

Monitoraggio di MAS distribuiti attraverso grafica 3D

Installazione e Uso

Matteo Campana

Indice

Introduzione	3
Installazione	4
Guida all'uso.....	6
3.1. Livello Organizzazione	6
3.2. Livello Nodo.....	7
3.3. Livello Tuple.....	9

Capitolo 1

Introduzione

L'applicazione ha lo scopo di monitorare l'evoluzione di sistemi distribuiti intelligenti attraverso una rappresentazione grafica tridimensionale delle entità. Considerando sistemi MAS (Multi-Agent System) realizzati attraverso *Tuple Space model* (TuCSoN) diventa interessante la rappresentazione di Nodi, Tuple-Center e agenti. Il progetto nasce quindi dalla necessità di avere uno strumento che offra una rappresentazione semplice ed intuitiva delle varie entità, dalla quale ottenere utili informazioni sulla distribuzione, strutturazione ed evoluzione dei MAS esaminati. Per soddisfare questa necessità, l'utente ha la possibilità di accedere a tre livelli distinti:

- **Livello organizzazione:** monitoraggio di un insieme di nodi
- **Livello Nodo:** monitoraggio di uno specifico nodo (a questo livello si avrà visione di agenti e Tuple Center)
- **Livello Tuple:** monitoraggio di uno specifico agente o tuple center (visione di tuple)

Per descrivere l'evoluzione e il comportamento dei sistemi monitorati vengono sfruttate le proprietà di ambienti grafici 3D (colori, trasparenze, animazioni ecc...).

L'applicazione è a sua volta un MAS che utilizza TuCSoN come infrastruttura. Gli agenti sono realizzati in gran parte utilizzando il linguaggio Java. Attraverso java è infatti possibile accedere ai numerosi servizi offerti dalla libreria grafica Java3D con la quale modellare visivamente il sistema distribuito osservato. Per realizzare le funzioni di *governing ed enabling* del sistema si sono utilizzati agenti tuProlog.

Capitolo 2

Installazione

L'applicazione è stata realizzata utilizzando diversi linguaggi di programmazione e librerie (Java, Java3D, tuProlog e TuCSoN). E' quindi necessario per prima cosa verificare che tutte le librerie siano presenti nella sistema sul quale si eseguirà l'installazione; in caso contrario scaricare le ultime versione attraverso le seguenti pagine web:

<http://java.sun.com/products/java-media/3D/>
<http://www-lia.deis.unibo.it/Research/TuCSoN/>
<http://lia.deis.unibo.it/Research/2P/>

Per quanto riguarda il TuCSoN sarà necessario utilizzarne una versione che supporti l'*osservabilità*.

Decomprimere il file zip dell'applicazione (monitoring.zip) in una directory a scelta (*dir*). Verificare che all'interno vi siano i seguenti file:

- monitoring.jar
- floor.gif
- theory.pl
- start.bat

Il file `monitoring.jar` contiene tutte le classi java utilizzate, compresa la classe `main` utilizzata per eseguire l'applicazione. `Floor.gif` è la texture applicata allo sfondo dell'ambiente tridimensionale. `Theory.pl` contiene una teoria prolog invocata inizialmente e capace di definire regole. Il file `start.bat` è un esempio di come lanciare facilmente l'applicazione. Vi sarà inoltre una cartella documentazione contenente questo manuale e la documentazione Javadoc relativa alle classi java.

Dopo aver installato il tutto nella directory *dir* digitare il seguente comando, assicurandosi che le directory contenenti le librerie di tuProlog e TuCSoN siano visibili:

```
java -cp monitoring.jar view.GraficHostAgent
```

In alternativa è possibile specificare le directory contenenti le classi tuProlog e TuCSoN direttamente nel comando:

```
java -cp dir1\2p.jar; dir2\tucson.jar; dir\monitoring.jar  
view.GraficHostAgent
```

dove `dir1` e `dir2` sono rispettivamente le directory contenenti le librerie `2p.jar` e `tucson.jar`, mentre `dir` è la directory di installazione. Come esempio si apra il file `start.bat` e se ne osservi il contenuto:

```
java -cp C:\tuProlog\lib\2p.jar; C:\tucson\lib\tucson.jar;  
C:\observer\monitoring.jar view.GraficHostAgent
```

Ricordando che `monitoring.jar` è un file jar eseguibile, allora con il seguente comando

```
java -jar monitoring.jar
```

è possibile accedere in automatico al programma di monitoring.

Capitolo 3

Guida all'uso

Se l'installazione è stata effettuata nel modo corretto, il comando di lancio dovrebbe aprire una GUI che rappresenta il *livello organizzazione* del monitoraggio.

3.1. Livello Organizzazione

La finestra grafica che appare fig. 3.1 rappresenta l'insieme dei *nodi* che è possibile monitorare.

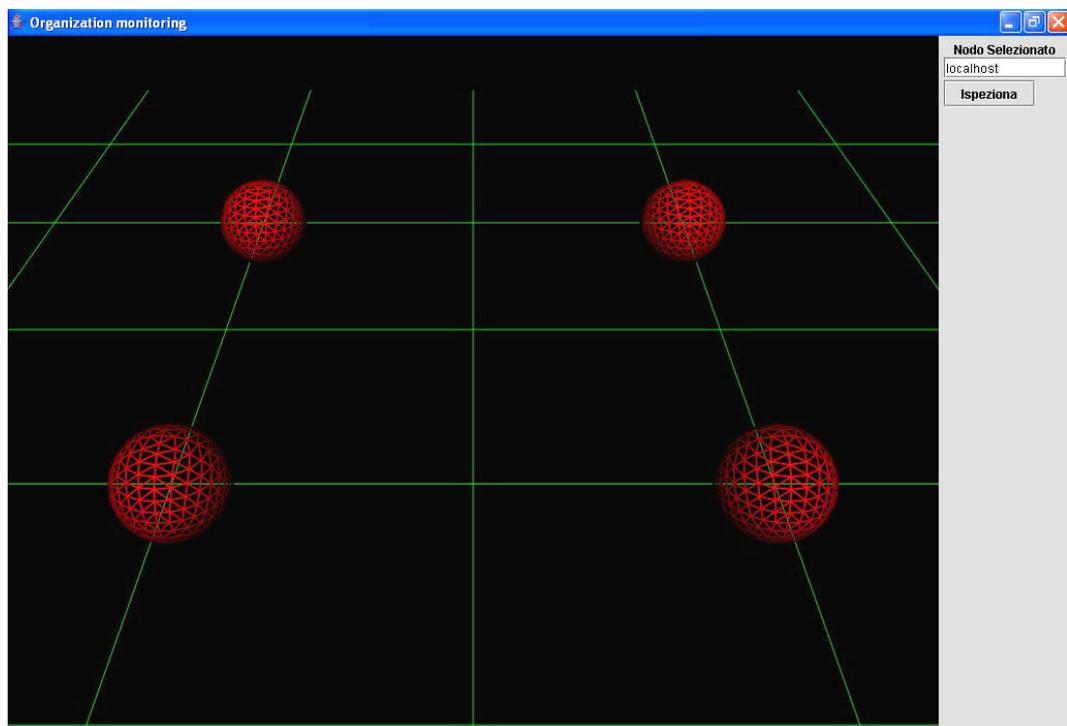


Figura 3.1

Le sfere rosse rotanti rappresentano i nodi TuCSon. Per muoversi nell'universo tridimensionale, l'utente dovrà agire sulla tastiera. La seguente tabella elenca le funzioni associate ad alcuni tasti:

Tasto	Movimento	Tasto+Alt
←	Ruota a sinistra	Trasla a sinistra
→	Ruota a destra	Trasla a destra
↑	Muovi avanti	
↓	Muovi Indietro	
Pag-Up	Ruota verso l'alto	Trasla in alto
Pag-Down	Ruota verso il basso	Trasla in basso

Tabella 3.1

Per selezionare un nodo è sufficiente portare il puntatore del mouse sopra lo stesso; trascinando il puntatore con il tasto sinistro si otterrà il nome (indirizzo) del nodo. Il nome apparirà nella casella di testo in alto a destra. Premendo il pulsante *ispeziona* si aprirà un'ulteriore interfaccia grafica che permetterà di osservare l'evoluzione di *Tuple Center* e *agenti* presenti nel nodo selezionato; si ha in questo modo accesso al secondo livello di monitoring: *Livello nodo*.

3.2. Livello Nodo.

Lo scenario nel quale si è immersi sarà simile a quello di fig. 3.2.

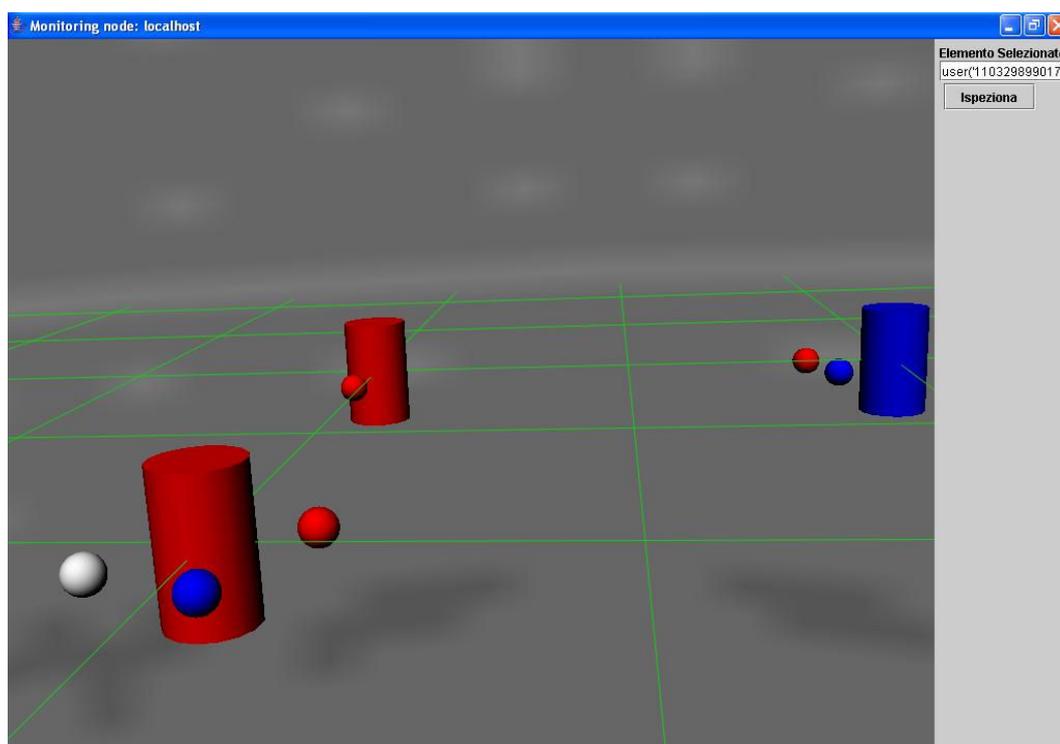


Figura 3.2

Gli agenti sono rappresentati da piccole sfere mentre i tuple center da cilindri di dimensioni maggiori. All'inizio dell'osservazione le entità vengono distribuite nello spazio con un criterio pseudo-casuale. La stessa procedura è eseguita anche per agenti e Tuple center che interagiscono per la prima volta sul sistema monitorato. In seguito gli agenti incominceranno a muoversi verso i TC (Tuple Center) con i quali comunicano. Dopo pochi istanti l'universo tenderà a stabilizzarsi in uno stato in cui gli agenti sono posizionati nelle vicinanze dei TC più utilizzati. In questo modo l'utente è agevolato nel riconoscere gruppi di agenti e TC tra loro correlati e presumibilmente appartenenti allo stesso MAS.

Per rappresentare al meglio l'evoluzione delle entità (o gruppi di entità) l'applicazione modifica anche l'aspetto degli oggetti 3D. In particolare si è deciso di associare il colore e la trasparenza delle entità presenti in base al loro stato o al comando da loro eseguito. Come si nota da fig. 3.2 gli agenti hanno tra loro colorazioni differenti; l'agente in basso a sinistra ad esempio è congelato (bianco) perché è inattivo. E' chiaro che attraverso l'uso di opportuni aspetti visivi diventa

immediato per l'utente individuare le entità di interesse. La seguente tabella mostra i colori associati ai possibili comandi TuCSon.

Apparenza	Comando
Rosso	Out
Verde	Rd
Blu	In
Arancione	Inp
Giallo	Rdp
Marrone	Set_spec
Grigio	Get_spec

Tabella 3.2

Tutti i comandi (tranne la out) saranno in un primo tempo richiesti dagli agenti e poi soddisfatti dal relativo TC. Per discriminare questo stato si è utilizzata la trasparenza: quando un agente farà una richiesta al TC il suo aspetto diventerà trasparente e del colore associato al comando eseguito. Nel momento in cui verrà soddisfatta la sua richiesta, perderà la trasparenza. Per chiarire supponiamo di essere nello scenario descritto in fig. 3.3 dove un agente ha appena eseguito un comando di in (es. `tc_1 ? in(Ciao)`). Esso rimarrà trasparente fino a quando nel TC verrà inserita una tupla di tipo `Ciao`. Quando questo accadrà il suo aspetto diventerà quello di fig. 3.4.

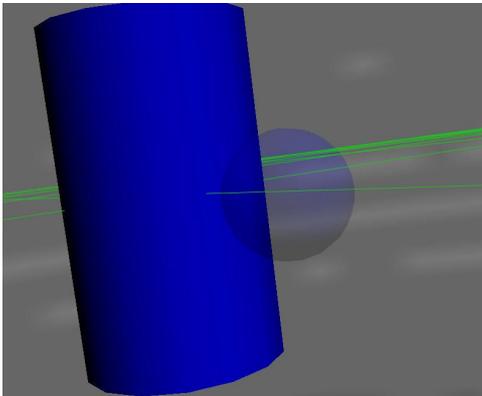


Figura 3.3

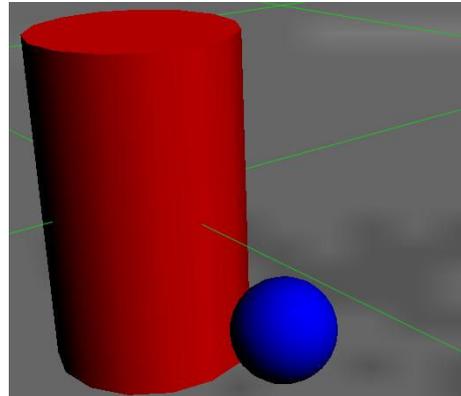


Figura 3.4

E' interessante notare anche l'evoluzione dell'aspetto del TC: in fig. 3.3 è di colore blu perché ha ricevuto una richiesta `in(Ciao)`; nell'altra figura è rosso perché ha ottenuto la tupla richiesta attraverso un comando `out` proveniente probabilmente da un secondo agente.

Come nell'interfaccia grafica del *livello organizzazione* l'utente ha la possibilità di navigare nell'ambiente tridimensionale utilizzando la tastiera. Un aggiuntivo servizio offerto a questo livello è la possibilità di muovere o ruotare agenti o TC a piacimento. L'utente sarà quindi libero di spostare agenti non attivi, lasciando spazio a quelli attivi. Potrà inoltre avvicinare due o più TC correlati e osserverà che gli agenti seguiranno il TC nella nuova posizione. Per poter modificare la posizione degli oggetti, l'utente utilizzerà il mouse; le risposte alle azione dell'utente sono elencate nella seguente tabella.

Azione del mouse	Risposta all'azione del mouse
Movimento del mouse tenendo premuto il tasto sinistro	Ruota l'oggetto
Movimento del mouse tenendo premuto il tasto destro	Trasla l'oggetto lungo un piano parallelo alla View (punto di vista) dell'utente
Movimento del mouse tenendo premuto il tasto centrale	Trasla l'oggetto lungo un piano ortogonale alla view

Tabella 3.3

3.3. Livello Tuple

LAVORI IN CORSO...per il momento è possibile utilizzare il TuCSoN inspector.