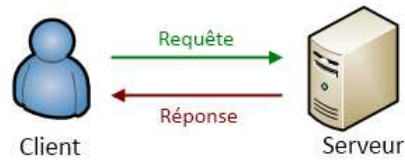
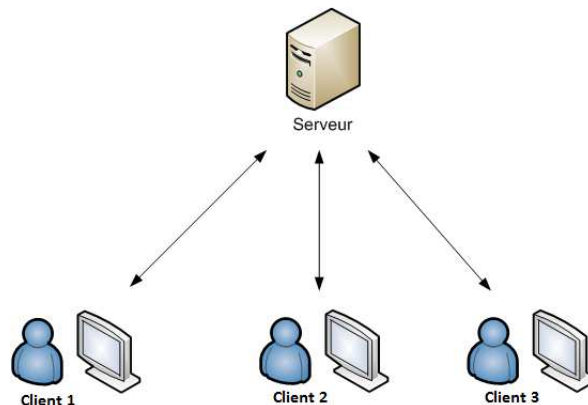


1 Le modèle client-serveur

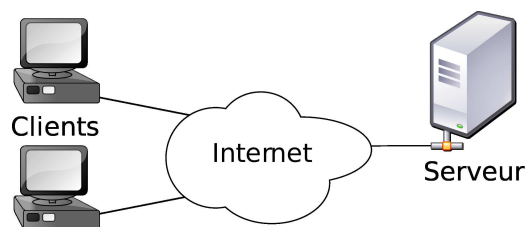
Dans une architecture d'interaction client-serveur, le serveur est une machine qui possède des ressources et qui renvoie ces ressources aux clients à la suite de requêtes de ces derniers.



Un même serveur peut répondre aux requêtes de plusieurs clients :



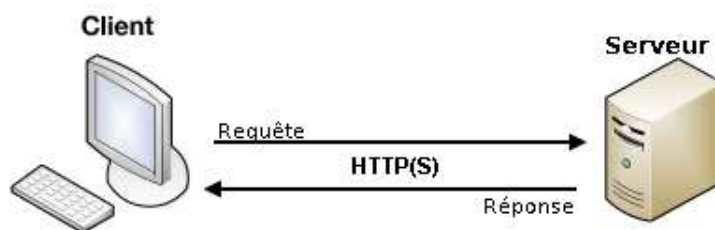
Et cette communication se fait bien souvent à travers le réseau Internet (par exemple lorsqu'on interroge un serveur Web) :



On peut citer par exemple des serveurs de pages Web, de mail, de fichiers...

2 Exemple du serveur Web : protocole HTTP

Le protocole **HTTP** (HyperText Transfert Protocol) est un protocole de la couche Application (cf cours Réseaux) mis en œuvre lorsqu'une requête est effectuée à la demande d'un logiciel navigateur (Firefox, Edge, Safari...).



2.1 Structure d'une requête HTTP

Une requête HTTP présente le format suivant :

```
Ligne de commande (Commande, URL, Version de protocole)
En-tête de requête
[Ligne vide]
Corps de requête
```

Exemple de requête (avec un corps vide, fréquent) :

```
GET serveur/dossier/fichier.html HTTP/1.1
User-Agent : Mozilla/5.0
Accept : text/html
```

Remarque : il existe plusieurs champs facultatifs dans l'en-tête de la requête (et ceci dépend aussi de la version du protocole).

Il est possible de transmettre des informations à un site à travers l'en-tête de la requête ou dans le corps de la requête (voir cours formulaire).

Les scripts Javascript en revanche sont exécutés au niveau du client, sans interaction avec le serveur.

2.2 Structure d'une réponse HTTP

Les réponses HTTP présentent le format suivant :

```
Ligne de statut (Version, Code-réponse, Texte-réponse)
En-tête de réponse
[Ligne vide]
Corps de réponse
```

Exemple de Réponse :

```
HTTP/1.1 200 OK
Date: Thu, 15 feb 2019 12:02:32 GMT
Server: Apache/2.0.54 (Debian GNU/Linux) DAV/2 SVN/1.1.4
Connection: close
Transfer-Encoding: chunked
Content-Type: text/html; charset=ISO-8859-1
```

```
<!DOCTYPE html>
<html lang="fr">
<head>
<meta charset="utf-8">
<title>Voici mon site</title>
</head>
<body>
```

```
<h1>Hello World! Ceci est un titre</h1>
<p>Ceci est un <strong>paragraphe</strong>. Avez-vous bien compris ?</p>
</body>
</html>
```

Dans cet exemple, le code réponse 200 (OK) indique que le document recherché par le client a bien été trouvé par le serveur. La page Web va alors s'afficher correctement sur le client.

On pourra retenir 2 autres codes de réponses assez classiques :

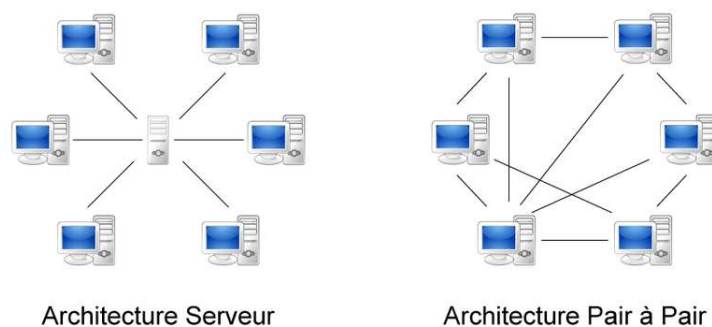
- 404 : page non trouvée (c'est souvent parce que l'URL a été mal écrite dans la barre d'adresse du navigateur, ou parce que le fichier a été déplacé ou retiré sur le serveur).
- 403 : accès refusé : le serveur a compris la requête mais les droits d'accès ne permettent pas au client d'accéder à la ressource.

3 Architecture pair à pair (peer to peer P2P)

Dans ce modèle d'interaction entre machines, chaque ordinateur (parfois nommé « serveur » contraction de serveur-client) joue en même temps le rôle de serveur et de client. Les ressources sont partagées sur tous ordinateurs présents sur le réseau.

Chaque machine peut donc soit effectuer une requête pour récupérer des ressources, soit fournir des ressources aux autres.

L'image suivante permet de distinguer l'architecture client-serveur et pair à pair :



Un réseau P2P n'est pas forcément illégal : il le devient si les ressources partagées sont protégées par la propriété intellectuelle.

Certains projets scientifiques nécessitant de grandes capacités de calculs s'appuient sur les réseaux P2P pour partager les calculs sur différentes machines.