

Sistemi Distribuiti 2 2003-2004

Esercitazione Sistemi Distribuiti 2

(Consegna: Mercoledì 15 Settembre 2004)

L'esercitazione prevede l'implementazione di una libreria per la gestione di gruppi di oggetti aperti e non deterministici.

Rispetto ai modelli presentati a lezione, l'esercitazione richiesta prevede le seguenti modifiche:

- il gruppo di processi coincide con il gruppo di oggetti, ogni oggetto sarà un processo con un proprio main;
- i messaggi sono chiamate di metodo, tutti gli oggetti appartenenti al gruppo ufficialmente rispondono ad una interfaccia comune, non necessariamente implementano tutti i metodi;
- il non determinismo è realizzato dal processo gestore, eventuali eccezioni propagate e/o sollevate dagli oggetti remoti, quando il metodo non è stato implementato, devono essere intercettate e mascherate;
- il gruppo è dinamico, quindi gli oggetti possono iscriversi e disiscriversi dinamicamente (tramite un'API);
- il naming è riferito al processo gestore che si occupa anche di filtrare le risposte degli oggetti su una base FCFS.

Traccia dell'Architettura del Sistema

Processo Gestore

Il processo gestore è a tutti gli effetti un oggetto remoto, come tale si registrerà presso l'`rmiregistry`. Il suo identificativo (la stringa con cui lo registrate) sarà l'identificativo dell'intero gruppo e permetterà di riferirsi all'intero gruppo. Nel nostro caso il ruolo del processo gestore è piuttosto attivo:

- terrà traccia di quali oggetti sono linkati al gruppo;
- provvederà a smistare il messaggio a tutti i membri del gruppo (le richieste devono essere fatte in parallelo o quasi pertanto dovete staccare un thread per ogni oggetto membro);

- provvederà a filtrare i risultati provenienti dagli oggetti appartenenti al gruppo, solo il primo risultato verrà fornito al richiedente gli altri, se ce ne sono, verranno scartati.

Siccome di gruppi, teoricamente, ne possiamo avere diversi e variegati l'idea è quella di avere una base comune da cui ereditano tutti i possibili gruppi. Ogni nuovo gruppo implementerà un'interfaccia di gruppo propria alla quale gli oggetti membri del gruppo "ufficialmente" dovranno essere in grado di rispondere.

API

Come detto il processo gestore si dovrà anche occupare di iscrivere e discrivere gli oggetti al gruppo che rappresenta, una possibile API per fare ciò potrebbe essere:

- `link(String s)` dove `s` è l'identificatore dell'oggetto che si iscrive al gruppo;
- `unlink(String s)` dove `s` è l'identificatore dell'oggetto che vuole uscire dal gruppo.

Tenete presente che gli oggetti che si iscrivono/discrivono sono a tutti gli effetti degli oggetti remoti, per la precisione dei server e come tali li dovrà trattare il processo gestore (es., deve ottenere un referente per poter smistare i messaggi, gli oggetti devono registrarsi presso un `rmiregistry`).

Interfacce di Gruppo

Ogniqualvolta che si definisce un nuovo gruppo di oggetti anche una nuova interfaccia deve essere definita. Tale interfaccia, detta *interfaccia di gruppo*, definisce l'insieme dei metodi a cui il gruppo risponde. L'interfaccia di gruppo viene implementata dal processo gestore. Non tutti gli oggetti membri del gruppo implementano tutti i metodi, sarà compito del processo gestore mascherare eventuali eccezioni sollevate (suggerimento usare la reflection ed il metodo `invoke`).