

Les réseaux et internet

David Roche, lycée G Fichet Bonneville (Haute-Savoie)

Vous êtes libres :

- de reproduire, distribuer et communiquer cette création au public
- de modifier cette création

Selon les conditions suivantes :



- **Paternité.** Vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'il vous soutient ou approuve votre utilisation de l'œuvre).



- **Pas d'Utilisation Commerciale.** Vous n'avez pas le droit d'utiliser cette création à des fins commerciales.

- À chaque réutilisation ou distribution de cette création, vous devez faire apparaître clairement au public les conditions contractuelles de sa mise à disposition.
- Chacune de ces conditions peut être levée si vous obtenez l'autorisation du titulaire des droits sur cette œuvre.
- Rien dans ce contrat ne diminue ou ne restreint le droit moral de l'auteur ou des auteurs.

Il est très important d'avoir quelques connaissances de base sur "qu'est-ce que le web et internet", "qu'est-ce qu'un client, qu'est-ce qu'un serveur». Il existe déjà sur internet d'excellents articles qui traitent du sujet, et comme j'essaie d'éviter "de réinventer en permanence la roue", je vous propose ici un cours de l'excellentissime site du zéro écrit par Herby Cyrille alias Cysboy (ce cours est l'introduction d'un cours sur Java EE disponible sur <http://www.siteduzero.com/tutoriel-3-112219-apprenez-a-creeer-des-applications-web-dynamiques-avec-jee.html>).

(début de l'introduction de Cyrille Herby)

Internet : qui ? quoi ? qu'est-ce ?

Je ne vais pas vous faire un historique sur la naissance du web tel que nous le connaissons maintenant, je vais juste vous rappeler le fonctionnement de celui-ci.

Cependant, si certaines personnes souhaitent tout de même en savoir plus sur l'histoire d'Internet, elles peuvent suivre [ce lien](#).

Pour faire court, ne confondez pas Internet avec le web !

Internet est un assemblage de multiples réseaux, tous connectés entre eux. Cet amas de câbles, de fibres optiques... de matériels, pour faire simple, constitue Internet, aussi appelé "le réseau des réseaux".

Le Web est un système de fichiers présent sur des machines (serveurs) transitant par un protocole particulier, consultable grâce à des navigateurs web et fonctionnant SUR Internet ! Le web est donc un système de fichiers que toute personne possédant un ordinateur (ou un téléphone, maintenant...) connecté à Internet peut consulter.

En fait, consulter les fichiers présents sur le web est chose courante, surtout pour vous !
Eh oui ! Surfer sur le web, aller sur le Site du Zéro, consulter vos mails chez votre FAI... tout ceci est en fait de la consultation de fichiers présents sur Internet.

Vous n'êtes pas sans savoir que, dans la majeure partie des cas, on surfe sur le web avec un navigateur tel que Firefox, Internet Explorer, Safari... Ne vous êtes-vous jamais demandé comment les navigateurs savent aller au bon endroit ? Comme, par exemple, aller sur le SdZ (Site du Zéro) ?

Pour ceux qui ne le sauraient pas, tout ordinateur actuel possède une adresse sur un réseau : son adresse IP.

C'est grâce à cette adresse qu'un ordinateur, ou un serveur peuvent s'identifier sur un réseau. Voyez ça comme sa carte d'identité.

Par exemple, chez moi, je suis connecté à ma box (fournie par mon FAI) qui me donne accès à Internet.

Sur Internet, cette box a une adresse qui lui est propre et celle-ci ressemble à quelque chose comme ça "242.231.15.123" : on appelle ces adresses des "adresses IP".

Lorsque vous demandez une page web à votre navigateur, vous lui demandez, de façon tacite, d'aller chercher ce qui se trouve à l'adresse demandée !

Partez du principe que toute adresse de site internet pointe vers un serveur (ou plusieurs) qui a une adresse. Par exemple, taper "<http://www.google.fr>" dans votre navigateur revient à saisir "<http://74.125.19.147>" (adresse d'un serveur Google sur Internet) : essayez, vous verrez !

Vous êtes d'accord sur le fait que cette suite de nombres n'est pas des plus faciles à retenir...

Il est bien plus simple de mémoriser *google.fr*.

Je ne m'éterniserai pas sur le sujet, mais sachez qu'il y a une machine qui fait le lien entre les

adresses de serveurs (suite de nombres) et les adresses littérales (google.fr) : les DNS. Voyez ces machines comme de gigantesques annuaires téléphoniques, mais pour les sites internet.

Et qu'est-ce que c'est que le "http://" ?

Si vous relisez bien ce que j'ai dit plus haut, vous devez voir que nous avons vu qu'avec l'URL que vous renseignez, vous spécifiez une machine à interroger, donc des fichiers à lire, il ne nous manque plus que le protocole.

Ici, il s'agit du protocole http.

C'est grâce à ce protocole que le navigateur envoie des "*requêtes*" (nous y reviendrons) aux serveurs que vous sollicitez. Il en existe d'autres comme le FTP, le SMTP...

Inutile de nous appesantir sur le sujet (c'est un tuto de programmation, pas de réseau, non, mais)...

Au final, une URL peut se décomposer comme suit :



Nous avons vu que, lorsque vous saisissez une URL dans votre navigateur, que vous validez cette dernière, votre navigateur envoie une "requête" au serveur concerné afin qu'il nous renvoie une page web.

Tout d'abord, on nomme vulgairement l'échange de données entre votre navigateur et le serveur qui fournit les pages web un échange client / serveur.

Le client représente votre navigateur et le serveur... Enfin, vous avez deviné.

Le moment est venu de vous en apprendre un peu plus.

Voici ce qu'il se passe :

- le client émet une requête `http` vers le serveur ciblé ;
- le serveur reçoit les éléments de la requête ;
- celui-ci les interprète ;
- il renvoie la page demandée en émettant une réponse `http` ;
- le client reçoit la page au format HTML ;
- le client affiche la page.

Nous pouvons résumer ce qu'il se passe avec ce schéma :



(fin de l'introduction de Cyrille Herby)

Revenons sur "le client reçoit la page au format HTML":

Le HTML

HTML est un langage. A l'origine, c'est le seul langage compris par les navigateurs (aujourd'hui, il faut ajouter le JavaScript, nous verrons cela un peu plus loin).

Il est très important d'avoir quelques notions de HTML avant de pouvoir continuer, ici aussi, il existe un excellent cours sur le sujet, toujours sur le site du zéro;

cours sur le XHTML (on considerera que XHTML=HTML) de M@teo21 (créateur du site du zéro)

<http://www.siteduzero.com/tutoriel-3-13666-apprenez-a-creer-votre-site-web.html?variant=1>

Voir aussi le document "HTML et CSS, les bases" écrit par mes soins.

les sites web statiques et dynamiques

Un site web est dit "statique" quand il est uniquement constitué de fichiers textes contenant du HTML. Un site web statique est très largement suffisant si le but du concepteur est uniquement de fournir des informations à ses lecteurs.

En revanche si vous recherchez une certaine interaction entre vos lecteurs et votre site (par exemple, l'utilisateur doit entrer son adresse pour obtenir des informations sur les restaurants de sa région), il vous faudra alors concevoir un site dynamique.

Dans le cas d'un site dynamique, les pages HTML ne sont pas écrites à l'avance, c'est le serveur qui sera chargé de "créer" les pages HTML en réponse à une demande du client. Si vous habitez en Haute-Savoie, après avoir indiqué votre adresse, le serveur créera une page HTML qui comportera uniquement les restaurants Haut-Savoyard. Un autre exemple, encore plus simple : imaginer un site qui, en page d'accueil, vous demande votre prénom (toto). Une fois votre prénom rentré, une deuxième page avec le texte : Bonjour "votre prénom" (Bonjour toto) s'affiche. Ce site, même très simple, est un site dynamique. En effet, la deuxième page n'a pas été conçue par le créateur du site (il ne peut pas connaître votre prénom à l'avance !), mais sera générée par le serveur en temps voulu, plus précisément le serveur créera une page HTML qui affichera "Bonjour toto" (si votre prénom est toto !)

Il existe un certain nombre de langages qui permettent la programmation côté serveur, je n'en citerai que deux :

- PHP (le plus simple à utiliser). Si le sujet vous intéresse je ne saurais trop vous conseiller la lecture du tutoriel sur le site du zéro sur le couple PHP/MySQL, toujours écrit par M@teo21 :

http://www.siteduzero.com/tutoriel-3-197288-introduction-a-php.html#ss_part_1

- Java, ici aussi, le site du zéro vous propose un tutoriel de cysboy :

<http://www.siteduzero.com/tutoriel-3-112219-apprenez-a-creer-des-applications-web-dynamiques-avec-jee.html>

Les sites dynamiques sont donc un vrai progrès et permettent l'interactivité entre l'utilisateur et le site. Mais il existe encore un problème, toute modification de la page (due à une éventuelle manipulation de l'utilisateur), entraînera : l'envoi d'informations du client vers le serveur, la génération d'une nouvelle page par le serveur, l'envoi de cette nouvelle page au client. Tout cela peut prendre du temps et ne va pas favoriser la "fluidité" du site. Heureusement, comme toujours, une solution existe : faire gérer par le navigateur (c'est-à-dire côté client) une grosse partie du travail (sans avoir besoin d'échanger un gros flux de données avec le serveur).

Et Ajax fut....

La solution se nomme Ajax. Ajax n'est pas un langage, mais un acronyme : Asynchronous Javascript and XML.

Avant d'entrer dans les détails, il faut bien comprendre que le développement de site web en Ajax vous permet de vous affranchir en partie des problèmes exposés ci-dessus : le temps de latence dû au grand nombre d'échanges entre le client et le serveur. Grâce à Ajax, une partie du traitement se fera au niveau du client, la communication client-serveur pourra être réduite au strict minimum.

Ajax a permis le développement de RIA (application internet riche) aussi appelée parfois (un peu pompeusement) web 2.0. L'idée est simple : l'utilisateur se trouve devant un site internet, mais il doit avoir l'impression qu'il utilise une application en local (sur son ordinateur). Google doc (<http://www.google.com/google-d-s/intl/fr/tour1.html>) est l'exemple typique de RIA, l'utilisateur a l'impression de se trouver devant une application de bureautique classique (comme Office de Microsoft ou Open Office la suite bureautique libre), mais, en fait, il se trouve devant un site internet !

Comment cela est-il possible ? C'est tout simplement le navigateur, en local, qui fait le plus gros du travail, les échanges client-serveur sont réduits au minimum, Google doc utilise une technologie de type Ajax.

Alors, qu'est-ce qu'Ajax ?

Tous les navigateurs modernes sont capables de "comprendre", en plus du HTML, un autre langage : le JavaScript (attention, le nom est trompeur, le JavaScript n'a pas grand-chose à voir avec le Java). La partie client (ce qui sera visible sur le navigateur) sera donc programmée en JavaScript.

En Ajax, les données sont échangées par l'intermédiaire de fichier XML (ou JSON). XML signifie eXtensible Markup Language (en français : langage extensible de balisage). XML n'est pas un langage de programmation, il n'y a pas de boucle for, de if, de while,.....

Il est presque exclusivement utilisé pour stocker (ou transférer d'un programme à un autre) des données (du texte) de façon structurée.

C'est un langage qui utilise des balises (comme le HTML).

Les balises sont ouvrantes, <balise_ouvrante> ou fermantes, </balise_fermante>. Quand vous

ouvrez une balise, vous devez à un moment ou un autre la refermer.

Il est possible de mettre du texte entre la balise ouvrante et la balise fermante :

```
<balise1> Voici le texte inséré </balise1>
```

Les balises peuvent être imbriquées : on peut insérer un ou plusieurs couples de balises (ouvrante et fermante) entre 2 balises (ouvrante+fermante)

voici un exemple permettant de stocker des informations sur votre cinémathèque :

```
<cinematheque>
  <film>
    <nom>Les deux tours</nom>
    <realisateur>P Jackson</realisateur>
    <annee_sortie>2002</annee_sortie>
  </film>
  <film>
    <nom>Bladerunner</nom>
    <realisateur>R Scott</realisateur>
    <annee_sortie>1982</annee_sortie>
  </film>
</cinematheque>
```

la balise <cinematheque> est appelée "élément racine", la balise <film> est le "premier enfant", etc. Enfin, l'essentiel est de retenir qu'XML va favoriser les échanges de données. Comme d'habitude, voici un tuto sur le site du zéro qui vous permettra d'approfondir la question :

<http://www.siteduzero.com/tutoriel-3-33440-le-point-sur-xml.html>

Dernière partie de l'acronyme Ajax, "Asynchronous" : l'idée est simple, dans un échange synchrone entre le client et le serveur, quand le client (donc le navigateur web) effectue une requête, le client attend la "réponse" du serveur avant de faire quoi que ce soit d'autre (ce qui peut donner l'impression que le navigateur est "bloqué"). Dans un échange asynchrone, le client n'attend pas la réponse du serveur pour poursuivre les tâches en cours, ce qui évite donc le blocage du processus (si la réponse n'arrive jamais !). En Ajax, tous les échanges sont asynchrones, d'où le "Asynchronous".