

LOG4420 - Introduction

Nikolay Radoev
École Polytechnique de Montréal



LOG4420

Conception de sites web dynamiques et transactionnels

- Le cours vise à donner une vue d'ensemble des technologies utilisées en développement de sites web.
- Pas de connaissances préalables en développement web nécessaires, mais celles-ci peuvent être pratiques.
- Le cours a une grande composante appliquée : une partie importante du travail sera fait dans les travaux pratiques.

Déroulement du cours

- Le cours se fait en classe inversée :
 - Les notes de cours et des lectures (souvent sous la forme de sites de références ou de vidéos) sont à lire avant le cours.
 - Les séances de cours seront dédiées à un court résumé de la matière suivi d'une période de réponses aux questions et des exercices.
 - Consultez le matériel et avancez à votre rythme. Si vous avez des questions ou des éléments sur lesquels vous voulez qu'on revienne, mettez un message sur Discord et on en discutera lors de la séance suivante.

Matière du cours

- Semaine 1: Introduction à l'architecture du Web
 - Aperçu global du fonctionnement d'un navigateur
 - Architecture haut niveau du Web
- Semaine 2: Introduction à HTML5
- Semaine 3 : Introduction à CSS3
- Semaine 4 : Introduction à JavaScript

Matière du cours

- Semaine 5: Javascript (suite)
 - Interactions avec le DOM
 - Introduction à jQuery
- Semaine 6: Protocole HTTP
- Semaine 7 : Contrôle périodique
- Semaine 8 : Applications côté serveur

Matière du cours

- Semaine 9 : Architecture REST
- Semaine 10: Persistence de données avec NoSQL
- Semaine 11 : Applications côté client
 - Introduction à React
- Semaine 12 : Applications côté client (suite)
- Semaine 13 : Sujet spéciaux
 - Introduction au Web Sémantique

Évaluations individuelles

- Contrôle périodique
 - 25% de la note globale
 - 22 octobre pendant les heures de cours
 - À distance, sur MoodleQuiz
- Examen final
 - 35% de la note globale
 - Pendant la période d'examens finaux en décembre
 - En présentiel à Polytechnique
 - En cas de reconfinement, à distance sur MoodleQuiz
 - En cas d'absence motivée, une évaluation alternative aura lieu

Travaux pratiques

- 5 travaux pratiques
 - 40% de la note globale
 - Voir le calendrier sur Moodle pour les dates de remise
 - Équipes de 2 personnes de la même section
- Matière couverte dans les travaux pratiques
 - TP1 : HTML
 - TP2 : CSS
 - TP3 : JavaScript, DOM et AJAX
 - TP4 : Développement côté serveur et base de données
 - TP5 : Décomposition d'application clientes (React)

Questions ?

Plan de la séance

- Historique rapide du Web
- Survol de l'architecture de l'Internet et le Web
- Navigateurs web
 - Fonctionnement
 - Comment on affiche une page web

Historique (très rapide) du Web

- 1960-1969 : début des travaux pour une communication entre machines.
- 29 octobre 1969 : première communication entre 2 machines sur le même réseau (ARPANet)
 - UCLA à Stanford. Le message LOGIN crash après l'envoi des 2 premières lettres...
- 1973 : ARPANet adopte le protocole FTP pour le transfert de fichiers.
- 1983: ARPANet adopte le protocole TCP/IP pour ses messages
- 1989 : NSFNET des ÉU est connecté avec CERNENT en Europe

Historique (très rapide) du Web

- 1990 : création du World Wide Web par Tim Berners-Lee à CERN.
 - Basé sur un mélange des concepts du HyperText et Internet
 - Introduction des concepts de HTML, HTTP et URI
- 20 décembre 1990: premier site web publié et accessible sur le premier navigateur (WorldWideWeb)
 - <http://info.cern.ch/hypertext/WWW/TheProject.html>
 - Un NeXT Computer à CERN est le seul serveur Web jusqu'à 1991
- 1993: création du navigateur graphique Mosaic qui rend le Web accessible à la population générale.
 - 1992 : moins de 30 sites web au monde, 1995 : plus de 10 000
 - Ancêtre d'Internet Explorer

Historique (très rapide) du Web

- 1994: création du navigateur Netscape Navigator.
 - Premier navigateur à intégrer JavaScript (1995)
 - Ancêtre du Mozilla Project et le Gecko engine
- 1995: Internet Explorer et les Browser Wars
 - IE domine le marché pendant près de 10 ans
 - 1996: CSS 1 proposé, mais pas complètement adopté avant 2000 avec IE5
- 1997 : ECMA-262 standardise le langage ECMAScript
 - JavaScript est un langage qui se conforme à ECMA-262
 - ES2 en 1998, ES3 en 1999, ES4 est abandonné et ES5 est publié seulement en 2009

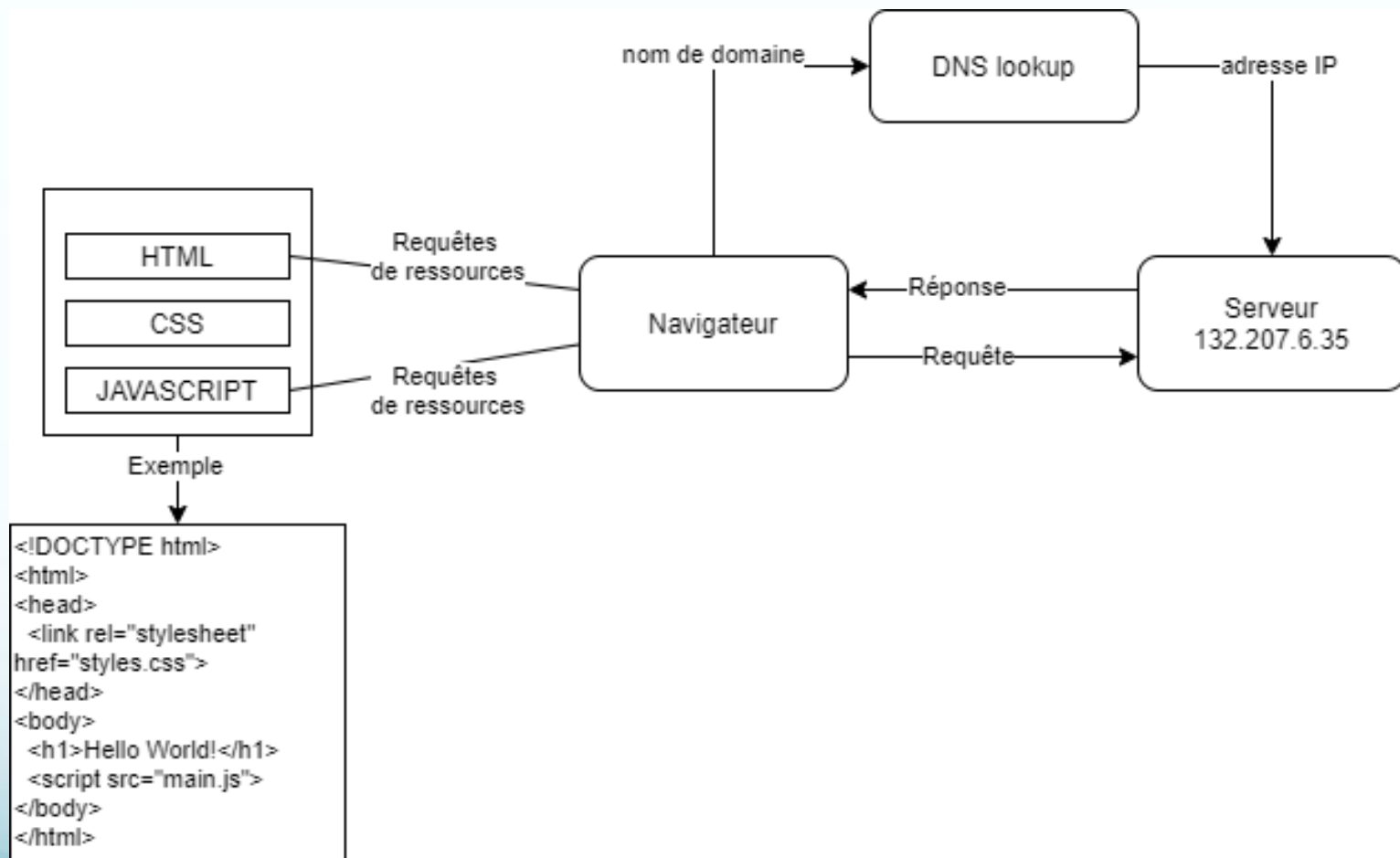
Historique (très rapide) du Web

- 2000 : *dot-com bubble*
 - Mélange de contexte économique particulier et un manque de compréhension générale de comment utiliser le web
 - Force beaucoup de compagnies et de personnes à revoir leur manière d'intéragir avec le web. Ex: Google et Amazon
- 2002-2010: Web 2.0
 - Le Web devient de plus en plus interactif : les utilisateurs deviennent aussi des créateurs (ex: YouTube 2005)
 - Création des réseaux sociaux : Myspace 2003, Facebook : 2004)
 - 2^{ème} Browser Wars : Chrome, Firefox, Safari et IE
- 2010-Aujourd'hui : le Web en tant qu'outil global
 - Le Web est un des outils informatiques les plus utilisés
 - Croissance et maturité des outils de développement

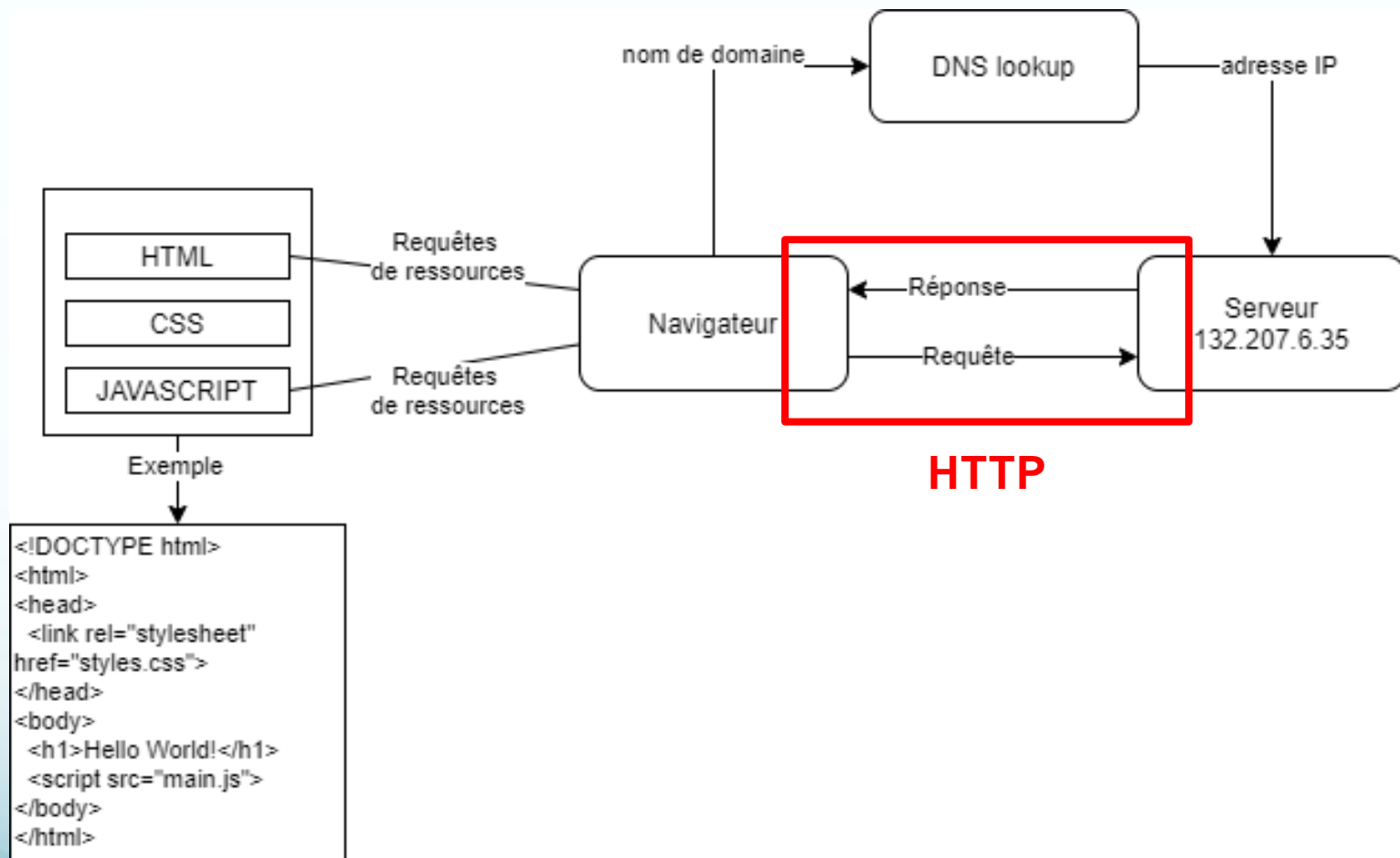
Architecture du Web (haut niveau)

- Notre navigateur communique avec un serveur à travers des requêtes (souvent utilisant HTTP)
- Le serveur répond en fournissant des fichiers avec leur type et un code spécifique
- Le navigateur gère ces réponses en fonction du type du fichier
 - HTML pour le contenu d'une page web
 - CSS pour le style
 - JavaScript pour l'interaction
 - Autres (images, audio, JSON, etc.)

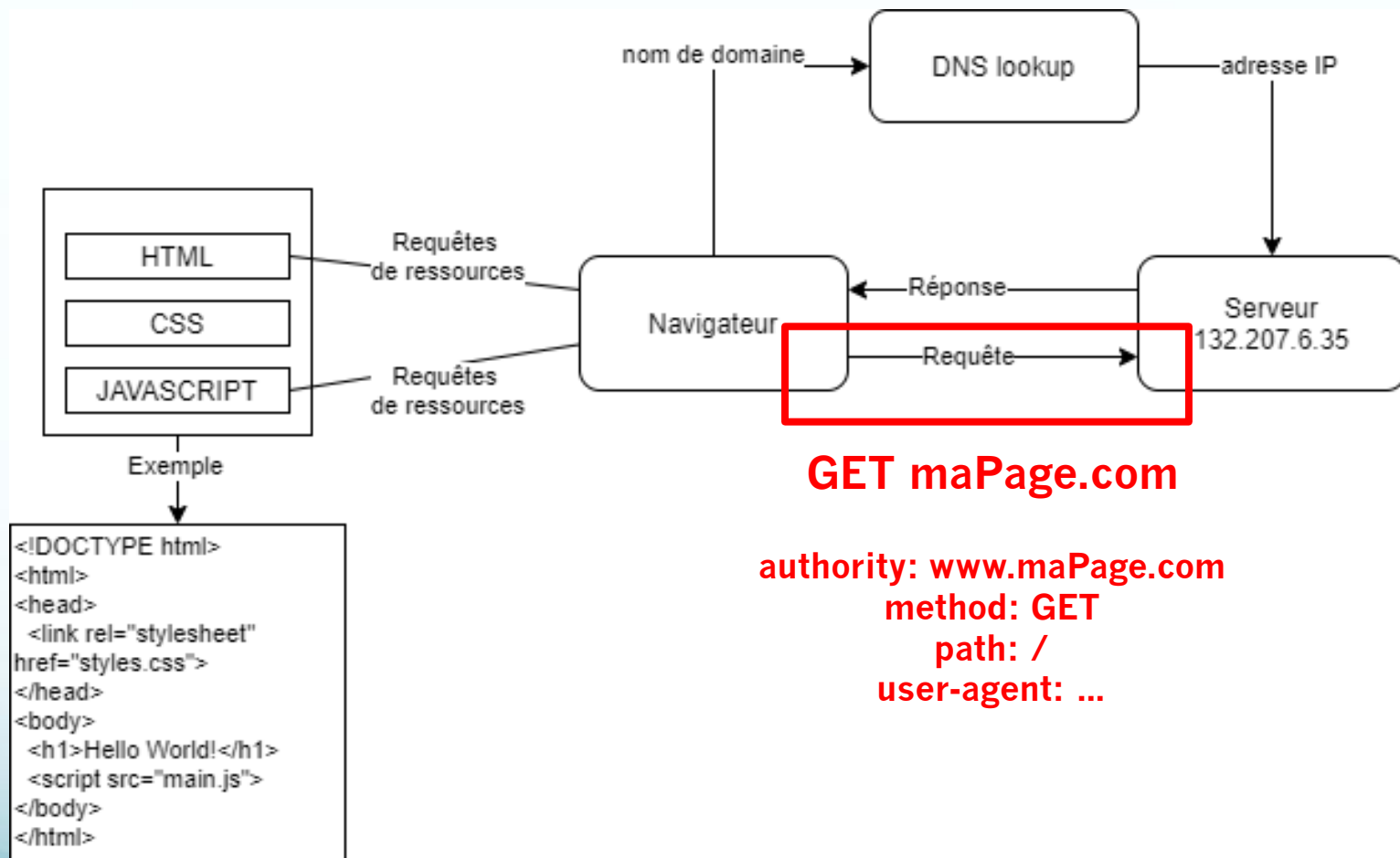
Architecture du Web (haut niveau)



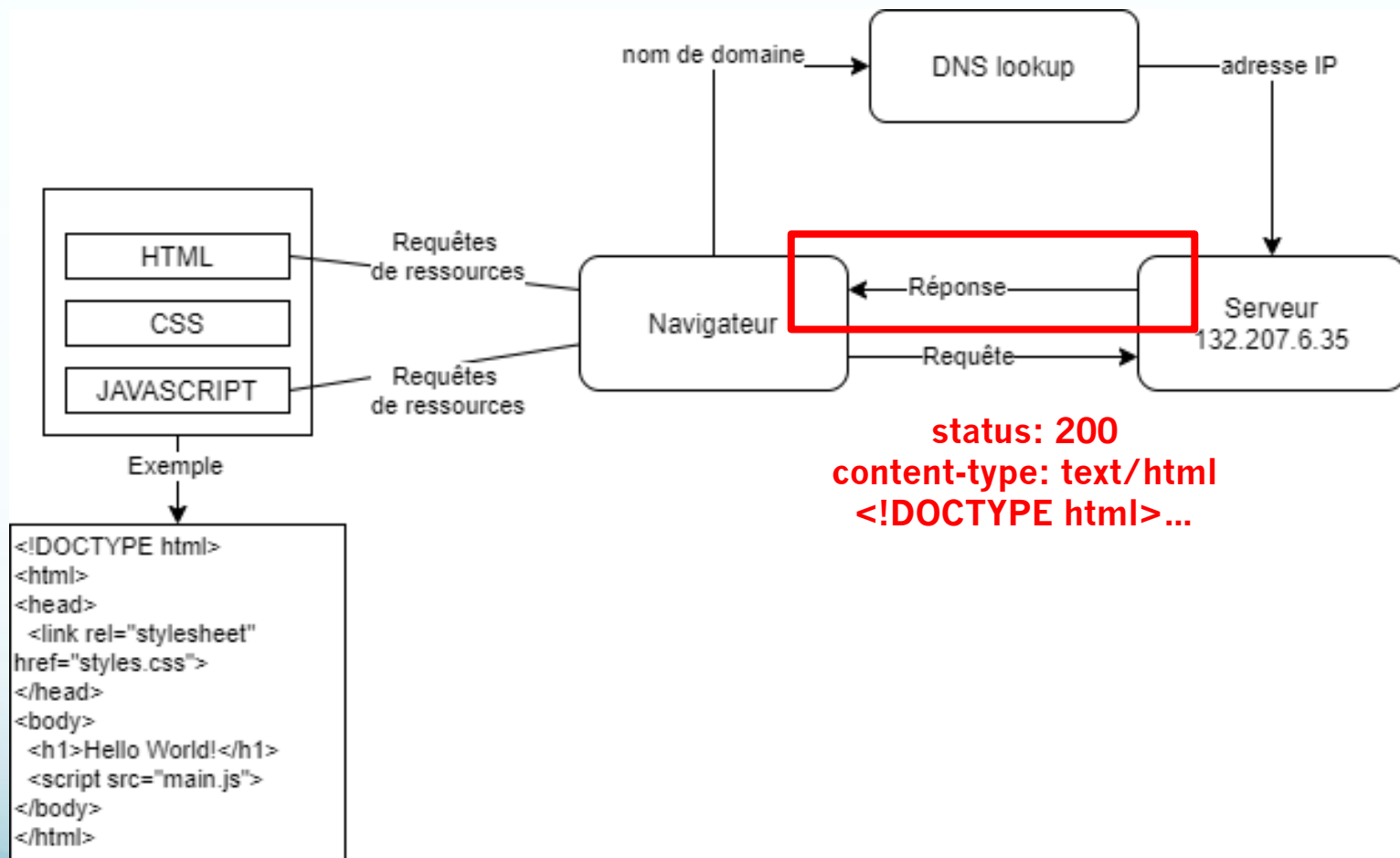
Architecture du Web - HTTP



Architecture du Web - HTTP

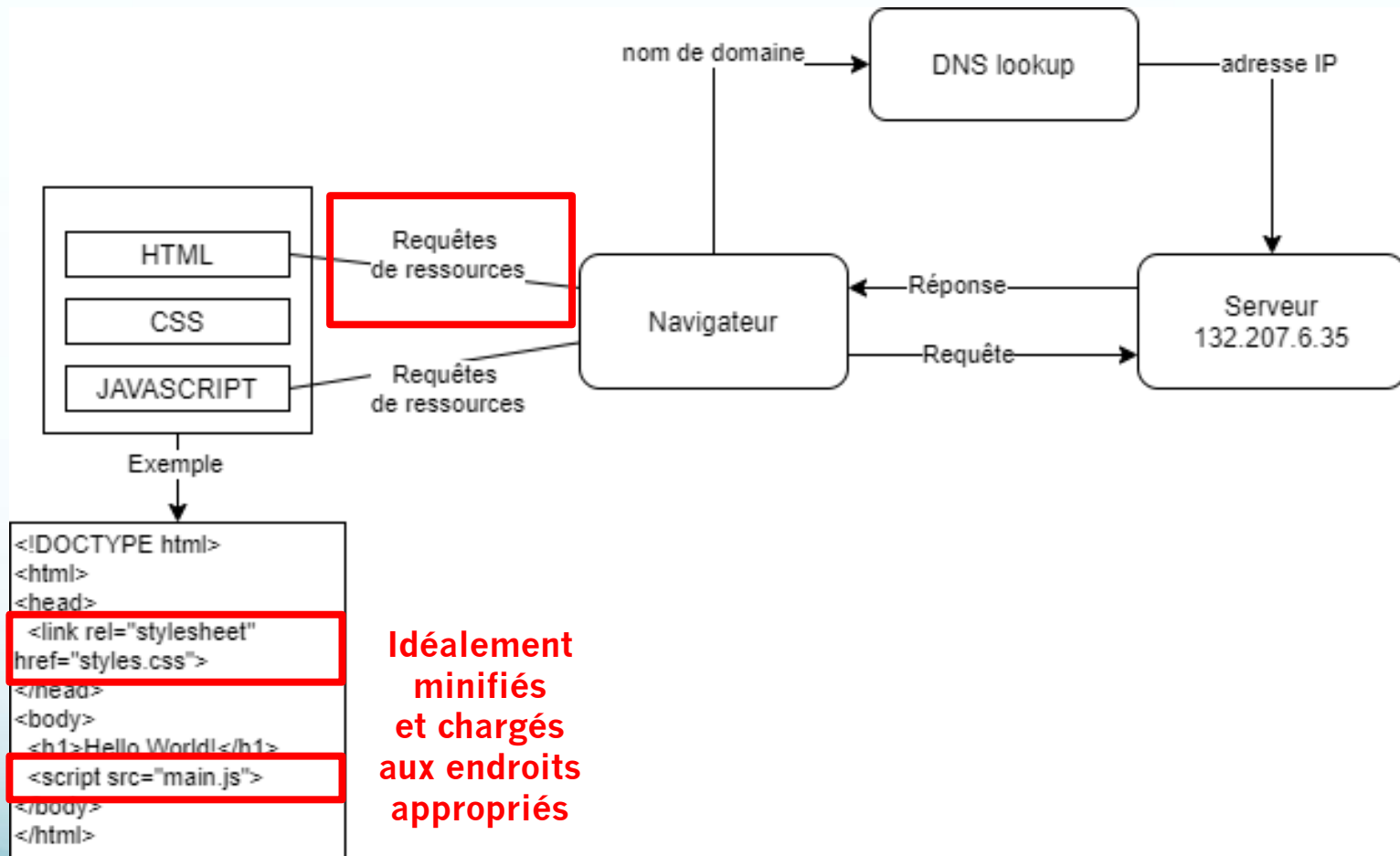


Architecture du Web - HTTP



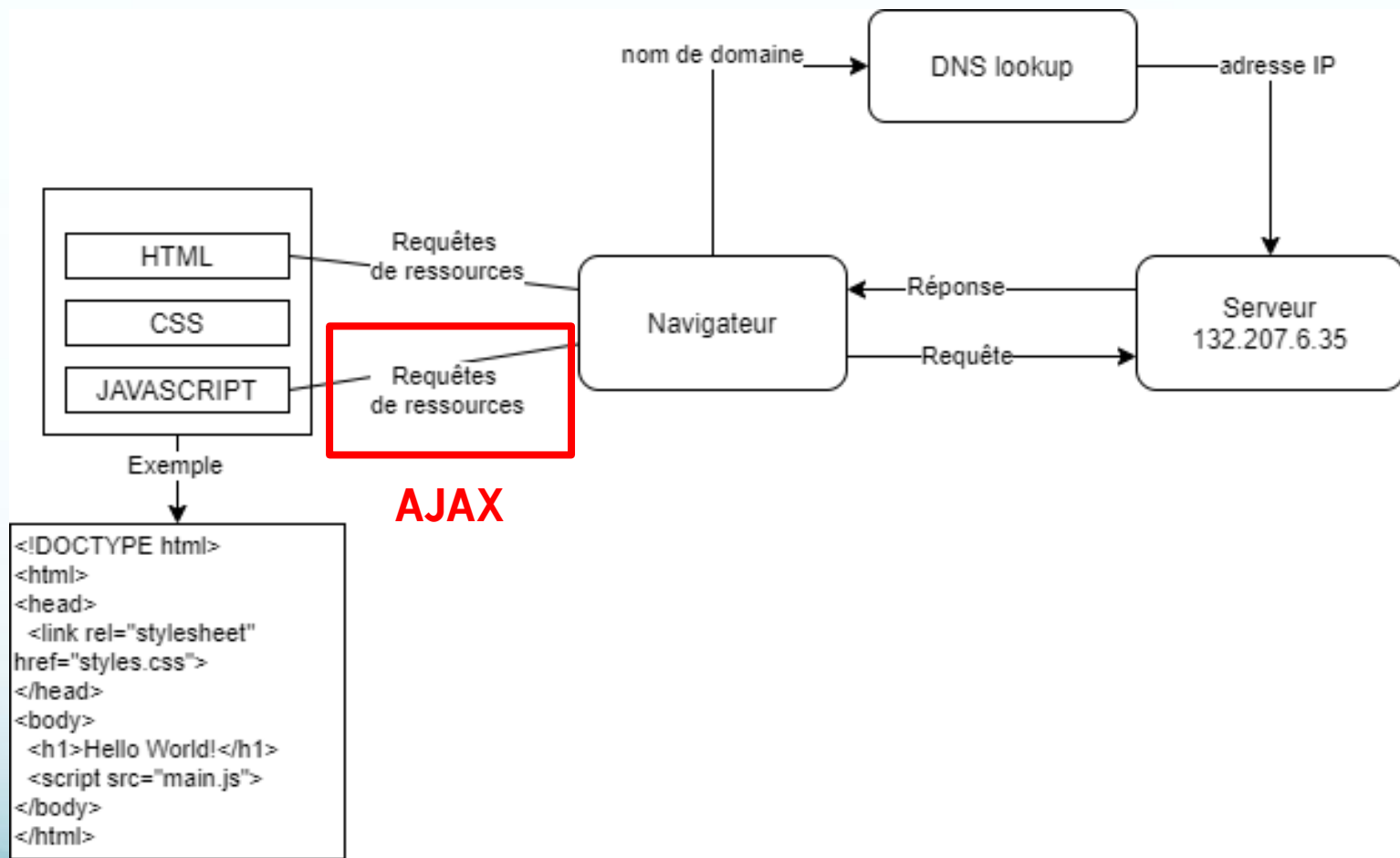
Architecture du Web

HTML et les ressources



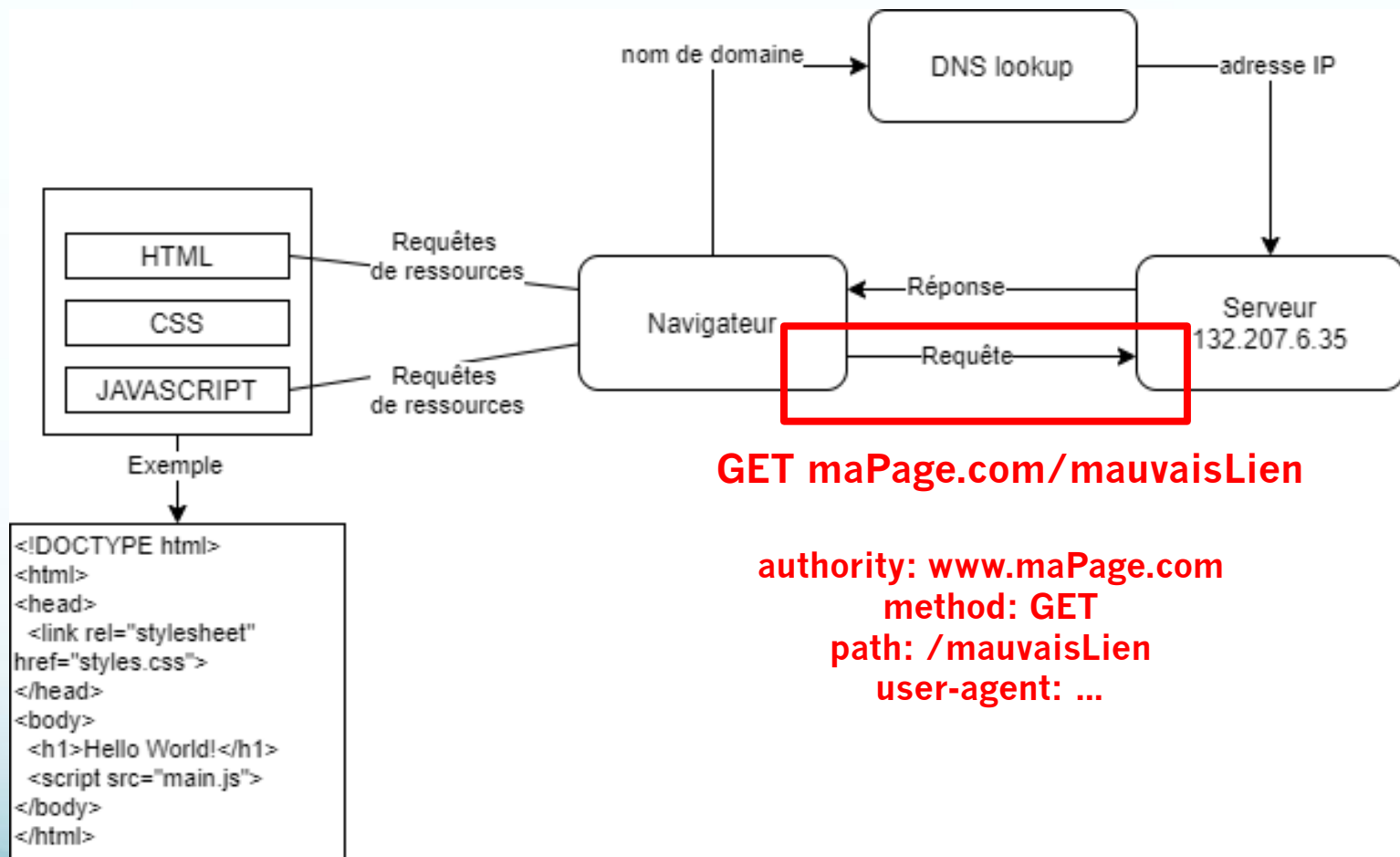
Architecture du Web

Asynchronous JavaScript and XML



Architecture du Web

Asynchronous JavaScript and XML

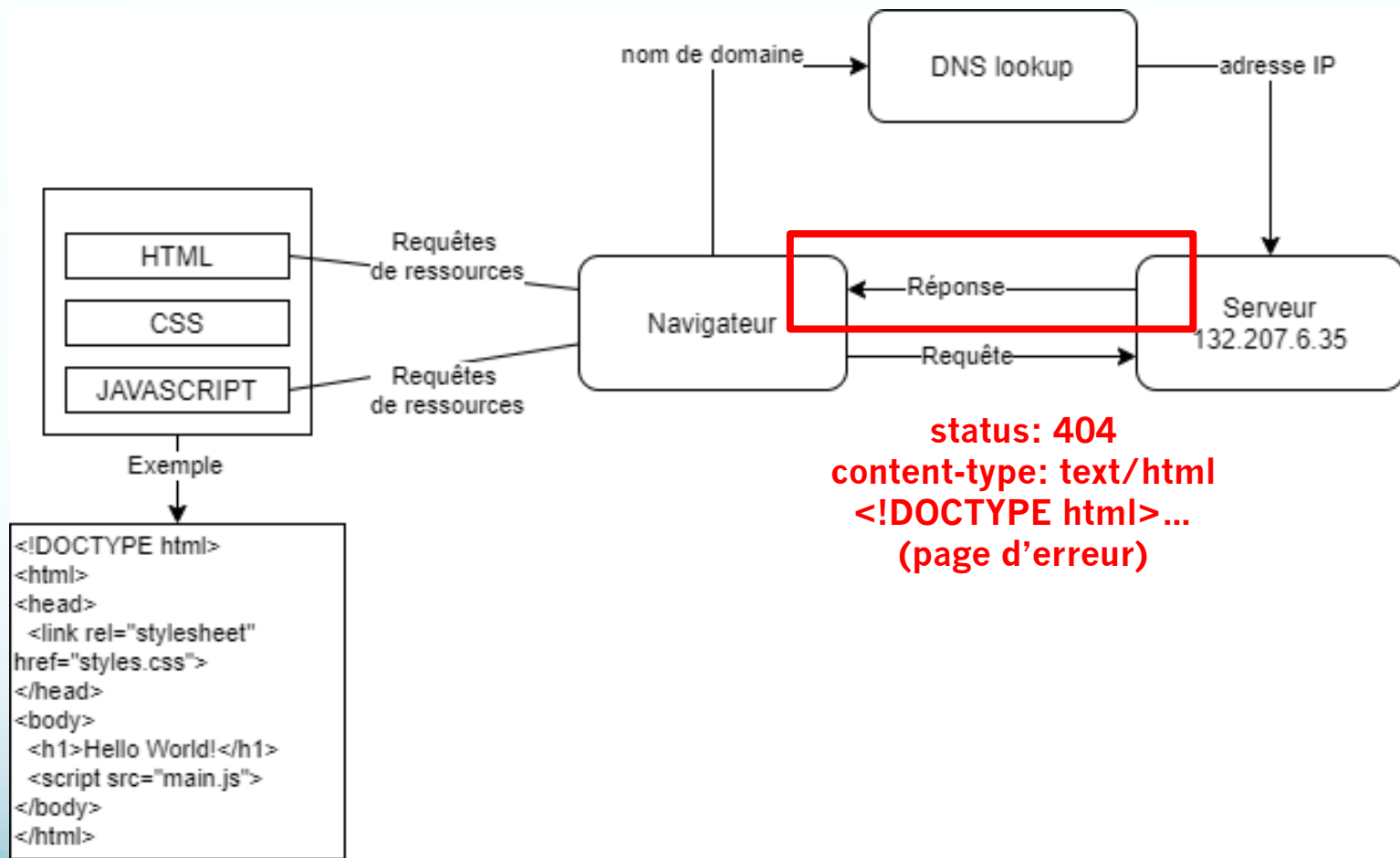


GET maPage.com/mauvaisLien

authority: www.maPage.com
method: GET
path: /mauvaisLien
user-agent: ...

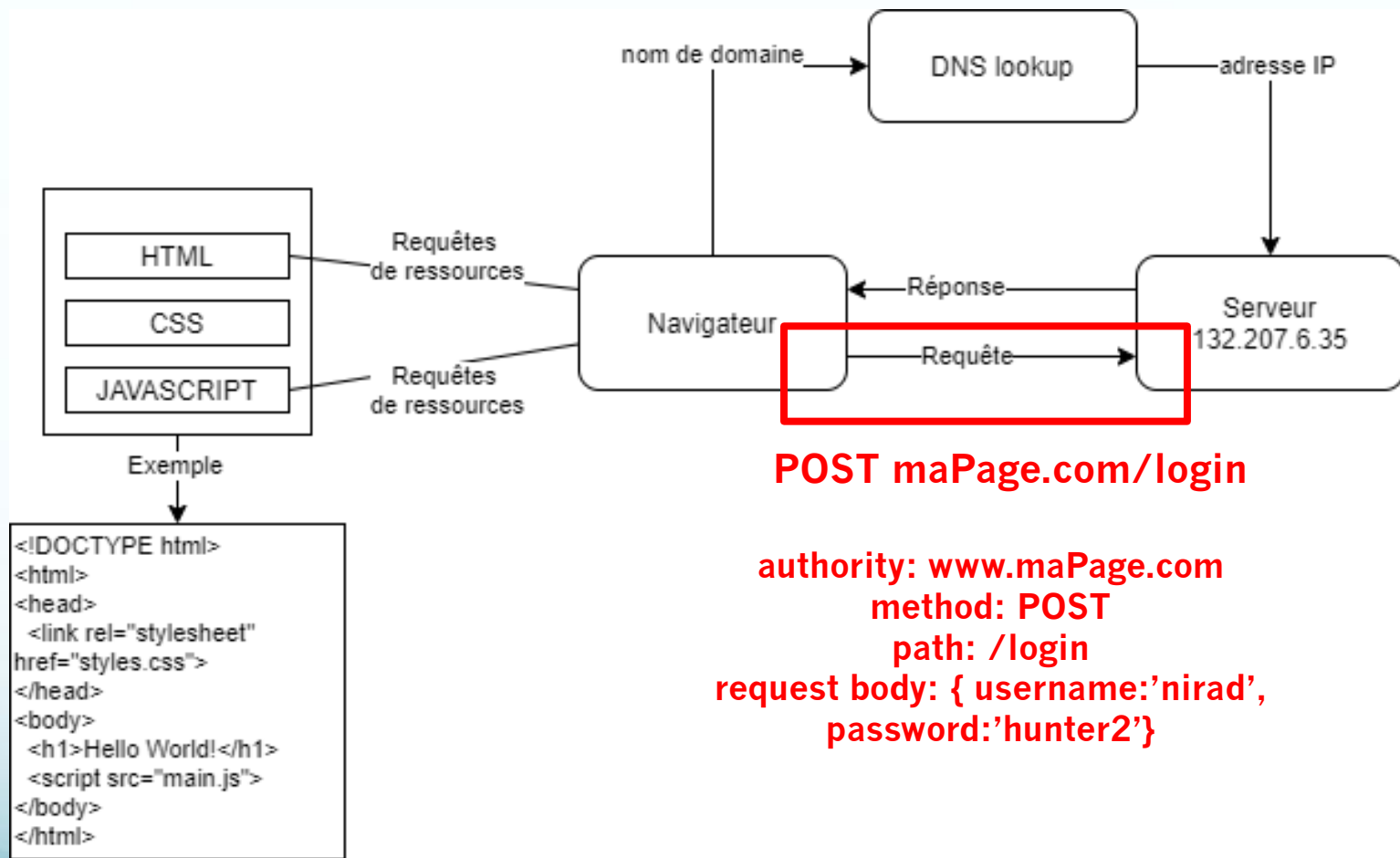
Architecture du Web

Asynchronous JavaScript and XML



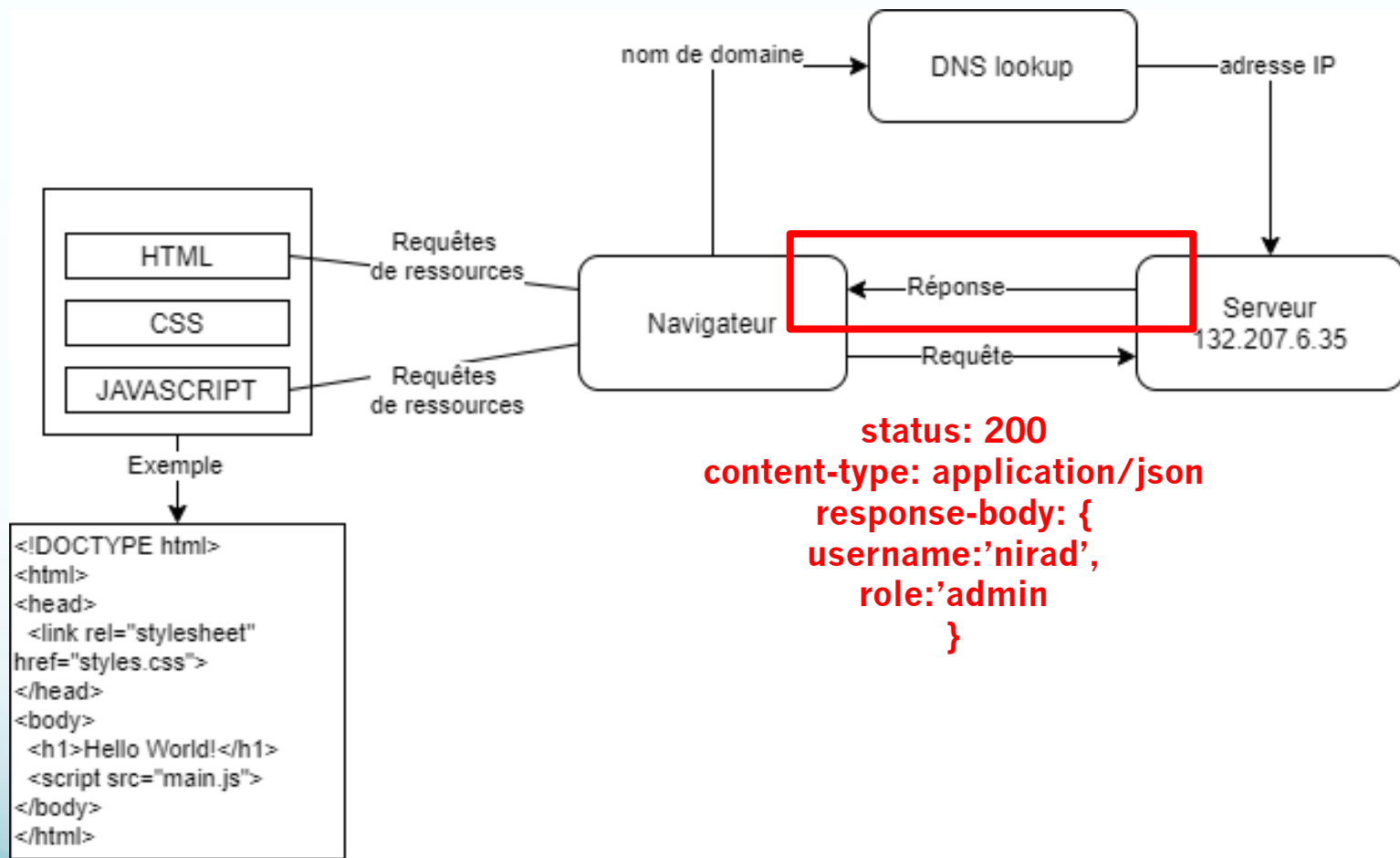
Architecture du Web

Asynchronous JavaScript and XML



Architecture du Web

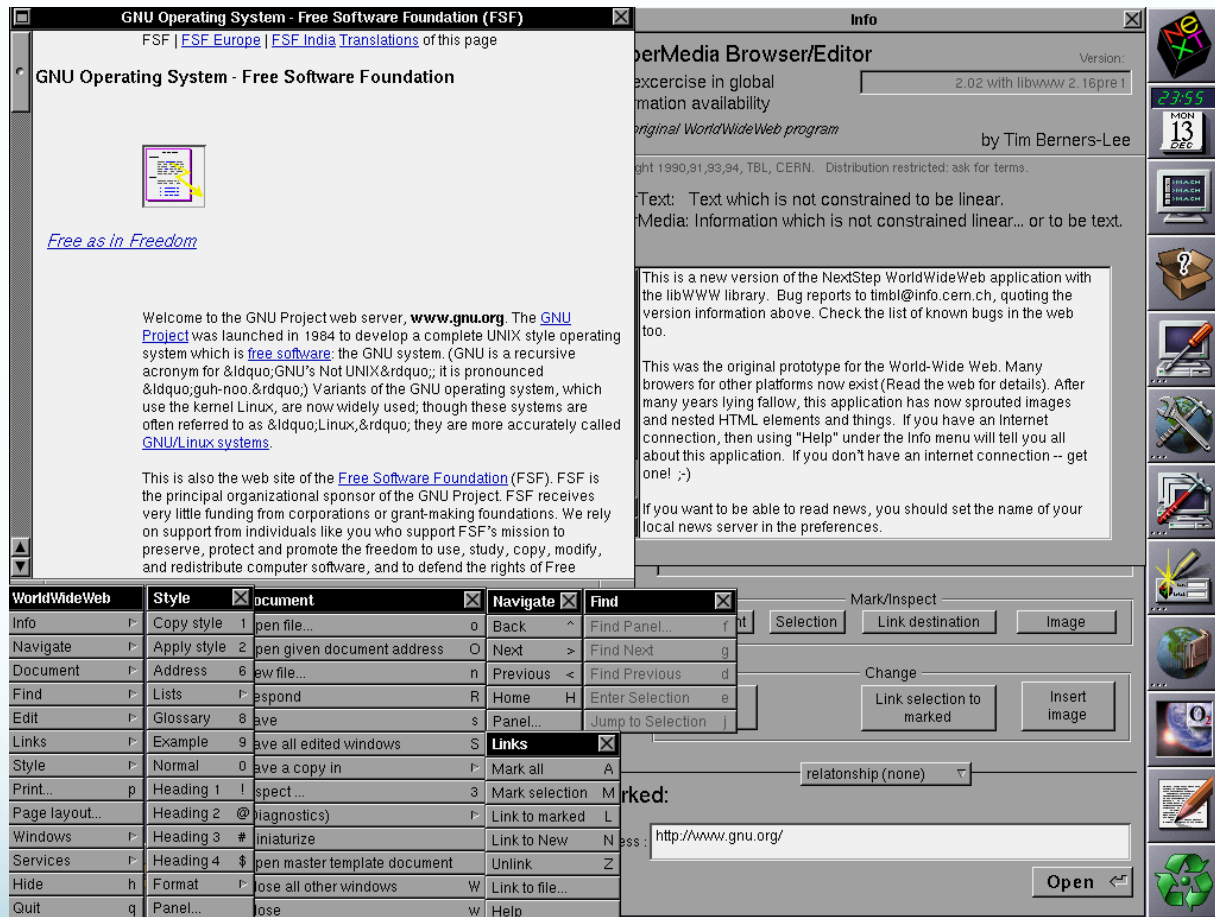
Asynchronous JavaScript and XML



Navigateurs Web

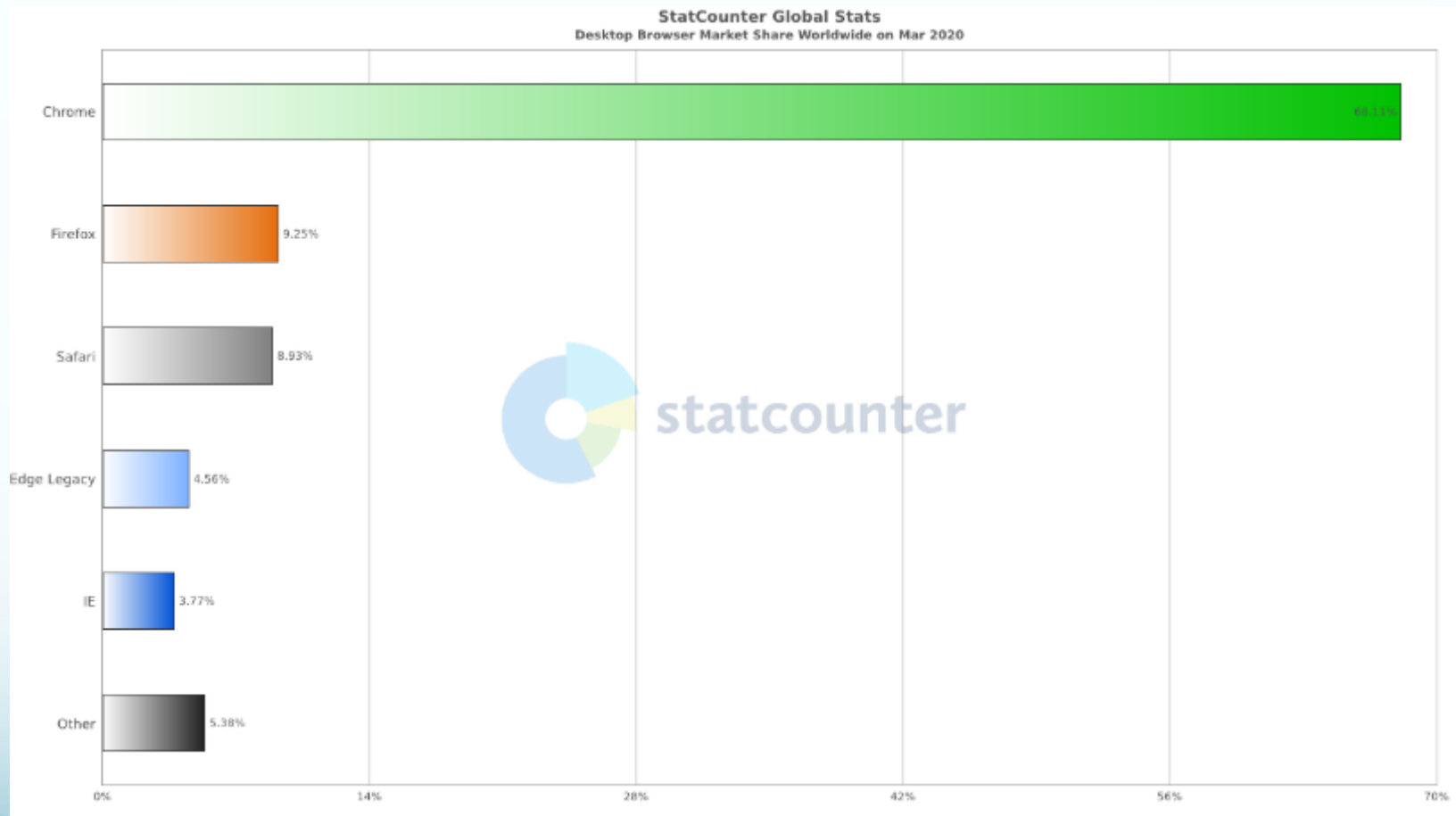
- Application logicielle qui permet d'accéder le Web.
- Rassemble un ensemble de technologies dans la même application
 - Moteur du navigateur (*Browser Engine*)
 - Moteur de rendu (*Rendering Engine*)
 - Moteur JavaScript
 - Gestionnaire de données
 - Gestionnaire de réseau
 - Etc.
- Disponible sur plusieurs appareils différents
 - Les appareils mobiles deviennent rapidement les appareils plus utilisés pour accéder au Web

Navigateurs Web - WorldWideWeb



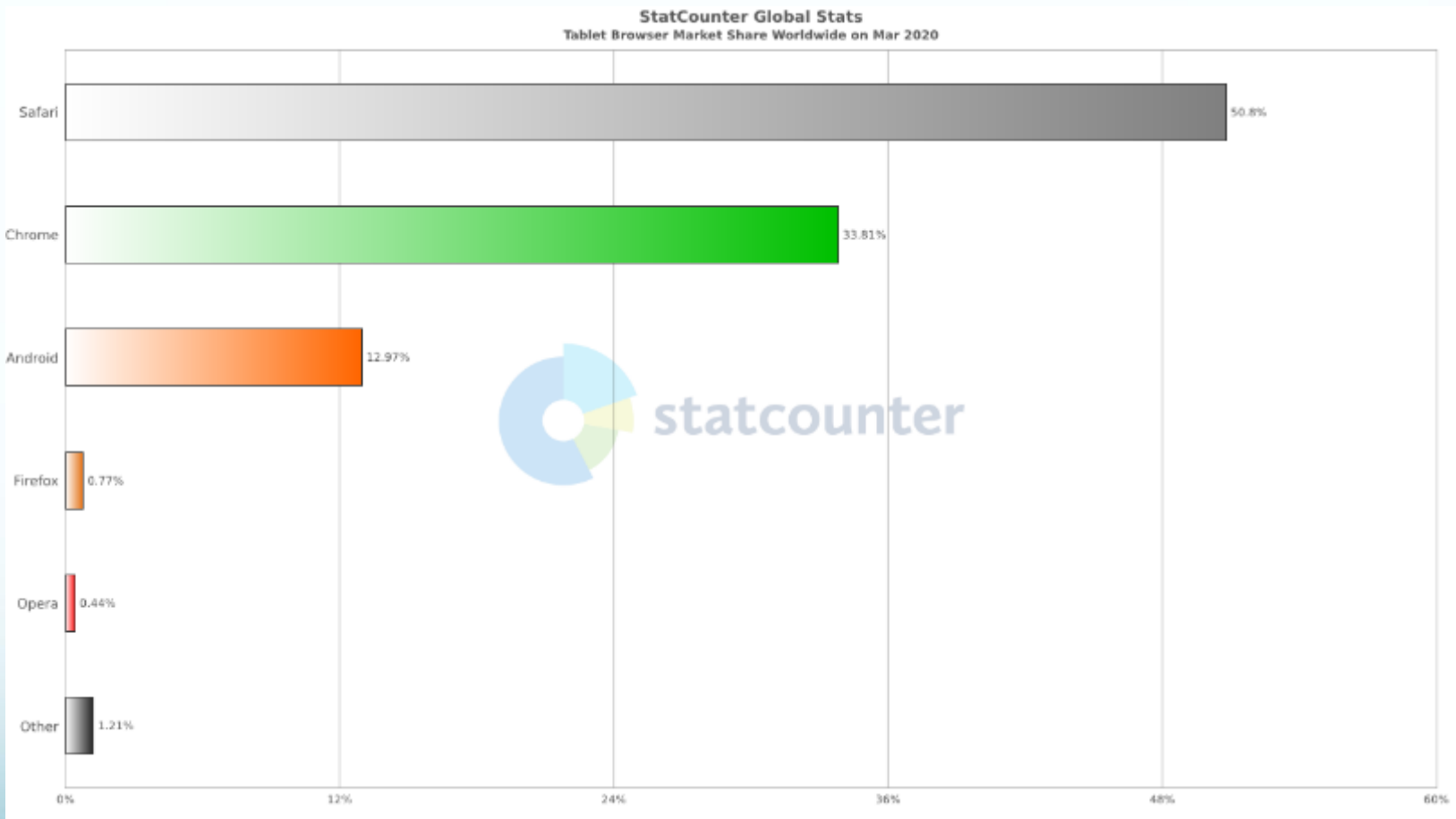
Source : <http://gnu.ethz.ch/www.levenez.com/firstbrowser.png>

Navigateurs Web – Distribution par utilisation - Bureau



Source : <https://gs.statcounter.com/browser-market-share/desktop/worldwide/#monthly-202003-202003-bar>

Navigateurs Web – Distribution par utilisation - Tablettes



Source : <https://gs.statcounter.com/browser-market-share/tablet/worldwide/#monthly-202003-202003-bar>

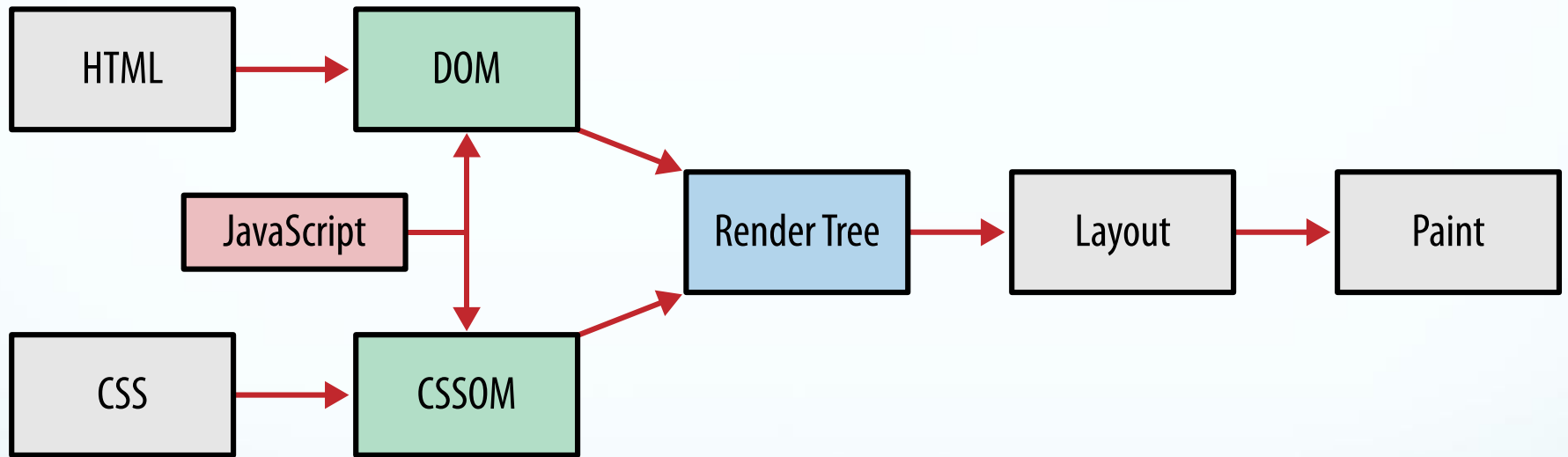
Browser Engine

- Les termes Browser Engine et Rendering Engine sont souvent interchangeables
- Transforme une page HTML + autres ressources en un rendu sur l'appareil de l'utilisateur.
- Peut être utilisé ailleurs qu'un navigateur web. Ex : Electron ou Thunderbird
- Plusieurs engins populaires :
 - Gecko (Mozilla, Thunderbird)
 - WebKit (Safari, tous les navigateurs sur iOS)
 - Blink (Chromium, Chrome, Edge)
 - EdgeHTML (ancien Edge, applications UWP)

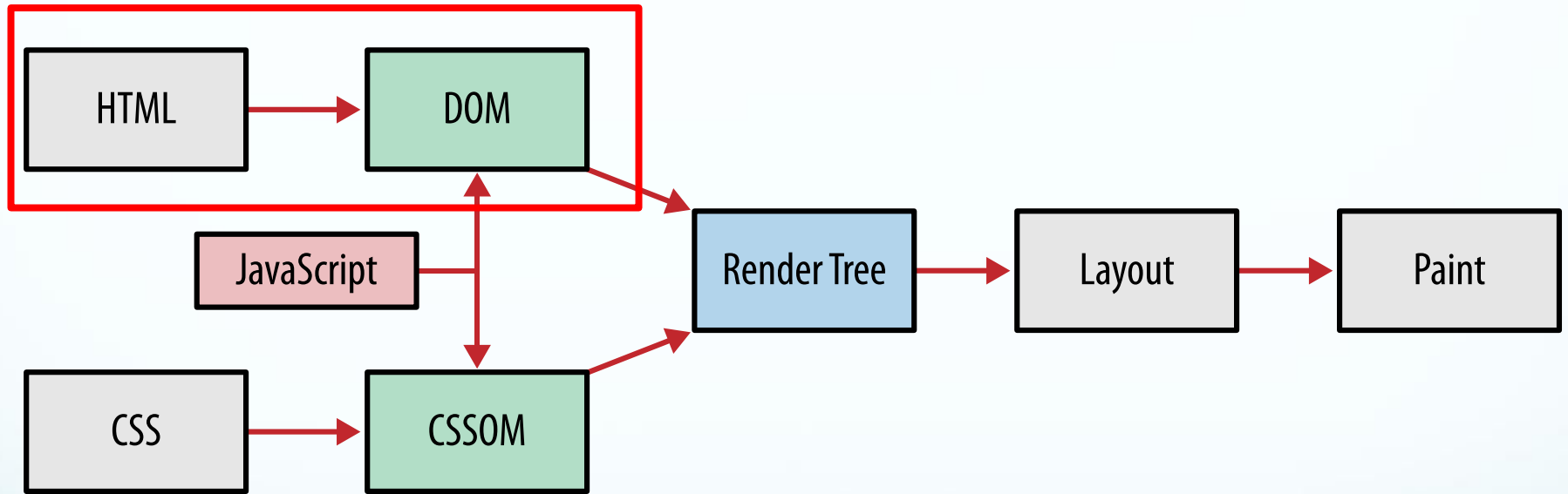
JavaScript Engine

- Compile et exécute du Javascript
- Initialement un simple interpréteur, la majorité des moteurs modernes compilent le JS
- Peut être utilisé ailleurs qu'un navigateur web. Ex : NodeJS
- Plusieurs engins populaires :
 - SpiderMonkey (Mozilla, Thunderbird)
 - Nitro (Safari)
 - V8 (Chromium, Chrome, NodeJS, Edge)
 - Hermes (applications Android avec React Native)

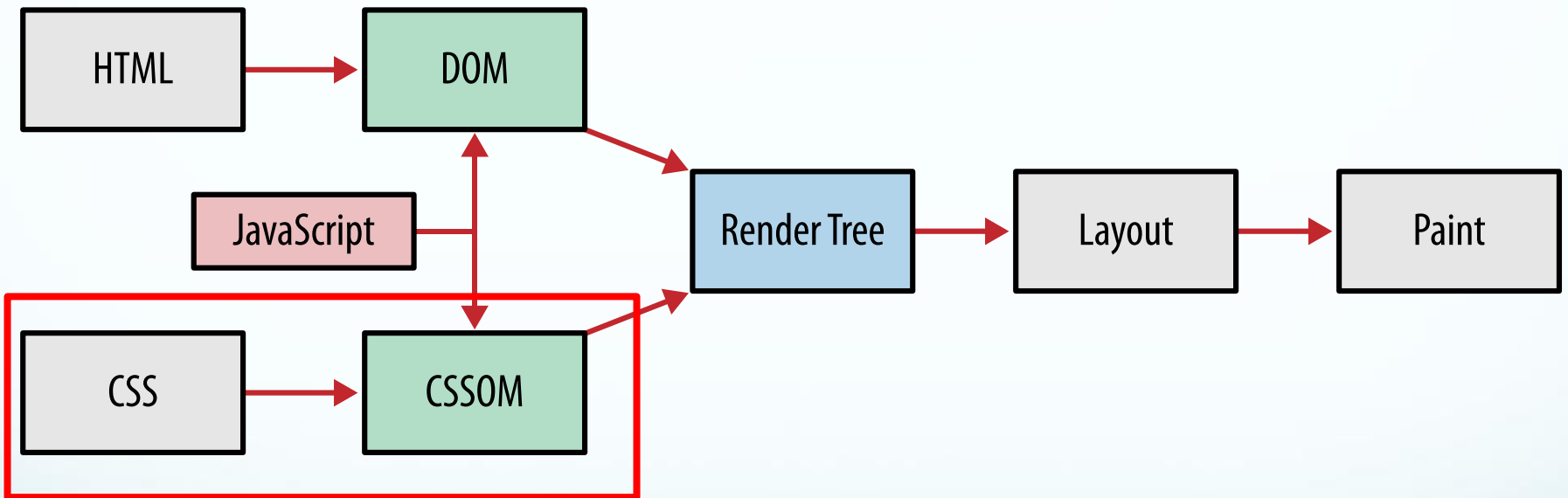
Rendu d'une page web



DOM – Document Object Model

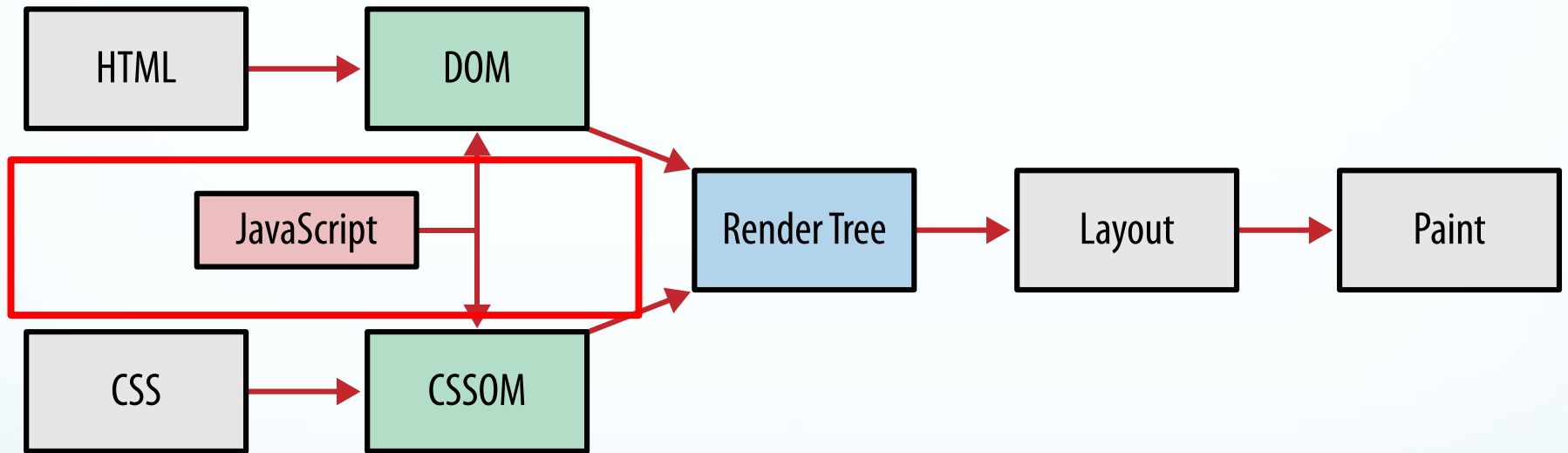


CSSOM – CSS Object Model



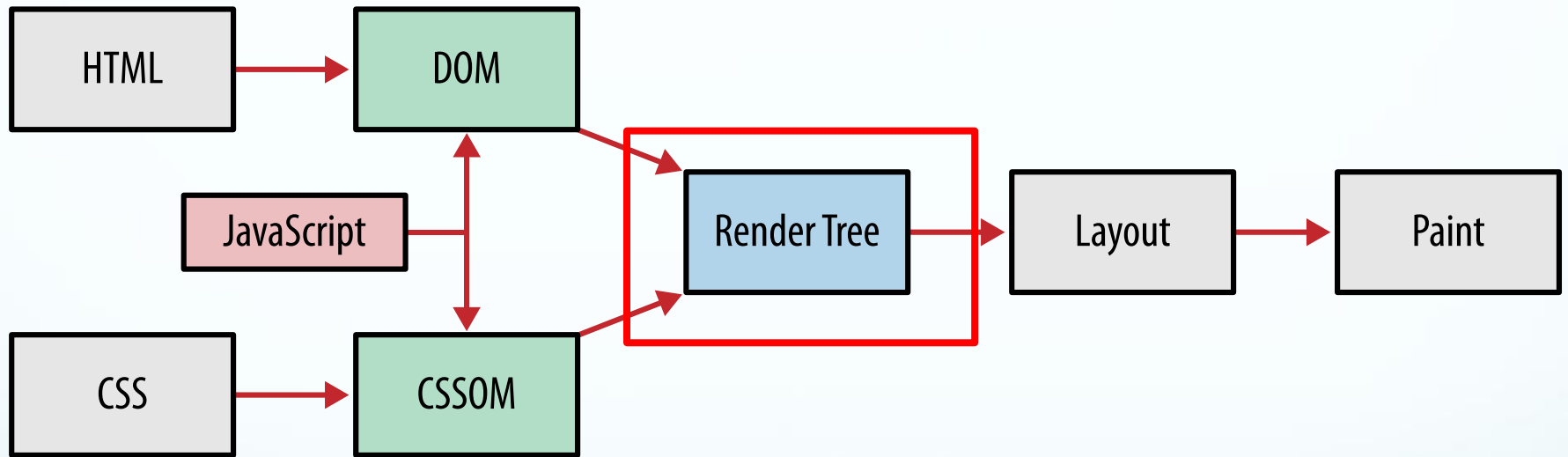
Peut interrompre la création du DOM
Ne peut pas être interprété séquentiellement

JavaScript



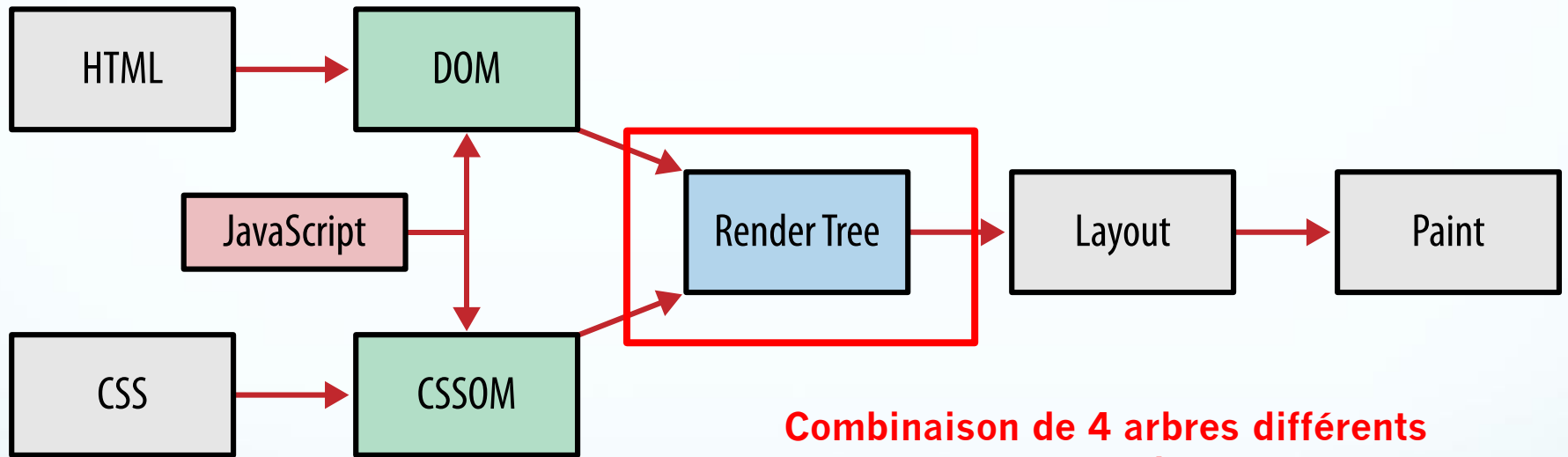
Peut interrompre la création du DOM
Peut modifier le DOM lorsqu'il est exécuté

Render Tree



Combinaison du DOM et CSSOM
Représente ce qui sera affiché à l'écran
N'est pas 100% pareil au HTML

Render Tree



Combinaison de 4 arbres différents

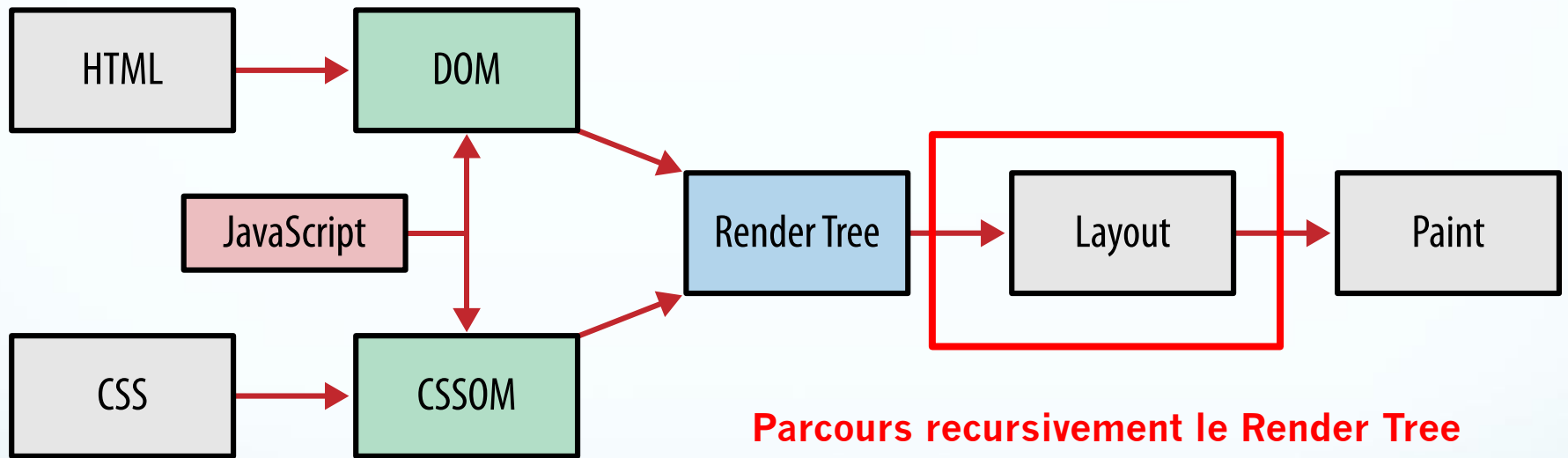
RenderObjects

RenderStyles

RenderLayers

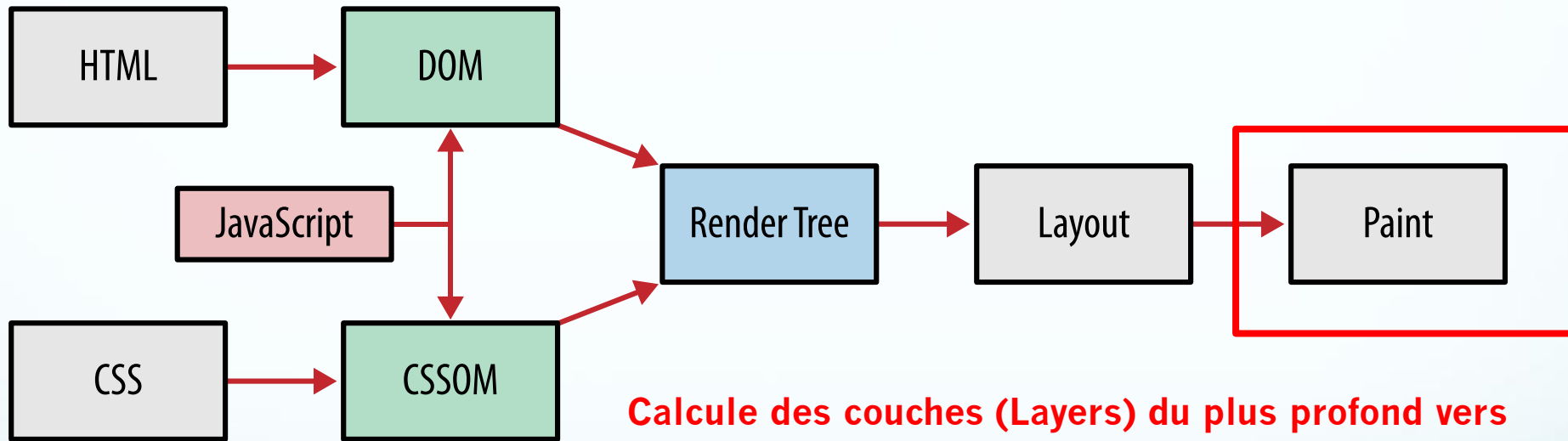
Line boxes

Layout



Parcours récursivement le Render Tree
Calcule la position et la taille de chaque élément
Logique supplémentaire pour la gestion des changements asynchrones

Paint



Calcule des couches (Layers) du plus profond vers le plus proche
Génère des images envoyés au GPU (si activé)
Composition : création de l'image finale

Ressources supplémentaires

- [Ryan Seddon: So how does the browser actually render a website | JSConf EU 2015](#)
 - Excellente vidéo qui résume le processus de dessin
 - Suggestion: à regarder à x1.25
- [Tali Garsiel : How Browsers Work](#)
 - Analyse détaillée de comment Chrome fonctionne
 - Écrit en 2011, mais les principes généraux restent pareils

Mention d'honneur

- Lynx : navigateur web utilisable dans un terminal
 - Existe depuis 1993
 - Ne supporte pas JavaScript
 - Utile pour accéder à des pages web sans interface graphique

Lynx : page d'accueil de Google

```
nirad@DESKTOP-785HP00: ~  
Search Images Maps Play YouTube News Gmail Drive More »  
Web History | Settings | Sign in  
  
Google  
  
Pourquoi Moodle est down ? _____  
Google Search I'm Feeling Lucky Advanced search  
  
Google offered in: Français  
  
Advertising Programs Business Solutions About Google Google.ca  
  
© 2020 - Privacy - Terms  
  
(Textfield "%s") Enter text. Use UP or DOWN arrows or tab to move off.  
Enter text into the field by typing on the keyboard  
Ctrl-U to delete all text in field, [Backspace] to delete a character
```