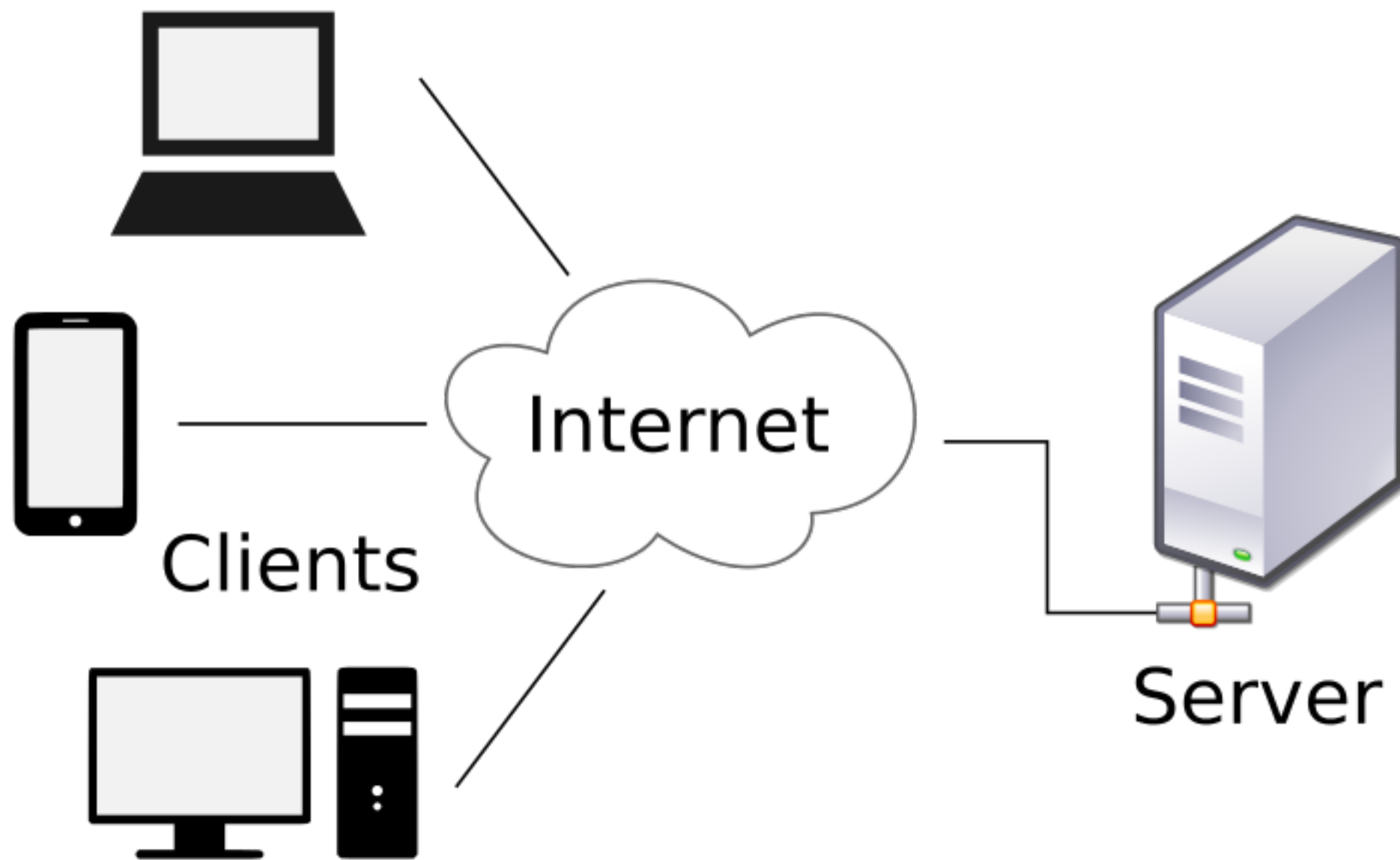


Applications Javascript

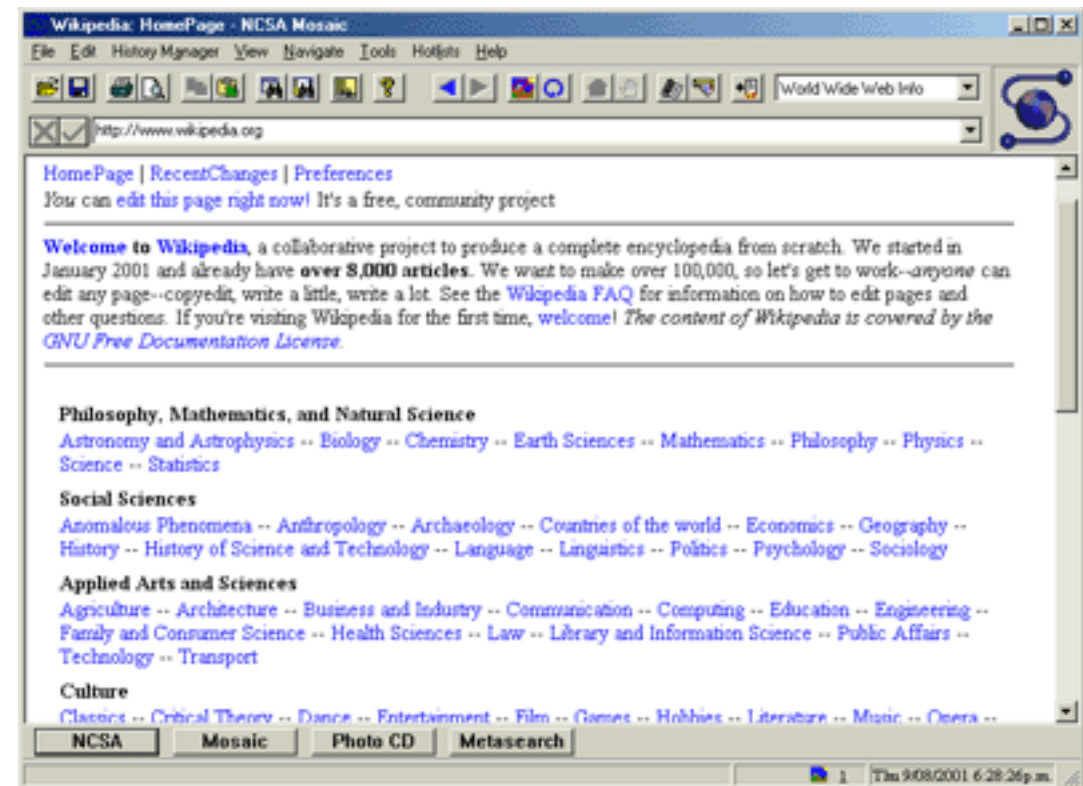
Jean-Marc Lecarpentier
GREYC - Université de Caen Normandie

Architecture client-serveur



Site web statique

- Page web statique = page servie telle quelle par le serveur
- 1 page web = 1 fichier HTML
- Modifier site $\Rightarrow$ modifier fichier
- Avoir accès au serveur
- De plus en plus fastidieux avec la croissance d'un site



Page web dynamique

- URL donnée
- Page différente selon les vues
- Évolutions en fonction de divers critères
 - Temps
 - Utilisateurs
 - Localisation
 - etc

Site web dynamique

- Pages créées par des programmes
- Sources de données diverses
- Stockage de l'information
- Langages de programmation



Programmation côté serveur

- Échanges d'informations avec le client : HTTP
- Méthode GET vs. POST
- URL rewriting
- Démonos

```
*  
* @var boolean  
*/  
define('PSI_INTERNAL_XML', false);  
  
if (version_compare("5.2", PHP_VERSION)  
    die("PHP 5.2 or greater is required");  
}  
if (!extension_loaded("pcre")) {  
    die("phpSysInfo requires the pcre  
        properly.");  
}  
  
require_once APP_ROOT.'/includes/  
  
// Load configuration  
require_once APP_ROOT.'/config.p  
  
if (!defined('PSI_CONFIG_FILE'))  
    $tpl = new Template("/templ  
    echo $tpl->fetch();  
    die();
```

Programmation côté serveur

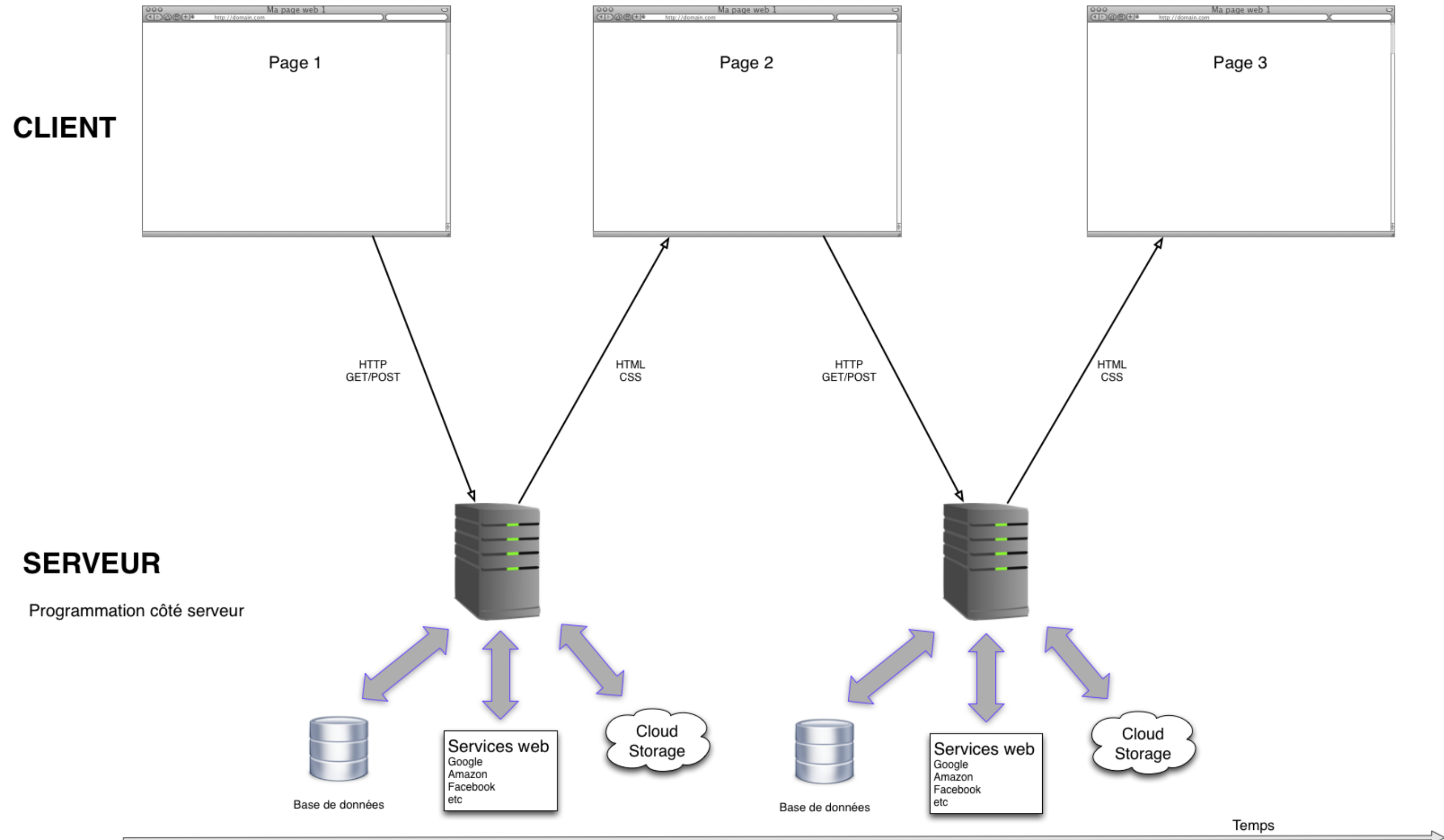
- Exemple avec PHP

```
<?php
if (isset($_GET['name'])) {
    $name = $_GET['name'];
} else {
    $name = "Bertrand";
}
?>
<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <title>Une page dynamique</title>
</head>
<body>

<h1>Bonjour <?php echo $name; ?></h1>

<p>Page créée le <?php echo date("d M Y"); ?></p>
</body>
</html>
```

Programmation côté serveur



Stockage des données

- Fichiers sur serveur
- Problèmes :
 - Redondance des données
 - Accès concurrentiels
 - Écriture des données
 - Performances



Stockage des données

- Bases de données
- Relations entre données
- Langage SQL
- Démo

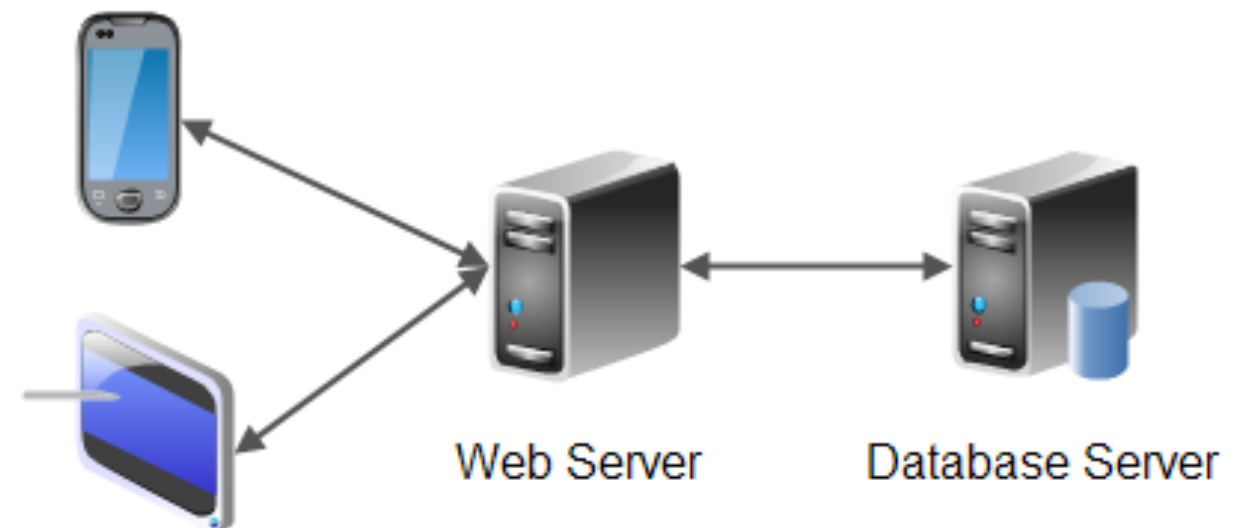
```
Welcome to MySQL Terminal. Commands end with ; or \g.
Script version: 1.0.0
Please type mysql -h [host] -u [user] -p [password] and press "Enter" to start.

> mysql -u root -p
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 884
Server version: 5.4.3-beta-community

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
> use world
Database changed
> select * from city limit 10
+-----+-----+-----+-----+-----+
| ID | Name          | CountryCode | District | Population |
+-----+-----+-----+-----+-----+
| 1  | Kabul         | AFG         | Kabul    | 1780000    |
| 2  | Qandahar      | AFG         | Qandahar | 237500     |
| 3  | Herat         | AFG         | Herat    | 186800     |
| 4  | Mazar-e-Sharif | AFG         | Balkh    | 127800     |
| 5  | Amsterdam     | NLD         | Noord-Holland | 731200    |
| 6  | Rotterdam     | NLD         | Zuid-Holland | 593321    |
| 7  | Haag          | NLD         | Zuid-Holland | 440900    |
| 8  | Utrecht       | NLD         | Utrecht  | 234323     |
| 9  | Eindhoven     | NLD         | Noord-Brabant | 201843    |
| 10 | Tilburg       | NLD         | Noord-Brabant | 193238    |
+-----+-----+-----+-----+-----+
10 rows in set (0.0010840892791748 sec)
>
```

Architecture 3 tiers

- Client
- Serveur web
- Serveur Base de données
- Serveur web doit se connecter au serveur de BD pour accéder aux données



Architecture orientée services

- Serveur web ne peut pas “tout” savoir
- Serveur web ne peut pas tout stocker
- Exemple : météo, cours de la bourse, résultats sportifs, etc
- Nécessité de sources externes de données
- Communications entre machines



Exemples

- APIs & web services
- geonames données géographiques
- amazon accès au catalogue
- Google books, maps, etc
- Images Flickr



Format des données

- XML eXtended Markup Language
- Syntaxe proche du HTML
- Possibilités bien plus larges
- Architecture lourdes en général

```
<Books>
  <Book ISBN="0553212419">
    <title>Sherlock Holmes: Complete Novels...
    <author>Sir Arthur Conan Doyle</author>
  </Book>
  <Book ISBN="0743273567">
    <title>The Great Gatsby</title>
    <author>F. Scott Fitzgerald</author>
  </Book>
  <Book ISBN="0684826976">
    <title>Undaunted Courage</title>
    <author>Stephen E. Ambrose</author>
  </Book>
  <Book ISBN="0743203178">
    <title>Nothing Like It In the World</title>
    <author>Stephen E. Ambrose</author>
  </Book>
</Books>
```

Format des données

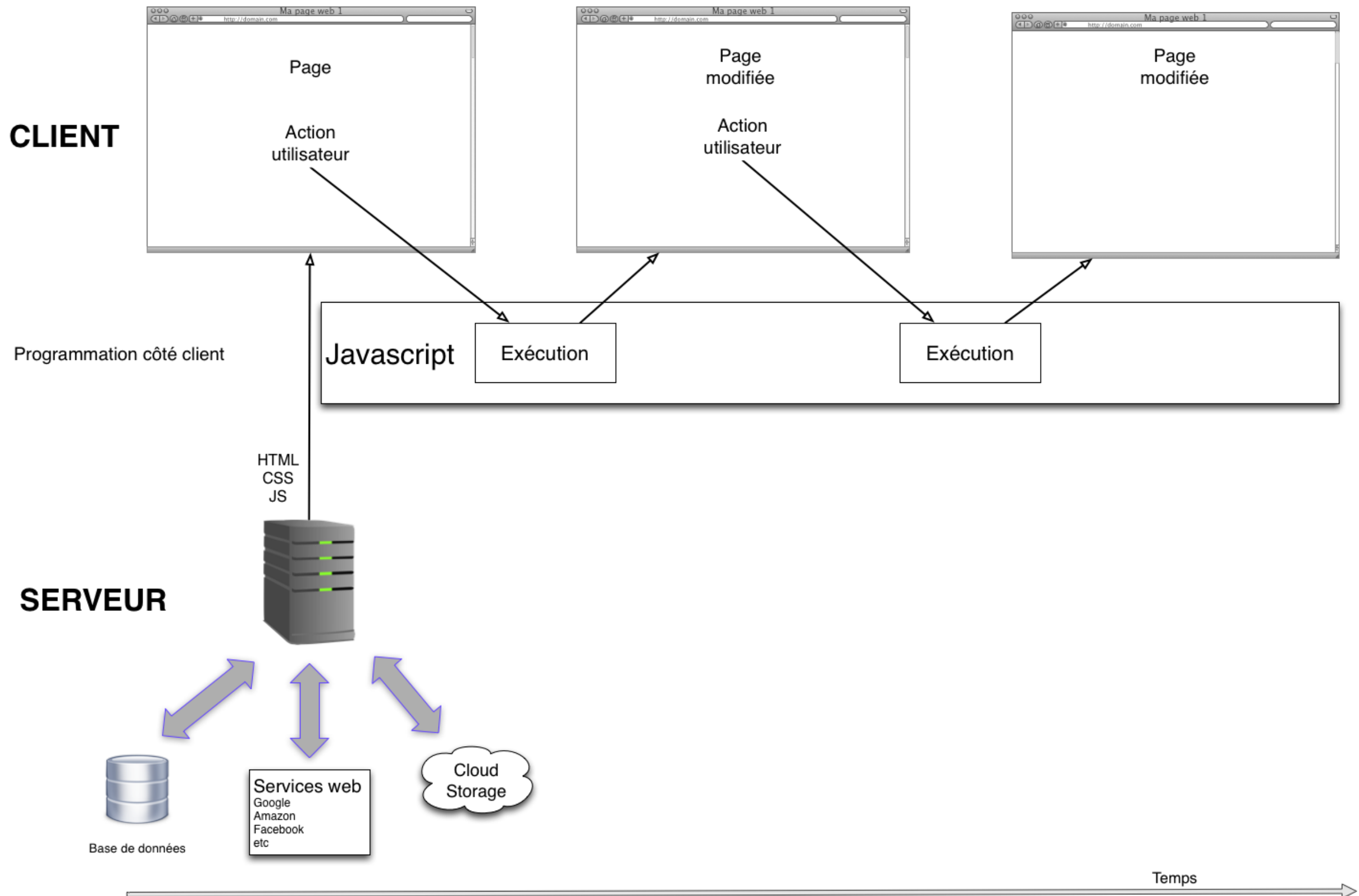
- JSON JavaScript Object Notation
- Applications légères
- Architecture REST
- Services web interrogés via une simple URL plus des données GET ou POST
- Préféré des développeurs web
- Interaction avec Javascript

```
{
  "Rail Booking": {
    "reservation": {
      "ref_no": 1234567,
      "time_stamp": "2016-06-24T14:26:59.125",
      "confirmed": true
    },
    "train": {
      "date": "07/04/2016",
      "time": "09:30",
      "from": "New York",
      "to": "Chicago",
      "seat": "57B"
    },
    "passenger": {
      "name": "John Smith"
    },
    "price": 1234.25,
    "comments": ["Lunch & dinner incl.", "\"Have a nice day!\""]
  }
}
```

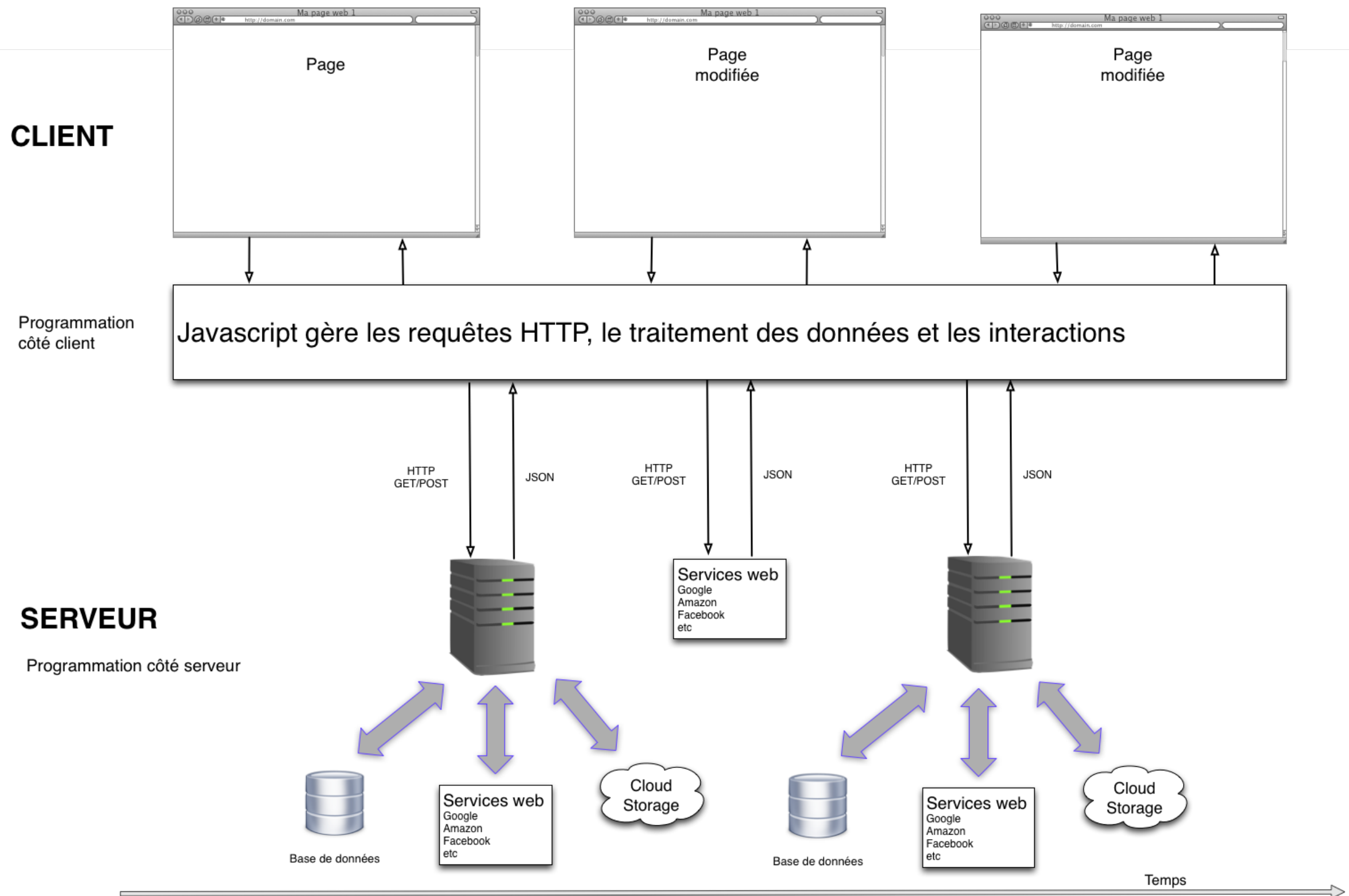
Programmation côté client

- Application fonctionnant sur le client
- Programmes chargés par la « page web »
- Notion de logiciel plus que de document
- HTML5 + CSS + JS
- **Limitations** : pas d'accès aux données stockées dans le Système d'Information (base de donnée, fichiers sur le serveur, etc)

Programmation côté client



Application web



Front-end et Back-end

- Front-end
 - Programmation côté client (le plus souvent en Javascript)
 - Javascript effectue les requêtes HTTP en arrière-plan (requêtes XHR ou AJAX)
 - Navigateur reçoit les données JSON et les traite
- Back-end
 - Programmation côté serveur
 - Application client plus légère mais charge du serveur plus lourde
 - Navigateur reçoit HTML/CSS

Conclusion

- Évolutions des techniques
- Évolutions des usages
- Données non centralisées
- Mashup d'informations
- Programmation d'applications web = agrégation de nombreux enseignements informatique vus en Licence