

Projet de POCA — Échecs Féeriques

Université Paris Diderot – Master 2 SRI/LC/LP

13 décembre 2011

1 Principe du projet — rappel

Le projet de POCA vise à vous faire progresser dans votre capacité à *concevoir un système objet extensible*.

Il se déroule en *deux étapes*. Dans la première étape, un sujet vous est fourni. Ce sujet est *volontairement* décrit de façon informelle, sans vous donner d'indication pour le réaliser. Vous devez donc définir vous-même une *architecture* pour votre projet et vous la décrirez dans le **délivrable 1**.

Une *implémentation* écrite en Scala de cette première version constitue le **délivrable 2**.

Vous recevrez ensuite un *autre sujet* qui est une extension du premier. Cette extension sera objet du **délivrable 3** (plus de détails seront fournis avec le deuxième sujet) et mettra à l'épreuve votre architecture initiale : un projet bien pensé *n'aura pas besoin de modifier* le code de la première version du projet pour traiter cette extension mais seulement de *rajouter de nouveaux composants*.

2 Sujet

Le jeu d'échecs a connu beaucoup de variantes : le livre *Encyclopedia of Chess Variants* en contient environ 2000. Le terme *échecs féeriques* (*Fairy chess* en anglais) est souvent utilisé pour parler, d'une façon générale, des variantes du jeu.

- une première généralisation consiste à permettre des **configurations initiales différentes** de l'échiquier pour :
 1. permettre de placer les pièces dans des positions autres que les deux dernières lignes de chaque côté ;
 2. permettre un nombre arbitraire de pièces (p.ex. joueur avec 2 reines et 3 fous dès le début) ;
 3. permettre des configurations différentes pour les blancs et les noirs (p.ex. donner un avantage au noir en lui donnant 2 reines).
- une deuxième généralisation consiste à changer la **dimension de l'échiquier**, pour permettre de jouer sur des échiquier plus grands (p.ex. 12x12), plus petits (p.ex. 6x6), ou encore rectangulaires (p.ex. 8x10)

- une autre généralisation consiste à supporter des **pièces féeriques** comme la princesse (ou centaure), l'impératrice (ou chancelier), la sauterelle, et le noctambule (une liste exhaustive d'exemples est disponible sur Wikipedia à la page http://en.wikipedia.org/wiki/Fairy_chess_piece). Combien d'entre eux arrivez-vous à supporter ?
- une autre généralisation consiste en l'utilisation des échiquiers où certains bords de l'échiquier communiquent. On parle des **échecs cylindriques** horizontaux (les rangées 1 et 8 communiquent), verticaux (les colonnes a et h communiquent), et toriques (la combinaison de deux précédents). Pour plus d'info sur les échecs cylindriques, voir la page Wikipedia http://en.wikipedia.org/wiki/Cylinder_chess
- une autre généralisation, popularisée par la série télé *Star Trek* sont les **échecs cubiques**, où le jeu se déroule sur plusieurs échiquiers 8x8, idéalement placés les uns au dessus des autres en vertical et les pièces peuvent se déplacer en horizontal comme en vertical. Voir http://en.wikipedia.org/wiki/Three-dimensional_chess pour plus d'info, en particulier sur la variante classique Raumschach.
- une dernière généralisation consiste à changer le **nombre de joueurs** et jouer, par exemple, à 3 personnes avec un échiquier circulaire comme décrit à <http://www.3manchess.com/>

On vous propose donc plusieurs extensions, vous devrez réfléchir à la réalisation de toutes ces extensions, et en réaliser le plus possible sous forme de code Scala.

3 Travail demandé

L'utilisation du SVN est obligatoire. L'historique de ce dernier permettra de déterminer la contribution des membres de l'équipe et la gestion du temps dont vous avez fait preuve.

Avant le 21 novembre 2011 Dans un répertoire `delivrables/architecture/` de votre SVN, vous devez nous soumettre une architecture sous la forme d'un ou plusieurs diagrammes de classes UML et d'autres diagrammes de votre choix accompagnés d'explications justifiant vos choix.

Ce document sera fourni au format PDF et pourra suivre le plan suivant :

Interprétation du sujet Vous expliquerez de façon informelle ce que vous avez compris du sujet et de ces enjeux. Quels sont les problèmes techniques et conceptuels que vous avez exhibés ?

Concepts Dans cette section, vous définirez les concepts utilisés pour modéliser le problème ainsi que les invariants essentiels du système.

Description de l'architecture Vous donnerez ici les diagrammes UML décrivant votre architecture et surtout sa justification.

Extensions envisagées Vous énumérerez ici les généralisations et les extensions que vous avez imaginées et vous expliquerez pourquoi votre architecture permet de les traiter facilement.

Avant le 12 décembre 2011 Dans un répertoire `delivrables/version1/` de votre SVN, vous devez soumettre une première version de votre système.

Celui-ci doit évidemment compiler à l'aide d'une commande `make` effectuée à la racine de ce répertoire. Un fichier `README` doit également être fourni et doit expliquer comment exécuter votre système et, le cas échéant, comment l'utiliser avec des logiciels externes (p.ex. `xboard`).

Votre projet doit aussi être testé. La qualité du code—c'est-à-dire sa correction, sa robustesse et son élégance—sera prise en compte dans la notation.

Avant le 9 janvier 2012 Dans le répertoire `delivrables/version2/` du SVN, vous déposerez votre implémentation de la version 2 du projet. Celui-ci doit évidemment compiler à l'aide d'une commande `make` effectuée à la racine de ce répertoire. Votre projet doit aussi être testé. La qualité du code—c'est-à-dire sa correction, sa robustesse et son élégance—sera pris en compte dans la notation. Enfin, l'ensemble des différences entre la version 1 et la version 2 devront être résumées dans un fichier nommé `CHANGES` que l'on trouvera à la racine de la version 2 du projet.