

# M1IFo3

# Conception d'applications Web



## INTRODUCTION

LIONEL MÉDINI  
SEPTEMBRE 2023



# Plan du cours



- Partie 1
  - Présentation de l'UE
- Partie 2
  - Généralités sur le World Wide Web

# Objectif général de l'UE



- **Comment envisager cette UE ?**
  - Permettre aux utilisateurs d'utiliser
    - un Système d'Information (SI)
    - un ensemble de services applicatifs à travers leurs navigateurs Web
- **Débouchés**
  - Concepteur / développeur « fullstack »
  - Consultant, chef de projet...

# Présentation générale de l'UE



- Intitulé de l'UE
  - « Conception ... »
    - ✦ GL, méthodes de conception, patterns
    - ✦ Et développement !
  - « ... d'applications... »
    - ✦ Ensemble de briques fonctionnelles d'un SI
    - ✦ Service rendu à l'utilisateur
  - « ... Web »
    - ✦ 2 définitions
      - Tout ce qui peut être transféré par le protocole HTTP(S)
        - Pages Web (HTML), autres types de documents (MIME)
      - Tout ce qui peut être traité dans un navigateur Web

# Présentation générale de l'UE



- Thématique générale « application Web »
  - Définition provisoire
    - ✦ « Application dont l'interface se situe dans un navigateur »
    - ✦ Sera précisée tout au long du déroulement de l'UE
  - Large éventail
    - ✦ De solutions techniques (sockets, serveurs web, services web...)
    - ✦ De paradigmes de programmation (POO, architectures distribuées ou orientées services...)
    - ✦ De niveaux de complexité (échanges de données, de documents, appels de méthodes distantes, chorégraphie de services...)
- ➔ Aperçu des outils existants forcément limité

# Positionnement de l'UE



- **Prérequis**

- Conception de pages web

- ✦ Description et mise en forme de pages Web : (X)HTML, CSS
- ✦ Structuration de données :
  - XML, DTD, XML-Schema, XSL
  - JSON, JSON-Schema

- Scripting

- ✦ Côté serveur : PHP (ou similaire)
- ✦ Côté client : JavaScript

- Autres

- ✦ Algorithmique et programmation (C, Java)
- ✦ Génie Logiciel (design patterns)
- ✦ Réseaux (couches « basses », sockets...)
- ✦ Persistance (bases de données)

# Positionnement de l'UE



- **Autres UE en parallèle**

- M1IFo5 : Réseaux
- M1IFo1 : POO
- M1IFo1 : gestion de projet
- M1IF24 : BD NoSQL



Projet commun : multimif  
(non alternants)

- **UE dépendant de celle-ci**

- M1IF13 : Programmation Web avancée et mobile
- M2 TIW : Intergiciels et Services
- M2 TIW : Technologies Web synchrones et multi-dispositifs
- M2 SRS + TIW : Cloud, Stockage et Virtualisation
- M2 SRIV : Applications de l'Internet et administration réseau

# Objectifs pédagogiques de l'UE



- **Approche DevOps**
  - Installation d'une « stack » complète
  - Déploiement / intégration continus
  - Gestion du « run » (performance)
- **Paradigmes de programmation**
  - Programmation déclarative (HTML, XSL, JSP...)
  - Programmation événementielle (JavaScript)
  - Mécanismes de scripting (côtés serveur et client)
- **Démarches et outils de conception**
  - Les mêmes qu'en M1IFO1 (modélisation, méthodes de GL)
  - Couche persistance non abordée



# Programme de l'UE



- Technologies côté serveur
  - Utilisation d'un serveur Web
    - ✦ Configuration et fonctionnement
    - ✦ Protocole HTTP
    - ✦ Sécurisation (HTTPS)
  - Programmation côté serveur
    - ✦ POO (Servlets, filtres)
    - ✦ Scripting (JSP)
  - Initiation aux services Web
    - ✦ REpresentational State Transfer (REST)

➔ Web APIs

# Programme de l'UE



- Technologies côté client
  - Rappels JavaScript et scripting côté client
    - ✦ « Dynamisation » des pages statiques
  - Mécanismes de requêtes asynchrones
    - ✦ Document Object Model (DOM)
    - ✦ Asynchronous Javascript And XML (AJAX)
    - ✦ Fetch API
  - Templating
    - ✦ Mustache
- Performance Web

# Remarque



Les technologies liées au Web sont nombreuses et très riches

- Cette UE est
    - un aperçu des types d'outils existants et de leur fonctionnement
    - une description détaillée d'une partie de ces outils
      - ✦ Langages dédiés au Web
      - ✦ Programmation côté serveur et côté client
      - ✦ Outils de conception d'applications Web complexes
  - Cette UE n'est PAS
    - une présentation exhaustive de tous les outils existants
    - une référence complète des outils présentés
- ➔ N'hésitez pas à aller voir les pointeurs (références bibliographiques) pour comprendre le cours et réaliser les TP

# Organisation pratique de l'UE



- 30 h de présentiel
    - 10 x 1h30 de CM
    - 10 x 1h30 de TP
  - Évaluation
    - TPs : Code + analyse automatique + compte-rendus + soutenances
    - CC final (~examen) : 2<sup>ème</sup> session en cours de 2<sup>nd</sup> semestre
    - Coefficients : **66% TP** – 33% Examen
- ➔ Il est impératif de travailler en dehors des cours et TP

**Tout TP rendu en plusieurs exemplaires sera noté  
0 pour tous les groupes l'ayant rendu**

# Organisation pratique de l'UE



- Répartition en TP
  - Votre groupe de TP est le même que pour M1IF24
  - Intervenants professionnels
    - ✦ Pratiquent les technos
    - ✦ Sont aussi là pour recruter...
    - ✦ Chaque intervenant passe dans un maximum de groupes de TP
      - ➔ Vous devez
        - Être présents aux TPs
        - Ne pas changer de groupe  
(*et avoir un binôme dans le même groupe*)

# Organisation pratique de l'UE



- Emplois du temps de TP
  - Les intervenants extérieurs ont des contraintes professionnelles
    - Emplois du temps qui « sortent » des créneaux habituels
    - « Tetris » avec d'autres UEs (M1IF01, M1IF24)
    - Récupération des créneaux libres
  - Pour simplifier : des **calendriers Google** (à jour)
    - ✦ [Planning des CM et de l'examen](#)
    - ✦ [Planning des TPs du groupe A](#)
    - ✦ [Planning des TPs du groupe B](#)
    - ✦ [Planning des TPs du groupe C](#)
    - ✦ [Planning des TPs du groupe D](#)
  - Certaines salles manquent encore (en attente)
  - **Ne venez qu'aux séances qui sont sur ces calendriers** (même si d'autres sont indiquées dans ADE)

# Méthode de travail



- Retours sur l'UE
  - UE de tronc commun, ambitieuse, mais nécessaire
    - ✦ Pas que pour « faire du Web » : GL, SI, DevOps...
  - **Beaucoup de travail en TP**
- Pour gagner du temps et être efficaces
  - Partagez-vous le travail (à l'intérieur d'un binôme ;-)
    - ✦ Git branches
  - Mettez en place une infra fonctionnelle **dès le début**
    - ✦ IDE, VM, CI/CD
  - Apprenez à bien utiliser votre IDE
    - ✦ Configuration (Maven, Git), raccourcis clavier, debugger, refactoring...
  - Si possible, utilisez votre propre machine...
- **Anticipez les deadlines**

# Important : à faire avant le début des TPs



- Installer et configurer votre environnement de travail
  - Dans cette UE, vous aurez besoin de :
    - ✦ Java : JDK  $\geq 17$
    - ✦ IDE : de votre choix...  
...mais vous avez accès à une licence JetBrains pour IntelliJ ([Free educational licence](#) avec votre adresse email Lyon1)
    - ✦ Serveur et moteur de servlets Java : [Apache tomcat](#)

**Réalisez la [procédure d'installation et de configuration](#)**



# Méthode de travail



- Utiliser le Chat

- Pour travailler en asynchrone

- 3 salons

- ✦ Général :

- <https://go.rocket.chat/invite?host=chat-info.univ-lyon1.fr&path=invite%2Fbt5Xdm>

- ✦ Tps :

- <https://go.rocket.chat/invite?host=chat-info.univ-lyon1.fr&path=invite%2FqYQnXA>

- ✦ Foire aux binômes :

- <https://go.rocket.chat/invite?host=chat-info.univ-lyon1.fr&path=invite%2FBQxsjA>

# Sources



- Reprise des supports de cours des années précédentes (Lyon 1)
  - de Sylvain Brandel, eux-mêmes issus de ceux d'Olivier Glück
  - de Pierre-Antoine Champin (IUT Lyon1)
- Ces supports s'appuient eux-mêmes sur ceux de
  - Fabrice Kordon, Isabelle Mounier, Christian Queinnec (PARIS 6), Dominique Bouillet (INT), Laurent Lefèvre (ENS LYON), Olivier Aubert (Univ. Nantes)...
- Autres sources
  - W3C (standards)
  - Autres organismes de standardisation
  - Sites et ouvrages cités en bibliographie
- Mais dans tous les cas...

**C'est sur le Web qu'on trouve le plus de choses sur le Web...**

# Bibliographie



- **Ouvrages**

- « Webmaster in a nutshell », S. Spainhour & R. Eckstein, O'Reilly
- « HTML et JavaScript », P. Chaléat et Daniel Charnay, Eyrolles
- « JavaScript, La référence », D. Flanagan, O'Reilly
- « Ajax en Pratique », D. Crane, E. Pascarello et D. James, CampusPress

- **Sites Web**

- <http://www.w3.org/>
- <http://w3schools.com/>
- <http://www.developpez.com/>
- <http://www.laltruiste.com/>
- <http://www.commentcamarche.net/>

# Plan de ce cours



- **Partie 1**
  - Présentation de l'UE
- **Partie 2**
  - Généralités sur le World Wide Web

# World Wide Web



- Principe original : accéder à des documents textuels
    - situés sur des machines accessibles par Internet
    - reliés entre eux par un mécanisme de lien « hypertexte »
  - Actuellement : servir des ressources
    - De différentes natures : texte, image, son, vidéo, contenu applicatif...
    - Hypermédia
    - Interactives
    - Permettant à l'utilisateur d'accéder à un service donné : rechercher de l'information, acheter un objet, accéder à ses mails, consulter ses comptes en banque...
- ➔ Nombreuses évolutions techniques

# Rappel : Internet



- Un réseau de réseaux interconnectés (d'où le nom)
- Un ensemble de matériels, logiciels et protocoles (notamment IP)
- Un ensemble de **services**
  - **Application** qui utilise un **protocole** et un numéro de **port**
  - e-mail, transfert de fichiers, connexion à distance, WWW...
- Une somme « d'inventions » qui s'accumulent
  - Mécanismes réseau de base (TCP/IP)
  - Nommage et adressage des ressources (DNS, URL)
  - Outils et protocoles spécialisés
  - Langages d'échange d'informations standardisés (HTML, XML...)

# Bref historique d'Internet



- **1959-1968 : Programme ARPA**
  - le ministère américain de la défense lance un réseau capable de supporter les conséquences d'un conflit nucléaire
- **1969 : ARPANET, l'ancêtre d'Internet**
  - les universités américaines s'équipent de gros ordinateurs et se connectent au réseau ARPANET
- **1970-1982 : Ouverture sur le monde**
  - premières connexions avec la Norvège et Londres
- **1983 : Naissance d'Internet**
  - protocole TCP/IP : tous les réseaux s'interconnectent
  - les militaires quittent le navire
- **1986 : Les autoroutes de l'Information**
  - la National Science Fondation déploie des super-ordinateurs pour augmenter le débit d'Internet
- **1987-1992 : Les années d'expansion**
  - les fournisseurs d'accès apparaissent
  - les entreprises privées se connectent au réseau
- **1993-2003 : L'explosion d'Internet**
  - ouverture au grand public
  - avènement du WEB et du courrier électronique

} marché considérable

# Genèse du Web : la notion d'hypertexte



- **Principe**

- S'abstraire de l'aspect linéaire du document textuel
- Mécanisme intellectuel permettant le cheminement d'une information à une autre → navigation, butinage, transclusion

- **Historique**

- 1945 : invention de la notion d'hypertexte
  - ✦ Vannevar Bush, As We may think, *Atlantic Monthly*, 1945
- 1965 : invention du terme d'hypertexte
  - ✦ Ted Nelson, projet Xanadu
- Années 1960 : premier système hypertexte fonctionnel
  - ✦ NLS (oNLine System), Douglas Englebart
- 1987-2004 : diffusion du logiciel HyperCard
  - ✦ Programme et environnement graphique de programmation, créé par Bill Atkinson pour Mac OS, livré avec les Mac
- 1987 : première conférence HyperText
  - ✦ Sponsorisée par l'ACM



# Naissance du Web (1989-1991)

- Mars 89 : projet de création d'un hypertexte documentaire distribué sur le réseau du CERN
  - Origine : Tim Berners-Lee, puis Robert Cailliau (1990)
  - Choix des technologies TCP/IP et ouverture de la première connexion du CERN avec Internet
  - Mise au point des **3 technologies de base du Web : URL, HTML et HTTP**
- Septembre 90 : 1er site Web fonctionnel (mode texte)
  - 1<sup>er</sup> serveur Web : nxoc01.cern.ch
  - 1<sup>er</sup> navigateur Web : WorlWideWeb (rebaptisé plus tard Nexus), développé en Objective C
- Août 1991 : publication du projet WorldWideWeb dans un message sur UseNet
- Décembre 91 : démonstration publique à la conférence Hypertext'91 (San Antonio)



Le premier serveur Web,  
un NeXT Cube  
(source : Wikipédia)

# Historique du Web



- 1993 : Mosaic : premier navigateur « grand public »
  - Marc Andreessen, NCSA : plateformes X, puis Mac et Windows
  - affichage d'images (GIF et XBM) dans les pages Web
  - prise en charge de formulaires interactifs
- 1994 : création du W3C
  - à l'initiative du CERN (Genève) et du MIT (Boston)
  - président : Tim Berners-Lee
  - but : standardisation et développement du Web
- 1994 : Apparition des navigateurs privés
  - M. Andreessen crée Netscape Communications Corp.
- 1995 : Microsoft lance la « guerre des navigateurs »
  - Apparition d'Internet Explorer pour Windows 95
- 1995 : Altavista : premier « gros » moteur de recherche
- 1996 : Navigateur Opera
- 1998 : Apparition de Google
- 2003 : Apple lance Safari
- 2004 : Première version de Mozilla Firefox
- 2004 : première conférence « Web 2.0 »
- 2008 : Google lance Google Chrome
- Actuellement : explosion du Web mobile
- En cours de développement : Web sémantique, de données, des objets...

## Nombre de sites référencés

**1990** : 1 (CERN)

**1991** : premier site hors d'Europe (SLAC, Stanford)

**1992** : 26

**Juin 1993** : 130

**Juin 1994** : 2738

**Juin 1995** : 23 500

**Janvier 1996** : ~ 100 000

**Avril 1997** : > 1 million

**Février 2000** : > 11 millions

**Février 2007** : > 100 millions

**Février 2009** : ~ 216 millions

# Aspects techniques du Web



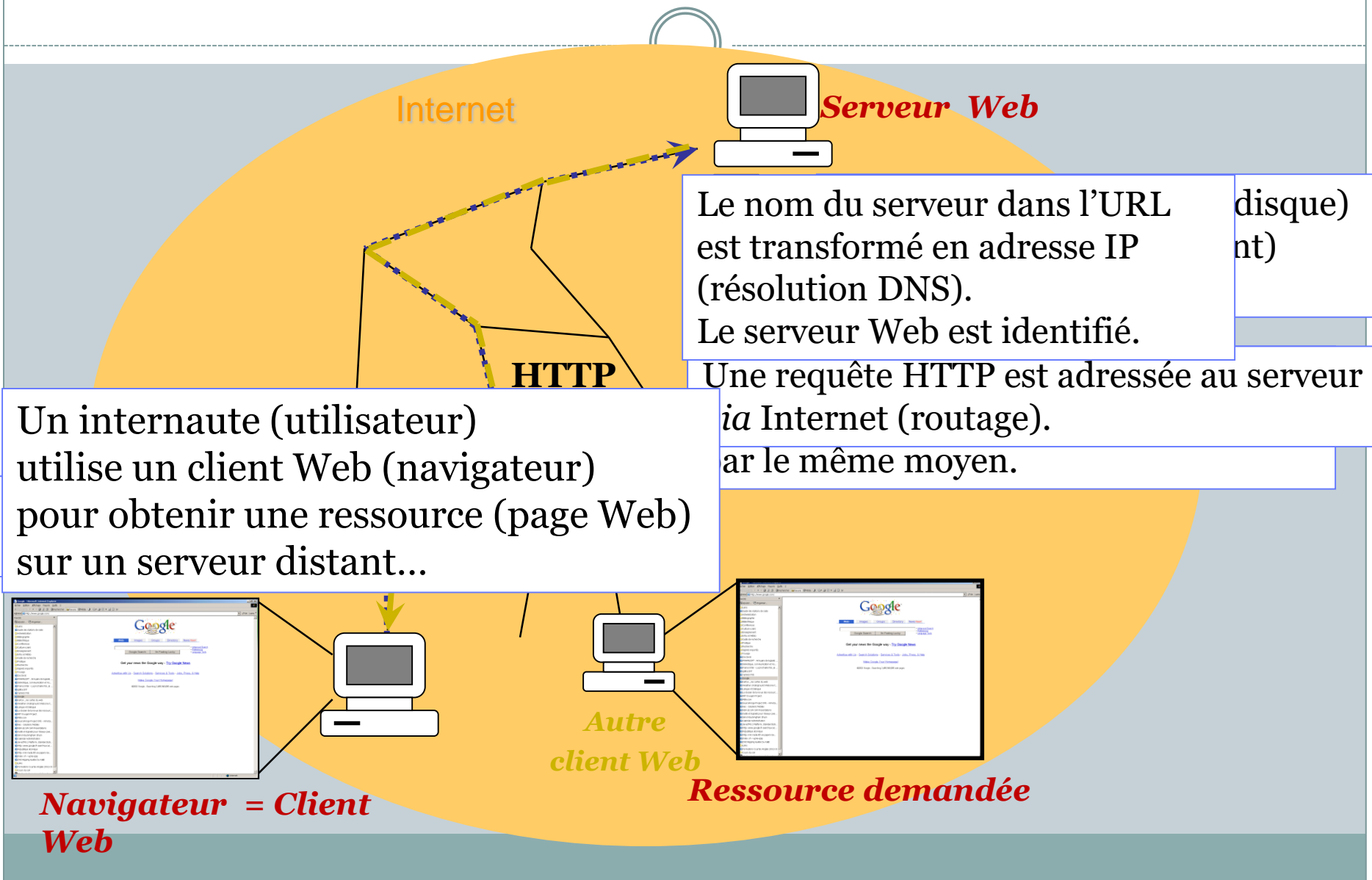
- Les 3 mécanismes de base du Web
  - URL
    - ✦ Le Web permet d'accéder à un ensemble de **ressources**
    - ✦ Le mécanisme de localisation peut faire appel au protocole DNS
  - HTTP
    - ✦ Protocole de niveau applicatif
    - ✦ Paradigme client-serveur
    - ✦ Protocole sans état (pas de « mémoire » des transactions précédentes)
  - HTML
    - ✦ Langage de description de « pages Web »
      - Texte, images et autres objets
      - Liens hypermédias entre les pages
    - ✦ Programmation déclarative

# Aspects techniques du Web



- Les forces en présence
  - Côté client : l'utilisateur utilise un navigateur
    - ✦ Client HTTP
    - ✦ Interprète les pages Web et les affiche à l'utilisateur
    - ✦ Peut effectuer des traitements plus complexes (plugins, moteur de scripts...)
  - Côté serveur : le Webmaster gère un serveur Web
    - ✦ Attend les requêtes HTTP et y répond
      - En renvoyant des ressources dont il dispose
      - En interrogeant plusieurs modules (sécurité, scripting, redirection...)
      - En interrogeant d'autres outils pour les traitements complexes
    - ✦ On parle aussi de serveur Web pour désigner la machine qui héberge le programme serveur (abus de langage)

# Fonctionnement du Web en un schéma



# Popularité du Web



- **Pour l'internaute**

- Accessibilité « world-wide »
- Interfaces graphiques conviviales
- Interactivité et richesse des services
- Grande quantité d'informations disponibles (Web « 1.0 »)
- Possibilité de contribuer en tant qu'utilisateur (Web « 2.0 »)

- **Pour le développeur**

- Simplicité des développements (technologies de base...)
- Liens avec les outils applicatifs installés sur le serveur (shell, Perl, Java...)
- Indépendance par rapport aux plateformes des clients

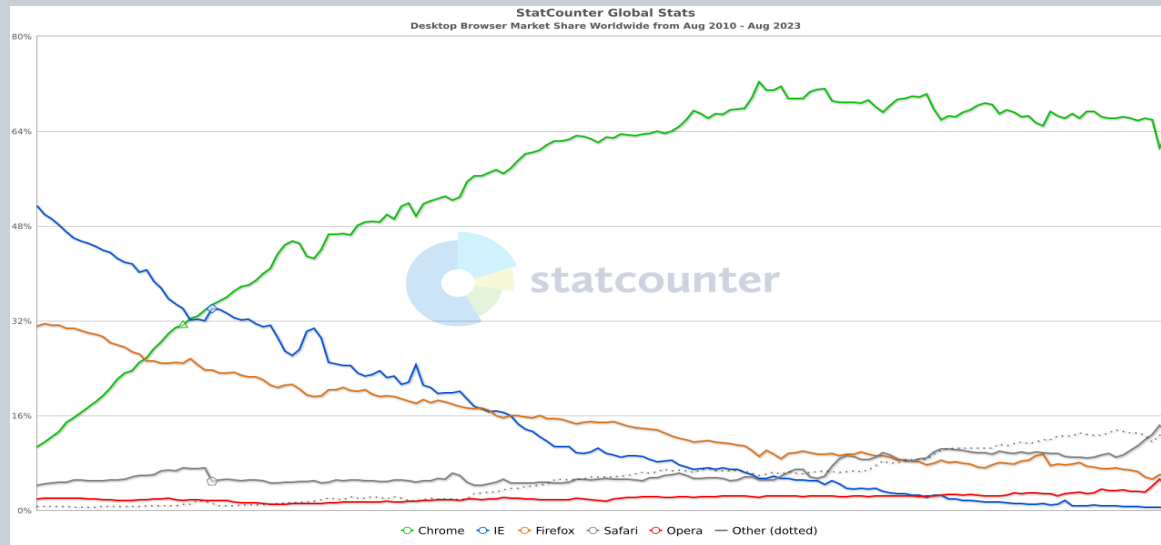
# Répartition des clients (sept. 2023)



## ● Fixes

Browser	StatCounter <sup>[14]</sup> February 2023	StatCounter <sup>[14]</sup> February 2022	NetMarketShare <sup>[15]</sup> October 2021	Wikimedia <sup>[16]</sup> October 2021
Chrome	65.76%	62.78%	66.64%	52.5%
Safari	18.84%	19.30%	13.92%	23.9%
Edge	4.28%	4.06%	4.55%	3.0%
Firefox	2.93%	4.21%	2.18%	4.4%
Samsung Internet	2.59%	2.77%	3.04%	2.2%
Opera	2.26%	2.26%	3.02%	1.0%
Others	3.34%	4.62%	6.65%	13.0%

✧ Source : [http://en.wikipedia.org/wiki/Usage\\_share\\_of\\_web\\_browsers](http://en.wikipedia.org/wiki/Usage_share_of_web_browsers)

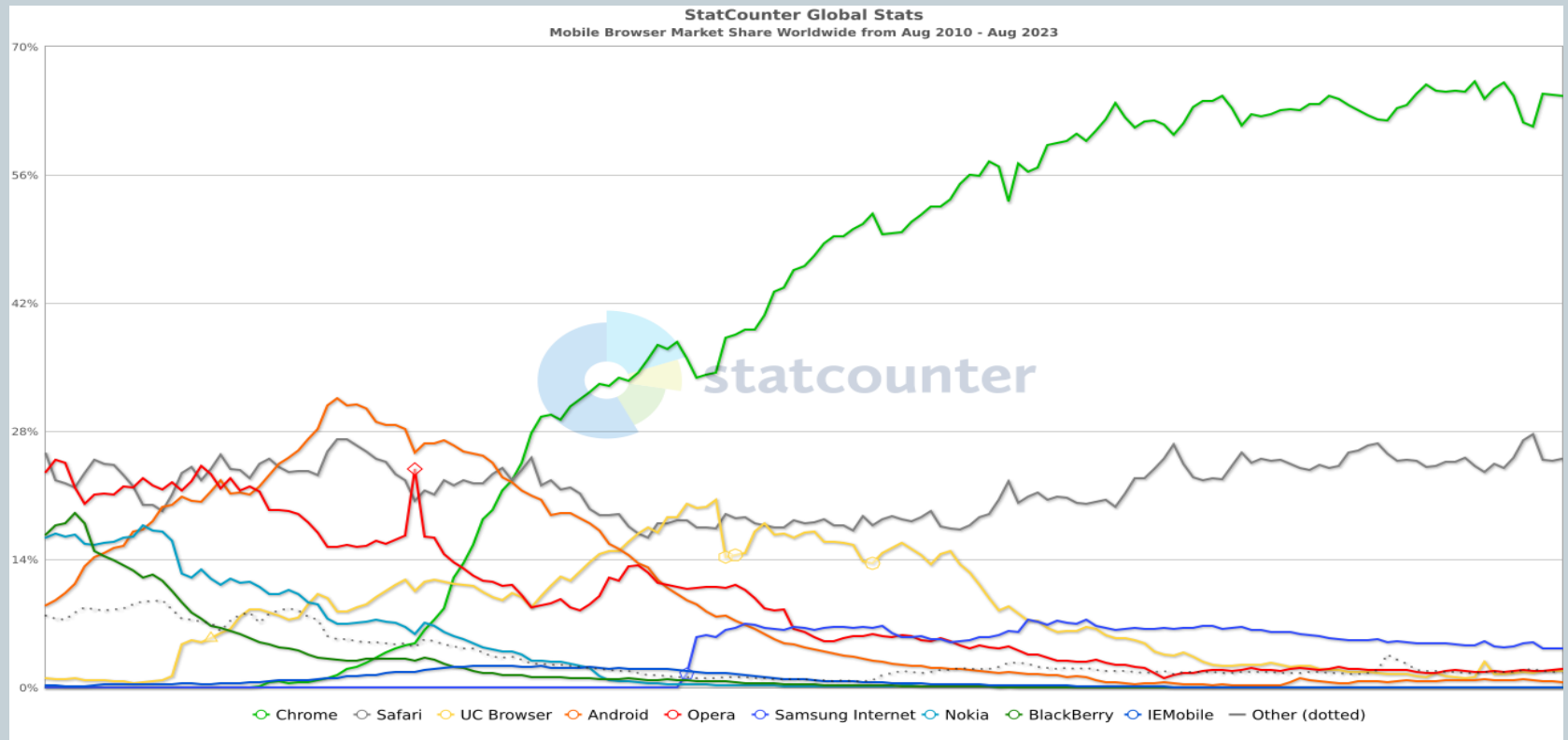


✧ Source : <https://gs.statcounter.com/browser-market-share/desktop/worldwide#monthly-201008-202308>

# Répartition des clients (sept. 2023)



## ● Mobiles



✦ Source :

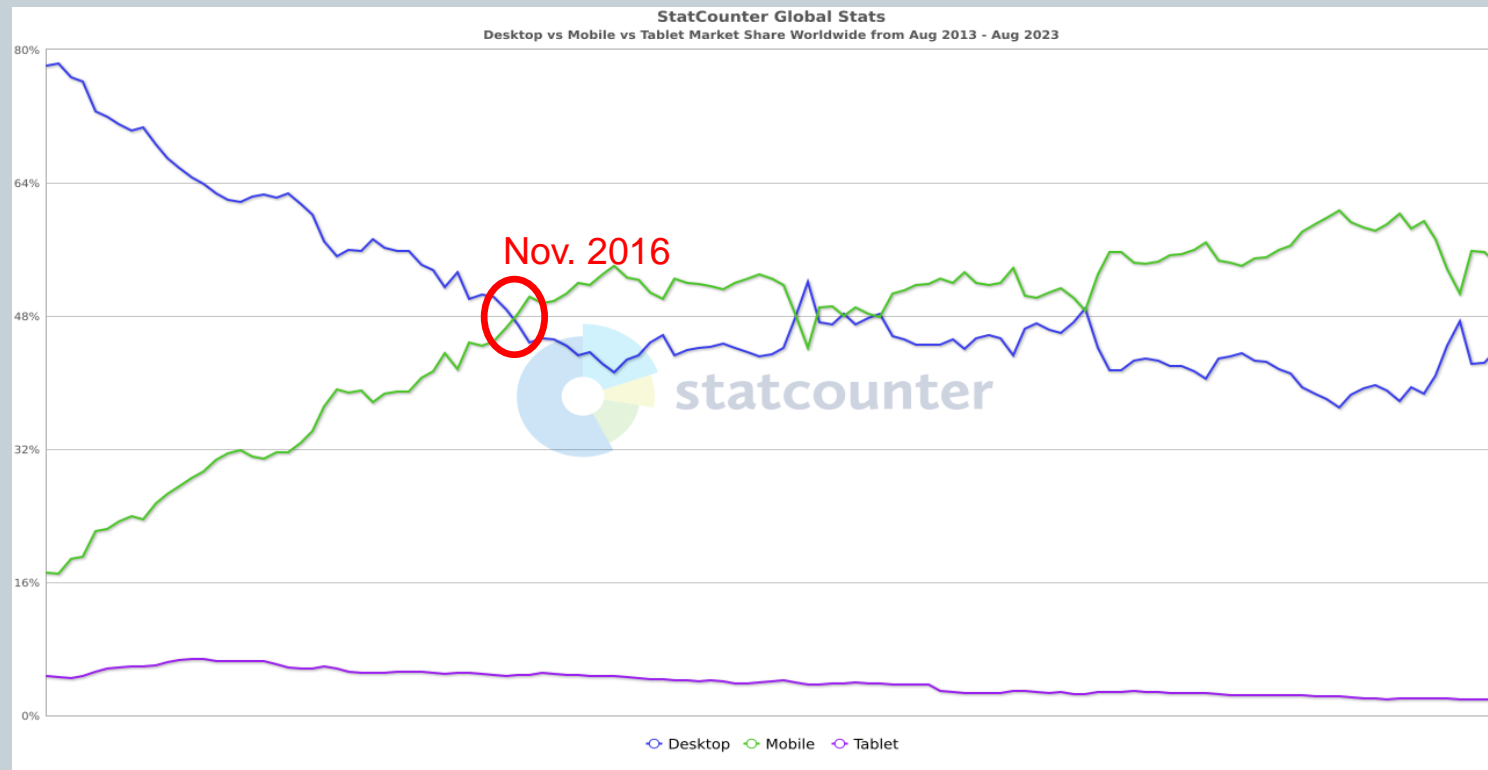
<https://gs.statcounter.com/browser-market-share/mobile/worldwide#monthly-201008-202308>



# Répartition des clients (sept. 2023)



- Tous dispositifs



✦ Source :

<https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet#monthly-201308-202308>

# L'approche DevOps

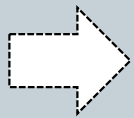
34

- Contraction de *Developers* et *Operations teams*
  - ➔ Un mouvement pour rapprocher ces deux métiers combinant une culture, des pratiques et des outils
- Objectifs
  - Augmenter la collaboration entre ces équipes
  - Améliorer la qualité de la solution produite
  - Augmenter la fréquence des mises à jour
  - Automatiser l'infrastructure, les workflows, les tests...
  - Mesurer les performances
  - ➔ Répondre plus rapidement à l'évolution des besoins métier

# L'approche DevOps

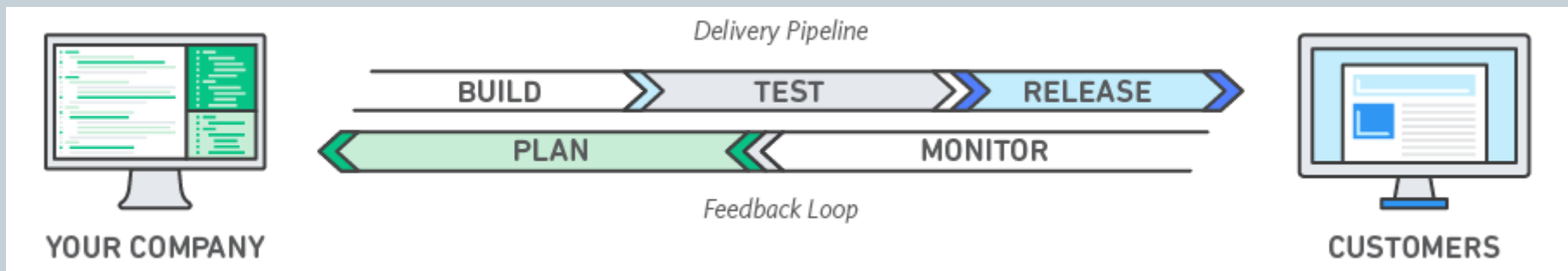
35

- Contraction de *Developers* et *Operations teams*



Un mouvement pour rapprocher ces deux métiers, combinant une culture, des pratiques et des outils

- Workflow



○ Source : [AWS](#)

# L'approche DevOps

36

- Culture

- Briser les silos entre les équipes

- ✦ Chaque équipe est consciente des préoccupations des autres
- ✦ Les équipes échangent des informations et se facilitent mutuellement le travail

- Augmenter les compétences des équipes

- ✦ Chaque personne connaît sa place dans un workflow global
- ✦ Chaque personne est capable d'améliorer son travail grâce à sa connaissance des entrants et des sortants des autres tâches

→ Employés plus impliqués, mieux valorisés

→ Gains de productivité (facilite l'agilité)

→ Accélération du « time-to-market »

→ Amélioration de la satisfaction du client

# L'approche DevOps

37

- Pratiques

- Déploiement continu

- ✦ Modulariser le code
    - ✦ Réduire le temps de déploiement

- Intégration continue

- ✦ Tester et merger fréquemment son code

- Livraison continue

- ✦ Déployer automatiquement chaque release sur un environnement de test similaire à l'environnement de production
    - ✦ Augmenter la fréquence de déploiement sur l'environnement de production

# L'approche DevOps

38

- **Pratiques**

- **Microservices**

- ✦ Facilite la modularisation du code et du déploiement

- **« Infrastructure as code »**

- ✦ Environnements de développement et de production identiques
    - ✦ Gestion de la configuration des environnements par du code
      - Opérations de configuration et de déploiement définies programmatiquement
      - On utilise des techniques de GL (versionning, patterns...) pour la gestion des infrastructures
    - ✦ Meilleure scalabilité des déploiements

# L'approche DevOps

39

- Pratiques

- Monitoring et logging

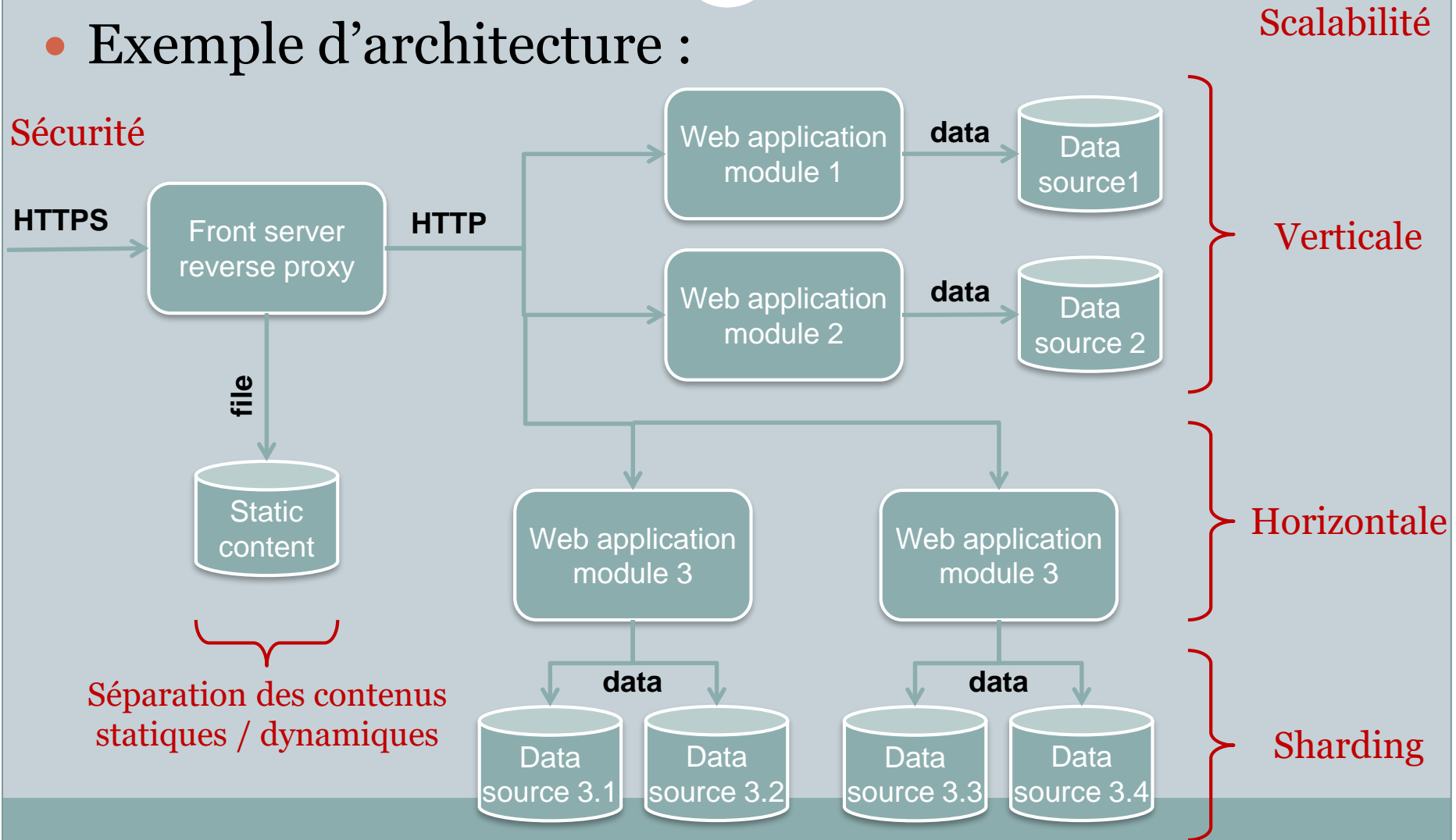
- ✦ En termes de : qualité, performance, évolution des besoins
- ✦ Permet d'analyser l'impact des déploiements
- ✦ Améliorer et optimiser le code en continu

- Communication et collaboration entre les équipes

- ✦ Unification des workflows de développement et de mise en production
- ✦ Mise en place de règles et documentation des processus
- ✦ Responsabilisation de toutes les équipes sur ces processus

# L'approche DevOps appliquée au Web

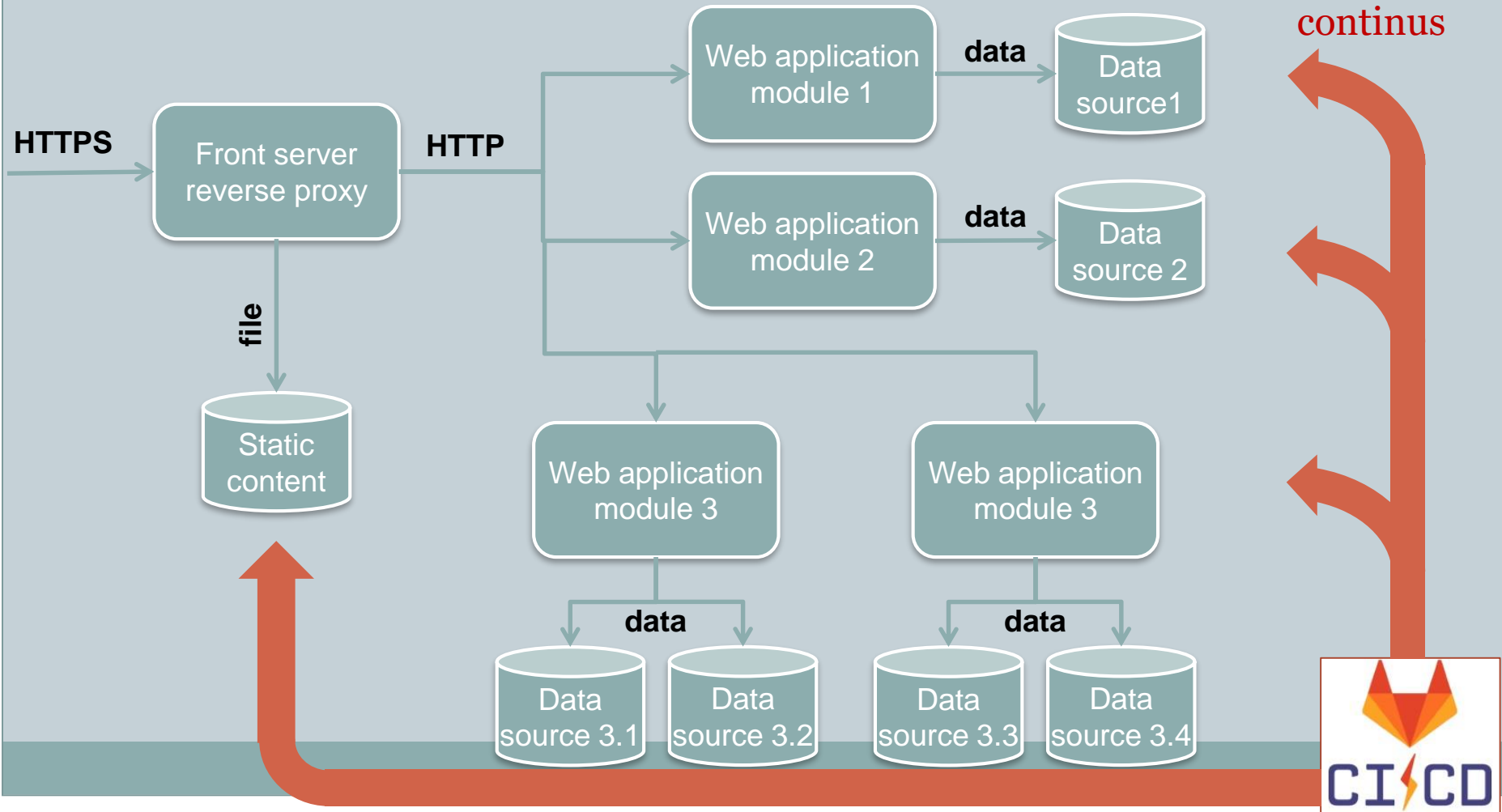
- Exemple d'architecture :





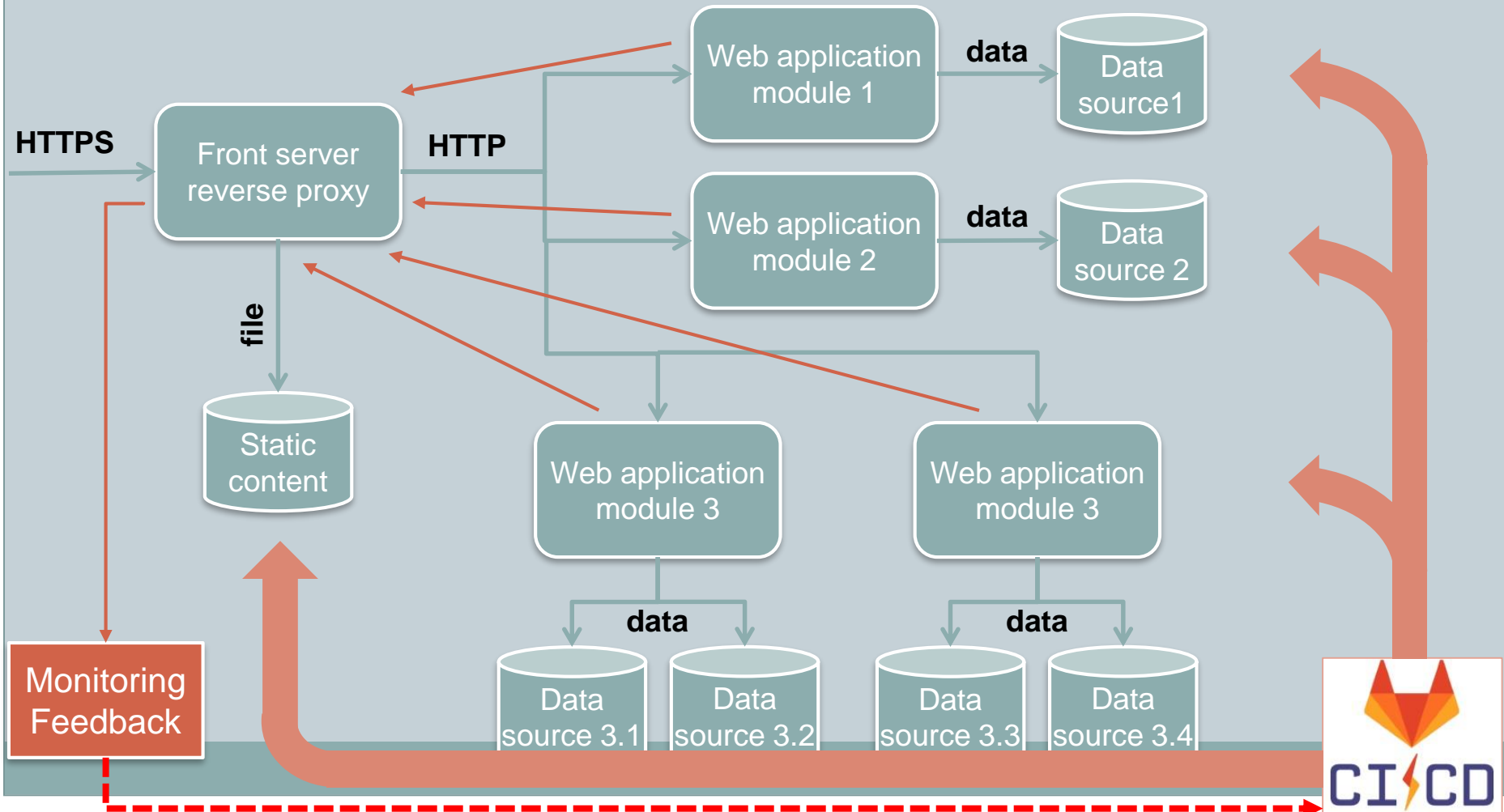
# L'approche DevOps appliquée au Web

- Exemple d'architecture :



# L'approche DevOps appliquée au Web

- Exemple d'architecture :



# L'approche DevOps

43

- (quelques) Outils
  - Gestion du code / documentation
    - ✦ [SVN](#), [Git](#), [Mercurial](#)
  - Issues / communication
    - ✦ [Redmine](#), [Mantis](#), [Bugzilla](#)
  - Tests
    - ✦ ..., [Chaos Monkey](#)
  - Packaging : Binary repository manager
    - ✦ [GitHub package registry](#), [Sonatype Nexus](#)
  - Déploiement / Intégration continue
    - ✦ [Jenkins](#)
  - Livraison : Application-release automation (ARA)
    - ✦ [Chef](#), [Puppet](#), [SaltStack](#), [Ansible](#)...
  - Monitoring / logging / mesure de perf : Application Performance Management (APM)
    - ✦ [Stackify Retrace](#), [New Relic](#)

# Les événements liés au Web



- À Lyon
  - Groupes d'utilisateurs
    - ✦ LyonJS, JUG, CARA, LYAUG, apéro PHP, CocoaHeads...
    - ✦ Référence : Lyon Tech Hub (calendrier Google)
- À l'UCBL
  - Fonds de Soutien et de Développement des Initiatives Etudiantes

<http://etu.univ-lyon1.fr/vie-etudiante/financer-votre-projet/fsdie-mode-d-emploi-506763.kjsp?RH=1197015715894>
  - Cellule concours :

<http://fst-informatique.univ-lyon1.fr/formation/cellule-concours/>

# Aucun rapport...



- Il reste de la place en alternance
  - Réunion d'info **jeudi 21 septembre 14h00 en TD12**

# Page d'accueil de l'UE



<https://perso.univ-lyon1.fr/lionel.medini/enseignement/#md=M1IF03/index>