



Outils et architectures pour le web



Alexandre.Demeure@inria.fr



Supposition

Vous savez à peu près utiliser

- Notion de clients et serveurs
- HTTP, notion de ressources
- HTML : notion de balises, d'attributs
- CSS : notion de sélecteur, style, cascade
- Javascript : objets, prototypes, fonctions, closures

Supposition

—

Vous

Voyons ça...!

Client/Serveur/HTTP ?

Supposition

Maîtriser ces technologies nécessite de se tenir à jour

- Evolution des langages : HTML5, ES 2015, CSS3, ...
- Evolution des frameworks : Angular 2
- Ce cours (comme tous les autres) sera en partie obsolète à la fin

Apprendre à apprendre...

Mozilla Developer Network : MDN, section Technologie pour le web

<https://developer.mozilla.org/fr/docs/Web/JavaScript>

Eloquent Javascript

<http://eloquentjavascript.net/>

Stack Overflow (forum)

IRC: Freenode #javascript

Objectifs et plan

- Apprendre à mettre en place une chaine de compilation (séance 1)
 - Modules et bases de modules (CommonJS, NPM)
 - Chaine de compilation (GULP, webpack)
 - Gestion de code (GIT)
- Comprendre une architecture logicielle (séance 2 et 3)
 - Modèles d'architectures Seeheim, MVC, MVVM
 - On va se concentrer sur le côté client (Angular 1.5)
 - Le projet fera intervenir NodeJS côté serveur.

Les modules

Quels sont les risques d'inclure plusieurs fichiers javascripts dans une même page ?

Les modules

Quels sont les risques d'inclure plusieurs fichiers javascripts dans une même page ?

- Trop de fichiers à inclure dans le HTML alors qu'on programme en JS.
- Pénible à maintenir.
- Variables ou fonctions qui ont le même nom !

Les modules

Quels sont les risques d'inclure plusieurs fichiers javascripts dans une même page ?

- Trop de fichiers à inclure dans le HTML alors qu'on programme en JS.
- Pénible à maintenir.
- Variables ou fonctions qui ont le même noms !

Comment assurer l'isolation des variables définies dans les fichiers ?

Les modules

Quels sont les risques d'inclure plusieurs fichiers javascripts dans une même page ?

- Trop de fichiers à inclure dans le HTML alors qu'on programme en JS.
- Pénible à maintenir.
- Variables ou fonctions qui ont le même noms !

Comment assurer l'isolation des variables définies dans les fichiers ?

- Utilisation des modules

Cas d'étude : une liste de choses à faire

Ce cas d'étude va permettre

- Illustrer un principe de base : NF <-> IHM
- Réviser quelques notions de Javascript
- Illustrer l'utilisation de modules
- Illustrer une chaine de compilation avec GULP

Nouvelle tâche	
○ Tâche 1	X
○ Tâche 2	X
○ Tâche 3	X

Cas d'étude : une liste de choses à faire

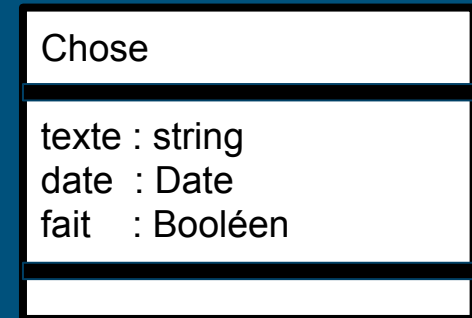
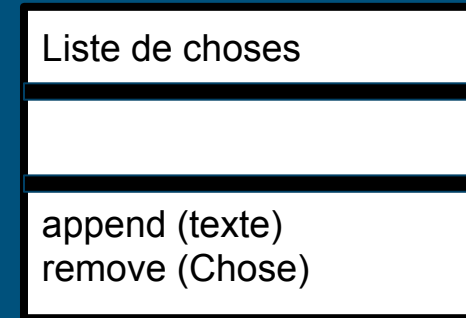
PRINCIPE : Bien séparer

IHM

et

Noyau Fonctionnel (NF)

Nouvelle tâche	
○ Tâche 1	X
○ Tâche 2	X
○ Tâche 3	X



Cas d'étude : une liste de choses à faire

PRINCIPE : Bien séparer IHM et Noyau Fonctionnel (NF)

- Noyau fonctionnel
 - Ne connaît pas directement l'IHM
 - Fournit des moyens d'être observé et manipulé programmatiquement
 - Gère les notifications, la prévention des erreurs, l'annulation
- IHM
 - Connait le noyau fonctionnel
 - Utilise les moyens fournis par le Noyau Fonctionnel

Cas d'étude : une liste de choses à faire

On s'attaque à l'implémentation :

- Un peu de révision sur les objets, les prototypes, les constructeurs
- Comment implémenter un mécanisme d'abonnement basique
- Closure, bind
- La recherche dans un tableau...
- Abonnements aux événements utilisateurs
- ...

Code



Chaine de compilation avec GULP

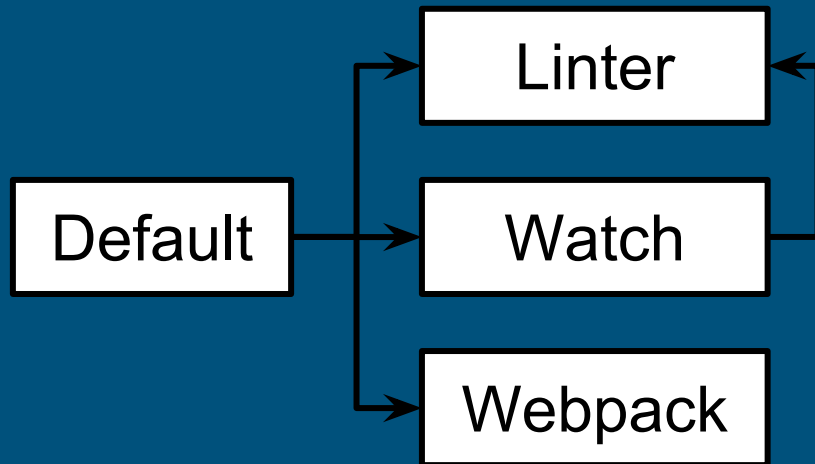
Dans quels cas est-ce utile ?

- J'écris mes feuilles de style avec LESS, SASS ou autre...
- Je veux éviter de préfixer mes attributs de styles avec webkit-, moz-, ...
- J'ai envie de coder en ES6 ou TypeScript mais il faut que ça tourne en ES3...
- Je veux limiter le nombre de fichiers à envoyer au client
- Je veux minifier/compacter le code pour économiser la bande passante
- Je veux faire de la vérification sur mon code
- etc.

Chaine de compilation avec GULP

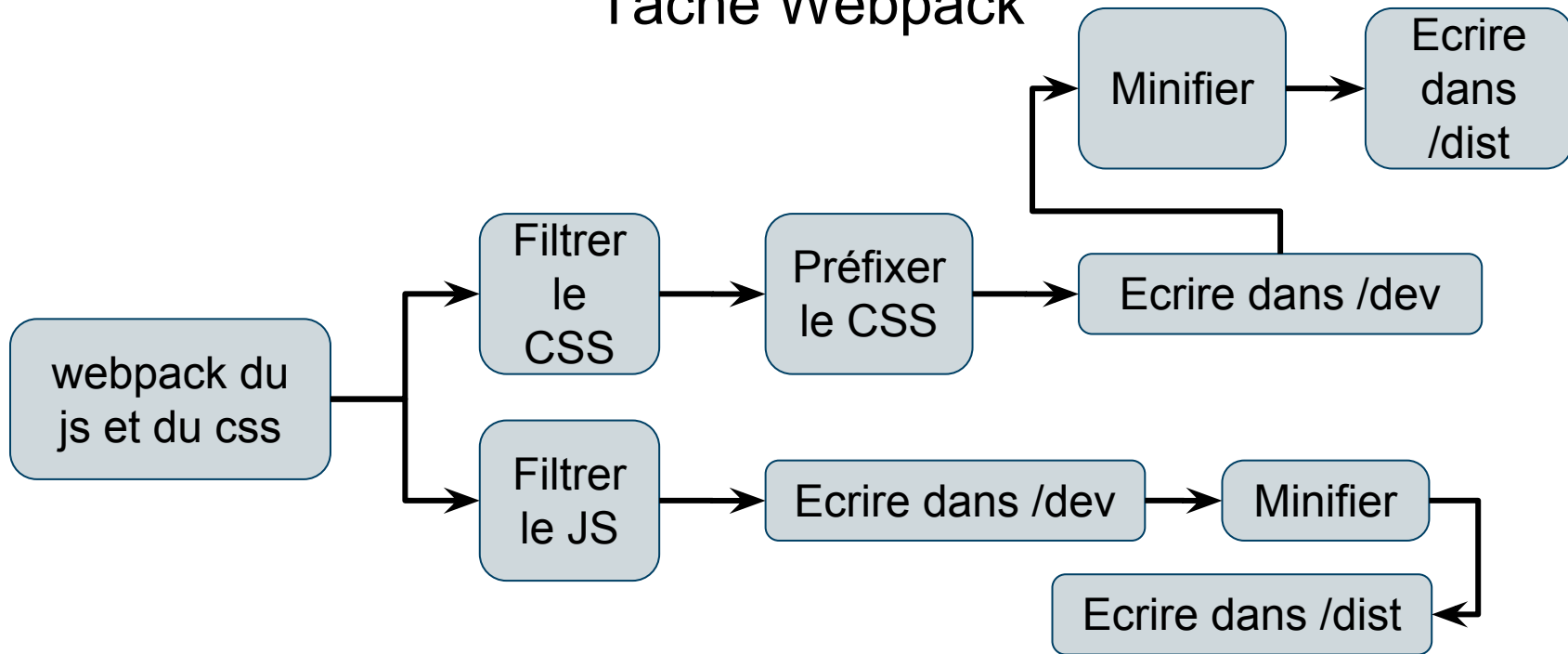
On définit des tâches et des enchaînements de tâches

Exemples avec notre code:



Chaine de compilation avec GULP

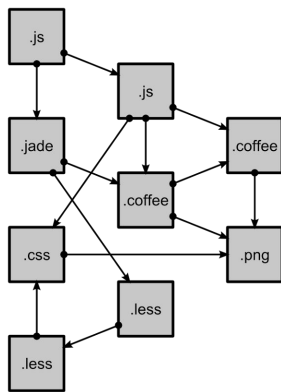
Tâche Webpack



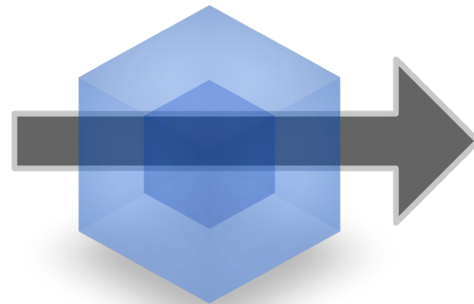
Code

Voir aussi :

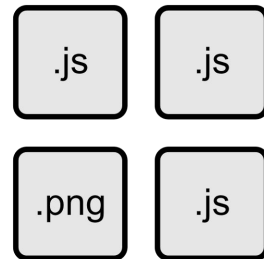
- <http://gulpjs.com/>
- Webpack



modules
with dependencies



webpack
MODULE BUNDLER



static
assets

Les bases de modules

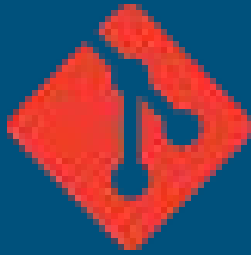
Capitaliser et partager les modules.

Gérer les versions.



Bower

et bien sur



git

<https://git-scm.com/about/>

<https://github.com/>

On reprend le code

On ajoute la commande vocale ?