Journée GL

CRIStAL

28 mai 2015

- Équipe créée officiellement en janvier 2015, à la création de CRIStAL
  - Début officieux en septembre 2013
  - 3 permanents (1 HDR) + 1 associé
  - 3 doctorants + 1 en début de signature + 1 à pourvoir

Cedric Dumoulin

cedric.dumoulin@univ-lille1.fr
http://cristal.univ-lille.fr/~dumoulin

Tiles

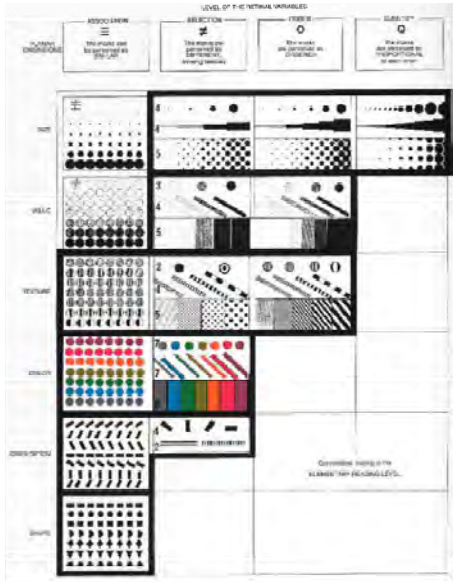
Equipe-Projet
INRIA DaRT

STRUTS

papyrus

Xavier Le Pallec

xavier.le-pallec@univ-lille1.fr
http://cristal.univ-lille.fr/~lepallec



Jean-Claude Tarby

jean-claude.tarby@univ-lille1.fr
http://cristal.univ-lille.fr/~tarby/

Interaction Homme-Machine

Ingénierie Dirigée par les Modèles

User Experience

Usages

Miny

WSE

BCI

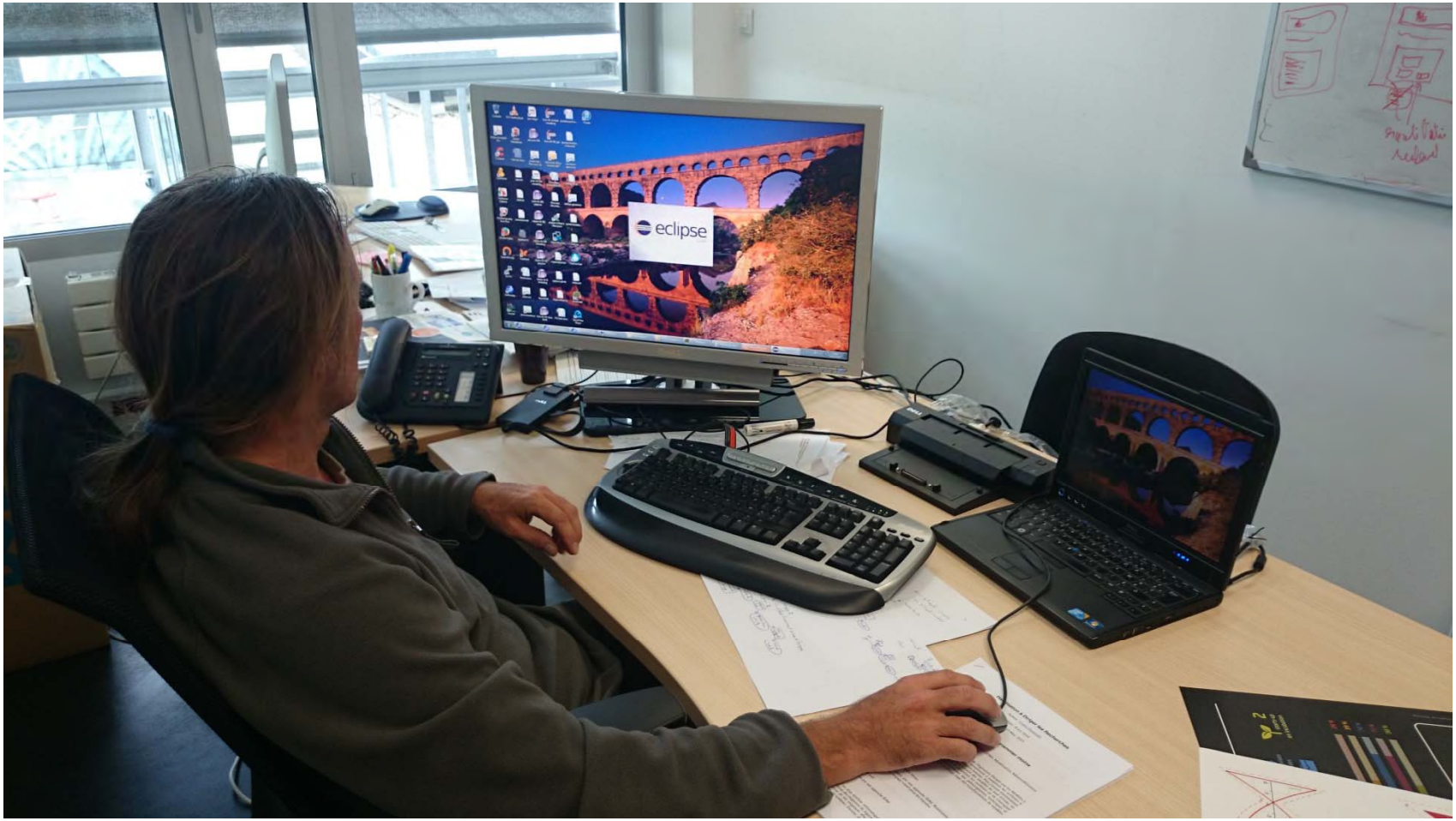
CSCW

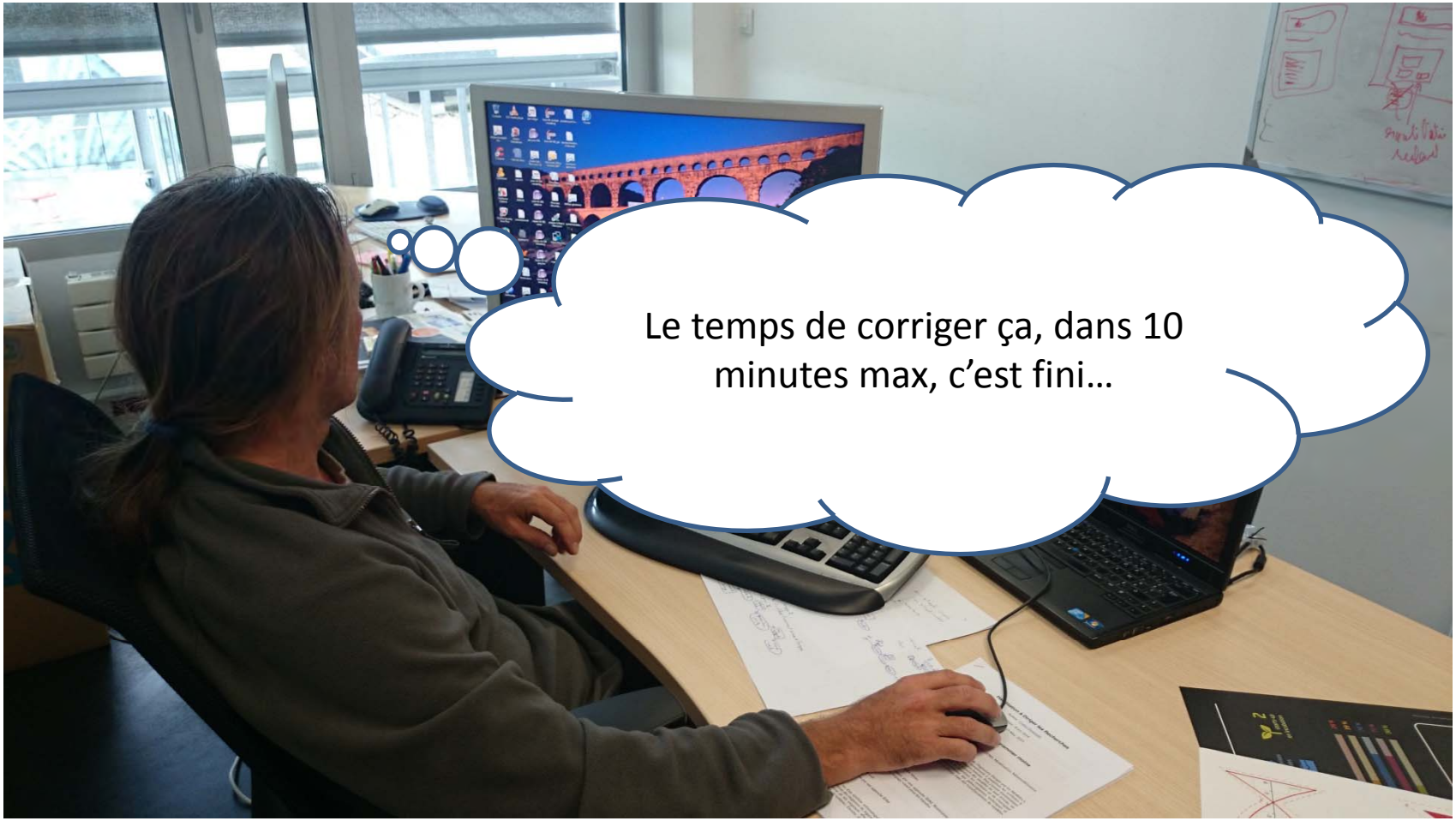
…

Scénario

- Cas fictif d'un développeur dans une grande société. Appelons-le par exemple « Cédric »…
  - dans une équipe de 10 développeurs
  - il travaille actuellement sur 5 grands projets…
    - Enfin, il « travaillait » car il a du s'absenter pendant 1 an. Il revient aujourd'hui, et son chef de projet lui demande de résoudre un problème dans un de ces 5 projets.

- « Facile », se dit-il, « je les connais bien, et même si ça fait un an que je n'ai pas touché, je vais m'y remettre sans problème ! »


eclipse


Le temps de corriger ça, dans 10 minutes max, c’est fini…

- Cédric ouvre le projet:
  - 4.000 classes,
  - 50 packages,
  - 600.000 lignes de code !


Ah crotte ! Je ne me rappelais pas qu'il y avait autant de « trucs » là-dedans !


Mais bon, c’est facile, je suis sûr que le bug vient forcément de la classe « Machin »


Alors, si je me rappelle bien, ça doit être dans le package « Brin_A_Terminer »

- Cédric ouvre la classe concernée…

- Mais malheureusement, comme souvent…


Non, mais ce n’est pas possible !!!!

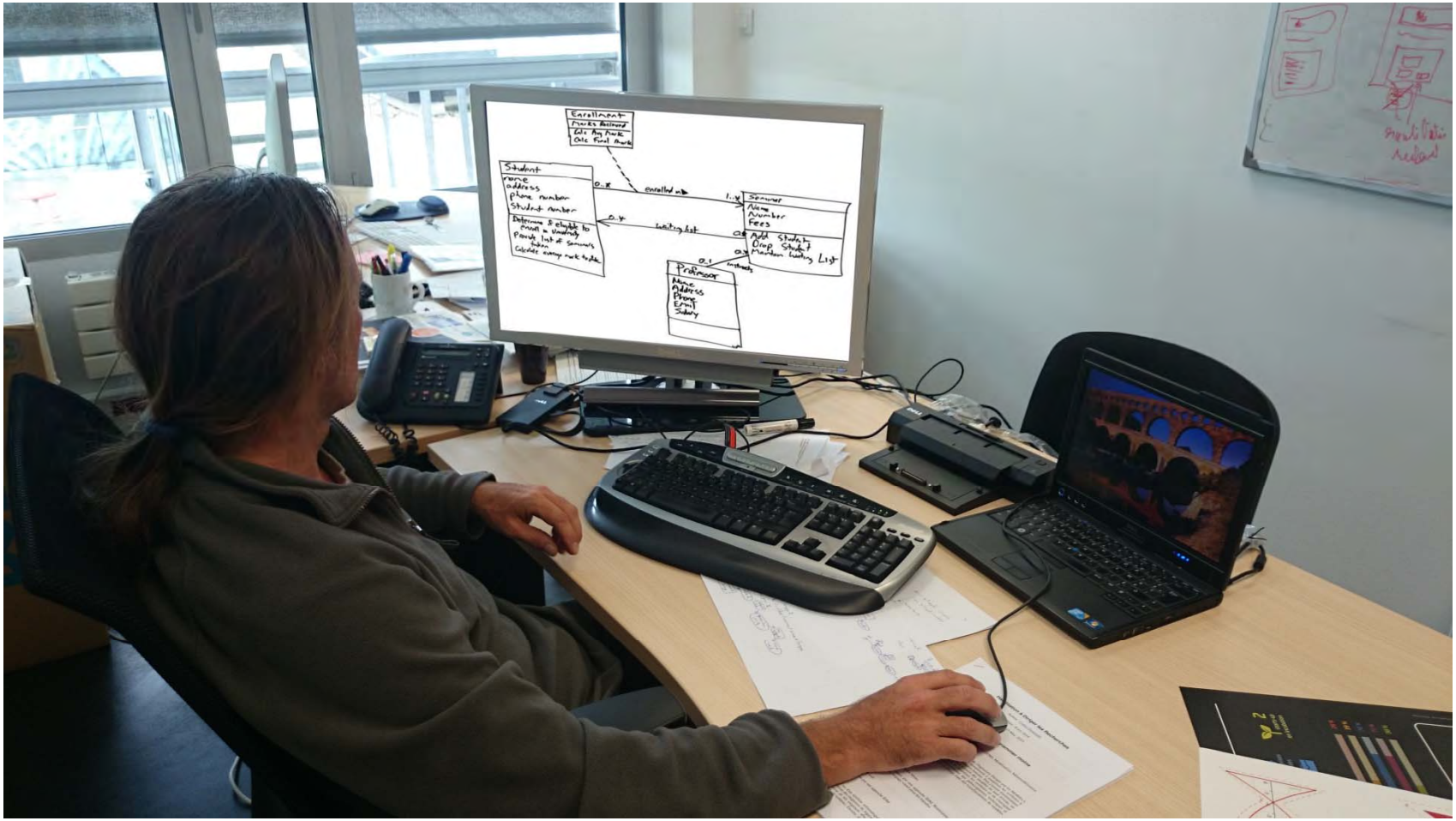
```
@Override
public View getView(int position, View convertView, ViewGroup parent) {
    ViewHolder holder;

    if (convertView == null) {
        holder = new ViewHolder();
        convertView = inflater.inflate(R.layout.item_list_affichage_animal,
                null);

        holder.animalName = (TextView) convertView
                .findViewById(R.id.animal_Name);
        holder.animalPicture = (ImageView)convertView
                .findViewById(R.id.animal_picture);

        convertView.setTag(holder);
    } else {
        holder = (ViewHolder) convertView.getTag();
    }

    Log.v("AnimalName",animalList.get(position).getName());


    holder.animalName.setText(animalList.get(position).getName());

    animalDao = DatabaseManager.getDao().getAnimalDao();
    Animal a = animalDao.findById(position+1);

    String picture = a.getPicture_location();
    String[] tab_picture = picture.split(",");
    if(tab_picture.length>0){
        resID = ctx.getResources().getIdentifier(tab_picture[0], "drawable", convertView.getContext().getPackageName());;
    }
    else{
        resID = R.drawable.no_image;
    }
    holder.animalPicture.setImageResource(resID);

    return convertView;
}
```

- Eh oui, le développeur n’a mis aucun commentaire dans le code !
  - C’est bien connu, « les commentaires, ça ne sert à rien… », n’est-ce pas ☺
- Pas immédiat de savoir qui a écrit ça, pourquoi, etc.


Pas grave, un petit « reverse » du projet et le tour sera joué…

- Après 20 minutes passées à réorganiser le schéma, le résultat est…


Ok, on oublie le reverse… !
Pas grave, il me reste la documentation…


Enrollment
Student
Seminar
Professor
enrolled in
waiting list


Mais bip c'est quoi cette doc de bip !!!!


Student
name
address
phone number
Student number
enrolled in
waiting list


CENSURÉ

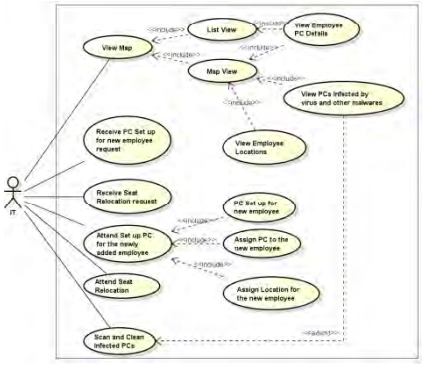
Conclusions/Pistes de réflexions

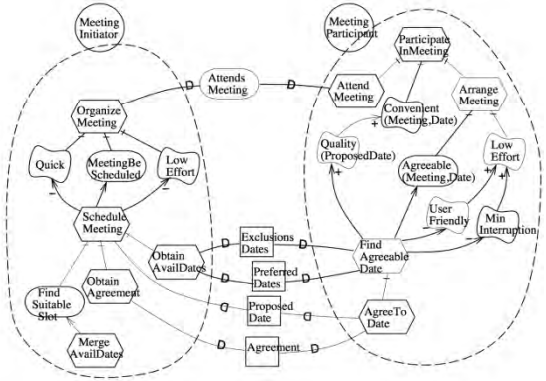
- **Le code ne suffit pas, le code n'est pas tout**
- Les **modèles en amont** (quand ils sont fait) **ne suffisent pas** (pas de mise à jour)
- Les modèles ne sont pas (jamais ?) faits en amont, ou alors **incomplets**, ou **pas utilisé par la suite**
- **Outils de modélisation trop complexes**, pas adaptés à « la tâche »
- **Développeurs toujours avec simple IDE textuel**, alors que IHM a fait des progrès immenses en 30 ans !
- On a **besoin de modèles, mais aussi d'interactions évoluées pour « naviguer » dans le code/projet**

Conclusions/Pistes de réflexions

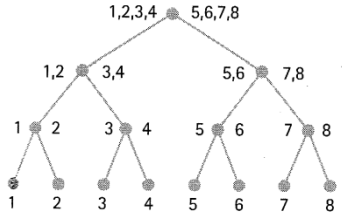
- On a **besoin de visualiser et d'interagir** avec tout ce qui est en relation avec le projet
  - Code
  - Modèles
  - Documentation
  - Compte-rendus de réunions
  - Photos
  - Enregistrements audios
  - …
- Besoin de modèles différents à différents moments, **multi-vues**…
- On a besoin de **connecter le code et le « reste »**
- **Prise en compte du contexte**
  - qui fait quoi, où, avec quels périphériques, avec qui, …

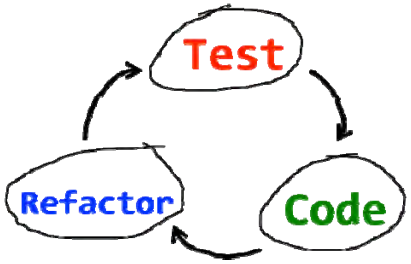
Software Industry



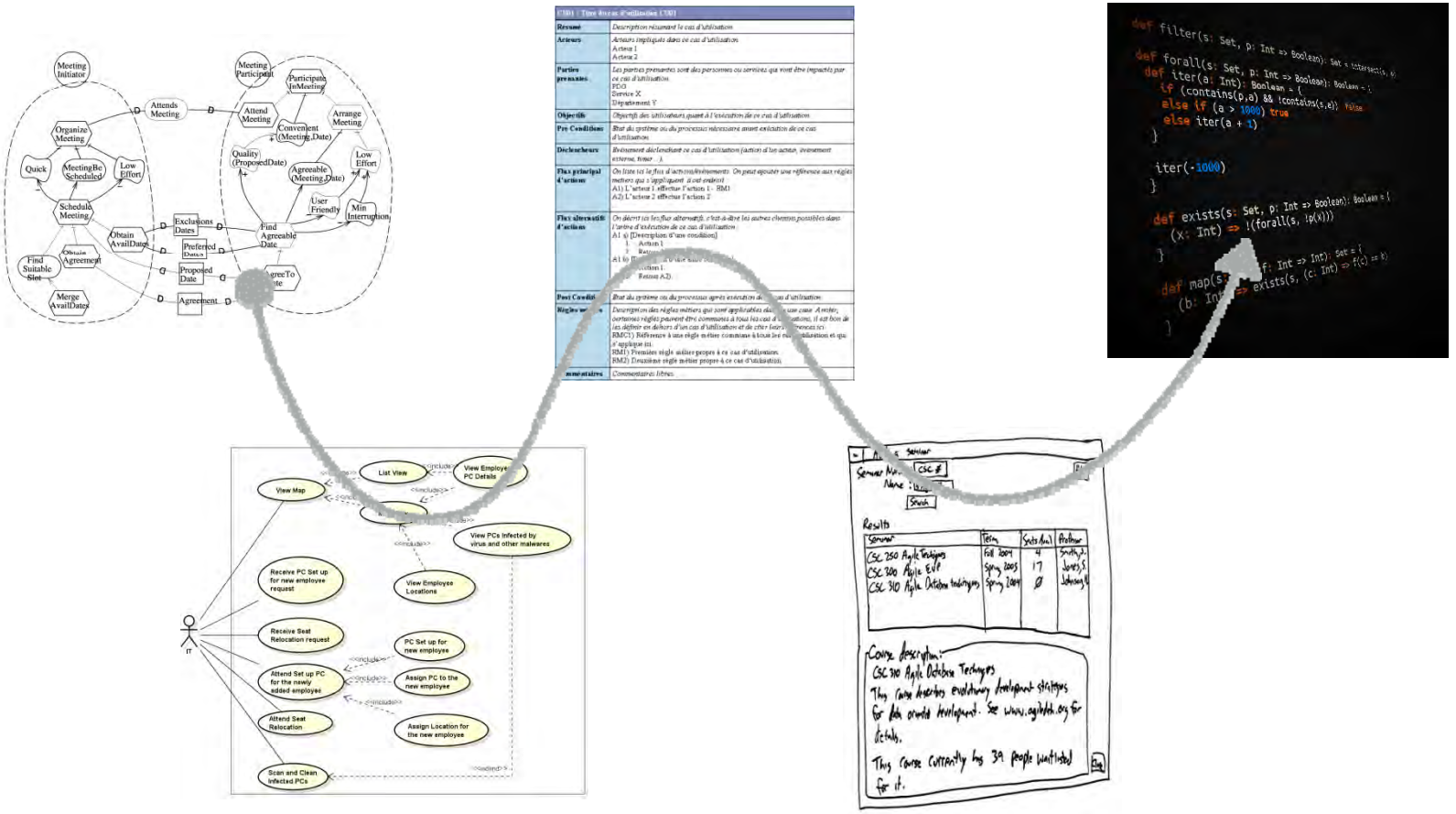


$$

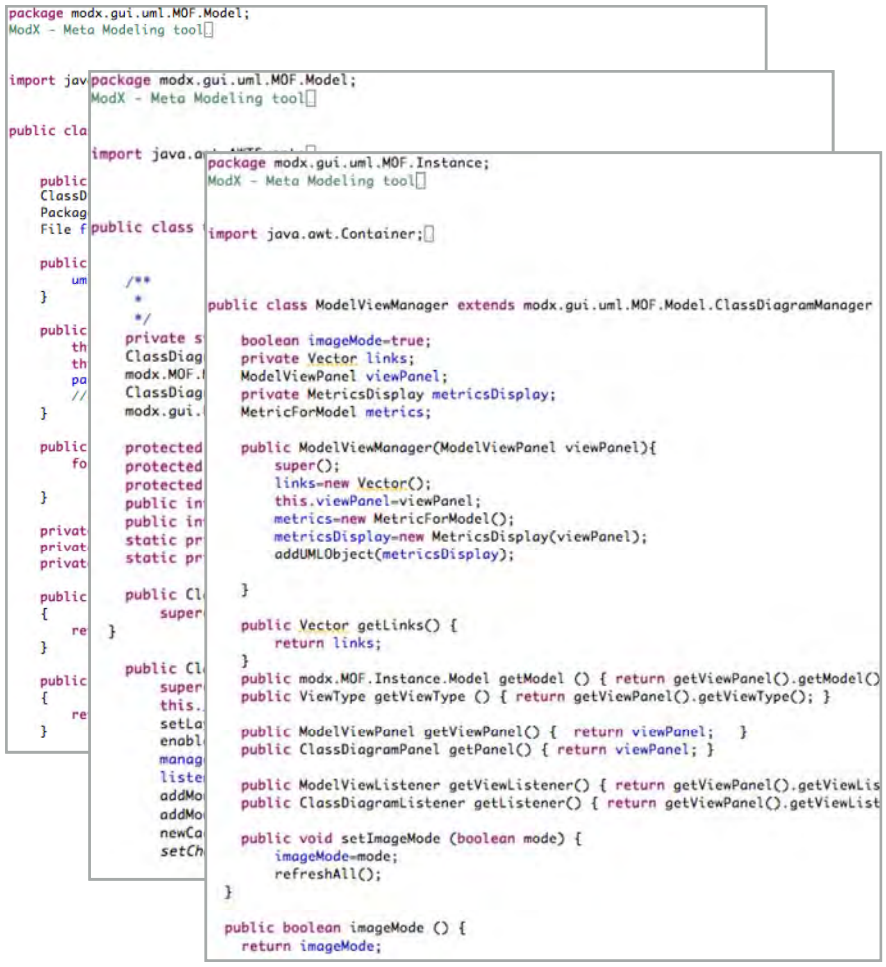




Software Engineering



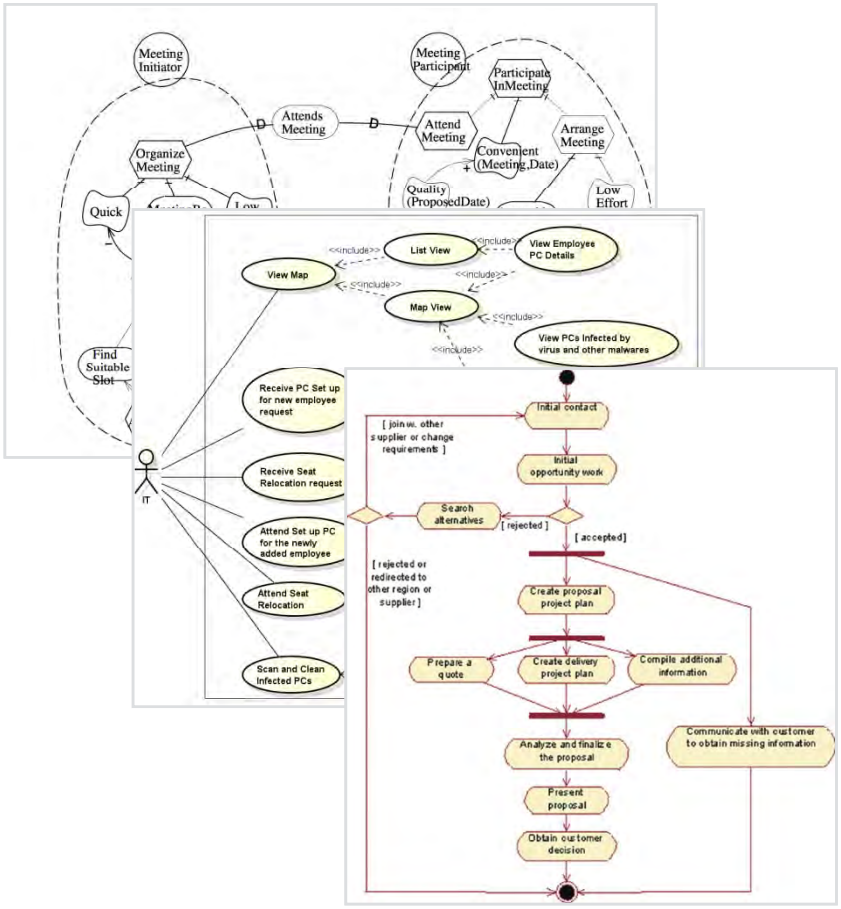
The Code rules

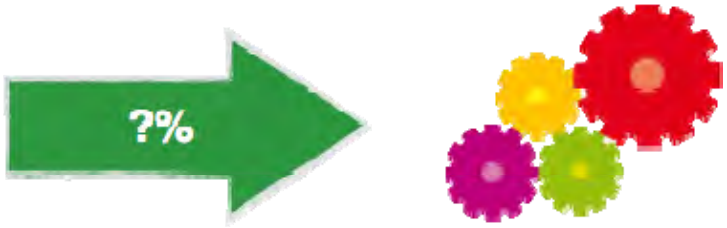


```
package modx.gui.uml.MOF.Instance;
ModX - Meta Modeling tool

import java.awt.Container;


public class ModelViewManager extends modx.gui.uml.MOF.Model.ClassDiagramManager

    boolean imageMode=true;
    private Vector links;
    ModelViewPanel viewPanel;
    private MetricsDisplay metricsDisplay;
    MetricForModel metrics;

    public ModelViewManager(ModelViewPanel viewPanel){
        super();
        links=new Vector();
        this.viewPanel=viewPanel;
        metrics=new MetricForModel();
        metricsDisplay=new MetricsDisplay(viewPanel);
        addUMLObject(metricsDisplay);

    }

    public Vector getLinks() {
        return links;
    }
    public modx.MOF.Instance.Model getModel () { return getViewPanel().getModel()
    public ViewType getViewType () { return getViewPanel().getViewType(); }

    public ModelViewPanel getViewPanel() {  return viewPanel;   }
    public ClassDiagramPanel getPanel() { return viewPanel; }

    public ModelViewListener getViewListener() { return getViewPanel().getViewLis
    public ClassDiagramListener getListener() { return getViewPanel().getViewList

    public void setImageMode (boolean mode) {
        imageMode=mode;
        refreshAll();
    }

    public boolean imageMode () {
      return imageMode;
```

The software

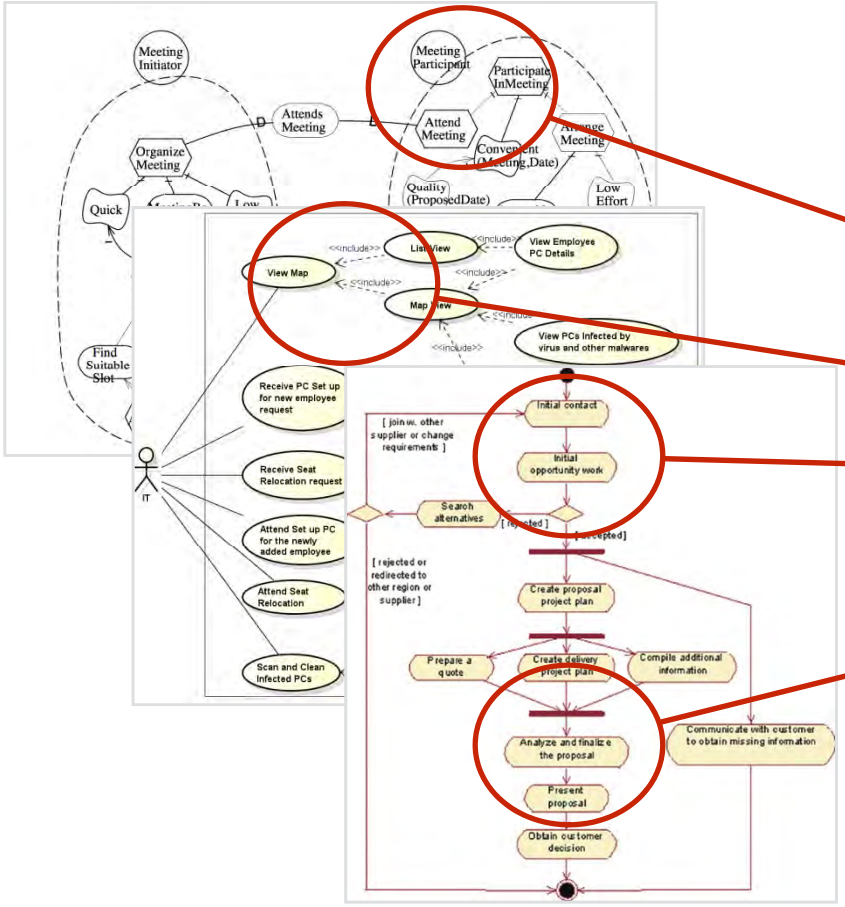
Models do not

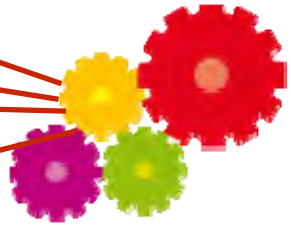




The software

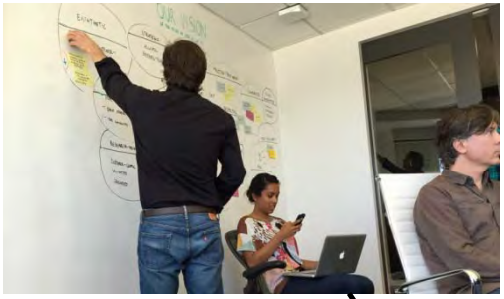
Todo: save the links

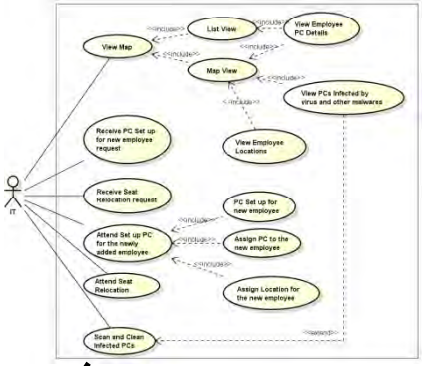


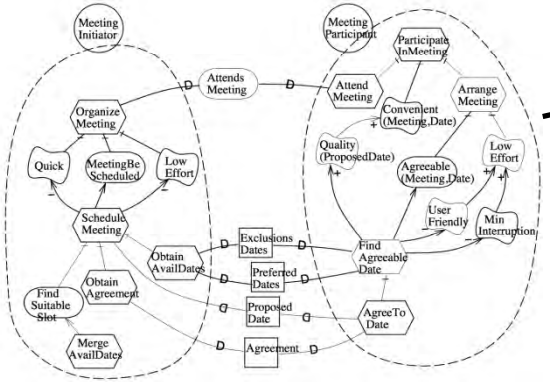


The software

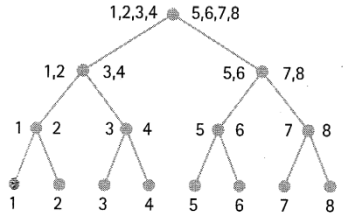
Graal of software engineering

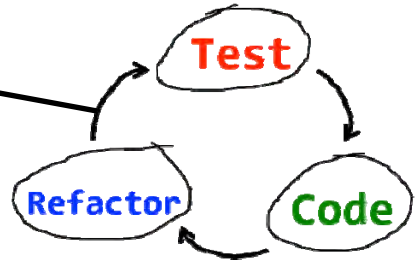




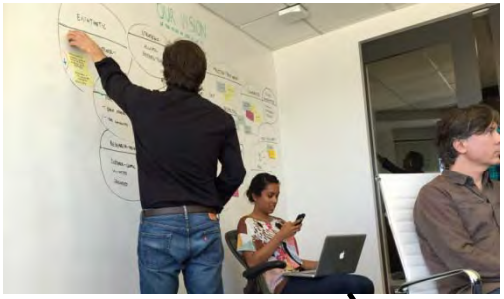


$$

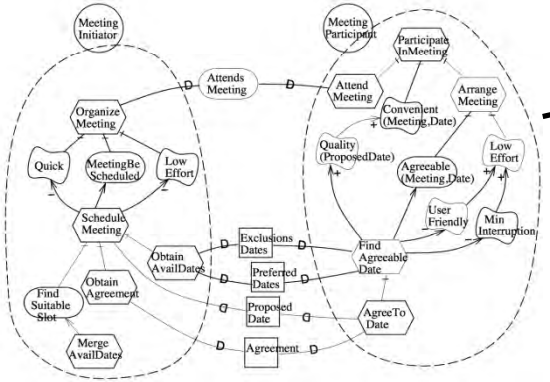


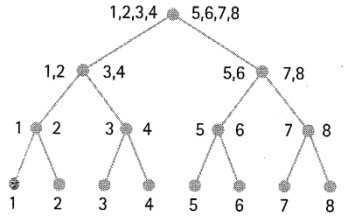


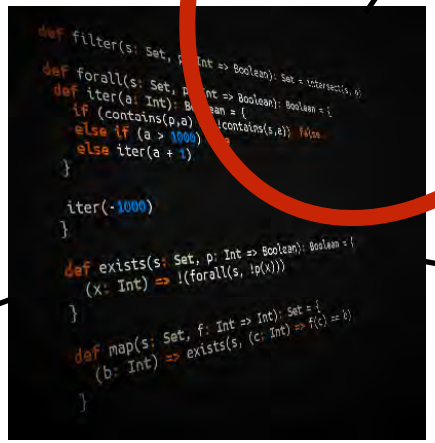
Graal of software engineering

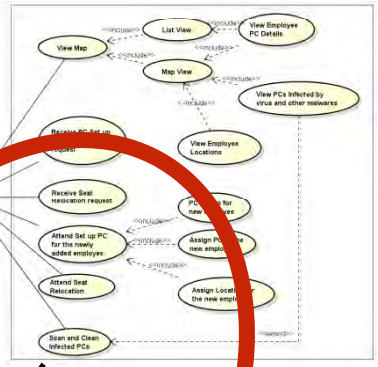


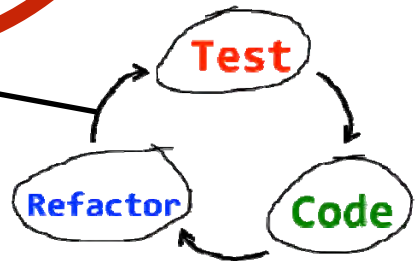
$$



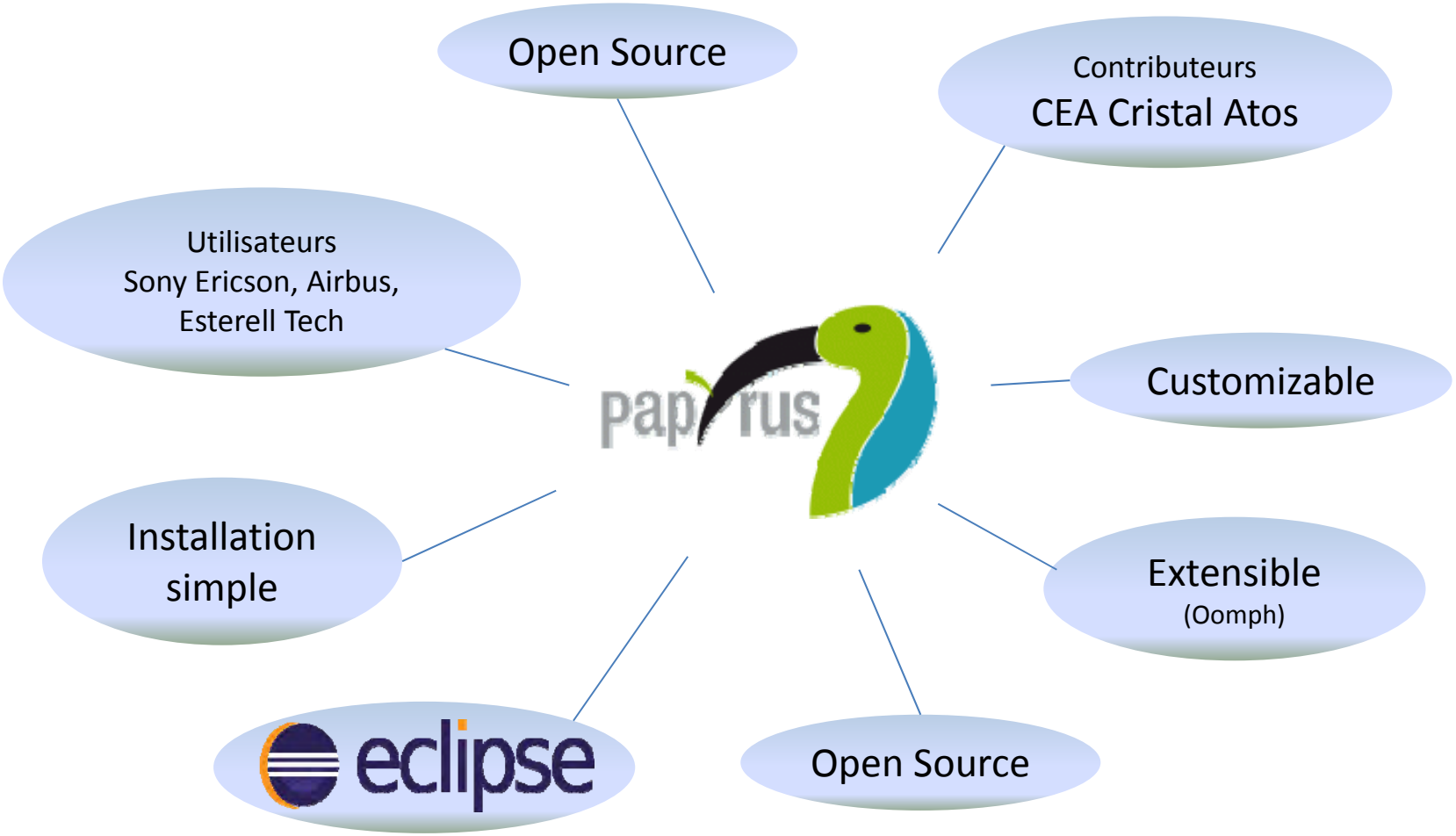








Papyrus


Open Source
Contributeurs
CEA Cristal Atos
Utilisateurs
Sony Ericson, Airbus,
Esterell Tech
papyrus
Customizable
Installation
simple
Extensible
(Oomph)
eclipse
Open Source

Actuellement…

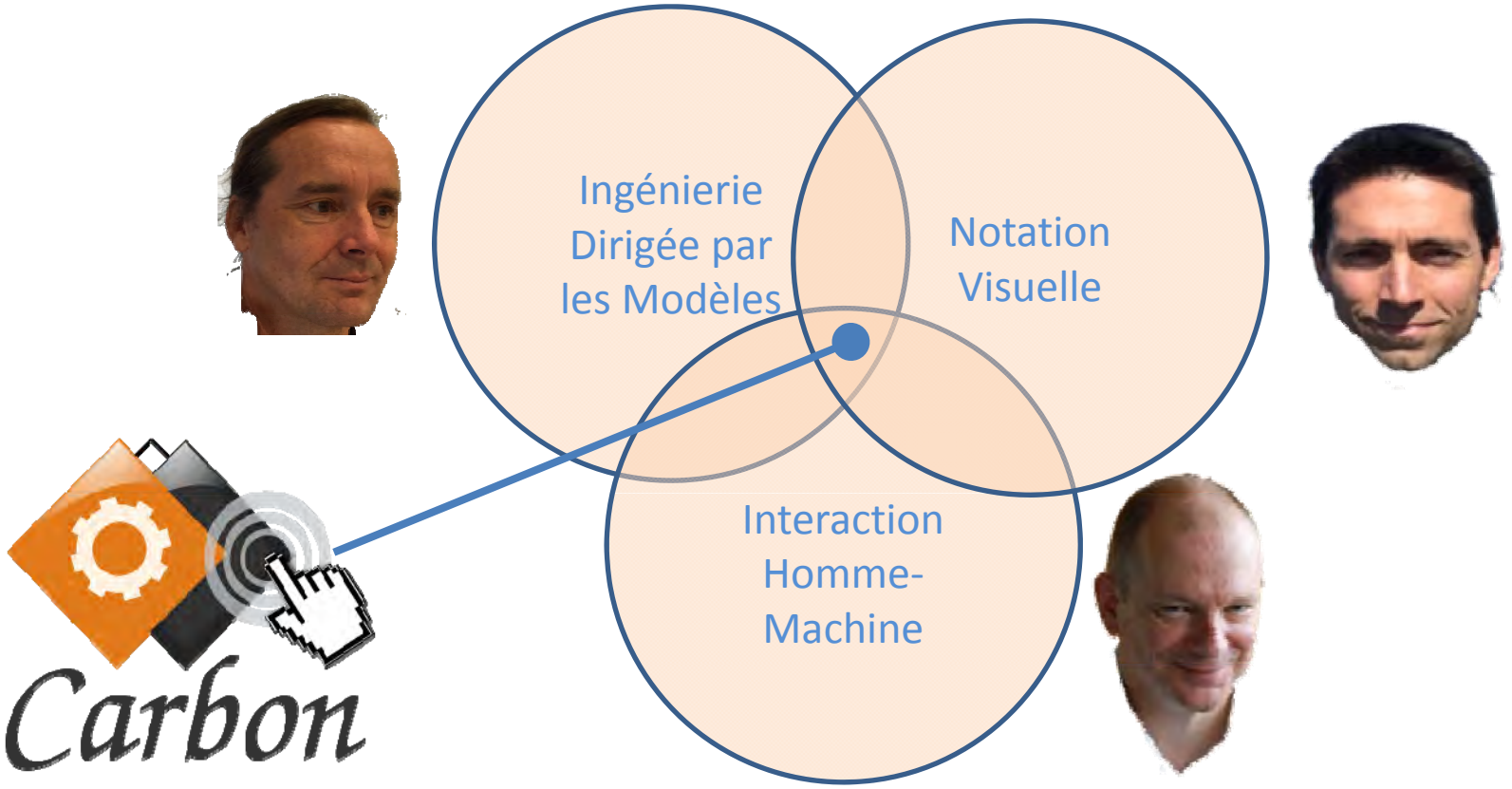
- **3 thèses en cours** (2 financements **CEA** + 1 CIFRE **Axellience**)
  - **Axellience/Michel Dirix** (mars 2013-2016) : *Ingénierie logicielle Dirigée par les Modèles et la Collaboration*
  - **CEA/Mickaël Duruisseau** (décembre 2014-2017) : *Améliorer l'activité de modélisation logicielle par de nouvelles techniques d'interactions*
  - **CEA/Yossr El Ahmar** (février 2015-2018) : *Improving Software Diagrams Efficiency for Developers. Application to Papyrus*

- **1 candidat « wanted »** (1 financement CEA) : *Améliorer les interactions entre sources d'informations hétérogènes, application à l'écosystème Papyrus*
  - https://sujets-these.lille.inria.fr/details.html?id=c32f4fbd044940c69554a603ecefa54a

- 1 thèse passée (Nadia Elouali, 2011-2014, bourse Ministère)
  - Sujet : Ingénierie dirigée par les modèles pour les systèmes pervasifs multimodaux

- Prototypes montrés à l'inauguration de CRIStAL

3 axes sous-jacents



Carbon

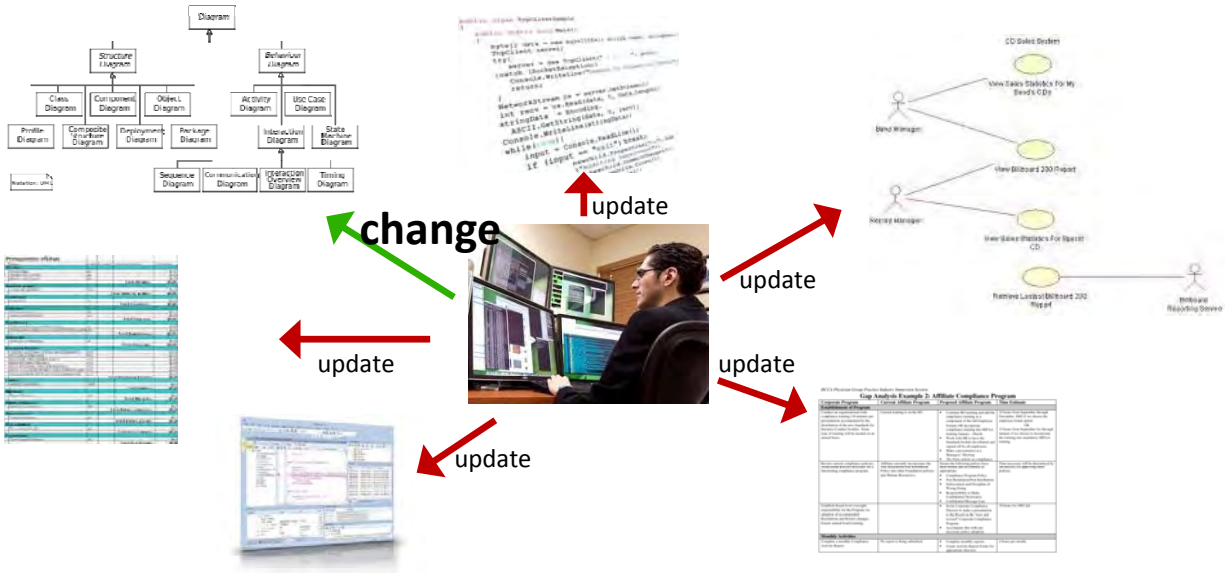
3 thèses avec le CEA

PhD #1

Improve interactions between heterogeneous information sources

PhD #1 - Improve interactions between heterogeneous information sources

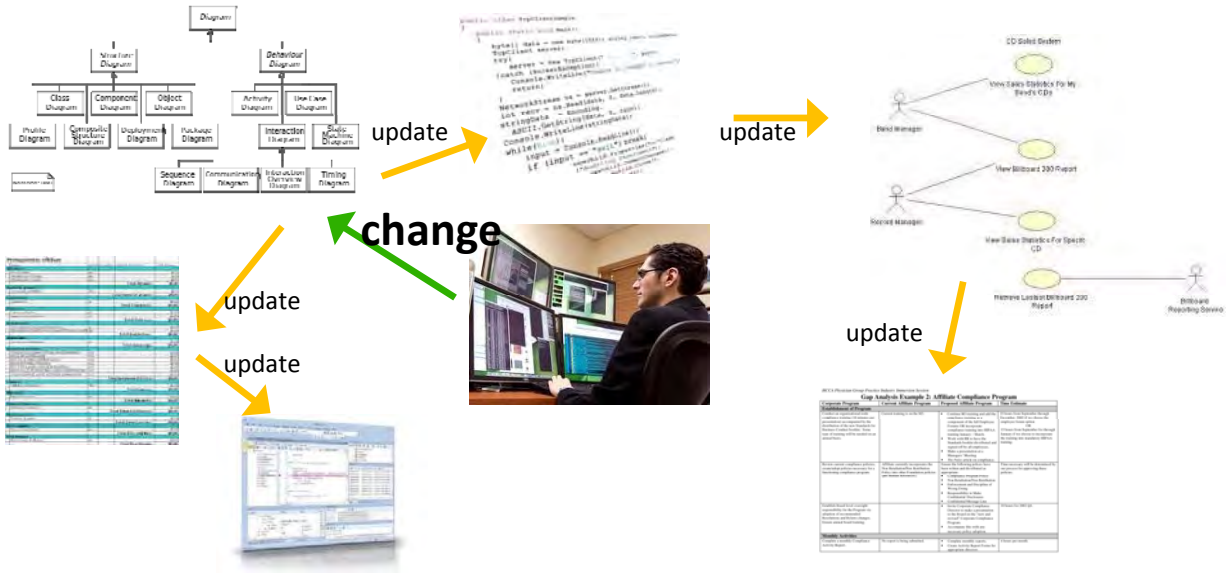
Problem



Software development produces a significant amount of information
Information are often related, changing one requires updating the others

Who has never facing an outdated documentation, with old screen captures, or where a menu entries don't exist anymore or have changed ?

Proposal
No more outdated document !



We want to propose mechanisms:

- to link information from different sources of information;
- to measure the impact of a change in an information source;
- to propagate changes;
- to maintain the consistency of information sources as automatically as possible.

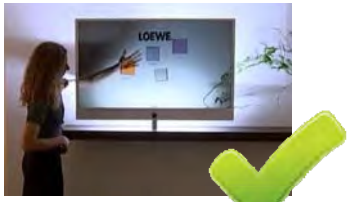
Carbon

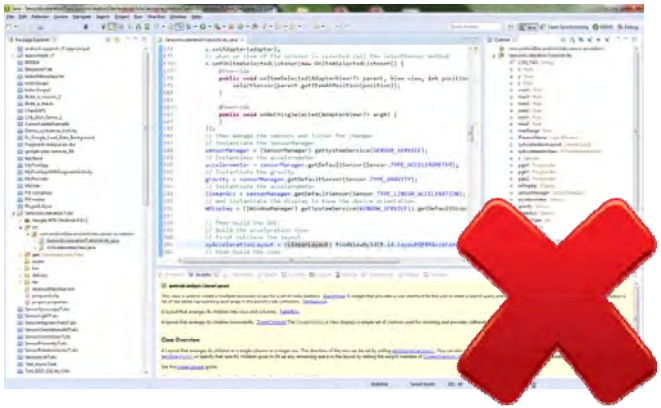
PhD #2

Enhance the activity of software engineering with new techniques of interaction

PhD #2 - Enhance the activity of software engineering with new techniques of interaction

Lot of **HCI improvements in our life…**





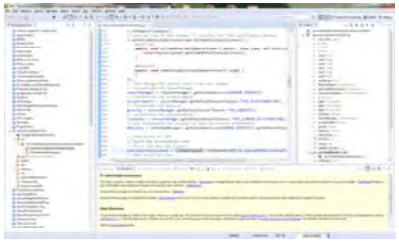
…but almost nothing for **developers still working only with textual IDE!**

PhD #2 - Enhance the activity of software engineering with new techniques of interaction

Study the **contribution of the Human-Computer Interaction** and associated technologies **to improve the work of developers** by focusing on the **creation and manipulation of contextual and interactive diagrams** associated with **daily activities**.

and the more adapted interaction devices

Find the better activity to indirectly change model



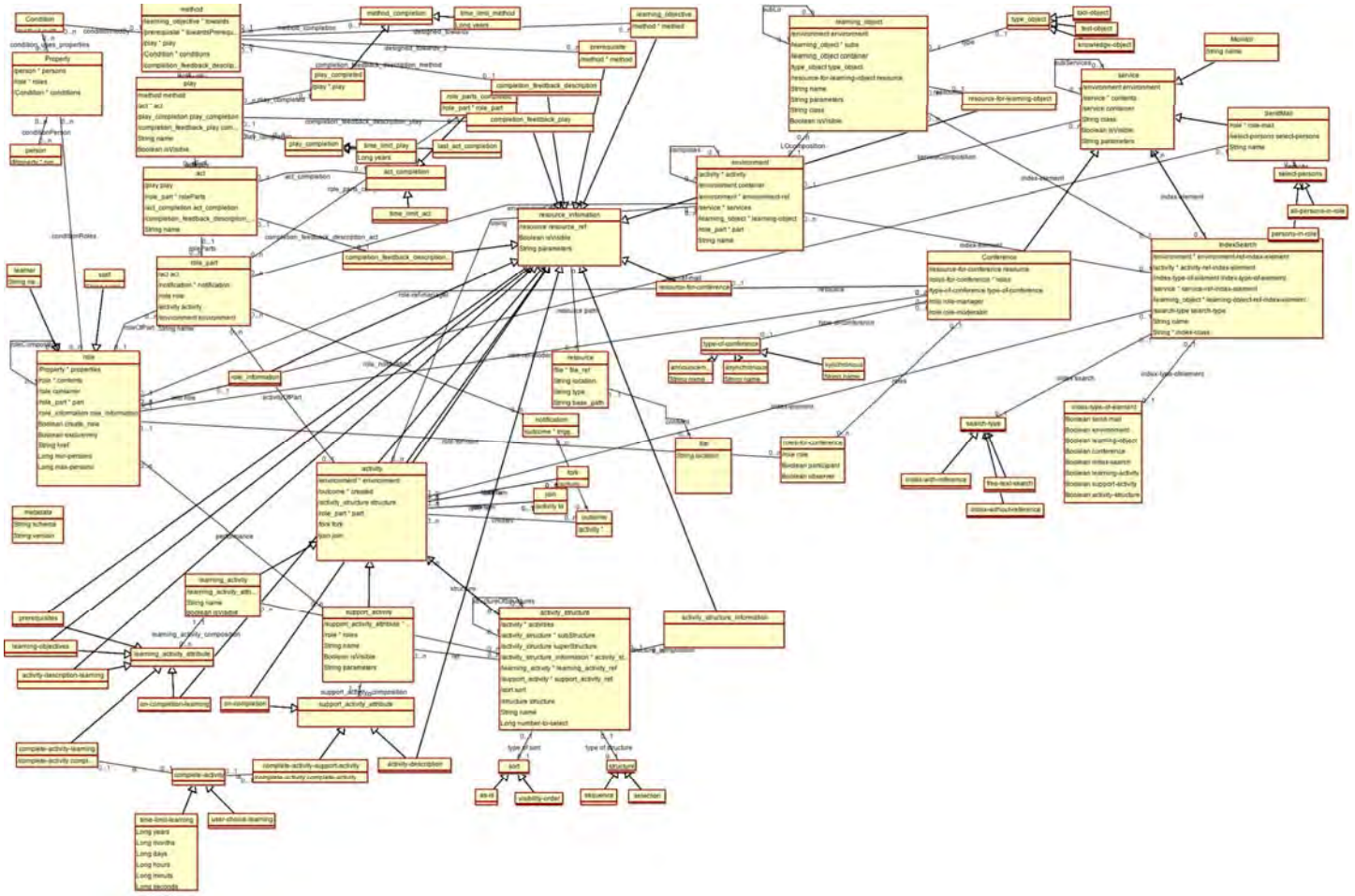
Carbon

PhD #3

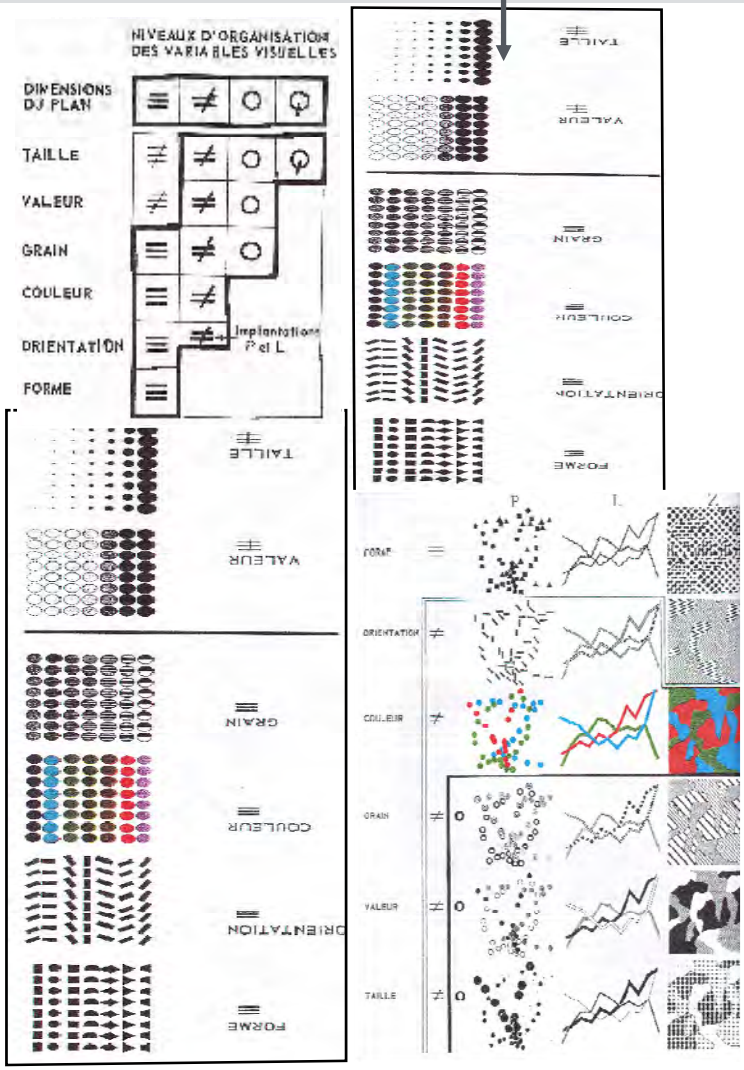
Semiologicaly meaningful diagrams for an efficient MDE

PhD #3 - Semiologicaly meaningful diagrams for an efficient MDE

Software diagrams are generally not transformed to fit their context of use



Semiology of Graphics (SoG) can help to adapt diagrams…



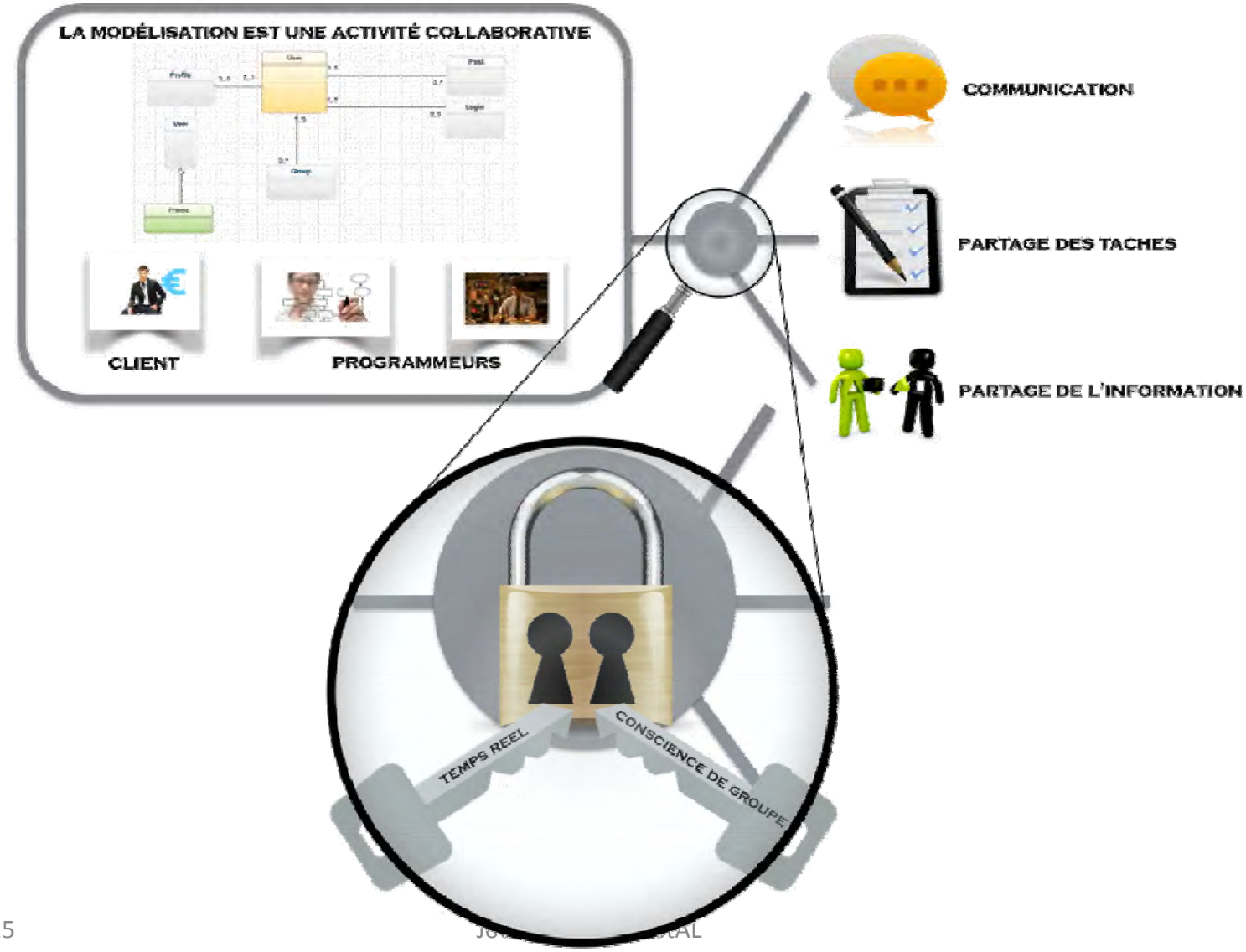
… but one needs to

- Study real tasks/concerns of software practitioners
- Adapt SoG to Software Modeling

Thèse avec Axellience

Sujet de thèse CIFRE avec Axellience

Ingénierie logicielle Dirigée par les Modèles et la Collaboration



Carbon

Questions ?