

## C.8 Concevoir le cas d'utilisation Démarrer

### Objectif

Concevoir l'initialisation du système.

### Préambule

Les systèmes possèdent (explicitement ou) implicitement un cas d'utilisation que l'on nomme souvent Démarrer qui correspond à l'initialisation de l'application.

#### Important : concevoir l'initialisation en dernier

Lors de l'implémentation, il faudra coder en premier au moins un cas d'utilisation Démarrer (l'opération système d'initialisation est la première à s'exécuter au lancement de l'application). Mais, pendant la modélisation, il faut le considérer en dernier après avoir découvert ce qui doit être créé et initialisé. L'opération système démarrer ou initialiser représente la phase initiale de lancement d'une application. Le lancement d'une application dépend du langage de programmation et du système d'exploitation.

### Méthode

#### Comment les applications démarrent-elles ?

L'opération système démarrer ou initialiser représente la phase initiale de lancement d'une application. Le lancement d'une application dépend du langage de programmation et du système d'exploitation.

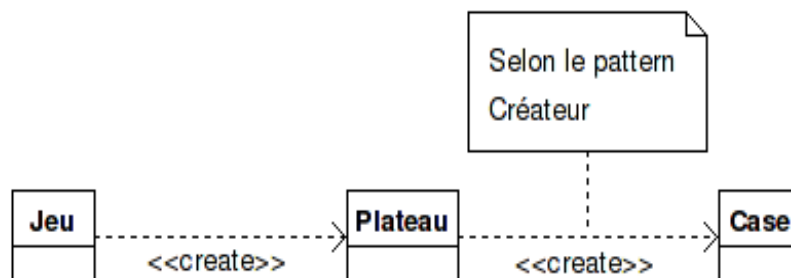
**La règle** : elle consiste à créer un objet du domaine initial qui sera le premier objet logiciel du « domaine » à être créé. Cette création peut avoir lieu dans la méthode main().

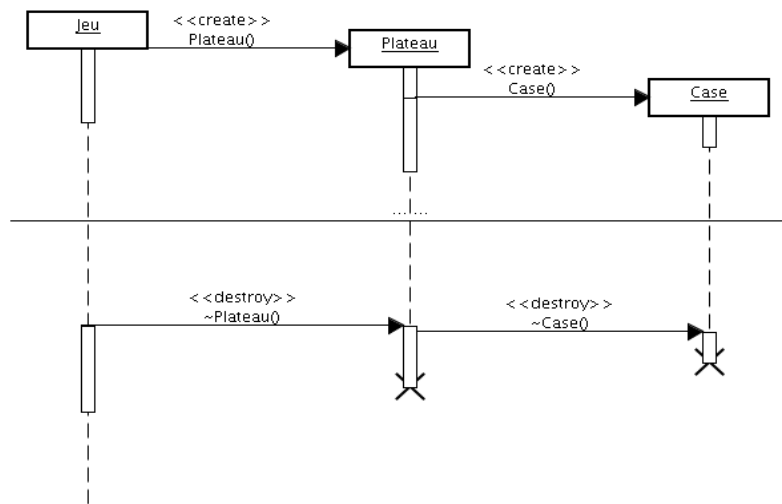
**Le choix** : on choisit un objet racine approprié qui sera le créateur de certains autres objets.

### Exemple

On peut représenter les détails de cette conception dynamique au moyen de diagramme d'interaction (séquence et/ou collaboration).

Une autre approche consiste à utiliser un diagramme (statique) de classes avec des dépendances utilisant le stéréotype « create ». Cette solution donne une bonne vue de l'ensemble des créations.





### Codage

Cela dépend évidemment du langage de programmation et du système d'exploitation.

```

#include "JeuDeLOison.h"

int main( int argc, char* argv[] )
{
    JeuDeLOison jeuDeLOison;

    jeuDeLOison.lancerPartie();

    return 0;
}
  
```

Dans le constructeur de la classe JeuDeLOison :

```

JeuDeLOison::JeuDeLOison() : nombreDeJoueurs(2)
{
    nombreDeTours = 3;
    initialiser();
}
  
```

Et la méthode initialiser() :

```

void JeuDeLOison::initialiser()
{
    plateau = new Plateau;
    plateau->construire();

    joueurs[0] = new Joueur("toto", *plateau);
    joueurs[1] = new Joueur("titi", *plateau);
}
  
```