

# Railway Simulation

Moreno Ambrosin  
mat. 1035635

Progetto Sistemi Concorrenti e Distribuiti

Corso di laurea magistrale in Informatica  
Università degli Studi di Padova  
Padova

27 marzo 2013

# Indice

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Il problema</b>  | <b>2</b> |
| 1.1      | Specifica arricchita . . . . .                                | 3        |
| <b>2</b> | <b>Analisi Concorrenza e Distribuzione</b>                    | <b>4</b> |
| 2.1      | Ingresso in un Segmento da parte di un Treno . . . . .        | 4        |
| 2.2      | Uscita da un Segmento e accesso alla Stazione . . . . .       | 5        |
| 2.3      | Acquisto di un Biglietto da parte di un Viaggiatore . . . . . | 6        |
| <b>3</b> | <b>Soluzione</b>  | <b>7</b> |
| 3.1      | Architettura Logica di Distribuzione . . . . .                | 7        |
| 3.1.1    | Regioni . . . . .   | 7        |
| 3.1.2    | Biglietterie . . . . .  | 8        |
| 3.1.3    | Controller Centrale . . . . .                                 | 8        |
| 3.2      | Architettura Logica Concorrente . . . . .                     | 9        |
| 3.2.1    | Segmento . . . . .  | 9        |
| 3.2.2    | Viaggiatore . . . . .   | 9        |
| 3.2.3    | Treno . . . . .   | 10       |
| 3.2.4    | Stazione . . . . .  | 10       |
| 3.3      | Interazione tra le Entità . . . . .                           | 11       |
| 3.3.1    | Viaggio di un Viaggiatore . . . . .                           | 11       |
| 3.3.2    | Viaggio di un Treno . . . . .                                 | 12       |

# Capitolo 1

## Il problema

Il progetto didattico prevede la progettazione e la realizzazione di un simulatore software di un sistema ferroviario. Tale sistema prevede i seguenti requisiti di base (la specifica originale è consultabile all'indirizzo <http://www.math.unipd.it/~tullio/SCD/2005/Progetto.html>):

- La presenza di Treni appartenenti a categorie a priorità e modalità di fruizione diverse.
- La presenza di Viaggiatori, che eseguono operazioni elementari come acquisto di un biglietto, salita/discesa su/da un Treno, ecc.
- La definizione di un ambiente costituito da un insieme di Stazioni, ciascuna composta da:
  - Piattaforme di attesa per Treni e Viaggiatori.
  - Un Pannello Informativo che visualizza informazioni sui Treni in arrivo, in transito e che hanno appena superato la Stazione corrente.
  - Una Biglietteria, presso la quale un Viaggiatore può acquistare un biglietto di viaggio.
- La presenza di Segmenti di collegamento tra Stazioni, a percorrenza bidirezionale.
- La presenza di un entità di controllo globale che mantiene lo stato di ciascun Viaggiatore e ciascun Treno in transito.
- L'obbligo da parte di un Viaggiatore di acquistare un Biglietto prima di poter usufruire del servizio ferroviario.

- La presenza di collegamenti multipli tra Stazioni, ovvero ciascuna Stazione può essere raggiunta da più Segmenti e da ciascuna stazione possono partire più segmenti.
- Il percorso portato a termine da un Viaggiatore può comprendere cambi di treno.

## 1.1 Specifica arricchita

La soluzione presentata con il seguente documento, è relativa alla seguente raffinamento della specifica originaria:

- Ciascun Treno appartiene ad una delle seguenti due categorie:

### **Regionale**

Treno a bassa priorità, senza posto garantito.

### **FB**

Treno a priorità più alta, che necessita prenotazione.

- Ciascun Treno possiede una capienza massima di Viaggiatori.

## Capitolo 2

# Analisi Concorrenza e Distribuzione

La modellazione di un sistema ferroviario presenta diverse problematiche relative alla concorrenza e alla distribuzione. Di seguito verranno presentate le più rilevanti.

### 2.1 Ingresso in un Segmento da parte di un Treno

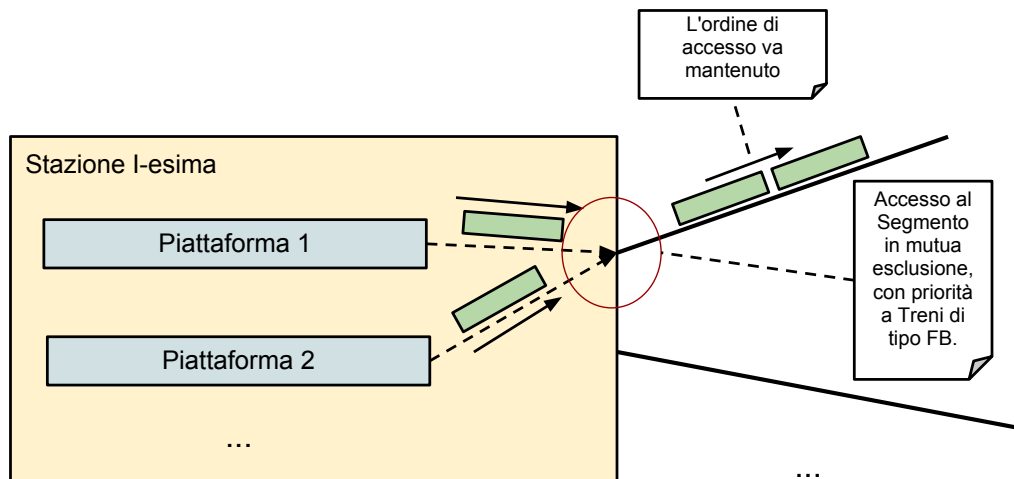


Figura 2.1: Accesso ad un Segmento da Parte di uno o più Treni.

L'accesso ad un segmento di collegamento tra due stazioni da parte di un Treno, è inerentemente concorrente. Tale azione presenta infatti i seguenti requisiti:

- L'ingresso presso un Segmento deve avvenire in *mutua esclusione*; è infatti impossibile che due o più entità Treno accedano ad uno stesso Segmento contemporaneamente.
- Più Treni possono circolare su un Segmento contemporaneamente. Questo comporta:
  - il mantenimento di un ordine di ingresso al Segmento;
  - la regolazione della velocità di transito di ciascun Treno in base alla velocità di quelli che lo precedono;
  - l'impossibilità di un Treno di accedere ad un Segmento qualora vi siano altri Treni che lo percorrono in senso opposto.
- Dev'essere data precedenza d'accesso al Segmento, ai Treni di tipo FB.

## 2.2 Uscita da un Segmento e accesso alla Stazione

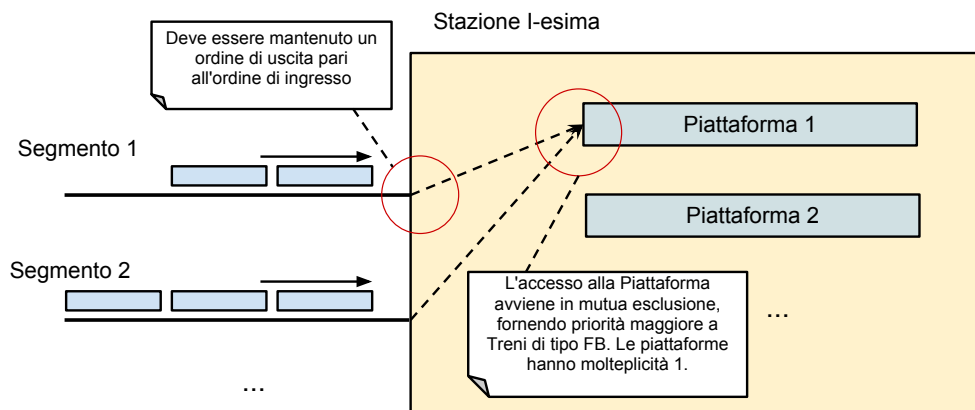


Figura 2.2: Uscita da un Segmento e accesso ad una Piattaforma di uno o più Treni.

Il problema introdotto in sezione 2.1, vincola i requisiti che dovranno essere soddisfatti relativamente all'uscita da un Segmento e conseguente accesso alla Stazione successiva. Per quanto riguarda l'uscita dal Segmento avremo quindi i seguenti requisiti:

- L'ordine di ingresso al Segmento dev'essere mantenuto all'uscita.
- L'ordine di uscita da un Segmento dev'essere mantenuto nell'accesso alla stazione successiva da parte dei Treni in transito.

L'ingresso in una Stazione, permette ad un Treno di occupare una della Piattaforme disponibili e, dato che da specifica una stazione può essere raggiunta da più Segmenti, tale azione sarà svolta in modo concorrente tra i treni in uscita dai vari Segmenti, secondo i seguenti vincoli:

- I Treni di tipo FB avranno priorità maggiore nell'occupare una Piattaforma.
- Ciascuna Piattaforma è acceduta in mutua esclusione, e può essere occupata da un solo Treno alla volta.

Questo significa che l'accesso ad una Piattaforma sarà ordinato, per tutti i Treni provenienti dallo stesso Segmento, in base all'ordine di uscita da quest'ultimo, e concorrente tra Treni provenienti da Segmenti diversi.

## 2.3 Acquisto di un Biglietto da parte di un Viaggiatore

Ciascun Viaggiatore deve acquistare un biglietto prima di poter usufruire dei servizi ferroviari. Per fare ciò, all'interno ogni stazione vi è una Biglietteria presso la quale l'acquisto può essere effettuato. Un biglietto sarà composto da una serie di Tappe, ciascuna relativa ad un tratto del percorso da portare a termine con uno specifico Treno, sia Regionale che FB. L'assegnazione di un Biglietto ad un Viaggiatore è semplice se il suo percorso prevede solo l'utilizzo di Treni Regionli, mentre è più complesso se vi sono tappe da raggiungere con Treni FB, i cui biglietti vengono erogati se e soltanto se vi è ancora posto all'interno del treno. É quindi necessario prevedere l'esistenza di una *biglietteria centrale* che mantenga le prenotazioni dei Treni di tipo FB. Si ricavano quindi i seguenti requisiti:

- La biglietteria centrale va acceduta in mutua esclusione, in modo da evitare prenotazioni inconsistenti di Biglietti.
- L'erogazione di biglietti può fallire in caso non vi siano posti per percorrere alcune tappe; il sistema deve reagire di conseguenza.

# Capitolo 3

## Soluzione

Di seguito, verrà presentata la soluzione realizzata, in termini di progettazione come sistema distribuito e concorrente.

### 3.1 Architettura Logica di Distribuzione

Un diagramma informale delle componenti distribuite che compongono il sistema è presentato in figura X. Di seguito verranno descritte le principali.

#### 3.1.1 Regioni

La simulazione è stata suddivisa in Regioni, le quali risiederanno su nodi di calcolo diversi. Questa scelta aggiunge i seguenti requisiti minimi alla specifica iniziale:

- I Treni, se previsto dal percorso, possono viaggiare da una Regione all'altra.
- I Passeggeri possono raggiungere destinazioni in Regioni diverse.
- Esisteranno stazioni che chiameremo di Gateway che permettono a Treni e Passeggeri di raggiungere Regioni diverse.
- Deve essere garantita consistenza temporale nel passaggio da una Regione ad un'altra.

Da questa scelta consegue inoltre l'introduzione di un semplice Server dei Nomi che mantiene traccia di ciascuna Regione, in modo tale da rendere agevole la risoluzione della locazione alla quale l'entità si trova.



#### 3.1.2 Biglietterie

Per poter gestire meglio la definizione di un percorso e l'erogazione di un Biglietto per un Viaggiatore, ho pensato di introdurre una gerarchia su due livelli, di Biglietterie. Ci saranno dunque tre categorie di biglietterie:

##### **Biglietterie di Stazione**

Forniscono un'interfaccia adeguata ai Viaggiatori per poter acquistare un biglietto.

##### **Biglietterie Regionali**

Hanno conoscenza regionale della topologia del grafo composto da Stazioni e Segmenti.

##### **Biglietteria Centrale**

Ha conoscenza di più alto livello; in particolare, essa mantiene traccia delle connessioni tra le varie regioni ( ovvero i collegamenti tra Stazioni di Gateway di regioni diverse).

#### 3.1.3 Controller Centrale

Il Conterollo Centrale è una entità distribuita, alla quale tutti i nodi inviano Eventi per notificare lo stato di avanzamento globale della simulazione. Esso fornisce una interfaccia alle varie Regioni per ricevere gli Eventi, ed un'interfaccia per permettere a client remoti di poter visualizzare gli effetti di tali Eventi. Quest'ultima possibilità è ottenuta mediante un meccanismo di tipo Publish/Subscribe, attraverso il quale client remoti possono registrarsi presso il Controller per ricevere, in modalità Push, gli Eventi. In questo modo è possibile per un qualsiasi client interfacciarsi al Controller e fornire, ad esempio, una rappresentazione grafica della simulazione.

## 3.2 Architettura Logica Concorrente

Internamente a ciascuna Regione di simulazione, sono definite entità che modellano le interazioni previste dalla specifica del problema. Di seguito sono descritte le principali.

### 3.2.1 Segmento

Un Segmento (**Segment**) è modellato come una *entità reattiva con agente di controllo a molteplicità  $N$* , dove  $N$  è il numero massimo di utilizzatori, la quale fornisce un'interfaccia utilizzabile da entità attive di tipo Treno per accedere ed uscire in maniera ordinata. Esso è caratterizzato da:

- le due stazioni che esso collega;
- una lunghezza;
- una velocità massima di percorrenza.

### 3.2.2 Viaggiatore

In prima analisi, un Viaggiatore **Traveler** può essere modellato come una *entità Attiva* che esegue una sequenza di operazioni semplici. La modellazione dell'entità Viaggiatore come Attiva non può però essere semplicemente associata ad un processo, soprattutto in presenza di un modello di distribuzione come quello presentato in sezione 3.1.1. In questa ipotesi infatti, nel caso in cui un Viaggiatore si spostasse da una Regione ad un'altra, avremmo a disposizione solo due possibili soluzioni:

- La migrazione del processo che rappresenta il Viaggiatore sul nodo (Regioni) di destinazione, come distruzione del processo sul nodo di partenza e creazione dinamica dello stesso sul nodo destinazione. Questa operazione è in generale computazionalmente molto costosa.
- La replicazione del processo che rappresenta il Viaggiatore su tutti i nodi, e l'attivazione, intesa come cambio di stato del processo in modo tale che possa competere per la CPU; tale soluzione è molto costosa in termini di memoria utilizzata, e non scala all'aumentare del numero di passeggeri.

La soluzione adottata consiste nel disaccoppiare le operazioni svolte da ciascun Viaggiatore dal processo che le esegue, prevedendo una struttura dati costituita da:

- un *pool di M processi* dimensionato in maniera opportuna;
- una *coda di operazioni* che man mano vengono estratte ed eseguite dai processi nel pool.

In questo modo è sufficiente replicare per ciascun Viaggiatore, su tutti i nodi che compongono il sistema, una struttura dati che contiene i suoi dati e le operazioni che esso eseguirà. Il cambio di Regione di un Viaggiatore sarà quindi potrà essere ottenuto semplicemente inserendo la prossima operazione da eseguire per il Viaggiatore nella coda del pool di processi del nodo destinazione.

#### 3.2.3 Treno

Un Treno (**Train**) è una *entità Attiva*, la quale esegue ciclicamente un numero finito di operazioni. Ciascuna entità Treno può appartenere a due categorie, FB a priorità più alta, e Regionale a priorità minore, ed è caratterizzata da:

- un identificativo univoco;
- una capienza massima;
- una velocità massima raggiungibile;
- una velocità corrente;

Anche in questo caso come per le entità di tipo Viaggiatore, ciascuna entità non viene mappata su un singolo processo. Infatti anche per le entità di tipo Treno ciò comporterebbe una complessa e costosa, in termini computazionali e di memoria, gestione del passaggio da un nodo (Regione) ad un altro. Ho optato invece nel progettare una struttura dati che mantiene:

- una coda di Descrittori di Treno, che contengono tutte le informazioni che distinguono una entità Treno;
- un pool di processi a dimensionato in maniera opportuna; ciascun processo sarà tale per cui una volta ottenuto un descrittore dalla coda, eseguirà per lui una fissata sequenza di azioni.

In questo modo un entità Treno eseguirà all'interno di un nodo se e soltanto se il suo descrittore sarà inserito nella coda di descrittori della struttura dati descritta.

#### 3.2.4 Stazione

Una Stazione (**Station**) è modellata come una struttura dati contenete:

- un certo numero  $n \geq 2$  di Piattaforme (**Platform**);
- una Biglietteria (**Ticket\_Office**);
- un Pannello Informativo (**Notice\_Panel**).

Essa offre una interfaccia alle entità Treno e Viaggiatore per l'accesso a Piattaforme e Biglietteria.

#### Piattaforma

Una Piattaforma è modellata come una *entità reattiva con agente di controllo, a molteplicità 1*. Essa espone un'interfaccia che permette alle entità Treno di potervi sostare ed effettuare discesa e salita delle entità Passeggero o di poter superare la stazione, e alle entità Passeggero di accodarsi in attesa di uno specifico Treno.

#### Biglietteria

Una Piattaforma è modellata come una interfaccia che permette al Viaggiatore di acquisire un Biglietto di viaggio.

#### Pannello Informativo

Una Piattaforma è modellata come una *entità reattiva con agente di controllo, a molteplicità 1*. Esso espone una interfaccia tale da permettere alla stazione di notificare lo stato delle entità Treno che stanno arrivando, quelle in sosta e quelle in partenza.

### 3.3 Interazione tra le Entità

#### 3.3.1 Viaggio di un Viaggiatore

Alla luce della definizione dell'entità Viaggiatore in sezione 3.2.2, è possibile definire le azioni che compongono il viaggio:

- Acquisto di un Biglietto (**Ticket**) presso la Biglietteria della Stazione (**Ticket\_Office**) di partenza. Ciascun Biglietto è composto da una sequenza ordinata di Tappe (**Ticket\_Stages**), ciascuna contenente:

- `start_station`
  - `next_station`
  - `train_id`
  - `start_platform`
  - `destination_platform`
- Una volta ottenuto un Ticket, vengono eseguite le seguenti operazioni per ciascuna Tappa del Biglietto:
    - Accodamento presso la Piattaforma `start_platform` della Stazione `start_station` in attesa del Treno `train_id`.
    - All'arrivo del treno `train_id`, Accodamento presso la Piattaforma `destination_platform` della Stazione `next_station` in attesa dell'arrivo di `train_id`.

Le tre operazioni sopraelencate possono essere incapsulate in strutture dati che rappresentano una specifica operazione. In questo modo viene generata una gerarchia di operazioni, tutte derivanti da un'unica operazione generica.

#### 3.3.2 Viaggio di un Treno

A ciascuna entità Treno, è assegnato un Percorso (**Route**) di andata e un Percorso di ritorno, ovvero sequenze di Tappe (**Stage**) successive, ciascuna composta dalla n-upla:

`<next_segment,next_station,next_platfom,action>`

dove `action` indica quello che un Treno dovrà compiere presso la prossima stazione, a scelta tra **ENTER**, per entrare ed effettuare discesa e salita passeggeri o **PASS** per non fermarsi e oltrepassare la Stazione.

Le operazioni effettuate sono, per ciascuna Tappa del Percorso corrente (di andata o di ritorno):

- Accesso al prossimo Segmento `next_segment` previsto.
- Percorrenza all'interno del Segmento come attesa finita di durata proporzionale alla lunghezza del Segmento e alla velocità massima alla quale il Treno può percorrerlo.
- Uscita dal Segmento e richiesta di Accesso alla Stazione successiva (`next_station`) presso la Piattaforma indicata da `next_platfom`, per eseguire l'azione `action`.

### 3.3. INTERAZIONE TRA LE ENTITÀ

---

- Se `action = ENTER` allore effettua discesa e salita dei Viaggiatori in attesa dell'arrivo del Treno.
- Uscita dalla Piattaforma `next_platfom`.