețe grafice (e.g., serverul Web *Apache* poate fi configurat prin intermediul fișierului text denumit *httpd.conf*). A se vedea și **browser Web** și **HTTP**.

Service-JSON. Pachet PEAR dezvoltat de Michael Migurski. Furnizează un codificator și un decodificator pentru formatul JSON. Pentru codificarea unei valori PHP într-un obiect JSON se utilizează metoda `encode()` a clasei `Services_JSON`, metodă care primește ca argument valoarea respectivă. În prealabil, trebuie creată o instanță a clasei amintite, de exemplu `$json = new Services_JSON`. A se vedea și **JSON** și **PEAR**.

serviciu de blogging (*blogging service*). Aplicație Web (de obicei gratuită) care oferă utilizatorului posibilitatea de a crea bloguri rapid, fără a-i fi solicitate cunoștințe de programare (e.g., *Blogger*, serviciu oferit gratuit de compania Google). A se vedea și **blog**.

serviciu pentru personalizarea paginii de start (*homepage personalization service*). Oferă posibilitatea includerii – sub forma unor module numite *widget*-uri – a unor legături către paginile Web și instrumentele *online* pe care utilizatorul le folosește în mod obișnuit în cursul navigării pe Web.

Cele mai cunoscute servicii de acest tip sunt *NetVibes*, *iGoogle*, *PageFlakes* și *SuprGlu*, toate oferindu-le utilizatorilor opțiuni interesante și posibilități de personalizare diverse. *iGoogle* deține un ușor avantaj în preferințele utilizatorilor, deoarece beneficiază de resursele companiei Google. Cu toate acestea, *NetVibes* oferă o mai mare compatibilitate în ceea ce privește

widget-urile incluse, permițând funcționarea acestora în diverse sisteme de operare, servicii și browsere Web (e.g., Mac OS, iPhone/iTouch, *Opera*, Windows Vista, *Windows Live*, *Yahoo!* *Widgets*).

serviciu Web (*Web service*). Aplicație sau funcție independentă și autodescriptivă care poate fi apelată în Internet sau în intranetul unei firme.

Serviciile Web pot realiza diverse operații, de la simpla validare a unor date până la efectuarea de calcule complexe sau procesarea afacerilor. Ele realizează schimbul de date în sisteme distribuite via TCP, prin intermediul protocolului HTTP, folosind formatul XML pentru a împacheta cererea și răspunsul.

Conceptul de serviciu Web a apărut în 1999, odată cu încercările unor companii (e.g., Hewlett-Packard) de a realiza aplicații distribuite în Internet. Un pas înainte a fost făcut de compania Microsoft prin intermediul platformei .NET, aceasta având drept element de bază noțiunea de serviciu Web. Apariția platformei amintite a constituit un semnal pentru marile companii, care au început încorporarea rapidă a noii tehnologii în produsele lor.

În principiu, un serviciu Web funcționează astfel: utilizatorul efectuează o cerere, iar serviciul Web furnizează un răspuns, ambele în format XML. Acest proces este denumit generic *Web service consuming*. Tehnologiile care stau la baza serviciilor Web sunt următoarele: WSDL (*Web Service*

Description Language) – este o modalitate de descriere standardizată a serviciilor Web; UDDI (*Universal Description, Discovery and Integration*) – este un sistem pentru înregistrarea și regăsirea serviciilor Web; XML-RPC (*XML-Remote Procedure Call*) și SOAP (*Simple Object Access Protocol*) – sunt protocoale de comunicare între serviciile Web și clienți.

Descrierea serviciilor Web (locație, operațiuni, parametri și tipuri de date) se realizează prin intermediul limbajului WSDL, dezvoltat de IBM și Microsoft. WSDL este similar cu un limbaj de schemă XML. De altfel, WSDL utilizează toate tipurile de date din specificațiile XML Schema. Folosind acest limbaj, se pot crea fișiere XML cu seturi de definiții care descriu serviciile Web. Un fișier cu extensia *.wsdl*, care include un document WSDL, furnizează toate informațiile necesare pentru a accesa și a utiliza un serviciu Web. Documentul WSDL este folosit în cursul dezvoltării unui serviciu Web pentru a crea interfața acestuia. Utilizând diverse instrumente (e.g., Axis), se pot crea dintr-un fișier WSDL clase Java care să reprezinte serviciul Web.

Înregistrarea și regăsirea serviciilor Web se realizează prin intermediul standardului UDDI, care le oferă utilizatorilor o modalitate sistematică și unitară de a regăsi furnizorii serviciilor Web prin intermediul unui registru centralizat al acestor servicii.

Mecanismul de apelare (invocare) a serviciilor Web și de schimb al mesajelor între client și server este descris de un protocol de comunicare. Acesta este independent de platformă, limbaj de programare, rețea sau protocol de transport. Ca exemple, pot fi menționate XML-RPC și SOAP. A se vedea și **SOA**, **SOAP** și **XML-RPC**.

sesiune HTTP (*HTTP session*). Constă în cererile inițiate de client și în răspunsurile serverului Web. În cazul în care conexiunea TCP este nepersistentă, în cadrul unei sesiuni HTTP se efectuează o singură tranzacție între server și client (alcătuită dintr-o singură cerere și un singur răspuns HTTP), ceea ce înseamnă transferul unei singure resurse. Folosirea conexiunilor nepersistente determină încărcarea serverului Web și posibile congestii în rețeaua de comunicare.

Browsersle și serverele Web actuale suportă mecanismul *keep-alive* (introdus ca extensie în HTTP/1.0 și implementat în HTTP/1.1), care păstrează conexiunea TCP deschisă între două tranzacții succesive (conexiune persistentă), astfel încât clientul poate efectua cereri multiple de resurse în cadrul aceleiași sesiuni HTTP. În HTTP/1.1, toate conexiunile sunt considerate implicit persistente, dacă nu se specifică altceva în mod explicit. A se vedea și **HTTP**.

sesiune Web (*Web session*). Totalitatea accesărilor unor pagini dintr-o aplicație Web efectuate de același utilizator, în aceeași fereastră a browserului. O

sesiune se încheie atunci când utilizatorul realizează explicit această operație (e.g., prin execuția unui *click* pe o legătură sau pe un buton care o implementează) sau când închide browserul.

set de caractere (*character set*). Grup de caractere (litere, cifre, semne de punctuație, caractere de control) utilizate într-un sistem de codificare (e.g., setul de caractere ASCII). A se vedea și **caracter**, **setul de caractere ASCII** și **sistem de codificare a caracterelor**.

set de instrucțiuni (*instruction set*). Ansamblul instrucțiunilor pe care le poate executa un microprocesor (CPU). A se vedea și **CPU** și **instrucțiune-mașină**.

set de instrumente de dezvoltare Java (JDK, *Java Development Kit*). Produs creat de Sun Microsystems și destinat dezvoltatorilor Java, este cel mai cunoscut și utilizat SDK Java (<http://java.sun.com/javase/>). JDK include și un mediu de execuție Java (*Java Runtime Environment*).

Începând cu 2006, JDK a fost pus la dispoziție sub licență GNU GPL. Ca rezultat, codul-sursă JDK a contribuit la dezvoltarea OpenJDK (2007). A se vedea și **Java**, **mediu de execuție Java** și **SDK**.

set de instrumente de dezvoltare pentru software (SDK, *Software Development Kit*). Produs complex care permite crearea aplicațiilor pentru un anumit pachet software, un software-cadru, o platformă hardware, un sistem informatic,

o consolă joc video, un sistem de operare etc. În forma cea mai simplă, un SDK poate fi un API. În mod obișnuit, un SDK include instrumente de depanare și alte utilitare prezente în mediile IDE. Uneori, producătorii de SDK-uri înlocuiesc termenul software cu unul mai specific (e.g., DDK, *Driver Development Kit*). A se vedea și **API** și **IDE**.

setul de caractere ASCII (*ASCII character set*). Set de caractere standard (litere, cifre, semne de punctuație, caractere de control) reprezentate pe 7 biți, asociate valorilor ASCII cuprinse între 0 și 127. A se vedea și **ASCII**, **ASCII extins**, **caracter** și **set de caractere**.

sfârșit de fișier (EOF, *end-of-file*). Cod plasat automat de o aplicație după ultimul octet al unui fișier și utilizat pentru a indica sistemului de operare că în fișier nu mai urmează alte date. În Microsoft Windows, EOF este introdus folosind caracterul de control CTRL-Z (codul ASCII zecimal este 26). A se vedea și **ASCII**.

SGBD (*Sistem de Gestiune a Bazelor de Date*). Sistemul de programe care permite construirea bazelor de date, introducerea informațiilor în bazele de date și dezvoltarea de aplicații care le exploatează.

Un SGBD îi oferă utilizatorului posibilitatea de a avea acces la date utilizând un limbaj de nivel înalt, apropiat celui natural, pentru a obține informații, făcând abstracție de algoritmi aplicați privind selecționarea

datelor implicate și a modului de memorare a acestora.

Principalele funcții ale unui SGBD sunt: funcția de descriere (folosind un model de structură a unei baze de date, SGBD realizează definirea entităților, a caracteristicilor și a legăturilor dintre acestea, folosind un limbaj de descriere a bazelor de date); funcția de manipulare (introducerea datelor în baza de date se realizează prin intermediul unui limbaj de manipulare care dispune de anumite instrucțiuni. De asemenea, limbajul de manipulare permite consultarea și actualizarea bazei de date); funcția de utilizare (este realizată, în general, în cadrul limbajului de manipulare, care permite scrierea unor algoritmi de prelucrare a datelor).

Iată câteva dintre avantajele oferite de utilizarea SGBD-urilor: controlul redundanței datelor (în sistemele tradiționale de stocare a datelor, bazate pe fișiere text, se făcea risipă de spațiu de depozitare, deoarece aceleași date erau stocate în mai multe fișiere, în timp ce în SGBD se încearcă eliminarea redundanței prin integrarea fișierelor, astfel încât să nu se stocheze copii ale acelorași date); coerența datelor (prin controlul redundanței se reduce riscul apariției incoerenței datelor; astfel, dacă un articol este stocat o singură dată în baza de date, orice actualizare a valorii sale se realizează o singură dată, noua valoare fiind disponibilă imediat; dacă articolul este stocat de mai multe ori, orice actualizare afectează, în principiu, toate copiile sale); partajarea datelor (o

bază de date poate fi partajată de mai mulți utilizatori autorizați); integritate ridicată a datelor (integritatea bazei de date se referă la validitatea și coerența datelor stocate; în mod obișnuit, validitatea se referă la constrângeri care stabilesc reguli de coerență; de exemplu, o astfel de regulă poate stabili că salariul unui angajat nu poate depăși o anumită valoare); securitate ridicată (securitatea unei baze de date este asigurată prin protecția acesteia față de utilizatorii neautorizați; de asemenea, accesul la date permis unui utilizator autorizat poate fi limitat la o serie de tipuri de operații, e.g., inserție, actualizare, ștergere, selectare); productivitate crescută (un SGBD furnizează un număr mare de funcții pe care, într-un sistem de gestiune a datelor bazat pe fișiere, ar fi trebuit să le scrie programatorul); acces concurent (SGBD gestionează accesul concurent la date al utilizatorilor, evitând alterarea sau distrugerea acestora).

Pentru organizarea bazelor de date sunt utilizate o serie de modele conceptuale (sau structurale), cele mai folosite fiind cele numite *rețea*, *ierarhic*, *relațional*, *obiectual* și *modele hibride* (e.g., *obiectual-relațional*).

Pentru implementarea SGBD-urilor sunt utilizate următoarele modele arhitecturale: *mainframe*, *integrat*, *file-server* și *client-server*. În cadrul arhitecturii *client-server*, datele sunt centralizate pe un calculator server, fiind gestionate de un program server, iar utilizatorii le pot accesa prin

intermediul unei rețele (*i.e.*, Internet, Intranet), folosind un program client. SGBD-urile asigură partea de server (program server), iar aplicațiile de exploatare a datelor se află pe client (program client). A se vedea și **SGBDR**.

SGBDR (*Sistem de Gestiune a Bazelor de Date Relaționale*). SGBD care implementează modelul relațional, folosind arhitectura client-server. Deoarece majoritatea serviciilor Internet sunt utilizate în regim client-server, devine naturală folosirea SGBDR-urilor în aplicații Internet (*e.g.*, *e-business*, *e-commerce*), cel mai popular dintre acestea fiind MySQL.

În ceea ce privește costurile, SGBDR-urile se pot clasifica în două categorii: comerciale (proprietary) și, respectiv, cu sursă deschisă. Sistemele comerciale – dintre care pot fi amintite Oracle, SQL Server (Microsoft) și DB2 (IBM) – sunt scumpe, motiv pentru care sunt practic inaccesibile unor companii/organizații mici, instituții de învățământ sau persoane particulare. În schimb, SGBDR-urile cu sursă deschisă (*e.g.*, MySQL) pot fi utilizate fără costuri. Cele mai multe sisteme din ultima categorie sunt folosite ca *back-end* pentru aplicațiile Web dinamice. A se vedea și **bază de date relațională**, **MySQL**, **SGBD** și **utilizatori ai bazelor de date**.

SGML (*Standard Generalized Markup Language*). Standard ISO (8879:1986), descendent al IBM GML (*Generalized Markup Language*), este un metalimbaj

utilizat pentru definirea limbajelor de prelucrare electronică a textelor. A fost creat de un grup de lucru al ANSI (*American National Standards Institute*), condus de Charles Card și avându-i ca membri pe Norman Scharpf și Charles Goldfarb (acesta din urmă, alături de Edward Mosher și Raymond Lake, a inventat GML, în 1969). Prima prezentare a limbajului SGML a avut loc în 1980.

SGML stabilește o modalitate pentru crearea unor documente independente de platforme și aplicații, care pot păstra formatarea, indexarea și informațiile înlănțuite.

SHA (*Secure Hash Algorithm*). Familie de algoritmi utilizați pentru crearea unei reprezentări de 160 biți (SHA-0 și SHA-1) sau 256–512 biți (SHA-2) a unui mesaj sau a unui fișier de date, numită *rezumat de mesaj*. Algoritmii sunt construiți pe principii similare cu cele utilizate în proiectarea MD4 și MD5. Mesajul creat este utilizat pentru a calcula și verifica o semnătură digitală, ca măsură de securitate. A se vedea și **MD5**.

Shannon, Elwood Claude (1916–2001). Inginer electronist și matematician american, cunoscut ca „părintele teoriei informației”. Este faimos pentru inventarea teoriei informației, prezentată într-o lucrare de referință publicată în anul 1948. De asemenea, este considerat fondatorul teoriei proiectării circuitelor digitale și calculatoarelor numerice. În acest sens, la vârsta de 21 de ani, în teza de masterat scrisă la MIT, a demonstrat

că, folosind aplicațiile în electronică ale algebrei booleene, se poate rezolva orice problemă de logică numerică. Lu-crarea lui Shannon este considerată una dintre cele mai importante teze de mas-terat scrise vreodată. A se vedea și **algebra booleană**.

shareware (*shareware*). Software pen-tru care au fost înregistrate drepturi de autor, distribuit prin diverse metode (e.g., Internet) pentru a fi încercat de posibili cumpărători. Aceștia au drep-tul de utilizare gratuită pentru o anumită perioadă de timp (e.g., 30 de zile), după care trebuie să-l cumpere sau să-l dezinstaleze. A se vedea și **careware** și **freeware**.

SharpDevelop. IDE *open source* pen-tru limbajele de programare C#, Visual Basic.NET, F# și IronPython, utilizat ca alternativă la *Microsoft Visual Studio.NET*. Este dezvoltat de ICSharp Code Team (<http://sharpdevelop.com/>). A se vedea și **IDE**.

shell (*shell*). Program care asigură comunicația directă, prin comenzi scrise, între utilizator și sistemul de operare (e.g., *Bourne shell*, *C shell*).

shell script (*script pentru shell*). Script executat de *shell*-ul (i.e., interpretorul de comenzi al) unui sistem de operare. A se vedea și **shell**.

Shirky, Clay (n. 1964). Autor, consultant și profesor american având ca preocupare principală studiul efectelor sociale și economice ale

tehnologiilor Internet (<http://www.shirky.com/>).

shut down (*a opri*). A închide corect un calculator. Închiderea incorectă a aceș-tuia poate determina pierderea unor date.

Silicon Valley. Partea sudică a orașului San Francisco, situat în California, SUA. Numele său este datorat nu-mărului mare de producători de produse electronice care înglobează circuite integrate. După cum se știe, acestea din urmă au la bază elementul chimic siliciu (Si).

Silverlight. Cadru de lucru creat de Microsoft în scopul dezvoltării aplica-țiilor Web RIA (<http://silverlight.net/>). Acestea rulează în mediul de execuție *Silverlight*, un *plugin cross-browser* (*Internet Explorer*, *Safari*, *Firefox* și *Chrome*) și *cross-platform* (cu implementarea *Moonlight* pentru Linux) și cu suport pentru dispozitive mobile. *Silverlight* integrează multi-media, grafică, animații și inter-activitate într-un singur mediu de execuție. Versiunea 2 permite scrierea de cod .NET și C#, dar și Python și Ruby (prin intermediul DLR, *Dynamic Language Runtime*), ceea ce ușurează considerabil dezvoltarea aplicațiilor. A se vedea și **RIA**.

simplă precizie (*single precision*). Denumit și *binary32* (IEEE 754-2008), este un format utilizat pentru repre-zentarea numerelor în virgulă mobilă, care folosește patru octeți (32 de biți). Simpla precizie este cunoscută în unele

limbaje de programare (C, C++, Java) și ca *float*. În aceste limbaje, tipurile reale sunt *float*, *double* și *long double*. A se vedea și **dublă precizie**.

SimpleXML. Metodă de procesare a documentelor XML implementată pentru prima oară în limbajul Perl (*XML::Simple*) de către Grant McLean. Ca și DOM, utilizează procesarea într-un arbore. Începând cu versiunea 5.0, PHP furnizează un set de instrumente cunoscut sub numele de *SimpleXML* pentru procesarea documentelor XML. A se vedea și **DOM** și **SAX**.

Simula. Numele a două limbaje de programare (Simula I și Simula 67), dezvoltate în anii 1960 la Computing Center din Oslo de către programatorii norvegieni Ole-Johan Dahl și Kristen Nygaard. Din punct de vedere sintactic, Simula este văzut ca un superset al limbajului ALGOL 60. Simula este considerat primul limbaj de programare orientat pe obiect (POO). Limbajul a fost conceput pentru simulări, necesitățile acestui domeniu furnizând multe dintre caracteristicile limbajelor orientate pe obiect. A se vedea și **Dahl, Ole-Johan; Nygaard, Kristen** și **programare orientată pe obiect**.

sintaxă (*syntax*). Cu referire la un limbaj de programare, reprezintă totalitatea regulilor de scriere corectă a programelor, astfel încât acestea să fie acceptate de analizorul sintactic al limbajului. Aceste reguli se referă la vocabularul, expresiile și instrucțiunile folosite, precum și la definirea

funcțiilor și claselor. Sintaxa poate fi descrisă cu ajutorul diagramelor de sintaxă sau a notației BNF. A se vedea și **diagramă de sintaxă**, **expresie**, **instrucțiune**, **notație BNF**, **semantică** și **vocabular**.

sistem adaptabil personalizat (*adaptable personalization system*). Sistem care poate fi personalizat de către utilizator într-o manieră explicită. Astfel, acesta poate schimba conținutul, aspectul, structura sistemului etc., în funcție de opțiunile sau nevoile sale.

sistem bazat pe cunoștințe (KBS, *knowledge-based system*). Tip particular de sistem expert, este un sistem bazat pe metode și tehnici de inteligență artificială. Componentele principale ale unui astfel de sistem sunt baza de cunoștințe și mecanismul de inferență. A se vedea și **bază de cunoștințe**, **inteligență artificială** și **sistem expert**.

sistem de calcul (*computer system*). Ansamblu format din elemente hardware și software care procesează date într-un mod semnificativ. Un sistem de calcul relativ simplu este calculatorul personal, iar un sistem de calcul complex este Internetul. A se vedea și **calculator**, **calculator personal** și **Internet**.

sistem de codificare a caracterelor (*character encoding system*). Include un set de caractere și codurile numerice asociate acestora (*e.g.*, codul Morse, ASCII, Unicode). A se vedea și **set de caractere**.

sistem de detectare a intruziunilor (*intrusion detection system*). Sistem automat de alertare a operatorilor în urma producerii unei penetrări sau a unei alte operațiuni care contravine politicilor de securitate existente. Unele sisteme de acest tip au posibilitatea de a răspunde prin închiderea accesului sau prin colectarea unor date despre intrus. A se vedea și **penetrare și semnătură de penetrare**.

sistem de ferestre (*window system*). Software care permite ca ecranul unui sistem de calcul să fie împărțit în zone dreptunghiulare (*i.e.*, ferestre) care se comportă ca dispozitive de intrare/ieșire separate, aflate fiecare sub controlul unei aplicații. Utilizatorul are posibilitatea de a selecta oricare dintre ferestre, în mod obișnuit folosind *mouse*-ul. Cel mai cunoscut sistem de ferestre este X Window System. A se vedea și **XFree86** și **X Window System**.

sistem de marcare a resurselor (*tagging system*). Software care oferă posibilitatea marcării resurselor Web. Un astfel de sistem utilizează reguli referitoare la: cine poate marca o resursă (*e.g.*, numai utilizatorii înregistrați, iar aceștia își pot marca numai propriile resurse); ce fel de resurse pot fi marcate (*e.g.*, numai fotografiile, URL-uri); ce fel de *tag*-uri pot fi utilizate (*e.g.*, numai cele formate dintr-un singur cuvânt).

În general, un sistem de marcare a resurselor oferă: posibilitatea de marcare a resurselor (*tagging*) de către

utilizatori; posibilitatea ca utilizatorii să introducă descrieri ale resurselor; *feed-uri* RSS; servicii adiționale (grupuri, resurse favorite, *rating*, căutare).

Cele mai cunoscute categorii de sisteme de marcare sunt următoarele: sisteme de *social bookmarking* (*e.g.*, *delicious*, *connotea*, *CiteULike*); sisteme de *social networking* (*e.g.*, *LinkedIn* – relații profesionale, *Friendster* și *Hi5* – relații sociale); sisteme de *social sharing* (*e.g.*, *YouTube*, *Flickr*, *SlideShare*). De asemenea, sistemele de *blogging* (*e.g.*, *Blogger*, *WordPress*) utilizează marcarea resurselor, astfel încât utilizatorii dispun de o modalitate simplă și rapidă de a-și clasifica propriile *post-uri*. Ulterior, aceste *tag-uri* sunt extrase de agregatoarele de bloguri (*e.g.*, *Technorati*), care le utilizează pentru a descoperi direcțiile în care evoluează preocupările din blogosferă.

Cu toate că magazinele *online* tradiționale clasifică articolele vândute în categorii ierarhizate, unele dintre ele au început să folosească marcarea realizată de utilizatori (*tagging*). Oferirea acestei posibilități are rolul de a ușura găsirea produselor prin căutare și navigare.

Marcarea resurselor prin utilizarea sistemelor de marcare se face rapid și simplu, fără a necesita o pregătire anterioară în domeniul clasificării și indexării. A se vedea și **marcarea resurselor**.

sistem de numerație pozițional (*positional numeral system*). Sistem de numerație în care valoarea indicată de o cifră (*digit*) este valoarea cifrei

înmulțită cu o putere a unei constante numite bază. Aceasta reprezintă numărul de cifre distincte necesare scrierii unui număr. A se vedea și **digit**.

sistem de operare (*operating system*). Ansamblu de programe care asigură utilizarea optimă a resurselor fizice și logice ale unui calculator. Are rolul de a gestiona funcționarea componentelor hardware, de a coordona și controla execuția programelor de aplicație și de a permite comunicarea utilizatorului cu sistemul de calcul. Obiectivele unui sistem de operare sunt: să fie ușor de folosit, să fie eficient și să aibă capacitatea de a se dezvolta.

Serviciile de bază ale sistemelor de operare moderne au fost izolate sub forma unui nucleu (*kernel*), iar celelalte servicii au fost implementate sub forma unor programe utilitare. De exemplu, componenta *shell* nu este implementată în cadrul sistemului de operare, astfel încât acesta să poată utiliza mai multe interpretoare de comenzi. Funcțiile nucleului sunt următoarele: administrarea proceselor (crearea și încheierea proceselor, planificarea și administrarea proceselor, comutarea între procese, sincronizarea proceselor și asigurarea comunicării între acestea, administrarea blocurilor de control al proceselor), administrarea memoriei (alocarea spațiului pentru adrese, asigurarea schimbului între memoria internă și disc – *swapping*, administrarea segmentelor sau a paginilor), administrarea sistemului de intrare-ieșire (administrarea *buffer*-elor), suportul funcțiilor

(administrarea întreruperilor, monitorizarea). După cum se observă, funcția principală a unui sistem de operare constă în administrarea resurselor disponibile și planificarea utilizării lor în diferite procese. Într-un sistem de calcul sunt utilizate următoarele tipuri de resurse: fizice (componentele hardware ale calculatorului: procesor, memoria internă, modulele de control ale operațiilor de intrare-ieșire, dispozitive periferice) și logice (componentele software: programe și date). Sistemele de operare sunt structurate pe mai multe niveluri, numărul acestora fiind diferit de la un sistem la altul. Primul model structurat de sistem de operare a fost THE, construit în 1968 de E.W. Dijkstra și un grup de studenți de la Technische Hogeschool Eindhoven. Nucleul sistemului de operare UNIX este structurat pe două niveluri: *kernel* dependent de mașină (conține: rutinele de tratare a întreruperilor; driverele dispozitivelor periferice care asigură transferul de intrare-ieșire la nivel fizic; o parte de software pentru administrarea memoriei) și *kernel* independent de mașină (conține: rutinele de tratare a apelurilor de sistem; software pentru administrarea și planificarea proceselor; driverele dispozitivelor periferice care asigură transferul de intrare-ieșire la nivel logic; software care asigură paginarea, schimbul de date între memoria internă și memoria externă; sistemul de gestiune a fișierelor).

sistem de operare în timp real (*real-time operating system*). Sistem de operare creat pentru a fi utilizat într-un mediu de control al proceselor. A se vedea și **sistem de operare**.

sistem duplex (*duplex system*). În telecomunicații, sistem în care semnalul poate circula în ambele sensuri între expeditor și destinatar. Un astfel de sistem poate fi semi-duplex (*half-duplex*) – semnalul circulă pe rând în cele două sensuri – și duplex integral (*full-duplex*) – semnalul circulă simultan în ambele sensuri. A se vedea și **transmisie duplex**.

sistem expert (ES, *expert system*). Software, produs al inteligenței artificiale, care încearcă să reproducă performanțele unuia sau mai multor experți umani, pentru a rezolva probleme dintr-un domeniu particular pentru care nu există o soluție algoritmică bine definită. Din punct de vedere funcțional, un ES este un program a cărui principală caracteristică este derivată dintr-o bază de cunoștințe, împreună cu un algoritm de căutare specific metodei de raționare. Oricare ES are următoarele cinci componente: baza de cunoștințe (structură de date care conține ansamblul cunoștințelor specializate, introduse de expertul uman sau preluate din alte surse), motorul de inferențe (preia informațiile din baza de cunoștințe, elaborează planul de rezolvare a problemei și execută acțiunile incluse în planul respectiv), interfața de dialog (permite dialogul cu utilizatorii în timpul

sesiunilor de consultare), modulul de achiziție (preia cunoștințele specializate furnizate de expertul uman într-o formă care nu este specifică reprezentării interne, dar care este cunoscută de sistem), modulul explicativ (explică utilizatorilor cunoașterea de care dispune, raționamentul pe care îl desfășoară sau soluțiile obținute). A se vedea și **inteligență artificială**.

sistem informațional (*information system*). Ansamblu de informații, principii, concepte, reguli, canale de comunicație și tehnici, proceduri și mijloace folosite pentru tratarea informațiilor (culegerea, înregistrarea, prelucrarea și transmiterea datelor) în scopul obținerii de informații care se află la baza luării deciziilor.

site FTP (*FTP site*). Ansamblu de fișiere care se găsesc pe un server FTP. A se vedea și **client FTP**, **FTP** și **server FTP**.

site map (sau *sitemap*). A se vedea **harta site-ului**.

site-oglină (*mirror site*). Copie a unui site original foarte solicitat de utilizatori. Site-urile-oglină sunt folosite pentru a furniza surse multiple pentru aceleași informații, distribuind astfel o sarcină de același tip (e.g., cereri pentru *download*) către mai multe servere. Un site de tip *live mirror* este actualizat automat în momentul în care originalul este schimbat. A se vedea și **download** și **site Web**.

site sigur (*secure site*). Site care le oferă utilizatorilor tranzacții sigure, împiedicând prin măsuri specifice accesul neautorizat la numerele de cărți de credit și la alte informații cu caracter personal (*e.g.*, magazinele *online*). Un indiciu că un site este sigur este acela că adresa paginii în care se introduc date sensibile (*e.g.*, numere de cărți de credit) trebuie să înceapă cu *https*, pentru a fi garantată criptarea datelor din browser până la server.

site WAP (*WAP site*). Site creat sau convertit dinamic utilizând WML și accesat folosind un browser WAP instalat pe un terminal mobil (*e.g.*, telefon, PDA). A se vedea și **browser WAP**, **WAP** și **WML**.

site Web (*site Web*). Ansamblu de pagini Web interconectate, aflat în proprietatea unei persoane, organizații sau companii, găzduit pe un calculator care rulează un server Web și având asociată o adresă vizibilă pentru utilizatorii din Internet.

Un site Web poate fi disponibil și în cadrul unei *intranet*, aceasta reprezentând o rețea privată a unei organizații care utilizează tehnologiile Internet. De asemenea, un site Web poate fi disponibil și într-o *extranet*, aceasta fiind o rețea privată folosită în scopul partajării informațiilor sau operațiunilor unei organizații cu partenerii de afaceri.

site Web care se adaptează (*adaptive Web site*). Site Web care își îmbunătățește organizarea și modul de prezentare a informațiilor pe baza com-

portamentului utilizatorilor în timpul navigării. Pentru a realiza acest lucru sunt utilizate tehnici de *data mining* (extragerea datelor). A se vedea și **data mining** și **site Web**.

site-uri-oglină (*mirror sites*). Site-uri identice utilizate în mod obișnuit pentru a le oferi utilizatorilor aceeași informație folosind surse multiple. În particular, utilizatorii aflați în diverse regiuni geografice au la dispoziție mai multe surse pentru a descărca rapid anumite conținuturi. Un site de tip *live mirror* este actualizat automat odată cu site-ul original.

Six Degrees of Kevin Bacon. Joc *online* în care orice actor poate fi pus în legătură (prin rolurile sale) cu actorul Kevin Bacon în cel mult șase conexiuni (condiție: doi actori pot fi conectați numai dacă au jucat în același film). În anul 2007, Kevin Bacon a înființat *Six Degrees* (<http://sixdegrees.org/>), site de *social networking* axat pe activități caritabile. A se vedea și **teoria celor șase grade de separare**.

sizeof. Operator unar având ca rezultat numărul de octeți necesari pentru stocarea unei variabile de tipul operandului sau pentru stocarea rezultatului expresiei, dacă operandul este o expresie (trebuie reținut că aceasta nu se evaluează). În C++, operatorul are efect la compilare (când se stabilește tipul operandului). A se vedea și **operator**.

sketchcast (*sketchcast*). Combinație între *sketch* și *podcast*, desemnează o

schită (*sketch*) sub forma unui fișier video (incluzând, eventual, voce), în care un concept este descris și pus la dispoziție pentru rulare într-un blog/site Web sau *feed* RSS. A se vedea și **blog**, **feed**, **podcast** și **siet Web**.

Skencil. Fost *Sketch*, este un editor gratuit de grafică vectorială, scris în limbajul Python (<http://www.skencil.org/>). Are versiuni pentru un număr mare de platforme. A se vedea și **Python**.

Sketsa SVG Graphics Editor. Instrument comercial scris în Java, produs de compania KIYUT (<http://www.kiyut.com>). Include un număr mare de facilități (editor pentru sursele SVG, editor DOM, editor de resurse, instrumente de transformare etc.) și utilizează SVG ca format nativ de fișier. A se vedea și **SVG**.

Slideshare. Aplicație *online* gratuită care le permite utilizatorilor postarea și partajarea prezentărilor (<http://www.slideshare.net/>).

Aplicația permite încărcarea prezentărilor, sincronizarea acestora cu un conținut audio și partajarea lor (publică sau privată). De asemenea, prezentările pot fi etichetate folosind *tag*-uri. Prezentările publice pot fi descărcate sau incluse (*embed*) în site-uri sau bloguri, pe baza unui cod generat automat de aplicație.

Slideshare le permite utilizatorilor individuali și organizațiilor să încarce: prezentări – *ppt*, *pps* & *pot* (*Power Point*), *odp* (*Open Office*), *pdf*; documente și foi de calcul: *doc*, *rtf*, *xls*

(*MSOffice*), *odt*, *ods* (*OpenOffice*). Mărimea fișierelor încărcate nu trebuie să depășească 100 MB. Documentele în format *pptx* și *docx* (*MSOffice 2007+*) vor trebui salvate – înainte de a fi încărcate – în formatele mai vechi *ppt* și, respectiv, *doc* sau salvate ca PDF-uri. După încărcare, prezentările vor fi vizualizate într-un *player Flash*.

Folosind *SlideShare*, pot fi încărcate prezentări de business, lecții, cursuri academice, arhive ale unor conferințe etc. Utilizatorii pot marca prezentările, le pot descărca sau include în propriile site-uri sau bloguri. *SlideShare* oferă posibilitatea alcătuirii de grupuri de utilizatori cu interese similare. De asemenea, oricine poate căuta și găsi prezentări, în funcție de subiectele care îl interesează.

În domeniul educațional, *SlideShare* poate fi utilizat: de profesori (pentru postarea cursurilor și prezentărilor susținute la conferințe); de studenți (pentru postarea referatelor și proiectelor proprii, revistelor *online* etc.). Un instrument similar cu *Slideshare* este *MyPlick* (<http://www.myplick.com/>).

SmartNetBook. Director Web românesc, este un catalog de site-uri organizat pe categorii (*e.g.*, Afaceri, Casă și amenajări, Auto, Comerț, Educație, Massmedia, Sănătate, Societate, Muzică) și subcategorii (<http://www.smartnetbook.ro/>). A se vedea și **director Web**.

smartphone (*telefon inteligent*). A se vedea **telefon inteligent**.

Smarty. Unul dintre primele cadre de lucru (din punct de vedere cronologic) care au implementat modelul MVC, fiind preferat de un număr mare de programatori (<http://smarty.php.net/>).

SMIL (*Synchronized Multimedia Integration Language*). Pronunțat *smile*, este un limbaj bazat pe XML dezvoltat de Consorțiul Web. Scopul în vederea căruia a fost creat SMIL este acela de a dispune de un format declarativ care să faciliteze crearea prezentărilor multimedia (e.g., audio-vizuale) sincronizate. O prezentare multimedia este alcătuită, în principiu, din mai multe componente de diverse tipuri media (e.g., text, imagini, animație, video, audio). Acestea pot fi derulate în timp real (i.e., sincronizate) în diverse moduri (secvențial, paralel, exclusiv sau combinat).

Prima versiune a limbajului SMIL, 1.0, a fost lansată în anul 1998. Versiunea 2.0 a fost lansată de W3C prin recomandarea din august 2001. Versiunea 3.0 a fost lansată în decembrie 2008.

Semnificația numelui limbajului SMIL este următoarea: *Synchronized* – asigură suport pentru temporizare și sincronizare; *Multimedia* – utilizează resurse Web de diverse tipuri media (text, imagini, animație, video și sunet); *Integration* – integrează resurse diverse într-o singură prezentare; *Language* – este un limbaj declarativ, a cărui sintaxă se bazează pe XML.

Componentele multimedia ale unei prezentări SMIL sunt reprezentate de obiecte ca fluxuri de date și fișiere,

fiecare dintre acestea având propriul URI. În prezentările multimedia SMIL sunt introduse numai referințe la obiectele respective, acestea fiind stocate pe diverse servere. De asemenea, obiectele respective pot fi utilizate (în principiu) într-un număr nelimitat de prezentări.

Utilizând SMIL, dezvoltatorii pot realiza prezentări multimedia interactive, pot descrie comportamentul temporal al prezentării, pot asocia hiperlegături obiectelor media și descrie șablonul spațial al prezentării (*layout*). Deoarece SMIL este bazat pe XML, fișierele în care sunt salvate prezentările multimedia realizate folosind acest limbaj (cu extensia *smil* sau *smi*) sunt în format textual. Din acest motiv, au dimensiuni reduse. În plus, prezentările SMIL pot fi ușor regăsite pe Web (în funcție de conținutul lor) prin intermediul unui motor de căutare, lucru mai greu de realizat pentru obiectele care utilizează formate multimedia proprietare (e.g., *Adobe/Macromedia Flash*).

SMILGen. Editor SMIL, este un proiect *open source* dezvoltat sub licență BSD, scris în limbajul de programare Python (<http://sourceforge.net/projects/smilgen/>). A se vedea și **SMIL**.

SMTP (*Simple Mail Transfer Protocol*). Protocol care funcționează la nivelul aplicației al modelului de referință TCP/IP, utilizat pentru rutarea mesajelor de poștă electronică și definit în RFC 821. Ultima actualizare a

acestui se regăsește în RFC 5321, care include și extensia ESMTP (*Extended SMTP*). În timp ce serverele de *e-mail* și alți agenți de transfer de mesaje (MTA, *Mail Transfer Agent*) utilizează SMTP pentru trimiterea și primirea mesajelor, clienții folosesc SMTP numai pentru trimiterea mesajelor la un server de poștă electronică. Pentru receptarea mesajelor, aplicațiile client utilizează unul dintre protocoalele POP sau IMAP. A se vedea și **IMAP4** și **POP3**.

sniffer (*detector*). Program care poate intercepta și înregistra traficul dintr-o rețea digitală sau dintr-o porțiune de rețea. *Sniffer*-ul capturează fiecare pachet transmis și, eventual, decodează și analizează conținutul acestuia. A se vedea și **pachet IP** și **password sniffing**.

SOA (*Service-Oriented Architecture, arhitectură orientată pe servicii*). Este un tip de arhitectură software (bazată pe un set de principii) care presupune distribuirea funcționalității unei aplicații în unități distincte, de mici dimensiuni, numite *servicii*, care pot fi distribuite într-o rețea și utilizate împreună pentru a crea aplicații complexe. Serviciile sunt unități funcționale cu un grad mare de independență (*decoupling*) ale căror implementări concrete sunt necunoscute dezvoltatorului aplicației. Astfel, serviciul poate fi văzut doar drept interfață, maniera de procesare (*business logic*) putând fi separată de serviciul propriu-zis. Conceptul SOA este derivat din mai

vechile concepte ale calculului distribuit și programării modulare.

Dintre principiile arhitecturale de bază impuse de reglementările SOA pot fi amintite: serviciile trebuie să partajeze un contract formal, fiind definite de unul sau mai multe documente descriptive; serviciile trebuie să fie slab conectate (*service loose coupling*); serviciile trebuie să ascundă detaliile de implementare (*service abstraction*); serviciile trebuie să fie reutilizabile (*service reusability*); serviciile trebuie să ofere suport pentru compunerea cu alte servicii (*service composability*); serviciile trebuie să fie executate în mod autonom; serviciile trebuie să poată fi ușor descoperite (*service discoverability*).

Metodologia de proiectare și dezvoltare a aplicațiilor bazate pe SOA este cunoscută sub numele de SOAD (*Service-Oriented Analysis and Design*). O astfel de metodologie a fost propusă de IBM în 2004 – SOMA (*Service-Oriented Modelling and Architecture*). A se vedea și **servicii Web**.

SOAP (*Simple Object Access Protocol*). Protocol simplu bazat pe XML, realizat pentru schimbul de informații structurate (mesaje) într-un mediu distribuit. SOAP este o metodă de comunicare în Internet, folosind XML ca limbaj pentru alcătuirea mesajelor și HTTP ca protocol de transport al acestora. Astfel, SOAP este independent de platforma sau limbajul utilizate. În ceea ce privește serviciile Web, SOAP este utilizat ca protocol de

comunicare, descriind mecanismul de invocare (apelare) a serviciilor Web și de schimb al mesajelor între client și server.

Alternativele la SOAP sunt reprezentate de: XML-RPC (predecesorul lui SOAP), XMLP (*XML Protocol*, adică SOAP v2), ebXML (*electronic business using eXtensible Markup Language*), JAXM (*Java API for XML Messaging*) și JAXP (*Java API for XML Processing*).

Protocolul SOAP a fost creat de Microsoft și dezvoltat în colaborare cu Developer, IBM, Lotus și User Land, fiind dezvoltat astăzi de Consorțiul Web (W3C). SOAP a fost proiectat pentru a lucra cu HTTP, SMTP și alte câteva protocoale native Internet. În mod obișnuit, SOAP folosește pentru transportul mesajelor protocolul HTTP; pentru a sublinia acest lucru, este utilizată următoarea egalitate simbolică: SOAP= HTTP+XML). Așadar SOAP este: un standard W3C, un protocol utilizat pentru comunicarea între aplicații, un format pentru trimiterea mesajelor bazat pe XML, independent de platformă și de limbajul de programare.

Specificațiile SOAP definesc: un plic (*SOAP envelope*) – stabilește cadrul de lucru pentru descrierea a ceea ce conține un mesaj și cum trebuie procesat acesta; un set de reguli de codificare (*SOAP encoding rules*) – pentru a exprima instanțe ale tipurilor de date definite de aplicație; o convenție pentru reprezentarea apelurilor

de proceduri la distanță (*SOAP RPC representation*). Trebuie menționat că protocolul SOAP oferă posibilitatea expedierii de mesaje cu atașamente.

Un mesaj SOAP de la client către server este inclus într-o cerere HTTP (*HTTP Request*). Rezultatul se numește *cerere SOAP*. Aceasta poate să fie o cerere HTTP POST sau GET. O cerere POST trebuie să specifice cel puțin câmpurile `Content-Type` și `Content-Length`. Un mesaj SOAP de la server la client este inclus într-un răspuns HTTP (*HTTP Response*). Rezultatul se numește *răspuns SOAP*. Atât cererea, cât și răspunsul SOAP conțin fiecare câte un antet și câte un corp.

Există un număr relativ mare de instrumente SOAP dedicate. Iată câteva dintre acestea: *Apache Axis*, cu versiunile *Axis* (a treia generație a *Apache SOAP*, scris în Java) și *Axis C++*; *SOAP:Lite* pentru Perl; *Microsoft.NET*, pentru limbajele suportate de platforma .NET Framework. De asemenea, *SoapWare.org* include o listă cu dezvoltatorii de instrumente SOAP pentru cele mai utilizate medii de programare (Java, C/C++, C#, Delphi, PHP, Python etc.). A se vedea și **serviciu Web și XML-RPC**.

social bookmarking (*social bookmarking*). Modalitate *online* de a stoca, organiza, căuta și partaja adresele resurselor Web (*i.e.*, de a opera *online* cu semne de carte), în funcție de nevoile și interesele individuale. Astfel, dacă se întâmplă ca interesele unui utilizator să coincidă cu ale altuia,

acesta din urmă poate urmări semnele de carte ale primului utilizator și, de asemenea, îl poate ajuta să găsească alte pagini Web care îi pot fi folositoare.

Această modalitate de stocare a informațiilor despre paginile Web care interesează un utilizator rezolvă următoarele neajunsuri ale modalității clasice de creare a semnelor de carte: sunt stocate pe calculatorul utilizatorului, rezultând un acces restrictiv la acestea (utilizatorul nu le va putea folosi prin intermediul altui calculator); utilizatorul nu poate dispune de aceleași semne de carte acasă, la birou, în sala de clasă sau în deplasare; partajarea semnelor de carte este relativ dificilă; pentru a asocia unui site/unei pagini Web mai multe semne de carte (încadrate în diverse categorii), este necesar ca fiecare dintre acestea să fie creat separat. A se vedea și **bookmark** și **delicious**.

Social Bookmarking System (*sistem de social bookmarking*). Serviciu *online* care implementează modalitatea de lucru cu semnele de carte denumită *social bookmarking*. Ideea principală care stă în spatele unui astfel de serviciu constă în utilizarea marcajelor (*tags*) asociate unor site-uri/unor pagini Web. Unui site/unei pagini îi pot fi asociate mai multe marcaje. De exemplu, unei pagini care include un articol despre predarea matematicii în școală îi pot fi asociate marcaje ca *matematica*, *scoala*, *predare* etc. Căutările în serviciile de *bookmarking* colaborativ se realizează pe baza titlurilor semnelor de

carte, a conținutului paginilor Web și a marcajelor.

Dintre zecile de astfel de servicii existente pot fi amintite: *delicious*, *magnolia*, *furl*, *citeulike*, *connotea* și *diigo*. Iată câteva modalități de utilizare a acestor servicii în educație: găsirea unor utilizatori cu interese similare, care poate îmbunătăți rezultatele muncii unui individ sau poate conduce la noi colaborări; practica marcării resurselor de către utilizatori poate oferi noi perspective asupra cercetării efectuate de un individ, așa cum *cluster*-ele de *tag*-uri evidențiază șabloane care nu sunt vizibile imediat prin examinarea unor URL-uri; abilitatea de a crea pagini multiautor de semne de carte poate fi utilă în proiectele de echipă, fiecare membru putând să includă resursele pe care le descoperă, indiferent de locație sau moment de timp; urmărind un cont al unui utilizator într-un serviciu de *social bookmarking* poate fi analizată destul de bine evoluția muncii acestuia. Aspectul le este util educatorilor, pentru a fi la curent cu progresul studenților, sau studenților, care pot urmări descoperirile educatorilor. A se vedea și **bookmark** și **social bookmarking**.

social media (*social media*). Termen global utilizat pentru definirea unor activități diverse, care integrează tehnologie, interacțiune socială și creație prin cuvinte (text), imagini, audio și video. Termenul se referă la genuri media *online* noi, caracterizate prin: participare (încurajează contribuția sau

reacția din partea oricui este interesat; ascunde delimitarea dintre media și audiență); deschidere (serviciile *social media* sunt deschise la reacție și participare, încurajând votarea, comentariile și distribuirea informației; sunt rare cazurile în care există piedici în accesarea și folosirea conținutului, deoarece conținutul protejat prin parolă nu este agreat de utilizatori); conversație (în vreme ce media tradiționale – ziare, televiziune, cărți, radio – sunt definite prin termenul *broadcast* – conținut transmis și distribuit unei audiențe –, *social media* sunt văzute mai ales prin analogie cu o conversație în dublu sens, adică dialog); comunitate (le permite comunităților să se formeze rapid și să comunice eficiente; comunitățile împărtășesc interese comune, ca de exemplu pasiunea pentru fotografie, probleme politice sau un spectacol TV favorit); conectivitate (cele mai multe tipuri de *social media* se dezvoltă prin interconectare, făcând posibilă includerea legăturilor către alte site-uri, resurse sau persoane).

Modelul de comunicație promovat de media tradiționale este monologul. Acesta este înlocuit, în *social media*, de dialog. Astfel, în cadrul noilor genuri media, comunicația (devenită bidirecțională) capătă caracteristicile dialogului: este transparentă, autentică, vibrantă și dirijată de consumator.

În mod obișnuit sunt utilizate următoarele genuri principale de aplicații *social media*: rețele sociale, bloguri, wiki-uri, *podcast*-uri, canale de știri

(RSS și Atom), comunități pentru organizarea și partajarea unui anumit tip de conținut (fotografiile, video, legături), *microblogging*, *chat*, forumuri și lumi virtuale (e.g., *Second Life*). Se poate observa ușor că o mare parte dintre genurile/aplicațiile de tip *social media* sunt și instrumente Web 2.0, ceea ce înseamnă că există o zonă importantă de suprapunere a celor două concepte. A se vedea și **microblogging**, **podcast**, **Web 2.0** și **wiki**.

social media marketing (*marketing social media*). Promovarea unui produs sau a unei mărci prin *social media* (e.g., bloguri, *Twitter*, *LinkedIn*, *Facebook*, *YouTube*). A se vedea și **social media**.

societatea cunoașterii (KS, *knowledge society*). Prescurtare pentru *societatea bazată pe cunoaștere* (*knowledge-based society*). Principalele caracteristici ale KS sunt următoarele: asigură o diseminare fără precedent a cunoașterii către toți cetățenii, prin mijloace noi, folosind cu prioritate Internetul, cartea electronică și metodele de învățare electronică (*e-Learning*); urmărește extinderea și aprofundarea cunoașterii științifice și a adevărului despre existență; reprezintă o societate sustenabilă din punct de vedere ecologic; constituie o nouă etapă în cultură; asigură bazele unei viitoare societăți a conștiinței, a adevărului, moralității, creativității și spiritului. Deoarece cunoașterea este informație cu sens și care acționează (produce efecte), societatea cunoașterii este grefată pe societatea informațională

(bazată, în esență, pe Internet), neputând fi separată de aceasta.

Un vector al societății cunoașterii este un instrument care transformă societatea informațională într-o societate a cunoașterii. Vectorii de acest tip sunt tehnologici (e.g., agenții inteligenți, Internetul, nanotehnologia și nanotehnica) și funcționali (e.g., sistemul de îngrijire a sănătății, dezvoltarea unei culturi a cunoașterii și inovării, un sistem de învățământ bazat pe *e-Learning*, protejarea mediului înconjurător, managementul utilizării morale a cunoașterii). A se vedea și **e-Learning, era informației și societatea informațională**.

societatea globală (*global society*). Termen care desemnează două stadii de dezvoltare ale omenirii în ceea ce privește utilizarea informației și a tehnologiei mijlocite de calculator. Conform lui J.A. O'Brien (*Management Information Systems. Managing Information Technology in the Internet worked Enterprise*, McGraw-Hill, Boston, 1999), omenirea se află în cel de-al treilea stadiu de dezvoltare, denumit *societatea globală interconectată* (început în anii 1992–1993) sau *societatea rețea*. Stadiul următor (care se preconizează că va începe după anul 2010) este denumit *societatea globală informațională*. Primele două stadii ale „graficului” realizat de O'Brien sunt stadiul întreprinderii informatizate (corespunde perioadei 1970–2010) și stadiul muncitorilor cunoscători interconectați (început în 1980).

Răspândirea la scară globală a noilor tehnologii oferă o unică șansă țărilor mai puțin dezvoltate de a ajunge din urmă țările bogate, prin accesul rapid la cunoaștere, aceasta reprezentând un instrument decisiv al dezvoltării economiilor emergente. A se vedea și **era informației, societatea cunoașterii și societatea informațională**.

societatea informațională (*information society*). O etapă a civilizației umane care implică folosirea intensivă a informațiilor în toate sferile activității și existenței umane, cu un impact economic și social foarte important. Societatea informațională bazată pe cunoaștere presupune, alături de progresul tehnologiilor informatice și de comunicații, integrarea dimensiunilor socială, culturală, ambientală și economică ale civilizației umane. A se vedea și **era informației și societatea cunoașterii**.

socket (*soclu*). Combinație dintre adresa IP a unei mașini și un număr de port, identifică un serviciu dintr-un nod al unei rețele. Un *socket* furnizează facilități pentru crearea de fluxuri de intrare și de ieșire, care permit datelor să fie schimbate între client și server. Atunci când se stabilește o conexiune sau un canal de comunicație, atât clientul, cât și serverul vor avea câte un *socket*. Un client caută serverul, stabilește o conexiune prin care poate fi trimisă informația, pentru ca la final să se deconecteze lăsând liber *socket*-ul pentru o utilizare ulterioară.

Numerele de port pentru *socket*-urile serverului sunt cunoscute clientului. De exemplu, un server HTTP utilizează un *socket* care „ascultă” la portul 80. Dacă un client dorește să comunice cu un server HTTP, va ști să contacteze *socket*-ul care ascultă la portul 80.

Numerele de port pentru client sunt stabilite de obicei de sistemul de operare. Este lipsită de importanță alegerea portului pentru client, deoarece serverul nu trebuie să cunoască dinainte de la care port vor trimite clienții cereri. Atunci când un *socket*-client trimite un pachet către un *socket*-server, acesta va conține și numărul de port, și adresa IP a mașinii clientului. Folosind informațiile primite, serverul va ști cine a trimis cererea și va putea să îi răspundă. A se vedea și **adresă IP și port**.

software (*software*). Sistem de programe pentru calculator. Desemnează partea logică a unui sistem de calcul și include sisteme de operare, drivere și aplicații. A se vedea și **hardware**.

software de sistem (*system software*). Programe care alcătuiesc sistemul de operare. A se vedea și **sistem de operare**.

software pentru analiza rețelelor sociale (*social network analysis software*). Instrumente utilizate pentru a reprezenta nodurile (agenții) și muchiile (relațiile) dintr-o rețea socială, precum și datele pe care aceasta le conține. Ca și în cazul altor instrumente, datele pot fi salvate într-un fișier

extern. Instrumentele amintite le permit cercetătorilor să investigheze orice rețea socială, inclusiv pe cele de mari dimensiuni, furnizând funcții matematice care pot fi aplicate modelului de rețea. Reprezentarea vizuală a rețelelor sociale este importantă pentru a înțelege rezultatul acestor analize. Instrumentele de analiză sunt utilizate pentru a schimba aspectul, culorile utilizate, mărimea și alte caracteristici ale reprezentărilor vizuale amintite.

Dintre instrumentele folosite pentru analiza rețelelor pot fi menționate *NetMimer*, *InFlow* (pentru aplicațiile de business), *GUESS* și *ORA* (pentru cercetările științifice și educaționale). A se vedea și **analiza rețelelor sociale și rețea socială**.

software pentru blogging (*blogging software*). Program care permite instalarea și personalizarea unui blog. În acest caz, este necesar ca utilizatorul să fie familiarizat cu limbajele folosite pentru crearea paginilor Web (*e.g.*, *WordPress*). A se vedea și **blog**.

software pentru wiki (*wiki software*). Numit și *motor wiki* (*wiki engine*), face posibilă crearea unui wiki. Un software pentru wiki se aseamănă cu alte sisteme de management al conținutului (CMS, *Content Management System*), existând totuși deosebiri în ceea ce privește controlul accesului, crearea conținutului și organizarea site-ului.

Software-ul pentru wiki se poate clasifica în două categorii: software numai pentru wiki, cu un set minimal

de facilități: proces de înregistrare simplu, istoricul schimbărilor, blocarea paginilor, blocarea IP-urilor (e.g., *MediaWiki*); software cu un set complet de facilități: înregistrare complexă, managementul fluxului de lucru, permisiuni pentru utilizatori, galerii de imagini, panou pentru administrare etc. În această ultimă categorie este inclus și *TikiWiki*, utilizat pentru crearea wiki-ului *Wiki Wacky World*.

O serie de pachete software pentru wiki sunt disponibile gratuit. Cerințele de instalare includ încărcarea/copierea fișierelor pe server, crearea bazelor de date (e.g., MySQL, PostgreSQL, Oracle) și a unui cont de utilizator, precum și rularea scriptului de instalare. Cunoștințele necesare pentru instalarea și exploatarea unui wiki depind de tipul particular de software utilizat. În mod obișnuit, nu sunt necesare cunoștințe de (X)HTML, SQL și CSS. Cunoștințele de PHP sunt utile pentru personalizarea wiki-ului.

Cele mai multe *wiki*-uri permit personalizarea, dar modul în care este implementată această facilitate diferă de la un software la altul. În unele cazuri, trebuie modificate o serie de fișiere. În altele, se va modifica un grup de șabloane. O serie de *wiki*-uri permit selectarea dintr-o serie de teme existente (de asemenea, se poate seta permisiunea ca utilizatorii să aleagă o altă temă, după plac). La început, este recomandat să se accepte cât mai multe setări implicite, urmând ca – după o perioadă de acomodare – acestea să fie

schimbate, dacă se va considera necesar.

Multe *wiki*-uri nu permit folosirea marcajelor (X)HTML. Unele (e.g., *Tiki Wiki*) lasă această decizie la latitudinea administratorului. Altele (e.g., *Media Wiki*) permit utilizarea unui număr limitat de marcaje. Această politică previne inserarea unui cod HTML malițios.

Din punctul de vedere al costului, *software*-ul pentru *wiki* poate fi gratuit (e.g., *DidiWiki*, *DokuWiki*, *PhpWiki*, *TigerWiki*, *TikiWiki*) sau comercial (e.g., *EditMe*, *JotSpot*, PAUX, *Thought Farmer*). În ceea ce privește licențierea, acest tip de software poate fi proprietar (e.g., *BrainKeeper*, *Clearspace*, *JotSpot*, *PicoWiki*) sau distribuit sub o licență *open source*. Aceasta din urmă poate fi BSD (e.g., *QwikiWiki*), GPL (e.g., *DidiWiki*, *MediaWiki*, *PhpWiki*) sau LGPL (e.g., *Tikiwiki*). A se vedea și **wiki**.

software proprietar (*proprietary software*). Software pentru care au fost înregistrate drepturi de autor de către o persoană sau o companie și care nu poate fi utilizat fără cumpărarea sau obținerea printr-o altă modalitate (e.g., donație către o instituție de învățământ) a unei licențe. Termenul este utilizat în opoziție cu cel de software liber.

software social (*social software*). Termen „umbrelă” pentru un număr mare de aplicații care facilitează atât comunicarea și interacțiunea, cât și partajarea informațiilor și a resurselor.

A fost utilizat pentru prima oară în 1987 de Eric Drexler. Conform Wikipedia, software-ul social este acel software „care le permite oamenilor să se întâlnească, să se conecteze sau să colaboreze folosind conectarea mediată de computer”. După Clay Shirky, creditat pentru popularizarea acestui concept începând cu anul 2002, software-ul social este „software-ul care suportă interacțiunea în grup”.

După Stowe Boyd, aplicațiile sociale „crează, descoperă, suportă sau manageriază într-un fel sau altul relațiile interpersonale”. Acesta afirmă că software-ul social se distinge de alte aplicații prin următoarele caracteristici: permite conversații între indivizi sau grupuri – sunt incluse aici atât conversațiile în timp real (e.g., mesageria instantanee), cât și cele asincrone (*slow time conversation*; e.g., spațiile de lucru colaborativ); asigură suport pentru *feedback* social – permite unui grup să aprecieze contribuția altora, conducând astfel la crearea unei reputații digitale; asigură suport pentru rețele sociale – asigură crearea și managementul unei expresii digitale a relațiilor dintre indivizi, ajutându-i în același timp să creeze noi relații.

În opinia lui Tim Coates, „software-ul social este o subclasă de software care are drept preocupare augmentarea socializării umane și/sau abilităților de relaționare prin structuri de mediere (distribuite sau centralizate)”.

Deși termenul software social este confundat adesea cu Web 2.0, trebuie

precizat clar că acești doi termeni nu desemnează același lucru, primul fiind mai cuprinzător. Astfel, primul termen include și aplicații care făceau posibile comunicarea și interacțiunea dintre oameni și în cadrul vechiului Web. Aici pot fi amintite grupurile de discuții, forumurile, jocurile multiplayer etc. Cu toate acestea, dezvoltarea software-ului social a cunoscut o adevărată explozie abia în epoca Web 2.0. De altfel, serviciile Web 2.0 orientate pe comunicare și pe dezvoltarea comunităților virtuale sunt incluse adesea în categoria software-ului social. A se vedea și **Web 2.0**.

Sogou. Motor de căutare în limba chineză, lansat în august 2004. Poate căuta text, imagini, muzică și hărți (<http://www.sogou.com/>). A se vedea și **motor de căutare**.

Sohu. Motor de căutare chinez (<http://www.sohu.com/>). A se vedea și **motor de căutare**.

Solaris. Mediu de calcul distribuit bazat pe UNIX, creat de Sun Microsystems, Inc. (1992) și utilizat pe scară largă ca sistem de operare pentru servere. Există versiuni de Solaris pentru calculatoare SPARC, platforme Intel și calculatoare PowerPC. A se vedea și **Sun Microsystems, Inc.**

sortare alfanumerică (*alphanumeric sort*). Metodă de reordonare (sortare) a datelor în care, spre deosebire de metoda lexicografică, semnele de punctuație, cifrele și literele (cele mici fiind așezate înaintea celor mari) apar

ca trei grupuri separate. A se vedea și **alfanumeric și sortare lexicografică**.

sortare ascendentă (*ascending sort*). Sortare prin care se obține așezarea datelor în ordine crescătoare.

sortare descendentă (*descending sort*). Sortare prin care se obține așezarea datelor în ordine descrescătoare.

sortare lexicografică (*lexicographic sort*). Metodă de sortare a datelor în ordine alfabetică, utilizată în dicționare, în care, spre deosebire de metoda alfanumerică, numerele sunt localizate în funcție de ortografierea acestora, nu într-un grup separat (adică în locul în care ar apărea dacă ar fi scrise cu litere; e.g., 341 ar face parte din grupul cuvintelor care încep cu litera T). Totuși, cele mai multe dicționare tehnice publicate astăzi plasează intrările numerice la începutul (sau, mai rar, la sfârșitul) cărții, și nu răspândite în ordine lexicografică. A se vedea și **sortare alfanumerică**.

sortare prin inserție (*insertion sort*). Algoritm pentru sortarea tablourilor, care seamănă oarecum cu sortarea prin selecție. Tabloul este împărțit imaginar în două părți, una sortată și alta nesortată. Inițial, partea sortată conține primul element al tabloului și partea nesortată conține restul tabloului. La fiecare pas, algoritmul ia primul element din partea nesortată și îl inserează la locul potrivit în partea sortată. Când partea nesortată nu mai conține nici un element, execuția algoritmului se încheie. Complexitatea algoritmului

este $O(n)$. Algoritmul este folosit destul de des pentru sortarea tablourilor cu număr mic de elemente. A se vedea și **algoritm de sortare**.

sortare prin interclasare (*merge sort*). Algoritm de sortare de tip *divide et impera*, inventat de John von Neumann în 1945, având complexitatea $O(n \log n)$. Are la bază algoritmul de interclasare. Acesta divizează lista de sortat în două subliste care sunt apoi ordonate recursiv, după care se execută o operație de interclasare pentru obținerea listei finale. Implementările clasice folosesc trei vectori, doi pentru subliste și unul pentru obținerea rezultatului final, însă algoritmul poate fi modificat astfel încât să lucreze cu doar doi vectori: unul pentru datele de intrare și unul în care se vor memora elementele ordonate. A se vedea și **algoritm de sortare și divide et impera**.

sortare prin metoda arborelui heap (*heap sort*). Algoritm de sortare a vectorilor, care nu necesită memorie suplimentară (sortează vectorul pe loc), având complexitatea $O(n \log n)$. Funcționează în două etape: 1) Vectorul care trebuie sortat este transformat într-un arbore heap; 2) Dimensiunea heap-ului se reduce treptat prin scoaterea valorii din rădăcină. Valorile rezultate în ordine descrescătoare sunt plasate în vector pe spațiul eliberat, de la dreapta la stânga. A se vedea și **algoritm de sortare și heap**.

sortare prin metoda bulelor (*bubble sort*). Denumit și *sortare prin schimb* (*exchange sort*), este un algoritm simplu pentru sortarea listelor (tablouri unidimensionale). Algoritmul constă în următoarele: se compară fiecare pereche de elemente pornind de la sfârșitul listei și, dacă elementele sunt în ordine inversă, se vor interschimba; dacă a fost făcută cel puțin o interschimbare, se repetă pasul anterior. Sortarea este terminată atunci când nu se mai realizează nici o interschimbare, cea mai mare valoare aflându-se la sfârșitul listei. Sortarea prin metoda bulelor are complexitatea $O(n^2)$. Algoritmul este ineficient pentru sortarea unui volum mare de date. A se vedea și **algoritm de sortare**.

sortare prin selecție (*selection sort*). Algoritm de sortare a tablourilor unidimensionale. Tabloul este împărțit în două părți imaginare, una sortată și una nesortată. La început, partea sortată nu conține nici un element, în timp ce partea nesortată conține întreg tabloul. La fiecare pas, algoritmul găsește elementul minim din partea nesortată și îl adaugă la finalul părții sortate. Atunci când partea nesortată nu mai conține nici un element, algoritmul se oprește. Complexitatea algoritmului este $O(n^2)$. Sortarea prin selecție este un algoritm destul de ineficient pentru sortarea unui volum mare de date. A se vedea și **algoritm de sortare**.

sortare rapidă (*quick sort*). Algoritm de sortare des folosit pentru sortarea tablourilor. Algoritmul utilizează meto-

da *divide et impera* și este cel mai cunoscut algoritm din clasa celor cu complexitate medie ($O(n \log n)$). Iată varianta recursivă a acestuia: 1) Alegerea unei valori pivot. De obicei, ca valoare pivot se alege valoarea elementului din mijlocul tabloului, dar se poate lua orice altă valoare aflată în intervalul valorilor sortate, chiar dacă nu este prezentă în tablou; 2) Partiționarea. Se rearanjează elementele astfel încât elementele care sunt mai mici decât pivotul sunt mutate în partea stângă a acestuia, iar elementele care sunt mai mari decât pivotul sunt mutate în partea dreaptă a acestuia. Valorile egale cu pivotul pot sta în orice parte a tabloului. Părțile tabloului nu trebuie să aibă toate aceeași dimensiune; 3) Sortarea ambelor părți. Se aplică recursiv algoritmul de sortare rapidă în partea stângă și în partea dreaptă. Algoritmul este foarte util pentru sortarea unui volum mare de date. A se vedea și **algoritm de sortare și divide et impera**.

SourceForge.net. Site care găzduiește aplicații *open source* (<http://sourceforge.net/>). Furnizează servicii care îi ajută pe programatori să partajeze aplicații la nivel mondial. La începutul anului 2009 găzduia peste 230 000 de proiecte software și avea peste 2 milioane de utilizatori înregistrați. Site-ul se află în proprietatea *SourceForge, Inc.* (<http://web.sourceforge.com/>).

spam (*spam*). Abuzul în utilizarea unor sisteme de trimitere a mesajelor pe cale electronică prin trimiterea de mesaje nesolicitate. Cea mai cunoscută formă

de *spam* constă în trimiterea mesajelor de *e-mail*. De asemenea, termenul este aplicat și abuzului referitor la: mesageria instantanee, bloguri, site-uri wiki, telefonie mobilă, forumuri, rețele sociale etc. A se vedea și **spammer**.

spamdexing (*spamdexing*). Cunoscută și ca *search spam* sau *search engine spam*, implică un număr de metode utilizate în scopul de a manipula relevanța resurselor indexate de motoarele de căutare. Aceste metode sunt incluse în următoarele categorii: *content spam* (e.g., folosirea *tag*-ului `meta`, folosirea paginilor de tip *doorway page*); *link spam* (e.g., *spam* în bloguri, *spam* în wiki-uri, comentarii *spam*); alte tipuri (e.g., utilizarea site-urilor oglindă, redirectare URL, *cloaking*). A se vedea și **cloaking** și **doorway page**.

spam in blog (*spam în blog*). Denumit și *blog spam* sau *comment spam*, este o formă de *spamdexing*. Constă în postarea unor comentarii în bloguri, *wiki*-uri etc., în scopul promovării unor site-uri. În general, orice aplicație Web care acceptă și afișează hiperlegături trimise de utilizatori este susceptibilă de a fi ținta acestei forme de *spam*. Adăugarea artificială a legăturilor către site-urile *spammer*-ilor prin această metodă este menită să determine creșterea *PageRank*-ului calculat de motoarele de căutare pentru acestea. A se vedea și **spam**, **spamdexing** și **PageRank**.

spammer (*spammer*). Persoană care folosește *spam*-ul pentru a-și atinge

obiectivele, contrare de multe ori intereselor persoanelor care suportă acțiunile sale. A se vedea și **spam**.

SPARC (*Scalable Processor Architecture*). Specificație de microprocesor RISC elaborată de compania Sun Microsystems. A se vedea și **RISC**.

(a) **spația selectiv** (*kern*). A ajusta spațiul dintre două litere ale unui cuvânt scris cu un font proporțional, independent de spațierea celorlalte litere ale cuvântului. A se vedea și **font proporțional**.

spații de nume XML (*XML Name-spaces*). Mecanism care le permite dezvoltatorilor XML să utilizeze în mod unic nume de elemente și atribute, evitându-se astfel conflictele de nume definite în diverse vocabulare.

Declarația unui spațiu de nume include atributul `xmlns`, a cărui valoare este un URI, numit identificator (al spațiului de nume), folosindu-se următoarea sintaxă: `xmlns:prefix="URI"`. Prefixul *prefix* poate conține caracterele folosite pentru alcătuirea numelor elementelor XML, cu excepția caracterului două puncte („:”). Când este utilizat în documentul XML, numele unui element/atribut trebuie precedat de prefixul spațiului de nume specificat anterior, acestea fiind separate folosind caracterul două puncte (*prefix:nume*). Împreună, reprezintă un nume calificat (*Qualified Name*, numit pe scurt *QName*).

Un spațiu de nume are un domeniu de valabilitate, ceea ce înseamnă că acesta

se referă la elementul în care este specificat, precum și la descendenții săi. Astfel, orice declarație a unui spațiu de nume plasată în elementul-rădăcină al unui document XML devine valabilă pentru toate elementele documentului. Într-un document XML se poate declara un spațiu de nume implicit, utilizat de toate elementele din documentul respectiv care nu sunt prefixate, folosindu-se următoarea sintaxă: `xmlns="URI"`. A se vedea și **XML**.

spider (*spider*). Parte a unui motor de căutare, numită și *crawler* sau *robot*, este o piesă software care funcționează automat adresând simultan cereri către pagini ale unui număr mare de site-uri. Atunci când găsește legături într-o pagină, le urmează efectuând cereri către paginile corespunzătoare. *Spider*-ul trimite *indexer*-ului motorului de căutare informații despre toate paginile găsite. Acțiunile de navigare, colectare și indexare desfășurate de *spider* se numesc *spidering*.

Principalul *spider* al motorului de căutare *Google* este *GoogleBot*, care „scotocește” Web-ul o dată pe lună. Pentru a evidenția schimbările care se produc mai des în unele site-uri, *Google* utilizează *FreshBot*, care „scotocește” Web-ul constant. A se vedea și **motor de căutare**.

splash page (*pagină de pornire*). Pagină Web care apare înainte ca vizitatorul să intre propriu-zis pe un site. De multe ori, o astfel de pagină conține un mesaj de întâmpinare sau o animație *flash*. Multe persoane

consideră că paginile *splash* nu sunt utile din punctul de vedere al optimizării Web (SEO). De multe ori, motoarele de căutare nu pot indexa conținutul informațional al acestora.

splog (*splog*). Prescurtare pentru *spam blog*. Blog creat cu scopul de a promova site-uri Web afiliate sau de a crește *PageRank*-ul unor site-uri asociate. În mod obișnuit, un astfel de site conține un număr mare de legături către site-urile asociate cu creatorul blogului. A nu se confunda cu *spam in blog*. A se vedea și **PageRank** și **spam in blog**.

spyware (*spyware*). Program de tip *malware* care, odată instalat pe calculatorul utilizatorului, colectează – fără știința acestuia – informații despre el și despre obiceiurile sale (referitoare la utilizarea calculatorului), pe care le trimite unui destinatar prestabilit. Uneori, chiar administratorul unui calculator public instalează programe de tip *spyware* (e.g., *keyloggers*), pentru a monitoriza în secret utilizatorii. În vreme ce denumirea *spyware* sugerează numai monitorizarea utilizatorilor, scopurile programelor de acest tip au fost extinse la colectarea unor informații personale de diverse tipuri (e.g., obiceiurile de navigare pe Web, site-urile vizitate), dar și la realizarea altor operații (e.g., redirectarea browserului). A se vedea și **adware**, **keylogger** și **malware**.

SQL (*Standard Query Language*). Limbaj neprocedural utilizat pentru crearea, modificarea, regăsirea și manipularea datelor de către Sistemele de gestiune a bazelor de date relaționale (SGBDR), standard ANSI (*American National Standards Institute*) din 1986 și ISO (*International Organisation for Standardization*) din 1987. Primul standard ANSI a fost SQL-86. Alte versiuni sunt SQL-89, SQL-92, SQL:1999, SQL:2003, SQL:2006. Versiunea curentă a limbajului este SQL:2008.

Fiecare dintre producătorii SGBDR-urilor implementează limbajul SQL într-o manieră specifică, adăugând extensii proprii standardului. Din acest motiv, este posibil ca implementările SQL să fie diferite – mai mult sau mai puțin – de la un sistem la altul.

Limbajul SQL include instrucțiuni care permit efectuarea unor operațiuni diverse, dintre care pot fi menționate: crearea bazelor de date, a tabelelor și a structurii acestora; indexarea și stabilirea relațiilor între tabele; efectuarea de operații de actualizare a datelor (e.g., modificare, introducere, ștergere); regăsirea (consultarea) datelor, având ca rezultat transformarea acestora în informații; modificarea structurii tabelelor; controlul drepturilor de acces pentru utilizatori, asupra bazelor de date, tabelelor și coloanelor acestora.

O instrucțiune SQL este compusă din cuvinte rezervate și cuvinte definite de utilizator (parametri). Cuvintele definite de utilizator reprezintă denumirile

bazelor de date, ale tabelelor, câmpurilor etc. La rândul lor, cuvintele rezervate sunt cuvinte-cheie și clauze. Rezultă că o instrucțiune SQL are trei părți: un cuvânt-cheie, parametri și clauze. Atunci când se face referire la o instrucțiune, se precizează componenta cuvânt-cheie (e.g., instrucțiunea `SELECT`). Cuvintele rezervate sunt *case-insensitive*. În schimb, unele dintre cuvintele definite de utilizator sunt *case-sensitive* (e.g., numele coloanelor unui tabel), altele fiind *case-insensitive* (e.g., numele bazelor de date și ale tabelelor).

Limbajul SQL are următoarele două componente: un limbaj de definire a datelor (DDL, *Data Definition Language*), utilizat pentru definirea structurii bazei de date; un limbaj de manipulare a datelor (DML, *Data Manipulation Language*), utilizat pentru regăsirea, introducerea, modificarea/actualizarea și ștergerea datelor. Instrucțiunile SQL din limbajul de definire a datelor permit realizarea următoarelor operații (între paranteze sunt prezente cuvintele-cheie): crearea (`CREATE DATABASE`) și ștergerea (`DROP DATABASE`) bazelor de date; crearea (`CREATE TABLE`) și ștergerea (`DROP TABLE`) tabelelor; crearea indecșilor (`CREATE INDEX`) și ștergerea indecșilor (`DROP INDEX`); actualizarea structurii tabelelor (`ALTER TABLE`).

Unele instrucțiuni SQL (e.g., `SELECT`) creează baze de date și tabele temporare pentru a păstra rezultatul unor comenzi. Totuși, aceste instrucțiuni nu sunt incluse în limbajul de definire a datelor.

Instrucțiunile SQL din limbajul de manipulare a datelor (e.g., SELECT, INSERT, UPDATE, DELETE) permit realizarea următoarelor operații (între paranteze sunt prezente cuvintele-cheie): regăsirea (selectarea) datelor (SELECT); introducerea datelor (INSERT); modificarea datelor (UPDATE); ștergerea datelor (DELETE). Ultimele trei operații enumerate realizează ceea ce se numește *actualizarea datelor*.

Limbajul de manipulare a datelor include și facilități pentru efectuarea de calcule aritmetice, precum și diverse funcții pentru executarea unor operații, ca de exemplu numărare, însumare, calculul mediei, minimului, maximului, abaterii standard etc.

A treia componentă, neoficială, a SQL este limbajul de control al datelor (DCL, *Data Control Language*), care grupează instrucțiuni utilizate pentru administrarea conturilor utilizatorilor (e.g., GRANT, REVOKE). Standardul ANSI include aceste instrucțiuni în DDL (la fel ca și în această lucrare).

Limbajul SQL poate fi utilizat în două moduri: interactiv, prin introducerea comenzilor de la un terminal folosind un program client de tip linie de comandă; prin integrarea comenzilor în programe client scrise într-un limbaj procedural (e.g., PHP). O comandă SQL este o instrucțiune SQL care se termină cu caracterul „punct și virgulă”. În felul acesta pot fi trimise serverului mai multe comenzi, separate prin caracterul amintit. Aproape toate

comunicațiile cu un SGBDR sunt făcute prin intermediul comenzilor SQL. Excepțiile sunt reprezentate de situațiile în care serverul nu funcționează corect, fiind necesare alte instrumente pentru a-l depana. În cadrul unui client SQL de tip linie de comandă, comanda este transmisă serverului după apăsarea tastei <ENTER>.

SQL Injection (*SQL Injection*). Tehnică prin care un atacator creează sau schimbă comenzi SQL existente, prin utilizarea unei aplicații Web, folosind caractere speciale, în scopul de a obține acces neautorizat (prin ocolirea mecanismului de autentificare) la date stocate în baze de date și de a realiza citirea, adăugarea și modificarea acestora, precum și crearea unor noi conturi de utilizator. Realizarea unor atacuri de acest tip este de multe ori consecința unei insuficiente filtrări a datelor. Atacurile *SQL Injection* pot fi blocate printr-o filtrare corectă a acestora. Rezultă că atacul este posibil din cauza vulnerabilității aplicațiilor, neavând drept cauză probleme ale serverelor de baze de date sau ale serverelor Web. Efectele atacurilor *SQL Injection* depind de codul utilizat de atacator, de importanța datelor stocate în baza de date și de gradul de filtrare a acestora.

SRAM (*static random acces memory, memorie RAM statică*). Memorie RAM care utilizează câte un circuit bistabil pentru stocarea fiecărui bit. Circuitul include patru tranzistori, alți doi fiind

necesari pentru controlul accesului în timpul operațiilor de citire și scriere, în total fiind necesari șase tranzistori pentru fiecare bit. Memoria se numește statică deoarece – spre deosebire de memoria RAM dinamică – nu necesită împrăștiere periodică. A se vedea și **DRAM** și **RAM**.

SRE (*Standard for Robots Exclusion, standard pentru excluderea roboților*). Creat de Martijn Kostner în anul 1994, descrie un protocol de limitare a accesului roboților în anumite zone ale Web-ului (care conțin date temporare sau private). Respectarea standardului de către roboți nu este obligatorie. Dezvoltatorii de roboți respectă standardul în mod voluntar, iar acesta nu conține nici un mecanism real pentru a bloca accesul la zonele interzise. A se vedea și **robots.txt**.

SSL (*Secure Sockets Layer*). Protocol pe care se bazează tehnologia care asigură securizarea avansată a comunicațiilor în Internet. Fiind un protocol separat de HTTP, SSL poate fi utilizat de toate aplicațiile care folosesc TCP pentru transportul datelor. În orice tip de comunicație, în particular pe Web, clientul decide dacă să utilizeze sau nu SSL. În combinația dintre SSL și HTTP, schema URI „https:” indică o sesiune securizată. Deși schema menționată permite precizarea explicită a unui port TCP, este utilizat și un port implicit (*default*), anume 443 (diferit, după cum se poate constata, de cel folosit pentru comunicația HTTP standard, portul 80).

Cele mai multe browsere furnizează o modalitate prin care le comunică utilizatorilor că sesiunea de lucru este securizată. Aceasta constă în afișarea unui icon specific (*e.g.*, un lacăt) în colțul din dreapta jos al ferestrei. Versiunile mai noi ale browserelor (*e.g.*, începând cu *Firefox 2.0* și *Internet Explorer 7*) afișează și bara de adrese în culoare aurie. A se vedea și **criptografia cu cheie publică**.

Stallman, Richard (n. 1953). Programator american. În 1983 a început să scrie un sistem de operare compatibil UNIX, denumit GNU, oferit oricărei persoane care dorește să îl folosească. A publicat *GNU Manifesto* (1985), a înființat Free Software Foundation (1985) în sprijinul proiectului GNU, a creat licența GPL (GNU Public License). A se vedea și **FSF, Licența Publică Generală GNU, proiectul GNU și UNIX**.

standard de facto (*de facto standard*). Specificație utilizată pe scară foarte largă, deși nu a fost recunoscută ca standard de o organizație de standardizare (*e.g.*, ISO). A se vedea și **ISO**.

standard de jure (*de jure standard*). Specificație recunoscută ca standard de una dintre organizațiile de standardizare existente (*e.g.*, ISO). Pentru a deveni standard, o specificație trebuie să parcurgă o cale oficială destul de lungă. A se vedea și **ISO**.

standarde pentru e-Learning (*e-Learning standards*). Standarde pentru crearea și administrarea conținutului

învățării, toate având drept caracteristică de bază interoperabilitatea.

La începutul apariției mediilor virtuale de învățare, producătorii acestora își organizau conținutul cursurilor în modalități specifice, utilizând propriul format de împachetare. Din acest motiv, transferul cursurilor între diverse sisteme era practic imposibil. De asemenea, procedura de transfer către student prin intermediul protocoalelor existente (e.g., HTTP) era foarte greoaie. Pentru a înlătura neajunsurile amintite mai sus, s-au pus la punct o serie de standarde pentru *e-Learning*.

Dintre beneficiile utilizării unui sistem pentru predare și învățare creat pe baza standardelor, pot fi amintite: furnizează un *feedback* imediat, astfel încât profesorii și studenții pot acționa rapid în sensul eliminării deficiențelor și creșterii eficienței învățării; oferă acces la rapoarte care permit o mai bună măsurare a rezultatelor învățării; asigură suport pentru un număr mare de tipuri de instrumente pentru crearea conținutului didactic; determină reducerea costurilor necesare pentru implementarea învățării *online*; oferă posibilitatea includerii unui număr mare de materiale de învățare care să se constituie sub forma unei biblioteci digitale.

În procesul de elaborare a standardelor pentru *e-Learning*, există o serie de aspecte discutate pe larg, referitoare la: rolul specialiștilor educației în raport cu preocupările și interesele dezvoltatorilor de software pentru acest domeniu; modul în care abordarea și

utilizarea pedagogică pot fi reprezentate într-un standard tehnic; ce trebuie inclus în standarde (pe de o parte, pentru a fi utile, trebuie să includă specificații cuprinzătoare; pe de altă parte, cu cât specificațiile sunt mai detaliate, cu atât efortul necesar pentru a le respecta este mai mare).

Procesul creării standardelor pentru *e-Learning* este iterativ, fiind compus din următoarele patru etape: cercetare și dezvoltare (se caută identificarea unor soluții posibile la probleme apărute în activitate, având ca promotori universități, companii și consorții); dezvoltarea specificațiilor (atunci când se propune o soluție, se dezvoltă un proces de documentare și specificații scrise, care pot fi implementate și codificate; aceste operații sunt realizate în mod obișnuit de un grup de lucru, un consorțiu, o organizație sau prin colaborare între instituții, e.g., ARIADNE, AICC, IMS); testare și dezvoltare (testarea specificațiilor pe sisteme-pilot și determinarea schimbărilor pe baza reacțiilor utilizatorilor, e.g., ALIC, ADL); acreditare și recunoaștere internațională (specificațiile testate sunt luate în evidența unui organism acreditat de standardizare, unde sunt reanalizate și transformate în standarde globale; urmează un proces deschis, bazat pe consens, în urma căruia rezultă un proiect de standard supus votului; aprobarea lui înseamnă certificare oficială primită din partea organizațiilor internaționale acreditate, e.g., IEEE *Learning Technology Standards Committee*, ISO/IEC).

Standardele educaționale pentru *e-Learning* pot fi grupate în următoarele categorii: standarde pentru metadata (au ca scop etichetarea într-o modalitate consistentă a conținutului educațional și a cataloagelor, care să permită indexarea, stocarea, descoperirea și regăsirea lor folosind multiple instrumente și diverse depozite; pentru crearea acestor standarde sunt folosite câteva standarde dintre care pot fi amintite IEEE LOM și Dublin Core Metadata); standarde pentru împachetarea datelor (au ca scop definirea unor formate care să permită transferul conținutului educațional de la un sistem la altul; dintre aceste standarde pot fi menționate: *IMS Content Packaging*, *IMS Simple Sequencing* și *ADL SCORM*); standarde pentru descrierea profilului cursantului (au ca scop utilizarea informațiilor personale ale cursantului/studentului – *e.g.*, date personale, planuri de învățare, istoricul învățării, cerințe de accesibilitate, certificate și diplome, evaluări – în vederea furnizării conținutului educațional optim din punct de vedere didactic; cel mai important efort în direcția standardizării informațiilor referitoare la profilul utilizatorului este reprezentat de specificațiile *IMS LIP*); standarde pentru înregistrarea cursantului (au ca scop definirea unei modalități unitare de înregistrare a cursanților/studentilor, care să faciliteze atât interacțiunea acestora cu materialele educaționale, cât și lansarea conținutului; cele mai cunoscute inițiative sunt *IMS Enterprise Specification* și *School Inter-*

perability Framework); standarde pentru comunicare (odată ce conținutul este lansat, este necesar să i se comunice cursantului/studentului informații despre acesta; cel mai important standard este *ADL SCORM*). A se vedea și **SCORM**, **LOM** și **mediu virtual de învățare**.

standard deschis (*Open Standard*). Standard public ce poate fi utilizat în mod liber de oricine. Pentru acest concept sunt utilizate mai multe definiții, practic echivalente, în continuare fiind prezentată cea a Uniunii Europene: „un standard deschis este un document care are următoarele caracteristici: este adoptat și menținut de o organizație nonprofit; a fost publicat, specificațiile sale fiind complet libere; nu există constrângeri cu privire la reutilizarea sa; proprietatea intelectuală (*i.e.*, posibilele patente existente) este declarată irevocabil ca fiind valabilă fără drepturi de autor.”

Standardele deschise pot fi incluse în următoarele categorii: sistem (*e.g.*, arhitectura World Wide Web, specificată de Consorțiul Web); formate de fișiere (*e.g.*, text, multimedia); protocoale (*e.g.*, IP, TCP); limbaje de programare (*e.g.*, C#).

Utilizarea standardelor deschise are numeroase avantaje, dintre care pot fi amintite: independența de platformă, adică posibilitatea portării materialelor/documentelor (inclusiv a celor utilizate în domeniul educațional) de la un calculator la altul, pe care rulează sisteme de operare diferite; posibilitatea

conceperii și dezvoltării de materiale (inclusiv educaționale) care pot comunica și se pot interconecta fără restricții unele cu altele; conservarea și reutilizarea resurselor inițiale pe baza cărora au fost construite materialele.

Standardele deschise care specifică formate de date poartă denumirea de *formate deschise (Open Format)*. A se vedea și **format deschis**.

starea programului (program state). Concept-cheie utilizat în programarea calculatoarelor, este un „instantaneu” al unui program, conținând informații referitoare la execuția acestuia (e.g., conținutul stivei, instrucțiunea aflată în curs de execuție, conținutul memoriei). A se vedea și **program și programare**.

Start Point Directory. Fondat în 1995 de Frank Addante, este un director Web general editat manual și organizat pe teme (<http://www.stpt.com/>). A se vedea și **director Web**.

statement (instrucțiune). A se vedea **instrucțiune**.

Steve Crocker (n. 1944). Unul dintre pionierii Internetului, inventatorul seriei de documente RFC (*Request for Comments*), fiind și autorul primului dintre ele, intitulat *Host Software*. Acesta a fost publicat pe 7 aprilie 1969, Crocker lucrând atunci la UCLA (University of California, Los Angeles). A contribuit la crearea protocoalelor de comunicație în rețeaua ARPANET (precursorul Internetului). Pentru munca sa desfășurată în domeniul tehnologiilor Internet, a

primit în 2002 *IEEE Internet Award*. A se vedea și **ARPANET** și **RFC**.

stivă (stack). Structură de date de tip listă liniară simplu înlănțuită cu proprietatea că operațiile de introducere (*push*)/extragere (*pop*) a nodurilor se fac în/din coada listei (i.e., vârful stivei). Ultimul nod introdus va fi primul șters. Din acest motiv, stivele se mai numesc și liste LIFO (*Last In First Out, ultimul intrat, primul ieșit*) sau liste *pushdown*. Cel mai natural mod de reprezentare a unei liste este implementarea secvențială într-un tablou $S[1..n]$, unde n este numărul maxim de noduri. A se vedea și **listă înlănțuită**, **listă liniară** și **structură de date**.

stivă de protocoale (protocol stack). Implementare software particulară a unei suite de protocoale de rețea. Cei doi termeni sunt considerați adesea ca sinonimi, deși, strict vorbind, o suită include definițiile protocoalelor, iar o stivă, implementările acestora. Cele mai cunoscute stive de protocoale sunt ISO/OSI și TCP/IP. A se vedea și **ISO/OSI** și **TCP/IP**.

stocarea culorii (storage color). Modalitate de păstrare a culorii pentru imaginile în formate de tip *raster*. Aceste formate utilizează trei metode principale de stocare a culorii: *index-color*, *grayscale* și *truecolor*. Metoda *index-color* (stocarea indexată a culorilor), cunoscută și ca *pseudocolor*, *colormapped* sau *palette-based*, stochează o copie a valorii fiecărei culori necesare imaginii într-o paletă.

Imaginea principală este compusă din indecși, care au diverse intrări în paletă. De exemplu, dacă o imagine este compusă numai din pixeli având culorile roșu, galben și albastru, paleta va avea trei intrări (*i.e.*, 0, 1 și 2) corespunzătoare acestor culori. Astfel, fiecare pixel va putea fi reprezentat prin indecșii 0, 1 sau 2 și va necesita 2 biți pentru stocare, chiar dacă reprezentarea culorilor roșu, galben și albastru necesită câte 24 de biți fiecare.

În cadrul metodelor *grayscale* și *truecolor*, octeții utilizați pentru fiecare pixel corespund gradului de *gri* (*gray*) sau culorilor utilizate. Astfel, într-o imagine *grayscale*, valoarea 0 poate reprezenta negrul, 255 (valoarea maximă), albul, iar valorile intermediare (între 0 și 255), diverse grade de *gri*. Aceste valori intermediare sunt calculate prin interpolare, utilizând o metodă numită *corecție gama* (*gamma correction*), care are rolul de a elimina diferențele dintre diverse sisteme de afișare (*e.g.*, PC și Macintosh). Metoda *truecolor* utilizează trei valori separate pentru fiecare pixel, corespunzând „cantităților” de roșu (*red*), verde (*green*) și albastru (*blue*). Aceste culori alcătuiesc un spațiu de culoare (*color space*). Imaginile *truecolor* (având adâncimea de culoare de 24 de biți) sunt cunoscute, de asemenea, ca *imagini RGB*. A se vedea și **formate raster**.

strat (*layer*). În domeniul rețelelor de calculatoare, este un element de organizare a acestora. Straturile sunt

situate unul deasupra celuilalt. Deși numărul și funcțiile straturilor variază de la un tip de rețea la altul, scopul lor este comun, anume de a furniza servicii nivelurilor superioare.

În modul de organizare pe straturi, un nivel de pe o mașină (*i.e.*, calculator) comunică cu nivelul similar de pe altă mașină. Regulile și convențiile utilizate în comunicarea dintre un nivel al unei mașini cu nivelul similar al altei mașini alcătuiesc protocolul nivelului. O listă de protocoale utilizate de o mașină alcătuiește o stivă de protocoale.

În realitate, datele nu sunt transferate de pe un nivel al unei mașini pe nivelul similar al altei mașini, ci fiecare nivel transferă date și informații de control (numite și *headers*, *i.e.*, *antetură*) nivelului inferior. Procesul continuă până când se ajunge la nivelul cel mai de jos (nivelul fizic), aceasta asigurând comunicarea efectivă. Interfața definește operațiile și serviciile pe care un nivel le oferă nivelului superior. Mulțimea nivelurilor și protocoalelor utilizate într-o rețea de calculatoare definește o arhitectură de rețea. Cele mai importante arhitecturi de rețea sunt modelele de referință ISO/OSI și TCP/IP. Serviciile oferite de un nivel celui superior pot fi accesate de acesta din urmă în așa-numitele SAP-uri (*Service Access Point*, puncte de acces la servicii). Fiecare SAP este identificat în mod unic prin intermediul unei adrese. Folosind SAP-ul propriu, un nivel trimite celui superior date (SDU, *Service Data Unit*, unitate de date de

servicii) și informații de control, care alcătuiesc împreună o unitate de date de interfață (IDU, *Interface Data Unit*). Este posibil ca nivelul care trimite un SDU să fie obligat să-l fragmenteze în mai multe bucăți; fiecare dintre acestea primește un antet și este trimisă separat, sub forma unui PDU (*Protocol Data Unit*, unitate de date a protocolului). Serviciile pot fi orientate pe conexiuni (analog sistemului telefonic) și fără conexiuni (analog sistemului poștal). A se vedea și **ISO/OSI** și **TCP/IP**.

strategii de căutare (*search strategies*). Strategii generale folosite în căutarea informațiilor din Web cu ajutorul motoarelor de căutare. Acestea sunt: specificitate ridicată (deoarece Web-ul este un rezervor gigantic de informații, conținând miliarde de pagini Web, aproape orice căutare oferă un număr foarte mare de rezultate; din acest motiv, este necesar ca operația de căutare să fie cât mai specifică, numărul rezultatelor obținute reducându-se astfel considerabil); concizie sporită (cu cât termenii de căutare sunt mai preciși și concisi, cu atât relevanța rezultatelor returnate va fi mai ridicată); schimbarea ordinii termenilor căutați (atunci când se realizează o căutare, pentru motor este importantă și ordinea termenilor incluși în expresia de căutare, primul termen având o pondere mai mare decât ceilalți; din acest motiv, cel mai sugestiv termen al expresiei trebuie să fie plasat primul; de asemenea, este util să se efectueze căutări succesive schimbând ordinea termenilor în

expresia de căutare). A se vedea și **motor de căutare**.

stratul de abstractizare a hardware-ului (HAL, *hardware abstraction layer*). În sistemele de operare, strat (incluzând cod în limbaj de asamblare) între hardware și software-ul care rulează pe calculator. Are rolul de a ascunde diferențele existente în hardware de *kernel*-ul sistemului de operare, astfel încât acesta să poată rula pe diverse sisteme având configurații hardware diferite. Este folosit de programatori pentru a scrie aplicații independente de dispozitiv. A se vedea și **hardware** și **software**.

streaming (*cu livrare continuă*). Metodă de livrare a informațiilor (audio și video) distribuite prin intermediul rețelelor de telecomunicații, caracterizată prin existența unui flux constant de date livrate unui utilizator. Televiziunea prin Internet este un exemplu de *streaming media*. A se vedea și **streaming media**.

streaming media (*media cu livrare continuă*). Metodă de livrare a conținutului multimedia în rețelele de telecomunicații. În general, conținutul multimedia are un volum ridicat, astfel încât stocarea și transmisia se fac cu costuri importante. Pentru reducerea costurilor, conținutul este comprimat. Acesta poate fi livrat sub forma unui flux media (*stream media*), la comandă (*on demand*) sau în direct (*live*). În primul caz, conținutul trebuie păstrat pe server, în timp ce în al doilea caz acesta

este disponibil numai în momentul transmisiei (e.g., un program TV). A se vedea și **streaming**.

Stroustrup, Bjarne (n. 1950). Programator danez, creatorul C++ și cel care a realizat prima implementare a limbajului (<http://www.research.att.com/~bs/>). Este autorul cărții *The C++ Programming Language*, binecunoscută programatorilor. A se vedea și **C++**.

structură arborescentă (*tree structure*). Structură care are proprietățile organizatorice ale unui arbore. A se vedea și **arbore**.

structură de control (*control structure*). În programarea structurată, porțiune de program definită de relațiile dintre instrucțiuni. În anul 1966, Böhm și Jacopini au demonstrat că orice program se poate scrie utilizând numai trei tipuri de structuri de control: secvențială (sau liniară), alternativă și repetitivă condiționată anterior. Pe lângă cele trei tipuri amintite, orice limbaj de nivel înalt implementează și alte structuri de control, introduse în scopul creșterii eficienței activității de programare. Acestea sunt structura repetitivă condiționată posterior și structura selectivă (care este o structură alternativă generalizată). Structurile enumerate (cu excepția celei secvențiale) pot fi grupate în următoarele două categorii: structuri de control condiționale (controlează fluxul programului, executând sau nu anumite porțiuni de cod, în funcție de îndeplinirea sau neîndeplinirea unor condiții); structuri de control repetitive

(execută anumite porțiuni de cod de un număr determinat de ori). Cele două categorii de structuri de control sunt implementate în limbajele de programare ca instrucțiuni condiționale și, respectiv, repetitive. A se vedea și **instrucțiune de control**.

structură de date (*data structure*). Ansamblu de date caracterizat prin relațiile existente între acestea și de operații care pot fi efectuate cu datele respective. Reprezintă o modalitate de stocare și organizare a datelor astfel încât acestea să fie utilizate cât mai eficient. În mod obișnuit, ansamblul de date care alcătuiesc structura este o mulțime cu un număr variabil de noduri (numite și articole, entități sau înregistrări). Nodul este o variabilă de un tip oarecare, de obicei structurat. Cele mai cunoscute structuri de date sunt cele elementare: tabloul, lista (cu cazurile particulare: stiva și coada), grafurile și arborele (cu cazul particular: heap). Tabloul și lista sunt denumite structuri liniare de date. A se vedea și **arbore**, **coadă**, **graf**, **heap**, **listă**, **stivă** și **tablou**.

Study in the USA. Site care oferă ajutor studenților care doresc să studieze în Statele Unite ale Americii (<http://www.studyusa.com/>).

Site-ul pune la dispoziția utilizatorilor numeroase opțiuni de învățare, oferind un număr important de servicii pentru programul de studiu vizat. Pentru fiecare opțiune de învățare, este furnizată o listă de școli și universități de pe teritoriul SUA. De asemenea, pentru

fiecare zonă este oferită o listă cu subzonele ei și cu școlile care se găsesc în zona respectivă. În cadrul site-ului este implementat un motor de căutare pentru găsirea programului dorit de utilizator, acesta constituind un sprijin deosebit de util pentru cei care doresc să studieze în Statele Unite.

subdirector (*subdirectory*). Director inclus într-un alt director. A se vedea și **director**.

submeniu (*submenu*). Meniu inclus într-un meniu de nivel înalt (vizibil numai după selectarea unei opțiuni incluse în acesta din urmă). A se vedea și **meniu**.

subrețea (*subnetwork*). Include calculatoarele care au același identificator de subrețea (numit și prefix de rutare), acesta fiind o secvență de biți care precede partea de gazdă a adresei IP.

Toate gazdele dintr-o rețea trebuie să aibă același identificator de rețea. Dacă – de exemplu – rețeaua este de clasă C, creșterea numărului de calculatoare peste 254 necesită crearea altor rețele, fiecare având un identificator de rețea. Astfel, se poate ajunge la mai multe LAN-uri interconectate, fiecare cu propriul său identificator. Această modalitate prezintă probleme legate de obținerea identificatorilor de rețea și de mutarea unei mașini dintr-un LAN în altul. Pentru evitarea problemelor menționate, se utilizează divizarea unei rețele în mai multe subrețele, dar care să fie văzute din exterior ca o singură rețea. Utilizarea subrețelelor determină

eficientizarea alocării adreselor IP. Pentru a diviza o rețea în subrețele, se împrumută pentru rețea un număr de biți din cei rezervați identificatorului de gazdă. Altfel spus, numărul de biți rezervați gazdei se împarte într-un identificator de subrețea și unul de gazdă, rezultând că o adresă IP are următoarea structură:

adresa IP = <id rețea><id subrețea><id gazda>

Combi-nația dintre identificatorul de subrețea și cel de gazdă este cunoscută sub numele de *adresă locală* (sau *parte locală*). Împărțirea unei rețele în subrețele este transparentă pentru alte rețele, ceea ce înseamnă că oricare gazdă din aceste rețele va trata adresa locală a unei gazde dintr-o subrețea ca pe un identificator de gazdă.

Numărul maxim de subrețele se calculează ca $2^p - 2$, unde p este numărul de biți alocați pentru subrețea (e.g., $p=6$, $2^6 - 2 = 64 - 2 = 62$). Numărul maxim de adrese asignabile unei subrețele se calculează ca $2^n - 2$, unde n este numărul de biți alocați fiecărei gazde (e.g., $n=10$, $2^{10} - 2 = 1024 - 2 = 1022$). De exemplu, dacă în cazul unei adrese de clasă B cei 16 biți (2 octeți) rezervați gazdei se împart în două, primii 6 fiind alocați identificatorului de subrețea, iar ultimii 10 identificatorului de gazdă, se pot obține 62 de LAN-uri (nu se folosesc prima și ultima subrețea, după cum se va vedea în continuare), fiecare având cel mult 1022 de gazde (nu se folosesc, după cum se va vedea, prima și ultima adresă IP).

Masca de subrețea este un număr pe 32 de biți și reprezintă filtrul care determină cărei subrețele îi aparține o adresă IP. Astfel, în urma unui ȘI logic cu o adresă IP, masca va separa identificatorul de rețea, anulând biții corespunzători identificatorului de gazdă. Fiecare bit din masca de subrețea care se află pe aceeași poziție cu un bit din identificatorul de rețea (și subrețea) va avea valoarea 1, în vreme ce toți biții corespunzători identificatorului de gazdă vor avea valoarea 0. Cel mai des utilizată reprezentare pentru măștile de subrețea este cea zecimală, în concordanță cu cea folosită pentru adresele IP. Măștile de subrețea mai pot fi reprezentate sub forma unui număr care stabilește numărul de biți cu valoarea 1 incluși în mască, acesta fiind denumit *prefix de rețea*. De asemenea, adresele IP se mai pot scrie sub forma IP/n, unde n reprezintă prefixul de rețea. De exemplu, 192.168.100.2/29, care este un IP de clasă C, va avea masca de subrețea 11111111.11111111.11111111.11111000, adică 255.255.255.248. Deci acest IP face parte dintr-o rețea cu 30 de subrețele (2^5-2), identificatorul de subrețea conținând 5 biți. Fiecare subrețea conține câte 6 IP-uri alocabile (2^3-2), deoarece identificatorul de gazdă include 3 biți. Rețeaua conține un număr total de 180 de IP-uri alocabile ($30*6$).

În urma creării de subrețele, se pierde o parte din spațiul de adrese inițial. De exemplu (pentru o rețea de clasă C), în cazul utilizării unui număr de doi biți

pentru subrețea, numărul de biți alocați gazdei va fi 6 (*i.e.*, 8-2), numărul de subrețele care pot fi construite va fi 2 (*i.e.*, 2^2-2), numărul gazdelor care pot fi incluse într-o subrețea va fi 62 (*i.e.*, 2^6-2), numărul total de gazde va fi 124 (*i.e.*, $2*62$), procentul de adrese IP de gazdă utilizabile fiind de aproximativ 49% (*i.e.*, 124/255). Gradul de utilizare a unui spațiu de adrese de clasă C, folosind subrețele, nu poate depăși 77%. A se vedea și **adresă IP și rețea de calculatoare**.

subrutină (*subroutine*). În programare, porțiune de cod dintr-un program, care efectuează o sarcină specifică și este relativ independent de restul programului. Termenul este înrudit cu alți termeni, ca funcție, procedură și rutină. A se vedea și **funcție, procedură și rutină**.

subscribe (*subscriere, abonare*). Abonarea la o sursă de informare electronică (*e.g.*, *e-newsletter*). Se face, de obicei, prin furnizarea unei adrese de *e-mail*. De multe ori subscrierea nu este făcută de utilizator, ci de sisteme automate, fără acordul acestuia, pe baza adresei de *e-mail*, găsită într-o bază de date publică sau procurată prin alte mijloace. A se vedea și **unsubscribe**.

subscript (*indice*). Caracter scris mai jos de linia de bază a textului din jurul lui, având dimensiunea mai mică decât acesta. A se vedea și **superscript**.

subsol (*footer*). Referitor la o pagină, zonă aflată în partea inferioară a acesteia, în care pot fi incluse note de

subsol, numărul paginii și alte informații. A se vedea și **antet**.

suma de control (*checksum*). Numită și *hash sum*, este o valoare calculată cu ajutorul căreia se verifică apariția unor erori accidentale în cursul transmisiilor de date sau scrierii pe disc. Integritatea datelor poate fi verificată oricând, calculând suma de control și comparând-o cu valoarea stocată. Dacă valorile respective nu coincid, datele au fost alterate. Totuși, pentru evitarea modificărilor intenționate ale datelor, se vor folosi funcțiile criptografice. A se vedea și **cifra de control**.

Sun Microsystems, Inc.. Companie multinațională cu sediul în SUA, producător și distribuitor de calculatoare, software și servicii informatice (<http://www.sun.com/>). A fost fondată în anul 1982 de patru studenți de la Stanford University: Vinod Khosla, Bill Joy, Scott McNealy și Andy Bechtolsheim. Numele inițial al companiei era SUN (*Stanford University Network*). În domeniul software este cunoscută mai ales pentru sistemul de operare Solaris, platforma de programare Java și suita de aplicații de birou *StarOffice*, din care a fost derivat pachetul *open source OpenOffice.org*. În februarie 2008, a achiziționat MySQL AB, dezvoltatorul SGBDR MySQL. În aprilie 2009, Sun Microsystems a fost cumpărată de Oracle Corporation. A se vedea și **OpenOffice.org**, **Solaris** și **Joy, William Nelson**.

supercalculator (*supercomputer*). Calculator cu performanțe deosebite, în special o putere de procesare foarte mare. Supercalculatoarele au fost construite începând cu anii 1960, un pionier al acestui domeniu fiind Seymour Cray. Astăzi, supercalculatoarele sunt construite de companii din SUA și Japonia: Cray Research, Fujitsu, IBM, Sun, SGI (*Silicon Graphics, Inc.*), Hewlett-Packard, Sutherland și ETA Systems.

În anul 1980 cel mai rapid supercalculator lucra cu circa un milion de operații în virgulă mobilă pe secundă (FLOPS, *Floating Point Operations Per Second*). În 1990 vitezele au ajuns la circa 1 Giga-FLOPS, estimându-se ca în 2010 viteza să ajungă la 1000 Tera-FLOPS.

Supercalculatoarele sunt folosite în aplicații complexe din domenii ca: reactoare nucleare, simulări nucleare, proiectarea aeronavelor, exploatare petroliere, seismologie, criptografie, prognoza vremii. A se vedea și **Cray, Seymour Roger**.

superscript (*exponent*). Caracter scris mai sus față de textul din jurul lui, având dimensiunea mai mică decât acesta. A se vedea și **subscript**.

superuser (*superutilizator*). A se vedea **superutilizator**.

superutilizator (*superuser*). Utilizator care, în sistemele de operare moderne, deține un cont cu privilegii totale. Acesta este de obicei administratorul de sistem. A se vedea și **administrator de sistem** și **cont-rădăcină**.

SuprGlu. Serviciu pentru personalizarea paginii de start (<http://www.suprglu.com/>). A se vedea și **serviciu pentru personalizarea paginii de start**.

sursa documentului (*document source*). A se vedea **sursa paginii**.

sursa paginii (*page source*). Documentul HTML asociat paginii Web curente (afișată în browser). Acesta poate fi vizualizat folosind navigatorul Web (e.g., *View→Page Source*, în *Firefox*).

SVG (*Scalable Vector Graphics*). Limbaj bazat pe XML (*eXtensible Markup Language*) utilizat pentru descrierea textuală a conținutului grafic vectorial bidimensional. Cu alte cuvinte, documentele SVG conțin text care definește imagini.

Pentru crearea conținutului SVG destinat dispozitivelor mobile se utilizează Mobile SVG Profiles: SVG Tiny (pentru telefoanele mobile multimedia) și SVG Basic (pentru PDA-uri/dispozitive *handheld*). Acestea sunt părți ale platformei 3GPP (*The 3rd Generation Partnership Project*), pentru a treia generație de dispozitive mobile (<http://www.3gpp.org/>). Telefoanele mobile care rulează aplicații SVG sunt deja folosite de zeci de milioane de utilizatori, majoritatea din Europa.

Fiind un limbaj bazat pe XML, SVG poate fi creat și transformat prin intermediul XSLT (*eXtensible Style Sheet Transformations*). SVG poate fi

utilizat, ca format vectorial, pentru a produce imagini statice scalabile (spre deosebire de imaginile în format *bitmap*), dar și grafică interactivă și dinamică (animată).

Animația SVG poate fi definită și declanșată declarativ (*i.e.*, incluzând elemente de animație în conținutul SVG) sau prin intermediul limbajului ECMAScript, acesta din urmă constituind baza standardizată a limbajelor JavaScript/JSript. Este posibilă realizarea unor aplicații SVG complexe prin utilizarea limbajului ECMAScript, care accesează SVG DOM (*Document Object Model*). Prin intermediul SVG DOM pot fi accesate toate elementele SVG, precum și atributele și proprietățile acestora.

SVG permite utilizarea a trei tipuri fundamentale de obiecte grafice: figuri (forme) bazate pe grafică vectorială (compuse din linii drepte și curbe), imagini *raster* și text. Obiectele grafice pot fi grupate, stilizate, transformate și incluse în obiecte mai complexe. Limbajul poate fi integrat cu tehnologiile pentru server (e.g., *servlet*-uri, JSP, ASP, PHP, Perl).

Pentru vizualizarea graficii SVG este necesară instalarea unui *plugin* în unele browsere. Pentru *Internet Explorer* este recomandabil să se utilizeze *Adobe SVG Viewer* (<http://www.adobe.com/svg/viewer/install/main.html>). Acesta poate fi utilizat și cu *Mozilla Firebird* (pentru versiuni anterioare 1.1). Începând cu versiunea 1.1, browserul *Mozilla Firefox* include un vizualizator SVG

propriu. De asemenea, browserul *Opera* include un vizualizator propriu.

Dintre domeniile în care este utilizat limbajul SVG pot fi menționate: platformele mobile (SVG Mobile este formatul grafic utilizat de a treia generație de telefoane mobile pentru mesaje multimedia); tipărire și arhivare (SVG Print este un format utilizat pentru generarea de conținut potrivit pentru tipărire și arhivare); realizarea interfețelor GUI pentru aplicațiile Web; crearea unor elemente componente ale aplicațiilor Web (e.g., grafice, logo-uri și imagini diverse); vizualizarea datelor (e.g., prin intermediul graficelor); proiectare industrială (aplicații CAD); GIS și cartografiere. A se vedea și **GIS** și **GUI**.

SVG-edit. Editor grafic SVG *online* care folosește numai JS, HTML și CSS (adică nicio funcționalitate care să țină de server), furnizat sub licență Apache 2.0 (<http://code.google.com/p/svg-edit/>). Programul poate fi utilizat folosind orice browser modern. A se vedea și **licența Apache** și **SVG**.

swap (*schimb*). 1. Mutarea unor porțiuni de program sau de date din memoria RAM pe hard-disk și invers. 2. Fișier aflat pe hard-disk în care sistemul de operare stochează porțiuni ale fișierelor de programe și date care nu încap în memoria RAM. Când anumite porțiuni de program și/sau date aflate în fișierul *swap* sunt necesare, sistemul de operare le aduce în memorie, le utilizează și le mută înapoi. Se recomandă ca fișierul *swap* să fie de

două ori mai mare decât memoria RAM. A se vedea și **hard-disk** și **RAM**.

SWF (*ShockWave Format*). Format grafic vectorial, proprietar Adobe Systems. Utilizează ActionScript – limbaj de programare bazat pe ECMAScript – pentru crearea animațiilor complexe. A se vedea și **ActionScript**, **Flash** și **formate vectoriale**.

Symantec Corporation. Companie producătoare de software, cu sediul în SUA, fondată în anul 1982 de Gary Hendrix (cu un grant de la NFS) și specializată în produse de securitate informatică (<http://www.symantec.com/>).

Iată câteva dintre aplicațiile produse de Symantec: *Norton Antivirus*, *Norton Internet Security*, *Norton SystemWorks*, *Norton Ghost*, *Partition Magic*, *Norton Online Backup* și *Norton pcAnywhere*.

Symbian Foundation. Organizație non-profit fondată în 2008 de o serie de firme de telecomunicații și producătoare de dispozitive mobile (e.g., Nokia, Motorola, Sony-Ericsson, Vodafone, Samsung, LG) cu scopul de a administra sistemul de operare pentru telefoane mobile Symbian OS (<http://www.symbian.org/>). A se vedea și **Symbian Ltd.** și **Symbian OS**.

Symbian Ltd. Companie de software, cunoscută mai ales datorită sistemului de operare pentru telefoane mobile Symbian OS (<http://www.symbian.com/>). A fost înființată în 1998 ca un parteneriat între companiile Nokia, Ericsson,

Motorola și Psion. Zece ani mai târziu a fost cumpărată de Nokia. A se vedea și **Symbian OS**.

Symbian OS. Sistem de operare pentru telefoane mobile, dezvoltat de compania Symbian Ltd. A se vedea și **Symbian Ltd**.

Symfony. Cadru de lucru pentru proiectele PHP 5. Învățarea acestuia este relativ simplă pentru programatorii PHP care cunosc șabloanele de proiectare ale aplicațiilor Web (<http://www.symfony-project.org/>).

Școala Online. Proiect care își propune să ofere o bază de date cu resurse gratuite utile studenților, elevilor, profesorilor, dar și altor persoane

interesate să se instruiască sau să se informeze (<http://www.e-scoala.ro/>). Site-ul este o platformă deschisă, oricine putând să contribuie la dezvoltarea lui.

șir de caractere (*string*). În programare, tip de date format dintr-o mulțime ordonată de caractere (text) care, de obicei, poate fi citită de om (e.g., „bună ziua”), fiecare dintre caractere fiind identificat prin poziția sa în cadrul mulțimii. În mod uzual, un șir de caractere este inclus între ghilimele. A se vedea și **tip de date**.

șir de căutare (*search string*). Șir alfanumeric utilizat pentru comparație într-un proces de căutare.

T

tabel (*table*). 1. În domeniul bazelor de date, formă de organizare a datelor, denumită și *relație*. Conține coloane (*columns*) sau atribute (*attributes*). O înregistrare (*record*) este un rând (*row*) dintr-un tabel. La intersecția unui rând cu o coloană se află o celulă. Deși, teoretic, ordinea în care înregistrările sunt incluse în tabel nu are nici o importanță, practic aceasta poate afecta eficiența accesării lor. Mulțimea de valori pe care le poate lua un atribut constituie domeniul atributului. Un atribut poate avea: o singură valoare, valori multiple sau valori derivate (care depind de valoarea altui atribut).

Un tabel este caracterizat prin grad (numărul de coloane ale tabelului; cel cu două atribute se numește *binar*, cel cu trei atribute se numește *ternar* etc.; în general, un tabel cu *n* atribute se numește *n-ar*) și prin cardinalitate (numărul de înregistrări conținute în tabel; se modifică pe măsură ce sunt adăugate sau șterse înregistrări). Intensitatea unei relații (a unui tabel) este compusă din structură (*i.e.*, atribute), domeniile atributelor (împreună cu alte restricții asupra valorilor posibile) și grad. Ordinea în care apar coloanele

într-un tabel nu este importantă. Chiar dacă aceasta se schimbă, relația rămâne neschimbată, păstrând același înțeles. A se vedea și **bază de date relațională**.

2. Modalitate de organizare a informațiilor în paginile Web. Conținutul unui tabel este împărțit în rânduri (*rows*) și coloane (*columns*). La intersecția unei linii cu o coloană se găsește o celulă (*cell*). O celulă poate include tipuri variate de conținut (text, imagini, legături), existând posibilitatea precizării unor opțiuni legate de culoarea de fundal, aliniere etc.

În (X)HTML, un tabel este creat folosind marcajul `nevid <table>`. Atributele specifice acestuia sunt: `border` (specifică lățimea bordurii din jurul celulelor tabelului, exprimată în pixeli; *e.g.*, `border="2"`); `cellpadding` (precizează spațiul, exprimat în pixeli, dintre bordura unei celule și conținutul acesteia; *e.g.*, `cellpadding="2"`); `cellspacing` (stabilește spațiul, exprimat în pixeli, dintre bordura unei celule și cea a tabelului sau a altei celule; *e.g.*, `cellspacing="2"`); `frame` (permite specificarea laturilor din chenarul tabelului care vor fi vizibile); `width` (precizează lățimea dorită a tabelului, exprimată în pixeli – *e.g.* `width="600"` – sau ca procente din dimensiunea orizontală a elementului-părinte, *e.g.*, `width="50%"`).

Un rând al tabelului este creat folosind elementul `tr`. Acesta trebuie să conțină cel puțin unul dintre elementele `th` (crează o celulă a antetului, pentru

afișarea conținutului acesteia fiind utilizat automat stilul *bold*) sau `td` (creează o celulă care conține date). Elementul `tr` trebuie să apară cel puțin o dată într-un tabel. Pentru elementul `td` mai pot fi utilizate atributele `colspan` (precizează numărul de coloane peste care se întinde celula curentă) și `rowspan` (are ca valoare numărul de rânduri peste care se întinde celula curentă), precum și atributele pentru alinierea conținutului. Acestea din urmă pot fi utilizate și pentru elementul `tr`. De asemenea, pentru elementele `tr` și `td` mai pot fi specificate atributele comune elementelor XHTML, precum și `bgcolor`. A se vedea și **HTML**.

Tablet PC. Calculator mobil echipat cu *touchscreen* sau cu *tabletă grafică*, componente care permit operarea cu ajutorul unui creion digital, acesta înlocuind *mouse*-ul și tastatura. În particular, Tablet PC se referă la produsul (anunțat de Microsoft în anul 2001) care rulează sistemul de operare Windows Tablet PC Edition.

tabletă grafică (*graphics tablet*). Numită și *tabletă de digitizare* (*digitizing tablet*), este un dispozitiv de intrare care permite desenarea manuală a figurilor și introducerea automată a acestora în calculator. Constă dintr-o suprafață plană pe care se poate desena folosind un creion special, precum și din circuite traductoare care transmit calculatorului poziția acestuia.

tablou (*array*). Colecție finită de elemente de același tip (numit tip de

bază al tabloului), care ocupă o zonă continuă de memorie. Fiecare dată (numită și element) este referită prin intermediul unor indecși. Tablourile pot fi unidimensionale (în acest caz, ele se numesc și vectori) sau multidimensionale (des folosite fiind cele bidimensionale). În primul caz, fiecare element este referit prin intermediul unui singur index. În cazul tablourilor cu mai multe dimensiuni, elementele sunt referite folosind doi sau mai mulți indecși (ansamblul acestora alcătuind o matrice sau un tensor). În mod uzual, indecșii sunt numere întregi (începând cu 0), dar – în cazul anumitor limbaje de programare (e.g., PHP) – pot avea și alte tipuri (e.g., șir de caractere).

Pentru accesarea unui element inclus într-un tablou se utilizează notația `T[i]` pentru tablourile unidimensionale și `T[i,j]` pentru cele bidimensionale, unde `i` și `j` sunt indecși. A se vedea și **structură de date**.

tablou superglobal (*superglobal array*). Variabilă globală automată (sau superglobală) predefinită de tip tablou, utilizată în limbajul de programare PHP. Variabilele de acest tip sunt următoarele: `$GLOBALS` (conține toate variabilele globale utilizate în scriptul curent), `$_SERVER` (include valorile variabilelor setate de server sau pe cele direct legate de mediul de execuție al scriptului curent), `$_GET` și `$_POST` (conțin toate datele GET și, respectiv, POST primite în scriptul PHP de la browser), `$_FILES`, `$_COOKIE` (include toate *cookie*-urile primite în scriptul

PHP de la browser), `$_SESSION` (include variabile înregistrate în sesiunea curentă), `$_REQUEST` (conține toate variabilele GET, POST și COOKIE primite în scriptul PHP de la browser) și `$_ENV` (conține variabile furnizate scriptului de mediul de execuție). A se vedea și **PHP și variabilă globală**.

tag (marcaj). 1. Folosit în contextul marcării resurselor Web, este un cuvânt-cheie relevant sau un termen asociat cu/atribuit unei piese de informație (e.g., imagine, videoclip, pagină Web, post al unui blog) pentru a o descrie, oferind și posibilitatea includerii sale într-un sistem de clasificare/sortare. Un *tag* (denumit și *etichetă*) poate descrie conținutul sau natura unei resurse, scopul în care poate fi utilizată, locația sa etc. *Tag*-urile pot fi cuvinte individuale (e.g., *educatie*) sau expresii/propoziții/fraze (e.g., *educatia permanenta*). Utilizatori diferiți pot utiliza șabloane diferite (personale) de marcare a resurselor (e.g., *tag*-urile unor utilizatori pot fi mai expresive, în vreme ce *tag*-urile altor utilizatori pot fi mai descriptive). Deoarece operația de marcare este practică liberă de constrângeri (cu excepția numărului de *tag*-uri care pot fi asociate unei resurse și a numărului maxim de caractere ale unei etichete), un *tag* pot fi orice termen/asociație de termeni, reprezentând în ultimă instanță un indicator personal asociat unei resurse (altfel spus, o definiție *ad-hoc* a acesteia).

În funcție de sistemul de marcare utilizat, *tag*-urile formate dintr-un singur cuvânt trebuie delimitate prin

virgulă sau spațiu. De asemenea, *tag*-urile compuse din mai multe cuvinte vor fi încadrate între ghilimelele duble. În afară de *tag*-urile create în momentul în care efectuează marcarea unei resurse, utilizatorul mai are la dispoziție *tag*-uri create de alte persoane, propriile *tag*-uri (create anterior), precum și *tag*-uri generate de sistem (în funcție de natura resursei). Unei resurse date îi pot fi asociate simultan un număr mare de *tag*-uri (e.g., în *Flickr* numărul maxim este 75), utilizatorii fiind complet liberi în alegerea lor. Totuși, atunci când există posibilitatea partajării *tag*-urilor, utilizatorii vor alege dintre cele existente. A se vedea și **marcarea resurselor**. 2. În limbajele de marcare (e.g., HTML, SGML), cod care identifică un element al unui document (e.g., tabel, imagine, listă, titlu, formular). A se vedea și **marcaj**.

TagCloud (*nor de tag-uri*). Text, imagine sau animație *flash* prin care se realizează explorarea vizuală a *tag*-urilor (marcajelor) și a relațiilor dintre acestea. Într-un astfel de nor, *tag*-urile sunt afișate în funcție de frecvența de apariție, cu ajutorul culorilor sau fonturilor de mărimi diferite. De exemplu, în *delicious* pot fi create trei tipuri de nori: personali (includ numai *tag*-urile unui utilizator), locali (includ *tag*-urile unei comunități de utilizatori) și globali (includ toate *tag*-urile *delicious*). A se vedea și **delicious**.

Tanenbaum, Andrew (n. 1944). Informatician american, profesor la Universitatea Vrije din Amsterdam

(Olanda), unde predă cursuri de sisteme de operare și arhitectura calculatoarelor. Este creatorul sistemului de operare demonstrativ Minix (clonă a UNIX-ului), creat în scop educațional. Este cunoscut datorită cărților sale cu un pronunțat caracter didactic (e.g., *Structured Computer Organization*, *Computer Networks*, *Modern Operating Systems*). A se vedea și **Minix** și **UNIX**.

Taptu. Companie și serviciu de căutare pentru dispozitive mobile: telefoane și PDA-uri (<http://taptu.mobi/>). Compania a fost fondată în 2005 (Cambridge, Marea Britanie), iar serviciul de căutare în 2008. A se vedea și **motor de căutare**.

tasta Alt (*Alt key*). Tastă-modificator utilizată pentru a schimba alternativ funcția altor taste apăsată. Cel mai cunoscut mod de utilizare a tastei Alt constă în apăsarea sa simultan cu tastele CTRL și Delete. De asemenea, apăsată simultan cu tasta TAB, permite schimbarea ferestrei active. A se vedea și **CTRL-Alt-Delete**.

tasta Backspace (*Backspace key*). Tastă care, în cazul calculatoarelor IBM PC, mută cursorul cu un caracter la stânga, ștergând de obicei caracterul respectiv.

tasta Caps Lock (*Caps Lock key*). Tastă cu două stări: activă, în care determină scrierea de la tastatură cu litere mari, și inactivă. În stare activă nu afectează numerele, semnele de punctuație și celelalte caractere non-literale.

Pentru a fi adusă în stare activă, trebuie apăsată. Rămâne în această stare până când este apăsată din nou.

tasta CTRL (*CTRL key*). Tastă-modificator a cărei denumire provine de la cuvântul *control*. Apăsată împreună cu alte taste, determină execuția unor operații speciale (e.g., CTRL-Alt-Delete). A se vedea și **CTRL-Alt-Delete**.

tasta Delete (*Delete key*). Tastă a cărei funcție (în cazul calculatoarelor compatibile IBM PC) se schimbă în funcție de aplicația utilizată. În mod obișnuit, șterge caracterul aflat în dreapta cursorului sau imaginea selectată. De asemenea, apăsarea tastei determină ștergerea textului selectat.

tasta End (*End key*). Tastă care, în cazul unui document needitabil (e.g., pagină Web), este utilizată pentru defilare la sfârșitul acestuia. Dacă documentul este editabil (e.g., realizat cu aplicația *Word*), apăsarea tastei trimite cursorul la sfârșitul rândului curent; pentru a obține defilarea la sfârșitul documentului, tasta End trebuie apăsată împreună cu tasta CTRL. A se vedea și **tasta Home**.

tasta Enter (*Enter key*). Tastă utilizată în general pentru a determina execuția unei comenzi. Reprezintă o alternativă uzuală la apăsarea butonului *Ok*. În programele de prelucrare a textului (e.g., *Word*) este utilizată pentru trecerea la un nou paragraf.

tasta Escape (*Escape key*). Numită și **ESC** (*Extra Services Control*), este o tastă utilizată pentru inițierea unei secvențe *escape*. Are ca efect ieșirea din programul curent sau revenirea la nivelul anterior într-o structură de meniuri. În cazul browserelor Web, utilizarea tastei ESC are același efect ca și execuția unui click pe butonul *Stop*, însemnând oprirea încărcării paginii Web. În calculatoarele pe care rulează Microsoft Windows, utilizarea tastei ESC are același efect ca și folosirea butoanelor *No*, *Quit*, *Exit*, *Cancel* sau *Abort* în casetele de dialog. Pe aceleași calculatoare, apăsarea simultană a tastelor ESC și CTRL determină deschiderea meniului *Start*.

tasta Home (*Home key*). Tastă care, în cazul unui document neditabil (*e.g.*, pagină Web), este utilizată pentru întoarcerea la începutul acestuia. Dacă documentul este editabil (*e.g.*, realizat cu aplicația *Word*), apăsarea tastei trimite cursorul la începutul rândului curent; pentru a obține întoarcerea la începutul documentului, tasta Home trebuie apăsată împreună cu tasta CTRL. A se vedea și tasta **End**.

tasta Insert (*Insert key*). Tastă etichetată de obicei *Ins*, având (pe calculatoarele compatibile IBM PC) funcția de comutare între modul de inserare și cel de suprascriere (în care, în timpul scrierii, sunt suprascrise caracterele aflate în dreapta poziției curente a cursorului) în timpul editării textelor.

tasta Num Lock (*Num Lock key*). Prescurtare de la **Numeric Lock Key**, este o tastă cu două stări: activă, în care comută în modul de lucru numeric (în acest mod, tastele zonei numerice pot fi utilizate pentru a introduce date) și inactivă (în care tastele zonei numerice sunt folosite pentru mișcarea cursorului și defilarea paginii). Este însoțită de un LED care se aprinde atunci când tasta este activată.

tasta Page Down (*Page Down key*). Tastă care, împreună cu *Page Up*, este utilizată în mod obișnuit pentru defilarea paginii, distanța variind de la o aplicație la alta. În cazul în care înălțimea documentului afișat este mai mică decât cea a ecranului, apăsarea celor două taste nu are nici un efect.

tasta Shift (*Shift key*). Tastă care, atunci când este apăsată împreună cu o altă tastă, îi conferă acesteia din urmă o semnificație diferită față de cea obișnuită. De exemplu, folosind Shift împreună cu tastele literale, pot fi obținute caractere reprezentate prin litere mari; dacă tasta Caps Lock este activă, prin acest procedeu se obțin litere mici. De asemenea, în combinație cu alte taste, Shift produce caractere non-literale (atunci când este apăsată simultan cu o tastă asociată unei cifre) sau execută operații speciale.

tasta TAB (*TAB key*). Tastă marcată adesea cu două săgeți, una indicând către stânga, iar cealaltă către dreapta, utilizată de obicei pentru inserarea caracterelor de tabulare în documente

(apăsarea ei determină avansul cursorului către următorul *tab stop*). În programele de calcul tabelar, permite deplasarea de la o celulă la alta. De asemenea, în programele de baze de date, utilizarea tastei TAB determină deplasarea în cadrul unei înregistrări, de la un câmp la altul. Apăsarea simultană a tastelor ALT și TAB permite navigarea printre ferestrele deschise la un moment dat și activarea uneia dintre acestea.

tasta Page Up (*Page Up key*). A se vedea **tasta Page Down**.

tasta Print Screen (*Print Screen key*). Tastă inclusă în tastaturile PC-urilor compatibile IBM, prin apăsarea căreia conținutul memoriei de ecran este trimis la portul standard de imprimantă (în mod obișnuit, LPT1). A se vedea și **port de imprimantă**.

tastatură de calculator (*computer keyboard*). Dispozitiv de introducere (intrare) a datelor, modelat parțial după tastatura unei mașini de scris. Utilizează o serie de taste (*keys*), având caracterele gravate sau tipărite pe suprafața exterioară. Prin apăsarea tastelor sunt produse semnale electrice transformate apoi în coduri ale unor caractere printabile (litere, cifre, semne) sau de control. De asemenea, apăsarea simultană a unor taste produce unele comenzi (e.g., TAB + ALT).

tastatură ergonomică (*ergonomic keyboard*). Tastatură de calculator proiectată astfel încât să minimizeze

durerile provocate de utilizarea sa îndelungată. A se vedea și **tastatură de calculator**.

tastă de scurtătură (*shortcut key*). A se vedea **accelerator**.

tastă modificatoare (*modifier key*). Tastă care schimbă rolul altei taste atunci când este apăsată împreună cu aceasta (e.g., Alt, CTRL, Shift).

taste de navigare (*navigation keys*). Taste care permit controlul deplasării cursorului: Backspace, Home, End, Page Down, Page Up și cele patru taste săgeți.

(a) **tăia** (*cut*). A elimina o porțiune din documentul curent, editat. În mod obișnuit, operația este însoțită de plasarea porțiunii tăiate în memorie (*clipboard*), astfel încât aceasta să poată fi inserată (*i.e.*, lipită) în documentul respectiv sau în alt document similar. A se vedea și (a) **lipi** și **clipboard**.

tăiere și lipire (*cut and paste*). A se vedea (a) **tăia** și (a) **lipi**.

TCP (*Transmission Control Protocol, Protocolul de control al transmisiei*). Protocol utilizat la nivelul transport în modelul de referință TCP/IP. Este un protocol sigur, orientat pe conexiune, asigurând trimiterea fără erori a unui flux de octeți de la o mașină la alta (mașina destinație confirmă sosirea datelor). Aceasta înseamnă că, dacă unele date (segmente TCP) se pierd, la destinație se cere retransmiterea lor. A se vedea și **TCP/IP** și **UDP**.

TCP/IP. Model de referință utilizat în Internet. Permite calculatoarelor din întreaga lume (care utilizează sisteme de operare diferite) să comunice între ele. Denumirea modelului provine de la cele mai importante protocoale utilizate în cadrul său: TCP (*Transmission Control Protocol*) și IP (*Internet Protocol*).

Din punct de vedere istoric, protocoalele TCP și IP au fost create de Vinton Cerf și Robert Kahn în prima parte a anilor 1970. Ulterior (începând cu 1 ianuarie 1983), acestea au fost utilizate pentru comunicare în rețeaua ARPANET. Mai târziu, a apărut și modelul de referință TCP/IP. Deși acesta a fost utilizat inițial numai pentru transmisiile de date în rețeaua militară ARPANET, în cele din urmă el a fost adoptat și de rețelele civile, ajungând să definească modul de funcționare a Internetului.

Spre deosebire de ISO/OSI, modelul de referință are patru niveluri, enumerate și descrise pe scurt în continuare: nivelul acces la rețea (similar cu nivelurile legătură de date și fizic din modelul ISO/OSI, se ocupă în principal cu toate aspectele legăturilor fizice, furnizând tot ceea ce este necesar pentru a transmite un pachet IP); nivelul internet (denumit și nivelul *rețea*, se ocupă de expedierea pachetelor și de transmiterea lor până la destinație, trebuind să găsească cea mai bună rută pentru acestea); nivelul transport (realizează controlul fluxului de date, precum și detectarea și recuperarea erorilor); nivelul aplicație (realizează reprezen-

țarea și codificarea datelor, precum și controlul dialogului între aplicații).

Nivelul aplicație nu este identic cu cel din ISO/OSI, conținând ultimele trei niveluri din acest model (sesiune, prezentare și aplicație). Nivelul include toate protocoalele de nivel înalt. Primele protocoale de acest tip includeau terminalul virtual (TELNET), transferul de fișiere (FTP) și poșta electronică (SMTP). Ulterior, în decursul timpului au mai fost adăugate și alte protocoale, dintre care pot fi amintite DNS și HTTP. În scopuri practice, protocoalele de la acest nivel pot fi împărțite în două grupuri: protocoale-utilizator (folosite de aplicațiile utilizatorilor; e.g., HTTP, SMTP, Telnet, FTP, IMAP) și protocoale de serviciu (asigură funcționarea corectă a Internetului; e.g., SNMP, utilizat în administrarea rețelei).

Nivelul transport este identic cu cel din modelul ISO/OSI. Permite conversații între entitățile pereche din gazdele-sursă și, respectiv, destinație, utilizând în acest scop două protocoale capăt-la-capăt: TCP și UDP.

Nivelul internet (cel mai important nivel al modelului) are ca scop emiterea de pachete de către oricare gazdă și transmiterea lor spre destinație, acestea circulând independent unele de altele, chiar și în alte rețele. Nivelul definește un format oficial de pachet (denumit *pachet IP*) și protocolul IP (*Internet Protocol*). Alături de acesta, la nivelul internet mai sunt utilizate alte protocoale, dintre care amintim ICMP

(*Internet Control Message Protocol*), ARP (*Address Resolution Protocol*) și RARP (*Reverse Address Resolution Protocol*). A se vedea și TCP și UDP.

TDM (*Time Division Multiplexing, multiplexarea cu diviziune în timp*). Asigurarea printr-un mediu de transmisie dat a unui număr de două sau mai multe canale care funcționează simultan, prin alocarea pentru fiecare a unei porțiuni distincte din timpul de transmisie disponibil. În mod obișnuit, timpul de transmisie alocat fiecărui canal este constituit din intervale periodice, separate de intervalele alocate celorlalte canale. A se vedea și **multiplexare**.

TeacherTube. Serviciu de *video sharing* (partajare video) asemănător *You Tube*, dedicat numai domeniului educațional (<http://www.teachertube.com/>). Permite: încărcarea, etichetarea și partajarea clipurilor video; atașarea la clipurile video încărcate a unor fișiere în diverse formate, care includ note, planuri de lecție, evaluări etc.; căutarea și vizualizarea clipurilor video încărcate de membrii comunității; crearea sau înscrierea în grupuri de persoane care au interese similare; integrarea clipurilor video încărcate pe *TeacherTube* în site-uri Web prin includere sau folosind API-uri; alegerea modalității de difuzare a clipurilor video încărcate, acestea putând fi publice (vizualizate de toată lumea) sau private (partajate numai cu persoanele invitate). A se vedea **video sharing** și **YouTube**.

tehnici de căutare (*search techniques*). Tehnici utilizate pentru a obține rezultate cât mai apropiate de cele dorite atunci când se folosesc motoarele de căutare pe Web: utilizarea cuvintelor-cheie, folosirea operatorilor logici, căutarea frazelor, utilizarea caracterului *wildcard*, restrângerea de domeniu, căutarea pe categorii. A se vedea și **motor de căutare**.

tehnologia informației și a comunicațiilor (ICT, *information and communication technology*). Termen-umbrelă care acoperă toate tehnologiile avansate dezvoltate în domeniul comunicațiilor și al manipulării și furnizării informațiilor.

tehnologie cu accesibilitate (*accessibility technology*). Produse, dispozitive sau echipamente care pot fi utilizate, cu sau fără asistență, de persoane cu dizabilități. A se vedea și **accesibilitate**.

tehnologie pentru server (*server technology*). Oferă posibilitatea generării dinamice a conținutului Web trimis utilizatorului – utilizând diverse surse de date (e.g., baze de date, fișiere text, documente XML, *stream*-uri multimedia) –, dar și efectuarea unor calcule și procesări complexe.

Din punct de vedere istoric, prima metodă de generare dinamică pe server a conținutului Web a fost standardul *de facto* CGI. Următoarea etapă în evoluția metodelor de generare dinamică a conținutului Web a fost utilizarea interfețelor de programare NSAPI

(*Netscape Server API*) și ISAPI (*Microsoft Internet Services API*), acestea funcționând conform modelului CGI.

Astăzi, sunt utilizate în mod obișnuit limbajele/platformele de programare pentru server, care oferă posibilitatea dezvoltării unor aplicații Web cu complexitate ridicată. Acestea, denumite adesea servere de aplicații, funcționează de regulă ca extensii ale serverului Web. A se vedea și **CGI** și **server de aplicații**.

telefon inteligent (*smartphone*).

Telefon mobil care oferă funcții avansate, similare cu cele ale unui calculator personal. Deși nu există un standard industrial pentru producerea telefoanelor inteligente, aceste dispozitive moderne au în comun: utilizarea unor procesoare puternice, memorie și ecrane de afișaj mari, precum și sisteme de operare avansate. Cele mai cunoscute sisteme de operare utilizate de telefoanele inteligente sunt Symbian OS, iPhone OS, Windows Mobile, Linux, Palm WebOS și Android.

telefon prin satelit (*satellite telephone*). Tip de telefon mobil care utilizează pentru conectare sateliți în locul celulelor terestre. Depinzând de arhitectura sa, sistemul folosit poate acoperi întregul Pământ sau numai o anumită regiune a acestuia.

Telnet (*Telecommunication Network*). Protocol client-server de rețea utilizat în conexiunile Internet sau în rețele locale (LAN), dezvoltat în anul 1969. În mod obișnuit, *telnet* furnizează acces

la o interfață în linie de comandă pe un calculator aflat la distanță.

Utilizarea protocolului presupune existența a două aplicații care cooperează, una pe calculatorul client și alta pe calculatorul server (*i.e.*, gazdă, cel care va fi accesat la distanță). După realizarea conexiunii *telnet* (care necesită, în cele mai multe cazuri, un nume de utilizator și o parolă), calculatorul client devine terminal al gazdei. Nu sunt posibile mai multe conexiuni în același timp.

Termenul *telnet* desemnează, de asemenea, aplicația care implementează partea de client a protocolului. Au fost creați clienți *telnet* pentru toate platformele. *Telnet* poate fi utilizat pentru accesarea poștei electronice, a bazelor de date, a fișierelor etc.

teoria celor șase grade de separare (*theory of six degrees of separation*).

Teorie care stă la baza rețelelor sociale *online*. Conform acestei teorii, propusă de scriitorul ungar Frygies Karinthy în povestirea *Chains*, publicată în 1929, fiecare locuitor al planetei poate fi contactat de orice străin printr-un lanț alcătuit din cel mult cinci intermediari. Dintre experimentele recente care au căutat să demonstreze teoria celor șase grade de separare în diverse structuri de rețea pot fi amintite *Six Degrees of Kevin Bacon* și *Find Satoshi*. A se vedea și **Find Satoshi**, **rețele sociale online**, **Six Degrees of Kevin Bacon** și **Stanley Milgram**.

Testing and Assessment Tool (TAT, *instrument pentru testare și evaluare*). Software care simplifică procesul de creare și administrare a testelor *online*.

În *e-Learning*, TAT-urile sunt utilizate împreună cu alte tipuri de instrumente. De exemplu, testele sunt incluse în pagini Web create cu WAT sau pot fi adăugate cursurilor și obiectelor de învățare create prin intermediul CAT-urilor, LMS-urilor și LCMS-urilor. Deși instrumentele de testare diferă în ceea ce privește modul de utilizare, există un ciclu comun de dezvoltare și administrare a testelor, precum și de raportare/folosire a rezultatelor acestora. Iată câteva TAT-uri cunoscute: *Exams Online*, *Perception Authoring Manager*, *Test Generator*. A se vedea și **Course Authoring Tool**, **Learning Management System** și **Perception Authoring Manager**.

TeX. Sistem de culegere a documentelor creat de Donald Knuth. Este gratuit și foarte popular în mediul academic. TeX este considerat de unii cea mai bună soluție pentru culegerea textului care conține formule matematice, mai ales în varianta LaTeX. A se vedea și **LaTeX** și **Knuth, Donald Erwin**.

text-ancoră (*anchor text*). Textul inclus într-o ancoră-sursă, vizibil utilizatorului. Având până la 60 de caractere, este textul asupra căruia execută click utilizatorul. Motoarele de căutare folosesc textul respectiv pentru a determina relevanța site-ului în care este inclusă hiperlegătura, precum și

a conținutului paginii către care este direcționat utilizatorul. Diversele browsere vor afișa diferit textul-ancoră. Atunci când utilizatorul plasează *mouse*-ul deasupra textului-ancoră, browserul afișează un *tooltip*. Iată un exemplu:

```
<a  
href="http://www.exemplu.com/  
pictori/luchian.html">  
Ștefan Luchian</a>
```

În exemplul prezentat, Ștefan Luchian este textul-ancoră. A se vedea și **ancoră** și **hiperlegătură**.

THESIS. Software comercial compatibil cu versiunile SCORM 1.2 și SCORM 2004, compus din patru aplicații care furnizează o modalitate rapidă de obținere a obiectelor de învățare pe baza unui conținut educațional existent într-o diversitate de formate (e.g., pagini Web, documente *MS Office*) (<http://getthesis.com/>). A se vedea și **obiect de învățare** și **SCORM**.

Thomson, Kenneth (n. 1943). Programator american, pionier al informaticii. În anii 1960 a lucrat împreună cu Dennis Ritchie la dezvoltarea sistemului de operare Multics. A contribuit, alături de D. Ritchie, la dezvoltarea sistemului de operare UNIX. În 1983 a câștigat premiul Turing, împreună cu Dennis Ritchie. A se vedea și **Ritchie, Dennis** și **UNIX**.

thread (*fir de execuție*). A se vedea **fir de execuție**.

three-tier architecture (*arhitectură pe trei niveluri*). Arhitectură de rețea care

constă din trei tipuri de noduri: clienți, servere de aplicații (care procesează datele pentru clienți) și servere de baze de date (care stochează datele utilizate de serverele de aplicații). Configurația descrisă este cel mai utilizat tip de arhitectură client-server. În general, configurațiile care conțin mai mult de două niveluri se numesc arhitecturi *multi-nivel* (*multi-tiered*) sau *n-nivel* (*n-tiered*). A se vedea și **arhitectura client-server**.

thumbnail (*miniatură*). A se vedea **miniatură**.

TIC. A se vedea **tehnologia informației și a comunicațiilor**.

TIFF (*Tag Image File Format*). Format de fișier utilizat pentru stocarea imaginilor. Este suportat pe scară largă de aplicațiile pentru manipularea imaginilor, procesare de text, recunoaștere optică a caracterelor, scanare, fax etc. Extensiile de fișier utilizate sunt *.tif* și *.tiff*. Tipul MIME folosit este *image/tiff*.

timp de răspuns (*response time*). **1.** Timpul scurs între lansarea unei cereri (*request*) și primirea răspunsului (*response*), care poate include datele solicitate sau precizarea că acestea nu pot fi furnizate, din diverse motive. **2.** Timpul necesar unui dispozitiv inclus într-un sistem de calcul pentru a furniza datele solicitate de CPU. A se vedea și **CPU**.

timp universal coordonat (*Universal Time Coordinated*, UTC). Timp standard bazat pe IAT (*International*

Atomic Time), cu un salt de câteva secunde adăugat pentru a compensa încetinirea în timp a rotației Pământului. Neoficial, UTC poate fi considerat echivalent GMT (*Greenwich Meridian Time*) atunci când fracțiunile de secundă sunt nesemnificative.

UTC a înlocuit GMT în numeroase regiuni ale lumii începând cu 1 ianuarie 1972. Timpii zonali (*i.e.*, în diverse regiuni de pe glob) pot fi exprimați în funcție de UTC, utilizând o compensare (*offset*) negativă sau pozitivă.

Timsoft. Companie din Timișoara. Pune la dispoziție cursuri *online*, găzduire/dezvoltare de cursuri, consultanță în *e-Learning* și în utilizarea blogurilor, dezvoltare de sisteme integrate de *e-Learning*, design Web (<http://www.timsoft.ro/>).

TinyURL. Serviciu care furnizează *alias*-uri scurte pentru redirectarea URL-urilor lungi (<http://tinyurl.com/>) utilizând tehnica denumită *URL shortening* (*scurtarea URL-ului*). A fost lansat în ianuarie 2002 de către dezvoltatorul Web Kevin Gilbertson. Pagina principală a aplicației include un formular folosit pentru a introduce URL-ul lung care va fi trimis în vederea scurtării. Pentru fiecare URL trimis, serverul adaugă într-o bază de date un *alias* al acestuia (un URL scurt), pe care apoi i-l returnează utilizatorului. De asemenea, TinyURL oferă un API, permițându-le astfel programatorilor să creeze aplicații care automatizează procesul de creare a URL-urilor scurte. A se vedea și **scurtarea URL-urilor**.

tip de date (*data type* sau *datatype*). În limbajele de programare, concept care include o mulțime de valori, o regulă de codificare a lor (modul în care sunt reprezentate în memorie) și o mulțime de operații definite pe mulțimea valorilor.

În fiecare limbaj de programare există o serie de tipuri de date predefinite (standard). În mod obișnuit, acestea sunt tipuri simple, numite și scalare (e.g., întreg, real) și tipuri compuse (e.g., tablou, obiect). Tipurile compuse pot împacheta un număr arbitrar de date simple sau compuse. În majoritatea limbajelor, utilizatorii au posibilitatea de a defini tipuri de date noi prin combinarea celor existente și prin definirea unor operații valide pentru noile tipuri.

tipuri de e-Learning (*e-Learning types*). În cadrul conceptului de *e-Learning* se deosebesc următoarele tipuri: 1) învățarea bazată pe computer (CBL, *Computer-based learning*) se referă la utilizarea calculatorului în sala de clasă, acesta fiind văzut ca o componentă-cheie a mediului educațional; în general, conceptul CBL este considerat diferit de cel în care învățarea reprezintă numai un element periferic al utilizării calculatorului (e.g., jocuri pe calculator, navigare Web); 2) pregătirea bazată pe computer (CBT, *Computer-based training*) – în cadrul serviciilor de acest tip studenții învață prin parcurgerea unor programe speciale de pregătire pe calculator, în funcție de

domeniul lor ocupațional; adesea, CBT este utilizat pentru a învăța personalul să utilizeze diverse aplicații software; de multe ori, aplicațiile livrate utilizatorilor includ tutoriale, acestea reprezentând o formă modestă de CBT; 3) pregătirea bazată pe Web (WBT, *Web-based training*) este similară cu CBT, fiind oferită în Internet și accesată prin intermediul unui browser Web; acest tip de pregătire utilizează frecvent metode interactive, așa cum sunt listele de discuții, camerele de *chat*, mesageria instantanee și videoconferințele; în mod obișnuit, WBT este utilizat pentru învățare adaptată la ritmul propriu al studentului, existând și sisteme folosite pentru testarea și evaluarea studenților la anumite date; 4) învățarea colaborativă susținută de computer (CSCL, *Computer-supported collaborative learning*) este cel mai nou tip de *e-Learning*, destinat îmbunătățirii predării și învățării cu ajutorul noilor tehnologii de comunicare; învățarea colaborativă (sau de grup) se referă la metodele de instrucție în care studenților li se cere să lucreze împreună pentru îndeplinirea unor sarcini de învățare; acest nou tip de învățare se deosebește radical de modelul tradițional, în care educatorul își asumă integral rolul de distribuitor de cunoștințe și formator de competențe. A se vedea și **e-Learning**.

tipuri MIME XHTML (*MIME XHTML types*). Tipuri de conținut pentru documentele XHTML. Acestea sunt: `text/html` (browserul va procesa documentul ca și cum acesta ar fi un

document HTML), `application/xhtml+xml` (browserul va procesa documentul utilizând regulile sintactice XHTML), `text/xml` sau `application/xml` (ultimele două pot fi precizate pentru orice document din familia XML, inclusiv XHTML). A se vedea și **MIME**.

TLS (*Transport Security Layer*). Protocol utilizat pentru securizarea comunicației în Internet. În ciuda numelui, acesta nu este altceva decât o versiune îmbunătățită a SSL (e.g., dacă versiunea SSL este 3.0, mesajele TSL vor indica versiunea sa de protocol 3.1). A se vedea și **SSL**.

Tomlinson, Ray (n. 1941). Programator american, născut la Amsterdam. În 1971 a implementat un sistem de *e-mail* utilizat în rețeaua ARPANET. Acesta a fost primul sistem capabil să trimită mesaje între gazde diferite conectate la aceeași rețea. A se vedea și **ARPANET** și **poșta electronică**.

tooltip (*tooltip*). Element utilizat în interfețele grafice în conjuncție cu un cursor, acesta fiind adesea indicatorul *mouse*-ului. Atunci când utilizatorul suprapune cursorul peste un item (fără a executa click), lângă acesta este afișată o casetă de dimensiuni reduse (i.e., *tooltip*-ul) care conține informații suplimentare (sub forma unui text, de obicei scurt) referitoare la itemul respectiv. *Tooltip*-urile sunt des utilizate în paginile Web. Pentru a indica browserului să afișeze un *tooltip*, se folosește atributul `title` al unor marcaje

(X)HTML, valoarea sa fiind chiar conținutul *tooltip*-ului. Acesta va fi afișat atunci când utilizatorul suprapune cursorul peste conținutul marcatului.

topologie de rețea (*network topology*). Disponerea fizică a calculatoarelor, liniilor de transmisie și echipamentelor de rețea. Există trei topologii standard: inel, magistrală și stea. LAN-urile cu difuzare utilizează topologiile magistrală și inel. În schimb, LAN-urile *peer-to-peer* pot fi: rețele în stea, inel, arbore (variantă de topologie stea) sau combinate. Aceste rețele conectează calculatoarele utilizatorilor direct sau prin intermediul unui dispozitiv numit *hub* (*distribuitor*). A se vedea și **topologie inel**, **topologie magistrală** și **topologie stea**.

topologie inel (*ring topology*). Topologie de rețea standard în care echipamentele sunt conectate în cadrul unei bucle închise a cablului de interconectare. Datele sunt trecute prin fiecare echipament până ajung la destinație. În practică, topologia inel este combinată cu topologia stea într-o structură mai complexă: echipamentele sunt legate în stea la un server local, iar acesta este conectat în inel cu echipamentele centrale. Rețelele bazate pe această topologie necesită puțin cablu de conectare, dar sunt greu de extins și de modificat. De asemenea, un singur echipament defect întrerupe întreaga rețea. Topologia inel este implementată de obicei prin tehnologia *TokenRing*. A se vedea și **topologie de rețea**, **topologie magistrală** și **topologie stea**.

topologie magistrală (*bus topology*). Denumită și *liniară*, este o topologie de rețea standard care asigură conectarea directă la server a oricărui calculator din cadrul rețelei. Datele sunt transmise direct prin cablu, fără să fie preluate sau prelucrate de vreun dispozitiv intermediar. O astfel de rețea este ușor de implementat, dar greu de diagnosticat. Pentru implementarea topologiei magistrală este utilizată tehnologia Ethernet la 10 Mbps. A se vedea și **topologie de rețea**, **topologie inel** și **topologie stea**.

topologie stea (*star topology*). Topologie de rețea standard în care echipamentele sunt legate la un calculator central, prin intermediul căruia sunt realizate transmisiile de date. O variantă este topologia *ierarhică* sau *arbore* (*tree*), în care datele sunt transferate prin mai multe dispozitive până ajung la destinație. O rețea construită pe baza acestei topologii este ușor de modificat și de diagnosticat. În schimb, necesită mult cablu de interconectare. A se vedea și **topologie de rețea**, **topologie inel** și **topologie magistrală**.

Torvalds, Linus (n. 1969). Inginer finlandez de software, născut la Helsinki. Este inițiatorul nucleului sistemului de operare Linux, astăzi coordonând dezvoltarea acestuia sub licență GNU GPL. A absolvit cursurile Universității din Helsinki în 1996, obținând un masterat în informatică pe baza tezei *Linux: un sistem de operare portabil*. A se vedea și **Linux**.

touchpad (*touchpad*). A se vedea **trackpad**.

touchscreen (*ecran tactil*). Ecran care poate detecta locul din zona de afișare în care este atins. Atingerea se poate face direct, cu mâna/degetul, sau prin intermediul unui creion special. Prin atingere, utilizatorul poate executa operații similare cu cele pe care le efectuează folosind *mouse*-ul: selectarea unui obiect pe ecran, mutarea cursorului pe ecran, deschiderea unui meniu sau punerea în execuție a unui program.

trackball (*trackball*). Dispozitiv de indicare folosit pentru controlul cursorului pe ecran. În acest scop, utilizatorul acționează o bilă plasată între două role. Mișcarea de rotație a bilei este transformată de dispozitiv în mișcări ale cursorului. De asemenea, dispozitivul include o serie de butoane folosite pentru realizarea altor acțiuni. A se vedea și **dispozitiv de indicare**.

trackpad (*trackpad*). Denumit și *touchpad*, este un dispozitiv de indicare folosit pentru controlul cursorului pe ecran. În acest scop, utilizatorul mișcă degetul pe suprafața *trackpad*-ului. Este folosit în special pentru calculatoarele laptop. A se vedea și **dispozitiv de indicare**.

trademark (*marcă înregistrată*). Semn distinctiv utilizat de o persoană, o organizație sau o companie pentru a arăta că produsele sau serviciile pe care le oferă consumatorilor sunt diferite de cele oferite de alte entități juridice.

Semnul este însoțit de simboluri specifice: TM, SM sau ®.

transferul argumentelor (*passing arguments*). În programare, înlocuirea argumentelor formale cu cele actuale, la apelarea unei funcții. A se vedea și **transferul argumentelor prin referință** și **transferul argumentelor prin valoare**.

transferul argumentelor prin referință (*passing arguments by reference*). Una dintre modalitățile în care se realizează transferul argumentelor către o funcție. Această modalitate impune ca argumentul funcției să fie o variabilă. În loc să se refere la copia valorii variabilei transmise, variabila corespunzătoare din funcție se referă chiar la valoarea variabilei respective, de fiecare dată când este utilizată. Drept rezultat, funcția va schimba chiar valoarea variabilei transferate. Iată un exemplu scris în PHP în care este folosită această modalitate de transfer:

```
<?php
function cub(&$numar) {
    $numar =
        $numar*$numar*$numar;
}
$nr = 4;
cub($nr);
echo $nr;
// Se afiseaza 64
?>
```

După cum se poate observa în exemplul prezentat, pentru a se realiza transferul prin referință, argumentul funcției trebuie prefixat folosind caracterul „&”. A se vedea și **transferul argumentelor**.

transferul argumentelor prin valoare (*passing arguments by value*). Una dintre modalitățile în care se realizează transferul argumentelor către o funcție. Expresia transmisă (transferată) ca argument este evaluată, iar valoarea obținută este atribuită în corpul funcției variabilei cu același nume. În continuare este prezentată funcția PHP `cub()` care întoarce argumentul care îi este transmis, ridicat la puterea a treia:

```
function cub($numar) {
    return $numar*$numar*$numar;
}
```

Iată un script în care se realizează transferul argumentului prin valoare către funcția `cub()`:

```
<?php
// definitia functiei
$numar = 10;
echo cub($numar).</br>;
echo cub(2*5);
?>
```

În exemplul prezentat, funcția `cub()` este apelată de două ori, argumentul fiind transmis de fiecare dată prin valoare și evaluat la valoarea 10.

Este posibil ca argumentele transferate prin valoare funcțiilor să fie și tablouri, ca în exemplul prezentat în continuare, în care funcția `suma_el()` returnează suma elementelor tabloului transferat ca argument:

```
<?php
function suma_elem($tablou) {
    $nrelem = count($tablou);
    $sumaelem = 0;
    for($i = 0; $i < $nrelem;
        $i++) {
        $sumaelem += $tablou[$i];
    }
}
```

```

return $sumaelem;
}
echo "Suma elementelor
tabloului este".
suma_elem(array(10, 5,
12));
?>

```

În exemplul prezentat anterior, programul va afișa după execuție 27. A se vedea și **transferul argumentelor**.

(a) **translata** (*translate*). În programare, a converti un program dintr-un limbaj de programare în altul. Operația se realizează folosind programe dedicate (e.g., asamblor, compilator, interpretor). A se vedea și **asamblor**, **compilator**, **interpretor** și **translator**.

translator (*translator*). Program care realizează translatarea unui program dintr-un limbaj în altul. A se vedea și (a) **translata**.

transmisie duplex (*transmission duplex*). Transmisia realizată într-un sistem duplex. A se vedea și **sistem duplex**.

trigger (*declanșator*). Într-o bază de date, procedură executată automat atunci când un utilizator încercă să efectueze actualizarea datelor. *Trigger*-ele au rolul de a menține integritatea bazei de date. A se vedea și **integritatea bazei de date**.

trimitere manuală (*manual submission*). Înscrierea manuală a unui site într-un director Web sau într-un motor de căutare. Deși necesită timp, este considerată de multe persoane singura metodă corectă de a realiza înscrierea.

Alături de trimiterea manuală, se utilizează și trimiterea automată, prin folosirea unui software specific.

troff (*typesetting runoff*). Unul dintre cele mai vechi programe UNIX de formatare a textului. Troff a fost dezvoltat în 1971 la AT&T Bell Laboratories de Joseph Ossanna.

troian (*trojan horse*). Program de tip *malware* care, în aparență, realizează o funcție dorită de utilizator, dar, în realitate, execută funcții ascunse, nedorite și/sau dăunătoare. Este creat cu scopul de a obține informații și de a administra calculatorul pe care este executat. Numele programului are ca sursă de inspirație legenda calului de lemn folosit de greci pentru a se ascunde în cetatea Troia, pe care voiau să o atace. A se vedea și **malware**.

Tumblr. Serviciu gratuit de *microblogging* (<http://www.tumblr.com/>). A se vedea și **microblogging**.

Turing, Alan Mathison (1912–1954). Logician, matematician, criptanalist și informatician englez. Este considerat adesea „părintele informaticii moderne”. Este inventatorul mașinii virtuale care îi poartă numele (*Turing machine*). În anul 1999, revista Time l-a desemnat ca fiind unul dintre cele mai importante 100 de personalități ale secolului XX, pentru rolul pe care l-a avut în dezvoltarea calculatoarelor moderne. A se vedea și **mașina Turing** și **premiul Turing**.

Twitter. Rețea de socializare și serviciu de *microblogging* (<http://twitter.com/>), a fost lansată oficial în octombrie 2006 de firma americană Obvious (<http://obvious.com/>), obținând în martie 2007 *South by Southwest Web Award* (Austin, Texas), la categoria *bloguri*.

Mărimea însemnărilor – numite *tweets* (*ciripeli*) – care pot fi create de utilizatori este limitată la 140 de caractere. Jack Dorsey, recunoscut ca fiind omul din spatele conceptului *Twitter*, a spus în discursul de acceptare a premiului: „Am dorit să mulțumim în 140 de caractere sau mai puțin. Și tocmai asta am făcut.”

Popularitatea deosebită a rețelei *Twitter* este datorată și API-ului propriu (*Twitter API*, <http://apiwiki.twitter.com/>), pe care

twitter-ii (*twitterers*) îl pot utiliza pentru a crea propriile lor aplicații. O astfel de aplicație este *mashup*-ul *Twitvision* (creat folosind și *Google Maps*), care afișează locul unde se află cei care postează însemnări. De asemenea, *mashup*-ul *TwitterDir* (*i.e.*, *Twitter Directory*, <http://twitdir.com/>), bazat pe același API, permite identificarea utilizatorilor din locuri (orașe și țări) precizate. A se vedea și **microblogging**.

typeface (*tip de caractere*). Model de caractere proiectat folosind un stil specific. Noțiunea se deosebește de cea de *font* (caractere de un anumit tip și o anumită dimensiune). A se vedea și **font**.

TypePad. Serviciu comercial de *blogging* (<http://www.typepad.com/>).

U

UAL (*Unitatea Aritmetică și Logică*). Parte a CPU care efectuează asupra datelor operațiile aritmetice și logice specificate de instrucțiuni. Înainte de efectuarea unei operații, datele sunt încărcate în regiștri CPU. După efectuarea operației, rezultatele sunt depuse tot în regiștri. A se vedea și **CPU**.

UCC (*Unitatea de Comandă și Control*). Parte a CPU cu următoarele funcții: decodificarea codului operației; calculul adresei operanzilor care participă la operație și extragerea lor din memorie, generarea comenzilor necesare pentru execuția instrucțiunii, generarea comenzilor necesare pentru memorarea rezultatului și a informațiilor de stare, calculul adresei instrucțiunii următoare și citirea acesteia din memorie. A se vedea și **CPU**.

UDDI (*Universal Description, Discovery and Integration*). Standard pentru înregistrarea și regăsirea serviciilor Web. A fost publicat pe 6 septembrie 2000, la el contribuind firmele Ariba, IBM și Microsoft. Ulterior, suportul pentru această tehnologie s-a extins, implicând o comunitate foarte mare de firme. Proiectul UDDI menține un

registru public de afaceri (cu servicii incluse) accesibil online, denumit UBR (UDDI Business Registry). Acesta conține, de fapt, câțiva regiștri replicați găzduiți de firme numite *operatori UDDI* sau *operatori de regiștri*.

Standardul UDDI furnizează două specificații care definesc structura registrului de servicii și operațiile care se pot efectua: o definiție a informațiilor care trebuie furnizate despre fiecare serviciu și despre cum trebuie codificate acestea (set de structuri de date); o specificație API care oferă un set de API-uri pentru publicare (înregistrarea serviciilor) și interogare (pentru a găsi serviciile). În plus, operatorii de regiștri furnizează fiecare câte o interfață utilizator bazată pe pagini Web, prin intermediul căreia pot fi realizate operațiuni de înregistrare, gestionare și căutare de servicii Web.

UDDI codifică trei tipuri de informații despre serviciile Web: informații de tip „pagini albe”, care includ nume și detalii de contact; informații de tip „pagini galbene”, care oferă o împărțire a serviciilor bazată pe tipul serviciului și pe categoria de afaceri; informații de tip „pagini verzi”, incluzând date tehnice despre serviciu.

Informațiile oferite în registru permit regăsirea serviciilor Web în funcție de anumite caracteristici ale acestora. Astfel, în mod obișnuit, căutarea serviciilor Web se face în funcție de categoria din care fac parte, în funcție de nume, în funcție de furnizor și în funcție de tipul serviciului. În limbajul

UDDI, tipurile de servicii sunt numite *tModel*. Fiecare *tModel* este caracterizat printr-un nume, o descriere și un identificator unic (*tModelKey*).

Un registru UDDI conține date despre un ansamblu de servicii Web, fiind actualizat automat la orice schimbare sau adăugare de noi servicii. Registrele UDDI pot fi publice sau private. Datele UDDI sunt găzduite de noduri UBR (ale unor companii), care au obligația de a le menține în conformitate cu specificațiile precizate de consorțiul UDDI.org. Firme ca Microsoft și IBM întrețin regiștri publici, conform specificațiilor UDDI. Mai multe noduri se pot organiza într-un nor UDDI, toate nodurile având aceleași înregistrări. Unui operator de nod UBR i se cere să replice datele celorlalte noduri prin utilizarea unui canal securizat, furnizând o redundanță întregului ansamblu de noduri care găzduiesc servicii Web înregistrate. Rezultă că, dacă un furnizor înregistrează un serviciu Web într-un nod, după replicare, utilizatorii pot descoperi acest serviciu și în celelalte noduri componente ale norului UDDI respectiv. A se vedea și **servicii Web**.

UDP (*User Datagram Protocol, protocolul datagramelor utilizator*). Protocol utilizat la nivelul transport în modelul de referință TCP/IP. Este un protocol nesigur, neorientat pe conexiune, utilizat de acele aplicații care pot realiza singure fragmentarea și controlul fluxului. UDP transportă date

folosind datagrame, a căror livrare nu este garantată. Cu alte cuvinte, sursa trimite datagrame fără a ține seama dacă acestea au ajuns sau nu. A se vedea și **TCP și TCP/IP**.

Ulead PhotoImpact. Program comercial de editare pentru grafică *raster*, creat de compania Ulead Systems (<http://www.ulead.com/>). Datorită facilităților pentru Internet design pe care le oferă, *PhotoImpact* produce nu numai elemente componente ale paginilor Web (*i.e.*, imagini), ci chiar aceste pagini, specificându-le titlul, cuvintele-cheie, culoarea de fundal, adăugând text HTML, stabilind mărimea caracterelor, alinierea textului etc. În septembrie 2009, Corel (aceasta are în proprietate din 2006 compania InterVideo, care – la rândul ei – cumpărase anterior Ulead Systems) a oprit dezvoltarea programului *Ulead PhotoImpact*, concentrându-se în schimb pe *Paint Shop Pro*. A se vedea și **Corel Corporation, formate raster și Paint Shop Pro**.

UML (*Unified Modelling Language*). Limbaj standard pentru descrierea de modele și specificații pentru software, dezvoltat de consorțiul OMG (*Object Management Group*) începând cu anul 1997. UML include un set de elemente grafice utilizate pentru reprezentarea vizuală a sistemelor software.

undo (*a anula*). A reveni asupra unei acțiuni într-un program de aplicație, pentru a corecta o eroare făcută neintenționat (*e.g.*, ștergerea unei secvențe

de text) sau pentru a reveni la o variantă mai bună a documentului. A se vedea și **redo**.

Unicode. Standard definit de organizația non-profit *Unicode Consortium* pentru reprezentarea și manipularea textului exprimat prin intermediul majorității sistemelor de scriere din lume (<http://www.unicode.org/>). Folosește un cod unic pentru fiecare caracter, indiferent de platformă, aplicație sau limbă. Este adoptat de cei mai mulți producători de software și implementat în cele mai recente tehnologii dezvoltate de aceștia. De asemenea, standardul este suportat de multe sisteme de operare, navigatoare Web actuale și alte aplicații. Unicode poate fi implementat folosind diverse codificări de caractere, cele mai cunoscute dintre acestea fiind UTF-8 și UF-16. A se vedea și **UTF-8**.

unire (join). Într-o bază de date relațională, operație care combină înregistrări din două sau mai multe tabele. Aceasta creează un set care poate fi salvat ca un tabel sau utilizat ca atare. A se vedea și **bază de date relațională** și **tabel**.

unitate flash USB (flash USB drive). Dispozitiv de stocare reinscribibil bazat pe tehnologia de memorare *flash*, integrat cu interfața USB. O astfel de unitate constă dintr-un mic circuit de memorare, protejat în interiorul unei carcase realizate din material plastic, conectorul USB fiind acoperit de un capac detașabil. Primele variante comerciale ale acestor unități au apărut

în anul 2000. A se vedea și **memorie flash**.

UNIVAC (UNIVersal Automatic Computer). Serie de calculatoare comerciale. UNIVAC I – proiectat de J. Presper Eckert și John Mauchly – a fost primul calculator comercial produs în SUA, construcția sa fiind terminată în 1951. UNIVAC II a fost livrat în 1958. UNIVAC II – construit cu tranzistori care înlocuiau tuburile cu vid utilizate anterior – a fost produs în 1962.

UNIX. Sistem de operare dezvoltat în 1969 la Bell Laboratories de un grup din care făceau parte, printre alții, Ken Thomson, Dennis Ritchie și Brian Kernighan. Din 1997, The Open Group (<http://www.opengroup.org/>) este proprietarul mărcii înregistrate UNIX. Sistemele de operare care respectă specificațiile SUS (*Single UNIX Specification*) pot utiliza această marcă, restul fiind numite *Unix system-like* (sau *Unix-like*). În anii 1970 și la începutul anilor 1980, UNIX a avut o influență majoră în mediile academice, contribuind la adoptarea pe scară largă a acestui sistem de operare și la dezvoltarea unor versiuni comerciale (e.g., Solaris, HP-UX). Linux și BSD sunt sisteme de operare *Unix-like* cunoscute. A se vedea și **BSD** și **Linux**.

unsubscribe (anularea subscrierii). Anularea subscrierii făcute anterior la o sursă de informare (e.g., *e-newsletter*). A se vedea și **subscribe**.

upload (*încărcare*). Procesul prin care se transferă o copie a unui fișier de la calculatorul local la un calculator aflat la distanță, prin intermediul unei rețele. A se vedea și **download**.

URI (*Uniform Resource Identifiers*). Descriptor care identifică în mod unic o resursă. În general, URI-urile sunt folosite pentru: localizarea și identificarea resurselor (*e.g.*, pagini Web, adrese de *e-mail*); specificarea spațiilor de nume utilizate în documentele XML sau derivate; specificarea unor tipuri de date (*e.g.*, tipurile XML Schema, mesajele WSDL); precizarea unor identificatori unici, care nu sunt legați de resurse, spații de nume sau tipuri de date (*e.g.*, câmpul `<wsa: MessageID>` în antetul SOAP).

Două subseturi importante ale URI-urilor sunt reprezentate de URL și URN. Recent, au fost introduși identificatorii internaționalizați de resurse (IRIs, *Internationalized Resource Identifiers*), complementari URI-urilor. Spre deosebire de aceștia din urmă, care utilizează setul de caractere ASCII, IRI-urile permit folosirea unor seturi extinse de caractere (*e.g.*, Unicode). De asemenea, grupul OASIS (<http://www.oasis-open.org/>) dezvoltă identificatorii extensibili de resurse (XRI, *eXtensible Resource Identifiers*). A se vedea și **URL** și **URN**.

update (*actualizare*). Trecerea la o nouă versiune a unui produs software. În mod obișnuit, o actualizare adaugă produsului caracteristici fără o importanță deosebită sau corectează

deficiențe (*bug-uri*) descoperite după lansarea acestuia. A se vedea și **bug**.

upgrade (*modernizare*). Trecerea la o nouă versiune a aceluiași produs (*firmware, hardware sau software*), cu scopul de a-l aduce la zi sau de a-i îmbunătăți caracteristicile. A se vedea și **firmware, hardware și software**.

uppercase (*litere mari*). Ceea ce ține de utilizarea literelor mari. A se vedea și **lowercase**.

UPS (*Uninterruptible Power Supply, sursă de alimentare neîntreruptibilă*). Dispozitiv care menține o tensiune constantă într-un anumit interval de timp, protejând astfel sistemul de calcul, inclusiv în cazul în care curentul electric este întrerupt accidental sau când apare o supratensiune.

URL (*Uniform Resource Locator*). Descriptor care identifică resursele prin intermediul unei reprezentări a mecanismului de acces la acestea (*e.g.*, localizarea în rețea, mai ales în Internet), având următoarea sintaxă:

```
[ schema:// ] domeniu :port [ cale ] [ ?sir_de_interogare ]
```

În structura unui URL intră următoarele componente: identificatorul de serviciu sau *schema* (reprezintă protocolul utilizat de serviciul Internet pentru accesarea resursei; dacă nu este precizat, se presupune că aceasta este valoarea implicită, adică `http`); numele serverului (*domeniu*) pe care este localizată resursa (identifică în mod unic o rețea/un calculator conectată/conectat

la Internet – este indicat, dar nu obligatoriu, să se evite utilizarea ca domeniu a adresei IP, în locul său folosindu-se numele simbolic asociat); portul (*port*) la care programul server „așteaptă” eventualele cereri de conectare (e.g., 8080, care este portul implicit al serverului *proxy*; în cazul în care – pentru serviciul Web – componenta *port* nu este specificată, se consideră că este utilizat portul 80, acesta fiind folosit implicit de serverul Web); calea absolută (*cale*) utilizată pentru precizarea resursei (este reprezentată prin intermediul unui șir de directoare separate între ele folosind caracterul „/”, terminat – eventual – cu numele fișierului în care este stocată resursa; dacă numele fișierului lipsește, serverul Web va livra clientului un fișier al cărui nume este precizat – pentru Apache – în intrarea `DirectoryIndex`, inclusă în fișierul de configurare al serverului Web); un șir de interogare (*sir_de_interogare*), utilizat cu schema `http` și care conține date necesare pentru accesarea resursei, în perechi de forma `nume_data = valoare_data`, separate prin „&”).

Unui URL i se poate adăuga ca sufix un identificator de fragment, precedat de caracterul „#”. Utilizând această modalitate se poate obține poziționarea într-o anumită secțiune a resursei identificate prin URL-ul respectiv. O adresă Web mai poate conține caracterul „~” (*tilda*), pentru a preciza pagina personală a unui utilizator care posedă cont pe serverul precizat prin URL, e.g., `http://www.exemplu.org/~mihai` (de

asemenea, caracterul *tilda* simbolizează directorul *home* al utilizatorului respectiv pe un calculator UNIX/Linux).

URL rewriting (*rescrierea URL-urilor*). A se vedea **rescrierea URL-urilor**.

URN (*Uniform Resource Name*). Desemnează un identificator de resursă persistent și independent de locație (chiar dacă resursa încetează să existe sau devine inaccesibilă), fiind utilizat pentru a desemna – prin intermediul unui nume – entități (e.g., tipuri de date, spații de nume) folosite de unele aplicații (e.g., servicii Web). Pentru a înțelege diferența dintre URL și URN, se poate folosi următoarea analogie: URN-ul este similar numelui unei persoane, iar URL-ul este similar adresei sale. URN-ul definește identitatea unei entități, iar URL-ul furnizează o metodă pentru a o găsi. De asemenea, URN-urile pot fi comparate cu sistemul ISBN (*International Standard Book Number*), utilizat pentru identificarea unică a cărților.

Iată câteva exemple de URN-uri: `urn:mpeg:mpeg7:schema:2001` (precizează spațiul de nume pentru video-metadatele MPEG-7); `urn:schemas-microsoft-com:datatypes` (desemnează tipurile de date definite de Microsoft); `urn:ISBN:973-681-907-8` (precizează numărul ISBN al unei cărți).

USB (*Universal Serial Bus*). Magistrală externă de comunicație bidirecțională, rapidă, care suportă rate de transfer de până la 12 Mbps, folosită în general

pentru conectarea unor dispozitive la calculatoarele personale. Un singur port USB poate suporta până la 127 de dispozitive periferice (e.g., microfon, mouse, tastatură, imprimantă). De asemenea, USB suportă *plug-and-play* și *hot plugging*. Începând cu 1998, standardul a căpătat o largă răspândire, așteptându-se ca porturile USB să înlocuiască total porturile serial și paralel. A se vedea și **USB 2.0**.

USB 2.0. Extensie USB 1.1, cunoscută și ca *Hi-Speed USB*, este o magistrală externă de comunicație care suportă rate de transfer de până la 480 de Mbps. Specificațiile USB 2.0 au fost publicate în aprilie 2000. A se vedea și **USB**.

USENET. Înființat în 1979 de Tom Truscott și Jim Ellis, absolvenți ai Duke University, este cel mai vechi sistem public de comunicații bazat pe o rețea de calculatoare, răspândit la scară mondială. Numele său provine de la *user* și *network*. A fost la început o rețea cu acces telefonic în comutație (*dial-up*), bazată pe UUCP (*UNIX-to-UNIX copy*). Oferea acces la *e-mail* și știri (*USENET News*). În prezent, poate fi accesat cu ajutorul unui server Web (e.g., EasyNews). USENET este locul în care Tim Berners-Lee a anunțat lansarea serviciului World Wide Web, Linus Torvalds pe cea a sistemului de operare Linux, iar Marc Andreessen crearea browserului *Mosaic* și a *tag*-urilor pentru imagini.

User-Agent. Unul dintre anteturile cererii HTTP. Conține informații despre

agentul de utilizator (*user-agent*) care a trimis cererea (e.g., browser, spider). A se vedea și **cerere HTTP**.

username (*nume de utilizator*). A se vedea **nume de utilizator**.

UTF-8 (*8 bit Unicode Transformation Format*). Codificare de caractere cu lungime variabilă, compatibilă cu ASCII, capabilă să reprezinte orice caracter în standardul Unicode. Datorită acestor caracteristici, este larg folosită în toate situațiile în care textele trebuie stocate sau transmise printr-o rețea de calculatoare (e.g., *e-mail*, pagini Web). În UTF-8, fiecare caracter este codificat folosind unul până la patru octeți. Un octet este utilizat pentru reprezentarea celor 127 de caractere ASCII (caracterele de bază ale alfabetului englezesc). Doi octeți sunt utilizați pentru reprezentarea caracterelor alfabetului latin cu toate diacriticele, dar și caractere din greacă, chirilică, arabă, coptă, ebraică și alte limbi. Trei octeți sunt utilizați pentru ceea ce se numește BMP (*Basic Multilingual Plane*), care este „stratul” de bază al Unicode, suportând practic toate alfabetele moderne. În sfârșit, patru octeți sunt utilizați în celelalte „straturi” Unicode, pentru reprezentarea caracterelor CJK (ale limbilor chineză, japoneză, coreeană) și a unor alfabet istorice. A se vedea și **Unicode**.

utilizator (*user*). Persoană care utilizează un calculator, un program sau un serviciu Internet. A se vedea și **cont de utilizator**.

utilizator al bazei de date (*database user*). Persoană care poate avea diverse funcții referitoare la o bază de date, de exemplu exploatarea sau crearea acesteia.

Principalele tipuri de utilizatori ai bazelor de date sunt: administratorul bazei de date (persoana care stabilește și descrie modelul bazei de date, după ce – în prealabil – realizează o analiză detaliată a sistemului informatic căruia îi este destinată, descriind toate entitățile, structura lor și legăturile dintre ele); administratorul aplicației (persoana care, pe baza modelului bazei de date și a structurii acesteia, analizează cerințele concrete ale unei aplicații, stabilind un submodel al bazei de date care va fi utilizat); programatorii de aplicații (persoanele care scriu programe utilizate pentru exploatarea bazei de date, folosind descrierea dată de administratorul aplicației; în acest scop, este folosit limbajul de manipulare al sistemului de gestiune a bazei de date); utilizatorii propriu-ziși ai bazei de date (persoanele

care utilizează baza de date prin intermediul aplicațiilor realizate de programatorii de aplicații). A se vedea și **bază de date**.

utilizator final (*end user*). Persoana care folosește un produs în forma sa finală, comercială (*e.g.*, un calculator sau o aplicație de calculator). Utilizatorul final poate fi diferit de cel care a cumpărat produsul (*e.g.*, un adult cumpără un calculator pe care îl dăruiește unui adolescent, acesta din urmă fiind utilizatorul final).

utilizator puternic (*power user*). 1. Utilizatorul unui calculator personal care are abilitatea de a folosi caracteristicile avansate ale programelor, dar nu este specialist în programare sau în administrarea sistemului. 2. În Windows 2000, Windows XP Professional și Windows 2003 Server, grupul *Power Users* include utilizatori cu drepturi de acces mai mari față de cele ale unui utilizator obișnuit, dar mai restrânse în raport cu cele ale administratorului. •

V

van Rossum, Guido (n. 1960). Programator olandez, absolvent al Universității din Amsterdam (1982), licențiat în matematică și informatică (<http://www.python.org/~guido/>). În anii 1980 a lucrat la dezvoltarea limbajului ABC, care a avut o influență majoră asupra limbajului Python, creat tot de van Rossum în 1991. În 2001 a primit *Award for the Advancement of Free Software*, decernat de Free Software Foundation. A se vedea și **Python**.

variabilă (*variable*). În programare, identificator asociat unei valori stocate în memoria calculatorului sau o expresie care poate fi evaluată. În unele limbaje de programare, o variabilă dată poate stoca numai un anumit tip de date (e.g., întreg, șir de caractere), în vreme ce în alte limbaje poate stoca orice tip suportat de limbaj. În limbajele de programare imperativă, variabilele pot fi în general accesate, iar valoarea lor poate fi schimbată oricând (spre deosebire de constante).

Pentru denumirea variabilelor se utilizează de obicei caractere multiple (e.g., `pret`), însoțite uneori (prefixate) de un caracter specific limbajului (e.g., „\$” în

PHP; „\$pret”). Variabilele ale căror nume sunt alcătuite dintr-un singur caracter (e.g., `i`, `j`) sunt utilizate de obicei în scopuri auxiliare (e.g., variabile de control). În unele limbaje de programare este obligatorie declararea variabilelor (e.g., C, C++, Java), în timp ce în altele, nu (e.g., PHP). A se vedea și **constantă**, **programare imperativă**, **tip de date** și **variabilă de control**.

variabilă automată (*automatic variable*). Numită și **variabilă locală** (*local variable*), este utilizată într-o funcție, având domeniul de valabilitate limitat numai la interiorul acesteia. Variabilele automate sunt create automat la începutul execuției funcției și distruse, tot automat, la sfârșit. Din acest motiv, funcțiile vor „pierde” după execuție valorile variabilelor utilizate în interiorul lor. A se vedea și **domeniu de valabilitate**.

variabilă de control (*control variable*). În programare, variabilă de program utilizată într-o instrucțiune de control care stabilește fluxul execuției acestuia. De exemplu, în instrucțiunea *for*, variabila de control este folosită pentru a stabili de câte ori se vor executa instrucțiunile incluse în corpul acesteia. A se vedea și **instrucțiune de control**.

variabilă globală (*global variable*). În programare, variabilă care poate fi accesată și modificată de oriunde (*i.e.*, din orice parte a programului/scriptului), inclusiv din interiorul funcțiilor. Deși unele limbaje de programare nu suportă variabilele globale implicite, există

posibilitatea extinderii explicite a domeniului de valabilitate a unei variabile. Variabilele declarate globale (i.e., în exteriorul oricărei funcții) pot fi utilizate peste tot în program, inclusiv în definițiile funcțiilor. În acest scop este necesar ca ele să fie „importate” în interiorul funcțiilor folosind (e.g., în PHP) cuvântul-cheie `global`.

În unele limbaje de programare care nu suportă utilizarea variabilelor globale implicite (e.g., PHP), există un număr de variabile globale automate (sau variabile superglobale) predefinite de tip tablou, cunoscute sub denumirea de *tablouri superglobale*, al căror comportament este similar cu cel al variabilelor globale utilizate în alte limbaje de programare. Astfel, aceste tablouri pot fi accesate în orice punct al scriptului, inclusiv în interiorul metodelor și funcțiilor, fără a fi necesară extinderea explicită a domeniului de valabilitate prin utilizarea cuvântului-cheie `global`. A se vedea și **domeniu de valabilitate**, **tablou superglobal** și **variabilă**.

variabilă locală (*local variable*). A se vedea **variabilă automată**.

variabilă statică (*static variable*). În programare, variabilă care poate fi accesată numai în domeniul local al funcției în care este definită, dar care nu își pierde valoarea după încheierea apelului funcției (așa cum se întâmplă – în mod obișnuit – cu variabilele locale). Rezultă că inițializarea variabilei statice trebuie să se facă într-un script o singură dată, la primul apel al funcției în care este declarată. Pentru a declara o

variabilă statică, se utilizează cuvântul-cheie `static`. Iată un exemplu scris în PHP în care se definește o astfel de variabilă:

```
<?php
function static_var() {
    static $a = 0;
    echo $a. "<br />";
    $a++;
}
static_var();
static_var();
static_var();
?>
```

Pentru exemplul anterior, rezultatele care vor fi afișate arată că la primul apel al funcției `static_var()` aceasta returnează valoarea 0, la următorul 1, iar la al treilea 2, ceea ce demonstrează că la încheierea unui apel, variabila `$a`, declarată statică, nu este distrusă și că inițializarea acesteia se face o singură dată. A se vedea și **variabilă**.

vector RAID (*RAID array*). A se vedea **RAID**.

ventilator de CPU (*CPU fan*). Ventilator electric plasat pe CPU sau în apropierea acestuia, având rolul de a contribui la disiparea căldurii degajate de cip în timpul funcționării.

Veronica (*very easy rodent-oriented netwide index to computerized archives*). Serviciu creat în 1992 la University of Nevada de Steven Foster și Fred Barrie, utilizat pentru căutare în arhivele *Gopher*. În operațiunea de căutare sunt folosite cuvinte-cheie și operatori booleeni (e.g., ȘI, SAU, SAU

EXCLUSIV). *Veronica-2* este o versiune complet rescrisă, de către Cameron Kaiser, a serviciului *Veronica*. A se vedea și **Gopher**.

versiune (*version*). O variantă a unui produs hardware sau software. În ceea ce privește software-ul, în orice moment există trei versiuni ale acestuia: stabilă (*stable*), în testare (*testing*) și instabilă (*unstable*).

versiune de rulare (*runtime version*). Cunoscută și ca *versiune runtime*, este o versiune limitată a unui program. Astfel, de exemplu, pentru a executa un program scris în limbajul Visual Basic, este nevoie de versiunea de rulare Visual Basic. Astfel, unele programe scrise în Visual Basic pot fi rulate, dar nu și dezvoltate. A se vedea și **runtime**.

videoclip (*secvență video*). Fișier în care, în mod obișnuit, este inclus un fragment dintr-o înregistrare video cu durată mai mare.

videoconferință (*videoconference*). Modalitate de comunicație bazată pe un set de tehnologii interactive care le permit utilizatorilor aflați la distanță să interacționeze simultan audio și video.

Videoconferințele sunt utilizate atunci când: este necesară o conversație în direct, informația vizuală este o parte importantă a conversației, persoanele care conversează nu se pot deplasa pentru a se întâlni (deplasarea necesită costuri ridicate sau consum mare de timp). Videoconferințele moderne sunt posibile prin intermediul unui MCU

(*Multipoint Control Unit*), care conectează apeluri din mai multe surse.

video microblogging (*video microblogging*). Formă de *microblogging* în care comunicația asincronă dintre utilizatori este realizată prin intermediul mesajelor video. Cel mai cunoscut serviciu de acest tip este *Seesmic*. Folosind un astfel de serviciu, utilizatorii se pot vedea și auzi prin intermediul *webcam*-urilor. Astfel, conversațiile se desfășoară practic în timp real, utilizatorii având posibilitatea de a se angaja în interacțiuni directe, cu persoane reale. A se vedea și **microblogging** și **Seesmic**.

video sharing (*partajare video*). A se vedea **partajare video**.

vierme (*worm*). Program de calculator de tip *malware*, capabil de auto-replicare prin intermediul unei rețele (e.g., Internet). Utilizează rețeaua pentru a trimite copii în diverse noduri (calculatoare) ale acesteia. Spre deosebire de un virus de calculator, un vierme nu trebuie să se atașeze la un fișier existent (de obicei executabil) pe care să-l infecteze și cu ajutorul căruia să se răspândească. Un vierme poate oferi posibilitatea unui utilizator rău intenționat să preia controlul unui computer sau poate trimite suficiente (auto)copii pentru a determina un computer, o rețea sau un server de rețea să nu mai răspundă cererilor utilizatorilor legitimi. A se vedea și **malware** și **virus de calculator**.

viewer (*vizualizator*). Program care îi permite utilizatorului să vizualizeze fișiere cu un anumit format. Cele mai cunoscute programe de vizualizare sunt cele pentru fișierele grafice. De asemenea, larg răspândit este *Adobe Reader*, program oferit gratuit de Adobe Systems, utilizat pentru vizualizarea fișierelor PDF. A se vedea și **Adobe Systems**.

virus cu acțiune directă (*direct action virus*). Virus care se încarcă imediat în memorie, infectează fișiere și apoi se (auto)descarcă. A se vedea și **virus de calculator**.

virus de boot (*boot virus*). Virus care infectează sectorul de boot al unui disc fix sau al unei dischete. Orice încercare de a realiza bootarea de pe un disk astfel infectat va determina virusul să devină activ în memorie. Din acest moment, de fiecare dată când se va realiza bootarea sistemului, virusul va deveni activ în memorie. A se vedea și **virus de calculator**.

virus de calculator (*computer virus*). Program care se autoreproduce și infectează un calculator fără permisiunea sau știința proprietarului. Virușii provoacă prejudicii atât în partea software, cât și în cea hardware a calculatorului. Adesea, termenul „virus” este utilizat eronat pentru a desemna alte tipuri de programe *malware* (e.g., *adware* și *spyware*), care nu se pot autoreproduce.

În general, virușii informatici se împart în două mari categorii: viruși hardware

(afectează hard-disk-ul și memoria) și viruși software (afectează datele și programele existente pe hard-disk). În 1983, Fred Cohen a dat prima definiție a virusului de calculator: „un program care poate afecta alte programe de calculator”. În 1991 a fost „lansat” *Tequila*, primul virus polimorf cu răspândire pe scară largă. Virușii de acest tip sunt cu greu detectați de programele antivirus, deoarece își schimbă modul de acțiune cu fiecare nouă infecție.

Virușii de calculator folosesc mai multe tipuri de gazde. Când au apărut, ținta virușilor era reprezentată de fișierele executabile care făceau parte dintr-o aplicație, precum și sectoarele de boot ale discurilor *floppy*. Astăzi, majoritatea virușilor sunt încorporați în mesajele de *e-mail* ca atașamente. Este exploatarea astfel curiozitatea destinatarului de a deschide atașamentul infectat. În cazul fișierelor executabile, codul malițios este executat atunci când se rulează fișierul respectiv. În mod obișnuit, programul-gază funcționează în continuare după ce a fost infectat. Virușii se pot răspândi pe alte calculatoare atunci când programul sau documentul infectat este transferat pe acestea. A se vedea și **antivirus** și **dezinfecție**.

virus de macro (*macro virus*). Virus informatic inclus ca macro într-un document. Multe aplicații (e.g., *Microsoft Word*, *Excel*) suportă limbaje macro puternice. Acestea permit încapsularea de macro-uri în documente, executate

de fiecare dată când sunt deschise documentele. Unele dintre acestea pot fi, de fapt, secvențe virale. Un virus de macro odată activat poate redefini comenzi ale programului care a deschis documentul (e.g., salvarea sau tipărirea la imprimantă). Cel mai des este modificată comanda de salvare a fișierelor, astfel încât acestea vor fi salvate numai ca șabloane, dar cu extensia specifică documentelor. În cele din urmă, virusul infectează fișierul șablon global, virușând apoi prin intermediul acestuia orice document care se va deschide. A se vedea și **virus de calculator**.

virus polimorf (*polymorphic virus*). Virus care își schimbă forma cu fiecare fișier pe care îl infectează. Deoarece nu au un tipar binar consistent, virușii de acest tip sunt greu de identificat. A se vedea și **virus de calculator**.

Visual Basic. Abreviat VB, este un limbaj de programare de uz general, interpretat, orientat pe obiect, produs de compania Microsoft începând cu 1991. Este derivat din limbajul Basic și inclus în pachetul *Visual Studio*. VB este deosebit de popular datorită interfeței grafice destul de simplă pe care o folosește. Pune la dispoziția programatorilor un număr mare de clase de uz general. VB poate utiliza un număr mare de tipuri de baze de date, direct sau prin intermediul interfeței ODBC. A se vedea și **Microsoft Visual Studio** și **ODBC**.

Visual Basic .NET. Limbaj de programare orientat pe obiect, implementat în *.NET Framework*. Începând cu versiunea 8.0 (lansată în anul 2005), compania Microsoft a decis să elimine terminația .NET, limbajul fiind denumit *Visual Basic*. A se vedea și **.NET Framework**.

Visual C++. Mediu de dezvoltare integrat (IDE) produs de compania Microsoft, utilizat pentru crearea aplicațiilor în limbajul de programare C++. Este inclus în *Microsoft Visual Studio*. A se vedea și **IDE** și **Microsoft Visual Studio**.

Visual FoxPro. Abreviat adesea VFP, este un limbaj de programare procedural, orientat pe obiect, produs de Microsoft. Este derivat din *FoxPro*, limbaj dezvoltat de Fox Software în 1984. Ultima versiune de *FoxPro* (2.6) poate fi utilizată pe DOS, Windows, Mac OS și UNIX. Visual FoxPro 3.0 poate fi utilizat pe Windows și Mac OS, iar versiunile ulterioare pot fi utilizate numai pe platforma Windows.

Visual FoxPro este integrat cu propriul motor de baze de date relaționale, care extinde suportul *FoxPro* pentru interogări SQL și manipularea datelor. VFP are un număr mare de facilități, motiv pentru care nu necesită utilizarea suplimentară a unui limbaj de programare de uz general. Poate fi utilizat atât pentru crearea unor aplicații tradiționale complexe, cât și pentru scrierea unor aplicații Web. A se vedea și **bază de date relațională** și **limbaj de programare**.

Visual J++. IDE inclus în *Microsoft Visual Studio* până în versiunea 6.0 a acestui pachet, utilizat pentru dezvoltarea aplicațiilor în limbajul Java. A se vedea și **IDE** și **Microsoft Visual Studio**.

Visual J#. Limbaj dezvoltat de compania Microsoft, a cărui sintaxă este apropiată de cea a limbajului Java. *Visual J#* este inclus în suita *Microsoft Visual Studio*, fiind creat cu scopul de a-i ajuta pe programatori să treacă de la J++ (e.g., Java) la .NET Framework. *Visual Studio 2005* a fost ultima versiune care a inclus *Visual J#*. A se vedea și **Microsoft Visual Studio**.

viteza CPU (*CPU speed*). Numită și *rată de ceas* (*clock rate*), este o măsură a capacității de prelucrare a datelor de către CPU (microprocesor). Are ca unitate de măsură Hz (*hertz*). CPU-urile actuale au viteze de ordinul GHz (Gigahertz; 1 GHz = 10^9 Hz). A se vedea și **CPU**.

vizitator unic (*unique visitor*). Vizitator al unui site contorizat o singură dată pe baza adresei IP, indiferent de numărul paginilor (aparținând site-ului) pe care le vizitează. Numărul real de vizitatori ai site-ului este egal cu numărul vizitatorilor unici, și nu cu numărul paginilor accesate de vizitatori.

VML (*Vector Markup Language*). Format vectorial deschis, dezvoltat de Microsoft și recunoscut numai de browserul *Internet Explorer*. Consorțiul Web a publicat – în mai 1998 – o notă

referitoare la VML. A se vedea și **formate vectoriale**.

vocabular (*vocabulary*). Într-un limbaj de programare, ansamblul alcătuit din setul de caractere, identificatorii, separatorii și tipurile de comentarii utilizate. Setul de caractere reprezintă totalitatea caracterelor cu ajutorul cărora se poate scrie un program în limbajul de programare respectiv. Identificatorii furnizează modalități de denumire a datelor (*i.e.*, variabile și constante), funcțiilor, claselor, cuvintelor-cheie etc., fiind formați în mod obișnuit din următoarele caractere: cifre, litere și caracterul special „liniuță de subliniere”. Separatorii delimitează între ei identificatorii și, în mod obișnuit, sunt spațiul (*blank*), caracterul „punct și virgulă” (*semicolon*) și virgula (*coma*). Comentariile sunt utilizate pentru a face mai ușor de înțeles un program. În mod obișnuit, un comentariu este precedat de două caractere *slash* (*//*). A se vedea și **limbaj de programare**.

vodcast (*vodcast*). Combinație între *video* și *podcast*, desemnează o serie de înregistrări video care pot fi descărcate și vizualizate pe calculatoare sau dispozitive mobile într-o manieră similară cu înregistrările audio oferite prin intermediul *podcast*-urilor. A se vedea și **podcast**.

VoiceThread. Instrument Web 2.0 utilizat pentru crearea *online* a prezentărilor multimedia interactive (<http://voicethread.com/>). A fost

dezvoltat de Steve Muth, un inventator de jucării din New York, și de Ben Pappel, un dezvoltator din Florida. Versiunea beta a serviciului a fost lansată în martie 2007. Scopul declarat al acestuia este de a face simplă și accesibilă colaborarea asincronă prin intermediul vocii în jurul conținutului media. Folosirea variantei gratuite *VoiceThread* presupune crearea unui cont de utilizator. În acest scop sunt solicitate numele utilizatorului, adresa de e-mail și parola. Un cont gratuit permite crearea a cel mult trei prezentări, utilizând un spațiu de stocare de până la 75 MB. Alături de varianta gratuită, *VoiceThread* le este oferit utilizatorilor în alte patru variante comerciale: *Pro*, *Business*, *Higher Ed* și *K-12*. Varianta K-12 este adresată profesorilor din învățământul liceal, oferind facilități multiple la un preț rezonabil.

Aplicația *VoiceThread* le oferă utilizatorilor posibilitatea de a crea așa-numitele *voicebooks* (sau *audiobooks*), într-un mod similar celui în care sunt create un post într-un blog sau o pagină într-o rețea socială. În vreme ce un blog este – mai ales – scris, un *voicebook* oferă beneficiul înregistrării unei conversații de grup.

În prezentările *VoiceThread* pot fi incluse imagini, documente și secvențe video, precum și comentarii, folosind: vocea (prin intermediul unui microfon), textul, telefonul, fișiere audio, fișiere video (prin intermediul unui *webcam*).

VoiceThread suportă mai mult de zece

fișiere standard și formate grafice, dintre care pot fi amintite: fișiere PDF; fișiere *Microsoft Word*, *Excel* și *Power Point* (inclusiv formatele utilizate în *Office 2007*); imagini (JPEG, BMP, GIF, PNG) și variate formate video.

Exportul unei prezentări *VoiceThread* permite descărcarea unei arhive video care este replica exactă a prezentării, incluzând toate elementele multimedia și comentariile acesteia. Fișierele video astfel obținute pot fi stocate pe calculatorul personal, vizualizate *offline*, înregistrate pe un CD/DVD sau descărcate pe un MP3 player (care poate reda fișiere video) sau pe un telefon mobil.

Creatorul unei prezentări *VoiceThread* are posibilitatea de a modera comentariile adăugate acesteia, stabilind care dintre ele vor fi vizualizate și de alte persoane. De asemenea, poate să aleagă cu ușurință cinci dintre cele mai bune comentarii și să le includă într-o relatare succintă.

Securitatea prezentărilor *VoiceThread* ia diverse forme: acestea pot fi publice, private sau între aceste două limite. Setările de securitate îi permit creatorului unei prezentări să aleagă ceea ce pot vedea alți utilizatori. De asemenea, setările amintite pot fi adăugate, modificate sau eliminate.

Prezentările realizate folosind aplicația *VoiceThread* sunt folosite în domeniul educațional în scopul creării de către educatori a unor cursuri comentate, accesibile *online*. Cursurile pot fi partajate cu studenții, care – la rândul lor – au posibilitatea de a le adăuga

comentarii. De asemenea, prezentările *VoiceThread* permit oricărui student al unei clase să înregistreze comentarii audio despre ideile și experiențele sale pe care le consideră importante. Urmărind prezentările, studenții vor conștientiza importanța aportului lor la crearea acestora, precum și la derularea evenimentelor/proiectelor asociate. Un beneficiu esențial al utilizării *VoiceThread* este acela că studenții au posibilitatea să efectueze activități multiple în legătură cu un obiect multimedia dat (*i.e.*, prezentarea): citesc (urmăresc prezentarea), scriu (crează comentarii de tip text), vorbesc (înregistrează comentarii folosind microfonul sau telefonul) și ascultă (urmăresc prezentarea). A se vedea și **prezentare multimedia interactivă**.

VoIP (*Voice over IP*). Numit și telefonie IP sau telefonie Internet, este o tehnologie care permite transmiterea conversațiilor umane sub formă digitală (ca pachete) prin legături de date IP (*Internet Protocol*) sau prin rețele în care este utilizat acest protocol, spre deosebire de rețelele publice tradiționale bazate pe comutarea de circuite. Telefonie IP folosește comutarea de pachete pentru transmiterea conversațiilor umane. În acest scop, semnalul vocal analogic este captat de un microfon aflat în aparatul telefonic sau conectat la placa de sunet a calculatorului. Acest semnal este eșantionat, digitalizat și împachetat. Pachetele obținute sunt transferate prin protocolul IP.

În cazul în care captarea semnalului s-a făcut cu un calculator și destinatarul utilizează un telefon clasic, pachetele IP provenite de la expeditor sunt transferate prin IP către o poartă (*gateway*) VoIP care realizează legătura cu rețeaua publică de telefonie comutată (PSTN, *Public Switched Telephone Network*) la care este abonat destinatarul. Rezultă că poarta VoIP are rol de interfață între două rețele care folosesc metode diferite de comunicare (*i.e.*, comutarea de pachete și comutarea de circuite).

În cazul în care comunicarea se face între două telefoane, semnalul audio analogic parcurge (o parte din) rețeaua PSTN, ajunge la poarta VoIP, unde este transformat în pachete IP. Acestea sunt trimise prin IP către o altă poartă VoIP (mai apropiată de destinatar), care le transformă în semnal analogic transportat prin PSTN către destinatar.

În sfârșit, în cazul în care comunicarea se face între două calculatoare, pachetele IP sunt trimise direct între corespondenți (nefiind necesare porți VoIP).

Avantajul principal al telefoniei IP este costul redus comparativ cu telefonie clasică. Iată și alte avantaje: utilizatorii pot folosi serviciul VoIP indiferent de locul unde se află; datorită utilizării unei conexiuni partajate, pot fi asigurate servicii suplimentare în paralel cu apelul VoIP (transmiterea de imagini, materiale video, fișiere sau mesaje text).

VoIP derivă din eforturile *VoIP Forum*, un grup format din producători importanți de echipamente (*e.g.*, Cisco, VocalTec, 3Com și Netscape) de a promova standardul ITU-T H.323 pentru trimiterea semnalelor audio și video folosind IP în Internet și intranet. Alături de IP, tehnologia VoIP utilizează RTP (*Real Time Protocol*), pentru a asigura livrarea pachetelor în timp real. Proliferarea furnizorilor de VoIP a început în 2004. A se vedea și **comutare de circuite, comutare de pachete și IP**.

VPN (*Virtual Private Network*). A se vedea **rețea virtuală privată**.

VRML (*Virtual Reality Modeling Language*). Cunoscut și ca *Virtual Reality Markup Language* înainte de anul 1995, este un limbaj utilizat pentru crearea de grafică interactivă 3D, proiectat special pentru a fi utilizat în Web. VRML a fost creat în 1994 de Mark Pesce și Tony Parisi, ca un subset al formatului *Inventor File Format* al companiei Silicon Graphics. Extensiile de fișier utilizate pentru formatul VRML sunt *.wrl* (fișier text) și *.wrz* (fișier comprimat), iar tipurile MIME folosite sunt `model/vrml`, `x-world/x-vrml` și `application/x-cc3d`.

W

W3C. A se vedea **Consortiul World Wide Web**.

Wall, Larry (n. 1954). Lingvist și programator american (<http://www.wall.org/~larry/>), cunoscut mai ales ca inventator al limbajului de programare Perl (1987). A se vedea și **Perl**.

wallpaper (*tapet*). Imagine folosită ca fundal în interfețele grafice cu utilizatorul, care poate fi aleasă de utilizator. Denumirea este folosită în sistemele de operare Windows. În Mac OS X este folosit termenul *desktop picture*.

WAN (*Wide Area Network*). Rețea de comunicații care conectează zone aflate la mari distanțe din punct de vedere geografic. A se vedea și **LAN**, **MAN** și **rețea de calculatoare**.

WAP (*Wireless Application Protocol*). Standard internațional deschis pentru stratul aplicație în rețelele de comunicații fără fir. Este utilizat în special pentru a asigura accesul la Web-ul mobil prin intermediul terminalelor mobile pe care sunt instalate browsere WAP (e.g., telefoane, PDA-uri). A se vedea și **browser WAP**.

Wapedia. Serviciu gratuit dezvoltat de Florian Amrhein, care permite accesarea site-ului *Wikipedia* prin intermediul dispozitivelor mobile, având și o versiune în limba română. Pentru accesarea serviciului este necesar un terminal cu acces la Internet, ca de exemplu un telefon mobil sau un PDA cu un browser WAP sau Web. Articolele solicitate vor fi afișate diferit în funcție de performanțele grafice ale terminalului.

Versiunea în limba română a serviciului *Wapedia* poate fi accesată la adresele <http://wapedia.mobi/ro/> (la care este detectat automat terminalul folosit) sau <http://wapedia.mobi/> (utilizată pentru PDA și *Smartphone*). Deoarece browserele actuale pentru PC vizualizează pagini WAP, serviciul *Wapedia* poate fi testat folosind un calculator personal. Ulterior apariției *Wapedia*, a fost lansat și site-ul oficial *Wikipedia* pentru terminale mobile (<http://mobile.wikipedia.org/>). A se vedea și **WAP** și **Wikipedia**.

WAV. Acronim pentru *Waveform* (cunoscut și ca *Audio for Windows*), este un format de fișier utilizat pe platforma Windows pentru stocarea fișierelor audio, dezvoltat de Microsoft și IBM. Extensia de fișier este *.wav*, iar tipurile MIME folosite sunt *audio/wav*, *audio/wave* și *audio/x-wav*.

WBMP (*Wireless BitMap*). Format grafic *raster*, optimizat pentru dispozitivele mobile. WML utilizează acest format pentru includerea imaginilor pe site-urile WAP. A se vedea și **formate raster**, **WAP** și **WML**.

WBT (*Web-based training, pregătirea bazată pe Web*). A se vedea **tipuri de e-Learning**.

Web. A se vedea **World Wide Web**.

Web 2.0. Concept care se referă la etapa actuală și imediat următoare a dezvoltării Web-ului. Reflectă schimbările tehnologice și, mai ales, reconsiderarea rolului și importanței utilizatorilor în ceea ce privește crearea conținutului internet.

Conform autorilor Richard MacManus și Joshua Porter, conceptul Web 2.0 este „...o viziune a Web-ului în care informațiile sunt divizate în microunități de conținut care pot fi distribuite într-o mulțime de domenii”. În opinia lui Jeff Clavier, noul concept poate fi sintetizat astfel: „...date și servicii deschise, experiență bogată a utilizatorului și livrare cu costuri reduse”.

Dario de Judicibus, analist în cadrul IBM, este de părere că „Web 2.0 este un mediu orientat spre cunoaștere, în care interacțiunile dintre oameni creează informații publicate, administrate și folosite în cadrul aplicațiilor de rețea de pe o arhitectură bazată pe servicii.” În opinia lui Tim O'Reilly, „Web 2.0 reprezintă revoluția *business* în industria calculatoarelor, cauzată de transformarea Internetului într-o platformă, și încercarea de a descoperi principiile succesului pe această platformă.” Conform noii imagini a Web-ului, acesta constituie o platformă *software* în care utilizatorul își controlează propriile date, punându-le și la

dispoziția altora, prin intermediul unor instrumente colaborative.

Aplicațiile Web 2.0 au o popularitate deosebită, oferind funcții colaborative și sociale care le permit utilizatorilor să creeze ei înșiși conținut nou. Web 2.0 reprezintă o anticipare a etapei următoare de dezvoltare a Web-ului (*Semantic Web*). În opinia lui Tim Berners-Lee, „Web-ul este mai mult o creație socială decât una tehnică.” (*Weaving the Web*, 1999).

O serie de tipuri de aplicații Web 2.0 utilizează tehnici și tehnologii dezvoltate încă din anii '90 ai secolului trecut (e.g., mediatizarea conținutului nou inclus în site-uri, servicii Web). Rezultă că noul concept are, mai degrabă, un rol integrator, oferind modalități de utilizare eficientă a unor tehnologii existente. Totuși, unele dintre tehnologiile utilizate în aplicațiile Web 2.0 au fost dezvoltate relativ recent (e.g., blog, microblog, wiki, podcast), ceea ce sugerează că noul concept este într-o continuă dezvoltare.

Web 2.0 poate fi perceput ca reprezentând tranziția modelelor de site-uri de la insule izolate de informație la platforme computaționale interconectate, care funcționează ca software local.

Principiile noului Web sunt: 1) interactivitatea – este un principiu de bază al Web 2.0, definind modul în care utilizatorii interacționează cu site-urile și modul în care serverele comunică între ele; aplicațiile Web 2.0 au o

interactivitate mult mai mare în raport cu site-urile Web 1.0, datorită utilizării unor tehnici/tehnologii specifice (e.g., AJAX, FLEX), iar serverele comunică unele cu celelalte și fac schimb de date prin intermediul serviciilor Web; 2) *software*-ul ca serviciu – acest principiu, cunoscut sub numele de SOA (*Service Oriented Architecture*) este un model arhitectural în care funcționalitatea este divizată în unități distincte, numite servicii, distribuite într-o rețea, combinate între ele și refolosite pentru a crea aplicații; aceste servicii comunică între ele prin pachete mici de date, numite mesaje; 3) colaborarea – succesul noilor aplicații este datorat în cea mai mare măsură chiar utilizatorilor care le folosesc; astfel, proiecte precum *Wikipedia*, *YouTube*, *Slideshare*, *digg*, *delicious* și altele nu ar fi putut atinge gradul actual de dezvoltare fără ajutorul numărului foarte mare de utilizatori care contribuie activ la crearea și clasificarea informațiilor.

Cele mai populare tipuri de aplicații Web 2.0 sunt blogurile, wiki-urile, site-urile de tip *social networking* (socializare în rețea), serviciile pentru *social bookmarking* (marcare colaborativă a resurselor) și site-urile pentru partajarea conținutului (documente, fotografii/imagini, secvențe audio și clipuri video).

Web 3.0. Denumit și *Web semantic*, reprezintă următoarea etapă de dezvoltare a Web-ului. Conform lui

Tim-Berners Lee, Web-ul semantic (ale cărui baze sunt puse astăzi) „reprezintă o rețea consistentă și logică a tuturor resurselor de pe Web”; accentul este pus pe interpretarea datelor de către calculatoare, și nu pe reprezentarea lor. În acest sens, Web-ul trebuie să constituie un set de relații stabilite între obiecte identificabile astăzi prin identificatori uniformi de resurse (URI), iar în viitor prin nume stabilite de utilizatori. Web-ul semantic va fi parcurs nu numai de către utilizatori umani, folosind browsere, ci și de programe specializate care: accesează selectiv (caută și filtrează) resurse Web; efectuează anumite prelucrări ale acestora; generează (eventual) alte resurse Web.

Aspectele reprezentative ale Web-ului semantic sunt următoarele: folosește date accesibile calculatoarelor (datele pe Web sunt definite și legate în modalități care să permită calculatoarelor să le utilizeze nu numai pentru reprezentare, ci și, mai ales, pentru căutarea automată și integrarea lor în diferite aplicații); utilizarea agenților inteligenți (scopul Web-ului semantic este acela de a-l face mai accesibil calculatoarelor, pentru a permite agenților inteligenți să manipuleze informații pertinente); adnotare mai bună (Web-ul semantic va asigura o adnotare mai bună pentru documente, în formate care să faciliteze căutarea automată și stabilirea legăturilor între acestea); rezolvarea unor sarcini complexe (Web-ul semantic va permite aplicațiilor să colecteze automat informații din diverse surse, să le

integreze și proceseze, dar și să coopereze cu alte aplicații în scopul rezolvării unor sarcini dificile precizate de operatori umani); servicii Web omniprezente (serviciile Web se vor extinde prin asigurarea unor proceduri automate accesibile prin intermediul unor agenți software).

Web-ul semantic are o arhitectură funcțională, bazată pe specificarea unor limbaje la diferite niveluri. Limbajele trebuie să satisfacă o serie de cerințe impuse de aplicațiile dezvoltate la nivelul respectiv. Se deosebesc: nivelul metadatelor (oferă cadrul general pentru adnotarea și metadescrierea resurselor Web, prin intermediul limbajului RDF și al unor vocabulare de metadata); nivelul schemelor (oferă posibilitatea specificării de ontologii simple, în scopul descrierii ierarhice a conceptelor prin intermediul cărora sunt modelate resursele Web); nivelul logic (introduce ontologii complexe).

Web Authoring Tool (WAT). Instrumente specifice care permit construirea și legarea paginilor Web pentru a crea un site. Sunt succesoarele editoarelor HTML simple. Unele WAT permit crearea unor site-uri Web cu un număr foarte mare de pagini, realizarea de animații interactive și conexiuni cu baze de date, fără ca utilizatorul să scrie vreun rând de cod.

Iată câteva instrumente WAT cunoscute: *Amaya*, *Dreamweaver*, *FrontPage*, *GoLive*, *HotDog*. A se vedea și **Dreamweaver**.

Web CMS (Web CMS). Tip de CMS folosit pentru stocarea, controlul și publicarea unor documente specifice, așa cum sunt știrile, articolele noi, manualele tehnice, ofertele de muncă, informațiile despre angajați, imaginile cu produse etc. Iată câteva dintre Web CMS-urile cele mai cunoscute: *Drupal*, *eZ Publish*, *Joomla*, *Mambo*, *osCommerce*, *PHP-Fusion*, *PHP Nuke*, *Plone*, *PostNuke*, *WordPress*, *XOOPS*. A se vedea și **CMS**.

webmaster (webmaster). Creatorul și administratorul unui site Web. Acesta poate avea și alte atribuții, ca de exemplu: oferă răspunsuri la mesajele utilizatorilor (primite inclusiv prin intermediul poștei electronice), revizuește și actualizează conținutul site-ului. A se vedea și **site Web**.

white hat SEO (white hat SEO). Categorie de tehnici SEO conforme cu o serie de norme specificate de dezvoltatorii motoarelor de căutare, fiind recomandate de aceștia. Tehnicile din această categorie tind să aducă rezultate pozitive pe termen lung în ceea ce privește poziționarea cât mai bună a site-ului în lista de rezultate returnate de motoarele de căutare, fiind utile inclusiv acestora din urmă. A se vedea și **black hat SEO**, **motor de căutare** și **SEO**.

Web hub (Web hub). Site Web B2B (*business-to-business*) pentru o anumită industrie. Furnizează un loc de întâlnire pentru cumpărători și vânzători într-un domeniu specific. Poate solicita plata

unei taxe de tranzacționare pentru fiecare act de vânzare-cumpărare. Este cunoscut și ca *portal vertical* (sau *vortal*).

Web semantic (*semantic Web*). A se vedea **Web 3.0**.

widget vizual (*GUI widget*). Numit și *control*, prescurtare pentru *window gadget*, este un element vizual de bază al unei interfețe grafice (GUI), utilizat pentru afișarea informațiilor într-un aranjament care poate fi modificat de utilizator. Un *widget* facilitează un tip specific de interacțiune cu utilizatorul. Denumirea a fost utilizată prima dată în 1988 pentru elementele interfeței grafice dezvoltate în *Athena Project* (mediu pentru uz educațional creat de MIT, DEC și IBM). Aceste *widget-uri* nu trebuie confundate cu cele pentru *desktop* sau pentru *Web*.

Widget-urile pot fi grupate în următoarele categorii: selecția și afișarea colecțiilor de date (*e.g.*, buton, *slider*, listă derulantă, *spinner*, meniu, bară de instrumente), navigare (*e.g.*, tab, *scrollbar*), introducerea textului (*e.g.*, casetă de text), ieșire (*e.g.*, etichetă, *tooltip*, bară de stare, bară de progres), containere (*e.g.*, fereastră). *Widget-urile* sunt conținute în pachete denumite *widget toolkits* și pot fi combinate pentru a construi aplicații cu interfață grafică. Cele mai multe sisteme de operare includ astfel de pachete. Fiecare categorie generală de *widget-uri* este definită ca o clasă POO, iar *widget-urile* incluse sunt derivate din ea prin moștenire. A se vedea și **POO** și **widget toolkit**.

widget engine (*motor pentru widget-uri*). Serviciu software care rulează și afișează *widget-uri* pe interfețele grafice. A nu se confunda cu *widget toolkit*. Iată câteva tipuri de *widget engine*: *desktop widgets*, *mobile widgets*, *Web widgets*. A se vedea și **widget pentru desktop** și **widget pentru Web**.

widget pentru desktop (*desktop widget*). Instrument interactiv afișat pe *desktop*, care furnizează un serviciu specific (*e.g.*, știri recente, starea vremii, afișarea orei, calendar, dicționar, calculator, vizualizator de imagini). *Widget-urile* de acest tip sunt găzduite de un motor de *widget-uri* (*widget engine*).

Iată câteva exemple de astfel de *widget-uri*: *gadget-urile* Microsoft pentru sistemul de operare Vista și serviciile *Windows Live*, *widget-urile dashboard* pentru Mac OS X, *portlet-urile* pentru *Google Desktop*, *widget-urile* Yahoo!. A se vedea și **widget engine**.

widget pentru Web (*Web widget*). Porțiune de cod portabilă care poate fi instalată și executată într-o pagină *Web* de către un utilizator fără a necesita o compilare suplimentară. În mod obișnuit, acest tip de *widget-uri* utilizează DHTML, JavaScript și *Flash*. A se vedea și **widget engine**.

widget toolkit (*pachet de widget-uri*). Denumit și *widget library* (bibliotecă de *widget-uri*), este un pachet de *widget-uri* care pot fi utilizate în proiectarea aplicațiilor cu interfață

grafică (GUI). Pachetul este furnizat în general de un sistem de operare sau de un manager de ferestre (*window manager*) și poate fi utilizat prin intermediul unui API. Iată câteva dintre pachetele de *widget*-uri cunoscute: GTK+, *XForms*, *Motif*, *Microsoft Foundation Class*, *Qt*, *Swing*, *Abstract Window Toolkit*. A se vedea și **API** și **widget**.

Wiener, Norbert (1894–1964). Matematician american, recunoscut ca întemeietor al ciberneticii, domeniu prezentat în lucrarea sa *Cybernetics or Control and Communication in the Animal and the Machine* (1948). A adus contribuții importante în domeniile ingineriei electronice, comunicațiilor electronice, informaticii și sistemelor de comandă și control.

wiki. Colecție de pagini Web proiectate astfel încât să asigure oricărui individ posibilitatea de a le accesa în scopul de a realiza modificări în conținutul existent, precum și de a crea conținut nou. Simplificat, un *wiki* este un site Web care le permite vizitatorilor să adauge și să elimine conținut, precum și să-l modifice.

Ward Cunningham, dezvoltatorul primului site wiki, denumit *WikiWikiWeb*, l-a descris ca fiind „cea mai simplă bază de date *online*”. Termenul *wiki* este echivalentul hawaian al cuvântului englezesc *fast* (rapid), reprezentând un *backronym* (sau *bacronym*), *WikiWiki* fiind o duplicare a aceluiași cuvânt.

Ward Cunningham a început dezvoltarea site-ului *WikiwikiWeb* în 1994 și

l-a plasat *online* în data de 25 martie 1995 (<http://c2.com/>). Acesta a dezvoltat ideile lui Vannevar Bush referitoare la permisiunea acordată utilizatorilor de a comenta sau de a efectua schimbări în textul scris de altcineva.

Numele site-ului a fost ales de Cunningham după ce acesta și-a amintit că un angajat al aeroportului internațional din Honolulu i-a sugerat să utilizeze autobuzul rapid *WikiWiki*, care făcea legătura între terminalele acestui aeroport. Ceea ce face din *wiki* un instrument unic este posibilitatea ca mai mulți utilizatori să vadă și să editeze colaborativ același document, în același „loc”. *Wiki*-ul – la fel ca și autobuzul lui Cunningham – le permite utilizatorilor să împartă același spațiu (aceeași pagină) și să vadă același lucru (textul pe care îl pot edita), în același timp.

La baza funcționării *wiki*-urilor stau următoarele principii: oricine poate edita orice; utilizează un limbaj de marcare simplificat; conținutul nu are „identitate” și nu este niciodată finalizat.

Indiferent de limbajul utilizat pentru crearea acestora, majoritatea *wiki*-urilor oferă câteva funcții de bază: editare, istoric, schimbări recente, căutare, sandbox. *Wiki*-urile sunt organizate utilizând o structură ierarhică plată (simplă), adică una în care nu există secțiuni și subsecțiuni. O consecință a acestui fapt este aceea că într-un *wiki* URL-urile sunt destul de simple (*i.e.*, nu au o lungime excesivă). De

exemplu, pe *Wikipedia*, informațiile despre J.S. Bach se găsesc la adresa <http://en.wikipedia.org/wiki/Bach>.

Un exemplu de utilizare a *wiki*-urilor este domeniul afacerilor, în care acestea sunt folosite pentru a furniza (în intranetul firmelor) sisteme de management al cunoștințelor. Într-o întreprindere pot fi create mai multe *wiki*-uri, corespunzătoare unor echipe, proiecte sau departamente diferite. Un *wiki* pentru întreprindere este construit prin implementarea unor funcționalități care răspund nevoilor respectivului tip de organizație, incluzând și elementele de securitate necesare pentru protecția informațiilor, precum și permisiuni de acces care le oferă angajaților posibilitatea de a accesa/edita propriile secțiuni sau pagini. *Wiki*-ul poate fi legat la alte servicii ale întreprinderii via autentificare LDAP.

Există numeroase avantaje ale utilizării *wiki*-urilor, dintre care pot fi amintite următoarele: 1) utilizatorii nu au nevoie de un software suplimentar, putând naviga, citi sau schimba conținutul acestor site-uri folosind un browser Web (astfel, pentru folosirea *wiki*-urilor, utilizatorii nu trebuie să instaleze *applet*-uri sau *plug-in*-uri); 2) nu sunt necesare cursuri extensive în care utilizatorii să învețe cum să utilizeze *wiki*-urile; problemele tehnice și necesitatea de a avea cunoștințe specifice pentru utilizarea unui instrument de comunicație sunt reduse la minimum; paginile pot fi editate *on fly*, iar textul este formatat utilizând câteva reguli; de

asemenea, paginile unui *wiki* pot fi legate între ele; 3) simplitatea acestui tip de software le oferă comunităților posibilitatea de a-l utiliza pentru managementul proiectelor. Pe lângă simplitatea tehnologiei, este importantă și filozofia acesteia. Astfel, în cazul proiectelor bazate pe conceptul de *wiki*, discuțiile și colaborarea pot genera noi idei și modalități de abordare a acestora.

Iată câteva dintre modalitățile în care educatorii și studenții pot folosi *wiki*-urile: crearea rapidă a unor site-uri Web simple; crearea unor portofolii personale; crearea unor baze colaborative de cunoștințe; coordonarea cercetării și colaborare; urmărirea unui proiect de grup; coordonarea conferințelor; realizarea prezentărilor; crearea și menținerea unui sistem FAQ; sindicarea/agregarea resurselor Web; gestionarea documentelor unei clase; crearea unui spațiu de dezbateri și discuții pentru membrii unei clase; colectarea datelor. A se vedea și **fermă de wiki și software pentru wiki**.

WikiEducator. Comunitate de educatori care promovează colaborarea pentru: planificarea proiectelor educaționale legate de dezvoltarea conținutului liber; dezvoltarea conținutului liber pentru *e-Learning* chiar în cadrul *WikiEducator*; dezvoltarea de resurse educaționale deschise și de modalități pentru crearea acestora; dezvoltarea unei rețele de comunicare în direcția descoperirii de oportunități de finanțare.

WikiEducator este o comunitate sprijinită de *Commonwealth of Learning* (COL), care asigură și infrastructura tehnică a portalului. Valorile acestei comunități sunt: incluziunea socială și participarea tuturor oamenilor la societatea interconectată; libertatea tuturor educatorilor de a preda folosind tehnologii și conținut la alegerea lor, punând accent pe tehnologiile și standardele deschise; conținutul educațional este unic, iar tehnologiile și gradul de reutilizare a resurselor digitale utilizate pentru învățare pot fi îmbunătățite numai prin colaborare; orientarea către viitor, precum și lucrul colaborativ în scopul de a găsi cele mai bune și durabile soluții pentru *e-Learning*. A se vedea și **e-Learning** și **wiki**.

Wikipedia. Enciclopedie *online* și cel mai cunoscut *wiki* (<http://www.wikipedia.org/>), proiect al *Wikimedia Foundation*.

Wikipedia s-a născut ca o ramură a enciclopediei *online* *Nupedia.com*, fondată de Jimmy Wales. *Nupedia* a luat naștere în martie 2000, înființată ca o enciclopedie în care revizuirea/editarea cunoștințelor se făcea în șapte pași. Acest proces s-a dovedit destul de greoi, determinând generarea unui număr mic de articole. Soluția problemei menționate a fost dată de Larry Sanger, care – pe 10 ianuarie 2001 – a propus un site pentru „alimentarea” *Nupedia*, bazat pe tehnologia *wiki*. Astfel, utilizatorii postau articole care – după examinare – erau mutate pe *Nupedia*. Se spera ca utilizarea unui *wiki* să facă mult mai ușoară contribuția

utilizatorilor și să ducă la creșterea vitezei procesului de generare a articolelor. Nimeni nu se aștepta atunci ca *Wikipedia* să înlocuiască *Nupedia*, deși acest lucru s-a întâmplat destul de repede.

La un an după ce a fost lansată, *Wikipedia* „adunase” circa 18 000 de articole. În septembrie 2003, la închiderea *Nupedia*, pe *Wikipedia* erau stocate peste 160 000 de articole scrise de voluntari. În trei ani și jumătate, *Nupedia* a „produs” numai 24 de articole. Acesta este un exemplu remarcabil referitor la puternicul impact pe care îl are deplasarea de la un proces de luare a deciziei formalizat, puternic centralizat, către unul mai informalizat și descentralizat. Astăzi, *Wikipedia* include peste 9 milioane de articole, în peste 200 de limbi (dintre care mai mult de 2 milioane în limba engleză). Informațiile postate sunt, majoritatea, de bună calitate, ceea ce vine în sprijinul utilizatorilor aflați în căutarea unor resurse informaționale utile. Proiectul *Wikipedia* în limba română (<http://ro.wikipedia.org/>) a prins viață pe 15 iulie 2003, având în 2009 peste 115 000 de articole.

Software-ul pe baza căruia a fost creată enciclopedia *Wikipedia* este *MediaWiki* (<http://www.mediawiki.org/wiki/MediaWiki>). Inițial, acesta a fost dezvoltat pentru *Wikipedia*, dar astăzi este utilizat și de alte proiecte ale *Wikimedia Foundation* (e.g., *Wikiquote*, *Wikinews*, *Wiktionary*, *Wikibooks*, *Wikisource*, *Wikispecies*, *Wikiversity*) sau de

companii, ca soluții interne pentru managementul cunoștințelor, precum și de multe alte *wiki*-uri. A se vedea și **MediaWiki** și **wiki**.

Wikispaces. Instrument *online* pentru crearea *wiki*-urilor, care oferă și găzduire pentru *wiki*-urile create (<http://www.wikispaces.com/>). Aplicația *Wikispaces* este rulată pe serverele *Tangient* (<http://tangient.com/>) din San Francisco, SUA, este scrisă în limbajul PHP și utilizează baze de date MySQL. Serviciul este pus la dispoziția utilizatorilor prin intermediul serverului *Web lighttpd* și utilizează software-ul de căutare *lucene*. Pe *Wikispaces*, *wiki*-urile pot fi create gratuit (acestea sunt publice), cât și în schimbul unei sume plătite lunar (în acest caz, *wiki*-urile pot fi și private). A se vedea și **software pentru wiki** și **wiki**.

WiMAX (*Worldwide Interoperability for Microwave Access*). Tehnologie pentru comunicații care asigură transmiterea fără fir a datelor în bandă largă (până la 3 Mbit/s) folosind o varietate de moduri de transport. WiMAX se bazează pe standardul IEEE 802.16 (denumit *Broadband Wireless Access*).

window manager (*manager de ferestre*). A se vedea **manager de ferestre**.

Windows. Sisteme de operare cu interfață grafică dezvoltate de compania americană Microsoft. A se vedea și **interfață grafică**, **Microsoft**, **Windows Vista** și **Windows XP**.

Windows Explorer. Utilitar Windows folosit pentru navigarea în structura arborescentă a dosarelor și fișierelor pe platforma Windows.

Windows Live. Fost MSN (*Microsoft Network*, <http://www.msn.com/>), este o colecție de servicii Internet (portal Web) oferite de Microsoft începând cu 1995 și reorganizate în 2006 sub brandul *Windows Live* (<http://home.live.com/>): *Windows Live Hotmail*, *Windows Live Messenger*, *Live Search Maps*, *Windows Live Search*, *Windows Live Spaces*, *Windows Live Alerts*, *Windows Live Groups*). A se vedea și **portal Web**.

Windows Live Gallery. Parte a sistemului de servicii *Windows Live* (<http://gallery.live.com/>). Este un centru de *gadget*-uri, extensii și *add-on*-uri pentru diverse servicii *Windows Live*, incluzând: *gadget*-uri, *Windows Live Toolbar*, *Bing*, *Windows Live Search*, *Windows Live Messenger*, *Windows Live Writer*.

Windows Live Hotmail. Cunoscut și ca *Hotmail*, fost *MSN Hotmail* (lansat în iulie 1996), este unul dintre primele servicii de poștă electronică bazate pe Web și al doilea după Yahoo! în ceea ce privește numărul de utilizatori (<http://www.hotmail.com>).

Windows Media Player (WMP). Player media digital dezvoltat de compania Microsoft, utilizat pentru rularea fișierelor audio și video, precum și pentru vizualizarea imaginilor pe PC-uri, PDA-uri și dispozitive mobile

(e.g., Tablet PC, Pocket PC) pe care sunt instalate sisteme de operare Microsoft Windows. De asemenea, *Windows Media Player* poate extrage/copia muzică de pe/pe CD-uri și „arde” CD-uri audio. Au fost realizate ediții și pentru platformele Mac OS X și Solaris, dar dezvoltarea acestora a fost oprită.

Formatele de fișier implicite suportate de WMP sunt WMV (*Windows Media Video*), WMA (*Windows Media Audio*) și ASF (*Advanced Systems Format*). De asemenea, WMP suportă formatul pentru playlist-uri WPL (*Windows Playlist*), bazat pe XML și dezvoltat de Microsoft.

Windows Mobile. Sistem de operare pentru dispozitive mobile, produs de compania Microsoft. Este rulat pe Pocket PC, telefoane inteligente (*smartphones*) și pe computerele de bord ale unor mărci de automobile. A se vedea și **Pocket PC** și **telefon inteligent**.

Windows Sidebar. Motor pentru *widget*-uri (*widget engine*) introdus în sistemul de operare Windows Vista. Afișează pe *desktop* o bară laterală (*sidebar*) care conține *gadget*-uri (*Microsoft Gadgets*). *Gadget* este denumirea folosită de Microsoft pentru *widget* (totuși, termenul a fost utilizat pentru prima dată în anul 1985, în contextul ingineriei software, de către dezvoltatorii sistemului de operare AmigaOS pentru calculatoarele *Amiga*). Sistemul de operare Microsoft Vista conține implicit 11 *gadget*-uri, dar pot

fi descărcate altele suplimentare folosind *Windows Live Gallery* (<http://gallery.live.com/>). A se vedea și **widget engine**.

Windows Vista. Linie de sisteme de operare utilizate pentru calculatoarele compatibile IBM PC, creată de compania Microsoft începând cu anul 2005. Vista este succesoarea liniei Windows XP.

Windows Vista este produs în mai multe ediții care sunt incluse în două categorii: *consumer* și *business*. Din prima categorie fac parte Windows Vista Starter Edition, Windows Vista Home Basic Edition și Windows Vista Home Premium. În categoria *business* sunt incluse Windows Vista Business, Windows Vista Enterprise și Windows Vista Ultimate. Toate edițiile, cu excepția Windows Vista Starter, suportă atât procesoarele pe 32 de biți, cât și cele pe 64 de biți.

Iată câteva dintre caracteristicile noi (care nu sunt prezente în versiunile anterioare ale sistemului de operare Windows): *Windows Aero* (noua interfață grafică), *Instant Search* (căutare mai rapidă), *Windows Sidebar* (panou transparent pe care este plasat *Desktop Applets*), *Backup and Restore Center*, *Windows Media Center* (facilitate inclusă anterior într-o versiune separată a Windows XP, denumită Windows XP Media Center), *Windows Mobility Center* (panou care include cele mai importante informații despre dispozitivul mobil – nivelul sonor, strălucire, schema de putere utilizată),

Parental Controls, Speech Recognition, Problem Reports and Solution. A se vedea și **Windows XP**.

Windows XP. Linie de sisteme de operare utilizate în calculatoarele personale compatibile IBM PC, creată de compania Microsoft începând cu anul 2001 și localizată astăzi într-un număr mare de limbaje.

Două ediții majore sunt Windows XP Home Edition (pentru utilizatorul mediu) și Windows XP Professional (pentru utilizatorii experimentați). Windows XP Media Center, lansat în 2002, include capabilități multimedia. Ediția Windows XP Professional x64 Edition suportă extensiile x86-64 ale arhitecturii Intel IA-32. Windows XP Tablet PC Edition este produsă pentru dispozitivele Tablet PC. Ediția Windows XP Embedded este destinată utilizării în dispozitive electrocasnice, ATM-uri, dispozitive medicale, terminale POS (*point-of-sale*). Windows XP Starter este o ediție cu preț scăzut vândută într-o serie de țări din Asia și America de Sud.

Există trei tipuri principale de licențe pentru Windows XP: Retail, Volume și OEM (*Original Equipment Manufacturer*). Toate trei sunt utilizate pentru Windows XP Professional (pentru versiunile pe 32 și 64 de biți). Windows XP Home Edition se vinde cu licențele Retail și OEM, în vreme ce edițiile Windows XP Tablet PC și Windows XP Media Center sunt produse exclusiv sub licențele Volume și OEM. A se vedea și **Windows Vista**.

wireless (*fără fir*). Caracteristică a comunicațiilor care se realizează fără utilizarea unor medii de transmisie de tipul cablurilor sau fibrelor optice. În acest caz, transmisia informațiilor se realizează prin intermediul undelor electromagnetice.

Wirth, Niklaus (n. 1934). Informatician de origine elvețiană, cunoscut pentru crearea unor limbaje de programare (Euler, Algol W, Pascal, Modula, Modula-2 și Oberon), cel mai popular dintre acestea fiind Pascal. În 1975 a publicat cartea *Algorithms + Data Structures = Programs*, care a dobândit o largă recunoaștere printre programatori. În 1984 a câștigat *Turing Award* (*Premiul Turing*). În 1995 a popularizat o așerțiune cunoscută mai târziu drept *legea lui Wirth: Software is getting slower more rapidly than hardware becomes faster* (Software-ul devine mai lent mai repede decât hardware-ul devine mai rapid). A se vedea și Pascal.

WMF (*Windows MetaFile*). Format grafic de tip *metafile*. A se vedea și **formate metafile**.

WML (*Wireless Markup Language*). Bazat pe XML, este un limbaj de marcare dezvoltat pentru a fi folosit în dispozitivele care implementează standardul WAP (e.g., telefoane mobile, PDA-uri). Precede utilizarea altor limbaje de marcare folosite acum cu WAP (e.g., HTML/XHTML). Specificațiile WML sunt dezvoltate de WAP Forum, un consorțiu alcătuit din Nokia, Phone.com, Motorola și Ericsson. WAP

Forum este membru al OMA. A se vedea și **OMA**, **WAP** și **XML**.

WordPress. Serviciu și software pentru *blogging*, valabil și în limba română (<http://ro.wordpress.com/>). Majoritatea facilităților sunt oferite gratuit. A se vedea și **blog**.

World Wide Web. Denumit, pe scurt, Web, este un sistem de documente de tip hipertext legate între ele și care pot fi accesate prin Internet. Documentele, stocate în diferite locații pe diverse servere, pot fi regăsite cu ajutorul unui sistem de identificare denumit URL (*Uniform Resource Locator*). Web-ul este unul dintre serviciile Internetului, alături de FTP, *e-mail*, telnet, chat etc. Webul a fost inventat în 1989 la *Centrul European de Cercetări Nucleare* (CERN) din Geneva (Elveția) de către Tim Berners-Lee și alți cercetători. A se vedea și **Berners-Lee, Tim**, **CERN**, **hipertext** și **URL**.

World Wide Web Virtual Library (WVL). Creat în 1991, la Geneva, de Tim Berners-Lee, este cel mai vechi director Web (<http://vlib.org/>). A fost dezvoltat de-a lungul anilor, incluzând în biblioteca principală circa 300 de sub-biblioteci. A se vedea și **director Web**.

Wozniak, Stephen Gary (n. 1950). Co-fondator, alături de Steve Jobs, al companiei Apple Computer (astăzi, Apple, Inc.). A creat computerele Apple I (1976) și Apple II (1977), având astfel o contribuție esențială la revoluția PC-urilor. Apple II a fost cel mai vândut

calculator personal până la începutul anilor 1980. Lui Wozniak îi este atribuit aforismul *Never trust a computer you can't throw out a window* (Să nu ai niciodată încredere într-un calculator pe care nu-l poți arunca pe fereastră). A se vedea și **Apple II**, **Apple, Inc.** și **Jobs, Steven Paul**.

WSDL (*Web Service Description Language*). Limbaj bazat pe XML care furnizează o modalitate de descriere standardizată a serviciilor Web, reprezentând o componentă importantă a triadei WSDL, UDDI, SOAP/XML-RPC.

Descrierea WSDL a unui serviciu Web este o colecție de metadate despre serviciul respectiv, care comunică: ce face acest serviciu (operațiunile pe care acesta le efectuează); unde este localizat (detalii ale adresei la care poate fi invocat); cum poate fi accesat (detalii ale formatelor de date și ale tipurilor de protocoale utilizate pentru accesarea operațiunilor). În esență, descrierea WSDL îi „spune” utilizatorului tot ceea ce are nevoie pentru a putea utiliza serviciul respectiv.

WSDL furnizează atât o descriere abstractă, cât și o descriere concretă a serviciilor Web, precum și modalitatea în care sunt legate elementele care constituie subiectele acestor descrieri. Descrierea abstractă include definiții ale tipurilor de date (utilizând sintaxa furnizată de XML Schema), ale mesajelor și tipurilor de port (interfețele serviciului Web). Descrierea concretă include locația în care este oferit serviciul Web, precum și protocolul

specific utilizat de client pentru a comunica cu acesta. A se vedea și **serviciu Web**.

WWW-Authenticate. Unul dintre anteturile răspunsului HTTP. Trebuie inclus în mesajele de răspuns cu codul de stare 401 (Unauthorized), cu scopul de a-i oferi clientului informațiile necesare pentru autentificare, relativ la resursa solicitată de acesta. A se vedea și **răspuns HTTP**.

WYSBYGI (*What You See Before You Get It, ceea ce vezi înainte de a obține*). Se referă la previzualizarea efectelor modificărilor făcute într-un sistem (e.g., aplicație) înainte de a realiza efectiv

schimbările respective. Astfel, utilizatorul are posibilitatea de a alege dacă va efectua sau nu schimbările previzualizate.

WYSIWYG (*What You See Is What You Get, ceea ce vezi este ceea ce vei obține*). Se referă la un sistem (e.g., aplicație) care îi permite utilizatorului să vadă documentul (e.g., text, pagină Web, prezentare) într-o formă foarte apropiată de cea finală și să-l editeze direct în această formă. Iată câteva categorii de aplicații WYSIWYG: programe pentru procesarea textului (e.g., *Word*), programe de editare grafică (e.g., *Adobe Photoshop*), editoare HTML (e.g., *Dreamweaver*).

X

X.Org.Project. Proiect *open source* al corporației educaționale non-profit *X.Org Foundation*, este implementarea de referință a sistemului de ferestre X Window System (<http://www.x.org/>). A se vedea și **sistem de ferestre și X Window System**.

x86. Cea mai de succes arhitectură de instrucțiuni folosită în CPU-urile utilizate de calculatoarele personale. A apărut pentru prima dată în 1978, în procesoarele Intel 8086, în care instrucțiunile foloseau 16 biți. Astăzi sunt fabricate procesoare x86 pe 32 și 64 de biți.

Arhitectura x86 a fost implementată în procesoarele produse de companiile Intel, AMD, Cyrix, VIA și multe altele; este omniprezentă astăzi pe PC-uri *desktop* și *laptop*, pe servere și stații de lucru. Un număr mare de sisteme de operare sunt utilizate pe aceste calculatoare: Microsoft Windows, Mac OS X, Linux, BSD, Solaris. A se vedea și **calculator personal**.

xajax. Bibliotecă PHP/JavaScript *open source*, utilizată pentru realizarea aplicațiilor Web bazate pe tehnica AJAX, folosind HTML, CSS,

JavaScript și PHP. Este posibilă actualizarea conținutului inclus în paginile componente fără ca acestea să fie reîncărcate (<http://www.xajaxproject.org/>). Spre deosebire de alte instrumente AJAX, *xajax* este destinată programatorilor care nu au cunoștințe avansate de JavaScript. A se vedea și **AJAX, CSS, JavaScript și PHP**.

XAML (*Extensible Application Markup Language*). Pronunțat *zamel*, este un format vectorial declarativ bazat pe XML, proprietar Microsoft, creat în scopul dezvoltării interfețelor cu utilizatorul pentru aplicațiile .NET Framework. Totodată, XAML este limbajul implicit al sistemului de operare Windows Vista pentru programarea interfețelor cu utilizatorul. A se vedea și **formate vectoriale**.

Xara Xtreme. Editor grafic vectorial 2D (fost *Xara Studio*) creat de compania Xara (<http://www.xara.com/>), achiziționată în 2007 de compania germană MAGIX AG. Are versiuni pentru platformele Windows (versiune comercială) și Linux (versiune *open source*).

XBM (*X Bitmap*). Format grafic *raster* monocrom, folosit pentru stocarea unor imagini *bitmap* (e.g., cursor, icon) utilizate de X Window System, acesta fiind principalul mediu de exploatare a interfețelor grafice cu utilizatorul în UNIX/Linux. A se vedea și **GUI și formate raster**.

Xerces. Procesor XML creat de *Apache XML Project*, cu variante scrise în Java, C++ și Perl (<http://xml.apache.org/>).

Xfig. Editor grafic vectorial *open source* care rulează sub X Window System, pe cele mai multe platforme compatibile UNIX (<http://www.xfig.org/>). Desenele realizate pot fi exportate în numeroase formate, raster (e.g., GIF, PNG, JPEG, PPM, XBM, PCX, TIFF) și vectoriale (e.g., EPS, SVG, CGM). A se vedea și **X Window System**.

XFree86. Implementare *open source* a sistemului de ferestre X Window System. Este dezvoltată (începând cu anul 1992) de organizația de voluntari *XFree86 Project, Inc* (<http://www.xfree86.org/>). A se vedea și **sistem de ferestre și X Window System**.

XHTML (*eXtensible HTML*). Limbaj declarativ de marcare, constituie o reformulare a HTML în termenii limbajului XML. Este dezvoltat de Consorțiul Web în scopul utilizării pentru marcarea conținutului furnizat în Web. Ca și limbajul XHTML, folosește marcate (*tag-uri*) descriptive în scopul definirii modului de prezentare și a structurii logice a unui document hipertext.

Prima versiune a XHTML a fost publicată pe 26 ianuarie 2000 și revizuită pe 1 august 2000, limbajul fiind specificat în trei variante, care definesc trei tipuri de documente: *XHTML 1.0 Strict* (documentele de acest tip nu conțin elemente sau atribute

considerate demodate); *XHTML 1.0 Transitional* (documentele de acest tip pot conține elemente HTML considerate demodate); *XHTML 1.0 Frameset* (acest tip de documente este folosit în situațiile în care se dorește utilizarea cadrelor).

A doua recomandare W3C XHTML este *XHTML Basic* (publicată pe 19 decembrie 2000), care include un set minimal de module XHTML. Versiunea este creată pentru a permite obținerea unor documente hipertext care vor fi accesate de clienții Web care nu suportă setul complet de module XHTML (e.g., telefoane mobile, PDA-uri).

A treia recomandare W3C XHTML este *Modularization of XHTML* (publicată pe 10 aprilie 2001), reprezentând o decompoziție a XHTML 1.0 într-o colecție de module abstracte care furnizează funcționalități specifice, precum și o implementare a acestora folosind limbajul DTD (*Document Type Definition*).

A patra recomandare W3C XHTML este *XHTML 1.1 – Module-based XHTML* (publicată pe 31 mai 2001). Aceasta definește un nou tip de documente XHTML, bazat pe modulele incluse în recomandarea anterioară și având scopul de a servi unor posibile extinderi viitoare ale familiei de tipuri de documente XHTML. Tipul de documente menționat este, de fapt, o reformulare a tipului *XHTML Strict* utilizând module XHTML.

Cele mai recente specificații XHTML sunt schițe de lucru și propuneri de recomandare referitoare la versiunea XHTML 2.0, care modifică structura și semantica unor module definite în *Modularization of XHTML*, incluzând și altele noi (*XML Events* și *XForms*).

De ce este recomandată utilizarea XHTML și nu a HTML? Familia de documente XHTML este proiectată pentru: a realiza (folosind exclusiv CSS) separarea prezentării de structura logică a conținutului propriu-zis, ceea ce determină o întreținere mai facilă, precum și creșterea accesibilității și interoperabilității pe diverse platforme; a extinde documentele XHTML prin includerea altor tipuri de conținuturi (e.g., ecuații matematice create utilizând limbajul MathXML, grafică vectorială realizată prin intermediul limbajului SVG); a permite introducerea unor noi module (conform specificațiilor *Modularization of XHTML*), care pot include noi elemente, în funcție de necesitățile dezvoltatorilor (se spune că XHTML este un limbaj *extensibil*).

Pentru redactarea documentelor XHTML este indicată utilizarea editoarelor de text cu suport pentru limbaje de programare/marcare, cu diverse facilități de editare (e.g., mecanismul *syntax highlighting*, care constă în utilizarea unor scheme de colorare a elementelor, atributelor etc.). În continuare sunt enumerate câteva dintre editoarele care pot fi folosite, cu observația că numărul celor disponibile

este destul de mare. Pentru platforma Windows pot fi utilizate *HAPedit* și *PSPad* (gratuite) sau *UltraEdit* și *TextPad* (disponibile în variante *shareware*). În schimb, pe platforma Linux pot fi utilizate *KWrite* (disponibil în mediul KDE) și *gedit* (disponibil în mediul GNOME), *vim* și *Emacs* (ultimul amintit fiind disponibil și pe alte platforme, inclusiv Windows). De asemenea, se pot utiliza editoare de cod XHTML specializate: *Coffee Free HTML*, *HotDog*, *Koala Edit*, *HTML Kit* și *FCKeditor* (pe platforma Windows) sau *Quanta Plus* și *BlueFish* (pe platforma Linux).

Pentru crearea documentelor XHTML mai pot fi utilizate și instrumentele cunoscute sub numele de *unelte de editare (authoring tools)*, care oferă medii de lucru vizuale pentru dezvoltarea site-urilor/aplicațiilor Web. Dintre acestea, amintim *Macromedia Dreamweaver* și *Adobe GoLive* (ambele pentru platforma Windows).

Documentele XHTML sunt valide dacă respectă sintaxa stabilită de recomandările Consorțiului Web. Pentru validare poate fi utilizat instrumentul *Validator*, pus la dispoziție *online* de W3C, sau aplicația gratuită, program disponibil de sine stătător pe mai multe platforme (e.g., Windows, UNIX/Linux, Mac OS, OS/2) sau inclus în alte aplicații. De asemenea, utilitarul *Tidy* poate fi folosit și pentru a converti paginile HTML în documente XHTML. Acesta este o aplicație în linie de comandă pentru care a fost dezvoltată

și o versiune cu interfață grafică, denumită *gui_tidy*.

Un document XHTML include: declarația XML (opțională), declarația tipului de document (opțională), antetul și corpul. Oricare dintre versiunile limbajului XHTML se conformează unei definiții formale stabilite de W3C, reprezentând un tip de document. Această definiție poartă numele de *definiția tipului de document (DTD, Document Type Definition)*, aflându-se la un URI din cadrul site-ului W3C. Pentru a putea fi procesate de analizoarele XML și/sau pentru a stabili exact setul de elemente și de atribute utilizate, documentele XHTML *pot* include la început – la fel ca și în HTML – o declarație a tipului de document utilizat (cunoscută și ca declarație DOCTYPE). Declarația DOCTYPE este utilă browserului Web sau altor dispozitive pentru a identifica versiunea limbajului XHTML folosită în scrierea documentului, altfel spus, setul de elemente care pot fi utilizate. În situația în care sunt folosite elemente neaparținând versiunii XHTML declarate, este posibil ca browserul Web să nu le proceseze în maniera în care se așteaptă proiectantul. În specificația XHTML 1.0 sunt definite trei tipuri de documente: *XHTML 1.0 Strict*, *XHTML 1.0 Transitional* și *XHTML 1.0 Frameset*.

După declarația DOCTYPE, un document XHTML trebuie să conțină (între marcasele `<html>` și `</html>` corespunzătoare instanței elementului-rădăcină) următoarele două secțiuni: antet (*head*),

precizată prin intermediul elementului *head* (este de menționat că elementul *title* trebuie folosit obligatoriu în această secțiune; antetul precizează o serie de informații despre document, iar conținutul elementelor incluse în antet nu este afișat în pagină); corp (*body*), precizată prin intermediul elementului *body* (include conținutul afișat al paginii Web). Elementele *head* și *body* trebuie să apară în document în ordinea enumerării lor, câte o singură dată. După marcajul de închidere al elementului-rădăcină (`</html>`), documentul XHTML nu trebuie să mai conțină alte elemente.

Marcajul de deschidere al elementului *html* trebuie să conțină *obligatoriu* o declarație pentru spațiul de nume XHTML folosit (prin intermediul atributului *xmlns*), ținându-se cont că documentele XHTML reprezintă un tip particular de documente XML. Adresa spațiului de nume XHTML 1.0 este <http://www.w3.org/1999/xhtml>.

Spațiul de nume XHTML poate fi utilizat împreună cu alte spații de nume XML (e.g., SVG, MathML).

Elementul *meta* (folosit, în cadrul secțiunii antet, de câte ori se dorește) furnizează un mecanism generic pentru precizarea unor metadate referitoare la document, specificate întotdeauna ca perechi nume-valoare. Astfel, elementul amintit poate fi utilizat pentru a specifica o descriere a paginii, o serie de cuvinte-cheie caracteristice acesteia, precum și informații care nu sunt precizate prin intermediul altor elemente

incluse în secțiunea antet. Atributele elementului meta sunt `name`, `http-equiv`, `scheme` și `content`, ultimul fiind obligatoriu. Acestea trebuie utilizate în grupuri de câte două, `content` fiind prezent de fiecare dată. Valorile atributelor `name` și `http-equiv` se numesc și *meta-tag-uri*. Valorile *meta-tag-urilor* sunt exprimate prin intermediul atributului `content`. Atributul `name` are ca valoare numele unei proprietăți. În acest caz, atributul `content` exprimă valoarea acesteia. Dintre proprietățile precizate în specificațiile XHTML menționăm `description`, `keywords`, `author`, `copyright`, `owner`, `distribution` și `robots`.

Pentru un document XHTML pot fi precizate următoarele tipuri de conținut (tipuri MIME): `text/html` (browserul va procesa documentul ca și cum acesta ar fi un document HTML), `application/xhtml+xml` (browserul va procesa documentul utilizând regulile sintactice XHTML), `text/xml` sau `application/xml` (ultimele două pot fi precizate pentru orice document din familia XML, inclusiv XHTML).

Documentele XHTML trebuie salvate în fișiere text (ASCII sau Unicode) cu extensia *html* sau *htm*. A se vedea și **DTD**, **hipertext**, **HTML**, **limbaj de marcare** și **marcaj**.

XING. Cea mai mare rețea *online* profesională din Europa, creată și administrată de compania germană XING AG (<http://www.xing.com/>).

XML (*eXtensible Markup Language*). Metalimbaj de marcare, oferă facilități deosebite în ceea ce privește atât stocarea datelor, precum și regăsirea acestora. Astfel, XML permite modelarea unui spectru larg de date, de la cele complet structurate (e.g., conturi bancare) până la text în limbaj natural. Documentele XML pot include diverse conținuturi, pot fi transformate în alte formate (e.g., XHTML) și pot fi interogate folosind limbaje specifice (e.g., XQuery, XPath). De asemenea, XML este destinat utilizării în Internet pentru adnotare, realizarea unor fișiere de configurare, schimb de date între aplicații și prelucrare automată de către clienți (agenți) inteligenți.

XML este un descendent simplificat al SGML (*Standard Generalized Markup Language*), standard al Consorțiului Web (1998, 2000, 2004). Este independent de platformă, extensibil și complet compatibil Unicode.

Limbajul XML reprezintă baza pentru implementarea multor altor limbaje (e.g., XSL, *eXtensible Stylesheet Language*; SVG, *Scalable Vector Graphics*; RSS, *Really Simple Syndication*) sau pentru definirea unor protocoale (e.g., SOAP, *Simple Object Access Protocol*). De asemenea, documentele XML pot fi citite și înțelese de oameni.

Folosind limbajul XML, programatorilor li se oferă posibilitatea să definească elemente/structuri de date independente de platforma hardware/software utilizată (deoarece limbajul nu

are un set predefinit de elemente) și să proceseze documentele XML create, utilizând instrumente disponibile practic în toate limbajele de programare.

Familia XML și limbajele derivate sunt des folosite, datorită posibilităților multiple pe care le oferă pentru reprezentarea (marcarea) datelor semistructurate. Astfel, XML reprezintă formatul preferat pentru marcarea datelor schimbate între aplicații.

Spre deosebire de (X)HTML, limbajul XML îi permite programatorului să separe informația de prezentarea ei (cu alte cuvinte, XML nu include precizări despre modul de afișare a acesteia). Separarea amintită îi permite utilizatorului să se concentreze asupra structurii și să realizeze documente auto-descriptive.

Ideea separării informației de prezentare a apărut din necesitatea utilizării informațiilor nu numai în PC-uri, ci și în alte dispozitive care pot fi conectate la Internet (în special dispozitive mobile, așa cum sunt PDA – *Personal Digital Assistant* – și telefoanele mobile).

Pentru prezentarea datelor XML, trebuie precizată o modalitate de redare utilizând foi de stiluri (*stylesheets*). Acestea sunt scrise utilizând limbajele CSS (care, după cum se știe, folosește o sintaxă non-XML) și XSLT (*Extensible StyleSheet Language Transformation*, care folosește sintaxa XML).

Un document XML poate cuprinde următoarele categorii de componente: declarația XML, elemente, atribute,

entități, secțiuni de marcare și instrucțiuni de procesare.

XML-RPC (*Remote Procedure Call*). Concept des întâlnit în dezvoltarea de software. Se referă la utilizarea de către o aplicație a unor proceduri (scrise, eventual, în alte limbaje de programare) incluse în programe aflate la distanță (*i.e.*, pe un calculator diferit de cel pe care se execută aplicația). Acest mecanism permite unor aplicații executate pe platforme diferite (*e.g.*, Windows și Linux) să comunice între ele. Programul care apelează o procedură aflată la distanță se numește *client RPC*. *Serverul RPC* cere execuția procedurilor și trimite clientului rezultatele generate. Este folosit ca protocol de comunicare, descriind mecanismul de invocare (apelare) a serviciilor Web și de schimb al mesajelor între client și server. Acest standard utilizează protocolul HTTP pentru transportul mesajelor și formatul XML pentru codificarea datelor schimbate.

Un mesaj XML-RPC este o cerere HTTP-POST, respectiv un răspuns HTTP. O conversație dintre două sisteme (numite client XML-RPC și server XML-RPC) începe cu cererea clientului și se termină cu răspunsul serverului. Un mesaj XML-RPC include antetul și corpul. Antetul unui mesaj conține anteturi HTTP. Corpul unui mesaj este scris în format XML. Corpul cererii conține numele unei metode (proceduri) care va fi executată pe server și parametrii necesari acesteia. Corpul răspunsului include

parametrii (sau datele) întorși (întoarse) în urma execuției metodei. A se vedea și **serviciu Web** și **SOAP**.

XMLHttpRequest. Interfață de programare a aplicațiilor (API) expusă de motorul unui limbaj de scripting și utilizată în tehnica AJAX, care permite scripturilor să realizeze funcționalități ale unui client HTTP, ca de exemplu trimiterea datelor incluse în formulare sau încărcarea datelor stocate pe un site Web.

Deși obiectul `XMLHttpRequest` este implementat în toate browserele moderne, aceste implementări nu sunt complet interoperabile. Scopul notei de lucru *The XMLHttpRequest Object*, publicată de consorțiul W3C în data de 5 aprilie 2006, este de a documenta un set minim de caracteristici interoperabile bazat pe specificațiile existente, care să le permită dezvoltatorilor Web să realizeze aplicații Web independente de platforma utilizată.

În ECMAScript, o instanță a obiectului `XMLHttpRequest` poate fi creată utilizând constructorul `XMLHttpRequest()`, astfel:

```
var cerere =  
    new XMLHttpRequest();
```

A se vedea și **AJAX**.

XML Infoset (*XML Information Set*). Recomandare a Consorțiului Web, specifică un model de date abstract pentru XML, în scopul de a asigura interoperabilitatea tehnologiilor, interfețelor de programare și aplicațiilor XML. Astfel, XML Infoset definește –

în manieră obiectuală – o serie de concepte de bază, comune membrilor familiei XML: document (*document information item*), element (*element*), atribut (*attribute*), caractere (*characters*), instrucțiuni de procesare (*processing instruction information item*), comentarii (*comment information item*) și declarația tipului de document (*document type declaration item*). A se vedea și **XML**.

XMLReader. API introdus pentru prima dată în *Microsoft.NET*. Realizează procesarea orientată pe flux a documentelor XML. Versiunea pentru limbajul PHP (denumită *PHP XML Reader*) este disponibilă ca extensie PECL. Această extensie este rapidă și suportă procesarea spațiilor de nume XML și reguli de validare a documentelor. Extensia *PHP XMLReader* furnizează acces *read only* și *forward only* la fluxul de date XML. Se utilizează un cursor (sau *cititor*) care este mutat de la un nod la altul al documentului XML. Se recomandă înlocuirea metodei SAX cu *XMLReader*. A se vedea și **PECL** și **SAX**.

XML Schema. Limbaj pentru crearea schemelor XML, ca și DTD. Spre deosebire de acesta, XML Schema, propus inițial de Microsoft și devenit standard al Consorțiului Web în mai 2001, prezintă o serie de facilități suplimentare, dintre care pot fi menționate: este bazat pe XML, astfel încât schemele XML scrise în acest limbaj sunt procesate într-un mod similar cu documentele XML pe care le descriu;

definește un set complet de tipuri simple de date; asigură suport pentru definirea de către utilizator a unor tipuri complexe de date; include suport complet pentru spațiile de nume. Construcțiile XML Schema au un spațiu de nume asociat al cărui URI este <http://www.w3.org/2001/XMLSchema>, iar atributele privitoare la scheme provin din spațiul de nume identificat prin <http://www.w3.org/2001/XMLSchema-instance>. A se vedea și **DTD** și **schemă XML**.

XOOPS. CMS *open source*, în întregime orientat pe obiect, scris în PHP; este un instrument potrivit pentru a crea site-uri și portaluri Web (<http://www.xoops.org/>). A se vedea și **CMS**.

XSL (*eXtensible Stylesheet Language*). Limbaj care permite transformarea și formatarea documentelor XML. Are ca părți componente XSLT, utilizat pentru a transforma documentele XML în alte documente XML sau derivate (e.g., XHTML), și XSL-FO (*XSL Formatting Objects*), vocabular de formatare a documentelor XML.

XSLT (*XSL Transformations*). Limbaj utilizat pentru transformarea și manipularea documentelor XML. Documentele XSLT sunt stocate în așa-numitele *foi de stil XSLT*. Fișierele în care sunt incluse foile de stil XSLT se numesc *fișiere șablon XSLT*. Scopul în care este utilizată o foaie de stil este acela de a transforma un document sau o clasă de documente XML.

Principalul rol al unui procesor XSLT este de a aplica o foaie de stil XSLT unui document XML sursă și de a produce un document rezultat (în format XML sau în altul, e.g., (X)HTML). Deoarece atât sursa, cât și rezultatul au fiecare ca structură fundamentală arborele, rezultă că un procesor XSLT manipulează arbori XML.

Dintre implementările XSLT pot fi menționate: *Xalan-Java* (este un subproiect al *Apache Xalan Project*; utilizează *Xerces-Java*), *Saxon*, *XT* (creat de James Clark), *Oracle XSLT*; *Xalan-C++* (subproiect al *Apache Xalan Project*), *libxslt* (biblioteca XSLT pentru GNOME, scrisă în C; poate fi utilizată în cadrul programelor scrise în C, Perl, Python sau PHP), *Sablotron* (interfață de programare scrisă în C++, folosită de PHP 4 și de modulul *XML:Sablotron* pentru Perl); *Google AjaXSLT* (proiect *open source* dezvoltat de Google ca o bibliotecă JavaScript, în scopul utilizării în aplicațiile AJAX); *MSXML* (bibliotecă proprietară Microsoft, utilizată în numeroase medii de dezvoltare și limbaje pentru platforma Windows, așa cum sunt .NET, Visual Basic, JScript și C); *Internet Explorer* suportă XSLT 1.0 prin intermediul *MSXML* (acest API poate fi utilizat în paginile Web, via COM – ActiveX); *Mozilla* și *Firefox* au integrat un suport nativ; *Safari 1.3+* are integrat un suport nativ XSLT (cu toate acestea, nu poate realiza transformări XSLT utilizând JavaScript, ceea ce constituie un mare dezavantaj); *Opera* are un suport nativ XSLT începând cu versiunea 9.

Procesorul XSLT modelează un document XML ca un arbore care conține câteva tipuri de noduri: elemente, text, atribute, spații de nume, instrucțiuni de procesare și comentarii. Un document DTD poate adăuga valori implicite ale unor atribute, care devin astfel noduri-atribut suplimentare în arbore. În urma procesării, documentul XML original nu este schimbat. În schimb, este generat un nou document pe baza conținutului său.

În cursul procesării, procesorul XSLT realizează – utilizând informațiile incluse în foaia de stil – diverse tipuri de operațiuni, dintre care menționăm: transformă fiecare element XML în alt element XML; adaugă elemente noi; elimină o parte dintre elementele documentului-sursă; sortează elementele; ia decizii în ceea ce privește elementele pe care trebuie să le afișeze, prin testarea unor condiții; formatează datele afișate. Ținând seama de (unele dintre) operațiile enumerate, se poate spune că XSLT este pentru XML ceea ce este SQL pentru bazele de date. Astfel, SQL realizează interogarea bazelor de date și actualizarea datelor incluse în ele, iar XSLT interoghează documentul XML sau porțiuni ale acestuia, producând un nou conținut.

XSLT este bazat pe reguli, ceea ce îl face un limbaj funcțional, spre deosebire de limbajele de programare (e.g., PHP, C++), care sunt limbaje procedurale. Din acest motiv, XSLT necesită un set de reguli șablon (*template rules*) furnizate de progra-

mator, prin care acesta comunică procesorului XSLT ceea ce trebuie să facă atunci când întâlnește diverse elemente într-un document XML. Mai exact, o regulă șablon – inclusă în foaia de stil XSLT prin intermediul elementului `xsl:template` – definește modul de procesare a nodurilor care se potrivesc cu un anumit model.

După încărcarea documentelor XSLT și XML, procesorul XSLT (e.g., navigatorul Web) începe să citească arborele corespunzător documentului XML căutând nodurile care se potrivesc cu modelele incluse în regulile șablon ale foi de stil XSL asociate. Când este găsit un astfel de nod, conținutul inclus în regula corespunzătoare comunică procesorului XSLT ce ieșire trebuie să genereze și, apoi, căutarea continuă. Când o regulă șablon conține un element `apply-template`, procesorul XSLT va căuta nodurile-copil ale nodului curent și le va aplica regulile șablon asociate. A se vedea și **XML**.

XUL (*XML User Interface Language*). Limbaj de marcare folosit pentru crearea interfeței cu utilizatorul, dezvoltat pentru a suporta aplicațiile *Mozilla: Firefox* (browser) și *Thunderbird* (client de e-mail). A se vedea și **XML**.

X Window System. Numit și X sau X11, publicat pentru prima oară de MIT în 1984, este un set standard de instrumente și un protocol pentru construirea de interfețe grafice pe majoritatea sistemelor de operare Unix-like și Open VMS. Versiunea de protocol X11 a fost publicată în 1987.

Sistemul de ferestre X folosește un protocol de tip client-server: un server X comunică cu diverse programe de tip client. Serverul X acceptă cereri pentru ieșirea grafică (ferestre) și trimite înapoi intrări de la utilizator (de la tastatură, mouse sau *touchscreen*). În mod intenționat, X nu specifică elemente grafice concrete (*e.g.*, buton, meniu), lăsând aceasta în sarcina software-ului utilizatorului (*e.g.*, manager de ferestre, mediu desktop, aplicație GUI). Rezultă că X poate lucra cu un număr mare de interfețe grafice.

Implementarea de referință a sistemului X este X.Org, aceasta fiind utilizată ca implementare canonică.

Conform protocolului X, este necesar ca serverul și clienții să opereze separat, cerință care – împreună cu cea referitoare la independența de platforma hardware – face ca X să nu fie atât de rapid ca sistemele de ferestre integrate în sistemele de operare (*de exemplu*, Windows, Mac OS). A se vedea și **manager de ferestre, sistem de ferestre, UNIX, X.Org și XFree86**.

Y

Yahoo! Corporație publică americană care furnizează servicii Internet în întreaga lume, fondată în 1994 de Jerry Yang și David Filo (<http://www.yahoo.com/>). Conform datelor furnizate de companiile de analiză a traficului Web (e.g., *Netcraft*), în anul 2008 domeniul *yahoo.com* a fost vizitat de peste 1,5 miliarde de utilizatori.

Oferă servicii de căutare (*Yahoo! Search*), e-mail (*Yahoo! Mail*, cu spațiu de stocare nelimitat), director Web (*Yahoo! Directory*), știri (*Yahoo! News*), hărți online (*Yahoo! Maps*), partajare video (*Yahoo! Video*), conversație pe Internet (*Yahoo! Messenger*), călătorie (*Yahoo! Travel*) etc. Toate serviciile oferite se bazează pe un cont unic de utilizator.

Yahoo! Directory. Director Web oferit de Yahoo! (<http://dir.yahoo.com/>). Furnizează atât un serviciu de director, cât și unul de căutare. A se vedea și **director Web și Yahoo!**.

Yahoo! Maps. Serviciu de hărți oferit gratuit de Yahoo! (<http://maps.yahoo.com/>). A se vedea și **Yahoo!**.

Yahoo! Maps API. Grup de API-uri oferit de Yahoo! pentru accesarea ser-

viciului *Yahoo! Maps: Yahoo! Maps Simple API, Yahoo! Maps Flash APIs, Yahoo! Maps Image API și Yahoo! Maps AJAX API*. De exemplu, *Yahoo! Maps Image Map API* permite obținerea unei referințe la o imagine în format PNG (componentă a unei hărți), generată în acord cu parametrii specificați în cerere (latitudine și longitudine sau adresă). *Yahoo! Maps AJAX API* le oferă dezvoltatorilor posibilitatea de a adăuga hărți dinamice în site-uri (<http://developer.yahoo.com/maps/>). A se vedea și **Yahoo!**.

Yahoo! News. Agregator de știri furnizat de Yahoo!. Utilizează servicii de știri ca: *Associated Press (AP), Reuters, France-Press (FP), USA Today, CNN.com, CBS News, BBC News* etc. A se vedea și **Yahoo!**.

Yahoo! Mail. Cel mai important serviciu global de poștă electronică, oferit gratuit de Yahoo! începând cu anul 1997 și având circa 280 de milioane de utilizatori în 2009 (<http://mail.yahoo.com/>). Este oferit în două versiuni: *Yahoo! Mail* și *Yahoo! Mail Classic*. Începând cu iunie 2009, *Yahoo! Mail* a fost integrat cu *Yahoo! Messenger* (cele două servicii pot fi accesate dintr-o singură pagină). Facilitatea nu este valabilă și pentru versiunea *Yahoo! Mail Classic*. A se vedea și **Yahoo!**.

Yahoo! Search. Motor de căutare dezvoltat de Yahoo! (<http://search.yahoo.com/>). Competitorii săi importanți sunt

Google Search și *Bing*. A se vedea și **Bing**, **Google Search**, **GYM** și **Yahoo!**.

Yahoo! Video. Serviciu oferit de Yahoo! în scopul partajării materialelor video, similar cu *YouTube* (<http://video.yahoo.com/>). A fost lansat în iunie 2006 și reproiectat în 2008. A se vedea și **Yahoo!**.

YAML (acronim recursiv pentru *YAML Ain't Markup Language*). Format, superset al JSON, utilizat pentru serializarea datelor, proiectat în scopul de a fi citit ușor de utilizatorii umani și folosit pentru interacțiunea cu limbajele de scripting, e.g., Perl, Python (<http://www.yaml.org/>). Formatul este optimizat pentru serializarea datelor, întreținerea fișierelor jurnal, setarea configurațiilor și crearea mesajelor în Internet. Dintre caracteristicile acestui format pot fi menționate: documentele YAML sunt foarte ușor de citit de către utilizatorii umani; interacționează foarte bine cu limbajele de scripting; utilizează procesarea bazată pe flux; este extensibil și ușor de implementat.

Una dintre cele mai importante implementări ale YAML este biblioteca *Syck*, utilizată pentru serializarea/deserializarea datelor, cu versiuni scrise în limbajele Python, Ruby și PHP. A se vedea și **JSON**.

Yandex. Lansat în 1997, este un motor de căutare și cel mai mare portal Web în limba rusă (<http://www.yandex.ru/>). A se vedea și **motor de căutare** și **portal Web**.

YouTube. Aplicație/serviciu de partajare video (*video sharing*), în care utilizatorii pot încărca, vizualiza și partaja clipuri video (<http://www.youtube.com/>).

YouTube a fost creat în februarie 2005 de trei foști angajați ai PayPal (Chad Hurley, Steve Chen și Jawed Karim). Pe 13 noiembrie 2006, compania Google a achiziționat *YouTube*. În ianuarie 2008, 79 milioane de utilizatori au efectuat peste 3 miliarde de vizionări ale clipurilor video plasate pe site-ul *YouTube*. În martie 2008, costul lățimii de bandă folosite de utilizatorii *YouTube* a fost estimat la un milion de dolari pe zi.

Serviciul *YouTube* permite încărcarea unei varietăți de conținut generat de utilizator, incluzând secvențe din filme, clipuri TV și videoclipuri muzicale. Redarea clipurilor (*playback*) se realizează prin intermediul tehnologiei *Flash*, care oferă o calitate comparabilă cu cea furnizată de alte tehnologii (e.g., *Windows Media Player*, *QuickTime*, *RealPlayer*). Toate tehnologiile amintite necesită descărcarea și instalarea unui *plug-in*, dar firma *Adobe* (proprietara formatului) consideră că *plug-in*-ul *Flash 7* este prezent în peste 90% dintre computerele conectate la Internet. De asemenea, clipurile încărcate pe *YouTube* pot fi redade și cu alte playere (e.g., *GOM Player*).

Conform regulilor impuse de *YouTube*, durata clipurilor încărcate de utilizator este limitată la zece minute, iar dimensiunea acestora la 1024 MB. După încărcare, *YouTube* convertește fișierul la formatul video *Flash*. *YouTube*

acceptă pentru încărcare numeroase formate video, incluzând WMV, AVI, MOV, MPEG, MP4, DivX, FLV și OGC. De asemenea, suportă 3PG, permițând încărcarea clipurilor direct de pe telefonul mobil.

Utilizatorii neînregistrați pot vizualiza cele mai multe clipuri video postate pe site, în vreme ce utilizatorilor înregistrați li se permite să încarce un număr nelimitat de clipuri. Este interzisă încărcarea unor clipuri video cu conținut nepotrivit, a reclamelor și a materialelor care încurajează criminalitatea. *YouTube* a fost adesea criticat pentru eșecul său în a asigura că postarea clipurilor este în concordanță cu legea *copyright*-ului. În timpul încărcării unui clip, utilizatorul este înștiințat că operațiunea nu trebuie să încalce drepturile de autor.

Clipurile video încărcate pe *YouTube* sunt apreciate cu diverse calificative sau onoruri: *Most Viewed*, *Top Rated*, *Most Discussed*, *Top Favorites*, *Rising Videos*, *Recently Featured* și *Most Responded*. Cele mai vizualizate clipuri (*Most Viewed*) sunt împărțite în patru categorii: astăzi (*today*), în săptămâna curentă (*this week*), în luna curentă (*this month*) și dintotdeauna (*all time*). Clipurile video legate de cel curent (vizualizat) prin titluri și marcaje sunt afișate într-o listă, în partea dreapă a acestuia.

Crearea unui cont pe *YouTube* este gratuită și presupune furnizarea următoarelor informații: adresa de *e-mail*, parolă, nume de utilizator (acesta poate

conține numai litere și cifre), țara în care locuiește utilizatorul, data nașterii și sexul. Dacă utilizatorul are un cont *Gmail*, este posibil să se logheze în *YouTube* folosind numele de utilizator și parola corespunzătoare. Totuși, este necesar în continuare să aleagă un nume de utilizator.

YouTube le oferă dezvoltatorilor un API propriu. Pentru a-l folosi este necesară obținerea gratuită a unui identificator de dezvoltator de aplicații (*developer ID*). Identificatorul obținut va fi utilizat la fiecare apel al API-ului respectiv. Apelurile pot fi efectuate în două modalități care corespund unor direcții importante de dezvoltare a aplicațiilor Web: REST (*Representational State Transfer*) și XML-RPC.

Una dintre aplicațiile importante construite pe baza API-urilor *Google Maps* și *YouTube* este *Virtual Video Map* (<http://www.virtualvideomap.com/>), care permite vizualizarea filmelor asociate unor anumite locații. De asemenea, poate fi remarcată aplicația *CoverPops* (<http://www.coverpop.com/>), care alcătuiește săptămânal un mozaic de imagini legate către cele mai vizionate clipuri video de pe *YouTube*.

Ca și alte instrumente Web 2.0, *YouTube* poate fi utilizat în domeniul educațional. Astfel, profesorii pot posta în format video: cursuri, lecții și experimente (de chimie, fizică etc.). Acestea pot fi utile atât altor profesori, cât și studenților. A se vedea și **TeacherTube** și **video sharing**.

Z

Zadeh, Lotfali Askar (n. 1921). Matematician și informatician american de origine azeră, cunoscut mai ales pentru inventarea logicii *fuzzy*, utilizată în unele sisteme de inteligență artificială (sisteme *fuzzy*). A se vedea și **logica fuzzy și inteligență artificială**.

Zamzar. Converter *online* excelent, realizează transformarea documentelor în diverse formate, imagini și clipuri video (<http://www.zamzar.com/>). Conversia se realizează în patru pași: selectarea documentului, alegerea formatului, precizarea adresei de *e-mail* la care vor fi trimise fișierele convertite și operația de conversie în sine.

Serviciul *Zamzar* poate fi utilizat și pentru stocarea și gestionarea fișierelor convertite (*Manage Files*), fiind necesară înregistrarea utilizatorului. Avantajele deținerii unui cont sunt următoarele: obținerea unui *inbox* pentru gestionarea fișierelor, conversia fișierelor mari (de până la 1 GB), obținerea unui spațiu de stocare de 100 GB și conversia prioritara a fișierelor (înaintea celor care aparțin utilizatorilor neînregistrați).

Zend Framework. Cadru de lucru pentru PHP 5, dezvoltat de compania

Zend Software, din Israel. A fost proiectat în scopul de a elimina detaliile plictisitoare legate de scrierea efectivă a codului, lăsându-l pe programator să-și concentreze atenția asupra întregului proiect pe care îl dezvoltă (<http://framework.zend.com/>).

Zend_Json. Bibliotecă inclusă în *Zend Framework*. Furnizează metode pentru serializarea valorilor native PHP în obiecte JSON, precum și pentru decodificarea obiectelor JSON în valori PHP native. A se vedea și **JSON și PHP**.

Zend Studio. IDE comercial produs de Zend Software. Pe lângă o mulțime de alte funcționalități (analizor de cod, suport CVS, inspectoare, suport FTP), oferă și un *debugger*. Depanarea codului se poate face inclusiv de la distanță, în acest caz folosindu-se componenta *Zend Studio Server*, inclusă în pachet (<http://www.zend.com>). A se vedea și **IDE**.

Ziua accesului deschis (OAD, *Open Access Day*). 14 octombrie a fost declarată astfel de SPARC (*Scholarly Publishing and Academic Resources Coalition*), PLoS (*Public Library of Science*) și *Students for FreeCulture*. Această zi a fost marcată pentru prima dată în anul 2008, inclusiv în România. A se vedea și **acces deschis**.

Ziua protecției datelor (*Data Protection Day*). Sărbătorirea acesteia în fiecare an pe data de 28 februarie, începând cu 2007, a fost stabilită de Comisia Europeană în scopul de a le

oferi cetățenilor Uniunii posibilitatea de a înțelege care date personale sunt colectate și procesate (și de ce), precum și drepturile lor privitoare la aceste operații. Un alt obiectiv este acela de a le oferi persoanelor care lucrează în domeniul colectării datelor oportunitatea de a discuta teme legate de securitatea datelor.

Ziua libertății documentelor (DFD, *Document Freedom Day*). Marcată pe 26 martie, pentru prima dată în 2008, în această zi se desfășoară acțiuni pentru promovarea standardelor și formatelor deschise. DFD a fost inițiată și este sprijinită de un număr mare de organizații și companii (e.g., *Free Software Foundation Europe*, *ODF Alliance*, *OpenForum Europe*, IBM, RedHat, Sun Microsystems, Inc.). În anul 2008, aproape 200 de echipe din peste 60 de țări au organizat activități locale pentru a marca DFD.

A se vedea și **format deschis și standard deschis**.

Zoho. Pachet *Office 2.0* gratuit oferit de *Zoho* (<http://www.zoho.com/>), care include ca aplicații principale *Zoho Writer* (pentru documente text), *Zoho Sheet* (pentru foi de calcul) și *Zoho Show* (pentru prezentări). Pentru folosirea aplicațiilor oferite de *Zoho* este necesară crearea unui cont de utilizator. În absența acestuia, se pot folosi conturile deținute pe Google sau Yahoo!. A se vedea și **Office 2.0**.

zoom (*a face zoom*). A mări o imagine sau un document pentru a avea acces la detalii. Posibilitatea de a face *zoom* este caracteristică unui număr mare de tipuri de aplicații, așa cum sunt programele de editare grafică.

Zune Marketplace. Software de tip *podcatching client*, creat de Microsoft. A se vedea și **podcatching client**.

