

Chapitre 3

Les patterns de conception

Durée : 1 heure 30

Mots-clés

patterns de conception, Composite, Visitor, Strategy, Abstract Factory, Singleton

Objectifs

Dans ce chapitre, vous apprendrez à utiliser les patterns de conception (dits aussi patrons de conception, en anglais design patterns) pour la modélisation. Les points abordés sont :

- la modélisation d'une structure à l'aide du pattern Composite ;
- l'usage du pattern Visitor ;
- la structuration d'algorithmes selon le modèle Strategy ;
- la modélisation selon le pattern Abstract Factory ;
- le principe du pattern Singleton.

Prérequis

Pour valider les prérequis nécessaires avant d'aborder le TP, répondez aux questions ci-après :

1. Les patterns de conception sont-ils des modèles de réutilisation ?
 - a. Oui
 - b. Non
2. Quel pattern de conception est adapté à la représentation de structures arborescentes ?
3. Quel est l'intérêt du pattern **Composite** ?
4. Quelles affirmations sont vraies en ce qui concerne le pattern **Composite** ?
 - a. Lors de l'ajout de nouveaux types de composants, il n'est pas nécessaire de modifier le code client.
 - b. Le client a besoin de distinguer dans son code les objets individuels des objets composites.
 - c. Les composants enfants d'un objet **Composite** sont stockés dans celui-ci.
5. Quelle est l'utilité d'un pattern **Visitor** ?
6. Le modèle **Visitor** définit deux hiérarchies de classes. Lesquelles ?

7. Est-il facile d'ajouter un visiteur qui effectue une nouvelle opération ?
 - a. Oui
 - b. Non
8. Dans quel cas est-il préférable de définir les opérations dans les classes plutôt qu'à travers un modèle **Visitor** ?
9. Sur quel concept de l'approche par objets se base le pattern **Strategy** ?
10. Quelles affirmations sont vraies en ce qui concerne le pattern **Abstract Factory** ?
 - a. Le modèle **Abstract Factory** est utile lorsque le système traité est constitué de familles de produits destinés à être utilisés ensemble.
 - b. Il est facile de substituer les familles de produits.
 - c. Il est facile d'ajouter de nouveaux types de produits.
11. Grâce à quel traitement, le modèle **Singleton** est-il mis en œuvre ?
 - a. En encapsulant le constructeur de la classe dans une sous-classe.
 - b. En changeant l'accessibilité du constructeur.
 - c. En testant l'existence d'une instance dans une méthode spéciale.
12. Quelle est la particularité des attributs et des méthodes de classe ?

Corrigé p. 113

Énoncé 3.1 Document XML

Durée estimative : 40 minutes

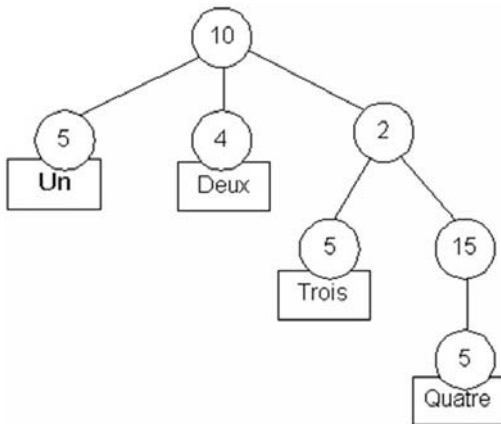
Un document XML est structuré en éléments à l'aide de balises qui marquent le début et la fin de chaque élément. Les éléments peuvent contenir du texte et éventuellement d'autres éléments. La syntaxe d'un élément est la suivante :

```
<nom attribut="valeur">contenu</nom>
```

Un élément peut contenir plusieurs attributs. Afin de simplifier la modélisation, supposons que le seul attribut est une valeur de type entier.

Ci-dessous un exemple de document XML et sa représentation arborescente.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes" ?>
<ELEMENT valeur="10">
  <ELEMENT valeur="5">Un</ELEMENT>
  <ELEMENT valeur="4">Deux</ELEMENT>
  <ELEMENT valeur="2">
    <ELEMENT valeur="5">Trois</ELEMENT>
    <ELEMENT valeur="15">
      <ELEMENT valeur="5">Quatre</ELEMENT>
    </ELEMENT>
  </ELEMENT>
</ELEMENT>
```



1. Modélisez un document XML par un diagramme de classes en utilisant le pattern **Composite**.
2. Utilisez le pattern **Visitor** pour calculer la valeur totale du document et le nombre d'éléments composés.
3. Quelle est la démarche à suivre pour ajouter une opération permettant d'imprimer le contenu des éléments simples ?

Corrigé p. 114

Énoncé 3.2 Site web multilingue

Durée estimative : 15 minutes

Un site de vente par correspondance souhaite étendre son activité à plusieurs pays. Proposez une solution flexible permettant de personnaliser l'affichage du site web selon le pays. Les messages et les prix affichés doivent être automatiquement adaptés. La solution proposée doit faciliter l'ajout d'un nouveau pays sans changer l'interface du client.

1. Quel pattern est adapté pour ce type de problèmes ?
2. Les pays visés sont la France, l'Espagne et les États-Unis. Proposez une représentation des différents algorithmes nécessaires à la mise en place de ce système.

Corrigé p. 118

Énoncé 3.3 Interface GUI

Durée estimative : 35 minutes

GUI est l'abréviation de *Graphical User Interface* (interface utilisateur graphique). Des exemples de systèmes supportant une interface GUI sont MacOS, Microsoft Windows et XWindow. Le but de cet exercice est de proposer une modélisation d'un système flexible, permettant d'afficher les différents composants d'une interface graphique {boutons, barres de défilement, menus, etc.) en fonction du système, le format de ces composants graphique étant différent d'un système à un autre.

1. Quel pattern de conception est adapté à ce type de problèmes ?
2. Proposez une modélisation sous forme de diagramme de classes.
3. Une seule instance d'un créateur de composants doit être disponible par système dans le modèle. Quel est le pattern à utiliser pour réaliser cette restriction ? Explicitez le code correspondant.

Corrigé p. 119

Chapitre 4

La composition d'objets et de patterns

Durée : 2 heures 20

Mots-clés

diagramme de structure composite, collaboration, application de collaboration, patterns de conception, paramètre de template, clé

Objectifs

Dans ce chapitre, vous apprendrez à :

- maîtriser la composition avancée d'objets, notamment pour représenter, dans certains cas, la spécialisation ;
- mettre en œuvre le diagramme de structure composite, notamment dans le cas de la collaboration relative aux patterns de conception ;
- réaliser une composition de patterns ;
- utiliser les paramètres de template.

Prérequis

Pour valider les prérequis nécessaires avant d'aborder le TP, répondez aux questions ci-après :

1. Qu'est-ce que la clé d'une classe ?
 - a. Une référence des instances de cette classe vers les instances d'une autre classe.
 - b. Un ou plusieurs attributs dont la ou les valeurs sont uniques pour chaque instance de la classe.
 - c. Un identifiant unique de la classe.
2. Que peut-on décrire dans un diagramme de structure composite ?
 - a. Les activités des parties d'un objet.
 - b. Les collaborations liées à un pattern de conception.
 - c. Les parties d'un objet et leurs connecteurs.
 - d. Les parties d'un cas d'utilisation.
3. Quel est l'intérêt principal d'un diagramme de structure composite par rapport à un diagramme de classes ?
4. Que signifie la cardinalité d'une partie ?

5. Qu'est-ce qu'un rôle associé à une partie ?
6. Quelle est la différence entre un connecteur et une association ?
 - a. Il n'y a aucune différence.
 - b. Un connecteur est l'extrémité d'une association.
 - c. Un connecteur relie deux parties tandis qu'une association relie deux classes.
7. Qu'est-ce qu'une collaboration dans un diagramme de structure composite ?
8. Une application d'une collaboration décrit l'utilisation d'un pattern dans un cas précis, en précisant le rôle de chaque classe intervenant dans le pattern ?
 - a. Oui.
 - b. Non.
9. Qu'est-ce qu'un paramètre de template d'une classe ?
10. Qu'est-ce que signifie « contraindre un paramètre de template » ?
 - a. Indiquer que le paramètre doit être substitué par une classe concrète.
 - b. Indiquer que le paramètre doit être substitué par une classe spécifiée dans la contrainte ou l'une de ses surclasses.
 - c. Indiquer que le paramètre doit être substitué par une classe spécifiée dans la contrainte ou l'une de ses sous-classes.

Corrigé p. 123

Énoncé 4.1 Mesure

Durée estimative : 30 minutes

En physique, une mesure s'exprime à l'aide d'une valeur numérique, d'un delta (plus ou moins) autour de cette valeur et d'une unité. Par exemple, la mesure d'une longueur vaut $10,3 \text{ m} \pm 0,1$.

Il existe des mesures basées sur une unité simple, telles que le mètre du précédent exemple. Il existe également des mesures dont l'unité est composée par plusieurs unités simples, comme la mesure de la vitesse (par exemple, en mètres par seconde: m/s ou m.s^{-1}) ou la mesure de l'énergie électrique facturée (en kilowatts-heure : kW.h). Une puissance liée à chaque unité simple apparaît également dans la composition (-1 pour les secondes dans le cas de la vitesse exprimée en m.s^{-1} , 2 pour les surfaces exprimées en m^2).

Le but est de décrire une mesure physique par un diagramme de classes en utilisant la composition. La valeur de la mesure et son delta sont décrits au sein d'une même classe en tant que composant de la classe représentant la mesure. Chaque unité de base (mètre, kilogramme, heure, etc.) est représentée par une instance unique de la classe des unités.

1. Tracez ce diagramme des classes sans y représenter une notion d'ordre dans les unités de la mesure, c'est-à-dire, par exemple, sans y introduire de distinction entre $m.s^{-1}$ et $s^{-1}.m$. Concernant les puissances associées aux unités, nous nous limitons à l'utilisation directe de l'unité (ou puissance 1) et à l'inverse (puissance -1). Utilisez deux associations d'agré- gation vers les objets représentant les unités.
2. Supprimez la limitation sur les puissances en utilisant une classe-association.
3. Tracez ce diagramme des classes en y introduisant une notion d'ordre dans les unités de la mesure, c'est-à-dire, par exemple, en distinguant la représentation de $m.s^{-1}$ et $s^{-1}.m$.

Corrigé p. 124

Énoncé 4.2 Arbre B

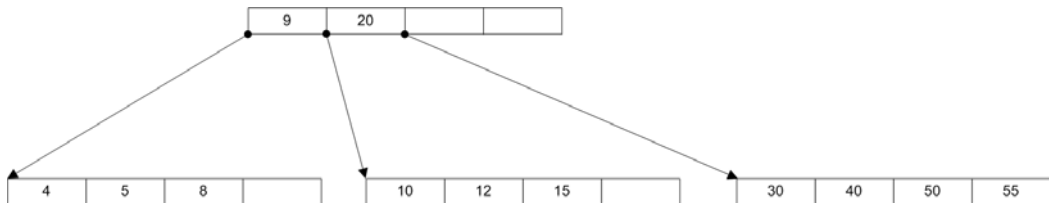
Durée estimative : 20 minutes

Un arbre B sert à indexer un ensemble d'objets de données et à pouvoir retrouver rapidement chaque objet de données à l'aide d'une clé associée de façon unique à cet objet. Il existe une relation d'ordre entre les clés : il est possible, lorsque l'on connaît deux clés, de déterminer la plus petite des deux.

Un arbre B est une extension d'un arbre binaire dont chaque nœud contient des clés et des fils. Un arbre B possède un ordre. Le nombre maximal de clés et donc d'objets de données référen- cés par un nœud est égal à 2^n , n étant le degré de l'arbre B.

Soit un nœud est une feuille (c'est-à-dire qu'il ne possède aucun fils), soit il possède un fils pour chaque clé ainsi qu'un dernier fils. Un fils correspondant à une clé C possède des clés supé- rieures à la clé précédant la clé C (si elle existe) et inférieures à la clé C. Le dernier fils contient des clés supérieures à la plus grande clé du nœud. Le nombre maximum de fils est donc $2^n + 1$.

La figure suivante illustre un exemple d'arbre B de degré 2.



La racine possède deux clés et trois fils. Les feuilles possèdent trois ou quatre clés. Les objets de données associés aux clés ne sont pas représentés dans cette figure.

Tracez le diagramme des classes qui décrit un arbre B.

Indices

Le plus simple est d'associer chaque fils (hormis le dernier fils) à chaque clé.

Il n'est pas nécessaire de décrire les contraintes de cardinalité sur le nombre de clés et de fils d'un nœud.

N'oubliez pas que les clés sont ordonnées au sein de chaque nœud.

Corrigé p. 126

Énoncé 4.3 Table

Durée estimative : 10 minutes

Il s'agit de décrire une table composée d'un plateau, de quatre pieds et deux tiroirs. Les pieds et le plateau ne peuvent pas être dissociés, à la différence des tiroirs.

1. Décrivez la table à l'aide d'un diagramme de classes.
2. Transformez le précédent diagramme en un diagramme de structure composite.

Corrigé p. 126

Énoncé 4.4 Dictaphone

Durée estimative : 15 minutes

Un dictaphone à cassettes est composé de deux têtes (une tête de lecture et une tête d'enregistrement), de trois touches (lecture, enregistrement et arrêt). Il peut être livré avec un casque.

1. Décrivez le dictaphone à l'aide d'un diagramme de structure composite sans distinguer les différents types de têtes et de touches. Tracez les connecteurs en spécifiant les cardinalités aux extrémités.
2. Nous souhaitons maintenant distinguer les différents types de touches et de têtes. Adaptez le diagramme en conséquence en attribuant les rôles aux parties.

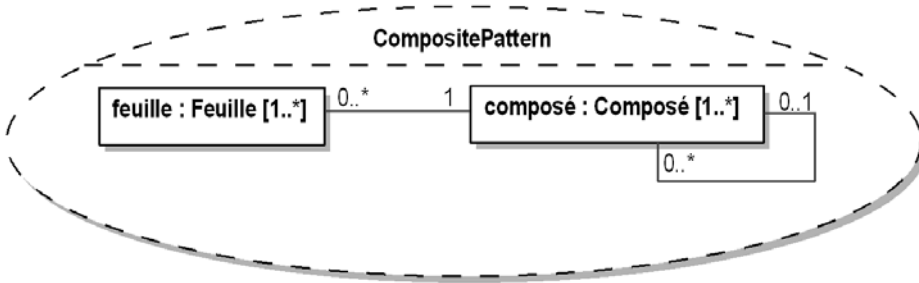
Corrigé p. 128

Énoncé 4.5 Document XML

Durée estimative : 15 minutes

Le pattern **Composite** a été introduit dans le chapitre sur les patterns de conception. Nous reprenons l'exercice du document XML.

Le pattern **Composite** est décrit à l'aide d'un diagramme de structure composite avec collaboration de la manière suivante :



Tracez l'application de cette collaboration pour un document XML.

Corrigé p. 129

Énoncé 4.6 Statut d'une personne

Durée estimative : 20 minutes

Nous nous intéressons à des personnes ayant soit le statut d'étudiant, soit celui d'employé, soit celui de chômeur, soit celui de retraité. Ces personnes changent de statut au cours de leur vie. Pour décrire ces différents statuts, le recours à la spécialisation d'une classe **Personne** tel que l'illustre la figure suivante semble la solution la plus appropriée. La classe **Personne** est introduite en tant que classe abstraite qui décrit les aspects communs de la personne, quel que soit son statut. Les quatre sous-classes introduisent les attributs spécifiques à chaque statut.