Programmazione con Oggetti Distribuiti: Java RMI

Vittorio Scarano

II edizione - Marzo 2010

A mio padre e a mia madre, ispiratori

A mia moglie, dolce compagna, amica e confidente

Ai miei tre figli, gioielli e luce dei miei occhi

EMERGENCY è un'associazione italiana indipendente e neutrale. EMERGENCY offre assistenza medico-chirurgica gratuita e di elevata qualità alle vittime civili delle guerre, delle mine antiuomo e della povertà.
Dal 1994 più di 3 milioni di persone sono state curate gratuitamente in ospedali, cliniche e centri di riabilitazione di EMERGENCY dove inoltre è garantita la formazione del personale medico locale.
Tre ospedali, un centro di maternità e 29 posti di primo soccorso in Afganistan, un ospedale e un posto di primo soccorso in Cambogia, un centro di riabilitazione in Iraq, un poliambulatorio per migranti a Palermo, un ospedale e un centro pediatrico in Sierra Leone, una clinica pediatrica e un centro di cardiochirurgia in Sudan, due centri pediatrici in Darfur e in Repubblica Centrafricana, e un centro di Ostetricia e Ginecologia in Nicaragua in costruzione.
Ecco gli ospedali che Emergency ha voluto, ha costruito e tuttora gestisce grazie al contributo di migliaia di persone.

EMERGENCY

Via Gerolamo Vida 11
20127, Milano
tel. 02/881881 - fax 02/86316336

http://www.emergency.it
e mail: info@emergency.it

Conto corrente postale
intestato a "EMERGENCY Ong Onlus"
n. 2842 6203
IBAN: IT37 Z076 0101 6000 0002 8426 203

Conto corrente bancario
intestato a "EMERGENCY Ong Onlus"
IBAN: IT 02 X 05018 01600 000000130130
presso Banca Etica, Filiale di Milano

L'autore devolve interamente i propri proventi in favore dei progetti di Emergency.

Indice

Prefazione

Obiettivi

Questo libro su Java RMI ha due obiettivi. Il primo è quello di fornire allo studente il collegamento tra i tradizionali corsi di programmazione di rete, che sono basati sulla comunicazione basata su TCP/IP ed i socket, e i corsi di programmazione di applicazioni distribuite che tipicamente usano strumenti di alto livello come il modello a componenti distribuite (ambienti Enterprise) oppure il modello orientato ai servizi (ambienti Web Services). Quello che manca, e che si intende fornire con questo libro, è il collegamento che permette allo studente di comprendere in quale maniera Java Enterprise Edition, ad esempio, realizza la infrastruttura di comunicazione tra oggetti, utilizzando Java Remote Method Invocation oppure quanto del modello delle architetture orientate ai servizi (Service Oriented Architecture) derivi direttamente dai modelli di invocazione remota strettamente accoppiati (come Java RMI).

Quindi, il viaggio che si intende far intraprendere allo studente illustrerà come la programmazione di rete serve alla invocazione remota di metodi che servono poi a realizzare la infrastruttura di comunicazione. In un certo senso, si vuole usare la programmazione di socket e la invocazione remota di metodi come la programmazione assembly per la programmazione con un linguaggio evoluto, illustrando i meccanismi e le tecnologie coinvolte nell'ambiente Java. Tutto questo passerà anche attraverso la presentazione di alcuni concetti fondamentali per le architetture di sistemi distribuiti per contestualizzare la presenza di Java RMI.

Il secondo obiettivo è quello di fornire le basi per la programmazione distribuita di oggetti, sviluppando quindi un certo numero di esempi pratici che guideranno lo studente, a partire dalla implementazione con i socket, a realizzare semplici applicazioni che però serviranno ad evidenziare problematiche tipiche dei sistemi client-server e dei sistemi peer2peer.

Dal punto di vista dello stile di programmazione, la scelta è stata quella di evitare di sovraccaricare il codice, senza presentare, ad esempio, interfacce grafiche ma limitandosi alla shell in modo da limitare la quantità di codice da presentare. Il cosiddetto *coding style* di Java viene utilizzato, anche se, per ragioni di spazio e di efficacia grafica, molti dei commenti nel codice sono ridotti all'osso, e mancano del tutto i commenti JavaDoc. Questo stile di programmazione (utilizzato per rendere più agevole lo studio e la presentazione grafica) non è da ritenere un suggerimento, anzi! Sia ben chiaro allo studente che commenti adeguati e informativi sono estremamente utili per la manutenzione e la evoluzione del software e che la presentazione stringata è esclusivamente per limitare la lunghezza del codice stampato.

Struttura del libro

Il libro consta di tre parti. Nella prima si introduce Java Remote Method Invocation attraverso l'utilizzo dei socket. Nella seconda si affrontano i diversi esempi di programmazione

usando Java RMI, allo scopo di introdurre tecniche e metodi utili per la realizzazione di semplici applicazioni remote. Nella terza parte si trovano alcuni approfondimenti sulla struttura interna di Java RMI, con la capacità di gestire a run-time le invocazioni, e RMI-IIOP.

A chi si rivolge il libro

Il libro può essere utilizzato come testo di riferimento per corsi universitari di Programmazione Distribuita di I livello (terzo anno) oppure di II livello (primo anno) in Informatica e Ingegneria Informatica. Si suppone che gli studenti abbiano una buona conoscenza di Java, con esperienza nell'uso di Integrated Develoment Environment (IDE) e qualche conoscenza di base di Ingegneria del Software. Tra le competenze specifiche di Java che sono necessarie, una qualche semplice esperienza con i socket (anche in altri linguaggi) e una semplice conoscenza dei thread.

Il materiale del libro può essere usato per un corso da 4-6 CFU, a seconda del livello di dettaglio e delle esercitazioni da fare in laboratorio (qualora previsto).

Gli strumenti utilizzati

Composizione tipografica. Per scrivere questo libro è stato utilizzato LaTeX ed in particolare la versione per Windows distribuita con MiKTeX 2.8 (`http://www.miktex.org`). TeXlipse 1.2.2 (`http://texlipse.sourceforge.net/`) è un plugin di Eclipse che è stato usato per facilitare la composizione.

Per la traduzione di programmi Java in LaTeX si è utilizzato, invece, Java2HTML versione 1.5.0 (`http://www.java2html.de`) che permette anche la conversione da Java in LaTeX, con il syntax-coloring ed è disponibile anche come plugin per Eclipse. Per la creazione delle immagini, si usa il plugin di Eclipse eDump 1.7.1 (`www.bdaum.de`) che fornisce la possibilità di catturare immagini delle varie componenti di Eclipse con notevole flessibilità.

Linguaggio e ambiente integrato di sviluppo. In generale, si è utilizzata la ultima versione disponibile al momento della Java VM di Sun, vale a dire la versione Java SE Development Kit (JDK) 6 Update 11. Insieme ad essa, suggeriamo di scaricare in locale la documentazione Java disponibile sul sito, che porta una notevole quantità di informazioni e di guide per i vari argomenti.

Per la scrittura e la documentazione dei programmi di esempio si + usato Eclipse (`http://www.eclipse.org`), versione 3.4.1 con il plugin per eUML2 Free di Soyatec versione 3.2.1 (`http://www.soyatec.com/euml2/`).

Convenzioni

L'uso e l'abuso della lingua inglese. In questo libro si cerca di evitare l'uso di forzate traduzioni di termini inglesi che sono oramai diventati di uso tecnico. Ben lungi dal prefigurarsi come sudditanza linguistica verso oltreoceano, questa scelta è motivata dal fatto che questo ben prepara gli studenti di oggi che, da professionisti dell'informatica di domani, si troveranno, necessariamente, a contatto con documentazione di tecnologia recente che sarà disponibile, probabilmente, solamente in inglese tecnico.

Convenzioni tipografiche. Per indicare una sessione di lavoro ad una shell (come ad esempio per la compilazione o esecuzione di un programma) si è scelto di utilizzare come esempio una semplice shell DOS[1]. Per rappresentare una sessione useremo la seguente convenzione:

```
Z:\>echo Ciao Ciao


Z:\>
```

Per indicare un listato di una classe Java di nome `Hello.java` si utilizzerà questo standard:

HelloWorld.java

```
1 // Una classe di esempio
2 public class HelloWorld {
3   public static void main(String[] args) {
4     System.out.println("Hello World!");
5   }
6 }
```

Fine: HelloWorld.java

In generale, comandi da shell vengono indicati in grassetto, come ad esempio, il compilatore Java **javac**.

[1] In alternativa, si può utilizzare un plugin per Eclipse chiamato WickedShell (`http://www.wickedshell.net`) che permette di creare shell molto flessibili come view di Eclipse.

Prefazione alla seconda edizione

Questa edizione aggiunge alla precedente la terza parte, con gli argomenti avanzati: gestione dinamica e RMI-IIOP. Ovviamente, sono stati corretti anche gli errori, così come segnalati dai precisi e puntuali commenti che gli studenti del corso di Programmazione Distribuita del corso di Laurea in Informatica dell'Università di Salerno hanno fornito, durante l'anno accademico 2008-2009.

Ringrazio sentitamente tutti gli studenti che hanno inviato commenti, suggerimenti, correzioni, perchè il libro ne ha beneficiato notevolmente, e con questo, spero, ne beneficeranno anche i loro colleghi che lo useranno per lo studio negli anni successivi.

Anche i diritti di questa II edizione saranno interamente versati ad Emergency, nel ricordo di Teresa Sarti Strada, scomparsa il 1^o settembre 2009.