

LOG8430 : Examen final

Zéphyrin Soh
Département de génie informatique et génie logiciel
École Polytechnique de Montréal, Québec, Canada
zephyrin.soh@poymtl.ca

Instructions

- La durée de l'examen est 2 heures et 30 minutes.
- La documentation est interdite.
- La présentation et l'orthographe sont importantes.
- Le barème est indicatif.
- Lisez tout le sujet avant de commencer.
- Répondez brièvement et précisément !
- Répondez dans les cadres prévus à cet effet.
- Justifiez et commentez **toujours** vos réponses.
- Calculatrices, ordinateurs, etc. sont interdits.

Questions

1 Connaissances

1.1 – 3 points. En quoi consiste la définition/description d'une architecture logicielle ?

- Décrire l'organisation générale d'un système et sa décomposition en sous-systèmes ou composants
- Déterminer les interfaces entre les sous-systèmes
- Décrire les interactions et le flot de contrôle entre les sous-systèmes
- Décrire également les composants utilisés pour implanter les fonctionnalités des sous-systèmes
 - Les propriétés de ces composants
 - Leur contenu (e.g., classes, autres composants)
 - Les machines ou dispositifs matériels sur lesquels ces modules seront déployés

1.2 – 4 points. Citer et définir quatre utilités d'une architecture logicielle

- Compréhension : facilite la compréhension des grands systèmes complexes en donnant une vue de haut-niveau de leur structure et de leurs contraintes. Les motivations des choix de conception sont ainsi mis en évidence
- Réutilisation : favorise l'identification des éléments réutilisables, parties de conception, composants, caractéristiques, fonctions ou données communes

- Construction : fournit un plan de haut-niveau du développement et de l'intégration des modules en mettant en évidence les composants, les interactions et les dépendances
- Évolution : met en évidence les points où un système peut être modifié et étendu. La séparation composant/connecteur facilite une implémentation du type plug-and-play
- Analyse : offre une base pour l'analyse plus approfondie de la conception du logiciel, analyse de la cohérence, test de conformité, analyse des dépendances
- Gestion : contribue à la gestion générale du projet en permettant aux différentes personnes impliquées de voir comment les différents morceaux du casse-tête seront agencés. L'identification des dépendances entre composants permet d'identifier où les délais peuvent survenir et leur impact sur la planification générale

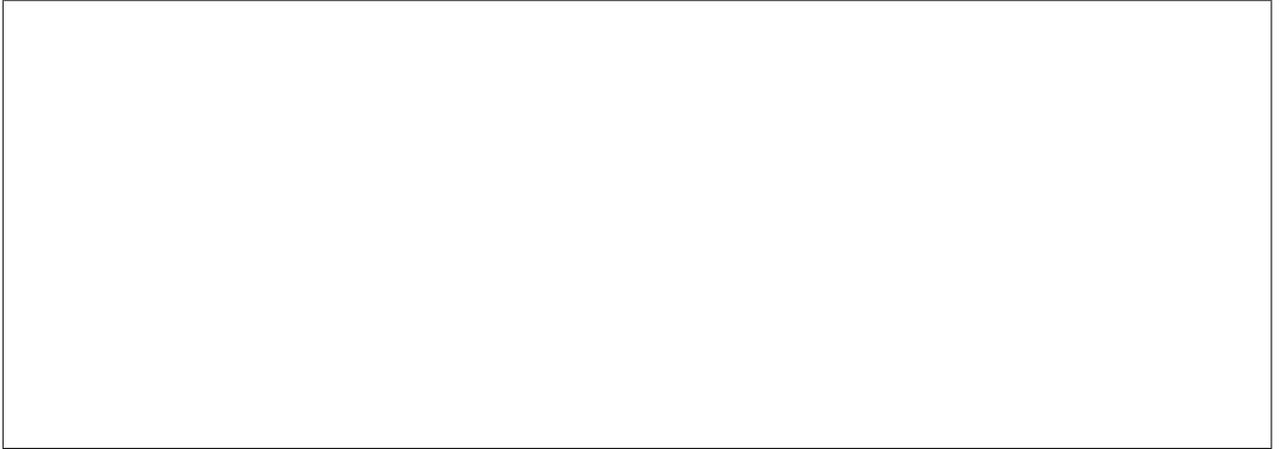
1.3 – 6 points. Quels sont les cinq types d'interconnexions possibles entre une bibliothèque/un cadriciel et le reste d'un programme ?

- Chaînage des liens (*linking*)
- Duplication de processus (*fork*)
- Sous-classage
- Protocole de communication (*IPC*)
- Chargement dynamique

1.4 – 6 points. Présentez le diagramme des relations entre classe et méta-classe en Java ?

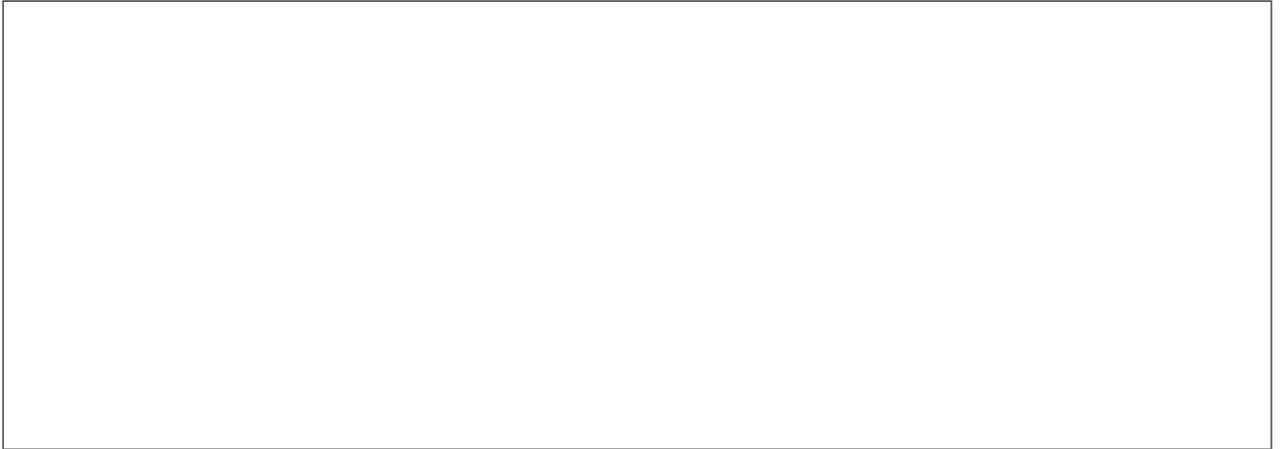
- Voir transparents, important : directions inversées de l'instantiation et du sous-classage

1.5 – 6 points. Présentez le diagramme du modèle de composants de Fractal ?



– Voir transparents

1.6 – 5 points. Qu'est-ce que le *BootStrap Component* de Fractal ?



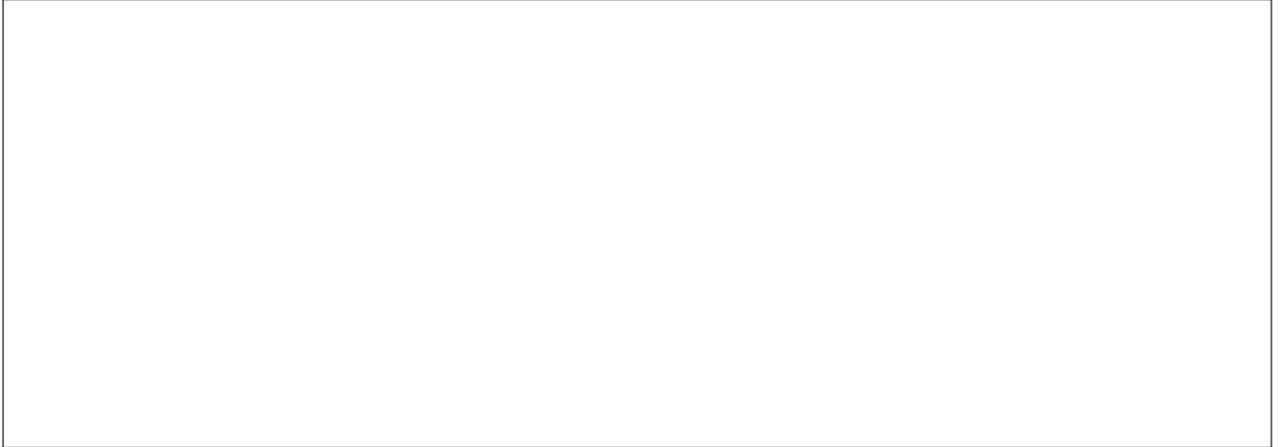
– Le premier composant par défaut qui peut instancier d'autres composants

1.7 – 2 points. En programmation par aspects, que caractérisent les notions de Point d'action et Point de jonction ? Quelle est la différence majeure entre ces deux notions ?



– Un Point d'action est un ensemble de points de jonctions.

1.8 – 6 points. Qu'est-ce qu'un défaut de conception ? Énumérez les étapes de détection et de correction des défauts de conception.



Mauvaise solution au problème de conception

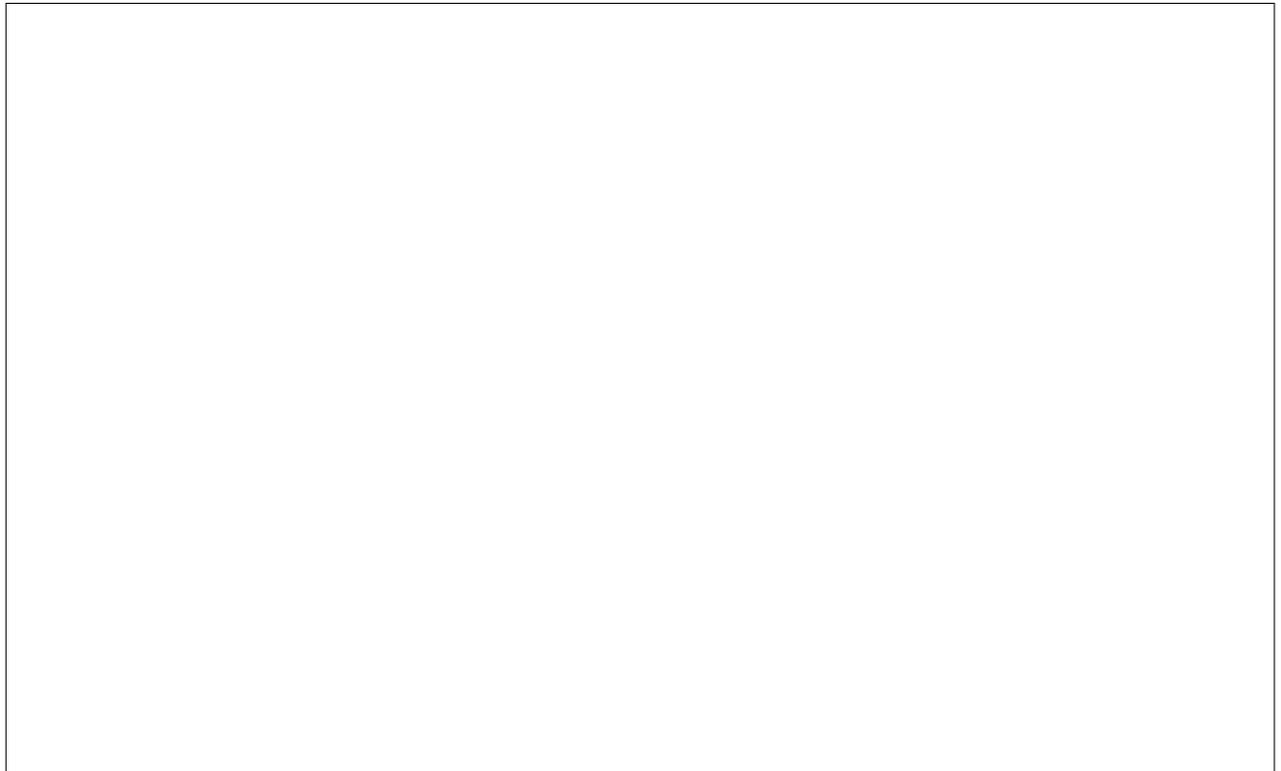
- spécifier les règles pour le défaut
- Analyser le code
- Calculer les métriques
- Discrétiser les métriques en niveaux
- application des règles
- Refactorisation

2 Interprétation

```
1    class Foo {
2        private static Helper helper = null;
3        public static Helper getHelper() {
4            if (helper == null) {
5                helper = new Helper();
6            }
7            return helper;
8        }
9        // Other functions and members...
10   }
```

Listing 1 – Du code.

2.1 – 6 points. Que décrit le code du listing 1? Expliquez le problème de ce code en présence de plus d'un fil d'exécution et proposez une solution à ce problème.



Patron Singleton

Pose un problème d'instanciation multiple (car affectation non-atmique)

Synchronisation

```
1 import java.util.*;
2
3 public class InstrumentedHashSet<E> extends HashSet<E> {
4     // The number of attempted element insertions
5     private int addCount = 0;
6     ...
7     @Override public boolean add(final E e) {
8         this.addCount++;
9         return super.add(e);
10    }
11    @Override public boolean addAll(final Collection<? extends E> c) {
12        this.addCount += c.size();
13        return super.addAll(c);
14    }
15    public int getAddCount() {
16        return this.addCount;
17    }
18    ...
```

Listing 2 – Du code.

2.2 – 6 points. Que représente le code du listing 2? Décrivez/expliquez le problème de ce code et proposez une solution à ce problème.



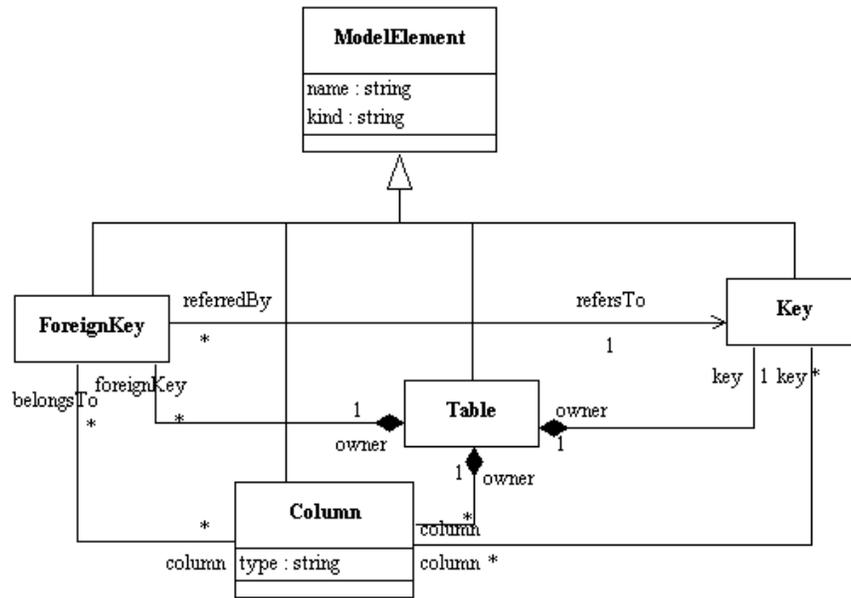


FIGURE 1 – Première figure.

2.3 – 10 points. Qu'est-ce qu'un méta-modèle ? Que permet de décrire le méta-modèle sur la figure 1 ?

- Un langage.
- La modélisation d'une base de données relationnelle.

3 Résolution de problèmes

Le département informatique de la grande entreprise de produits d'entretien WashMe, qui compte environ 10 000 employés de par le monde et est basée à Churchill, Manitoba, a reçu la commande d'un système de gestion de la paie.

L'objectif du projet est de suppléer/remplacer le programme actuellement utilisé par un programme entièrement accessible depuis l'intranet de l'entreprise et l'Internet.

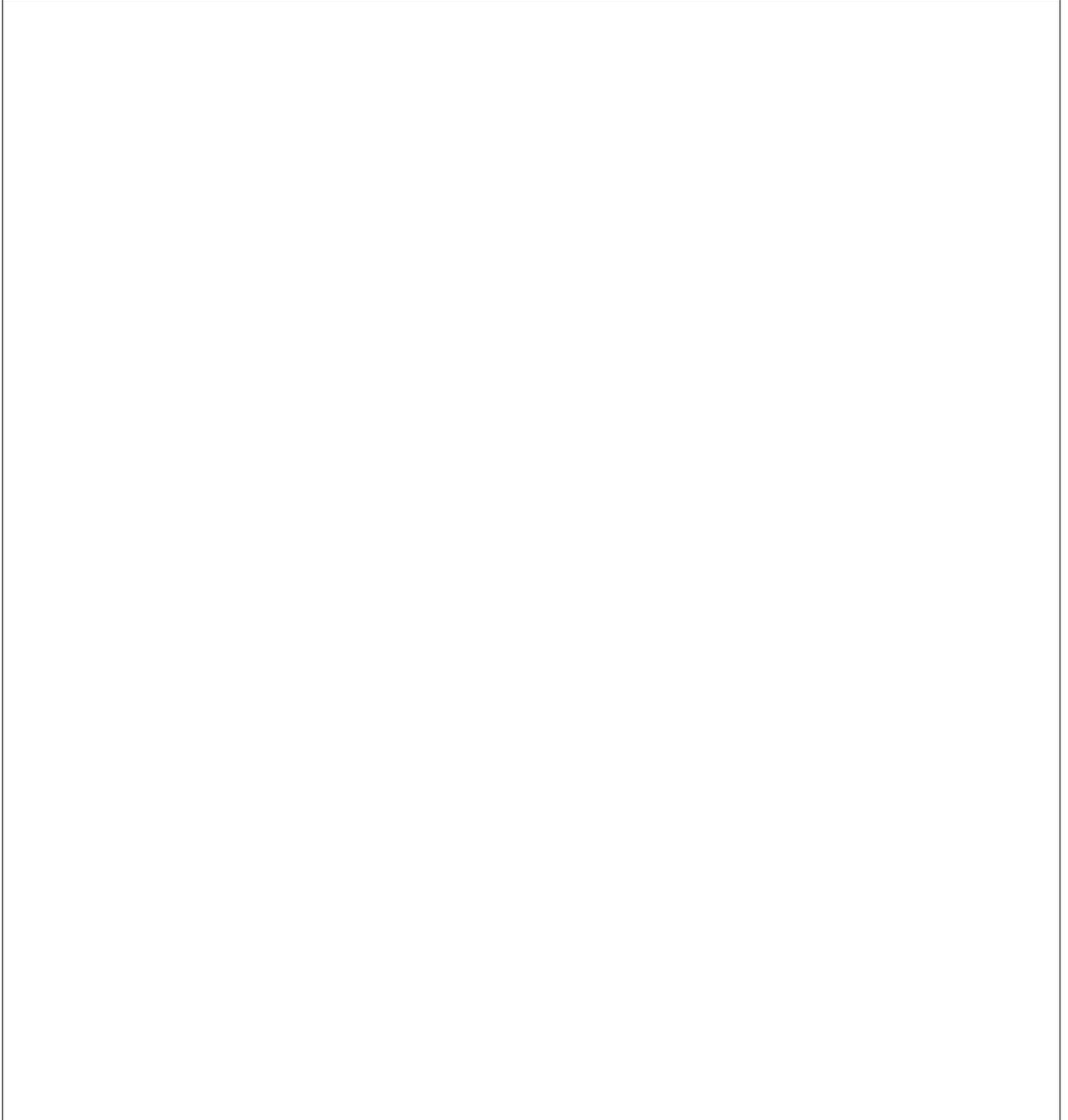
En particulier, chaque employé pourra se connecter au programme avec son numéro d'employé et un mot de passe personnel pour voir son statut, son historique de paie. De plus, chaque employé pourra modifier ses contributions à des œuvres caritatives et aux activités (payantes) de divertissement de son usine.

Les employés autorisés, par exemple ceux appartenant aux services financiers ou aux ressources humaines, pourront également payer les employés. Aussi, les employés autorisés pourront créer et mettre à jour la liste des œuvres caritatives (code, nom, objectifs, type de donation) et la liste des activités de divertissement offertes par les usines (code, nom, location, calendrier).

Le chef du service informatique de l'entreprise WashMe est optimiste même si ses départements avaient jusqu'à présent eu à s'occuper principalement du réseau informatique et du parc des ordinateurs des départements financiers, commerciaux.

En effet, le département informatique à Hongkong vient de recruter plusieurs spécialistes en UML, le département informatique à Reykjavík plusieurs programmeurs Java et le département informatique à Heidelberg plusieurs testeurs.

1. *40 points.* Décrivez les étapes que vous suivriez pour mettre en œuvre le nouveau système sans brusquement remplacer un système par un autre. Les étapes peuvent être organisationnelles, logistiques et/ou informatiques et possiblement itératives, répétitives. En particulier, pensez aux styles architecturaux et/ou patrons de conception et/ou techniques de programmation qui pourraient vous aider dans votre tâche.



1. Créer une *façade* enveloppant les parties de l'ancien système qui vont être remplacée au fur et à mesure.

2. Possiblement utiliser la programmation par aspect pour ajouter à l'ancien système le code nécessaire pour que son interface de programmation soit conforme au nouveau système.
3. Possiblement créer des *ponts* entre le nouveau et l'ancien système.
4. Possiblement rendre composants l'ancien système.