

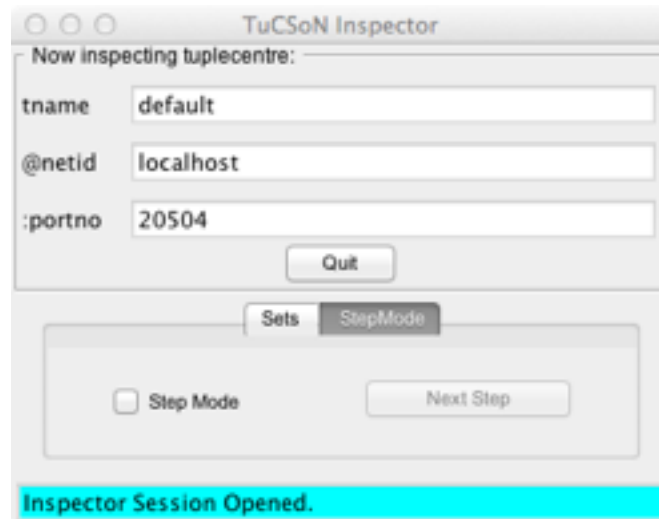
INTERAZIONE TRA L'INSPECTOR E IL CICLO DI VITA DELLA MACCHINA VIRTUALE RESPECT

Lo scopo del progetto è quello di implementare l'interazione tra il tool Inspector e la macchina virtuale ReSpecT: più precisamente, si vogliono inserire dei opportuni "hook" all'interno del ciclo di vita della macchina a stati per consentire all'Inspector di metterla in pausa e farla proseguire step-by-step (ad esempio consumando un solo evento e poi aspettando il via dall'Inspector).

StepMode Design

Per consentire alla macchina virtuale ReSpecT di evolvere in maniera step-by-step, si è introdotta una nuova modalità chiamata step mode, attivabile tramite una checkbox direttamente dall'interfaccia grafica dell'Inspector sullo specifico centro di tuple: essa permette di addormentare la macchina a stati dopo ogni step completato.

Il comportamento della macchina virtuale ReSpecT influenza, chiaramente, tutti gli Inspector che sono eventualmente attivi nell'ispezionare quello specifico centro di tuple e inoltre, a maggior ragione, anche le loro relative finestre selezionabili dal pannello "sets": se la macchina virtuale, a seguito dell'attivazione della modalità step, dovesse trovarsi in uno stato di pausa, tutti i relativi Inspector sperimenterebbero il comportamento a step del centro di tuple. Per questo, si è scelto di inserire il pannello di controllo della nuova modalità "stepMode" all'interno dell'interfaccia principale dell'Inspector, insieme al pannello "sets".



Il pannello contiene una CheckBox per l'attivazione della modalità e un pulsante per eseguire lo step successivo.

Per gli stessi motivi citati poc'anzi, l'implementazione della modalità step non poteva sfruttare l'InspectorProtocol.

L'InspectorProtocol, viene utilizzato dall'Inspector per comunicare al centro di tuple quali informazioni egli è interessato; esso però è specifico per ogni Inspector in quanto dipende dalla sua attuale configurazione (quali finestre ha attivato ecc..). La soluzione adottata, perciò, consiste nell'inviare un opportuno messaggio al nodo, dove verrà processato richiamando il relativo metodo (in questo caso "setStepMode").

Se la modalità non era già attiva, il metodo attiverà un flag e attraverso l'utilizzo di un monitor chiamato "step" farà sospendere il thread che gestisce il ciclo di vita ReSpecT.

//Executes a virtual machine behaviour cycle

```
public void execute() {
    while (!this.currentState.isIdle()) {
        this.currentState.execute();
        this.currentState = this.currentState.getNextState();
        if (this.rvm.hasInspectors()) {
            this.rvm.notifyInspectableEvent(new InspectableEvent(this,
                InspectableEvent.TYPE_NEWSTATE));
        }
        if (this.isStepMode()) {
            try {
                this.step.awaitEvent();
            } catch (final InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
```

```
}  
}  
}  
-----
```

Come vedremo più avanti, durante questa fase è necessario spedire a tutti gli Inspector attivi sul centro di tuple un ulteriore messaggio allo scopo di avvertirli della attivazione di questa modalità.

Il pulsante `nextStep` non fa altro che eseguire una signal sul monitor, sbloccando così temporaneamente la macchina virtuale.

Coordinazione tra gli Inspector

Quando ispeziono lo stesso centro di tuple con un ulteriore Inspector, devo verificare se la virtual machine si trova già in modalità step oppure no, così da abilitare o disabilitare di conseguenza la checkbox “`stepMode`” che verrà visualizzata.

Questo viene fatto inviando un messaggio “`isActiveStepModeMsg`” al momento del click sul pulsante “Inspect!” con la stessa modalità vista prima.

Attivando/disabilitando la checkbox, però, verrebbe avvertito il relativo listener che ha il compito di inviare il messaggio al node per l’attivazione o la terminazione della modalità step.

In questo caso, perciò, è necessario utilizzare un flag “`onRefreshCB`” per evitare l’invio del messaggio.

Come detto in precedenza, nel processo di attivazione della modalità step viene inviato un ulteriore messaggio a tutti gli Inspector attivi su quel nodo, allo scopo di coordinare l’attivazione / disabilitazione della checkbox rappresentante di tale modalità.

Un’ulteriore complicazione si ha quando viene eseguita una quit per poter ispezionare un altro centro di tuple; in questo caso è necessario deselectare sempre la checkbox in modo tale da non ritrovarla selezionata nel caso in cui si decida di ispezionare un nodo dove non è attiva la modalità step. Se questa fosse attiva, invece, allora la checkbox verrebbe risSelected secondo le modalità viste in precedenza.

Per i scenari sopra descritti, sono stati aggiunti i metodi `selectStepModeCB` e `deselectStepModeCB` che, gestendo opportunamente il flag `onRefreshCB`, permettono di selezionare *la checkbox senza che ciò causi l’invio del messaggio al nodo*.

Specification Space

E' stato risolto il problema relativo al funzionamento della finestra "Specification space" che causava il blocco dell'Inspector.

Il malfunzionamento era legato al fatto che le richieste relative alle funzioni messe a disposizione dalla finestra venivano eseguite direttamente dall'event dispatcher thread.

Con l'attivazione della modalità step, l'event dispatcher thread non riceveva più risposta dal nodo (che giustamente veniva sospeso all'interno del monitor) e questo causava il blocco dell'intero programma.

Per risolvere la questione, viene creato un ulteriore thread con il compito di eseguire in maniera asincrona le varie funzioni attivabili dallo specification space, delegando all'event dispatcher thread il solo compito di aggiornare l'interfaccia grafica una volta che queste risultano completate.

Inspecting as an Agent / Tuple Centre

Infine, è stato risolto il problema del non tempestivo aggiornamento delle finestre dell'Inspector. Questo era dovuto al fatto che la notifica del cambio di stato della macchina ReSpecT ai vari Inspector non avveniva effettivamente ad ogni cambio di stato ma soltanto quando questa tornava in IDLE.

Il problema è stato rilevato grazie all'attivazione della modalità step (solo all'interno di essa, infatti, è possibile sperimentare questa differenza).

In realtà l'Inspector aggiornandosi ad ogni stato IDLE, "vede" lo spazio di tuple come lo vedrebbe in effetti un agente, mentre nel caso che ciò avvenga ad ogni stato è come se avesse gli occhi del centro di tuple stesso.

Ciò ha aperto l'opportunità all'Inspector di rendere configurabile questi due diversi modi di venire notificato.

A differenza di quanto è avvenuto per lo stepMode, la realizzazione della nuova feature può essere effettuata attraverso l'estensione dell'InspectorProtocol, in quanto si tratta di una caratteristica settabile per una specifica istanza dell'Inspector e nel fare ciò non vengono influenzate tutte le altre.

Lato Inspector, perciò, viene definito di default una nuova opzione all'interno dell'InspectorProtocol che consente di ispezionare il centro di tuple desiderato con gli occhi del centro di tuple stesso.

Sono stati poi inseriti, quindi, due checkbox all'interno del pannello stepMode per poter cambiare a piacimento la modalità di ispezione.

Lato service, a seconda della relativa situazione (IDLE o un qualsiasi altro stato), viene generato un evento di tipo TYPE_IDLESTATE oppure TYPE_NEWSTATE.

L'effettiva notifica degli eventi (che vengono comunque generati sempre entrambi) agli Inspectors avverrà soltanto in seguito alla verifica dei singoli InspectorProtocol (vedi classe InspectorContextSkel, metodo onInspectableEvent()).