

COMPONENTELE UNUI SERVER WEB

LECTOR UNIV. LUMINIȚA ȘERBĂNESCU
Universitatea din Pitești
Pitești, str. Negru-Vodă, bl.D3, sc. A, ap.1, Argeș
Tel: 0745030906
Email: luminitaserb@yahoo.com

Usually, the "client" is actually a software program, like Netscape Navigator, that is being operated by a person who is the one who really wants to see the file. The client software however, deals with all the underlying client/server protocol stuff and then displays the document (that usually means interpreting HTML, but we'll get there in just a bit) to the human user.

Sistemul web se bazează pe o serie de componente:

- Serverul web;
- Clientul web (navigatorul);
- Protocolul de comunicație (HTTP);
- Limbajul de descriere a informațiilor (HTML)

1. Serverul web Serverul web este în esență o aplicație care pe baza unor cereri formulate de un client în baza protocolului HTTP îi oferă acestuia un document și/sau un mesaj. El reprezintă un program ce rulează în regim de proces (daemon) pe o mașină cu un sistem de operare multitasking (UNIX, Windows NT). Numărul mare de utilizatori web face ca un astfel de server să fie foarte utilizat, fiind nevoie de resurse importante pentru buna lui funcționare.

2. Clientul web (Browser) Un client web, denumit generic browser (navigator) este o aplicație ce rulează pe un sistem ce oferă suport pentru protocolul HTTP în principal (și altele în particular) și este capabil să afișeze informații codificate în limbajul HTML și text, oferind suport și pentru alte tipuri de informație (imagine, animații). Totodată clientul dă posibilitatea prin intermediul interfeței cu utilizatorul (GUI) să interacționeze cu conținutul paginii de tip hipermedia, preluând acțiunile utilizatorului.

Programele client sunt diverse și există pe orice sistem de operare, motiv pentru care serviciul WWW este accesibil practic oricărui utilizator. Datorită limbajului HTML, conținutul paginilor este redat aproximativ sub aceeași formă tuturor utilizatorilor, diferențele fiind minore, depinzând mai mult de capacitatea tehnică a sistemului și de versiunea clientului.

Dacă la început programele client nu dispuneau de capacități grafice, informațiile WWW fiind în general sub formă de text, ulterior, datorită progresului în domeniul sistemelor de operare și al interfețelor grafice, capacitățile browser-ului au crescut, ajungându-se la înglobarea tuturor serviciilor Internet în cadrul unui client web. Practic pentru utilizatorul obișnuit browser-ul este practic Internet-ul.

În esență un client web trebuie să ofere acces la următoarele funcții:

- navigare – posibilitatea de a interacționa cu conținutul prin ceea ce se numește legătură (link), esența mediului hipermedia;
- comunicare – prin suportarea protocoalelor Internet (în special HTTP);
- afișare și redare – în conformitate cu specificațiile HTML trebuie să ofere suport pentru imagini (formatele GIF, JPEG și PNG);
- suport pentru tehnologii și standarde noi – Java, ActiveX, Flash, XML etc.
- acces la resursele sistemului de operare – salvare de fișiere, imprimare etc.

3. *Protocolul HTTP* Protocolul web (HTTP) a fost formulat în 1990 și standardizat în 1996 într-o primă versiune (HTTP 1.0). Ulterior acesta a fost completat cu alte standarde pentru a face față noilor cerințe impuse de creșterea Internetului.

Protocolul HTTP este un protocol de nivel aplicație pentru sisteme distribuite, colaborative și hipermedia, folosit în sistemul global WWW începând cu 1990.

Majoritatea comunicațiilor HTTP sunt inițiate de un agent și constă într-o cerere aplicată unei resurse pe un anumit server. Cazul cel mai simplu îl reprezintă o singură conexiune între agent și server. Cel mai important și vizibil element al protocolului HTTP îl reprezintă URL-urile (Uniform Resource Locators). URL-ul are în forma lui generală utilizată în HTTP sintaxa următoare:

<protocol>://<user>:<password>@<host>:<port>/<url-path>

Acestea semnifică:

- protocol – protocolul folosit, în mod obișnuit http. Uzual se folosesc ftp, mailto, gopher, news, nntp, telnet și file;
- user și password – pentru o eventuală autentificare. Lipsa lor presupune o conectare anonimă;
- host – numele serverului conform protocolului numelui de domeniu (DNS)
- port – portul TCP folosit. Lipsa lui implică folosirea porturilor implicite pentru protocoalele respective;
- url-path – pentru serviciile ce oferă acces la o structură de fișiere organizată arborescent (cum este și WWW).

4. *Limbaajul de descriere a informațiilor*. Tehnologiile utilizate în proiectarea și design-ul site-urilor web:

- **HTML** (Hyper Text Markup Language). Un fișier HTML este un fișier text cu extensia ".htm" sau ".html" conținând pe lângă text pur și instrucțiuni standard de machetare (web design). Efectul machetării este vizibil în browserele de Internet (Internet Explorer, Netscape Navigator, Opera, etc...). Practic marcasele de tip HTML din cadrul documentului spun browser-ului cum să afișeze conținutul util al fișierului. HTML, limbaajul de marcare a informației, este limbaajul folosit pentru publicarea informației în format hipertext pe Internet. HTML care descrie formatul primar în care documentele sunt distribuite și văzute pe web. Multe din trăsăturile lui, cum ar fi independența față de platforma, structurarea

formatării și legăturile hipertext, fac din el un foarte bun format pentru documentele Internet și web. Documentele HTML sunt documente în format ASCII și prin urmare pot fi create cu orice editor de texte.

- **CSS** (Cascading Style Sheets). Un fișier CSS este un fișier text cu extensia ".css" definind stiluri pentru paginile HTML. Stilurile indică într-un mod mult mai riguros decât atributele HTML cum să afișeze browserul elementele componente ale unui document HTML (text, grafică, formulare, etc...), oferind posibilități mult superioare standardului HTML. Fișierele CSS permit stilizarea în bloc a documentelor HTML cu un efort semnificativ mai mic decât în cazul stilizării elementelor de pagină în cadrul fiecărui document HTML în parte, prin intermediul atributelor tagurilor.

- **XML** (EXtensible Markup Language). A fost proiectat pentru a manipula date în mod riguros. Ineditul pe care îl aduce limbajul XML este faptul ca nu conține taguri standard (predefinite). Practic, tagurile XML sunt definite (construite) exclusiv de utilizator. Este așadar un limbaj extensibil. Trebuie menționat faptul că standardul XML nu este decât într-un mod foarte general asemănător cu HTML. În timp ce standardul HTML urmărește afișarea datelor având ca obiectiv central modul în care acestea arată în browser, scopul XML este acela de a structura, stoca și transmite date sub formă de informații utilizator, având ca obiectiv central semnificația profundă a acestora. Într-un cuvânt, HTML afișează informații, în timp ce XML descrie și manipulează informațiile în mod riguros.

Atenție: XML nu este un înlocuitor pentru HTML. O evoluție previzibilă a web-ului sugerează o completare reciprocă a celor două standarde prin utilizarea împreună în cadrul acelorași documente: XML va fi utilizat pentru a descrie date, în timp ce HTML va forma și afișa corespunzător aceste date.

- **XSL** (EXtensible Style Language). Practic, XSL reprezintă pentru documentul XML ceea ce reprezintă CSS pentru HTML, adică o foaie de stiluri. Diferența dintre CSS (HTML Style Sheets) și XSL (XML Style Sheets) este foarte mare. HTML utilizează taguri (marcaje) predefinite și semnificația acestora este de la bun început cunoscută și înțeleasă de browser. Spre exemplu, tagul <TABLE> definește un tabel, iar browserul știe ce este acesta și cum trebuie afișat. Adăugând stiluri fișierului HTML vom indica browserului să afișeze un element de pagină într-un anumit mod, uzând de data aceasta de atribute CSS predefinite - indicând, spre exemplu, margine de dimensiune 1 și culoare albastră. În schimb XML știm deja ca nu lucrează cu taguri predefinite (am văzut ca putem defini câte taguri personalizate dorim). Așadar, semnificația tagurilor XML utilizate de proiectant nu este cunoscută browser-ului. Tagul <TABLE> în XML poate desemna, într-adevăr, un tabel, la fel de bine ca și un formular sau un obiect abstract iar browserul nu va ști cum sa-l afișeze. De aceea trebuie să existe o componentă adițională documentelor XML care să traducă browserului modul în care un document XML va fi afișat. și aceasta este XSL. Totuși, XSL este mai mult decât un standard de stiluri pentru documentele XML. Practic, XSL se constituie din 3 părți componente: XSLT, care este un limbaj menit să transforme documentele XML în documente XHTML; XPath care este un limbaj menit să definească și să izoleze porțiuni din documentele XML; XSL-FO, practic un limbaj de formatare (structurare) superioară a documentelor XML

- **XHTML** (EXtensible HyperText Markup Language). Practic este un înlocuitor modern al mai vechiului limbaj HTML, fiind aproape identic cu standardul HTML 4.01, o versiune

îmbunătățită, mult mai strictă și mai curată a standardului HTML 4.0. Abstractizând puțin, afirmăm ca XHTML reprezintă o definiție de tip XML a unui document HTML, pe scurt o combinație între HTML și XML. Avantajele unei aplicații XHTML sunt multiple: pot fi citite de toate dispozitivele XML, în timp ce păstrează compatibilitatea cu toate browserele de Internet mai vechi sau mai noi fără a necesita specificații suplimentare.

- **JavaScript.** JavaScript reprezintă un limbaj de script utilizat la ora actuală de majoritatea paginilor web din Internet pentru a îmbunătăți design-ul general, valida datele utilizator din formulare, adăuga interacțiune cu utilizatorul prin efecte speciale aplicabile elementelor de pagină în funcție de evenimentele generate de utilizator. La ora actuală este cel mai important limbaj de script de pe Internet.

- **Java.** Java este la ora actuală una dintre cele mai puternice platforme de dezvoltare software - limbaj de programare independent, aducând în spectrul web posibilități excepționale de interacțiune client-server, cu mult dincolo de binecunoscutele formulare, animații, jocuri sau efecte vizuale interactive. Practic tehnologiile Java (de multe ori combinate cu alte standarde sau tehnologii cum ar fi XML spre exemplu), reprezintă o modalitate foarte apropiată de natural pentru implementarea de aplicații manipulând cu succes mai toate tipurile de generatori și consumatori de informație, de la telefoane mobile până la sisteme de gestionare a tranzacțiilor online. Portabilitatea și extensibilitatea celor două tehnologii (Java și XML) face din acestea alegerea ideală pentru satisfacerea unei impresionante game de cerințe ale Internetului, de la cele mai simple programe până la aplicații rulând pe diferite platforme interconectate și solicitând nivele dintre cele mai performante de securitate, stabilitate și portabilitate.

- **PHP.** Ca să adaugi sau să modifice o pagină HTML a unui site trebuie să o descarci de pe server, să aplici modificările, să găsești și celelalte pagini care ar urma să aibă de suferit de pe urma schimbării link-urilor și să aplici și la acestea modificările de rigoare, iar apoi să transferi totul înapoi pe server. Dacă site-ul de 500 de pagini actualizabil în acest mod în fiecare zi a săptămânii mai trebuie să și poată vinde produsele online, să ofere facilități avansate de căutare în paginile de produse și în plus să poată înregistra și o evidență exactă vânzărilor, stocurilor și clienților, atunci perspectiva unei astfel de munci nu poate decât să sperie până și pe cel mai tenace creator de site-uri web. Utilizând însă o bază de date legată la o pagina web prin intermediul unui limbaj "server-side" (pe parte de server), web designer-ul nu mai are nevoie de ajustări în codul paginilor pentru a schimba periodic conținutul acestora sau pentru a oferi facilitățile amintite mai devreme. Crearea de site-uri web dinamice și interactive folosind tehnologia PHP combinată cu bazele de date operate în MySQL, totul utilizând un server Apache, este unul dintre cele mai bune lucruri pe care le puteți învăța.

- **JSP.** Este o tehnologie alternativă la PHP, utilizată în același scop și anume implementarea site-urilor web dinamice și interactive. Când a devenit clar ca folosirea Java pe parte de server era o idee foarte bună, s-a propus dezvoltarea unei interfețe API Java Servlet care a fost adoptată de mulți programatori de Java care lucrau la extinderea funcționalității serverelor web. Interfața API Java Servlet trata în mod elegant multe dintre problemele cărora trebuiau să le facă față programatorii Java pe partea de server și oferea un standard pe baza căruia furnizorii de aplicații puteau construi motoare de servlet-uri. Toate dispuneau de facilități de rulare și zone de izolare pentru securitate, ceea ce a simplificat mult programarea servlet-urilor pentru dezvoltatorii de aplicații Java. Servlet-

urile aveau însă și câteva dezavantaje: trebuia să scrii instrucțiuni de ieșire speciale pentru fiecare bucată de cod HTML care voiai să fie afișată clientului; trebuia să ai cunoștințe de Java; în plus, existau și alte tehnologii concurente pe piață, ca de exemplu ASP (de la Active Server Pages, produs marca MicroSoft), care le permiteau programatorilor să creeze mai ușor conținut dinamic pentru web. Era de dorit ca puterea servlet-urilor Java să fie pusă la dispoziția celor care se ocupau de dezvoltare web, fără a-i obliga să învețe Java. Așa s-a născut specificația Java Server Pages (JSP) lansată de firma Sun la începutul anului 1998. Combinând puterea și extensibilitatea limbajului Java cu simplitatea și ușurința de folosire a scriptării pe baza de etichete, Java Server Pages a devenit rapid limbajul preferat de mulți programatori profesioniști pentru dezvoltarea aplicațiilor bazate pe web.

Un motor de căutare este un server cu conexiune permanentă la rețeaua Internet, care are instalată local o bază de date cu informații despre diverse site-uri de web existente în Internet. Motoarele de căutare sunt principalul instrument prin care utilizatorii Internet ajung la informațiile dorite.

Pentru ca un motor de căutare (search engine) să "știe" de existența unui site web, trebuie ca acesta să fie "anunțat". Operațiunea prin care un motor de căutare este "anunțat" de existența unei pagini web sau a unui site se numește înregistrare. Fiecare search engine are o metodă proprie prin care acceptă o înregistrare nouă. Rezultatele unei căutări depind de felul în care cuvintele conținute în cerere (întrebare) sunt regăsite în: titlul paginii, descrierea paginii, textul conținut în pagină, cuvintele cheie declarate în pagină, informațiile declarate în momentul înregistrării (prin completarea formularelor de înregistrare).

Principiul de indexare al fiecărui search engine diferă de la un motor la altul. Se pot defini câteva tipuri de motoare de căutare:

1. Motoare automate cu "spider"

ex: <http://www.altavista.com> , <http://www.lycos.com>

Aceste motoare au cea mai simplă procedură de înregistrare. Este suficient să anunțați o nouă adresă din site și motorul va indexa automat toate paginile din site cu legături directe din acea pagină. Programul prin care este realizată indexarea automată se numește "spider". Acest spider va face un "rezumat" al site-ului pe baza informațiilor pe care le va găsi în titlu, descriere, conținut, key-words, etc. și le va stoca în baza de date. Avantajul acestor motoare automate este acela al unei înregistrări rapide. Același avantaj poate constitui o problemă prin faptul ca numărul mare de înregistrări poate face căutările dificile.

2. Motoare bazate pe formulare de înregistrare

ex: <http://www.yahoo.com>

Aceste motoare au principalul avantaj de a fi foarte selective. În cazul serverului Yahoo, administratorii acestuia își rezervă dreptul de a refuza anumite înregistrări. Selecția înregistrărilor este făcută de către operatori în funcție de calitatea paginilor, conținutul acestora și de modul în care sunt respectate regulile impuse de echipa care gestionează motorul de căutare. Aceste reguli sunt menționate explicit în momentul înregistrării.

Principalul dezavantaj al acestor motoare de căutare este intervalul de timp relativ mare în care sunt validate înregistrările (în cazul fericit în care acestea sunt acceptate).

Pentru ca un site web să fie citit de un număr cât mai mare de utilizatori are nevoie de o înregistrare de calitate.

Bibliografie:

1. Sabin Buraga, Proiectarea site-urilor web. Design și funcționalitate, Editura Polirom, 2005
2. Postel, J., "Transmission Control Protocol- DARPA Internet Program Protocol Specification", RFC-793, USC/Information Sciences Institute, September, 1981.
3. St. Johns, M., "The Identification Protocol", RFC 1413, February 1993.
4. Kris Jamsa, Programarea în Web, Editura. All Educational București, 1997
5. Dan Somnea, Inițiere în JavaScript și tehnologiile Netscape, Editura Tehnică, 1998