

APPLICATIONS WEB : INTRODUCTION

et aux requêtes HTTP

Contenu

Serveur HTTP.....	1
Communication avec un serveur Web	1
Site Web	2
Site web statique	2
Site web dynamique.....	2
Les requêtes HTTP	3
Les paramètres de requête HTTP.....	4
GET.....	5
POST.....	5
Synthèse des différences entre GET et POST	5

Serveur HTTP

Un serveur HTTP (ou serveur Web) est un serveur multi-services qui délivre du contenu via le protocole HTTP.



HTTP : HyperText transfert Protocol est un protocole de communication client-serveur développé pour le Web. Sa variante sécurisée est le HTTPS (avec « S » pour « secured »).

Par défaut, un serveur Web « écoute » sur le port 80 (port 443 pour le HTTPS). Il est cependant possible de modifier cette valeur afin que le serveur écoute sur un port différent voire même sur plusieurs ports en même temps. Un serveur web diffuse la plupart du temps des « sites web » et est généralement associé à d'autres services tels qu'un serveur de bases de données, un serveur FTP, un serveur de streaming etc...

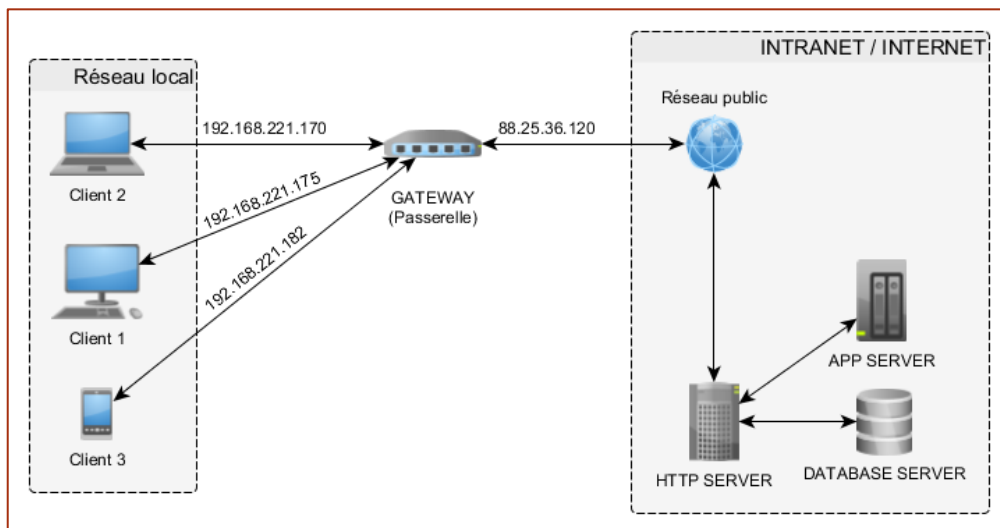
Communication avec un serveur Web

Typiquement, un client demande l'affichage d'une page web au travers de son navigateur web.

La requête est transmise à la passerelle de votre réseau local (votre box internet ou un routeur de l'entreprise) qui la transfère vers internet à destination du serveur web correspondant au nom de domaine/adresse IP.

Le serveur réceptionne la requête, la traite et renvoie la réponse au client qui en a fait la demande.

Le navigateur récupère la réponse et effectue le traitement approprié (typiquement : affichage de contenu HTML, téléchargement d'un fichier...).



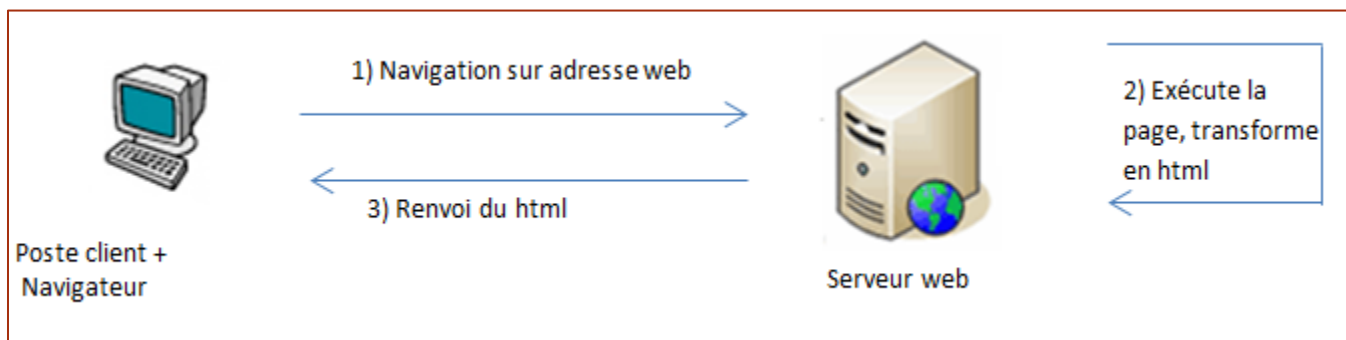
Site Web

Un site web est un ensemble de fichiers, hébergés sur un serveur et accessibles via un navigateur (Chrome, Firefox, Edge, et pour les plus téméraires... Internet Explorer).

Ces fichiers peuvent être de simples fichiers HTML statiques ou des fichiers exécutables interprétés par un moteur de rendu tel que PHP (Hypertext Preprocessor), JSP (Java Server Pages) ou ASP.NET (Active Server Pages). Un fichier accessible via un serveur web est ce qu'on appelle une « ressource web ».

Un serveur Web est un ordinateur (en général assez puissant) qui héberge des fichiers qui sont accessibles via une adresse internet (exemple : www.arfp.asso.fr).

Le schéma suivant démontre l'échange entre un navigateur web et un serveur web.



Site web statique

Un site web statique est un contenu codé avec le langage HTML et mis à disposition « tel quel » sur le serveur web. Un site web statique ne permet pas de réaliser des opérations aujourd'hui communes : connexion à une base de données, identification d'un utilisateur, rédaction de commentaires etc...

Pour réaliser ce type d'opération, il est nécessaire de développer un « site dynamique ».

Site web dynamique

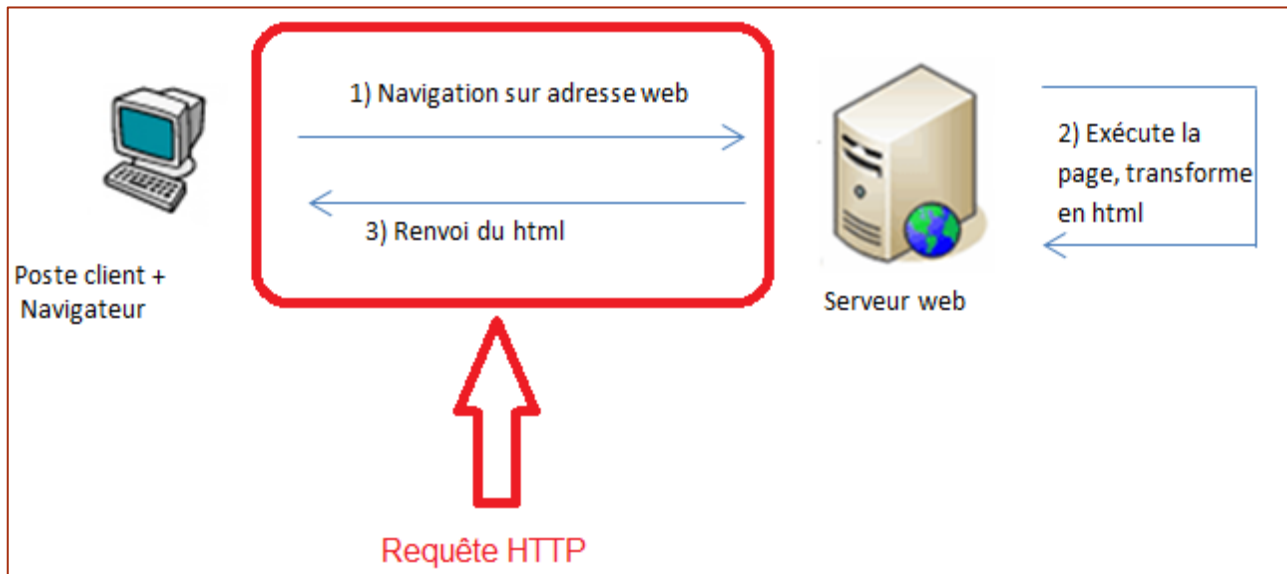
Mis à part quelques antiquités du web ou quelques sites de présentation, les sites web « statiques » se rarifient. Les sites dynamiques sont capables d'exécuter des programmes qui vont être en mesure d'afficher des informations différentes en fonction du besoin. C'est le rôle premier du serveur web, il exécute la page dynamique (qui sera codée en PHP, Java ou C#) à partir de laquelle sera généré du code HTML afin qu'elle soit affichable par le navigateur web.



Le terme « application web » est l'appellation moderne pour un site web dynamique.

Les requêtes HTTP

L'accès à un site web par un navigateur se fait au travers de requêtes HTTP.



Un serveur web a la possibilité de gérer plusieurs types de requêtes HTTP dont les plus importantes sont : GET, POST, PATCH, PUT, DELETE. Ces termes sont appelés des « verbes HTTP ».

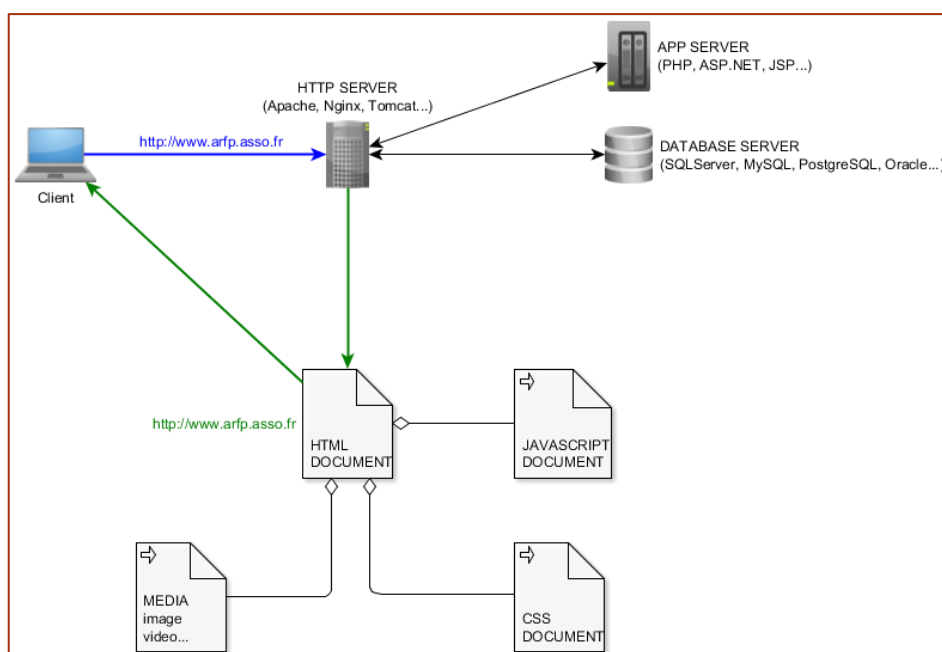
Nous nous intéresserons particulièrement à GET et POST. Les verbes HTTP « PATCH, PUT, DELETE » servent généralement aux requêtes effectuées auprès de services Web et ne seront pas traités dans ce document.



Un service web (ou Web Service) est un protocole d'échange de données entre applications et/ou environnements distribués.

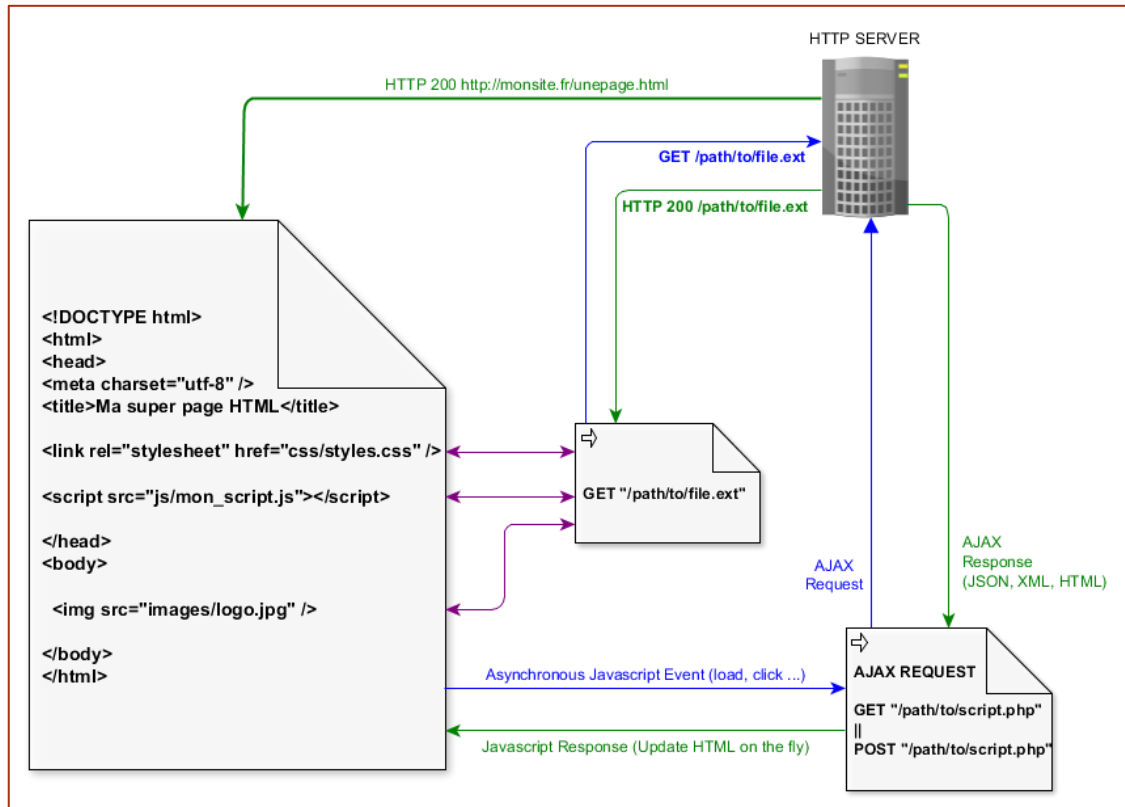
Le contenu est affiché grâce au code HTML récupéré par le navigateur.

Le code HTML peut contenir des appels vers d'autres fichiers tels que des fichiers CSS, Javascript, des images, vidéos et tout autre contenu potentiellement affichable dans un navigateur.



Chaque appel vers un fichier génère une requête HTTP. Le code HTML présenté dans l'exemple ci-dessous, va générer pas moins de 4 requêtes HTTP :

- 1) La page HTML en elle-même
- 2) Le fichier CSS « css/styles.css »
- 3) Le fichier Javascript « js/mon_script.js »
- 4) L'image « images/logo.jpg »



De plus, des événements Javascript (un clic sur un élément, un survol de souris sur une image...) peuvent générer des requêtes HTTP supplémentaires.

Les paramètres de requête HTTP

Le format des paramètres GET et POST est sensiblement identique, les données sont transmises sous forme de chaîne de caractères ou chaque paire « clé=valeur » est séparée par le caractère « & ».

Cette chaîne de caractères est communément appelée « QueryString ».

Exemples :

QueryString	Nombre Paramètres	Paramètre(s)	Valeur(s)
« page=articles »	1	page	articles
« page=products&id=1 »	2	page	products
		id	1
« page=users&id=2&action=edit »	3	page	users
		id	2
		action	edit

La différence entre GET et POST réside dans la manière dont les données seront transmises.

GET

Une requête de type « GET » correspond à une requête en lecture seule d'une ressource web. C'est la requête HTTP par défaut et donc celle qui est le plus souvent utilisée.



Une requête « GET » permet de consulter un contenu à l'infini et ne devrait pas permettre de le modifier.

Les données d'une requête de type GET sont transmises directement dans l'url, ce qui les expose à la vue de l'utilisateur.

Les paramètres s'ajoutent à la fin de l'url en utilisant un séparateur entre l'url et ses paramètres. Ce séparateur est le point d'interrogation « ? ». Chaque paramètre d'url est une paire « clé=valeur ». Il est possible de définir plusieurs paramètres d'url en les séparant par le caractère « & ».

Exemples :

- <http://localhost/index.php> est un url non paramétré.
- <http://localhost/index.php?id=1> est un url paramétré contenant un paramètre nommé « id » et ayant pour valeur « 1 ».
- <http://localhost/index.php?id=1&page=articles> est un url comprenant 2 paramètres :
 - o « id » ayant pour valeur « 1 »
 - o « page » ayant pour valeur « articles »

Il est possible de paramétrer « manuellement » une requête de type GET en construisant l'url et en respectant le format du queryString.

POST

Une requête de type « POST » est généralement utilisée pour soumettre des données au serveur web notamment via des formulaires HTML.



Une requête « POST » permet d'insérer ou de modifier un contenu et ne devrait pas pouvoir être exécutée à l'infini.

Les données d'une requête de type POST n'apparaissent pas dans l'url et sont envoyées après les entêtes HTTP.

Synthèse des différences entre GET et POST

	GET	POST
Longueur maximale	2048 octets (paramétrable au niveau du serveur)	∞
Mise en cache dans le navigateur	✓	✗
Rétention dans l'historique du navigateur	✓	✗
Peut être mis en favori	✓	✗
Lire une ressource	✓	✗
Mettre à jour une ressource	✗	✓
Transmettre des données sensibles	✗	✓

--- FIN DU DOCUMENT ---

cnam-grandest.fr