

Progettazione object-oriented di un'interfaccia grafica JavaFX per il simulatore Alchemist

Tesi in Programmazione ad Oggetti

Niccolò Maltoni
Matricola: 0000719734

Realtore: Prof. Mirko Viroli
Correlatore: Prof. Danilo Pianini

Alma Mater Studiorum - Università di Bologna
Campus di Cesena

12 dicembre 2017

- 1 Introduzione
 - Obiettivo
- 2 Analisi
 - L'interfaccia classica di Alchemist
 - Requisiti & Casi d'uso
- 3 Interfaccia grafica
 - Stato dell'arte e fonti d'ispirazione
 - Mockup
- 4 Design e implementazione
 - Caratteristiche e miglioramenti
 - Coverage
 - Prestazioni
- 5 Conclusioni

Introduzione

Obiettivo

Lo scopo di questa tesi è la progettazione e la successiva implementazione di un'interfaccia grafica 2D più moderna e usabile per il simulatore *Alchemist* avvalendosi del framework JavaFX.

Alchemist¹ è un meta-simulatore estendibile completamente open-source che esegue su JVM, nato all'interno dell'Università di Bologna.

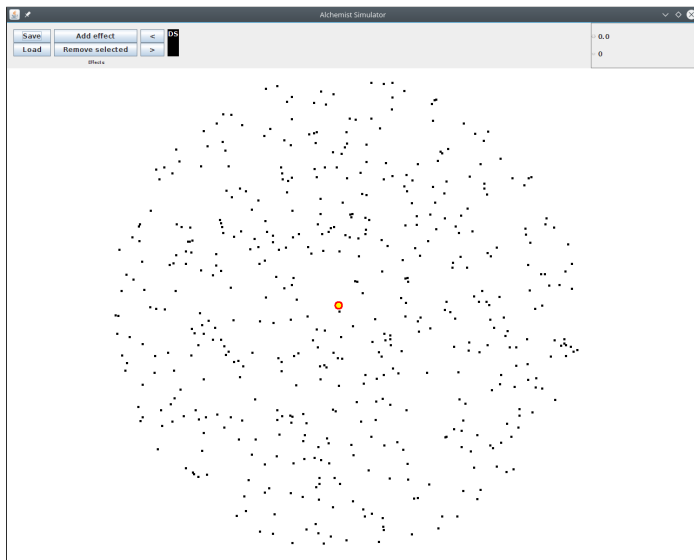
¹<http://alchemistsimulator.github.io>

Outline

- 1 Introduzione
 - Obiettivo
- 2 Analisi
 - L'interfaccia classica di Alchemist
 - Requisiti & Casi d'uso
- 3 Interfaccia grafica
 - Stato dell'arte e fonti d'ispirazione
 - Mockup
- 4 Design e implementazione
 - Caratteristiche e miglioramenti
 - Coverage
 - Prestazioni
- 5 Conclusioni

Analisi

L'interfaccia classica di Alchemist



Analisi

L'interfaccia classica di Alchemist

L'interfaccia utente classica di Alchemist ha diverse criticità:

Analisi

L'interfaccia classica di Alchemist

L'interfaccia utente classica di Alchemist ha diverse criticità:

- è implementato con il framework Swing, soluzione ormai considerata *legacy* e sostituita da JavaFX

Analisi

L'interfaccia classica di Alchemist

L'interfaccia utente classica di Alchemist ha diverse criticità:

- è implementato con il framework Swing, soluzione ormai considerata *legacy* e sostituita da JavaFX
- il sistema di controllo non è intuitivo

Analisi

L'interfaccia classica di Alchemist

L'interfaccia utente classica di Alchemist ha diverse criticità:

- è implementato con il framework Swing, soluzione ormai considerata *legacy* e sostituita da JavaFX
- il sistema di controllo non è intuitivo
- l'aspetto estetico è datato e non aderisce ad alcun design grafico in particolare

Analisi

L'interfaccia classica di Alchemist

L'interfaccia utente classica di Alchemist ha diverse criticità:

- è implementato con il framework Swing, soluzione ormai considerata *legacy* e sostituita da JavaFX
- il sistema di controllo non è intuitivo
- l'aspetto estetico è datato e non aderisce ad alcun design grafico in particolare
- le capacità di rappresentazione, rappresentate dagli effetti, sono legate strettamente ai nodi e limitano la libertà di rappresentazione

Analisi

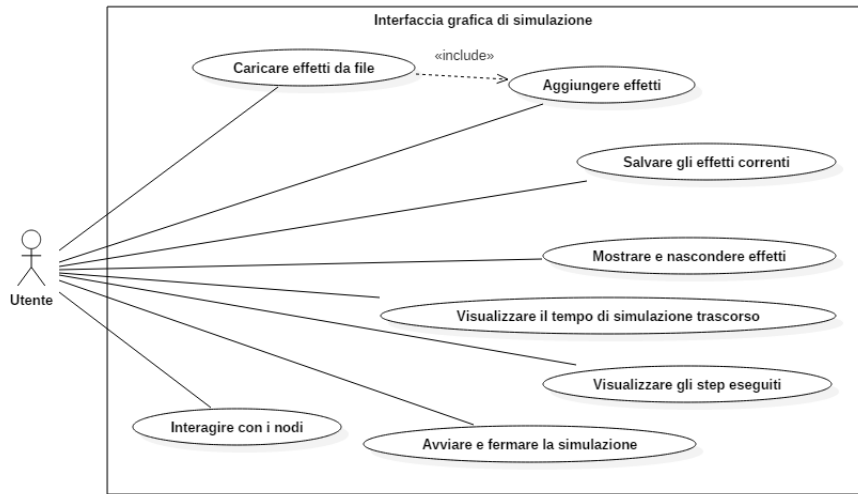
L'interfaccia classica di Alchemist

L'interfaccia utente classica di Alchemist ha diverse criticità:

- è implementato con il framework Swing, soluzione ormai considerata *legacy* e sostituita da JavaFX
- il sistema di controllo non è intuitivo
- l'aspetto estetico è datato e non aderisce ad alcun design grafico in particolare
- le capacità di rappresentazione, rappresentate dagli effetti, sono legate strettamente ai nodi e limitano la libertà di rappresentazione
- non ha il supporto HiDPI

Analisi

Requisiti & Casi d'uso



- 1 Introduzione
 - Obiettivo
- 2 Analisi
 - L'interfaccia classica di Alchemist
 - Requisiti & Casi d'uso
- 3 **Interfaccia grafica**
 - **Stato dell'arte e fonti d'ispirazione**
 - **Mockup**
- 4 Design e implementazione
 - Caratteristiche e miglioramenti
 - Coverage
 - Prestazioni
- 5 Conclusioni

Interfaccia grafica

Stato dell'arte e fonti d'ispirazione

- L'obiettivo era semplificare quanto più possibile l'interazione:

Interfaccia grafica

Stato dell'arte e fonti d'ispirazione

- L'obiettivo era semplificare quanto più possibile l'interazione:
 - ▶ simulatori a scopo scientifico dello stato dell'arte non hanno l'esperienza utente al centro del design

Interfaccia grafica

Stato dell'arte e fonti d'ispirazione

- L'obiettivo era semplificare quanto più possibile l'interazione:
 - ▶ simulatori a scopo scientifico dello stato dell'arte non hanno l'esperienza utente al centro del design
 - ▶ ci si è ispirati a simulatori videoludici di successo

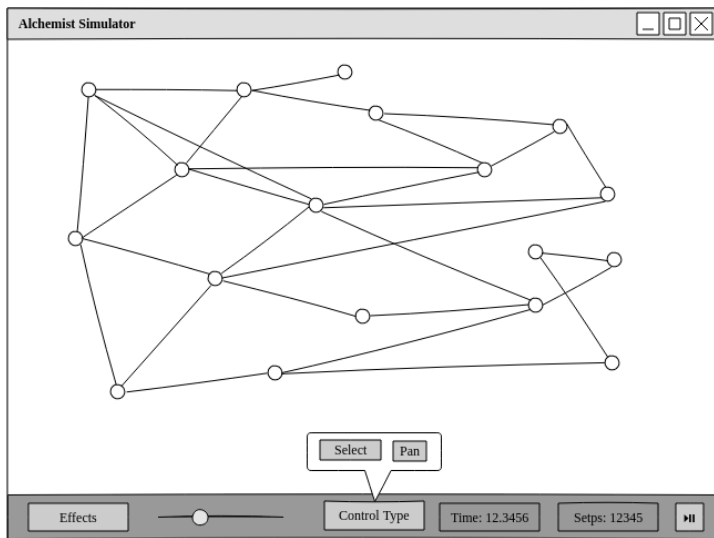
Interfaccia grafica

Stato dell'arte e fonti d'ispirazione

- L'obiettivo era semplificare quanto più possibile l'interazione:
 - ▶ simulatori a scopo scientifico dello stato dell'arte non hanno l'esperienza utente al centro del design
 - ▶ ci si è ispirati a simulatori videoludici di successo
- Lo stile grafico è basato sul Material Design di Google

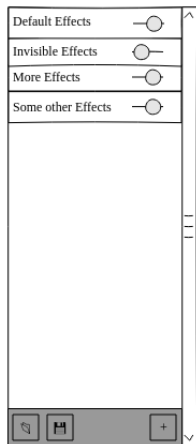
Interfaccia grafica

Mockup



Interfaccia grafica

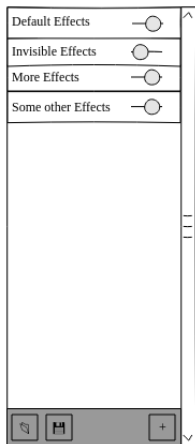
Mockup



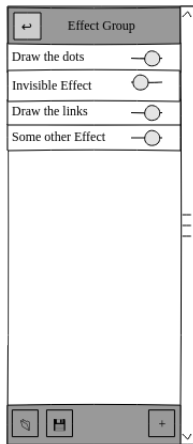
Gruppi di effetti
dello *stack*

Interfaccia grafica

Mockup



Gruppi di effetti
dello *stack*

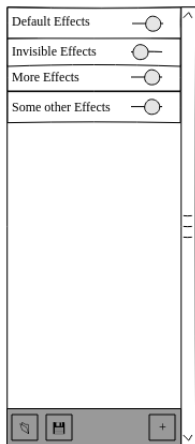


Effetti di un
gruppo

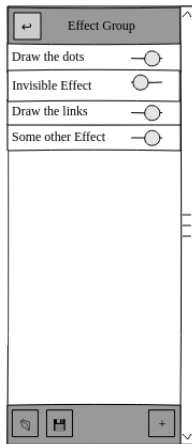


Interfaccia grafica

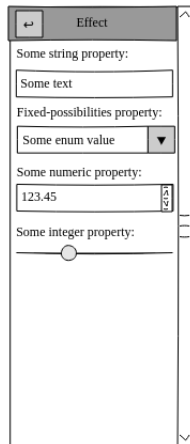
Mockup



Gruppi di effetti
dello *stack*



Effetti di un
gruppo

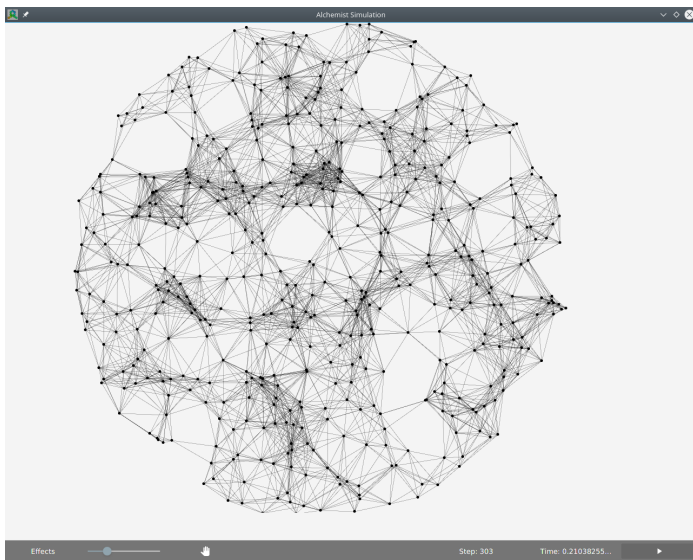


Proprietà di un
effetto

- 1 Introduzione
 - Obiettivo
- 2 Analisi
 - L'interfaccia classica di Alchemist
 - Requisiti & Casi d'uso
- 3 Interfaccia grafica
 - Stato dell'arte e fonti d'ispirazione
 - Mockup
- 4 Design e implementazione
 - Caratteristiche e miglioramenti
 - Coverage
 - Prestazioni
- 5 Conclusioni

Design e implementazione

Caratteristiche e miglioramenti



Design e implementazione

Caratteristiche e miglioramenti

- La libreria utilizzata è JavaFX

JavaFX

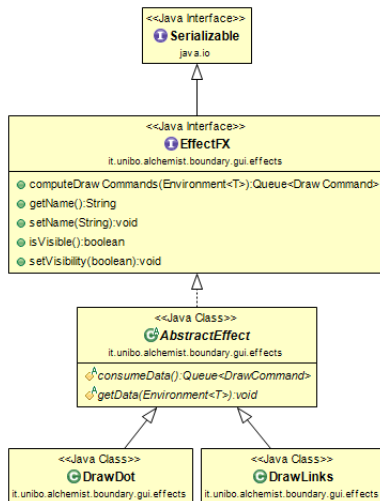
JavaFX è un framework che permette di costruire interfacce grafiche a partire da file FXML di layout e codice Java di controllo; rispetto a Swing:

- si presta di più al pattern MVC
- scala meglio su differenti densità di pixel
- le performance sono generalmente migliori

Design e implementazione

Caratteristiche e miglioramenti

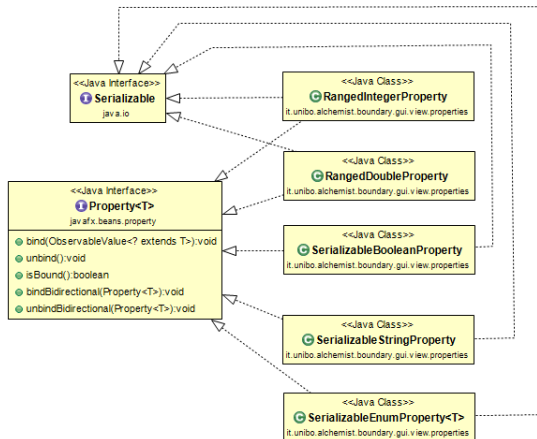
- La libreria utilizzata è JavaFX
- è stata riprogettata l'architettura degli effetti



Design e implementazione

Caratteristiche e miglioramenti

- La libreria utilizzata è JavaFX
- è stata riprogettata l'architettura degli effetti
- è stato introdotto l'uso di proprietà specifiche per gli effetti



Design e implementazione

Caratteristiche e miglioramenti

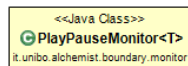
- La libreria utilizzata è JavaFX
- è stata riprogettata l'architettura degli effetti
- è stato introdotto l'uso di proprietà specifiche per gli effetti
- sono stati implementati diversi monitor per la rappresentazione grafica:

Design e implementazione

Caratteristiche e miglioramenti

- La libreria utilizzata è JavaFX
- è stata riprogettata l'architettura degli effetti
- è stato introdotto l'uso di proprietà specifiche per gli effetti
- sono stati implementati diversi monitor per la rappresentazione grafica:

- ▶ dello stato



Design e implementazione

Caratteristiche e miglioramenti

- La libreria utilizzata è JavaFX
- è stata riprogettata l'architettura degli effetti
- è stato introdotto l'uso di proprietà specifiche per gli effetti
- sono stati implementati diversi monitor per la rappresentazione grafica:
 - ▶ dello stato
 - ▶ dell'avanzamento

Step: 303

Time: 0.21038255...

<<Java Class>>

FXStepMonitor<T>

it.unibo.alchemist.boundary.monitor

<<Java Class>>

FXTimeMonitor<T>

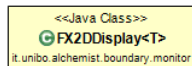
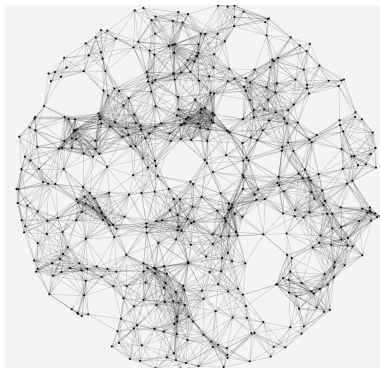
it.unibo.alchemist.boundary.monitor

Design e implementazione

Caratteristiche e miglioramenti

- La libreria utilizzata è JavaFX
- è stata riprogettata l'architettura degli effetti
- è stato introdotto l'uso di proprietà specifiche per gli effetti
- sono stati implementati diversi monitor per la rappresentazione grafica:

- ▶ dello stato
- ▶ dell'avanzamento
- ▶ degli effetti



Design e implementazione

Coverage

Overall Coverage Summary

Package	Class, %	Method, %	Line, %
all classes	96,2% (25/ 26)	66% (163/ 247)	60% (566/ 943)

Coverage Breakdown

Package ▲	Class, %	Method, %	Line, %
it.unibo.alchemist.boundary.gui.effects	83,3% (5/ 6)	57,7% (64/ 111)	47,1% (197/ 418)
it.unibo.alchemist.boundary.gui.effects.json	100% (8/ 8)	90,6% (29/ 32)	92,5% (98/ 106)
it.unibo.alchemist.boundary.gui.view.properties	100% (12/ 12)	67,3% (70/ 104)	64,7% (271/ 419)

Con render

Nuova GUI	
Nodi + Link:	1,3153ms
Solo Nodi:	0,3041ms

Vecchia GUI	
Nodi + Link:	1,1059ms
Solo Nodi:	0,6841ms

Senza render

Nuova GUI	
Nodi + Link:	1,2648ms
Solo Nodi:	0,2855ms

Vecchia GUI	
Nodi + Link:	0,6584ms
Solo Nodi:	0,6696ms

Outline

- 1 Introduzione
 - Obiettivo
- 2 Analisi
 - L'interfaccia classica di Alchemist
 - Requisiti & Casi d'uso
- 3 Interfaccia grafica
 - Stato dell'arte e fonti d'ispirazione
 - Mockup
- 4 Design e implementazione
 - Caratteristiche e miglioramenti
 - Coverage
 - Prestazioni
- 5 Conclusioni

L'obiettivo della tesi è stato per la maggior parte raggiunto: le funzionalità implementate adempiono al loro compito e l'impatto sulle performance è accettabile.

Per portare sul canale stabile il lavoro svolto è ancora necessario implementare un monitor che sia in grado di rappresentare ambienti di simulazione con mappe di sfondo e aumentare le possibilità di interazione con l'ambiente.