Projet de Développement Logiciel

https://github.com/acherm/teaching/tree/master/PDL/

Master 1 - MIAGE

Mathieu Acher

Maître de Conférences

mathieu.acher@irisa.fr

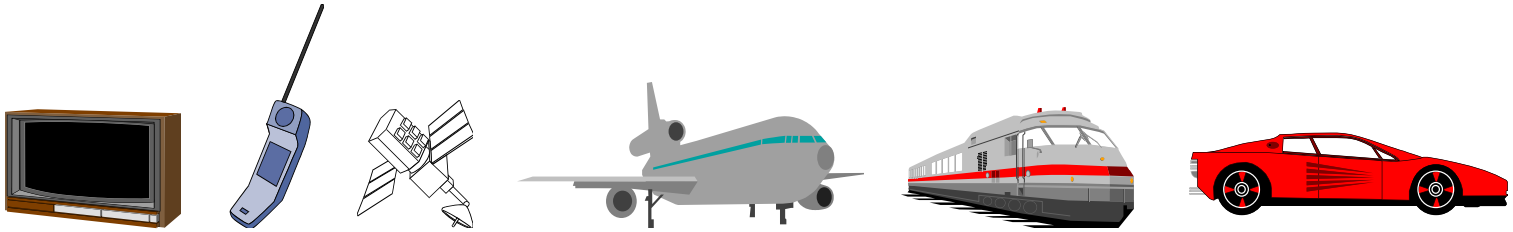
UNIVERSITÉ DE RENNES 1

istic Informatique Électronique

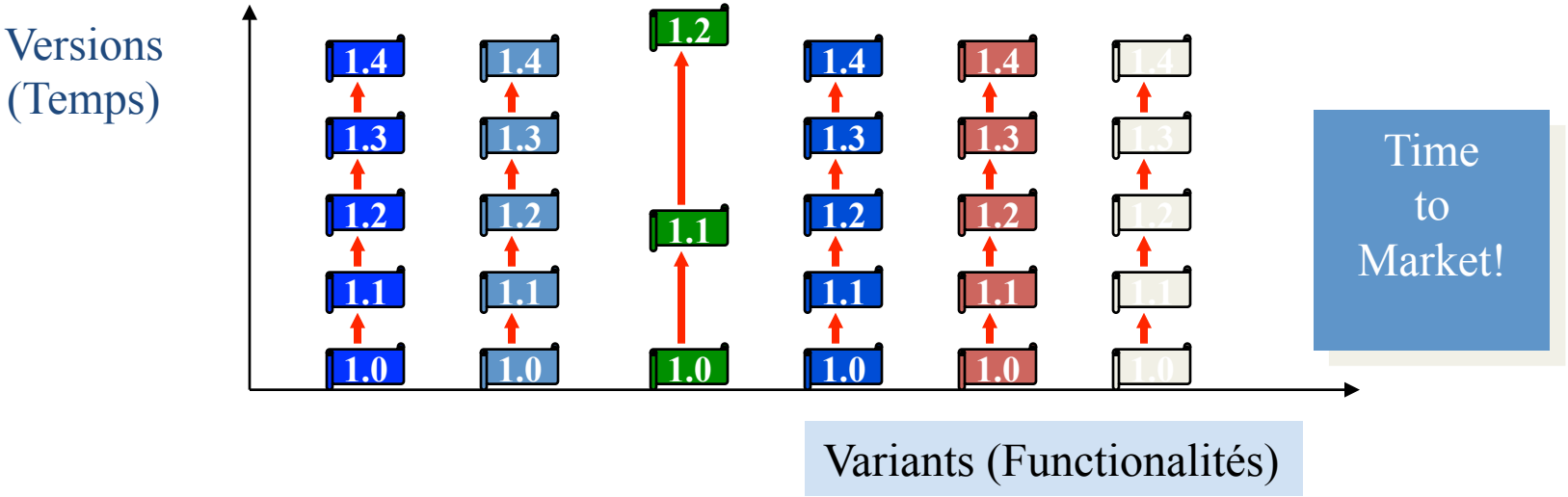
PDL: le “projet”

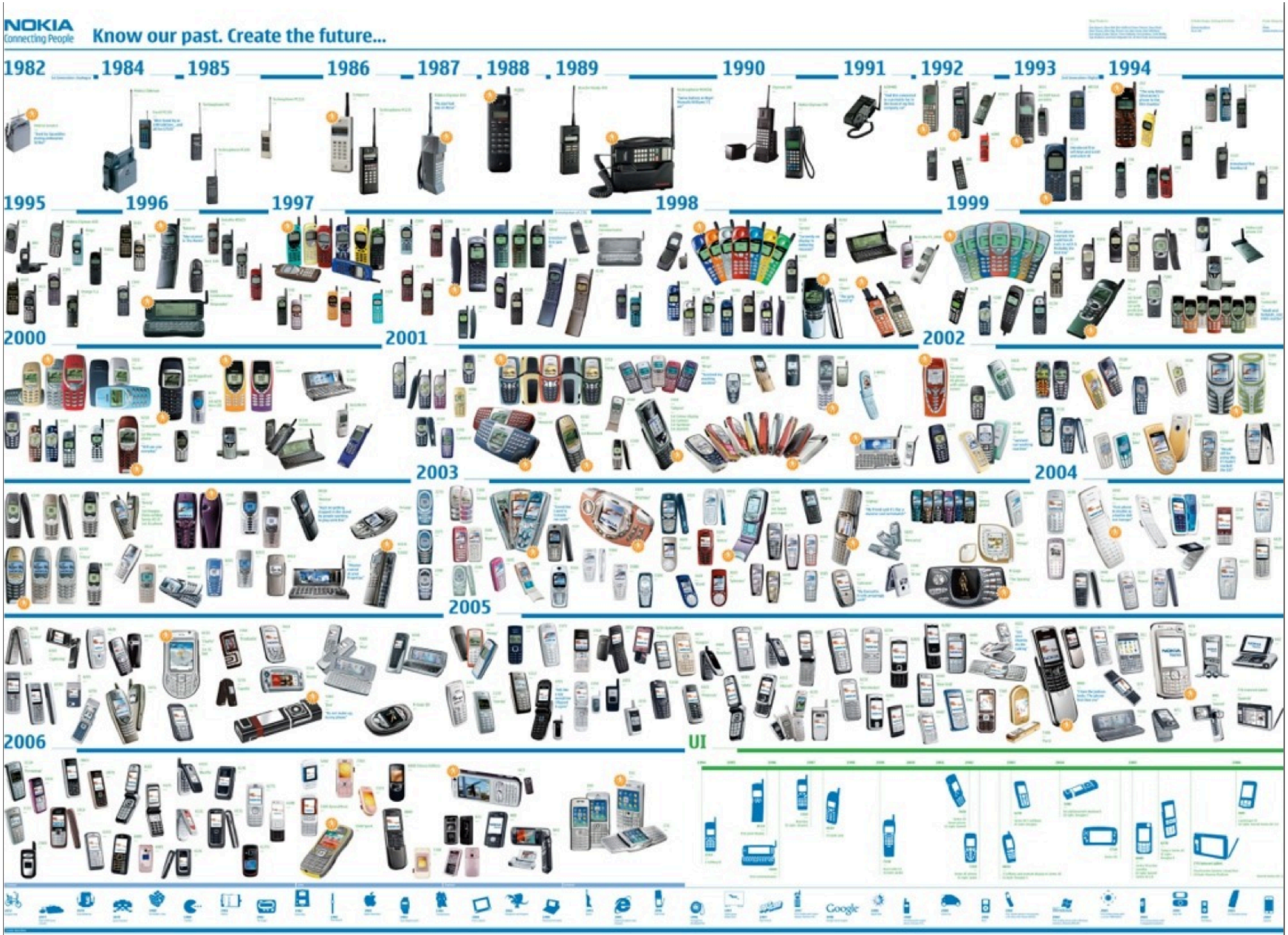
- Développement d’un système de chronométrage
  - Client: Charles Quéguiner
  - Enregistrer, gérer et imprimer, les chronométrages de courses de voitures et de séances d'essais d’une course automobile (e.g., 24 heure du Mans)
- Pratique et (re-)visite de votre cursus
  - UML, Programmation OO, testing, design patterns, etc
- Une **expérience** de la difficulté du développement logiciel
  - indispensable pour votre future vie professionnelle

Ingénierie du logiciel



- De plus en plus complexe
  - Systèmes distribués
  - Qualité de service: performance, sécurité, sûreté, utilisabilité, etc.
- Explosion des fonctionnalités
  - Lignes de produits (espace/temps)




NOKIA
Connecting People
Know our past. Create the future...
1982
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
UI
Google

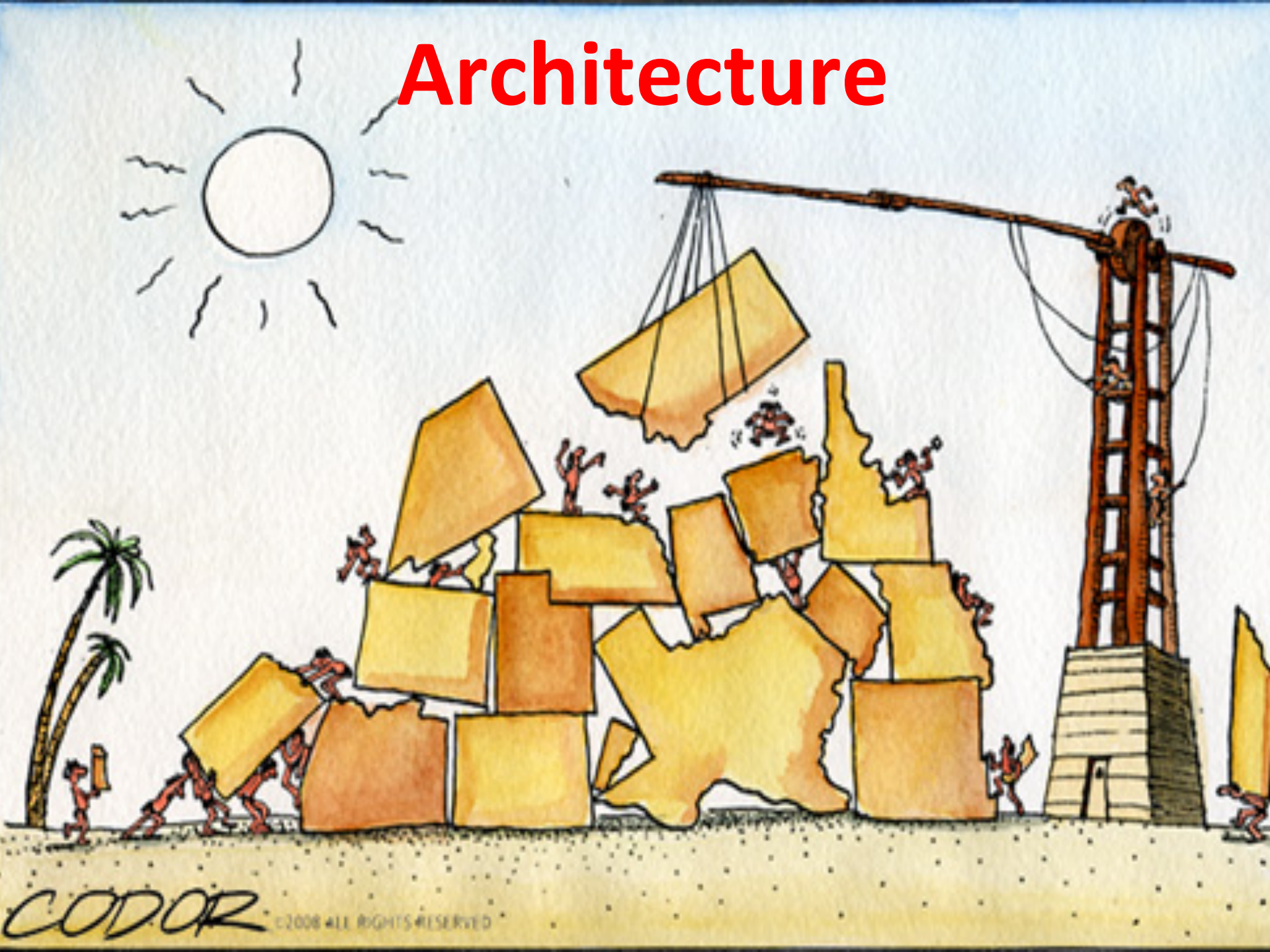

4°

Travail d’équipe

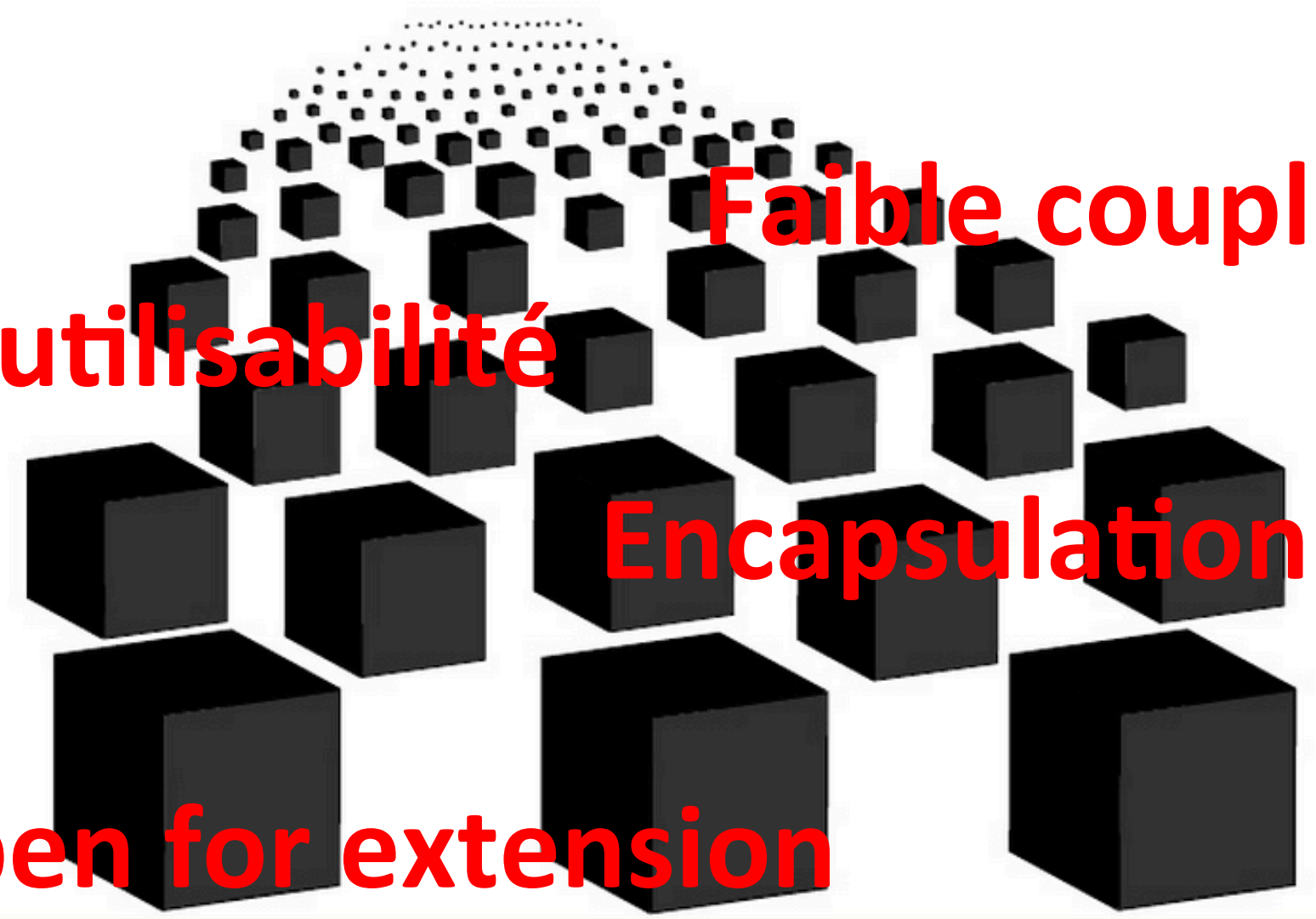
Travail d'équipe

- Organisation
  - Partage des tâches
  - Planification
  - Communication
- Code idéalement…
  - Bien conçu, modulaire, documenté
  - Maintenable, compréhensible
- Outils
  - Collaboratifs (e.g., système de versions)

Architecture



Idéalement:
« modular black boxes »



Faible couplage

Réutilisabilité

Encapsulation

Open for extension

Testing


I don't make mistakes
Microsoft Visual Studio
Package Load Failure
Package 'ArtOfTest.WebAiiVSIP.WebAiiVSIP, ArtOfTest.WebAiiVSIP, Version=1.0.0.0, Culture=neutral, PublicKeyToken=14176981d1873d86' has failed to load properly (GUID = {D5EDEC6D-8025-40CF-987C-B8EABA2C7451}). Please contact package vendor for assistance. Application restart is recommended, due to possible environment corruption. Would you like to disable loading this package in the future? You may use 'devenv /resetskippkgs' to re-enable package loading.
Yes
No

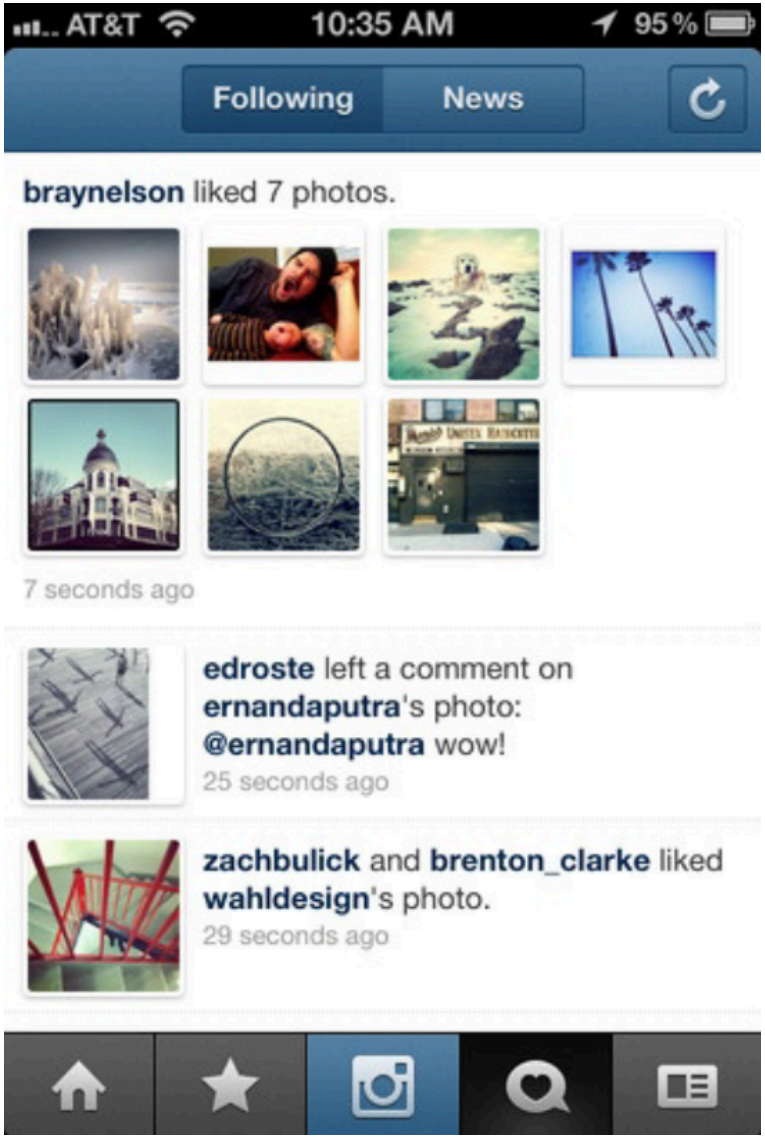
Djikstra


Program testing can be used
to show the presence of bugs, but
never to show their absence!

Software Integration

Développement Logiciel

Instagram Story



Instagram Story

« Instagram is an app that **only took 8 weeks** to build and ship, but was a product of over a year of work. »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« While I was there working in marketing, I started doing more and more engineering at night on simple ideas that helped me learn how to program **(I don't have any formal CS degree or training) »**

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« We spent 1 week prototyping a version that focused solely on photos.
It was pretty awful. So we went back to creating a native version of Burbn. We actually got an entire version of Burbn done as an iPhone app, but it felt cluttered, and overrun with features. It was really difficult to decide to start from scratch, but we went out on a limb, and basically cut everything in the Burbn app except for its photo, comment, and like capabilities. What remained was Instagram. »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« So 8 weeks later, we gave it to our friends, beta tested, bug fixed, etc. and this Monday we decided it was ready to ship. »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« Who is responsible for Instagram's UI design?

For better or for worse, I've done most of the pixel pushing in our app. ;) »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

- 30+ millions d'utilisateur en 2 ans
- 25k inscriptions le premier jour
  - « best & worst day of our lives so far »
  - « favicon » cause des milliers d'erreurs 404
    - « 404-ing on Django, causing tons of errors »
- Un seul serveur au lancement
  - Moins puissant qu'un MacBook Pro
- La suite: passage à l'échelle, cloud (EC2) et ingénierie du logiciel

https://speakerdeck.com/mikeyk/scaling-instagram
http://zoompf.com/blog/2012/04/instagram-and-optimizing-favicons

Instagram Story

- Sur la trentaine de composants, 4 seulement ont été écrits à partir de zéro
  - App iOS, App Android, Android Push Notification Service et Redis Query analyzer

node2dm

Fabric

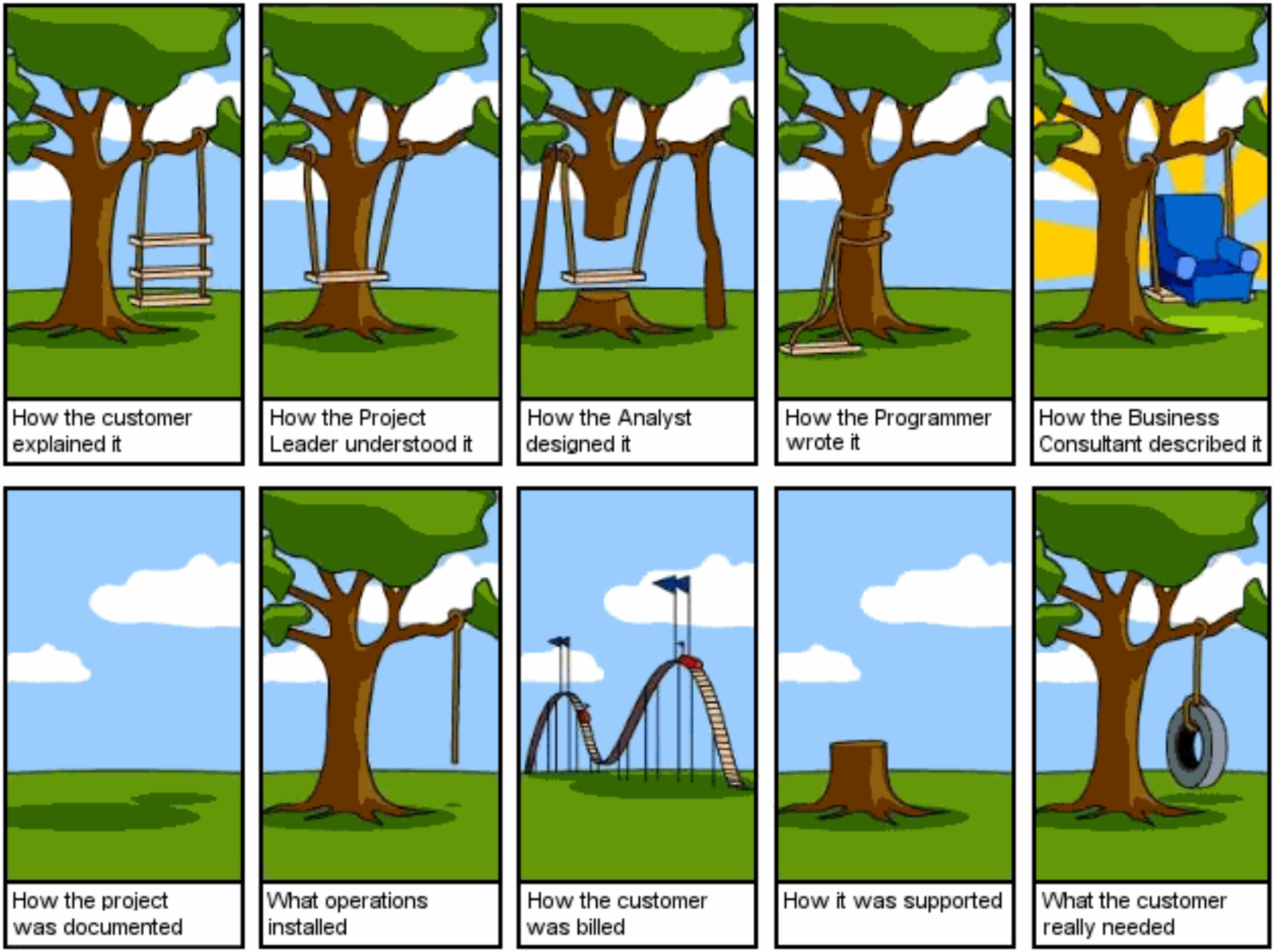
Instagram Story (key lessons)

- Sélection et intégration de multiples librairies

- Open source community
  - Apprendre, partager, demander, répondre, etc.
- Auto-apprentissage
  - « Product guys » sont maintenant à même de rivaliser…

- Agilité, développement incrémental

http://instagram-engineering.tumblr.com/


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
What operations installed
How the customer was billed
How it was supported
What the customer really needed

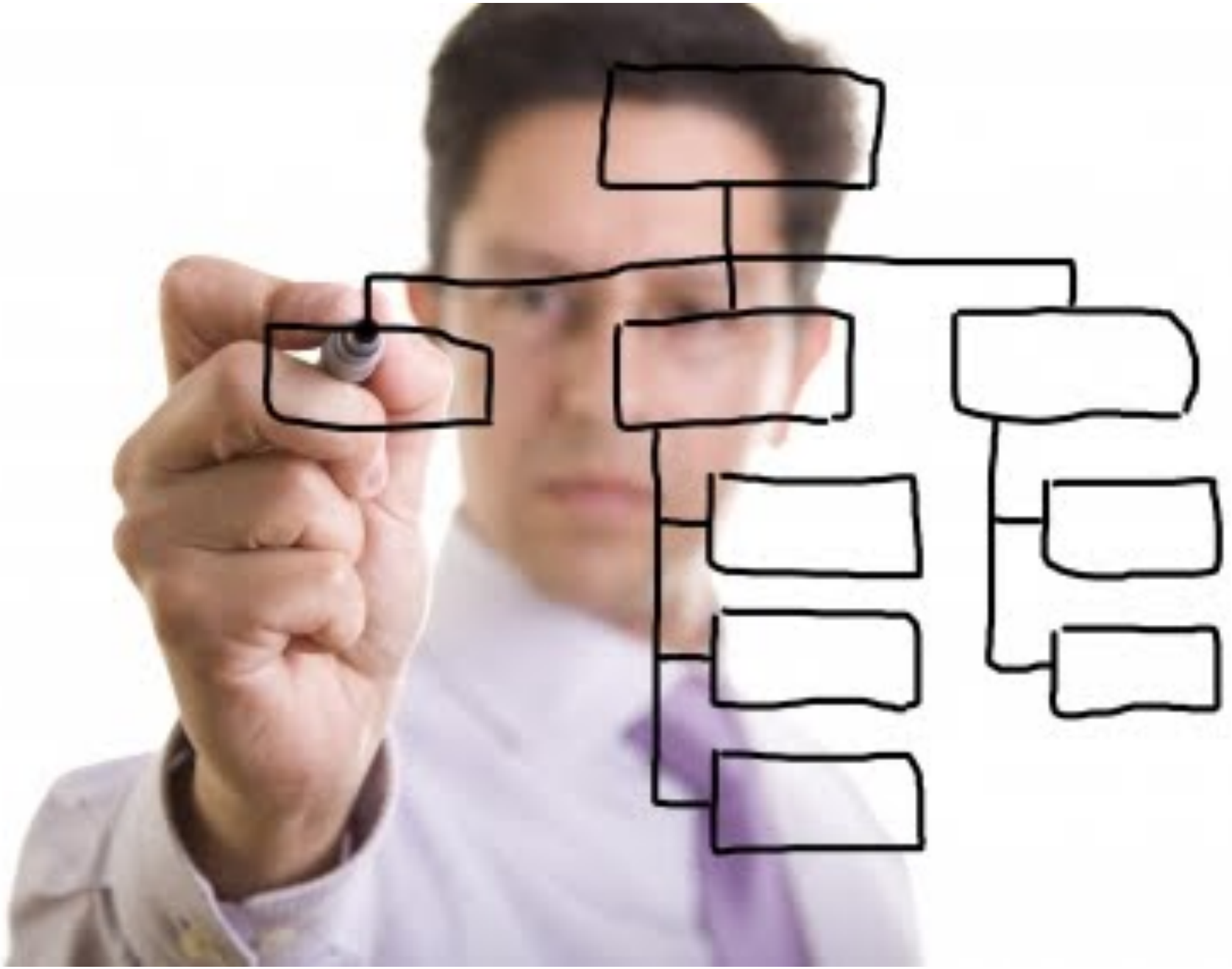
PDL: Objectifs

- Analyse, conception, réalisation, test, par la pratique
  - (Re)visite de votre cursus (UML, Programmation OO, etc.)

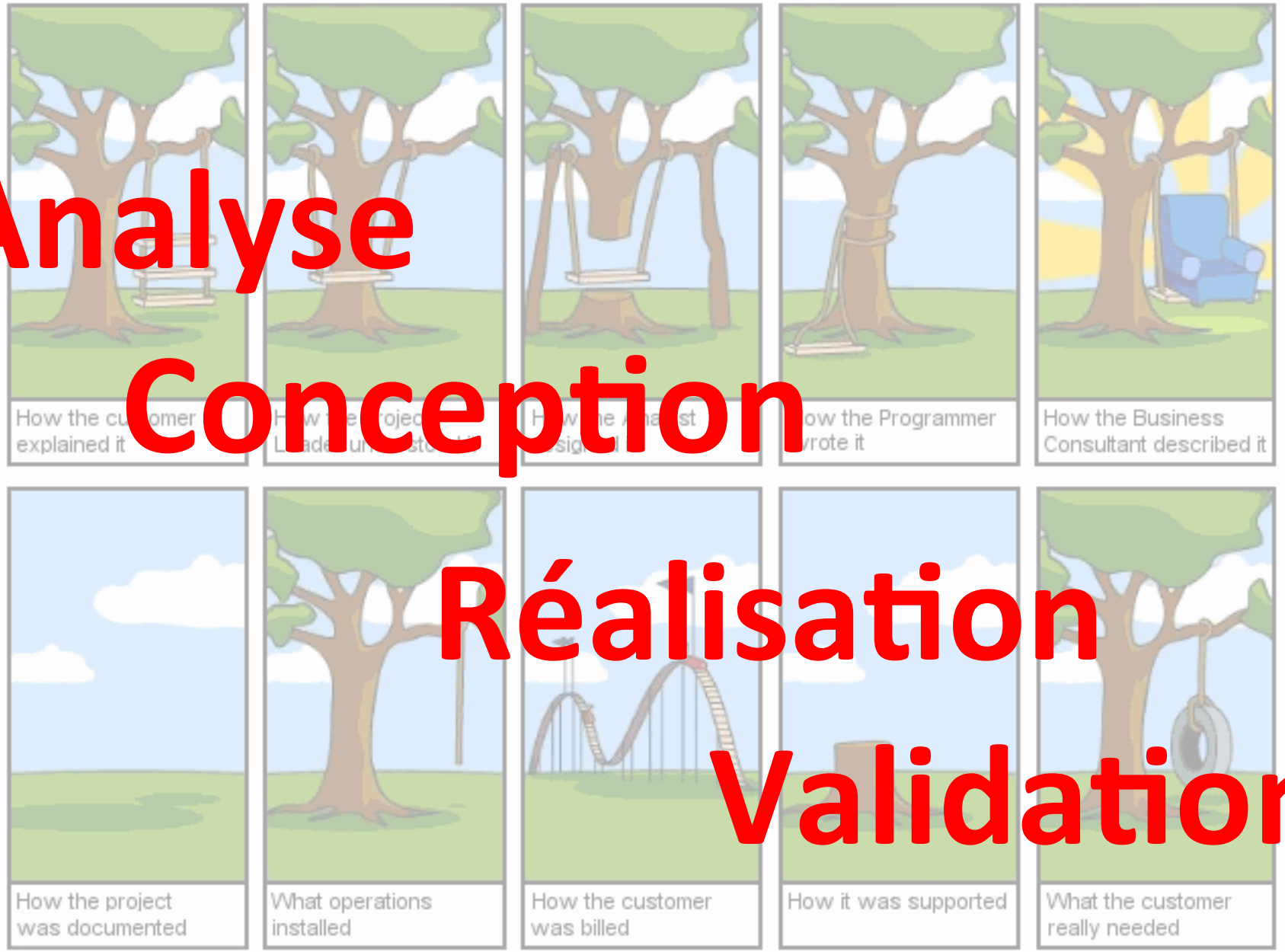
- Gestion de projets
  - Sur un exemple « joué » mais bien réel de projet où des résultats sont attendus
  - Projet en groupe

- Préparation pour le stage au 2ème semestre
  - Et pour votre future vie professionnelle !


Système



PDL en pratique?



Analyse

Conception

Réalisation

Validation

Quatre objectifs, quatre rendus

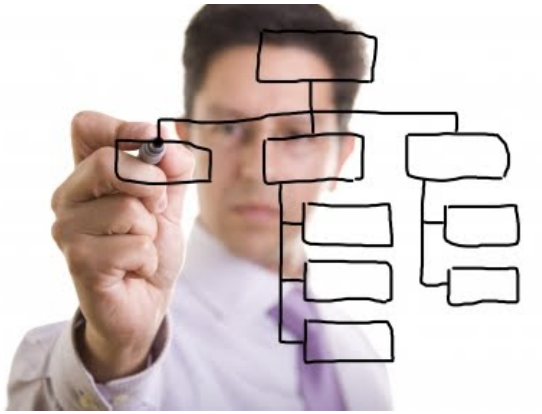
- Ingénierie des exigences, gestion de projet
  - Analyse des besoins avec un client
  - Organisation
- Conception
- Réalisation et test
- Déploiement de la solution
  - Présentation au client (présentation)

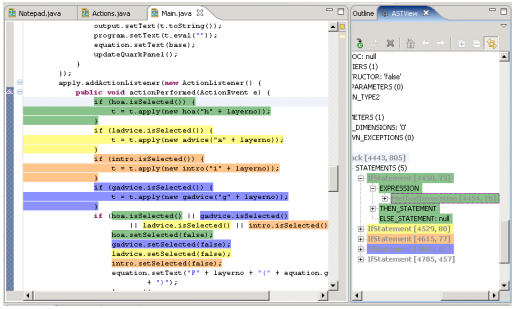
PDL: le “projet”

- Développement d’un système de chronométrage
- Client: Charles Quéguiner
- Enregistrer, gérer et imprimer, les chronométrages de courses de voitures et de séances d'essais d’une course automobile (e.g., 24 heure du Mans)

PDL: le “projet”

- https://github.com/acherm/teaching/tree/master/PDL/
- Prochaine séance: document cahier des charges





A rendre

- Livrable d'analyse (A1)
- Livrable de conception (B1)
- Code (C1)
- Présentation (D1)

Livrable d'analyse (A1)

- Analyse du cahier des charges de l'application
- Qualités attendues:
  - non ambigüe
  - complet
- Contenu (typiquement) attendu:
  - Les cas d'utilisations de l'application
  - Pour chaque cas d'utilisation au moins un scénario nominal et un scénario exceptionnel
    - Ces scénarios doivent décrire les interactions entre les acteurs et le système (diagramme de séquences UML)
  - Un diagramme de classes d'analyse
  - Prototype de l'interface graphique
    - Avec des modèles d'UI (sketchs)
    - Machine à états UML

Livrable de conception (B1)

- Détaille la façon dont vous réalisez le système
  - réponse au cahier des charges (~ A1)

- Contenu (typiquement) attendu:
  - Un découpage en sous-systèmes (paquetages) de l'application
  - Un diagramme de classes complet de l'application.
  - Pour chacun des scénarios décrits dans le livrable A1, un scénario détaillé montrant les interactions entre les objets internes au système.
    - Diagrammes de séquences ou machines à états décrivant le comportement des principales classes du système

Livrable de conception (B1), suite

- Processus de validation à mettre en oeuvre pour assurer la qualité du logiciel
- Détail des techniques que vous allez utiliser pour vous assurer que
  - le système développé répond au cahier des charges (A1) et est conforme à la conception que vous avez proposée (B1)
- Idéalement
  - Contraintes OCL
  - Au minimum: invariants, préconditions/ postconditions en langage naturel

Code (C1)

- Code source de l'application
  - Cela inclue évidemment les tests
  - La documentation
  - Un packaging du tout
- Nécessairement en Java
- Liberté totale pour les "frameworks"
  - À condition qu'ils soient open source
    - À discuter avec le client

Code (C1), bis

- Tests pour accompagner la livraison du code de votre application
  - décrit dans V1 sur le code que vous avez développé détaille les résultats du processus de validation

- Vise à montrer à votre client que le processus de validation a été mis en œuvre sur le code que vous lui avez livré

Soutenance (D1)

- 15’ de présentation
  - Rappel du cahier des charges
  - Expliquer les choix de conception principaux
  - Description de l’implémentation
  - Retour d’expérience
  - Démo de 5’
- 10’ de questions par le jury

Séances

- 6 séances
  - Une partie TD
  - Et une partie TP
  - 6 x 2 = ?
- Groupe: par 3 ou 4
- Cours magistraux
  - Adaptatifs: wait & see

Evaluation

- ¼ analyse
- ¼ conception
- ¼ code + validation
- ¼ soutenance

Projet

- Liberté…
- Groupe
  - Outils de versioning (git)
  - Outils collaboratifs
- Répartissez-vous les rôles
  - Impossible de rendre en temps et en heure sinon
- Résultats attendus
  - Très fortes contraintes sur les dates de rendus (cela fait partie intégrante de l'exercice)

- Constitution des groupes
- Inscription sur github

- Avant le 21 octobre 2013 midi, feuille excel avec la constitution des groupes (nom des membres + nom d'utilisateur sur github)

A rendre

- Livrable d'analyse (A1)
  - 22 novembre, 20h
- Livrable de conception (B1)
  - 6 décembre, 20h
- Code (C1)
  - 18 décembre, 20h
- Présentation (D1)
  - Début janvier

A rendre

- Livrable d'analyse (A1)
  - 22 novembre, 20h
- Livrable de conception (B1)
  - 6 décembre, 20h
- Code (C1)
  - 18 décembre, 20h
- Présentation (D1)
  - Début janvier
- Approche itérative et agile
  - Ne faîtes pas A1, puis B1, puis C1: **cela ne marchera pas**
  - Cela ressemblera plus à A1, B1, C1, puis A1, B1, puis de nouveau A1, puis B1, C1, puis de nouveau A1, B1, C1, etc.
- Contraintes de temps
  - Faîtes des choix (idéalement validés par le client)
  - Obtenez des résultats rapidement et itérez