

# Programmation Web – Avancé

JavaScript & Node.js

Partie 14 : Communications client / serveur

Version 2020



Attribution –  
Partage dans les  
Mêmes Conditions  
4.0 International  
(CC BY-SA 4.0)

*Presentation template  
by [SlidesCarnival](#)*

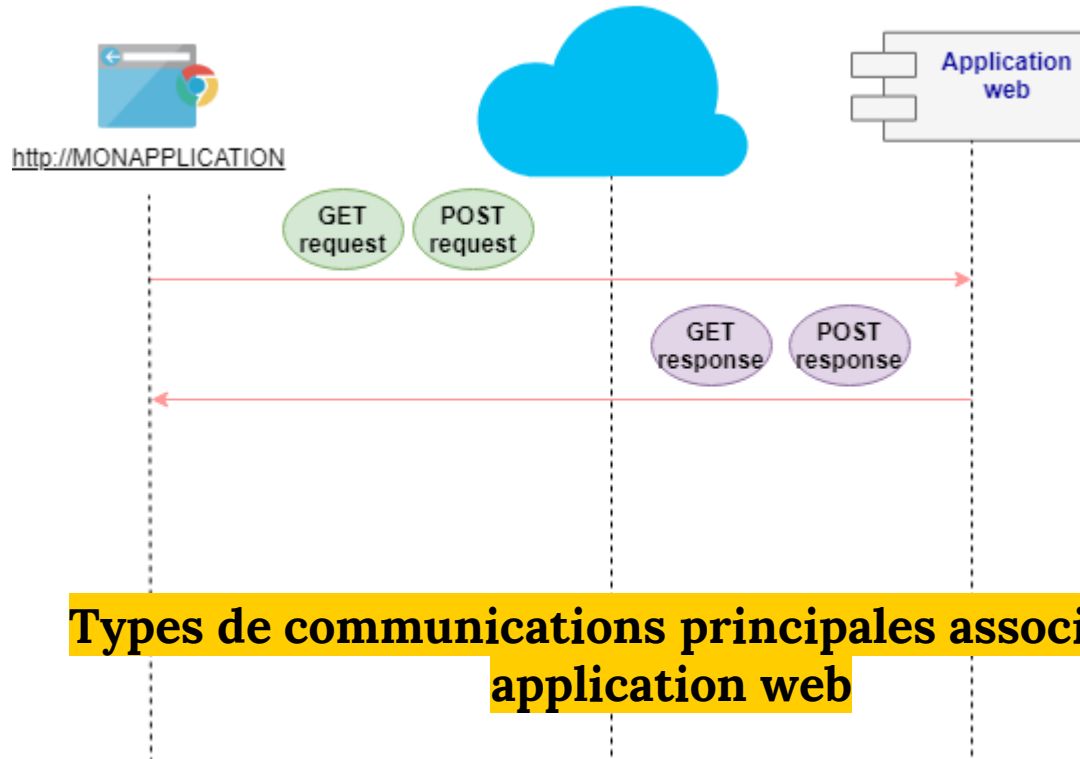


# Introduction aux communications client / serveur

Qu'est-ce qu'il me faut pour communiquer au sein d'une application web ?



## Quels types de communications ?

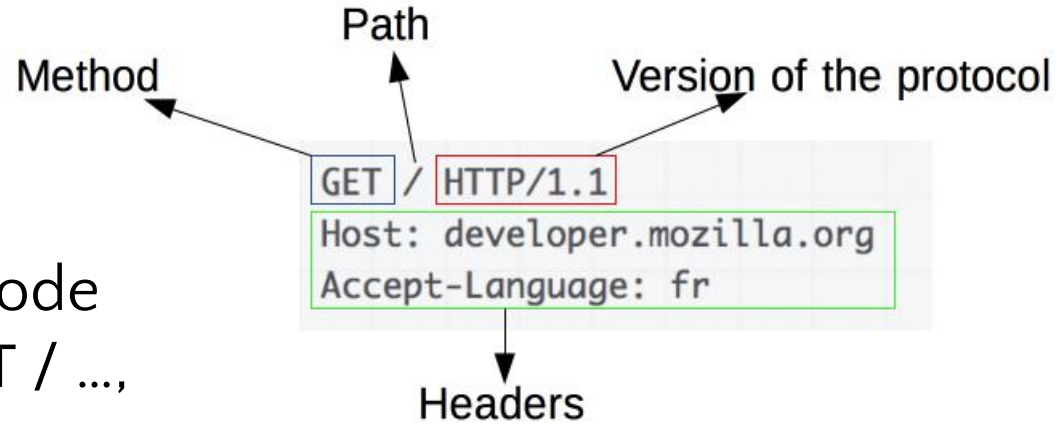


- Communications synchrones : MPA
- Communications asynchrones : SPA
- Données : HTML, JS, Images, XML, JSON...



## Requêtes HTTP

- Start line (Méthode HTTP GET / PUT / ..., URL)
- Headers (host, cookies...)
- Body (HTML form parameters, JSON...)



**Basics of HTTP [53.]**



## Réponses HTTP

- Status line (status code & text...)
- Header (content type, Location, cookies...)
- Body (ressource demandée, message d'erreur, JSON...)

Status code  
Version of the protocol      Status message

HTTP/1.1 200 OK

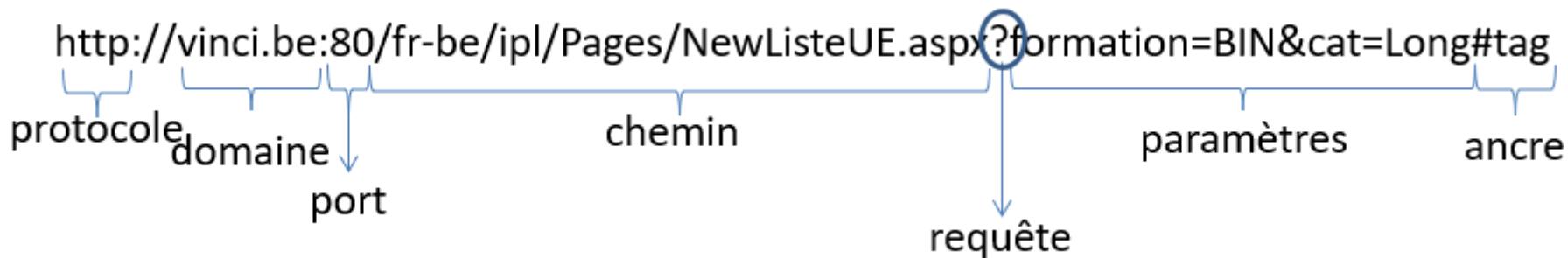
```
Date: Sat, 09 Oct 2010 14:28:02 GMT
Server: Apache
Last-Modified: Tue, 01 Dec 2009 20:18:22 GMT
ETag: "51142bc1-7449-479b075b2891b"
Accept-Ranges: bytes
Content-Length: 29769
Content-Type: text/html
```

Headers

**Basics of HTTP [53.]**



## GET : obtention d'une ressource



- Paramètres dans l'URL
- Bookmarkable
- Mise en cache possible



## **POST : soumission d'une ressource**

---

- Soumission des données d'un formulaire
- Paramètres dans le body de la requête
- Non bookmarkable
- Généralement non mis en cache



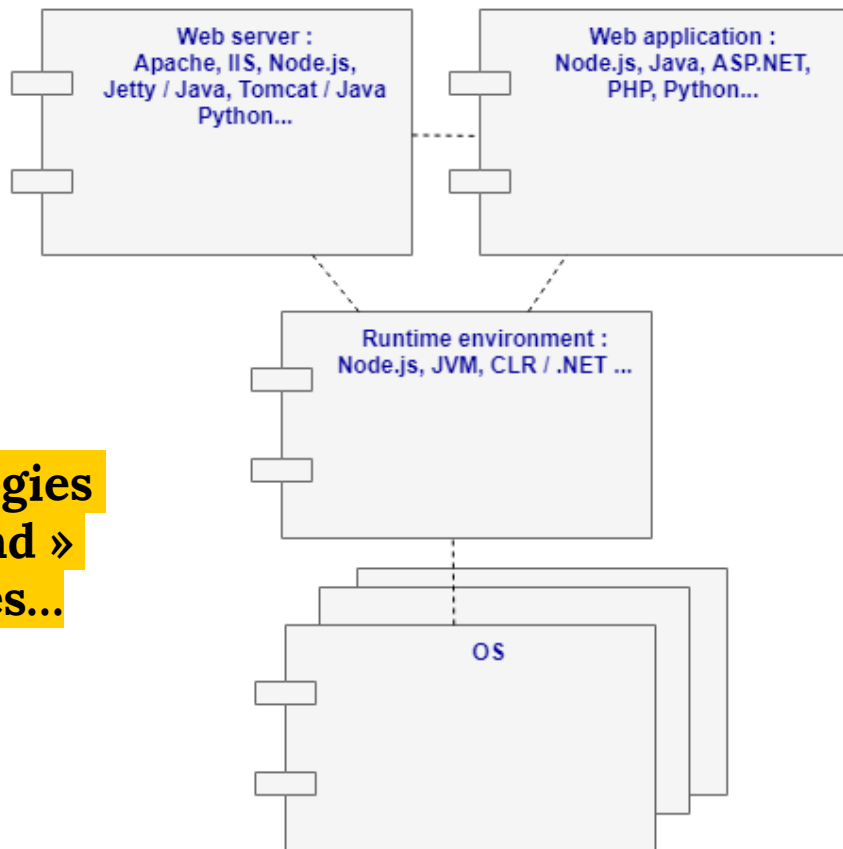
## CRUD Operations via HTTP

---

- POST
- GET
- PUT & PATCH
- DELETE



Server



**Technologies  
« backend »  
possibles...**

**Technologies  
« backend »  
choisies pour  
ce cours...**

