

# Génie Logiciel



Master Management de la Sécurité des  
Systèmes Industriels et des Systèmes  
d'Information

Adrien Gosseaume – Julien Vehent  
promo 2006/2007





# Diagramme de classes de l'Université de Poitiers

- Les classes

**1. l'Université de Poitiers**

Université

2. l'ascenseur

Ecole doctorale

3. le Scrabble

Services généraux

4. Versioning

Laboratoire

Composante

Département

Formation

Unité d'Enseignement

Cours

Intervenant extérieur

Enseignants-chercheur

Enseignant

Equipe

Etudiant



# Diagramme de classes de l'Université de Poitiers

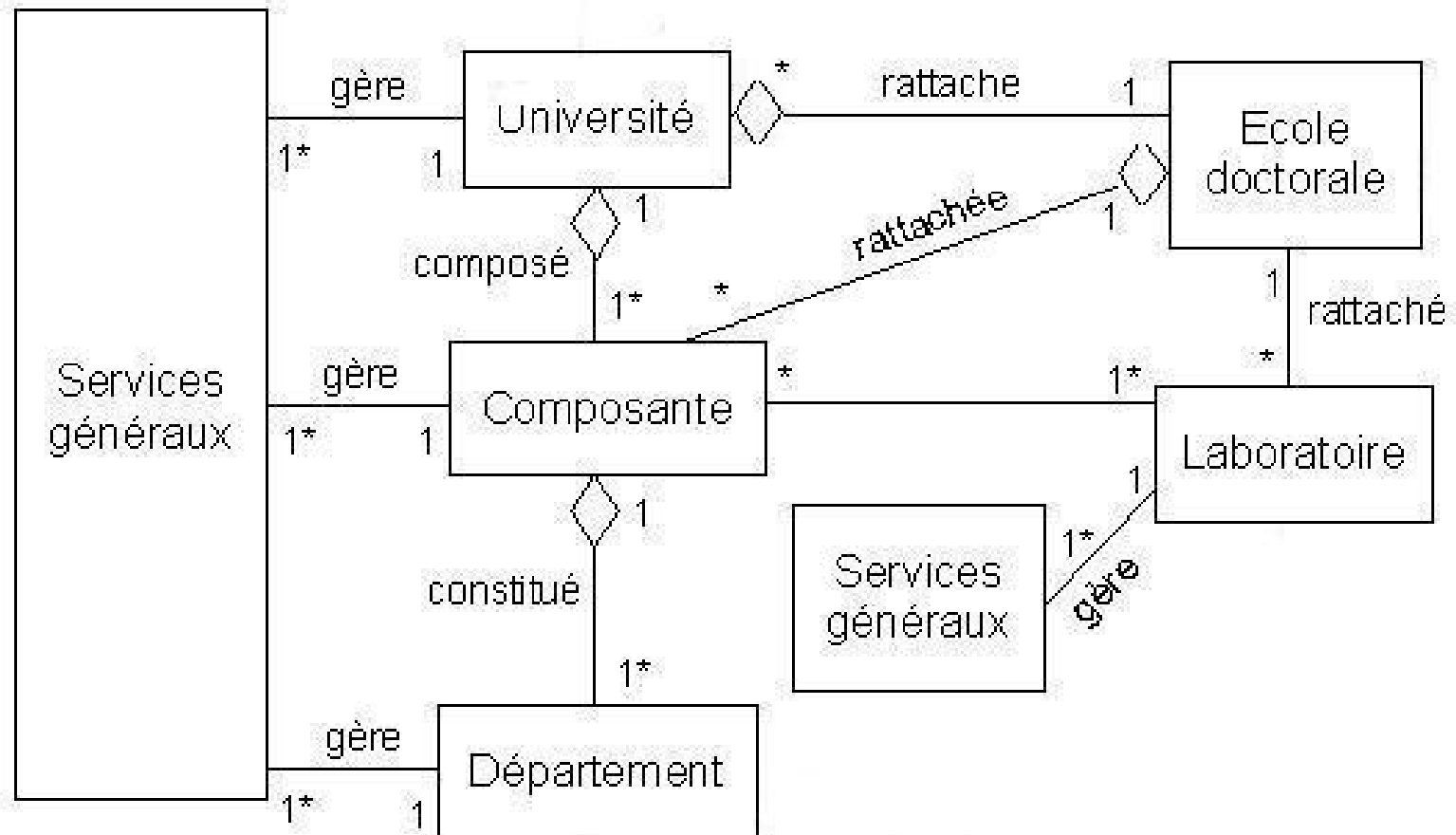
- Autour de la classe université...

## 1. l'Université de Poitiers

## 2. l'ascenseur

## 3. le Scrabble

## 4. Versioning







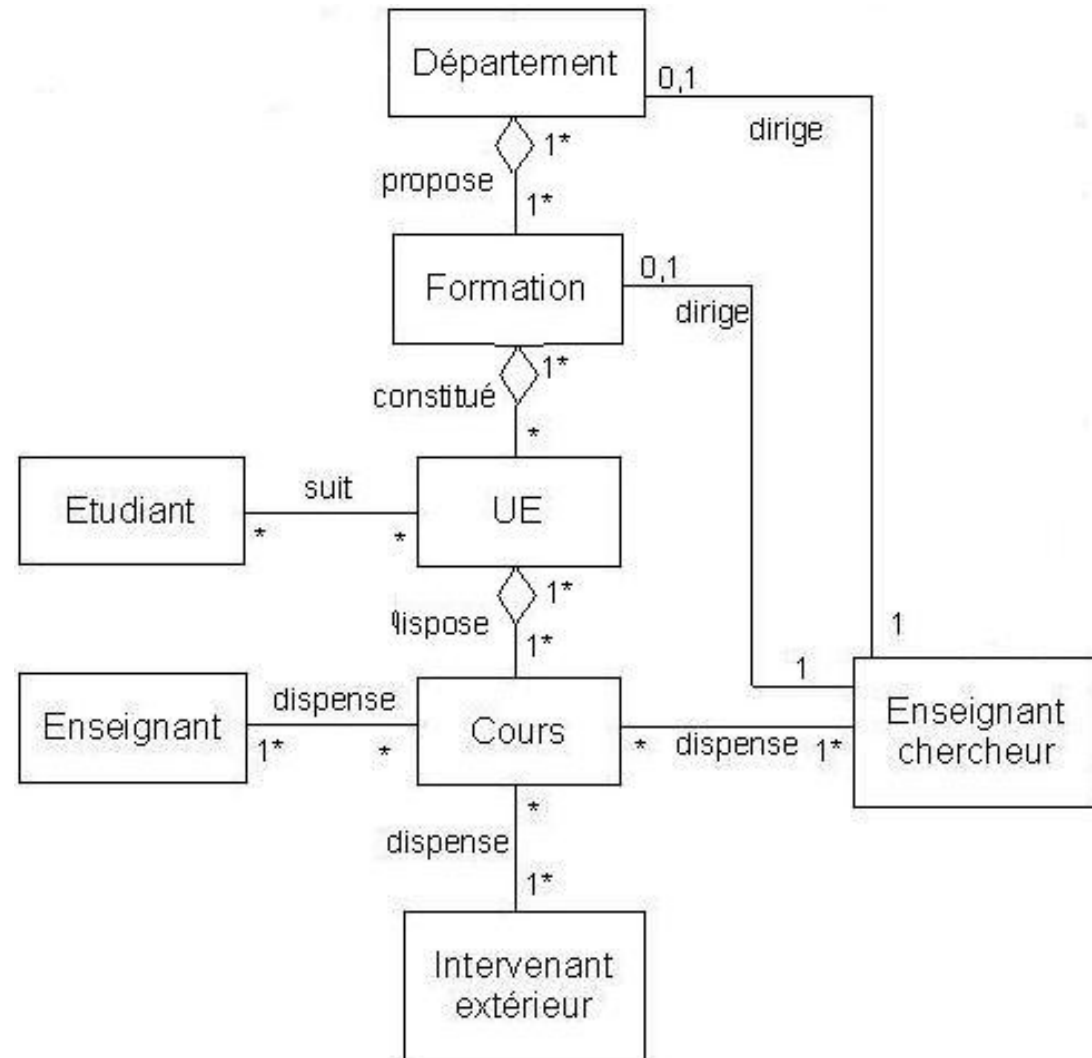
# Diagramme de classes de l'Université de Poitiers

## 1. l'Université de Poitiers

## 2. l'ascenseur

## 3. le Scrabble

## 4. Versioning





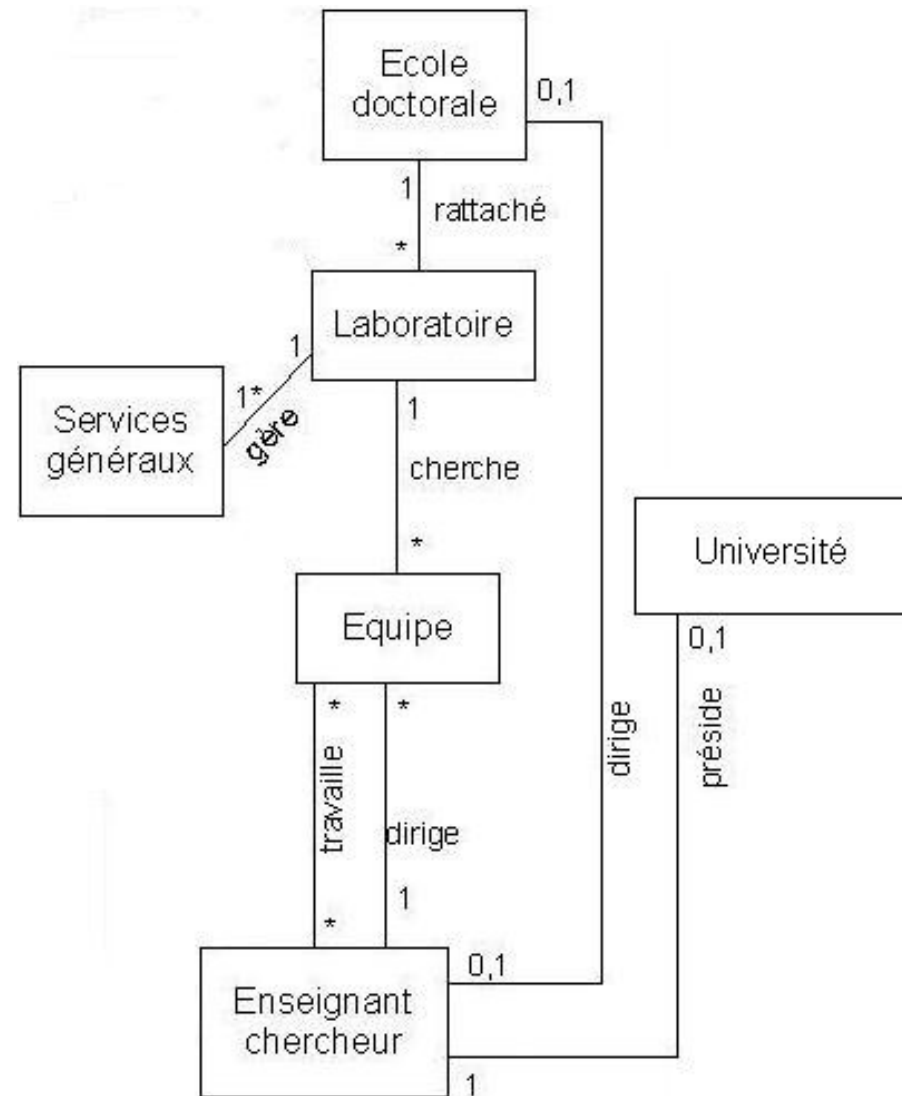
# Diagramme de classes de l'Université de Poitiers

## 1. l'Université de Poitiers

## 2. l'ascenseur

## 3. le Scrabble

## 4. Versioning



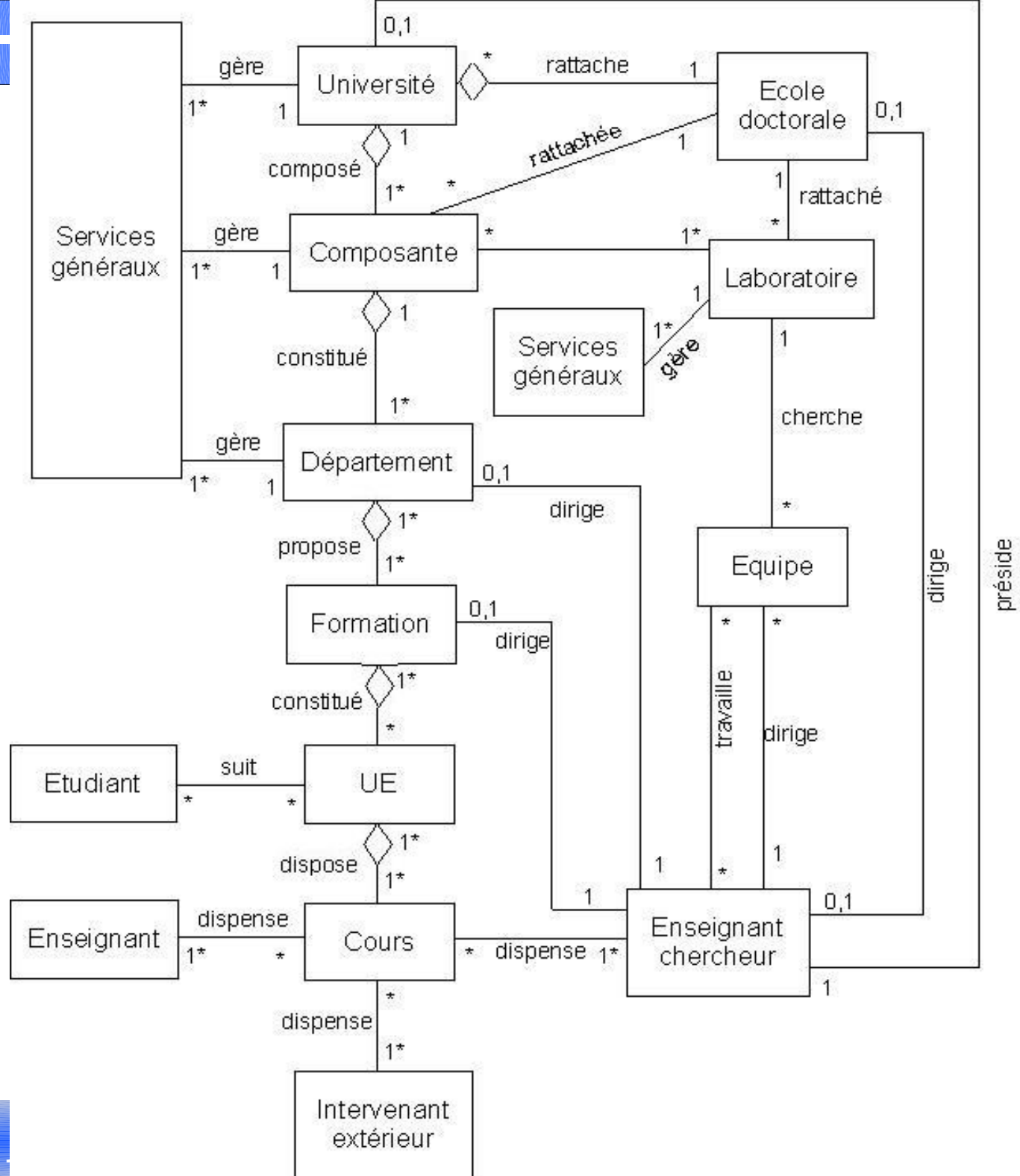


# 1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning





# Diagramme Etat-Transitions de l'ascenseur

- Hypothèses de fonctionnement
  - ascenseur simple
  - gestion de la destination par file d'attente
  - gestion de la surcharge
  - choix rationnel de la destination
- Objectif
  - Modéliser le fonctionnement de la cabine munie d'une ouverture dans une cage d'ascenseur possédant des boutons à l'intérieur et à l'extérieur de la cabine

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning





# Diagramme Etat-Transitions de l'ascenseur

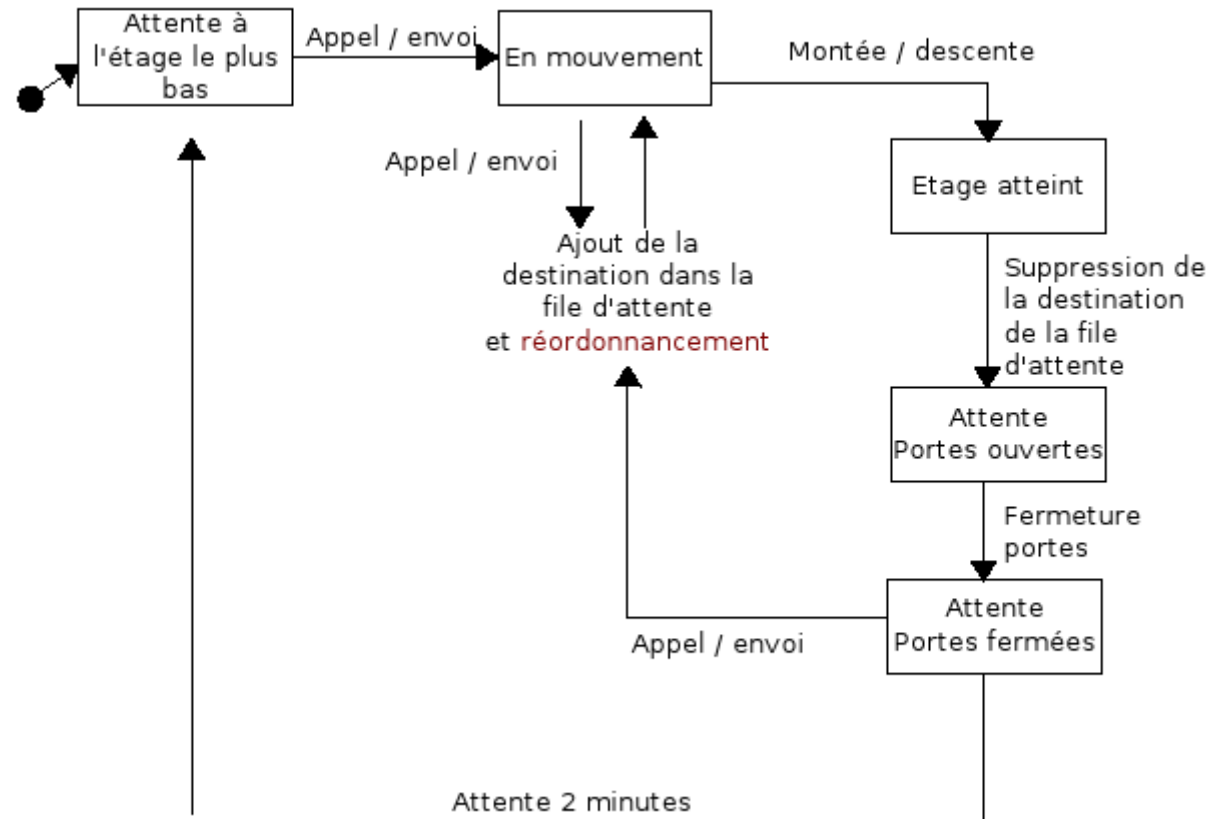
- Première version

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning





# Diagramme Etat-Transitions de l'ascenseur

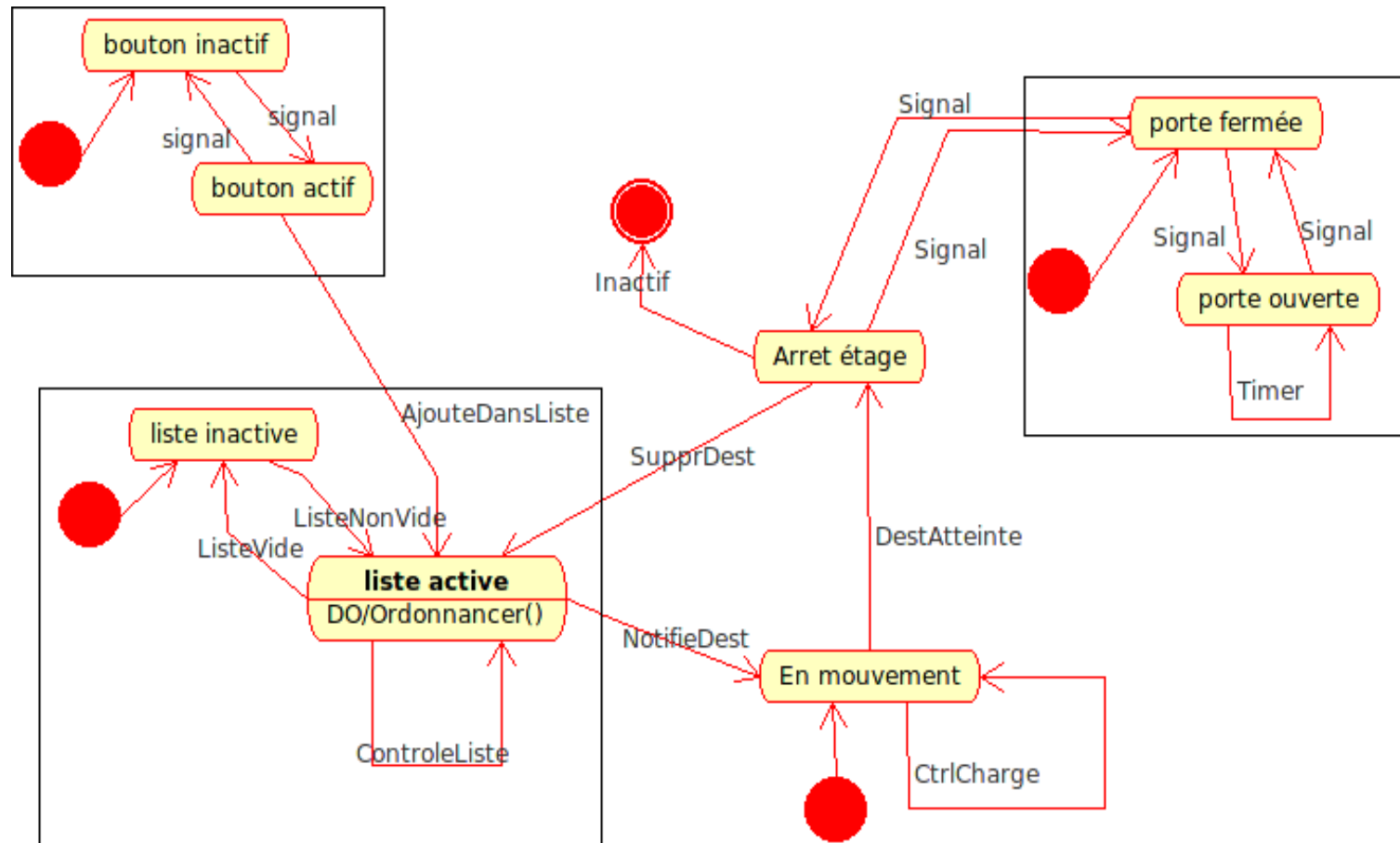
- Généralisation

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning





# Diagramme Etat-Transitions de l'ascenseur

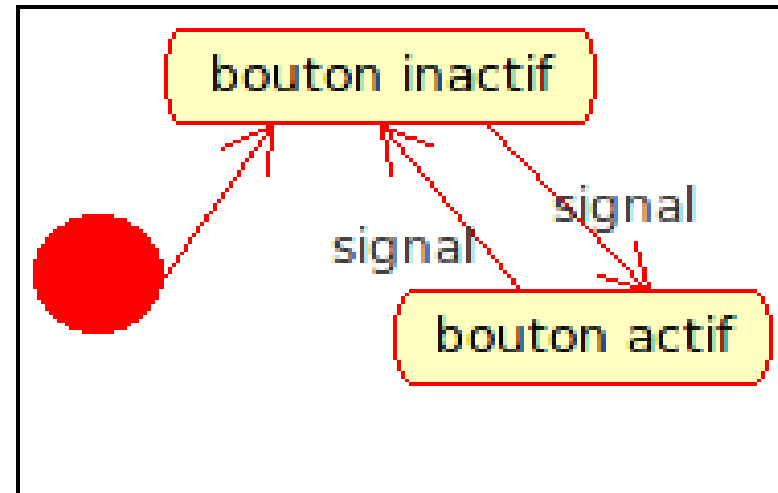
- Gestion des boutons

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

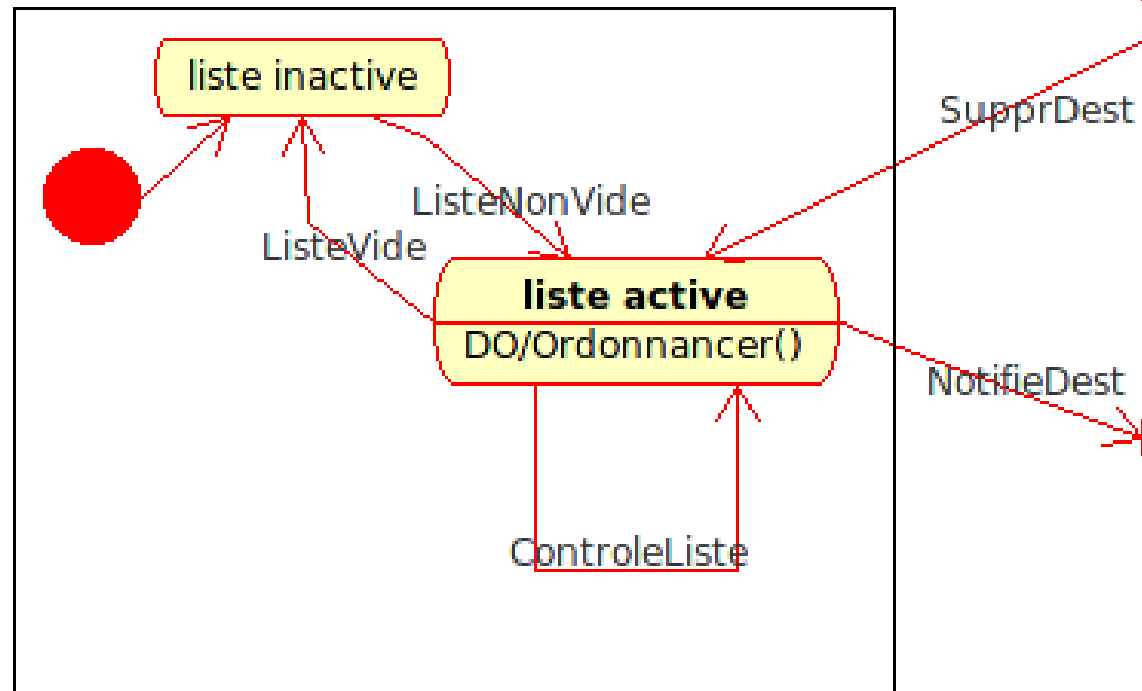


*passer un bouton de l'état inactif à l'état actif et envoi un signal du bouton vers la liste des boutons actifs*



# Diagramme Etat-Transitions de l'ascenseur

- gestion de la liste



*Mémoire les boutons actifs et les ordonnancement pour choisir une destination notifié à l'ascenseur*

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning





# Diagramme Etat-Transitions de l'ascenseur

- L'ordonnancement
  - fonction de la liste des boutons actifs utilisée pour choisir une destination

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

Si (( $O = \text{montant ET } P < E < D$ ) OU  $D = \text{null}$ )  
alors  $E$  devient  $D$ .

Sinon

Si ( $O = \text{descendant ET } P > E > D$ )  
alors  $E$  devient  $D$ .

avec

*O le sens actuel de l'ascenseur*

*P la provenance de l'ascenseur*

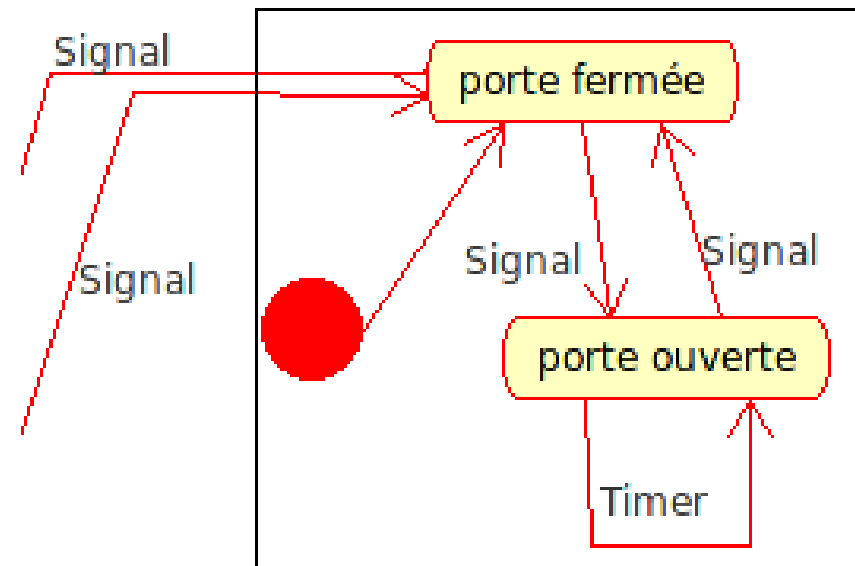
*D la destination actuelle de l'ascenseur*

*E la nouvelle entrée testée*



# Diagramme Etat-Transitions de l'ascenseur

- gestion de la porte



*Une porte s'ouvre avec un signal, un timer se déclenche et la porte se ferme à la fin du timer.*

1. l'Université de Poitiers

2. l'ascenseur

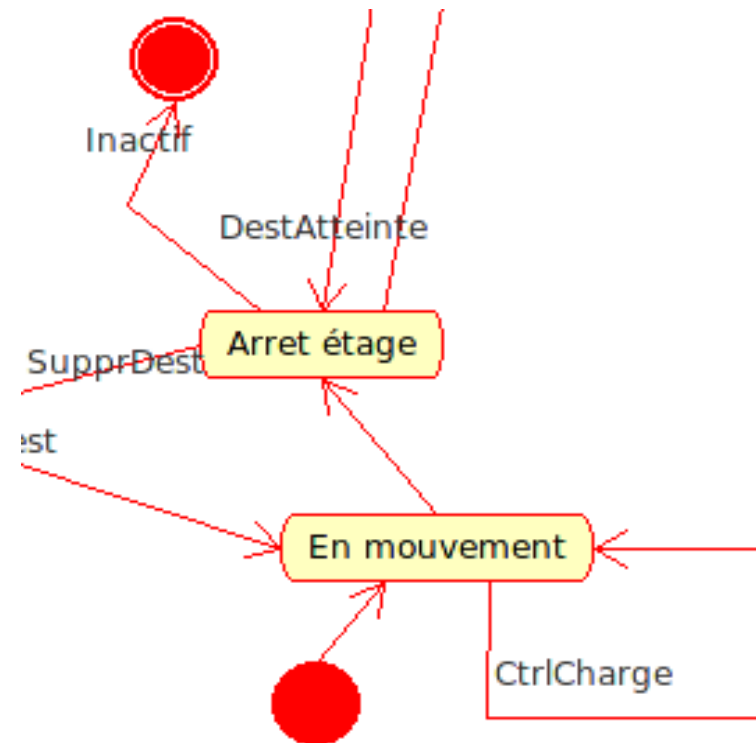
3. le Scrabble

4. Versioning



# Diagramme Etat-Transitions de l'ascenseur

- La cabine



*Se met en mouvement avec une destination si la charge est correcte, supprime la destination lors de la fermeture des portes et retourne a son point initial après un timer d'inactivité*

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning



# Diagramme Etat-Transitions de l'ascenseur

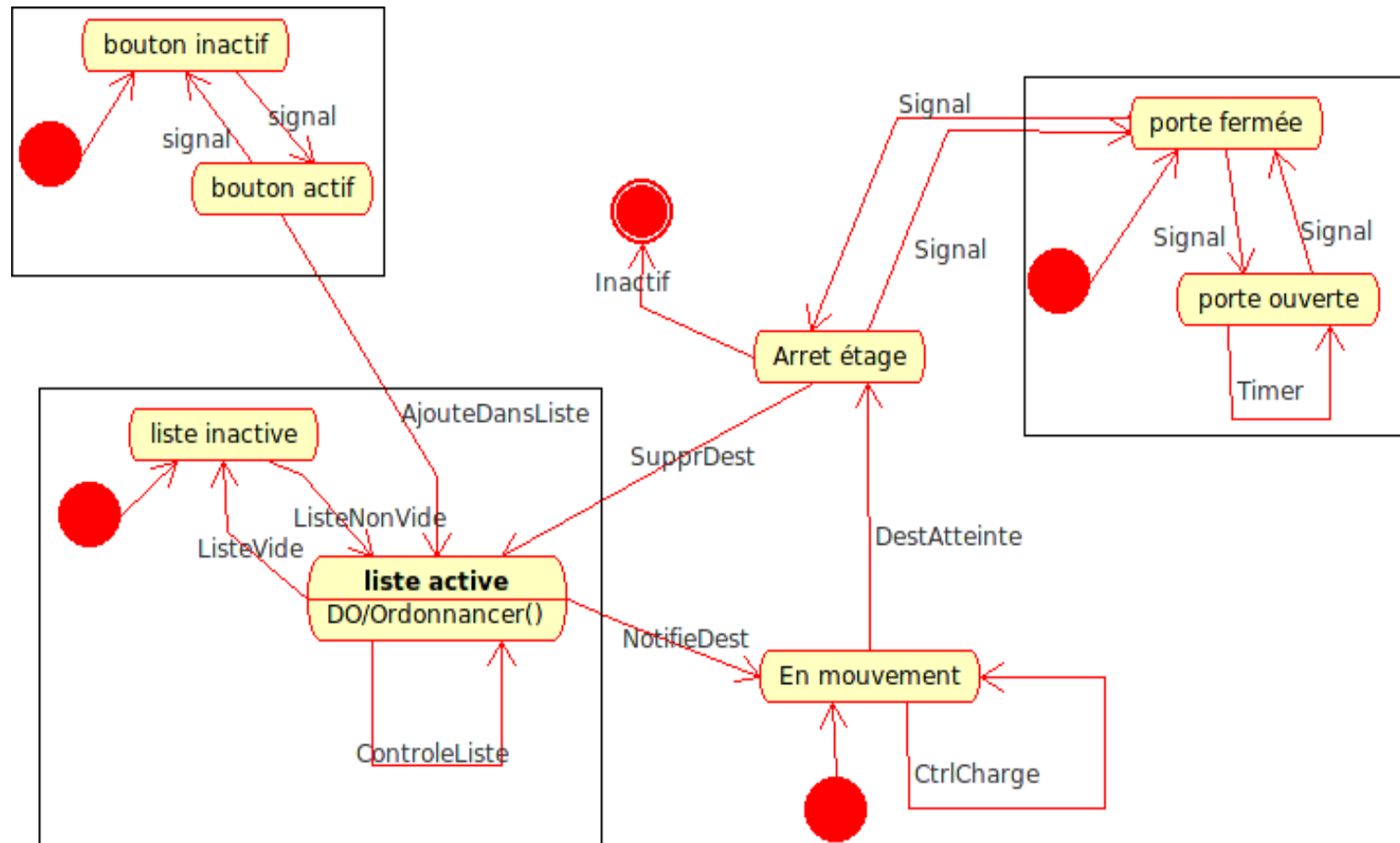
- Retour sur la généralisation

1. l'Université de Poitiers

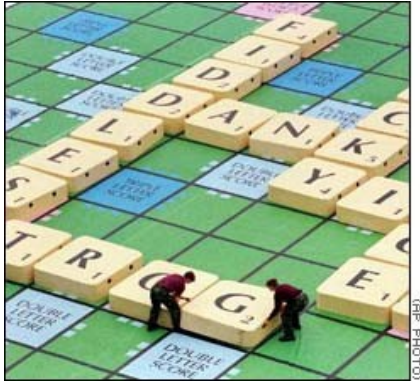
2. l'ascenseur

3. le Scrabble

4. Versioning







# Paradigmes de programmation du Scrabble

- Paradigme utilisé : Sérialisation

Les actions du jeu sont régulières.

Les étapes se succèdent en boucle dans le même ordre.

-> On pourra donc exécuter des fonctions en série.

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning



# Paradigmes de programmation du Scrabble

- Fonctions d'initialisation

Initialisation de la pioche

Initialisation du chevalet

Initialisation de la grille de jeu

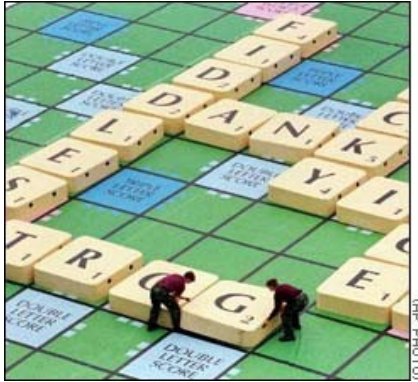
Initialisation des coefficients de la grille (mot+lettre)

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning



# Paradigmes de programmation du Scrabble

- Fonctions du jeu

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning

Gestion du chevalet du joueur

Placer un mot

Vérifications sur le mot posé

Compter les points d'un mot posé

Tirage des lettres et remplissage du chevalet du joueur



# Paradigmes de programmation du Scrabble

- Vérifications

Il y a une série de tests à exécuter avant la validation du mot et avant de compter les points.

1) Le joueur ne peut pas positionner de lettres en dehors des limites du plateau  
 -> Vérification des coordonnées saisies

2) Vérifier que le lettre jouée est sur le chevalet du joueur et qu'il ne s'en sert pas plusieurs fois..

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning





# Paradigmes de programmation du Scrabble

- Vérifications

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning

3) Vérifier si le mot ainsi constitué est bien en ligne (verticale ou horizontale)

4) Vérifier si ce mot s'appuie sur au moins une lettre déjà posée

5) Vérifier sur un dictionnaire l'existence du mot

-> L'échec d'un test suffit pour faire remonter une action de jeu non conforme au règlement, le joueur doit replacer un autre mot.



# Paradigmes de programmation du Scrabble

- Compter les points

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning

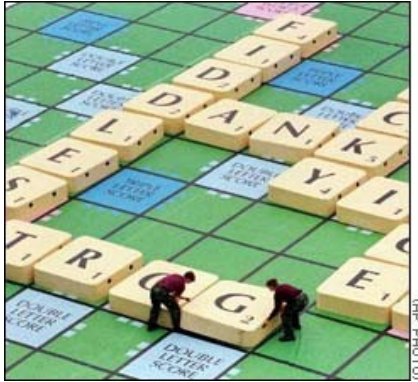
## Paradigme utilisé : **Affaiblissement – renforcement**

### L'affaiblissement :

On considère le mot orienté de façon horizontale.  
On compte uniquement le mot posé.

### Le renforcement :

Rotation du plateau si c'est un mot vertical  
On compte les mots qui sont croisés



# Paradigmes de programmation du Scrabble

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning

```

Somme=0, coefficients=1, i=abscisse, j= ordonnée
coef_mot=1
Tant que la case(i,j) n'est pas vide
    somme = somme + valeur_plateau(i) *
grille_coef_lettre(i,j)
    grille_coef_lettre(i,j)=1 // coef à 1 car plus utilisé
    coef_mot=coef_mot*grille_coef_mot(i,j)
    i=i+sens
    coefficients=coefficients*tableau.mot(i,j)
Fin tant que

somme = somme * coef_mot
total_points_joueur = total_points_joueur + somme
  
```



# Paradigmes de programmation du Scrabble

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning

On renforce en intégrant dans la boucle le calcul des mots croisés

\* SI  $\text{case}(i,j+1)$  n'est pas vide ALORS compter mot en sens inverse à partir de la case  $(i,j+1)$

\* SI  $\text{case}(i,j-1)$  n'est pas vide ALORS compter mot en sens inverse à partir de la case  $(i,j-1)$



# Paradigmes de programmation du Scrabble

- Paradigmes

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning

Le règlement du jeu définit des méthodes de comptage compliquées.

L'affaiblissement est très pratique pour simplifier le problème et avoir une meilleure idée de ce que l'on code.

Reprendre un par un les éléments ignorés au départ est ensuite plus simple.





# Paradigmes de programmation du Scrabble

1. l'Université de Poitiers

2. l'ascenseur

**3. le Scrabble**

4. Versioning

Les lettres jouées sont retirées et le mot est enregistré sur le plateau de jeu.

Le chevalet est rempli par la pioche s'il reste des lettres dedans.

On teste si il s'agit de la fin du jeu sinon on recommence un tour de jeu avec le joueur suivant.

# Versioning



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

**4. Versioning**

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

- Egalement appelé « contrôle de révision » (Revision Control)
  - C'est la gestion de multiple version d'une même unité d'information
- Problématique de stockage
  - plusieurs développeurs travaillant sur le même programme en même temps
- 3 principes majeurs
  - Blocage de fichiers
  - Contrôle de versions distribué
  - Fusion de versions

# Versioning



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. **Versioning**

| -Presentation

| -GIT

| -Subversion

| -Trac

` Et les autres

- Contrôle de versions distribué
  - Chaque développeur possède une copie complète du dépôt
  - Les changements sont propagés par fusions entre les dépôts lors d'une étape optionnelle
- Pas de serveur central
  - Améliore la rapidité
  - Facilite le contrôle d'accès
- Comparable à un système de fichier
- Utilisé pour le développement de Linux



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

# GIT

- GIT n'est pas un système de gestion de logiciels, c'est plus un ensemble d'outils bas niveau permettant de construire un système de contrôle de version
  - Sa principale tâche est de gérer l'évolution du contenu d'une arborescence
  - Indexe les fichiers d'après leur somme de contrôle
  - Les changements sont propagés par fusions entre les dépôts lors d'une étape optionnelle
- Linus Torvalds explique que, « *par bien des aspects, vous pouvez considérer git comme un système de fichiers* »



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

# GIT

- GIT utilise une base d'objets et un répertoire de cache
- La base peut contenir 3 types d'objets
  - BLOB (Binary Large Object) : le contenu d'un fichier
  - TREE : une collection des BLOBs avec leurs noms et permissions
  - COMMIT : l'historique d'un arbre. Il contient un journal, un arbre et des pointeurs vers les COMMITS parents
- Le répertoire de cache est un fichier binaire contenant un objet TREE



# Versioning



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. **Versioning**

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

- Fusion de versions
  - Plusieurs développeurs éditent le même fichier, chaque changement est fusionné et sauvegardé
  - Permet de conserver un historique des versions
- Architecture Client/Serveur
  - Facile à implémenter/maintenir
  - Intégré dans les IDE (eclipse, anjuta, ...)
- Comparable à un serveur de fichier

# Subversion



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. **Versioning**

| -Presentation

| -GIT

| -Subversion

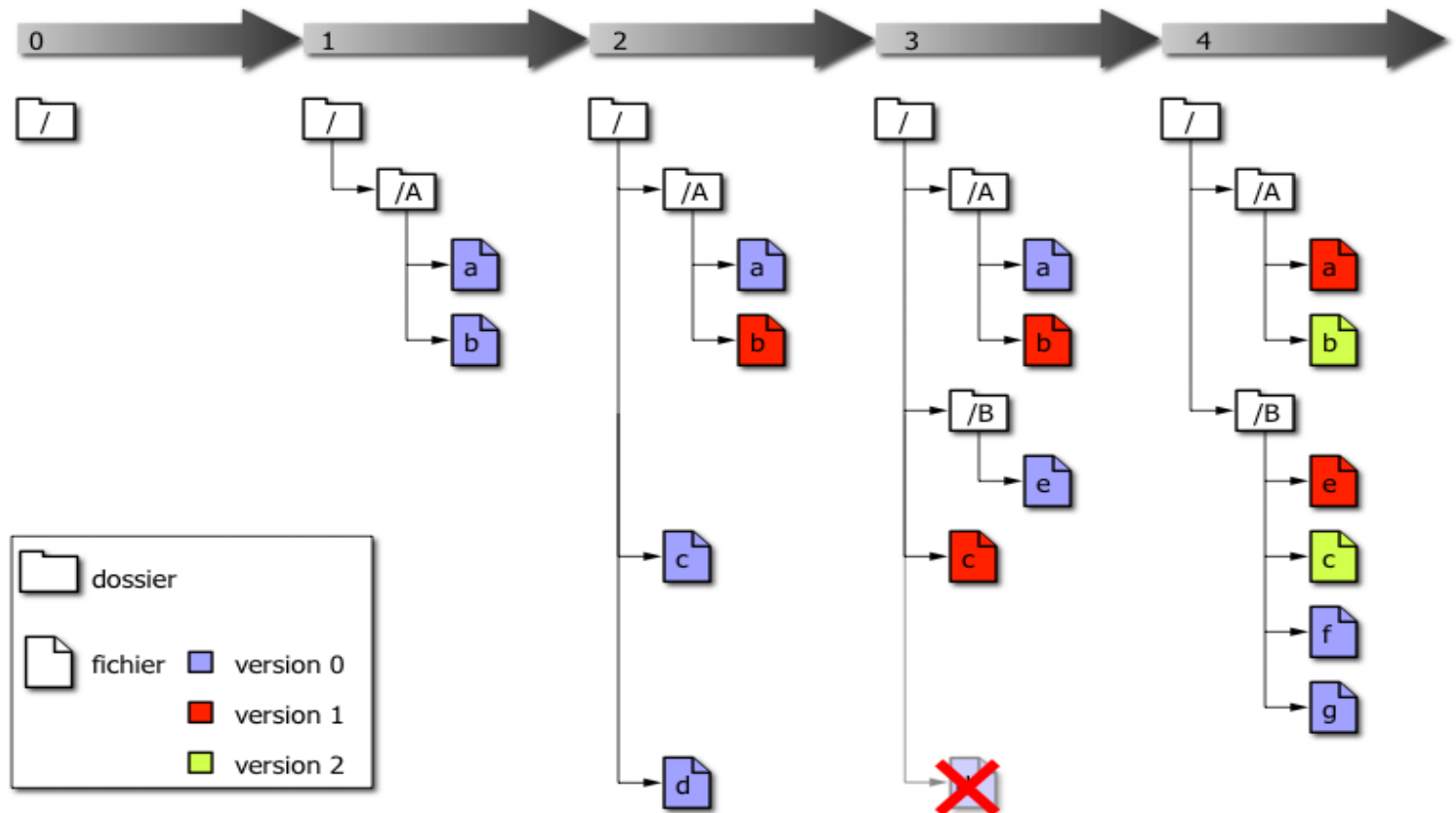
| -Trac

\ Et les autres

- Subversion vient du latin subvertere : renverser.
- Logiciel libre de gestion de versions publié sous licence GPL
- Remplaçant de CVS -> même concept mais meilleure implémentation
- projet lancé en février 2000 par CollabNet

# Subversion

- Principe de fonctionnement



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

| Et les autres



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

| Et les autres

# Le dépôt Subversion

```

-- test
|-- README.txt
-- conf
| |-- authz
| |-- passwd
| `-- svnserve.conf
-- dav
| `-- activities
-- db
| |-- current
| |-- format
| |-- fs-type
| |-- revprops
| |-- 0
| |-- 1
| |-- 2
| |-- 3
| `-- 4
| |-- revs
| |-- 0
| `-- 1
| `-- 2
| `-- 3
| `-- 4
| `-- transactions
| `-- uuid
| `-- write-lock |-- format
-- hooks
| |-- post-commit.tmpl
| |-- post-lock.tmpl
| |-- post-revprop-change.tmpl
| |-- post-unlock.tmpl
| |-- pre-commit.tmpl
| |-- pre-lock.tmpl
| |-- pre-revprop-change.tmpl
| |-- pre-unlock.tmpl
| `-- start-commit.tmpl
-- locks
| |-- db-logs.lock
| `-- db.lock
  
```

# Subversion



- Fonctionnalités et améliorations par rapport à CVS:
  - fonctionnement en dépôts
  - **commit atomique** (*commit racine*)
  - Gestion des répertoires.
  - Possibilité de renommer et déplacer des fichiers.
  - Ajout de méta-données sur les fichiers et historique de celles-ci.
  - Possibilité d'utilisation d'Apache comme serveur et de WebDAV /DeltaV comme protocole.

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. **Versioning**

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres



# Subversion



- Implémentation dans Apache 2
  - **Utilisation du protocole WebDAV**
    - Extension d'HTTP définie par l'IETF (RFC 2518).
    - WebDAV permet de simplifier la gestion de fichiers avec des serveurs distants. Il permet de récupérer, déposer, synchroniser et de publier des fichiers.
    - Rend possible l'écriture à travers le web
    - Sait gérer les droits d'accès

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. **Versioning**

| -Presentation

| -GIT

| -Subversion

| -Trac

` Et les autres



# Subversion



- Implémentation dans Apache 2

## Chargement des modules

```
# ls /etc/apache2/mods-enabled/ |grep dav
dav.load
dav_svn.conf
dav_svn.load
```

```
# ls /etc/apache2/mods-enabled/ |grep authz
authz_default.load
authz_groupfile.load
authz_host.load
authz_user.load
```

## Gestion des utilisateurs

- « authz » intégré à Subversion
- « htaccess » intégré à Apache

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

### 4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

` Et les autres

# Subversion

- Test



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

Revision 3: /

- ◆ [CHANGES](#)
- ◆ [Doc/](#)
- ◆ [Makefile](#)
- ◆ [README](#)
- ◆ [SeaBattle](#)
- ◆ [TODO](#)
- ◆ [affichage.c](#)
- ◆ [affichage.h](#)
- ◆ [affichage.o](#)
- ◆ [jeu.c](#)
- ◆ [jeu.h](#)
- ◆ [jeu.o](#)
- ◆ [main.c](#)
- ◆ [main.o](#)
- ◆ [reseau.c](#)
- ◆ [reseau.h](#)
- ◆ [reseau.o](#)
- ◆ [test.c](#)
- ◆ [test.h](#)
- ◆ [test.o](#)

Powered by [Subversion](#) version 1.4.2 (r22196).

Terminé



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

#### 4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

# Trac



- Système Open-Source de gestion de projet logiciel
- Développé en Python et distribué sous licence BSD
- Trac inclus
  - Un Wiki
  - Gestion de feuilles de route
  - Historique
  - Rapport de bugs
  - Explorateur subversion



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

| Et les autres

# Trac



# trac

Integrated SCM & Project Management

- Le système de ticket



# trac

Integrated SCM & Project Management

[Login](#) | [Settings](#) | [Help/Guide](#) | [About Trac](#)

[Wiki](#) | [Timeline](#) | [Roadmap](#) | [Browse Source](#) | **View Tickets** | [New Ticket](#) | [Search](#)

[Available Reports](#) | [Custom Query](#)

{1} Active Tickets (2 matches)

- List all active tickets by priority.
- Color each row based on priority.
- If a ticket has been accepted, a '\*' is appended after the owner's name

| Ticket | Summary  | Component  | Version | Milestone  | Type   | Owner    | Created    |
|--------|--|------------|---------|------------|--------|----------|------------|
| #1     | création du dépôt subversion et de l'interface track | component1 |         |            | defect | somebody | 11.03.2007 |
| #3     | crypto connection réseau                             | component1 | 2.0     | milestone2 | defect | julien   | 11.03.2007 |

Note: See [TracReports](#) for help on using and creating reports.

Download in other formats:

[XML](#) | [RSS Feed](#) | [Comma-delimited Text](#) | [Tab-delimited Text](#) | [SQL Query](#)



Powered by Trac 0.10.3  
By Edgewall Software.

Visit the Trac open source project at  
<http://trac.edgewall.org/>





1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

# Trac



# trac

Integrated SCM & Project Management

- Le système de ticket

|  |      |          |         |               |                     |            |        |
|--|------|----------|---------|---------------|---------------------|------------|--------|
|  | Wiki | Timeline | Roadmap | Browse Source | <b>View Tickets</b> | New Ticket | Search |
|--|------|----------|---------|---------------|---------------------|------------|--------|

Ticket #3 (new defect)

| crypto connection réseau      |                                | Opened 6 hours ago |                       |
|-------------------------------|--------------------------------|--------------------|-----------------------|
| Reported by:                  | julien <julien@linuxwall.info> | Assigned to:       | julien                |
| Priority:                     | major                          | Milestone:         | milestone2            |
| Component:                    | component1                     | Version:           | 2.0                   |
| Keywords:                     | crypto, reseau                 | Cc:                |                       |
| Description                   |                                |                    | <a href="#">Reply</a> |
| couche TLS sur connection TCP |                                |                    |                       |

## Attachments

- [ssl\\_tls.c](#) (36.9 kB) - added by julien <julien@linuxwall.info> on 11.03.2007 11:56:33.  
code source d'exemple de support TLS

[Attach File](#)



# Trac



# trac

Integrated SCM & Project Management

- La consultation du code

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

| Et les autres

root

View revision:

| Name ▲                      | Size      | Rev | Age     | Last Change  |
|-----------------------------|-----------|-----|---------|--|
| Doc                         |           | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">affichage.c</a> | 11.6 kB   | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">affichage.h</a> | 439 bytes | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">affichage.o</a> | 15.2 kB   | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">CHANGES</a>     | 0.8 kB    | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">jeu.c</a>       | 16.2 kB   | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">jeu.h</a>       | 157 bytes | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">jeu.o</a>       | 14.9 kB   | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">main.c</a>      | 2.3 kB    | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |
| <a href="#">main.o</a>      | 4.2 kB    | 1   | 4 weeks | anonymous: Import des fichiers de la premiere seance |



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

| Et les autres

# Trac



# trac

Integrated SCM & Project Management

- La consultation du code

|  |      |          |         |                      |              |            |        |
|--|------|----------|---------|----------------------|--------------|------------|--------|
|  | Wiki | Timeline | Roadmap | <b>Browse Source</b> | View Tickets | New Ticket | Search |
|--|------|----------|---------|----------------------|--------------|------------|--------|

[Last Change](#) | [Revision Log](#)

root / **affichage.c**

View revision:

**Revision 1, 11.6 kB (checked in by anonymous, 4 weeks ago)**  
 Import des fichiers de la premiere seance  
 Property svn:executable set to \*

| Line |  |
|------|--|
| 1    | <i>/*! \file affichage.c</i>   |
| 2    | <i>\brief Fonctions d'affichage</i>  |
| 3    |  |
| 4    | <i>Ensemble de fonctions permettant de créer et de modifier l'affichage du jeu</i> |
| 5    |  |
| 6    | <i>\Author D.Bernaudeau</i>  |
| 7    | <i>\Author J. Vehent</i>   |
| 8    | <i>*/</i>  |
| 9    |  |
| 10   | <i>#include &lt;ncurses.h&gt;</i>  |
| 11   | <i>#include &lt;string.h&gt;</i>   |
| 12   |  |
| 13   | <i>/*!</i>   |
| 14   | <i>\var grille_adverse</i>   |
| 15   | <i>\brief Ensemble de cases pour la grille adverse</i>                             |
| 16   | <i>*/</i>  |
| 17   | <b>WINDOW*</b> grille_adverse[10][10];   |
| 18   |  |
| 19   | <i>/*!</i>   |
| 20   | <i>\var grille_locale</i>  |
| 21   | <i>\brief Ensemble de cases pour la grille du joueur</i>                           |
| 22   | <i>*/</i>  |
| 23   | <b>WINDOW*</b> grille_locale[10][10];  |
| 24   |  |
| 25   | <i>/*!</i>   |
| 26   | <i>\var bateau</i>   |
| 27   | <i>\brief Fenetre d'affiche centrale pour la liste des bateaux</i>                 |
| 28   | <i>*/</i>  |

# Et les autres...



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

`Et les autres

| Software          | Atomic commits | File renames   | Merge file renames | Symbolic links | Pre/post-event hooks                  | Signed revisions                      | Merge tracking      | End of line conversions | Tags                |
|-------------------|----------------|----------------|--------------------|----------------|---------------------------------------|---------------------------------------|---------------------|-------------------------|---------------------|
| <b>AccuRev</b>    | Yes            | Yes            | ?                  | Yes            | Yes                                   | Yes                                   | Yes                 | ?                       | ?                   |
| <b>Aldon</b>      | Yes            | Yes            | ?                  | Yes            | Yes                                   | Yes                                   | Yes                 | ?                       | ?                   |
| <b>Bazaar</b>     | Yes            | Yes            | Yes                | Yes            | Yes                                   | Partial <sup>[11]</sup>               | Yes                 | No                      | Yes <sup>[12]</sup> |
| <b>ClearCase</b>  | No             | Yes            | ?                  | Yes            | Yes                                   | Yes                                   | Yes                 | Yes                     | Yes                 |
| <b>Code Co-op</b> | Yes            | Yes            | ?                  | No             | limited support                       | No                                    | ?                   | ?                       | ?                   |
| <b>CVS</b>        | No             | No             | No                 | No             | limited support                       | No                                    | No                  | Yes                     | Yes                 |
| <b>CVSNT</b>      | Yes            | Yes            | ?                  | No             | Yes                                   | No                                    | Yes                 | Yes                     | Yes                 |
| <b>darcs</b>      | Yes            | Yes            | ?                  | No             | Yes                                   | Yes                                   | N/A <sup>[13]</sup> | ?                       | Yes                 |
| <b>Git</b>        | Yes            | Yes (implicit) | Yes                | Yes            | Yes <sup>[47]</sup> <a href="#">↗</a> | Yes <sup>[48]</sup> <a href="#">↗</a> | Yes                 | No                      | Yes                 |

source : wikipedia.org

# Et les autres...



| Software           | Atomic commits      | File renames        | Merge file renames | Symbolic links     | Pre/post-event hooks       | Signed revisions | Merge tracking             | End of line conversions | Tags                            |
|--------------------|---------------------|---------------------|--------------------|--------------------|----------------------------|------------------|----------------------------|-------------------------|---------------------------------|
| <b>GNU arch</b>    | Yes                 | Yes                 | ?                  | Yes                | Yes [49] <a href="#">↗</a> | Yes              | ?                          | ?                       | Yes                             |
| <b>Mercurial</b>   | Yes                 | Partial             | ?                  | No                 | Yes (written in Python)    | Yes              | ?                          | Yes                     | Yes                             |
| <b>Monotone</b>    | Yes                 | Yes                 | Yes                | No <sup>[14]</sup> | Yes [50] <a href="#">↗</a> | Yes, mandatory   | Yes                        | No                      | Yes                             |
| <b>Perforce</b>    | Yes                 | Partial             | ?                  | Yes                | Yes                        | Yes              | Yes [51] <a href="#">↗</a> | Yes                     | Yes                             |
| <b>SourceHaven</b> | Yes                 | Yes                 | ?                  | Yes                | Yes                        | No               | ?                          | ?                       | ?                               |
| <b>StarTeam</b>    | Yes <sup>[15]</sup> | Yes                 | ?                  | Yes                | No                         | No               | Yes                        | ?                       | Yes                             |
| <b>Subversion</b>  | Yes                 | Yes <sup>[16]</sup> | No                 | Yes                | Yes                        | No               | No                         | Yes                     | limited support <sup>[17]</sup> |

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

` Et les autres

source : wikipedia.org



# Et les autres...



1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

\ Et les autres

| Software                      | Atomic commits | File renames | Merge file renames | Symbolic links | Pre/post-event hooks | Signed revisions | Merge tracking | End of line conversions | Tags            |
|-------------------------------|----------------|--------------|--------------------|----------------|----------------------|------------------|----------------|-------------------------|-----------------|
| <b>SVK</b>                    | Yes            | Yes          | No                 | Yes            | Yes <sup>[18]</sup>  | ?                | Yes            | Yes                     | limited support |
| <b>Vault</b>                  | Yes            | Yes          | ?                  | ?              | Yes                  | ?                | ?              | ?                       | ?               |
| <b>Visual Source Safe</b>     | No             | Yes          | ?                  | No             | Yes                  | No               | No             | ?                       | ?               |
| <b>Team Foundation Server</b> | Yes            | Yes          | ?                  | ?              | Yes                  | ?                | ?              | ?                       | ?               |
| <b>Plastic SCM</b>            | Yes            | Yes          | Yes                | Yes            | Yes                  | Yes              | Yes            | ?                       | Yes             |

source : wikipedia.org

# La méthode La Rache



Rapid  
rache/  
Application  
Conception and  
Heuristic  
Extreme-program

<http://www.cafenware.org/la-rache/>

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

- | -Presentation
- | -GIT
- | -Subversion
- | -Trac
- \ Et les autres





# La méthode La Rache

1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

| -Presentation

| -GIT

| -Subversion

| -Trac

` Et les autres

1. Résumer le cahier de charges en un post-it de 8 mots maximum

2. Extrapoler ce post-it un sujet de développement vaseux

3. La multiplication du nombre de mot sur le post-it par un chiffre tiré au sort entre 20 et 200 donne la durée du projet en jours/homme. On prendra soin de ne rien planifier dans cette phase.

# La méthode La Rache



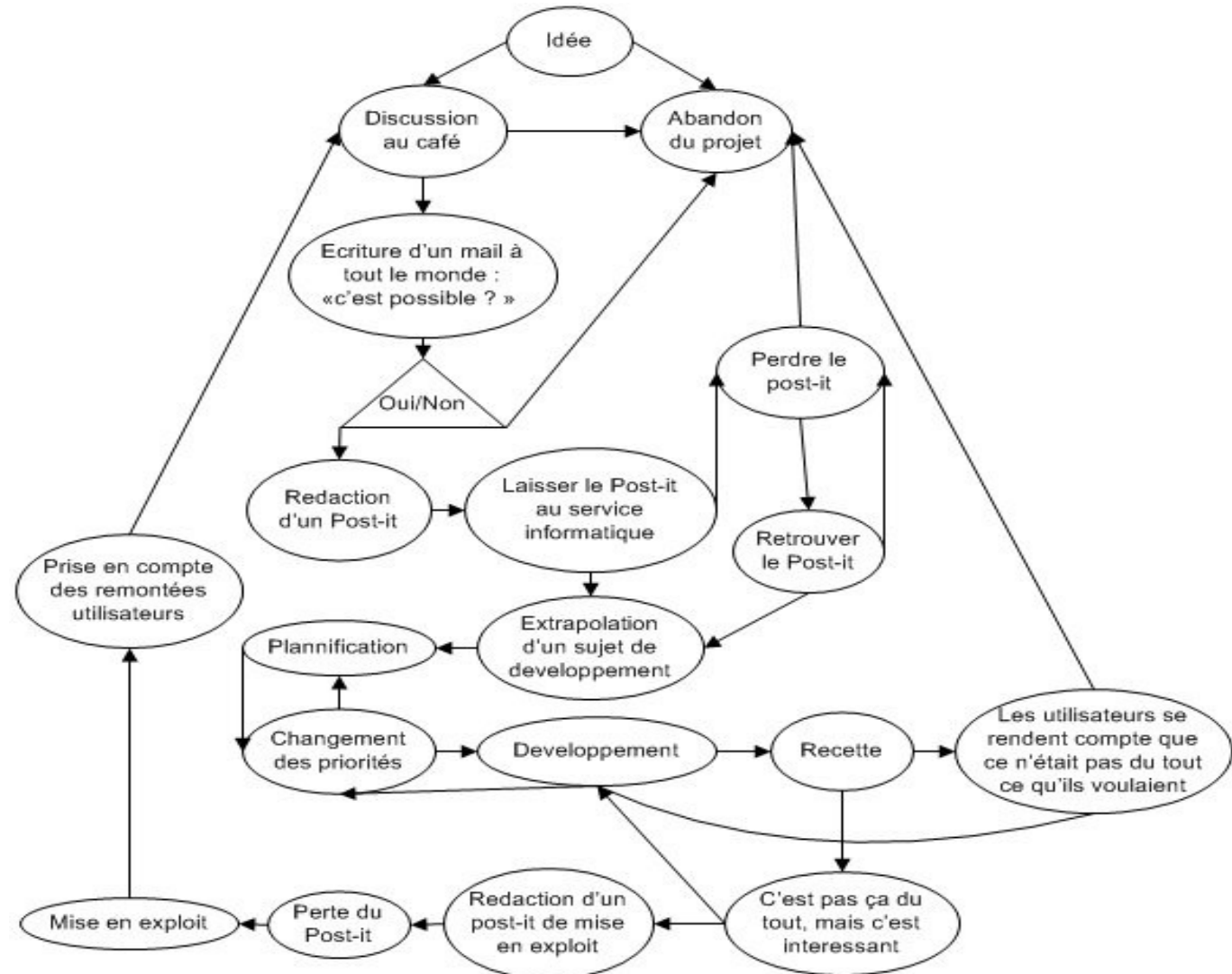
1. l'Université de Poitiers

2. l'ascenseur

3. le Scrabble

4. Versioning

-Presentation  
-GIT  
-Subversion  
-Trac  
Et les autres

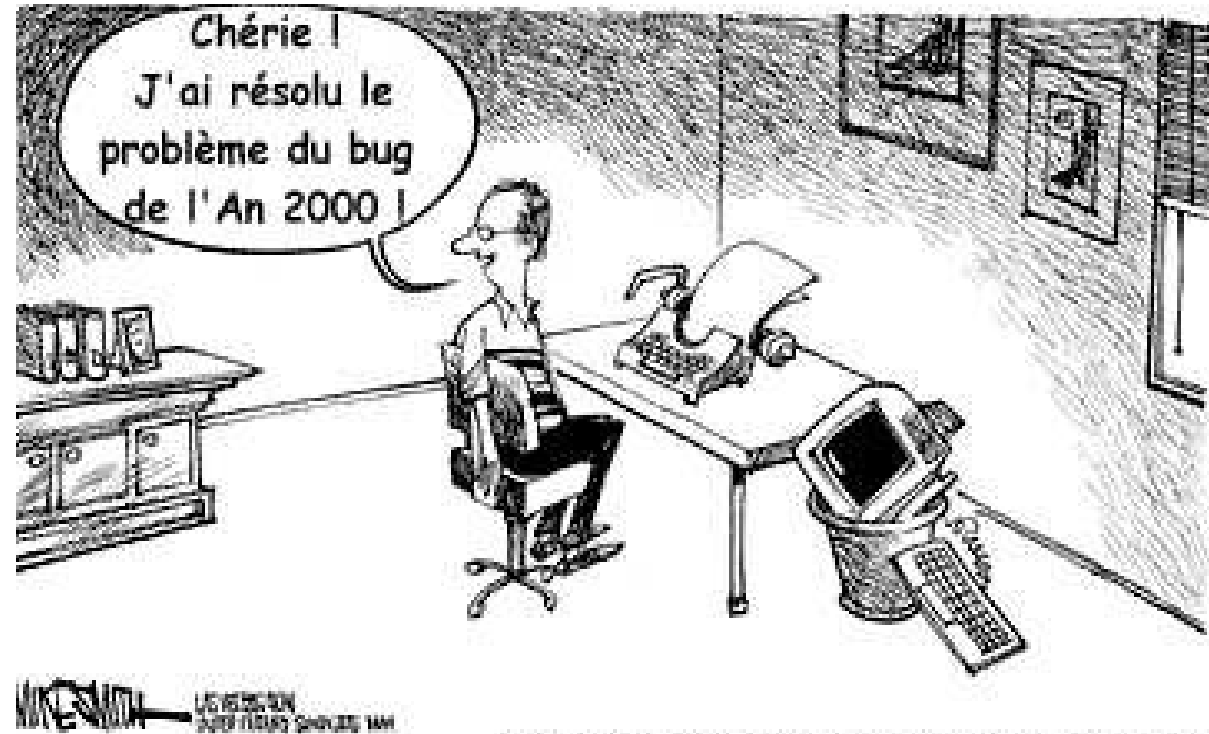






# Merci de votre attention

1. l'Université de Poitiers
2. l'ascenseur
3. le Scrabble
- 4. Versioning**



Traduction DDE (<http://www.jememarre.com>)