
Semaine 2

Les notions

1. usage : synopsis et DCU.
2. structure : DC, DO.
3. comportement : DSS et DC.

Programme de la semaine

CM : zoom sur les diagrammes de séquences système et comment s'en servir pour réfléchir au comportement des classes.

Complément sur les fragment d'interaction et les diagrammes d'activité.

TD : DCU et DC

TP : synopsis, DSS et DC

Concepts à connaître sur les diagrammes de séquence • ligne de vie • message synchrone et message de retour (appel de méthode) • message asynchrone • message de création • message de destruction • message de séquence système vs message de séquence détaillé

TD n° 2

Exercice 1 (usage). On s'intéresse à un système permettant de gérer un jardin, composé de parcelles, et de son évolution dans le temps. Ce système doit pouvoir permettre de retrouver l'état du jardin à un instant donné (un peu comme un système de machine à remonter dans le temps qui permet de retrouver l'état antérieur d'un système de fichiers).

Les parcelles. Pour simplifier on supposera que les parcelles sont des rectangles. Chaque jardin est initialement une seule parcelle, c'est-à-dire un rectangle d'une certaine taille en cm (par exemple 100 fois 300).

À tout instant (le temps jardinier n'est pas très précis, une date suffira), un jardinier en chef peut décider de couper une parcelle en 2 (verticalement ou horizontalement) pour en former deux nouvelles. La parcelle initiale existe toujours et contient dorénavant deux sous-parcelles.

À l'inverse, il peut aussi décider de réunir deux sous-parcelles plus tard (à condition qu'elles soient bien issue de la même parcelle), ce qui revient à faire disparaître ses sous-parcelles.

Les jardins pouvant devenir complexes, à l'instar d'une gestionnaire de fichiers, il faudra pouvoir prévoir un système de navigation adapté permettant de ne voir qu'une partie du jardin ou au contraire d'occulter des détails. Cette navigation est accessible à n'importe quel jardinier.

Historique. Le système permet normalement de gérer un découpage évolutif des parcelles.

À un instant donné, un jardinier doit pouvoir indiquer qu'il a fait une action sur une parcelle. Il en existe trois types :

- gérer les légumes (semier, transplanter, récolter, arracher) ;
- travailler le sol (double-béchure, béchure, sarclage, désherbage, amendement de la terre, traitement avec un purin) ; et,
- observer (maladie, ravageur, etc).

Dans la vue, le jardinier doit pouvoir avoir une navigation intuitive dans l'espace-temps pour visualiser facilement les actions qui ont eu lieu en particulier dans les 3 ou 4 dernières années (durée approximative de la rotation de cultures).

Communication. Un membre du public peut visualiser l'état actuel des potagers, avec pour chaque parcelle le détail sur le légume et la date probable de récolte.

Un responsable de la communication peut indiquer diverses informations comme par exemple la prochaine distribution de légumes.

Les utilisateurs peuvent aussi lire et proposer des recettes de cuisine en fonction de légumes choisis comme ingrédients.

On souhaite un DCU de ce système. Pour faciliter la lisibilité, on considérera qu'il s'agit de 3 sous-systèmes indépendants qui peuvent tous interagir avec une base de données. Vous devez donc en fait proposer trois DCU, un pour chaque partie exposée ci-dessus.

Exercice 2 (structure). On s'intéresse à un système permettant de gérer un jardin, composé de parcelles rectangulaires.

Chaque jardin est initialement une seule parcelle, c'est-à-dire un rectangle d'une certaine taille. À tout instant, une parcelle peut être sous-divisée en 2 (verticalement ou horizontalement) pour en former deux nouvelles. La parcelle initiale existe toujours et contient dorénavant deux sous-parcelles. À l'inverse, on peut aussi réunir deux sous-parcelles plus tard (à condition qu'elles soient bien issue de la même parcelle), ce qui revient à faire disparaître ses sous-parcelles.

Les jardins pouvant devenir complexes, à l'instar d'une gestionnaire de fichiers, il faudra pouvoir prévoir un système adapté permettant de naviguer et d'occulter des détails ou de les révéler.

À tout instant, on doit pouvoir indiquer qu'on réalise une action sur une parcelle. Il en existe trois types :

- gérer les légumes (semier, transplanter, récolter, arracher) ;
- travailler le sol (double-béchure, béchure, sarclage, désherbage, amendement de la terre, traitement avec un purin) ; et,

- observer (maladie, ravageur, etc).
1. Proposer un diagramme de classe (notez qu'il s'agit du diagramme de classe du modèle de l'application, on ne souhaite pas avoir de classe correspondant à la vue ou au contrôleur).
 2. Vous pouvez pour vous aider donner un diagramme d'objet illustrant l'exemple de la figure

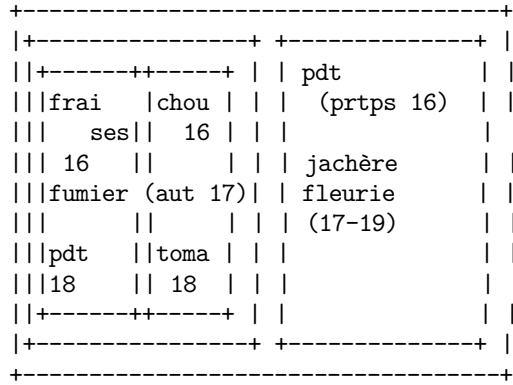


FIGURE 1 – Esquisse d'un jardin (NB. prtps veut dire Printemps, aut veut dire Automne et hvrs veut dire hivers ; pdt dénote un tubercule cher à Parmentier).

TP n° 2

Exercice 3 (comportement). Un jardinier interagit avec un système de jardins, qui est une version proche mais simplifiée de celui entrevu dans des exercices précédents.

Pour simplifier, il y aura donc un seul jardin et les parcelles de ce dernier ne seront pas sauvées si elles sont fermées. Il y aura un seul acteur qui pourra à la fois changer les parcelles et pour chaque parcelle ajouter ou enlever un légumes .

On a les classes suivantes dont on ne connaît pas encore toutes les méthodes (voir Figure 2 pour le diagramme de classe).

- Classe `VueJardin`. Elle contient un attribut de type `parcelle` qui correspond à la parcelle courante que le jardinier regarde et une méthode `SetParcelle` qui prend en argument une parcelle et va mettre à jour cet attribut.
- Classe `Parcelle`. Attributs : un booléen `root` vrai ssi c'est la parcelle principale ; une association vers deux sous-parcelles `sp0` et `sp1` ; une orientation `split` (H ou V) (les sous-parcelles seront gauche et droite ou haut et bas en fonction de l'orientation, l'attribut vaut `Null` si il n'y a pas de sous-parcelle) ; un légume (un type énuméré). La méthode `getFirst` retourne la parcelle gauche si `split` vaut V, ou haute si `split` vaut H. La méthode `reset` va détruire les sous-parcelles.

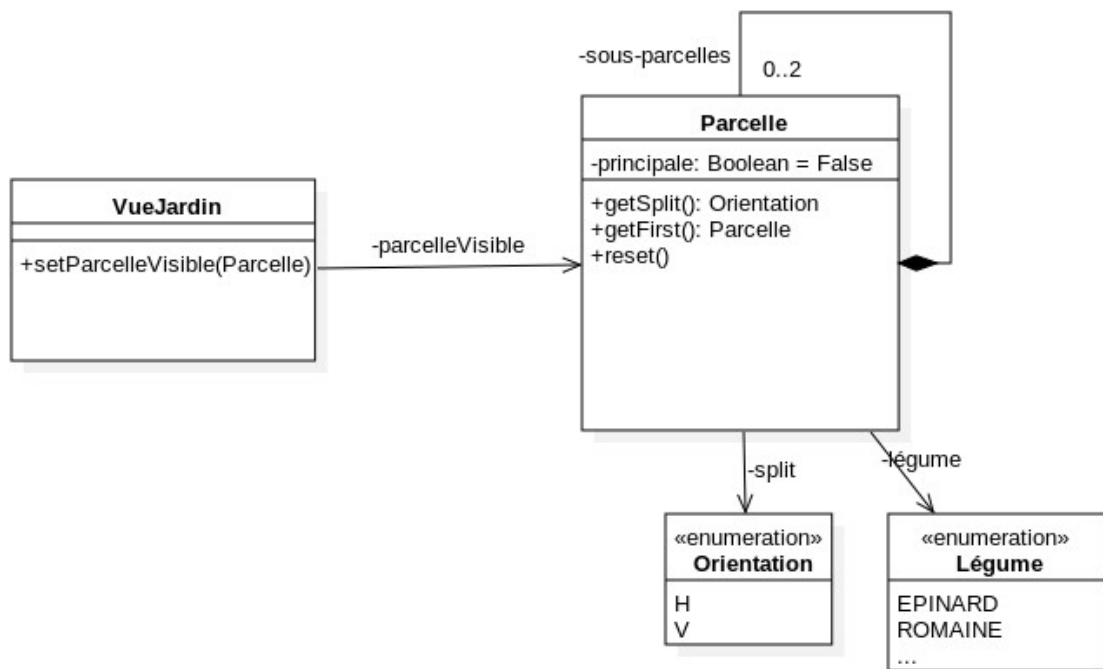


FIGURE 2 – Diagramme de classe partiel

Question 1. On considère le diagramme de séquence système de la Figure 3. Notez que P0 est la parcelle racine du jardin. Donnez un synopsis listant ce que Papé14 a fait sur l'appli. Vous pouvez illustrer le synopsis par un dessin des parcelles.

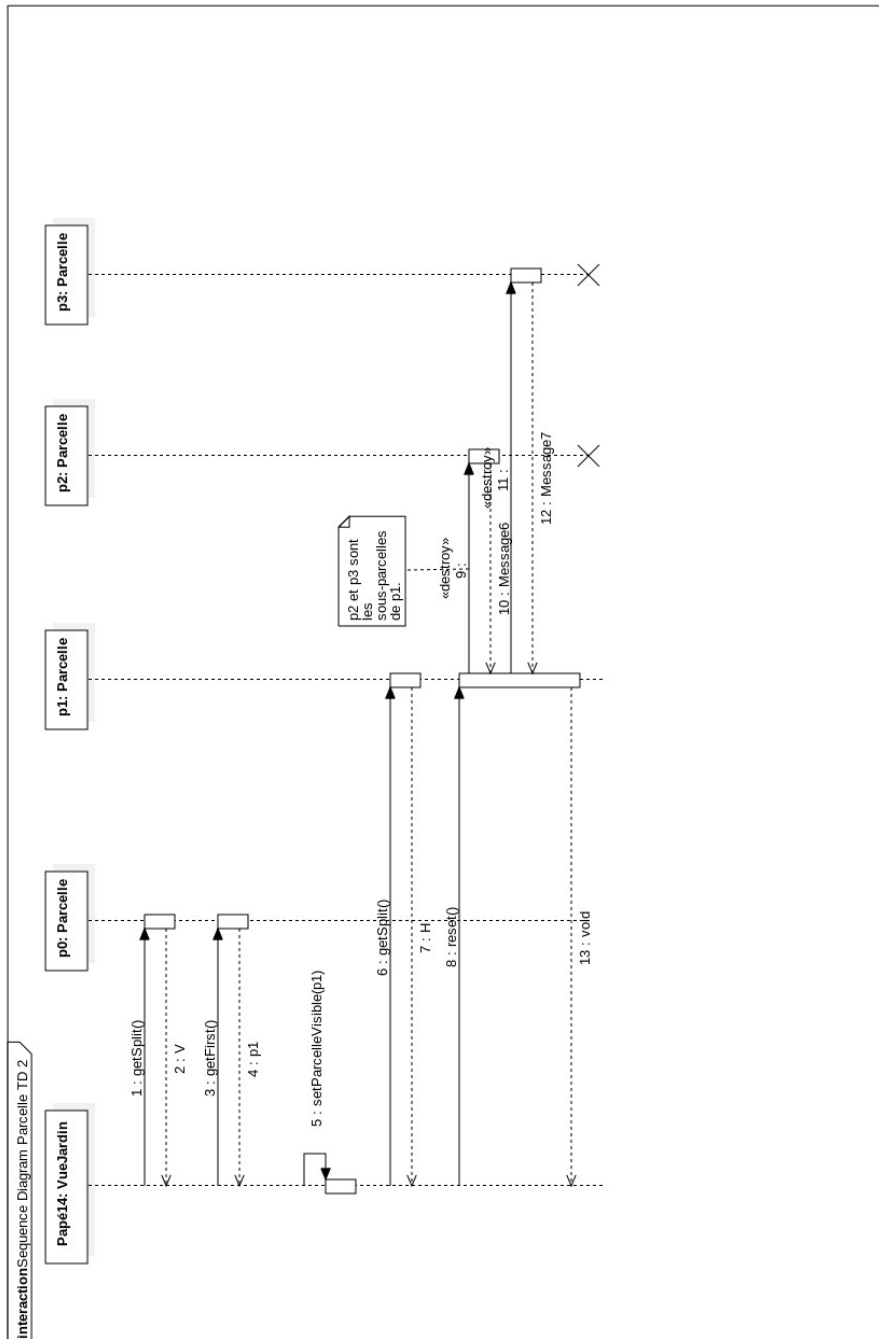


FIGURE 3 – Diagramme de séquence système.

Question 2. Nous disposons du synopsis ci-dessous. Proposez un diagramme de séquence système correspondant à ce synopsis.

Si une méthode est nécessaire dans une classe, indiquez sa signature et donnez une brève description de son comportement si le nom n'était pas assez clair.

Synopsis

- Le jardinier Papé14 visionne la parcelle initiale de son jardin.
- Il voit que cette parcelle est coupée verticalement en deux sous-parcelles.
- Il sélectionne et navigue dans la parcelle droite. Cette parcelle est coupée verticalement en deux.
- Il sélectionne la partie droite et voit qu'elle n'est pas découpée.
- Il la partage en 2 verticalement.
- Il change d'avis et réunit ces deux parcelles en une seule parcelle.
- Il la partage en 2 horizontalement.
- Il navigue successivement sur la parcelle initiale.
- Il sélectionne la parcelle gauche et voit qu'elle contient de la romaine (une salade).
- Il enlève la romaine.
- Il ajoute des épinards.