



Angular

Journées Mathrice 2019

PARIS

Simon Chevance (MATHDOC)

versions



<https://angularjs.org/>

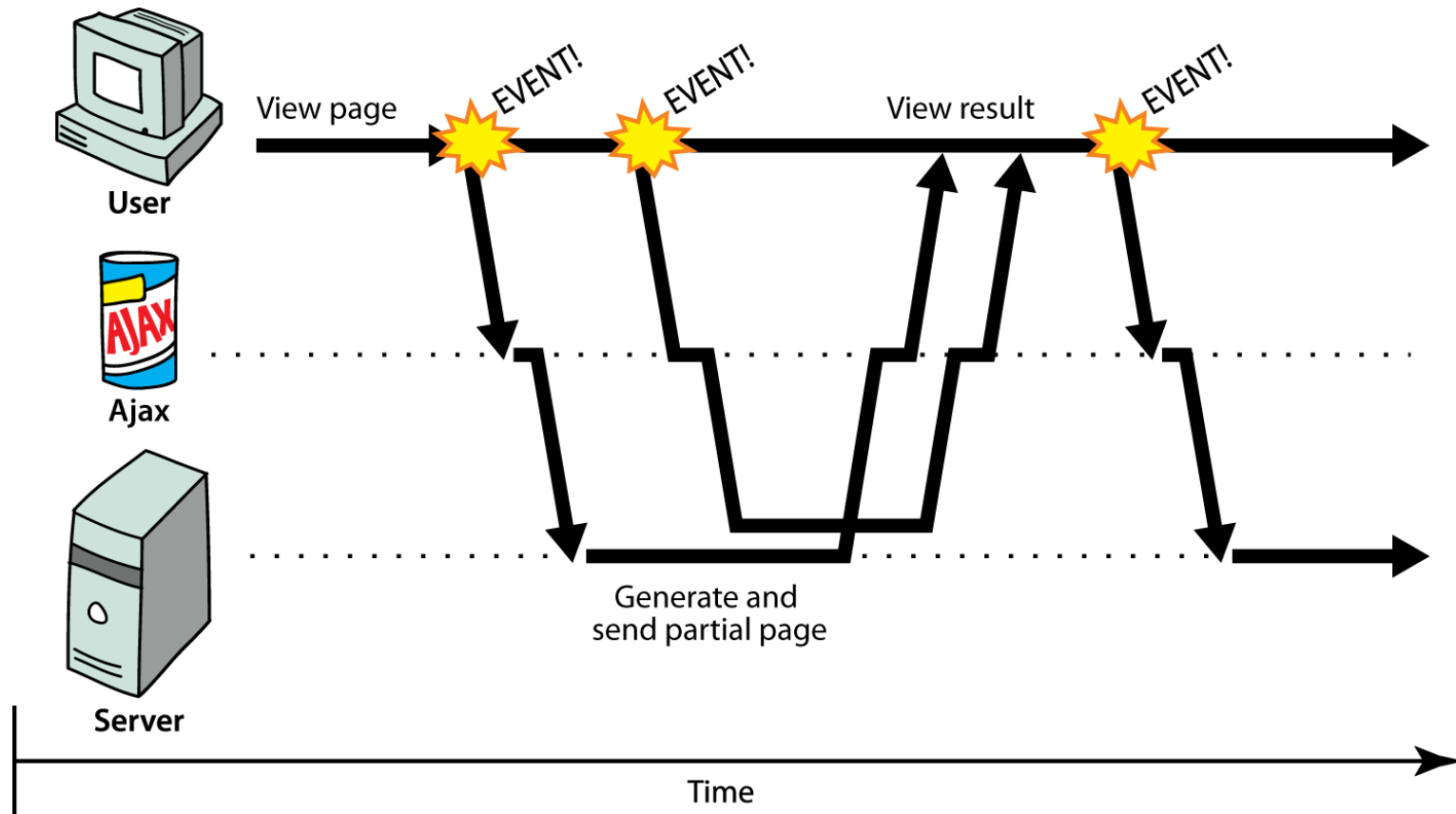
Version < 2



<https://angular.io>

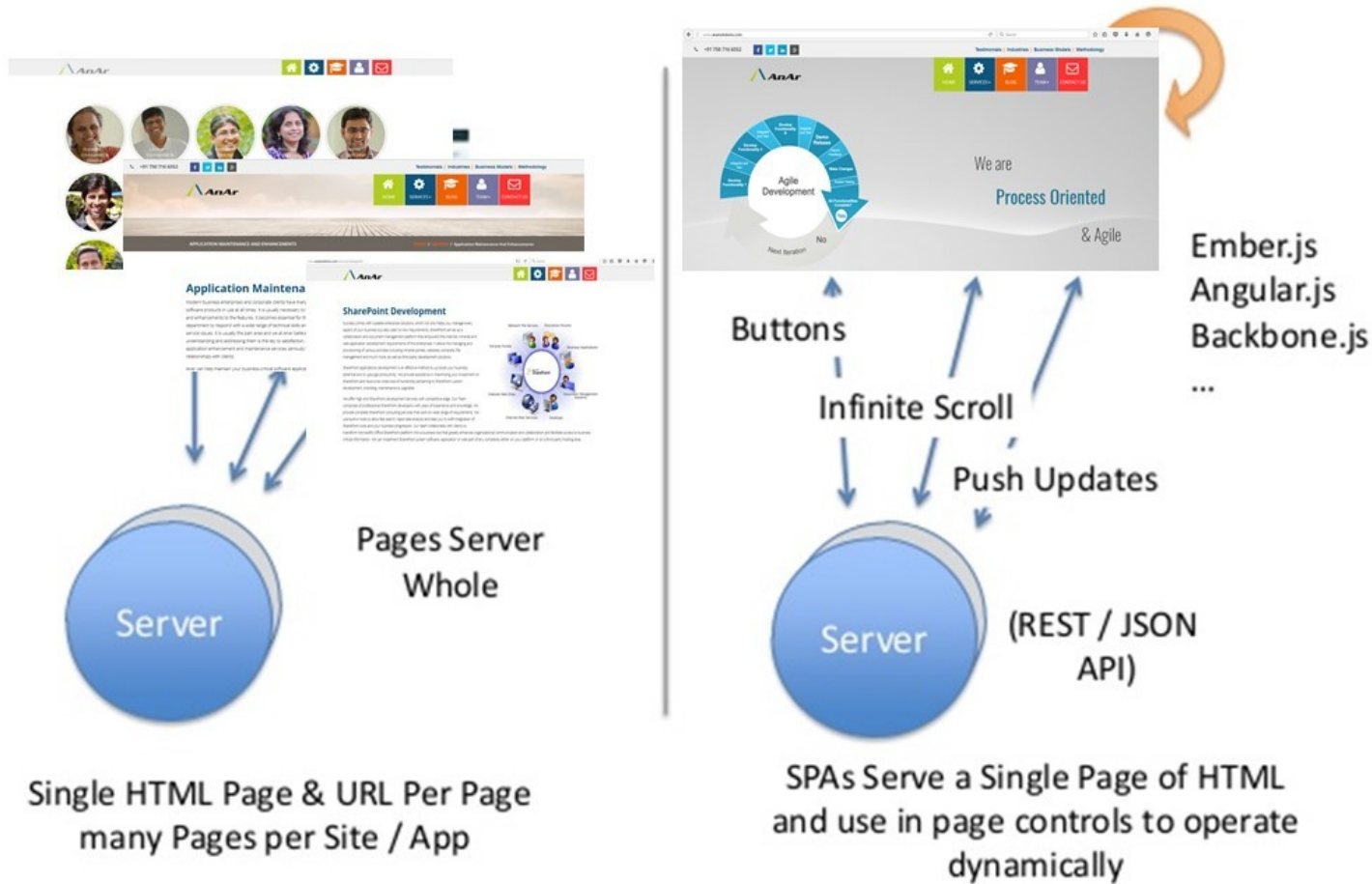
Version \geq 2 : actuellement v7

Asynchrone



SPA

Single Page Web Applications



concepts

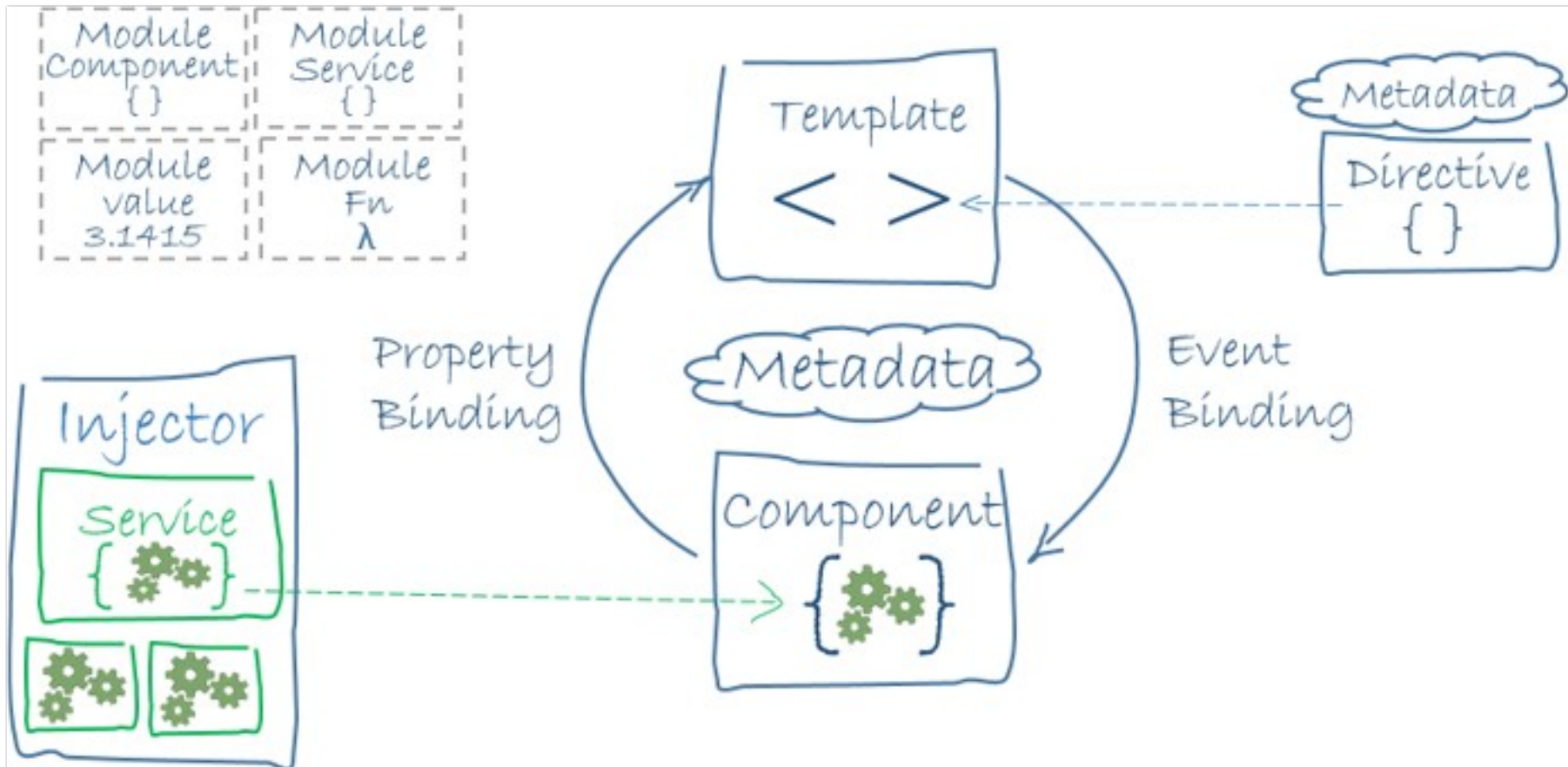
- Concepts :
 - Component/Module (composant angular utilisable dans l'application)
 - Service (composant commun utilisé en injection de dépendance)
 - Template / Data Binding (two-way data binding)
 - SPA Routage : Router

- Typescript... « nouveau » langage

Demo Simple

- Installation angular : *npm install -g @angular/cli*
- Creation projet : *ng new monprojet*
- Start projet : *ng serve --open*
- Creation composant : *ng g c mycomponent*
- Build (*ng build*)
- Taille ? (518Mo / 8.2Mo dont la moitié pour les map et 11ko pour le main.js)

concepts



Objectif

- Application gestion users et groupes
 - Angular Material (<https://material.angular.io/>)
 - Firebase (<https://firebase.google.com/docs/reference/rest/database/>)
 - Code sur plmlab
<https://plmlab.math.cnrs.fr/simon.chevance/angular-manage-users>
 - *ng add @angular/material*
 - *ng generate @angular/material:material-nav --name=Menu*
 - ~~*ng generate @angular/material:material-table --name=Users*~~

Difficultés rencontrées

- Compréhension Observable (il faut utiliser les Subject)
- template de MatTable non ok : pas pu faire de filtre car non héritage de MatTableDataSource
- Organisation Composant par page : sans doute pas la meilleure pratique
- Debug : à tester, debug de TS dans IDE
- TypeScript

Conclusion

- Pour des applications « professionnelles » SPA
- CLI plus capable que les autres
- Lourd ... lent ?
- Courbe d'apprentissage plus longue ?
- Un certain nombre de concepts à appréhender pour bien l'utiliser en particulier two-way binding, Observable/Subject, DI
- Bonne communauté et a priori bonne pérennité (Google)