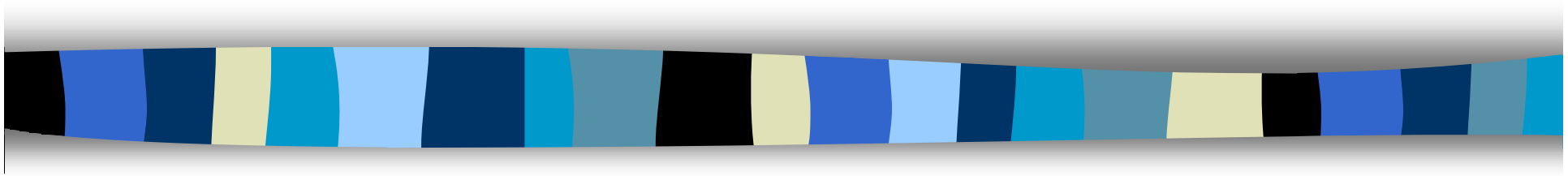
Le protocole HTTP



Didier DONSEZ

Université de Valenciennes
Institut des Sciences et Techniques de Valenciennes

donsez@univ-valenciennes.fr

Au sommaire

- Historique
- Le protocole HTTP
- Les méthodes GET et POST
- Les Cookies
- Les Serveurs du Marché
- Apache
- JavaServer et les Servlets
- Autour d ’HTTP

Le Protocole HTTP

- HTTP : HyperText Tranfert Protocol *(RFC 1945 et 2068)*
  - protocole de rapatriement des documents
  - protocole de soumission de formulaires
- Fonctionnement *(très simple en HTTP/1.0)*
    - connexion
    - demande (GET) d ’un document
    - renvoi du document (status=200) ou d ’une erreur
    - déconnexion
- *Cependant*
  - *dialogue plus complexe en cas d ’identification*
  - *optimisation :*
    une série de plusieurs requêtes sur une connexion
    - Connexion « KeepAlive » de HTTP/1.1 *(RFC 2068)*

Dialogue HTTP

- Dialogue
  - en mode ASCII 7
    - telnet www.sun.com 80
- Types de dialogue
  - Récupération d ’un document
    - méthode GET
  - Soumission d ’un formulaire
    - méthodes GET ou POST
  - Envoi de Document et Gestion de Site
    - méthodes PUT, DELETE, LINK, UNLINK
  - *Gestion de proxy/cache*
    - *méthode HEAD (récupération des informations sur le document)*

Format de la requête (i)

<Méthode> <URI> HTTP/<Version>
[<Champ d 'entête>: <Valeur>]
[<tab><Suite Valeur si >1024>]

- Méthodes
  - GET
    - demande pour obtenir des informations et une zone de données concernant l 'URI
  - HEAD
    - demande pour seulement obtenir des informations concernant l 'URI
  - POST
    - envoie de données (contenu du formulaire vers le serveur, …). Ces données sont situées après l 'entête et un saut de ligne.

Format de la requête (ii)

- Autres méthodes
  - PUT
    - enregistrement du corps de la requête à l ’URI indiqué
  - DELETE
    - suppression des données désignées par l ’URI
  - LINK / UNLINK
    - association (et désassociation) des informations de l ’entête au document sur le serveur
  - OPTIONS
    - demande des options de communication disponibles
  - TRACE
    - retourne le corps de la requête intacte (déboguage)

Format de la réponse

HTTP/<Version> <Status> <Commentaire Status>
Content-Type: <Type MIME du contenu>
[< Champ d ’entête >: <Valeur>]
[<tab><Suite Valeur si >1024>]
<Ligne blanche>
Début du Document

Status des réponses HTTP *(RFC2068)*

- réponse donné par le serveur au client

Status de la requête

- 100-199 Informationnel
  - 100 : Continue (le client peut envoyer la suite de la requête), ...
- 200-299 Succès de la requête client
  - 200: OK, 201: Created, 204 : No Content, ...
- 300-399 Redirection de la Requête client
  - 301: Redirection, 302: Found, 304: Not Modified, 305 : Use Proxy, ...
- 400-499 Requête client incomplète
  - 400: Bad Request, 401: Unauthorized, 403: Forbidden, 404: Not Found
- 500-599 Erreur Serveur
  - 500: Server Error, 501: Not Implemented,
  - 502: Bad Gateway, 503: Out Of Resources (Service Unavailable)

Entêtes HTTP

- 4 types de champs d ’entête
  - Général
    - commun au serveur, au client ou à HTTP
  - Requête du client
    - formats de documents et paramètres pour le serveur
  - Réponse du serveur
    - information concernant le serveur
  - Entité
    - informations concernant les données échangés

Entêtes Généraux HTTP

- Cache-Control = contrôle du caching.
- Connection = listes d 'option
  - close pour terminer une connexion.
- Date = date actuelle (format RFC1123 mais aussi RFC850).
- MIME-Version = version MIME utilisé.
- Pragma = instruction pour le proxy.
- Transfer-Encoding = type de la transformation appliquée au corps du message.
- Upgrade = indique le protocole soutaité.
- Via = utilisé par les proxys pour indiquer les machines et protocoles intermédiaires.

Entêtes de requêtes client HTTP

- Accept = type MIME visualisable par l ’agent
- Accept-Encoding = méthodes de codage acceptées
  - compress, x-gzip, x-zip
- Accept-Charset = jeu de caractères préféré du client
- Accept-Language : liste de langues
  - fr, en, …
- Authorization = type d ’autorisation
  - BASIC nom:mot de passe (en base64)
  - donc transmis en clair !!!
  - NB : Préalablement le serveur a répondu un WWW-Authenticate
- Cookie = cookie rétourné

Entêtes de requêtes client HTTP

- From = adresse email de l ’utilisateur
  - rarement envoyé pour conserver l anonymat de l ’utilisateur
- Host = spécifie la machine et le port du serveur
  - un serveur peut héberger plusieurs serveurs
- If-Modified-Since = condition de retrait
  - la page n ’est transférée que si elle a été modifiée depuis la date précisée. Utilisé par les cachs
  - indique si le document demandé peut être caché ou pas.
- If-Unmodified-Since = condition de retrait
- If-Match = condition de retrait
- If-None-Match = condition de retrait
- If-Range = condition de retrait

Entêtes de requêtes client HTTP

- Max-Forwards = nombre max de proxy
- Proxy-Authorization = identification
- Range = zone du document à renvoyer
  - bytes=x-y (x=0 correspond au premier octet, y peut être omis pour spécifier jusqu ’à la fin)
- Referer = URL d ’origine
  - page à contenant l ’ancre à partir de laquelle le visualisateur a trouvé l ’URL.
- User-Agent = modèle du visualisateur

Entêtes des réponses serveur HTTP

- Accept-Range = accepte ou refus d ’une requête par intervalle
- Age = ancienneté du document en secondes
- Proxy-Authenticate = système d ’authentification du proxy
- Public = liste de méthodes non standards gérées par le serveur
- Retry-After = date ou nombre de secondes pour un ressai en cas de code 503 (service unavailable)
- Server = modèle de HTTPD
  - utilisé par Satan !!!!
- Set-Cookie = créer ou modifie un cookie sur le client
- Vary = l ’entité possède plusieurs sources
- Warning = informations supplémentaire du code d ’état
  - 14 Transformation applied : le proxy a changé de Content-Type ou le Content-Encoding
- WWW-Authenticate = système d ’authentification pour l ’URI

Entêtes d ’entité HTTP

- Allow = méthodes autorisées pour l ’URI
- Content-Base = URI de base
  - pour la résolution des URL
- Last-Modified = date de dernière modification du doc.
  - Utilisé par les caches
- Content-Length = taille du document en octet
  - utilisé par le client pour gauger la progression des chargements
- Content-Encoding = type encodage du document renvoyé
  - compress, x-gzip, x-zip
- Content-Language : le langage du document retourné
  - fr, en ...
- Content-Location : URI de l ’entité
  - quand l ’URI est à plusieurs endroits

Entêtes d ’entité HTTP

- Content-MD5 : résumé MD5 de l ’entité
- Content-Range : position du corps partiel dans l ’entité
  - bytes x-y/taille
- Content-Transfert-Encoding : transformation appliqué du corps de l ’entité
  - 7bit, binary, base64, quoted-printable
- Content-Type = type MIME du document renvoyé
  - utilisé par le client pour sélectionner le visualisateur (plugin)
  - RFC2045
- ETag : transformation appliqué du corps de l ’entité
  - 7bit, binary, base64, quoted-printable

Entêtes d ’entité HTTP

- Expires : date de péremption de l ’entité
- Last-Modified : date de la dernière modification de l ’entité
- Location : URI de l ’entité
  - quand l ’URI est à plusieurs endroits
- URI : nouvelle position de l ’entité
- Refresh : 10
- Refresh: 10; Location=/newloc.htm

Internationalisation

- Language Accept
  - fr, de, it, en, sq (albanais), ru, (russe), ja (japonais), zh (chinois), el (grec), he (hébreu), ca (catalan) ...
- Charset (table de caractère)
  - par défaut ISO-8859-1 (Latin-1)
    - ISO-8859-2 (hongrois, albanais, …)
    - ISO-8859- 4
    - ISO-8859-5, KOI8-R (russe, bulgare, polonais)
    - ISO-8859-7 (grec)
    - ISO-8859-8 (hébreu)
    - ISO-8859-9 (turc)
    - Shift_JIS, ISO-2022-JP, EUC-JP (japonais)
    - Big5 (chinois simplifié)
    - GB2312(chinois traditionnel - Taiwan)

Codage des « paramètres »

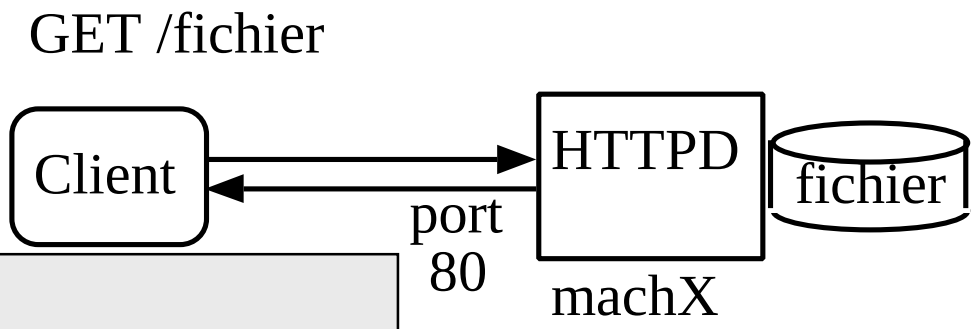
- Les valeurs passées (URL et contenu des entrées des formulaires) doivent être sur 7 bits et sans caractères spéciaux

- Format d ’encodage : **x-www-form-urlencoded**
  - Espace -> « + »
  - Tous les caractères spéciaux et accentués -> %code ascii
    - **@ %40**
    - **é %e9**
  - Les entrées des formulaires sont encodés dans une chaîne composée de paires (nom de l ’entrée)=(valeur de l ’entrée) séparé par de &
    - **nom=Dupont+Jean&adresse=3+rue+de+la+Gait%e9%0a75014+Paris**
  - il existe de nombreuses bibliothèques d ’encodage/décodage
    - dans le JDK :

      static String java.net.URLDecoder.decode(String urlencoded)

      static String java.net.URLEncoder.encode(String str)

Récupération d ’un Document
Méthode GET



le Client envoie

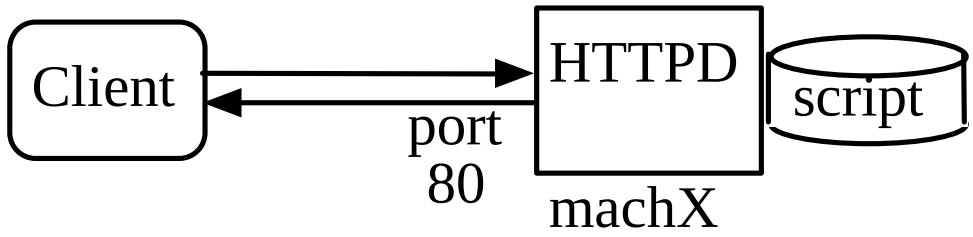
GET /docu2.html HTTP/1.0 *méthode,chemin,version*
Accept: www/source *documents acceptés*
Accept: text/html
Accept: image/gif
User-Agent: Lynx/2.2 libwww/2.14
From: alice@pays.merveilles.net
*** une ligne blanche ***

le Serveur répond

HTTP/1.0 200 OK *ligne de status*
Date: Wed, 02Feb97 23:04:12 GMT
Server: NCSA/1.1
MIME-version: 1.0
Last-modified: Mon,15Nov96 23:33:16 GMT
Content-type: text/html *type du document retourné*
Content-length: 2345 *sa taille*
*** une ligne blanche ***
<HTML><HEAD><TITLE> ...

Soumission d ’un Formulaire
Méthode GET

GET /script?name1=value1&name2=value2



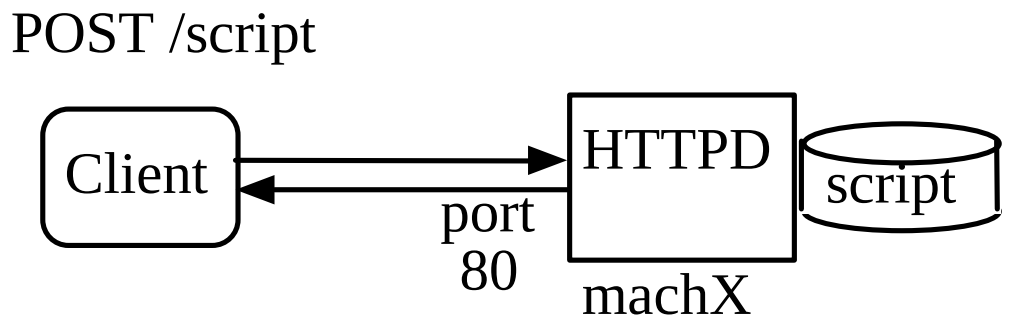
le Client envoie

```
GET /script?name1=value1 HTTP/1.0
Accept: www/source
Accept: text/html
Accept: image/gif
User-Agent:  Lynx/2.2  libwww/2.14
From:  alice@pays.merveilles.net
    * une ligne blanche *
```

le Serveur répond

```
HTTP/1.0 200 OK
Date: Wed, 02Feb97 23:04:12 GMT
Server: NCSA/1.1
MIME-version: 1.0
Last-modified: Mon,15Nov96 23:33:16 GMT
Content-type: text/html
Content-length: 2345
     * une ligne blanche *
<HTML><HEAD><TITLE> ...
```

Soumission d ’un Formulaire
Méthode POST



le Client envoie

```
POST /script HTTP/1.0
Accept: www/source
Accept: text/html
Accept: image/gif
User-Agent:  Lynx/2.2  libwww/2.14
From:  alice@pays.merveilles.net
    * une ligne blanche *
name1=value1&
name2=value2
```

le Serveur répond

```
HTTP/1.0 200 OK
...
Content-length: 2345
    * une ligne blanche *
<HTML><HEAD><TITLE> ...
```

Comportement du Client face au type du document retourné

- A partir du type MIME de Content-Type
  - Visualisation native
    - la fonction de visualisation est dans le noyau (core) du client

      text/html, image/jpeg
  - Visualisation par plugin
    - la fonction est présente dans un DLL, SO, ou un JAR
    - elle est liée dynamiquement pour réaliser la visualisation

      world/vrml, text/tex
  - Visualisation externe
    - la fonction n ’est pas présente dans le client
    - le client rapporte le document et le sauvegarde dans un fichier temporaire

      video/mpeg, application/postscript

Requête Multi-parties (multipart)

- Motivation
  - Requête multi-document [RFC1867]
    - formulaire HTML contenant des Upload de fichiers

```
<FORM ACTION="/servlet/UploadTest" ENCTYPE="multipart/form-data"
   METHOD=POST>
Your name? <INPUT TYPE=TEXT NAME=submitter> <BR>
Your first file to upload? <INPUT TYPE=FILE NAME=file1> <BR>
Your second file to upload? <INPUT TYPE=FILE NAME=file2> <BR
<INPUT TYPE=SUBMIT>
</FORM>
```

  - Remarque : Mail multi-documents
    - (fichiers attachés, mail enrichi d ’images, audio-mail …)

Requête Multi-parties (multipart)
Codage de la requête

Content-Type : multipart/form;boundary=End9989822
--End9989822

Content-Disposition; form-data; name="file1"; filename="test.htm"
Content-Type : text/html

<HTML><BODY> Ceci est un fichier de test !</BODY></HTML>
--End9989822

Content-Disposition; form-data; name="file2"; filename="test2.txt"
Content-Type : text/plain

Ceci est un deuxieme fichier de test !
--End9989822

- Voir [Hunter Ex4-18 p 107]

Réponse Multi-parties

- Codage
  - Content-Type : multipart/x-mixed-replace;
  - Frontière entre les parties
    - Déclaration : boundary=*chaîne_aléatoire*
    - Séparateur : --*chaîne_aléatoire*

- Comportement
  - le navigateur affiche le sous-document suivant dès qu ’il commence à le recevoir après avoir effacer la fenêtre.
    - Voir [Hunter ex6-12 p193]

Réponse Multi-parties
Exemple

Content-Type : multipart/x-mixed-replace;boundary=End65577565679001838

--End65577565679001838
Content-Type : text/html

<HTML><BODY><H1>Un ... </H1><BODY></HTML>
--End65577565679001838
Content-Type : text/html

<HTML><BODY><H1>Deux ... </H1><BODY></HTML>
--End65577565679001838
Content-Type : text/html

<HTML><BODY><H1>Trois ... </H1><BODY></HTML>
--End65577565679001838
Content-Type : text/html

<HTML><BODY><H1>Partez ! </H1><BODY></HTML>
--End65577565679001838

Suivi de Sessions avec HTTP
(Session Tracking)

- Motivations :
  - La notion de session est importante dans une application conversationnelle
    - commerce électronique
      « j ’ajoute ce produit à mon panier (existant)»
  - Cependant HTTP est un protocole « stateless »
    - le serveur ne maintient pas d ’informations liées aux requêtes précédentes d ’un même client.
    - HTTP est donc « sessionless »
  - Comment implanter la notion de session sur plusieurs requêtes HTTP
    - documents, CGI, SSS, Servlet, ASP

Suivi de Sessions avec HTTP
(Session Tracking)

- Méthodes
  - Le serveur génère un identificateur de session et associe un état (et une date limite de validité) à une session
  - Le client renvoie l ’identificateur de session à chaque requête HTTP vers le serveur

- Echange et Stockage de l’identificateur de session
  - Input HIDDEN dans les formulaires
  - Réécriture des URLs (EXTRA_PATH)
  - Cookies (déactivable)
  - Identificateur de session SSL (Secure Socket Layer)

Suivi de session avec HTTP

- Une session s ’étend sur plusieurs requêtes
  - documents, CGI, SSS, Servlet, ASP
  - le serveur maintient un contexte de session et y associe un identifiant de session
- 3 solutions de suivi
  - input HIDDEN
    - contient l ’identifiant de la session
  - la Ré-écriture d ’URL
    - l ’identifiant dans chaque URL (dans les documents)
  - les Cookies
    - information positionnée par le serveur sur le client
      la durée de vie du cookie dépasse la session
    - puis envoyé par le client à chaque requête
- Implantation
  - objet Session (ASP), classe HttpSession (JSP/Servlet)

Suivi de Session
une entrée HIDDEN par formulaire (i)

- Chaque réponse retournée par le serveur est un formulaire qui contient un identifiant caché dans une entrée HIDDEN
- Exemple

• ***page de proposition***

```
<FORM METHOD="POST" ACTION="/cgi-
bin/command">
   <INPUT TYPE="ckeckbox" NAME="art12387">
Chaussures
   ...
</FORM>
```

• ***réponse de /cgi-bin/command***

```
<FORM METHOD="POST" ACTION="/cgi-bin/envoi">
   <INPUT TYPE="hidden" NAME="TransID"
VALUE="54109848932>
   Nom:                          <INPUT TYPE="text"
NAME="nom">
   Adresse:                          <INPUT
TYPE="text" NAME="adresse">
   N° de Carte de Credit:        <INPUT
TYPE="text" NAME="numcarte">
   ...
   <INPUT TYPE="hidden" NAME="Language"
VALUE="French>
</FORM>
```

Suivi de Session
une entrée HIDDEN par formulaire (ii)

- Inconvénients
  - dialogue uniquement par formulaire
    - car pas de persistance de l'identifiant côté client
  - Ambiguïté dans le cas des retours-arrière de l ’utilisateur
    - annulation d ’une série d ’actions
    - ou série d ’action supplémentaire

Suivi de Session
la ré-écriture des URLs

- L ’identifiant de session est encodé dans les URLs des documents HTML retourné par le serveur.
  - Dans le PATH

    `http://www.mycomp.com/cgi-bin/envoi?name=toto`

    devient

    `http://www.mycomp.com/182993954/cgi-bin/envoi?name=toto`
  - Dans l ’EXTRA-PATH

    `http://www.mycomp.com/cgi-bin/ envoi?name=toto`

    devient

    `http://www.mycomp.com/cgi-bin/envoi/sid$182993954?name=toto`
- Limites
  - URL générée par un script (=>programmation)
  - ou parsing des documents HTML retournés
    - mais disfonctionnement en présence de scripts JavaScript ou VBScript générant eux aussi des URL !

Suivi de Session
les Cookies [Netscape puis RFC2109]

- chaîne décrivant l'état d'une session
  - NAME=VALUE;
  - expires=DATE;
  - path=PATH_HEAD; / << /foo << /foobar ou /foo/bar.html
  - domain=DOMAIN_TAIL; fr << mycomp.fr << sales.mycomp.fr
- stocké sur le client
  - Limite
    - 300 cookies simultanées par client, 20 cookies par serveur ou domaine, 4Ko par cookie (limite la taille des VALUEs)
- communiqué dans les entêtes de requêtes et dans les entêtes des réponses HTTP
- accessible par les scripts JavaScript dans une page HTML

Positionnement des Cookies

Client demande

GET /registration.html

⇄ **Server sales.mycomp.fr répond**

Set-Cookie: CUSTOMER=DUPONT;
path=/; expires=Monday, 09-Nov-96

Client demande

GET /command.html
Cookie: CUSTOMER=DUPONT;

⇄ **Server sales.mycomp.fr répond**

Set-Cookie: PARTNUM=01;
path=/command;

Client demande

GET /shipping.html
Cookie: CUSTOMER=DUPONT;

⇄ **Server sales.mycomp.fr répond**

Set-Cookie: SHIPPING=FEDEX;
path=/shipfexed;

Client demande

GET /commandother.html
Cookie: CUSTOMER=DUPONT;
PARTNUM=01;

⇄ **Server sales.mycomp.fr répond**

Set-Cookie: PARTNUM=02;
path=/command;

Client demande

GET /shipfedex1.html
Cookie: CUSTOMER=DUPONT;
SHIPPING=shipfexed;

→

...

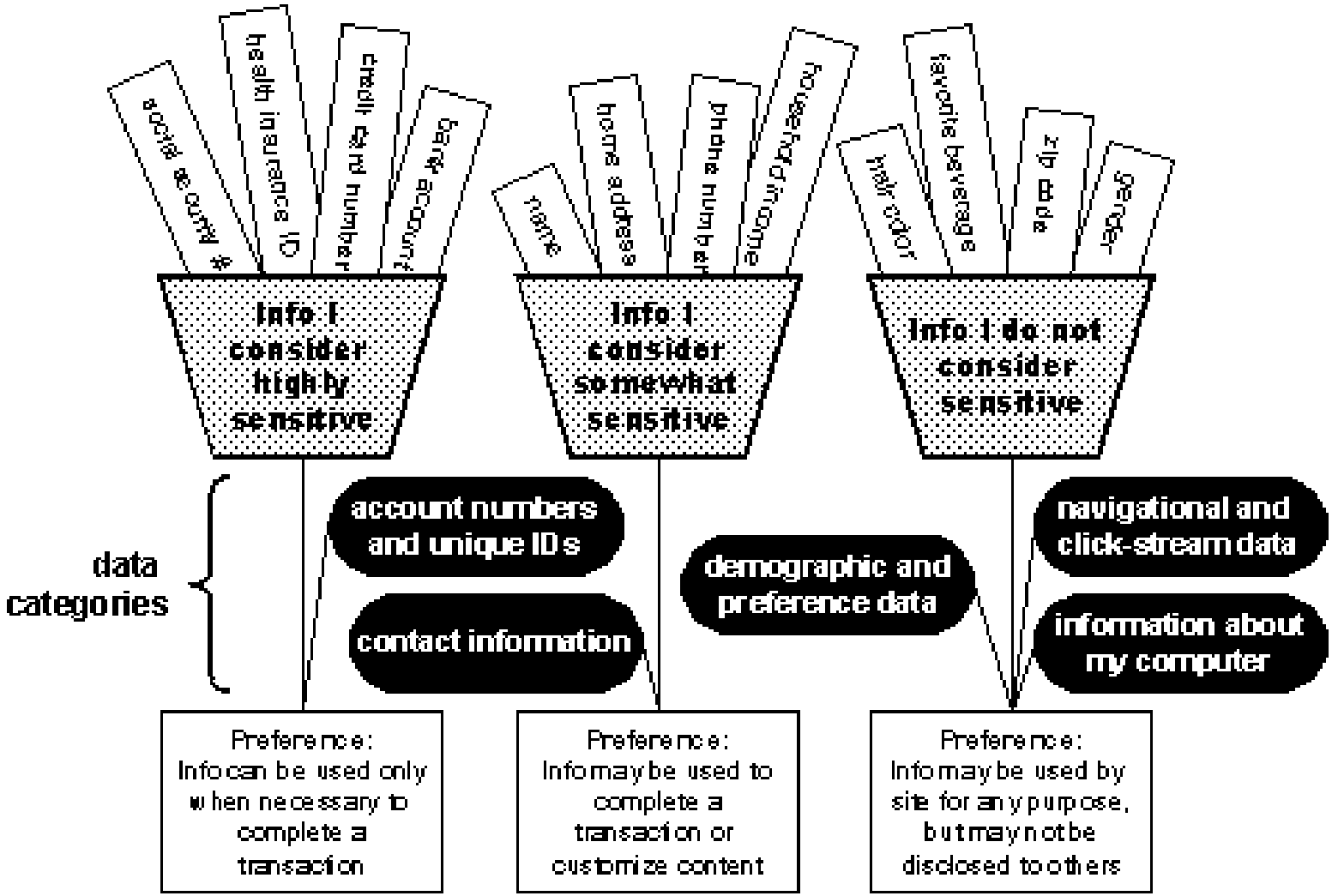
L ’évolution des Cookies

- Les cookies menacent la vie privée (privacy) des cybernautes bien qu ’ils soient très utiles
- les navigateurs peuvent désactiver les cookies

Un remplacant : P3P (Platform for Privacy Preferences)

- en vue d ’un accord juridique entre le client et le site sur
  - la définition du champs des divulgations
    ex : nom, prénom, adresse mais pas l ’age ou le nombre d ’enfants
  - définition de l ’utilisation de ces données par le propriétaire du site
    ex : cession des informations à des tiers
  - définition de la procédure de modification des données ultérieurement.
    ex: je me suis marié
- TUID/PUID Temporary et Pairwise Unique ID
  - identifiants de session (et multi-sessions) sans information attachée
- P3P exprimé en RDF/XML, Certificats / Signatures

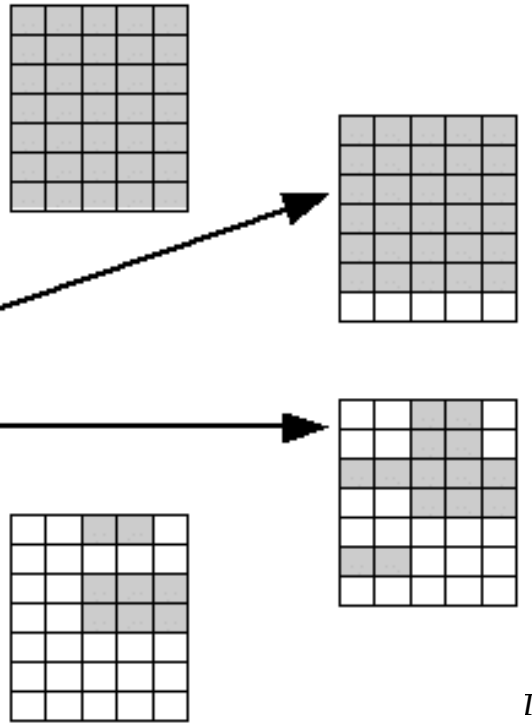
Catégories des Informations Personnelles



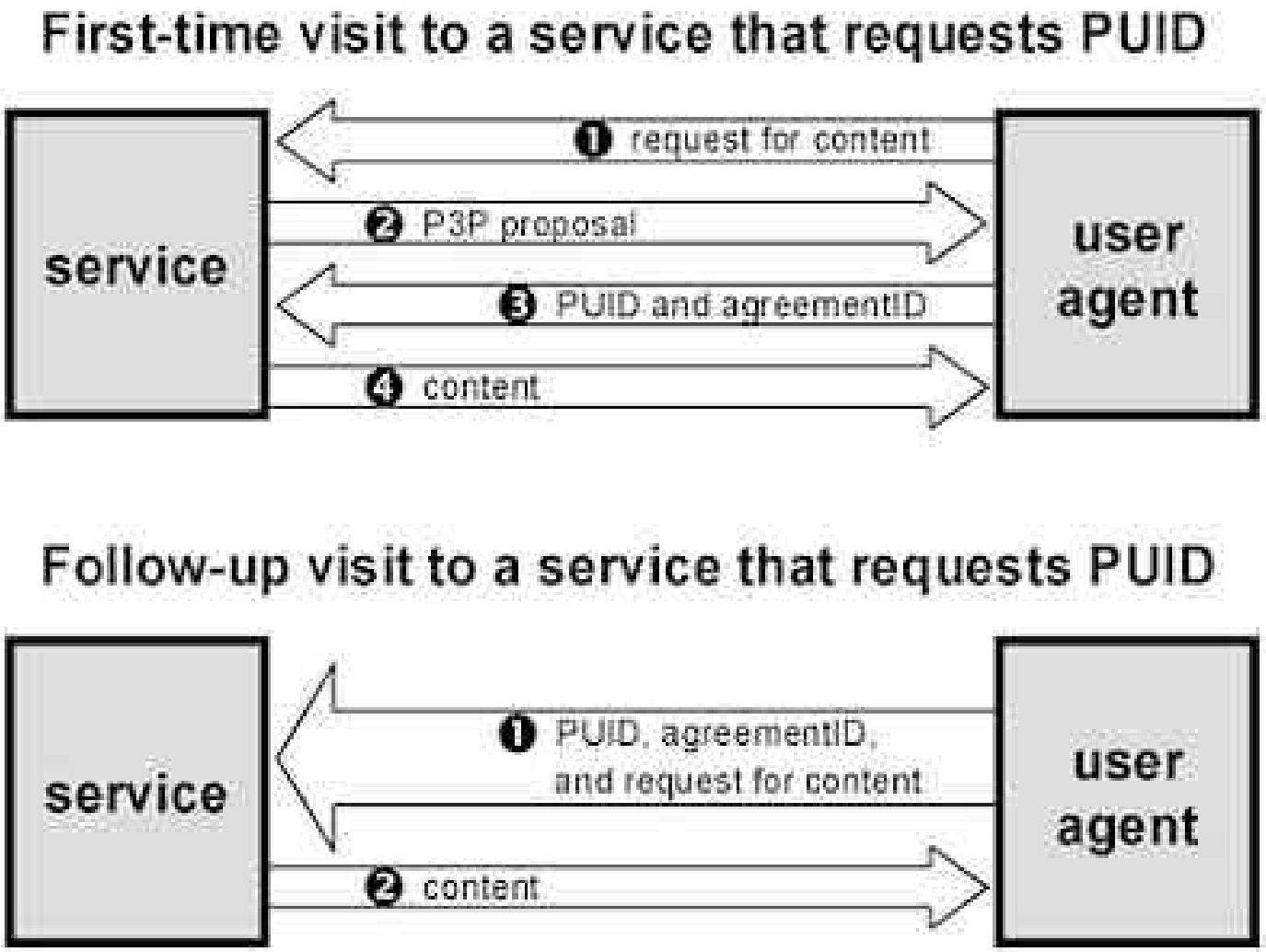
Divulgation d ’informations personnelles

Sample Privacy Vocabulary		Data Categories				
		Contact information	Account numbers and unique IDs	Information about my computer	Navigational and click-stream data	Demographic & preference data
Data Practices	Used for system administration					
	Used for research and/or product development					
	Used for completion and support of current transaction					
	Used for customization of content and/or design of site					
	Used for marketing purposes					
	Used for linking other collected information					
	Disclosed for marketing purposes					

Sample Recommended Settings
No privacy – Sites may share my data with others for any purpose
No disclosure – Sites may only use my data internally
Moderate privacy – Sites may only use my data for the purpose for which I supplied it
Anonymous surfing – Sites may not collect identifiable data from me



Echange P3P


First-time visit to a service that requests PUID
service
user agent
1 request for content
2 P3P proposal
3 PUID and agreementID
4 content
Follow-up visit to a service that requests PUID
service
user agent
1 PUID, agreementID, and request for content
2 content

Exemple de proposition P3P

```
<PROPOSAL realm="http://www.CoolCatalog.com/catalogue/"
        entity="CoolCatalog" agreeID="94df1293a3e519bb"
        assurance="http://www.GoodPrivacy.org">
        <USES>
          <STATEMENT purpose="1" recipient="0" id="0">
           <REF name="Web.Abstract.ClientClickStream"/>
          </STATEMENT></USES>
        <USES>
          <STATEMENT purpose="2,3" recipient="0" id="0"
           consequence="a site with clothes you'd appreciate.">
           <WITH><PREFIX name="User.">
            <REF name="Name.First"/>
            <REF name="Bdate.Year" OPTIONAL="1"/>
            <REF name="Gender"/>
           </PREFIX></WITH>
        </STATEMENT></USES>
       <DISCLOSURE discURI="http://www.CoolCatalog.com/PrivacyPractice.html"
        access="3" other="0,1"/>
</PROPOSAL>
```

HTTP-NG *Next Generation*

- Problématique
  - Protocoles Existants : HTTP/1.0 et HTTP/1.1
    - standardisé par le W3C
    - Extensibilité lourde
- Proposition du HTTP-NG Working Group
    - *http://www.w3.org/Protocols/HTTP-NG*
  - Extensibilité simple, modulaire et par couche
    - modèle d ’objets distribués
- Working Drafts (07/98)
  - Modèle d ’Architecture, Interfaces Web
  - Binary Wire, SMUX

HTTP-NG *SMUX*

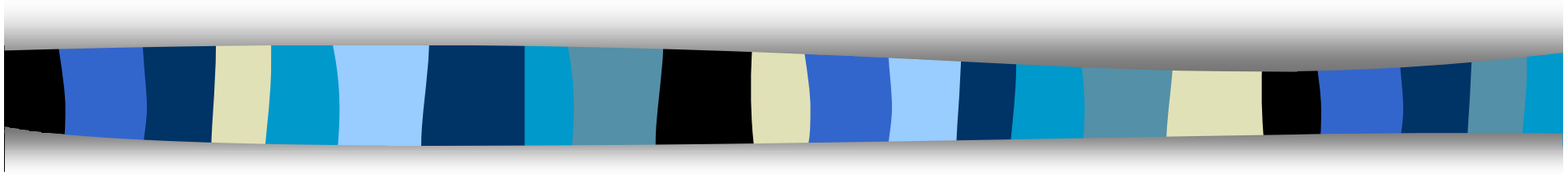
- Problématique de SMUX
  - HTTP : protocole très lié à TCP/IP
  - Consommation des ressources (réseaux, proxy, serveurs)
    - Plusieurs connexions HTTP par document

      document + inclusions : images, applets, …
    - Insuffisance des connexions «KeepAlive» de HTTP/1.1

      reste sérielle, force les navigateurs à « paralléliser »

      » évite les reformatages pour les images de dimensions inconnues
- Multiplexage
  - niveau couche transport
  - TCP et Non TCP

Les clients et les serveurs HTTP



Didier DONSEZ

Université de Valenciennes
Institut des Sciences et Techniques de Valenciennes

donsez@univ-valenciennes.fr

Les clients HTTP

- Butineur ou Arpenteur Web *(browser Web)*
  - Récupère un document puis « visualise » ce document
  - *le tout premier : NCSA Mosaic*
  - *Lynx (pour VT)*
  - *Netscape Communicator*
  - *MS Internet Explorer*
  - *Javasoft HotJava*
  - …
- Palmares d ’utilisation
  - MS IE 86.08%
  - NS 33.43%
  - Reste 0.5%

Rôle d'un serveur HTTP

- Transformation de l'URL en fichier ou en script
- Vérification d'identité
  - Le client est-il qui il prétend être ?
- Vérification d'accès
  - Le client est-il autorisé à effectuer cette requête ?
    - ACL, …
- Constitution de l ’entête de la réponse
  - Type MIME des données
    - mime.types fichier de correspondance extension vers type MIME
  - Taille des données, Langage, ...
- Envoi de la réponse au client
    - éventuellement transformé à la volée
- Mise à jour des journaux d ’audit (log)
  - access_log, error_log, ...

L ’authentification dans HTTP

- Indiqué dans les ACL
- Mode d ’authentification
  - BASIC
    - nom d ’utilisateur et mot de passe échangé en clair (base64) !
    - base des mots de passe dans un fichier htpasswd

      utilitaires de gestion du fichier
  - DIGEST
    - sécurisation de BASIC
    - hachage sécurisé MD5 du (nom,password,URI, méthode,nombre aléatoire fourni par le serveur)
  - SSL
    - Secure Socket Layer (TLS : Transport Layer Security)
    - authentification avec CA du serveur (2.0) et du client (3.0)
    - confidentialité avec DES
    - puis dialogue HTTP sur la connexion SSL

L ’authentification applicative

- Motivations
  - interface de login
  - identification externe
    - BD, Annuaire LDAP, …
  - authentification plus forte
- L’application gère l ’authentification de l ’usager
  - formulaire d ’accueil HTML (nom, password)
    - attention le mot de passe est en clair
  - gestion des tables d ’usager
  - une session est ensuite ouverte associé à un usager authentifié (ou non : par exemple rejet à bout de 3 tentatives)

Contrôle d ’Accès dans HTTP

- ACL (Access Control List)
  - spécifie les autorisations (ALLOW) ou les interdictions (DENY) d ’accès à une arborescence virtuelle du serveur
  - en fonction
    - de l ’authentification
    - de la localisation du client

      sous domaine DNS

      réseau ou adresse IP
- ACF (Access Control File)
  - fichier regroupant les ACL
    - global : access.conf dans Apache
    - par arborescence : .htaccess
  - combinaison des ALLOW et des DENY

Audit des Requêtes

- Journaux des requêtes
  - les accès (access.log, refferee.log), et les erreurs (error.log), … sont journalisés
- Exploitation des Journaux
  - erreur dans les liens, …
  - clientèle, analyse d ’activité, …
- Reporting (Présentation Synthétique)
  - Pour Apache
    - AccessWatch, Wusage,Analog, wwwstat
  - IIS, NS
    - intégré et visualisé par un script
  - Généraux
    - Net Analysis (Net Genesis), Enterprise Suite (Web Trends)

Les Serveurs du Marché

- Offre très large
  - Apache HTTPD
  - Netscape Entreprise Server
  - Microsoft Internet Information Server
  - W3C Jigsaw
  - Sun JavaServer
  - Oracle Web Server
  - IBM Web Sphere
  - …
- Fonctionnalités supplémentaires
  - gestion des sessions, des transactions, …
  - accès aux serveurs d ’applications, ...

Apache
(www.apache.org, java.apache.org, xml.apache.org)

- A patch of NCSA HTTPD
  - serveur le plus répandu (60% des serveurs au 09/2000)
  - gratuit, issu du serveur NCSA HTTPD
  - très nombreuses plates-formes Unix et Windows NT
  - extensible par des modules tiers
- Nombreux Modules Tiers
  - possibilité d'étendre Apache avec des modules externes (*http://www.zyzzzyva.com/server/module_registry*)
    - mod_auth_cookies_file, mod_auth_cookies_msql, mod_cgi_sugid, mod_perl, mod_perl_fast, mod_auth_kerb, mod_auth_dbi, mod_rewrite, mod_jserv (servlet), mod_java (CGI écrit en Java), php3
  - nombreux sous-projets autour de Java (Jakarta) et XML (Xerces, Xalan, XSP, Cocoon, …)

Configuration Apache

- Fichiers de configuration
  - httpd.conf
    - comportement de base

      port TCP/IP, journaux, keepalive, UID, virtualhost, proxy, ...
  - srm.conf
    - traitement des ressources locales lors des requêtes

      index, script, répertoire, AddType, AddIcon, Alias, DocumentRoot
  - access.conf
    - contrôle d ’accès global (ACF : access config file)
  - mime.types
    - table de correspondance

      suffixe fichier -> type MIME document
- Outil
  - GUI : Vision (focus-array.com)
  - …

JavaServer *(jserv.sun.com)*

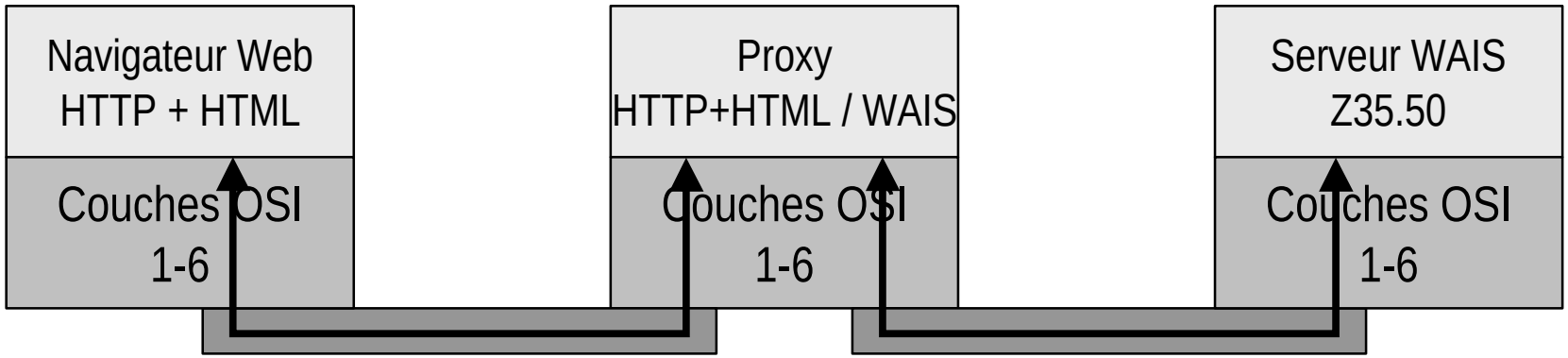
- Serveur HTTPD de SUN
  - anciennement « Jeeves », écrit en Java
- Servlets
  - Scripts serveur écrit en Java
  - Servlets de Base : FileServlet, CGIServlet, …
- Remarque
  - le bytecode d ’une servlet peut être téléchargé et executé par la JVM du Serveur (dans un espace protégé ou non)
  - JigSaw du W3C fonctionne suivant le même principe

Autour d ’HTTP

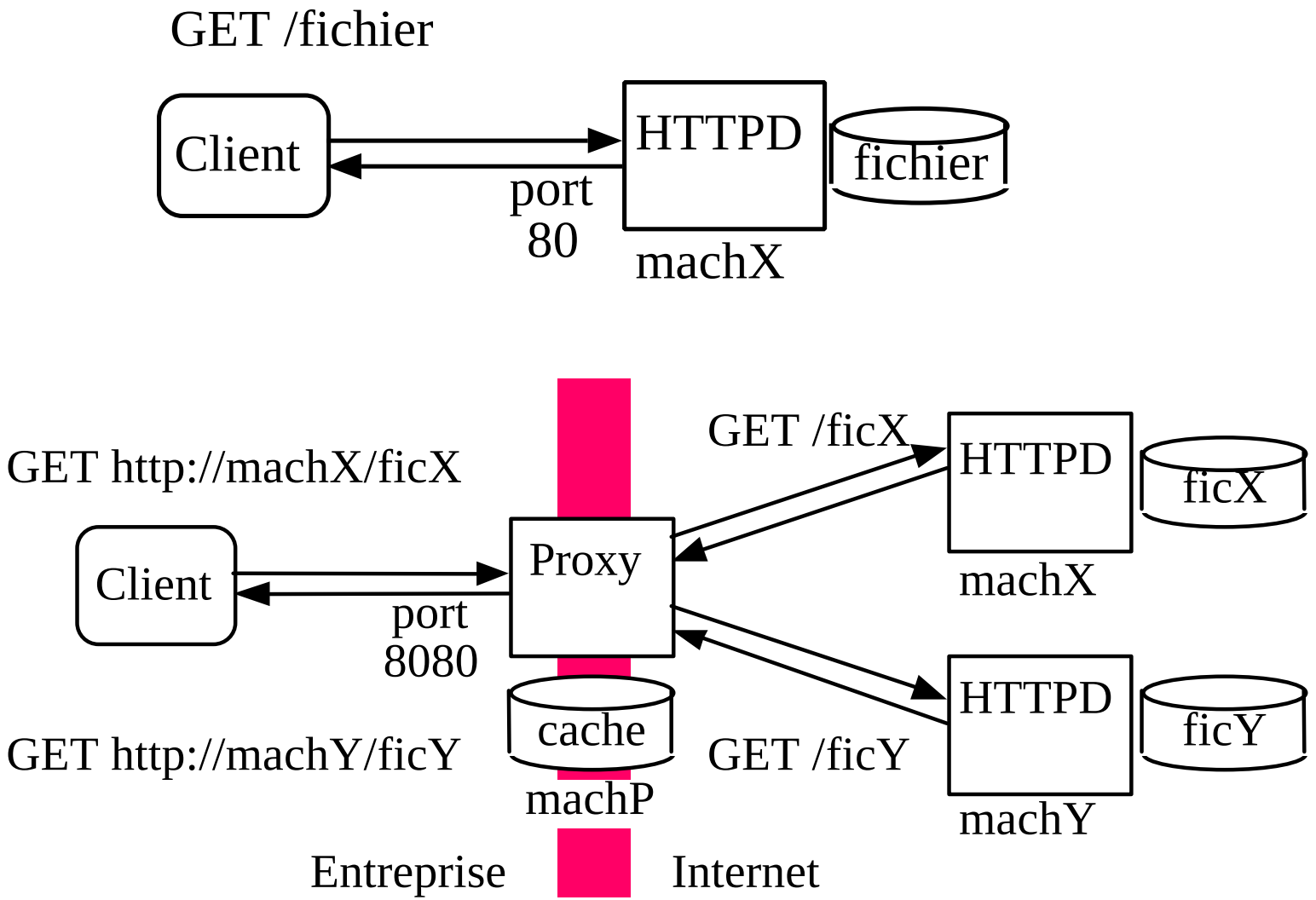
- Proxy
  - seul point de passage entre le réseau d ’entreprise et l extérieur
    - sécurité, firewall
  - accès à des protocoles non implémentés par les clients Web
    - WAIS
- Cache
  - soulager les accès externes (moins de bande passante)
- Miroir
  - réplication d ’une base documentaire
  - améliorer le temps de réponse, soulager le réseau
- Robot
  - récupération online/offline d ’une arborescence de documents
  - constitution d ’un miroir local
- Mise à jour de sites
  - copie publique / copies de travail

Proxy

- Fonctions
  - seul point de passage
    entre le réseau d ’entreprise et l ’extérieur
    - firewall, contrôler le profil d'utilisation
  - accès à des protocoles non implémentés
    par les clients Web
    - WAIS
- Passerelle réseau de niveau applicatif



Fonctionnement du Proxy/Cache



Cache i

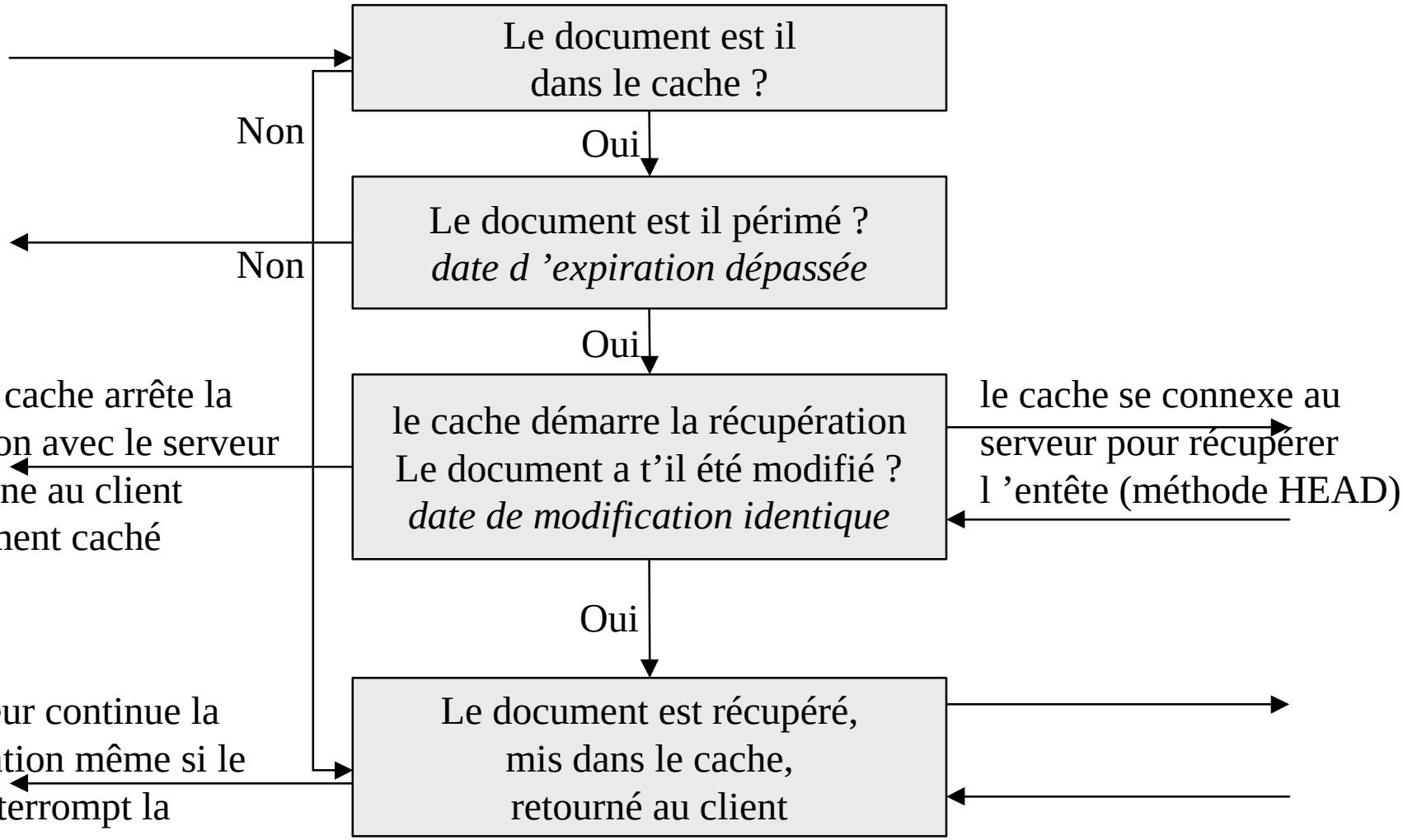
- But d ’un cache Réseau
  - soulager le réseau fédérateur
    en cachant les documents (sauf CGI)
    demandés par les usagers du sous-réseau
- Documents
  - Textes (HTML, XML, ...) et Images fixes
  - Flux Audio et Video
    - têtes de réseau cable pour la Buffered-VOD
- Serveurs
  - Il fonctionne en mode Proxy
  - Squid, Harvest, Apache, MicroSoft, Sun, Netscape , ...

Cache ii

- Cache de Caches
  - Hiérarchie multi-niveaux de caches
    - exemple : RENATER
    - *cache Client -> cache Sous Réseau -> cache Backbone -> ...*
  - équilibrage dynamique
    entre plusieurs caches de même niveau

- ICP : Inter Cache Protocol
  - mise à jour
  - redirections de requête

Gestion du Cache



Miroir

- But
  - Créer une copie miroir d ’un site (d ’une partie de site)
    - accélère l ’accès au document
    - diminue la contention d ’accès au site
    - diminue le trafic global sur Internet
- Mise à jour du miroir
    - déclenchement automatique, manuel, push
    - incrémentale / non

Robot
(Aspirateur de Site, Glaneur, ...)

- Récupération d ’une arborescence de documents à partir d ’une URL « racine »
  - Règles de récupération
    - par rapport aux extensions (que les textes)
    - profondeur du suivi de URL et limite de la récupération
  - Déclenchement programmé
    - heures creuses (débit et $)
- Produits
  - nombreux freewares/sharewares
    - webcopy, wget, w3mir, ...
  - Fonction miroir intégrée dans les logiciels auteurs
    - MS FrontPage, Goto MemoWeb, ...
- Remarque
  - permet de créer facilement un miroir

Mise à jour des sites

- Actuellement, chaque auteur travaille sur une copie locale et remplace la copie publiée sur le Web par celle-çi régulièrement

Remplacement des pages

- par FTP
  - problème de gestion des versions multiples lors d ’un travail de groupe
- par un script HTTP serveur propriétaire
  - et information propriétaire associée
  - utilisé par MS FrontPage, NetObjects Team Fusion, Macromedia DreamWeaver, …
  - mais pas d ’interopérabilité
- WebDAV [RFC 2518]
  Web Distributed Authoring and Versioning Protocol
  - le dernier effort de standardisation

WebDAV (i)

Web Distributed Authoring and Versioning Protocol

- Extension de HTTP pour la mise à jour de site (RFC 2518)
  - www.webdav.org, www.ics.uci.edu/pub/ietf/webdav

Notions

- Propertie (propriété)
  - décrit un document (auteur, taille, date de dernière modification, ...) au format XML/RDF
- Collection
- Locking (verrouillage)
  - verrou partagé (shared) / verrou exclusif (exclusive)
- Namespace (espace de nommage)
  - groupement logique de ressource pour la gestion (verrouillage, contrôle d 'accès, ...)

WebDAV (ii)

Web Distributed Authoring and Versioning Protocol

- Ajout de nouvelles commandes HTTP
  - PROPFIND
    - retourne les propriétés
  - PROPPATCH
    - modifie les propriétés
  - MKCOL
    - crée une nouvelle collection
  - COPY & MOVE
    - copie ou déplace une ressource au sein d ’un espace de nommage
  - LOCK & UNLOCK
    - verrouille et déverouille un ressource

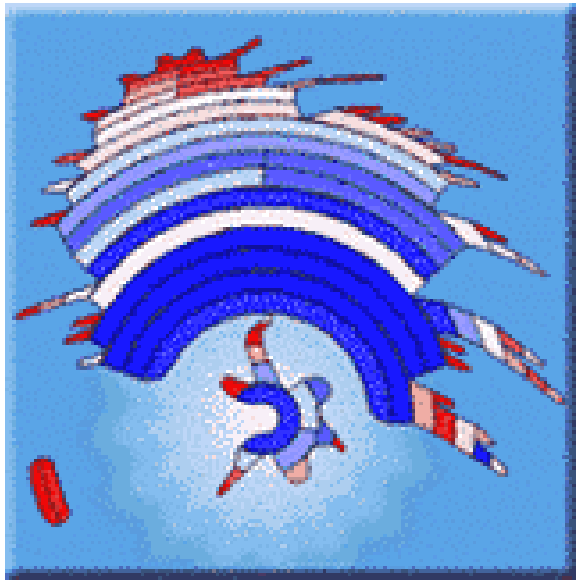
WebDAV (iii)

Web Distributed Authoring and Versioning Protocol

- des Serveurs WebDAV
  - MS IIS5
  - mod_dav pour Apache
    - www.webdav.org/mod_dav
  - CyberTeams ' Web Site Director
- des Clients WebDAV
  - MS Office 2000 et Explorer 5
  - SiteCopy
    - www.lyra.org/sitecopy
  - WebDAV Explorer
    - www.ics.uci.edu/~webdav

Cartographie du Web

- Grouper les sites en « régions »
  - en fonction de leur relation sémantique, des références d ’URL
- Représentation de la carte
  - hyperbolique, cible 2D, …
- Outils et sites
  - www.semio.com, www.acetic.fr, www.umap.com, ...



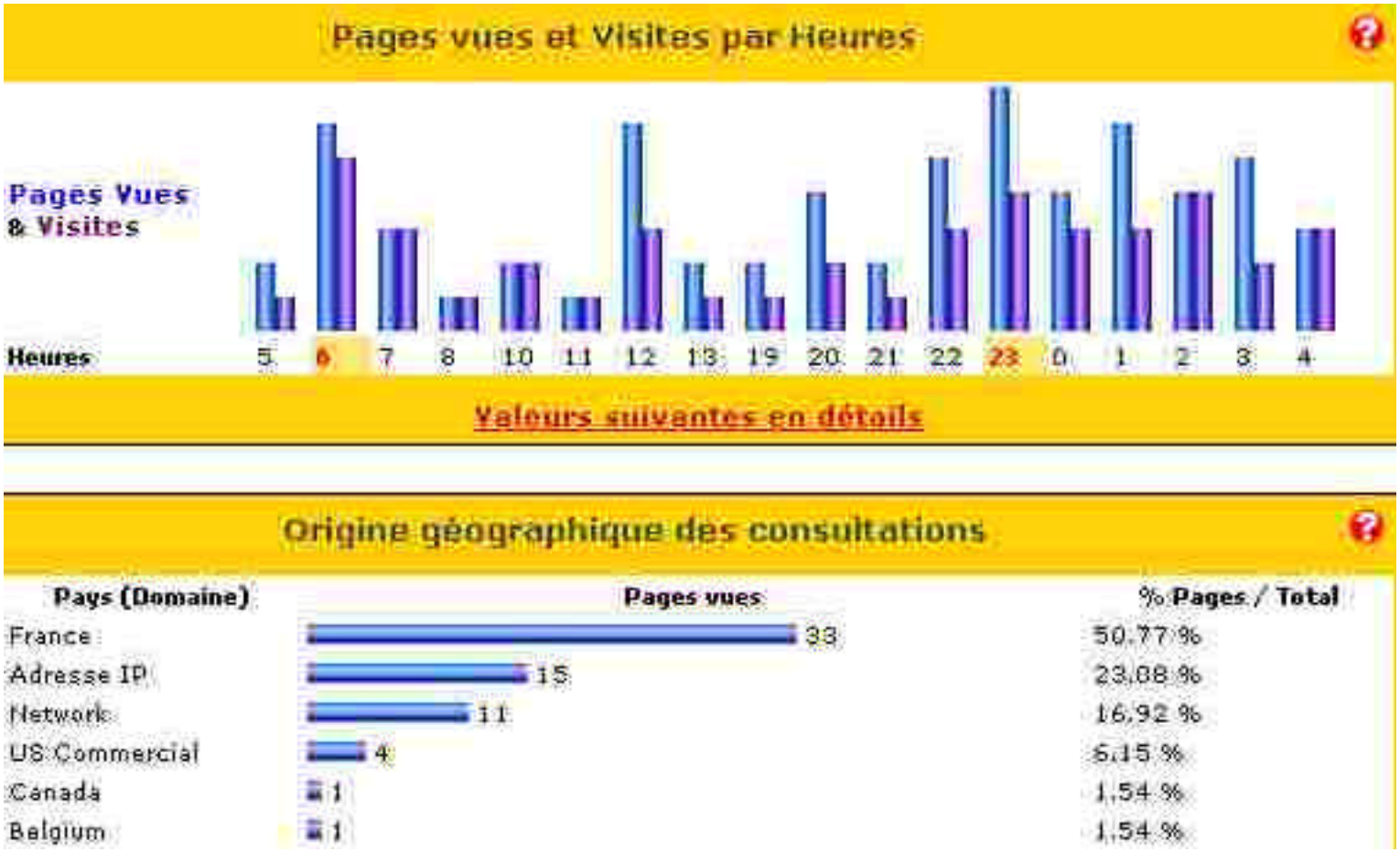
Rapports d ’audience

- Pourquoi faire : suivi et analyse de l ’activité du site
  - webmaster : dimensionner le système (période de charge)
  - commercial : pister le client dans sa navigation
  - investisseur : le rassurer avant l ’introduction en Bourse ;-)
  - publicitaire : fixer le prix des bandeaux publicitaires
- Comment
  - à partir des journaux du serveur HTTP (access.log)
  - externalisation
    - société tierce « indépendante » proposant des rapports « normalisés » (Exemple : www.estat.com, ...).
    - chaque page à tracer contient une image transparente de 1x1 pixel qui est chargée depuis la société tierce. La société tierce comptabilise les chargements.

      <IMG SRC="http://www.tierce.com?idcust=190874+doc=/product/index.htm">

Rapports d ’audience
Exemple

- Rapport d ’audience de www.estat.com



La personnalisation

- Motivations
  - Les usagers d ’un site ont des goûts et des besoins différents
  - La personnalisation tente d ’offrir à l ’usager une interface correspondant le plus possible à ses goûts et ses besoins
- Solutions
  - Analyse *(E-Analytics)*
    - Analyse des navigations précédentes de l ’usager

      Net Perceptions, Accrue Software, NetGenesis Corp
  - Personnalisation sur règles *(Rules-based Personalization)*
    - Création des pages personnalisées à la volée en fonction de régles

      BroadVision, Vignette Corp.
  - Filtrage collaboratif *(Collaborative Filtering)*
    - similarité entre utilisateurs : comportement de groupes

      Net Perceptions, Macromedia, Be Free

Test de performance / charge

- Motivation
  - Vérification du niveau de charge supporté avant la mise en production
- Simulateur de charge
  - 1à N clients en parallèle qui émulent M Web surfing sur le site
  - Remarque : Mêmes outils pour les attaques DoS (Deny-Of-Service) et Distributed DoS
- Benchmark
  - TPC-W http://www.tpc.org (voir Cours Benchmarks BD)
- Analyse et diagnostique de la charge
  - Détecter quel composant (HTTPD, EJB, MT, SGBD, Mailhost, …) crée l'embouteuillage (*bottleneck*)

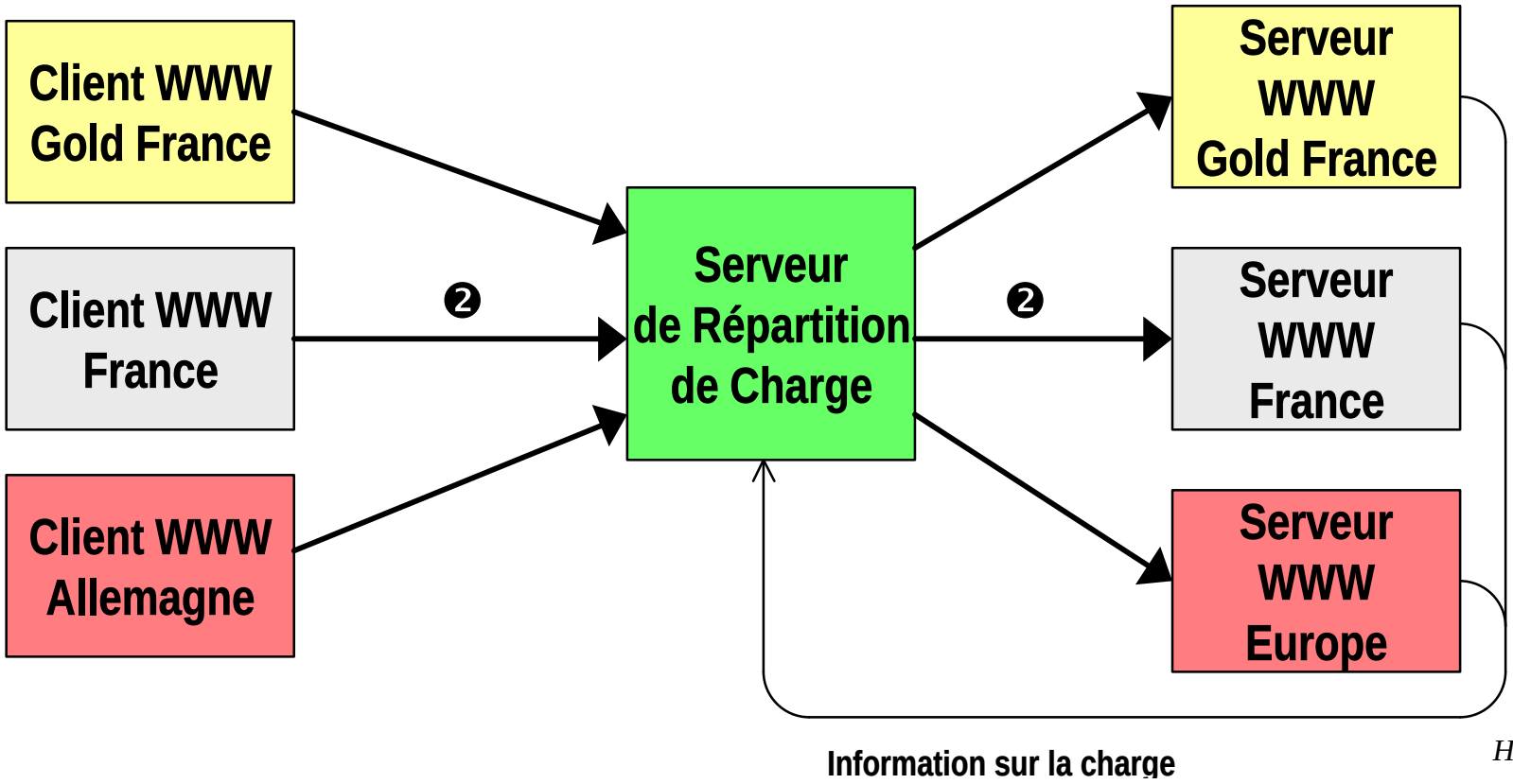
Répartition de Charge *(Load Balancing)* Disponibilité *(Avaibility)*

- Motivation
  - Performance
    - améliorer les temps de réponse
    - QoS pour différentes classes d'usagers (gold, silver, …)
    - Évolution incrémentale de la ferme (parc) de serveurs
  - Tolérance aux pannes
    - fonctionnement dégradé en cas de panne

      Le serveur Europe reçoit les requêtes des clients français

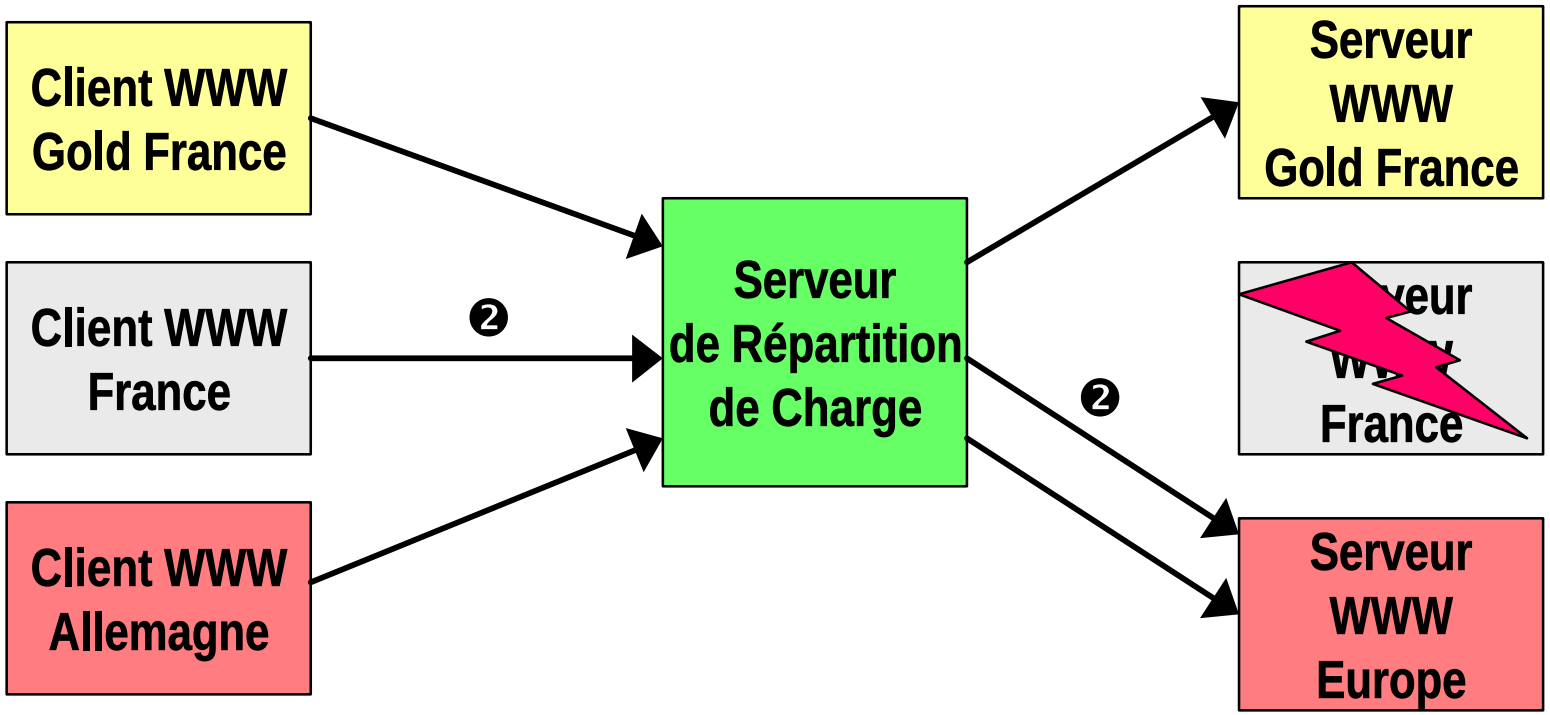
Répartition de Charge *(Load Balancing)*

- Architecture
  - Un serveur de répartition de charge
  - Une ferme de serveurs Web



Disponibilité *(Avaibility)*

- Cas d'une panne
  - Les clients français sont servis par le serveur Europe
    - La panne peut être détectée par des pings réguliers !



Serveur de Répartition de Charge

- Fonction
  - répartir les requêtes entre les serveurs Web
- Politiques de répartition
  - donneur de cartes *(Round Robin)*
  - aléatoire
  - vers les serveurs les moins chargés
  - topologique
    - Optimiser la bande passante de l'infrastructure réseau

      L'entreprise dispose de serveurs répartis sur plusieurs backbones du Web
  - en fonction de la classe d'usagers
    - Offrir différentes QoS (temps de réponse, …)
  - en fonction des sessions en cours
    - Une session démarrée sur un serveur doit continuer sur ce serveur

      SSL, Cookies de session pour Http, Stream du RealServer, …
- Fournisseurs
  - Alteon WebSystems, F5, Foundry Networks, Radware, CISCO, …

Niveaux de la Répartition de Charge par le Serveur de Répartition

- Niveau 3 – IP
  - Répartition par le DNS
    - La résolution DNS n'est pas une constante
      une adresse DNS correspond une liste d'adresses IP
      à tester nslookup plusieurs fois de suite sur www.sun.com
  - Future RFP VRRP (Virtual Router Redondancy Protocol)
    - Grappe de routeurs derrière un routeur virtuel
- Niveau 4
  - Session SSL
- Niveau 7 – Applicatif
  - Protocole HTTP
    - Détection du Cookie de la classe d'usager dans une requête HTTP
    - Détection du Cookie de l'identifiant de session dans une requête HTTP
  - Protocoles FTP, SMTP, …
  - Flux Audio/Vidéo, …

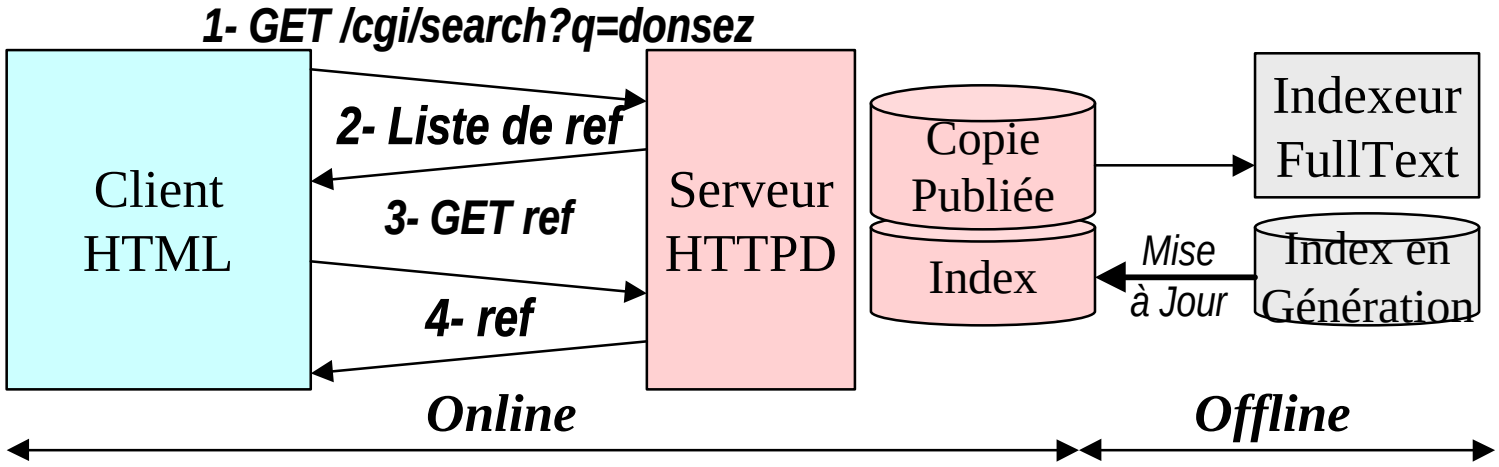
Performance grâce aux caches

- 2 types de page
  - Dynamique
    - Le contenu varie à chaque requête
    - Le contenu peut rester constante pour un client au cours de sa session
  - Statique
    - Le contenu ne varie pas jusqu'à la prochaine modification

      Exemple : Mise à jour des nouvelles du jour tous les matins à 7H00 GMT
- Les caches
  - dans les proxys et dans les navigateurs
  - utilise la date d'expiration du document
- Conseil
  - Pour éviter que le navigateur recharge le document inutile (ce qui charge le serveur), configurez les dates d'expiration des documents sur le serveur Web et dans les entêtes des réponses des scripts.

Les Moteurs d ’Indexation (i)

- L ’utilisateur utilise deux modes de parcours d ’un site
  - Le suivi de liens
  - La recherche Full-Text par combinaison de mots-clé et de concepts
    - `(antoine NEAR(15) cléopatre) AND NOT césar`
- Recherche Plein Texte (Full-Text)
  - pré-indexation du corpus documentaire
    - *Remarque : le corpus peut provenir du Web (i.e. Portails)*
    - *Problème : les documents générés à la volée*
  - le script /cgi-bin/search interroge l ’index généré en Offline



Les Moteurs d ’Indexation (ii)

- Indexeurs (gratuit ou payant)
  - FreeWAIS, Glimpse, ht://Dig, Harvest, AltaVista, Verity Search, AIRS, Basis, Oracle Context, Doris-Floras, RetrievalWare, Virage, QBIC …
- Caractéristiques
  - Indexation incrémentale vs totale
    - l ’ajout d ’un document au corpus ne nécessite pas de reconstruire l ’index
  - Types de document indexés
    - Texte, Texte structuré
    - Multimédia : Images, Sons, Vidéo (MPEG7), Monde 3D …
  - Formats de document analysés
    - HTML, XML, PDF, Word/RTF, GIF, JPEG, MPEG, ...
- Bibliographie
  - C. Leloup, “Moteurs d'Indexation et de Recherche”, Ed. Eyrolles, 1998, ISBN 2-212-08976-7
  - Livre OReilly sur WAIS

HTTP dans le JDK

- Fonctionnalité
  - Implémente directement un client HTTP
    - **java.net.URL**
    - **java.net.URLConnection**
    - **java.net.HttpURLConnection**
    - **java.net.JarURLConnection**
  - récupère un InputStream sur le document distant
- Usage
  - robot de récupération, ...
  - dialogue client-serveur en **tunneling** HTTP/TCPIP
    avec un serveur HTTP
    - voir l ’exemple du compteur sur [Orfali]
- Voir le cours « Programmation Réseau en Java »

HTTP dans le HTML

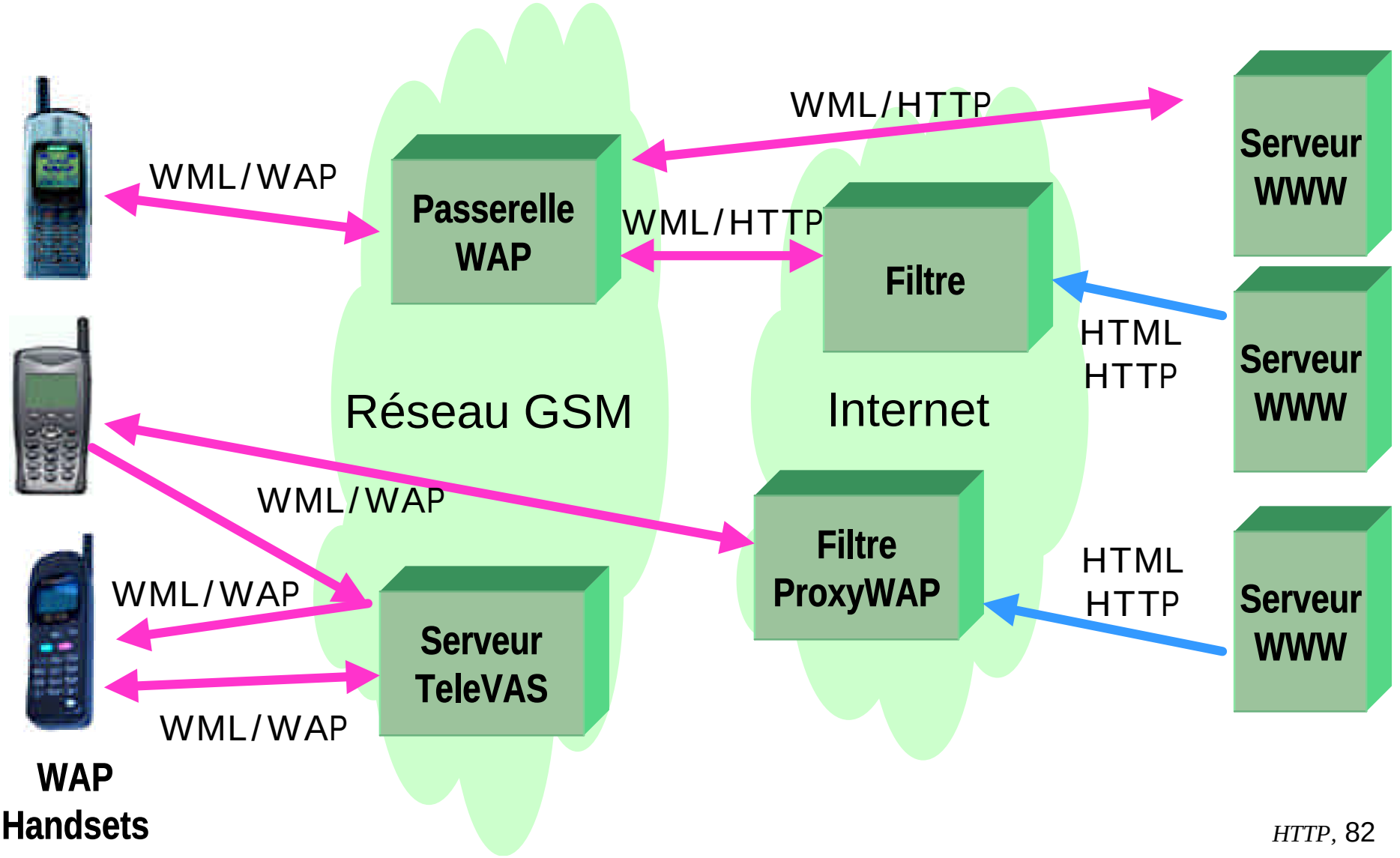
- Eléments META dans l ’élément HEAD d ’un document HTML 3.2
  - <META HTTP-EQUIV=name CONTENT=value>
    - équivalent à ajouter des Header ’s
    - pour documents statiques
- Exemple

```
<HTML><HEAD>
<META HTTP-EQUIV="Refresh" CONTENT="3; URL= http://newsite.mycomp.com ">
</HEAD><BODY>
<H1> This site moves to <A HREF=" http://newsite.mycomp.com ">
    http://newsite.mycomp.com </A></H1>
</BODY></HTML>
```

WAP et WML

- Motivation
  - Adapter HTTP et HTML aux « handsets » nomades
    - Affichage limité, contrôle limité (i.e. clavier, pointage, …)
    - Débit limité et coût de communication
- WAP *Wireless Application Protocol*
  - protocole HTTP « like » et « light »
    - Utilise les couches réseaux des réseaux cellulaires (à la place d ’IP)
    - Utilise la sécurité des réseaux cellulaires
- WML *Wireless Markup Language*
  - langage hypertextuel adapté aux handsets nomades
    - basé sur XML/DTD : il existe un format plus compact (bytecodé)
- *voir www.wapforum.org et le cours sur WAP et WML*

WAP
Architecture Réseau



WML

Exemple : les prévisions météo

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN
          "http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
    <card id="card1" title="Weather Forecast">
        <p>
        <table columns="3" align="LCC">
        <tr><td>Date</td><td>F&apos;cast</td><td>T &#xB0;C</td></tr>
        <tr><td>M 6/7</td><td><img src="rainy.wbmp" alt="rain"/></td>
            <td>25&#xB0;C</td></tr>
        <tr><td>T 6/8</td><td><img src="partcldy.wbmp" alt="part cldy"/>
            </td><td>27&#xB0;C</td></tr>
        <tr><td>W 6/9</td><td><img src="cloudy.wbmp" alt="cloudy"/></td>
            <td>24&#xB0;C</td></tr>
        <tr><td>T 6/10</td><td><img src="rainy.wbmp" alt="rainy"/></td>
            <td>28&#xB0;C</td></tr>
        <tr><td>F 6/11</td><td><img src="sunny.wbmp" alt="sunny"/></td>
            <td>29&#xB0;C</td></tr>
        </table>
        </p>
    </card>
</wml>
```

Bibliographie

- Beaucoup de Guides, Tutoriels, Manuels
  - http://ds.internic.net/rfc/rfc2068.txt
  - http://www.w3.org
  - http://search.yahoo.fr/search/fr?p=HTTP
- Des livres
  - *Attention, ca change très vite !*
  - *La traduction en français a au moins 1 an de retard sur la version anglaise*

Bibliographie - HTTP

- Stephen Spainhour & Robert Eckstein, « Webmaster in a Nutshell », 2nd Edition, June 1999 (est.), ISBN 1-56592-325-1, Ed : O'Reilly
  - très complet
- Robert Orfali, Dan Harkey, « Client/Server Programming with Java and Corba », 2ème édition, 1998, Ed Wiley, ISBN 0-471-24578-X. voir les chapitres 11 et 12
  - compare HTTP à Corba
- Clinton Wong, « Programmation de clients Web avec Perl », Ed Oreilly, 1997, ISBN 2-84177-050-8
  - *l 'automatisation de requêtes sur HTTP (en Perl)*

Bibliographie - Autres

- Robert Orfali, Dan Harkey, Jeri Edwards, « Client/Server Survival Guide », 3rd edition, February 1999, Ed John Wiley & Sons; ISBN: 0471316156
  - information générale sur les composants d ’un SI
- Louis Rosenfeld, Peter Morville , "Information Architecture for the World Wide Web - Designing Large-scale Web Sites", 1st Edition February 1998, ISBN 1-56592-282-4,OReilly, 226 pages, $24.95
- C. Leloup, “Moteurs d'Indexation et de Recherche”, Ed. Eyrolles, 1998, ISBN 2-212-08976-7