

Bases de la conception orientée objet

À destination des étudiants de
1e année IUT (S2)

Mireille Blay-Fornarino
Université Nice Sophia Antipolis
blay@unice.fr
<http://mireilleblayfornarino.i3s.unice.fr/>

Site web du module :
<https://mbf-iut.i3s.unice.fr/>



The greatest enemy of knowledge is not ignorance, it is the illusion of knowledge.



Objectifs du cours

- **Connaître la modélisation UML**
 - ▶ Savoir lire des modèles ; savoir les construire
 - ▶ Faire le lien entre un modèle et le code qui pourrait correspondre.

- **Savoir analyser un problème pour ensuite pouvoir l'implémenter**
 - ▶ En répondant aux besoins des utilisateurs
 - ▶ En assurant la qualité du logiciel produit (performance, utilisabilité, sécurité, maintenabilité, ...)

Notation

- Des notes de TD
- **Un rendu final type carnet de bord**
- Un examen final portant sur une étude de cas
- Des contrôles surprises

Soyez Rigoureux :

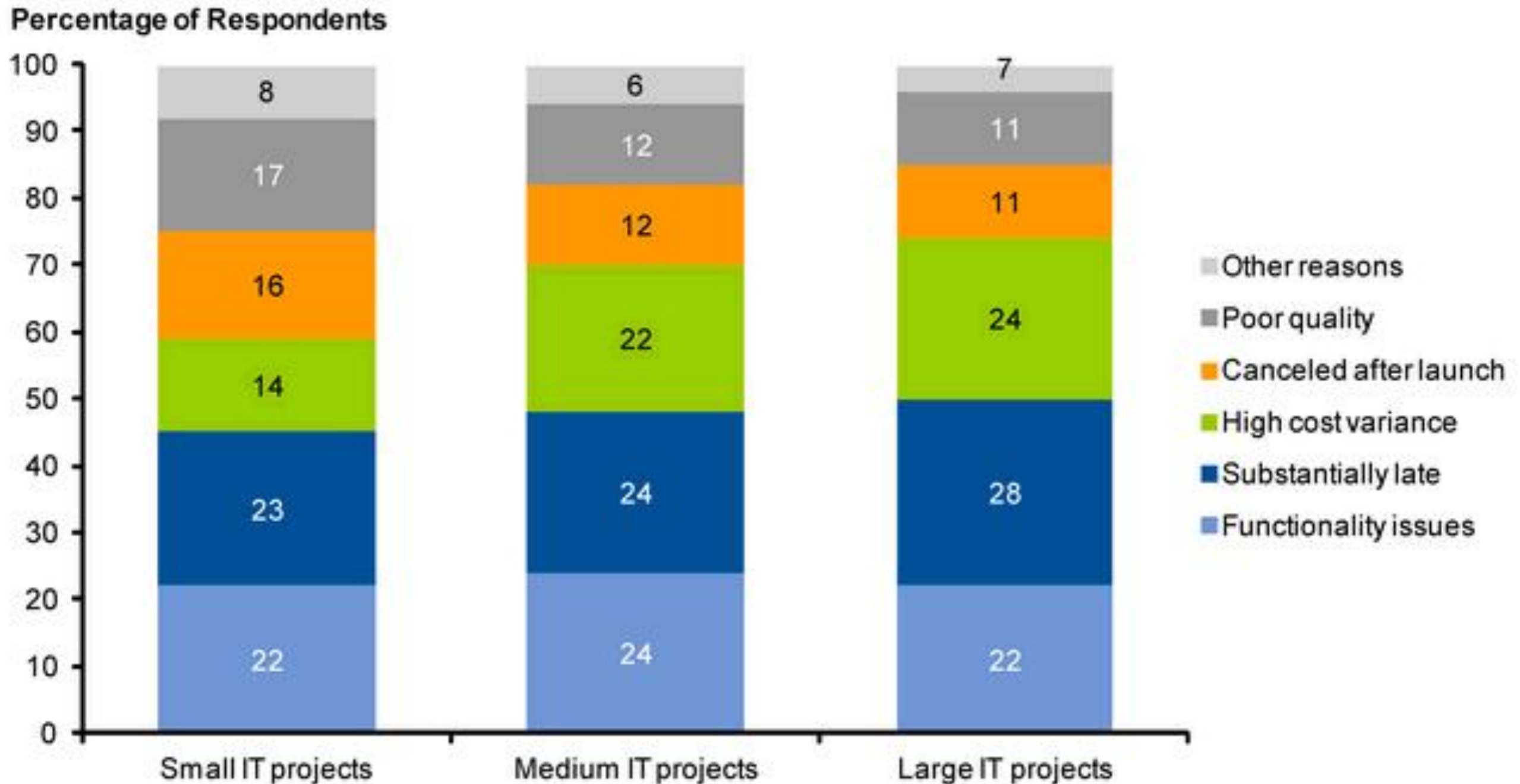
- Respectez les dates;
- Respectez les **notations**;
- Respectez les **consignes**.



Introduction à la COO

On vous propose de réaliser
une application logicielle
pour visualiser les données
météo dans un cockpit,
Que faites-vous?

Raisons de l'échec des projets (Gartner 2012)



Raisons de l'échec des projets

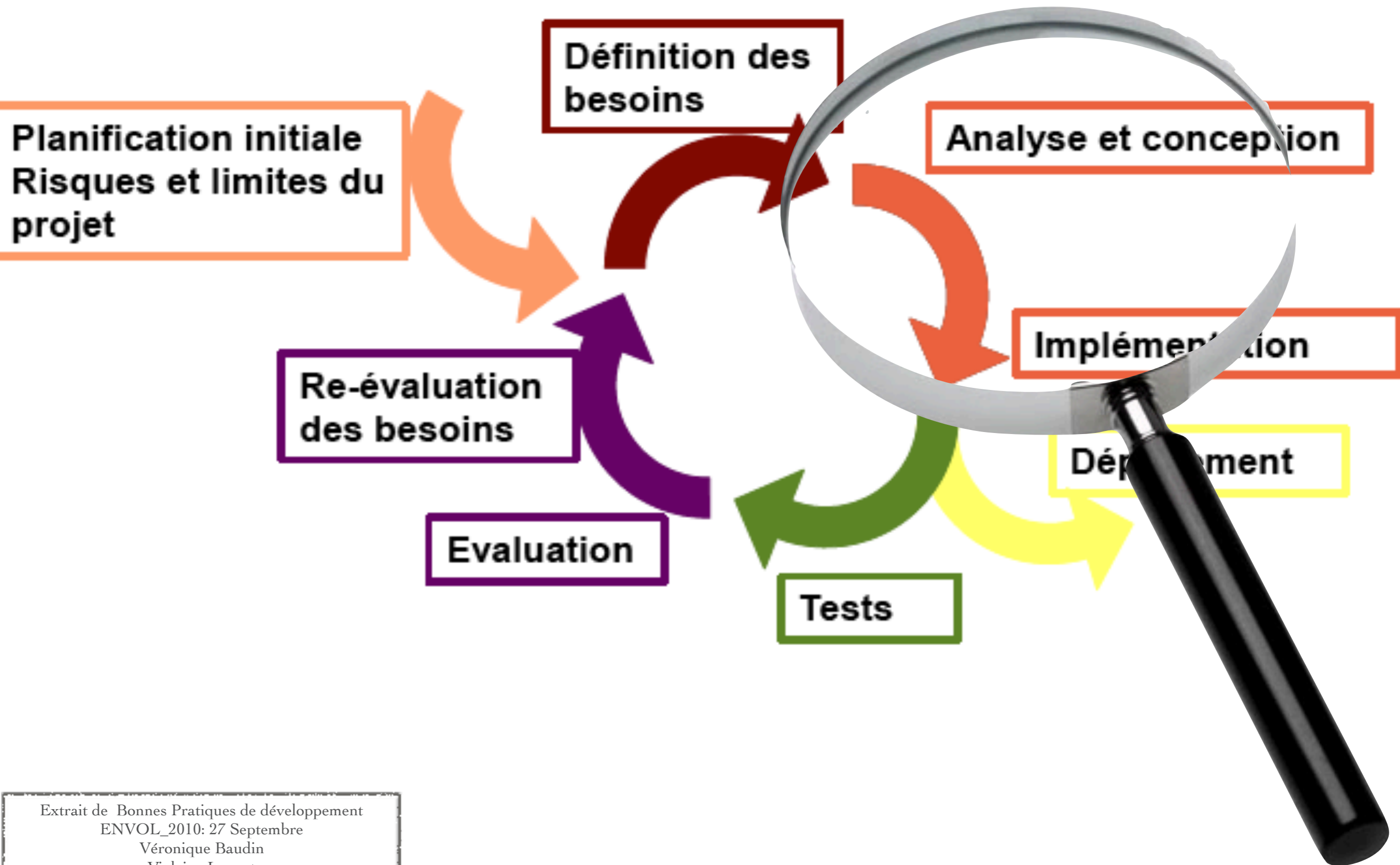
Project Management Institute 2017

Causes of project failure? Business-IT alignment is (still) to blame.

When asked to pick the top three causes of project failure, four of the top five “winners” are all about the all-too-familiar, long-lived challenge of business-IT alignment.

Projects fail because teams can't keep pace with changing organizational priorities (41%) and project objectives (38%). They're trying to execute projects without a clear vision or goals (31%). And they continue to struggle to define requirements accurately (37%).

Activités du développement logiciel



Introduction à UML

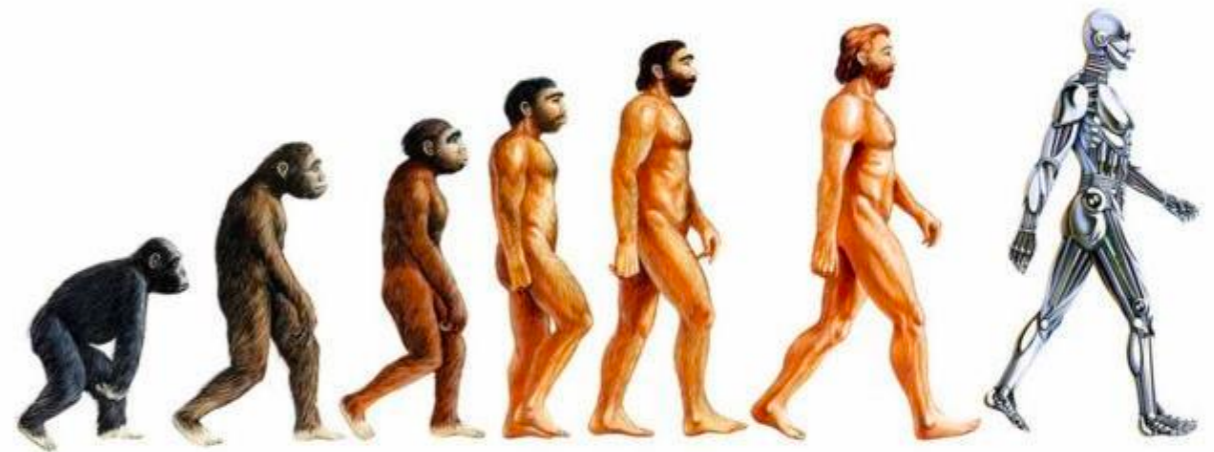
UNIFIED
MODELING
LANGUAGE



Un peu d'histoire : La guerre des méthodes

Fin des années 90 , des méthodes : Booch, OMT, Coad/Yourdon, Fusion, SADT, OOSE, Schlaer/Mellor, HOOD...
(Merise 80)

On a au moins besoin d'un « langage » de modélisation standard !



Un langage unique !



"Lingua franca"

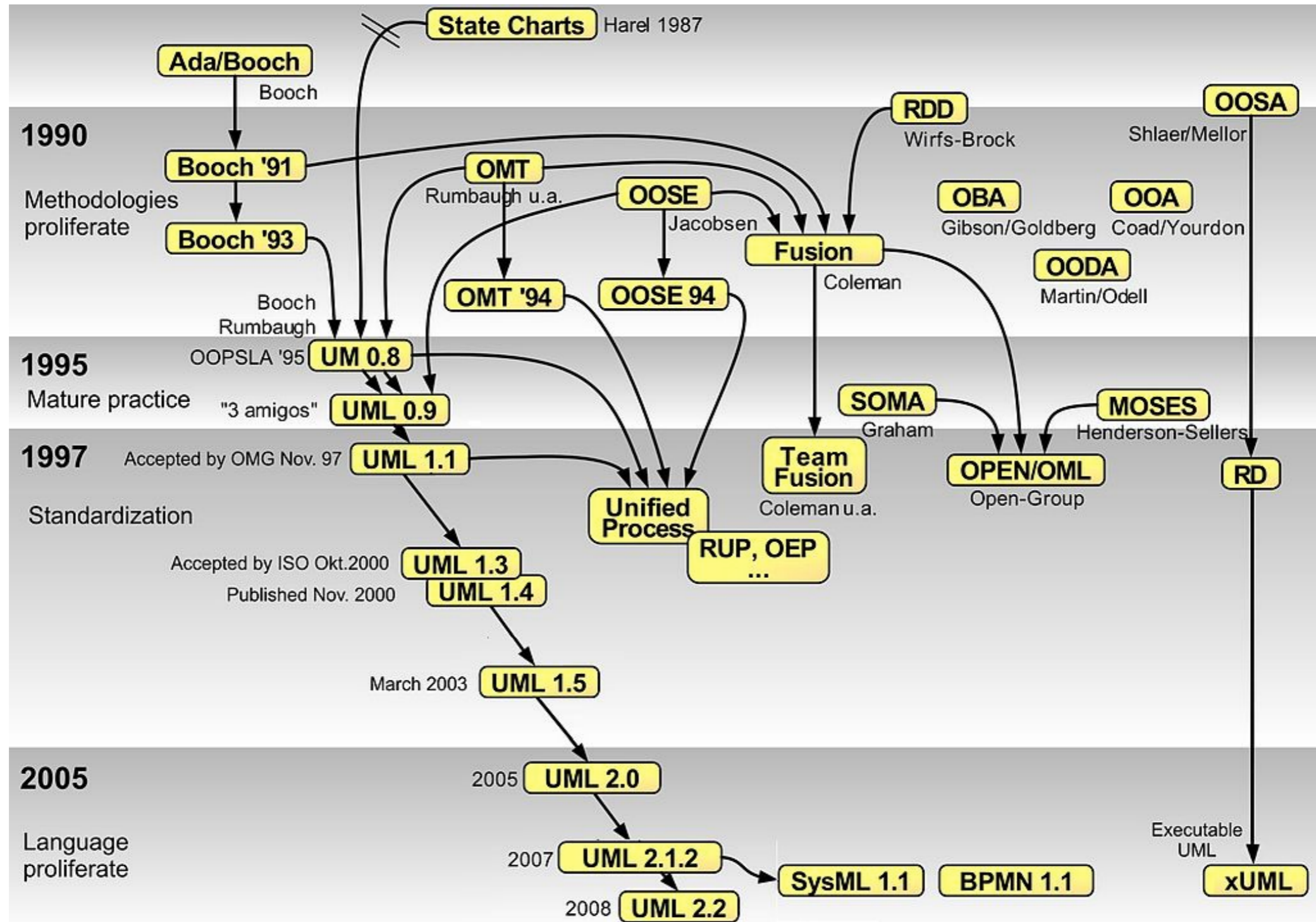
- **Auteurs** : Grady Booch, Ivar Jacobson, James Rumbaugh.
- **Standardisation **OMG**** (Object Management Group) en 1997
- **Promoteurs** :
 - **Rational Software, Oracle**
 - **HP, Microsoft, IBM**

<https://www.omg.org/index.htm>

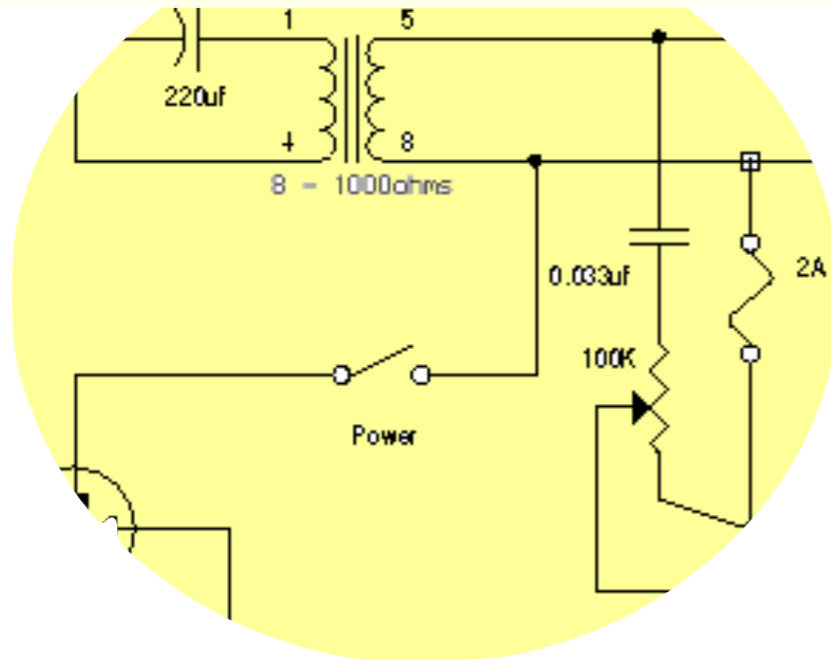
UML 2.5.1 décembre 2017

<https://www.omg.org/spec/UML/2.5.1/>

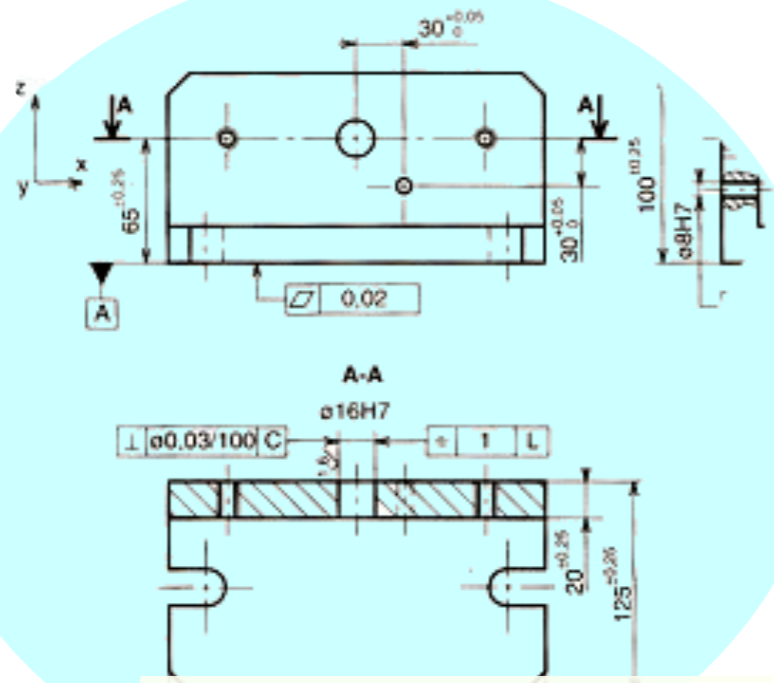
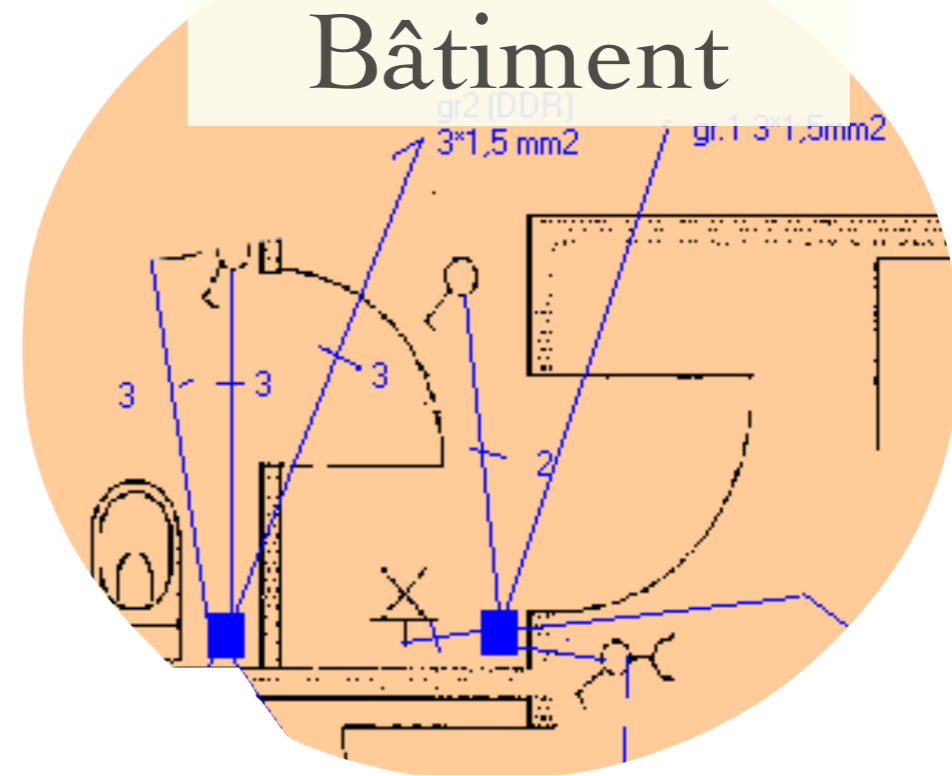
796 pages



Ingénierie Électrique



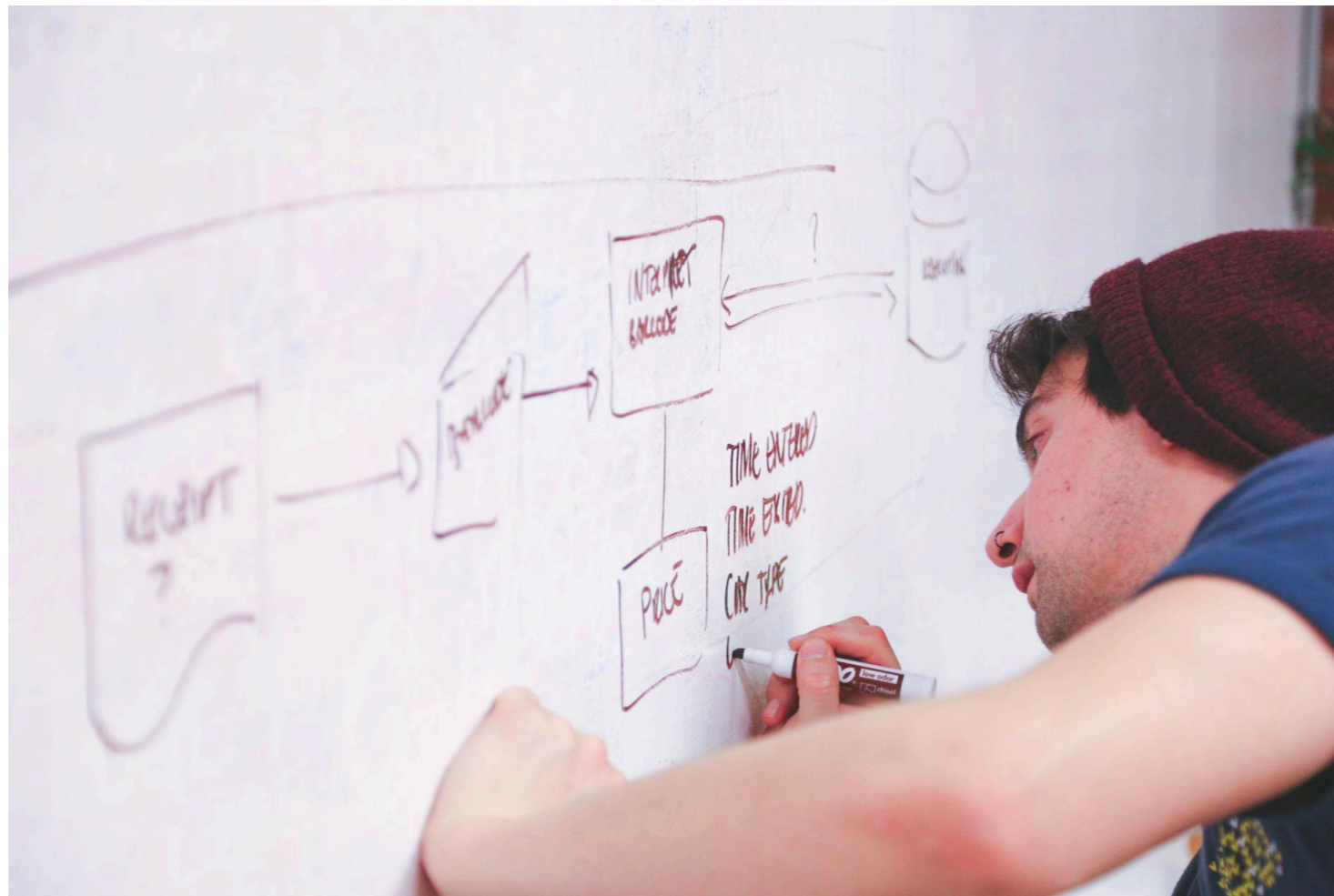
Ingénierie du Bâtiment



Ingénierie Mécanique

Qu'est-ce qu'UML ?

Qu'est-ce qu'UML ?



un support à la modélisation

Qu'est-ce qu'UML ?



un support visuel à la communication : « client », équipe..

Qu'est-ce qu'UML ?

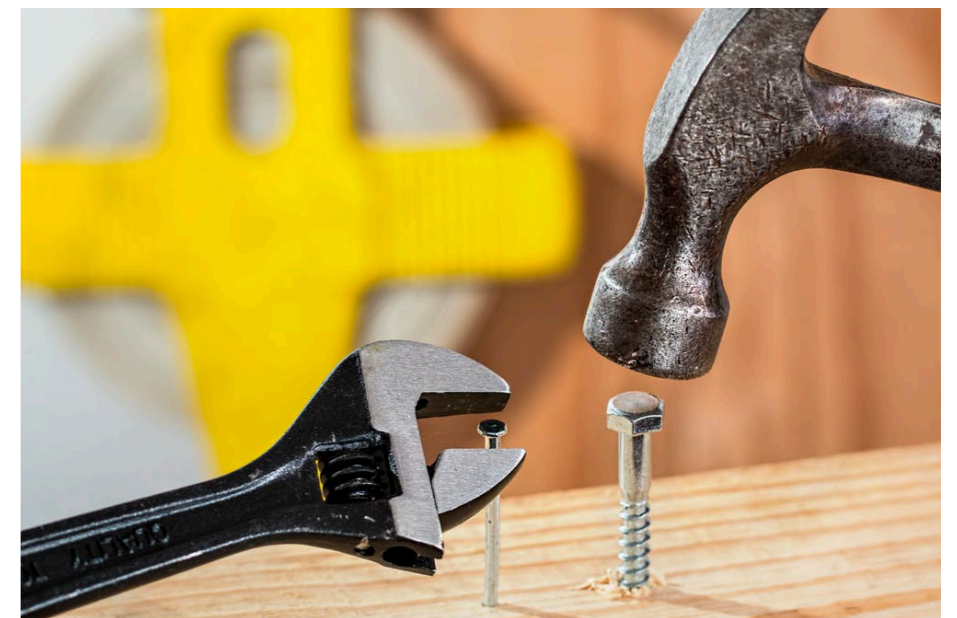


Une aide au développement

Les points forts d'UML

- UML est un langage normalisé
 - gain de précision
 - gage de stabilité
 - encourage l'utilisation d'outils

Mais bien sûr.... C'est vous qui les utilisez !



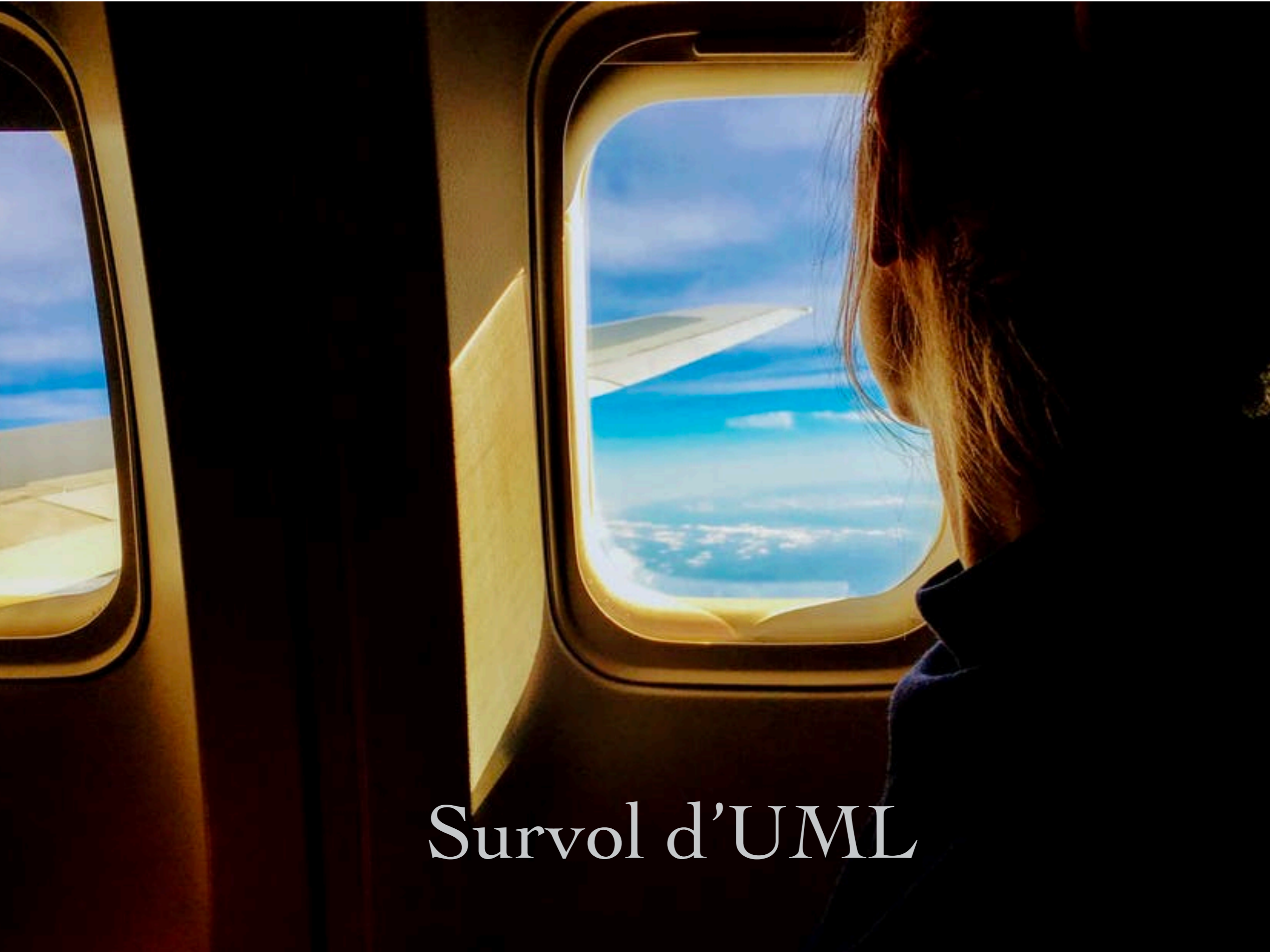
Les points forts d'UML

- UML est un support de communication performant
 - Il cadre l'analyse.
 - Il facilite la compréhension de représentations abstraites complexes.
 - Son caractère polyvalent et sa souplesse en font un langage universel.

Il doit vous aider à comprendre les concepts de la programmation par objets!

Les points faibles d'UML

- La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation.
- Le processus de développement (non imposé par UML) est une autre clé de la réussite d'un projet.



Survol d'UML

Vue fonctionnelle

la vue fonctionnelle cherche à appréhender
les interactions entre les
acteurs/utilisateurs
et le système,

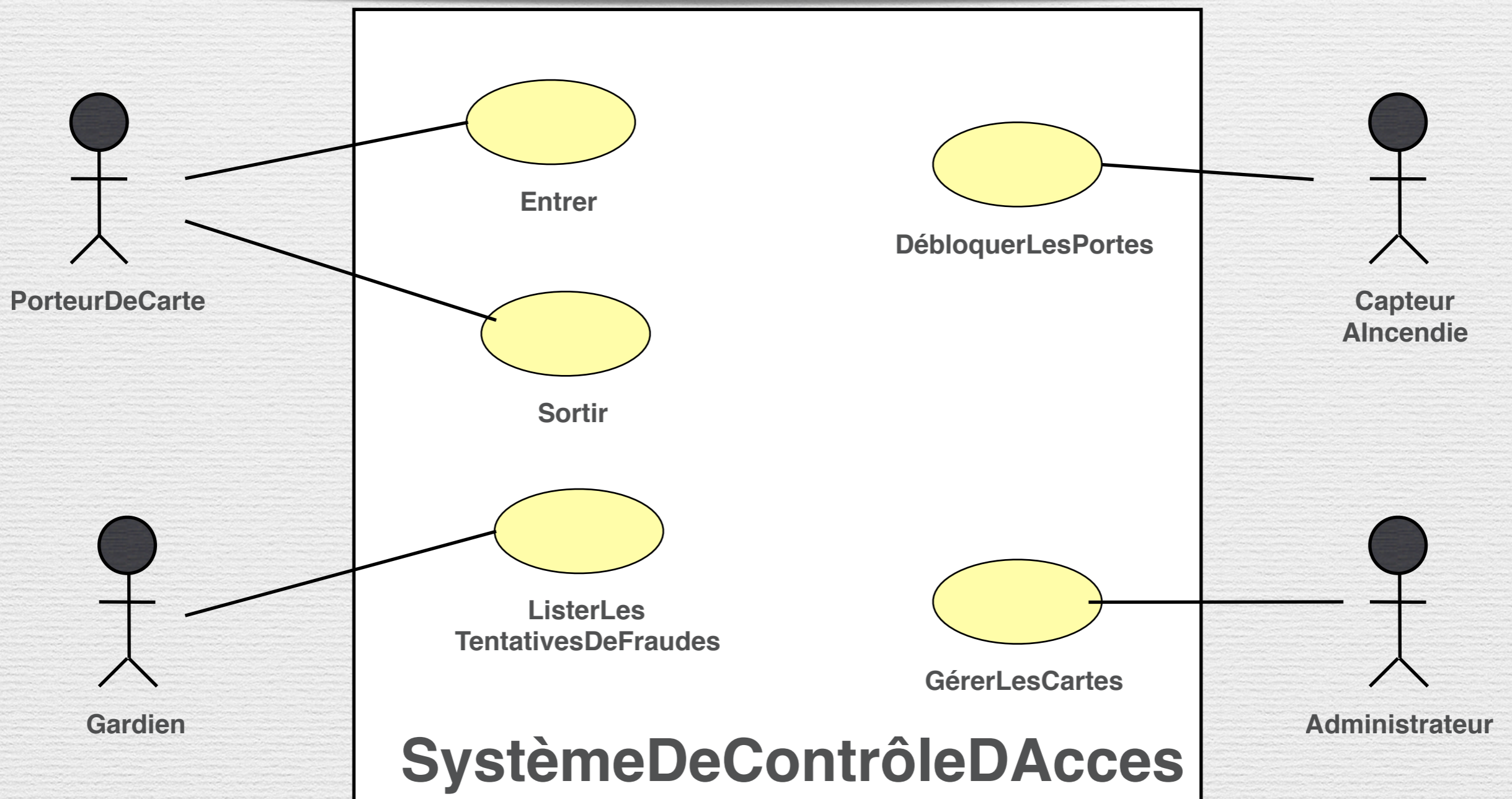
sous forme d'objectifs à atteindre (**cas d'utilisation**)

et sous forme chronologique de scénarios d'interaction
typiques (**diagrammes de séquences**)

Qu'est-ce qu'UML ?

Diagrammes des cas d'utilisation

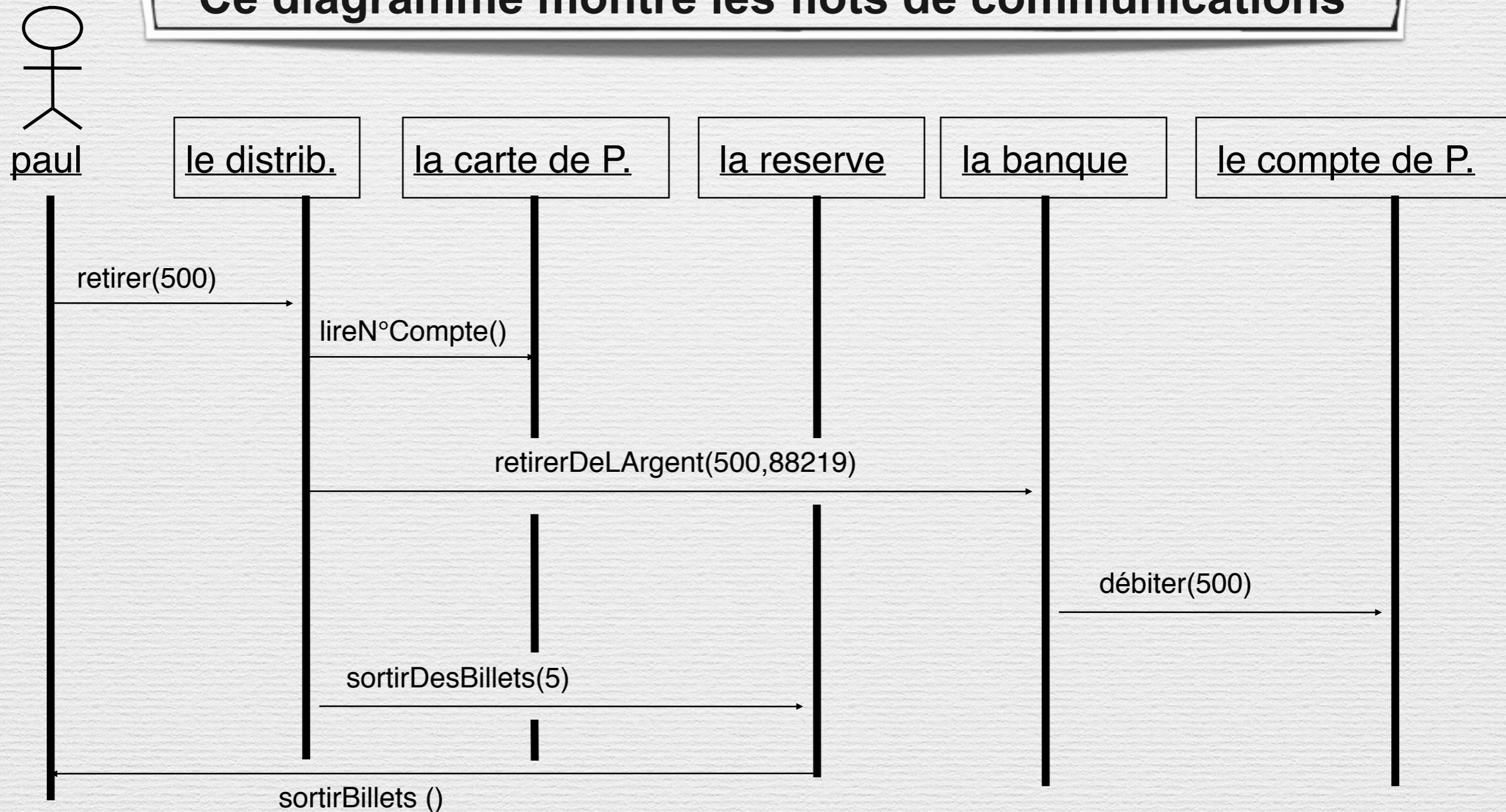
Ce diagramme montre ce que fait le système et qui l'utilise



Qu'est-ce qu'UML ?

Diagrammes de séquence

Ce diagramme montre les flots de communications



Vue Structurelle

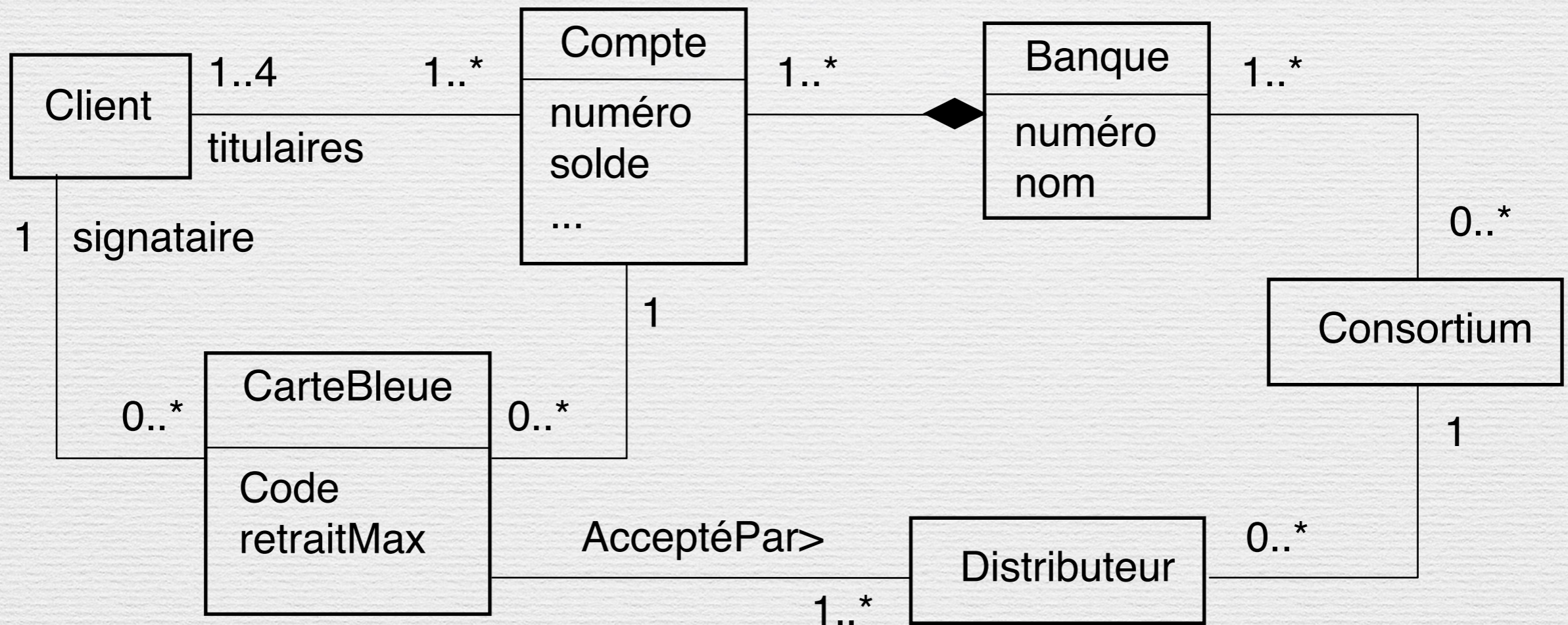
la vue structurelle, ou statique, vise à **identifier les objets/composants** constituant le programme, leurs attributs, opérations et méthodes, ainsi que les liens ou associations qui les unissent (**diagramme de classes**).

Elle permet aussi de regrouper les classes fortement liées entre elles en des composants les plus autonomes possibles (**diagramme de packages**).

A l'intérieur de chaque package, on trouve un diagramme de classes.

Qu'est-ce qu'UML ?

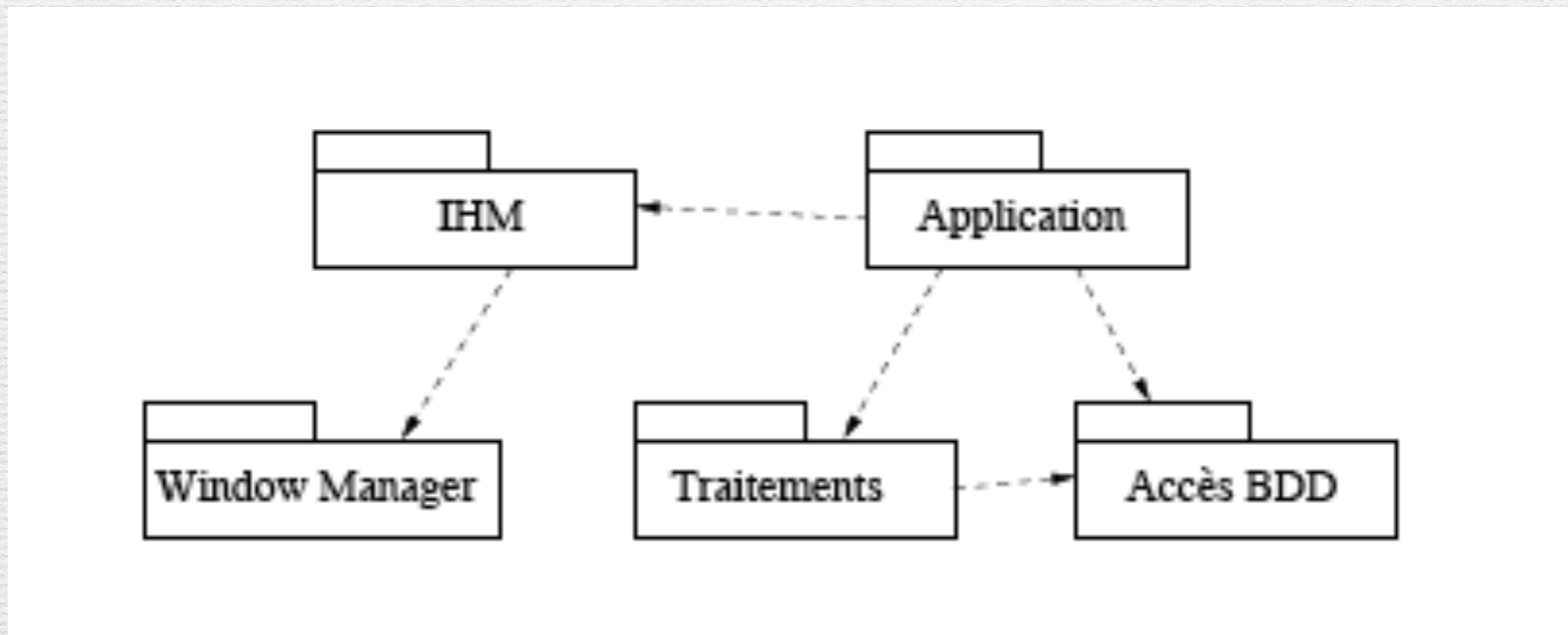
Diagrammes de classes



Ce diagramme montre les classes et les relations entre elles

Qu'est-ce qu'UML ?

Diagrammes de packages



Regrouper entre elles des classes liées les unes aux autres de manière à faciliter la maintenance ou l'évolution du projet et de rendre aussi indépendantes que possible les différentes parties d'un logiciel.

Vue Dynamique

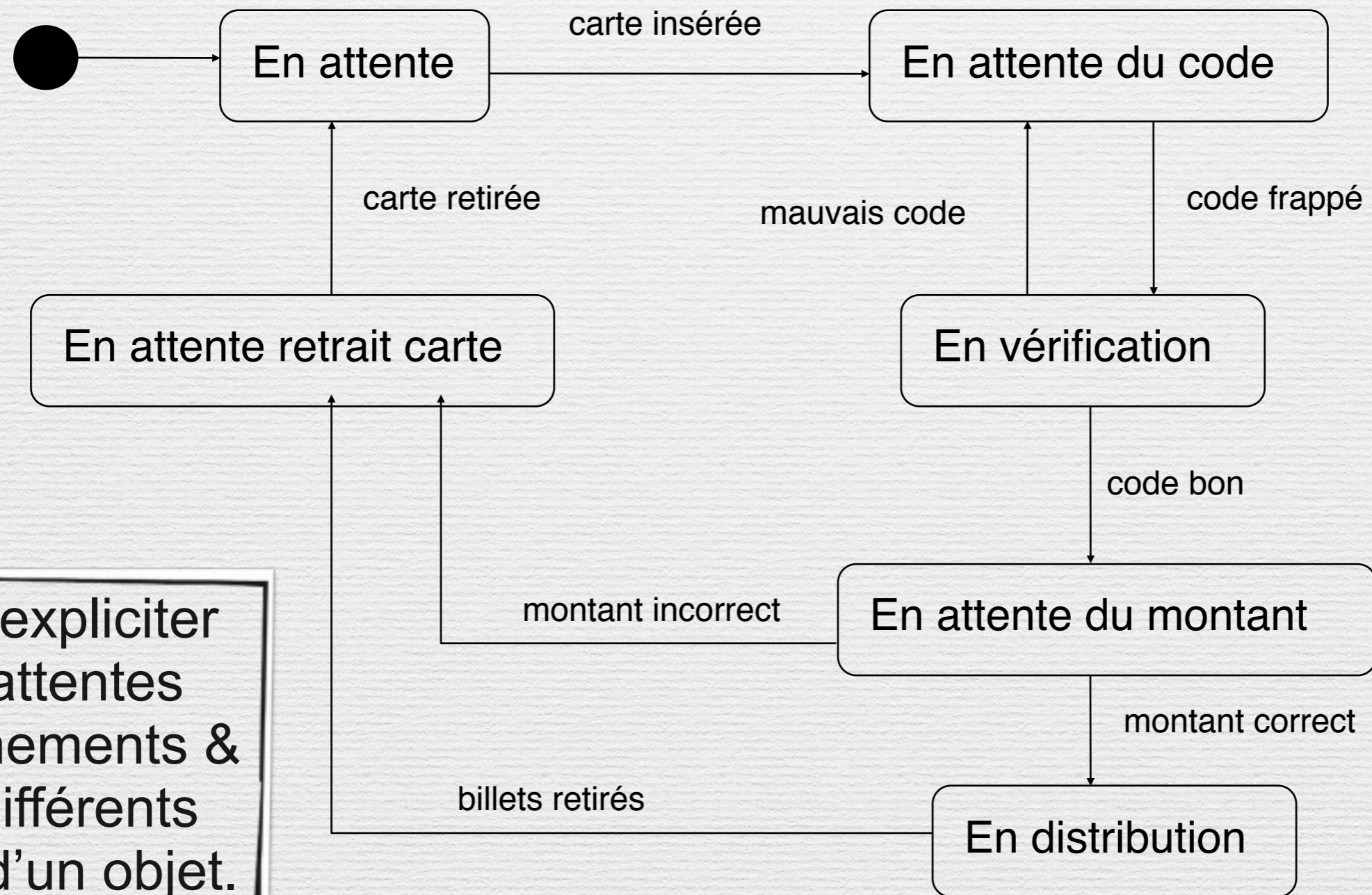
la vue dynamique vise à décrire l'évolution (la dynamique) des objets complexes du programme tout au long de leur cycle de vie.

De leur naissance à leur mort, les objets voient leurs changements d'états guidés par les interactions avec les autres objets (les **diagrammes d'états**).

Le **diagramme d'activité** est une sorte d'organigramme correspondant à une version simplifiée du diagramme d'états. Il permet de modéliser des activités qui se déroulent en parallèle les unes des autres, quand ce parallélisme peut poser problème.

Qu'est-ce qu'UML ?

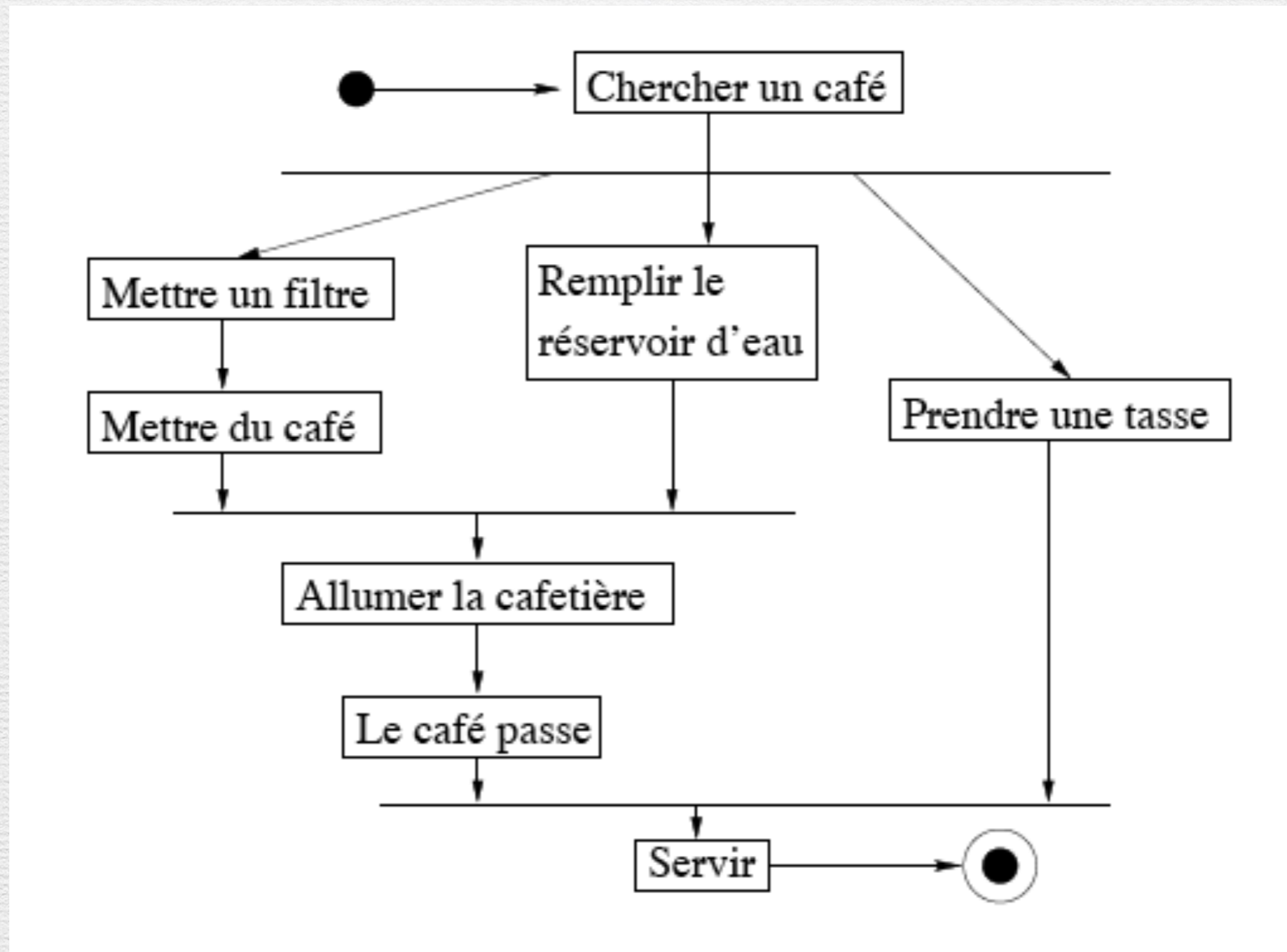
Diagrammes d'états



Pour expliciter
les attentes
d'évènements &
les différents
états d'un objet.

Qu'est-ce qu'UML ?

Diagrammes d'activités





The purpose of abstraction is not to be vague, but to create a new semantic level in which one can be absolutely precise.

— *Edsger Dijkstra* —

AZ QUOTES

Then, because I was still a beginner, I was assigned responsibility for what was thought to be the least important software to be developed for the next [unmanned] mission . . . And it was developed for what would happen only if the mission aborted. So nobody really paid much attention to what I was doing, because it was "never going to happen." And I called, I still remember the name of the program was called "Forget it." I don't know that many people really had a chance to even see what was in there since I was pretty much left to my own devices, but when the mission was actually aborting, then I became the expert of the "entire mission" because control in the software had gone to "Forget it". So I had to come in for the emergency. I was called in, and I was the one who had all the answers to all of the questions in "Forget it."

"Many of the things I was intrigued by had to do with how to make the mission software safe and reliable. And one of the things I remember trying very hard to do was to get permission to be able to put more error detection and recovery into the software. So that if the astronaut made a mistake, the software would come back and say 'You can't do that.' But we were forbidden to put that software in because it was more software to debug, to work with. So one of the things that we were really worried about is what if the astronaut made a mistake -- We were also told that the astronauts would never make any mistakes, because they were trained never to make mistakes.

So we were very worried that what if the astronaut, during mid-course, would select pre-launch, for example? Never would happen, they said. Never would happen. It happened."



Bibliographie

Ce cours a été monté en utilisant de nombreux supports dont je remercie chaleureusement ici les auteurs
D'autres références se trouvent sur le site du module.

- Merise: 5ème Partie Dossier "SAM l'Informaticien" du 5 Mars au 18 Mars 2001 par Stéphane Lambert <http://www.vediovis.fr/index.php?page=merise5>
- Introduction au langage UML, SUPINFO
- De Merise à UML, Nasser Kettani, Dominique Mignet, Eyrolles
- http://www.compucycles.com/nouveausite/articles/Merise/Article_07.htm
- UML-MERISE Etude Comparative, OSITEC-Consultants, 2004-2005
- Modélisation Orientée objet, M.Grimaldi – janvier 2010

