

UNIVERSITÀ DEGLI STUDI DELLA CALABRIA
FACOLTÀ DI INGEGNERIA
CORSO DI LAUREA SPECIALISTICA IN INGEGNERIA INFORMATICA

Esame di SISTEMI DISTRIBUITI

APPELLO DEL 7 GENNAIO 2004 –TEMPO A DISPOSIZIONE: 2 ORE

Si consideri un sistema di calcolo distribuito composto da N nodi, in cui le connessioni tra i nodi sono descritte mediante un grafo non orientato. Su ogni nodo è presente un server, generico e multithreaded, che riceve particolari oggetti (detti *job*) ed esegue metodi contenuti all'interno di essi. Ogni server ha una potenza di calcolo finita, ed è accessibile solo tramite un processo locale detto *controller*. Un processo che necessita dell'esecuzione di un job, pertanto, effettua una richiesta al controller presente sul proprio nodo, il quale decide se eseguire il job localmente oppure passarlo ad uno dei nodi vicini (ovviamente interagendo con i controller presenti su di essi).

Le richieste di esecuzione di job sono descritte mediante quadruple (*node*, *job*, *req*, *timestamp*), dove *node* è il nodo su cui risiede il processo richiedente, *req* indica la potenza di calcolo necessaria al job e *timestamp* è l'istante di tempo in cui la richiesta è stata costruita. Per ogni richiesta ricevuta (sia da processi locali che da altri controller), il controller confronta la potenza di calcolo necessaria al job con quella attualmente a disposizione. Qualora non ritenga possibile (o conveniente) soddisfare la richiesta localmente, il controller passa la richiesta ad un controller su un nodo vicino. Nel caso in cui sia trascorso più di un certo lasso di tempo dalla costruzione della richiesta, viene invece notificato un timeout al controller presente sul nodo *node*, che si occupa di inoltrare tale informazione al processo che aveva generato la richiesta iniziale.

Si progetti il sistema descritto e lo si implementi utilizzando i meccanismi RMI di Java e definendo $N = 20$ e *req* compresa tra 1 e 100, dove per 100 si intende la potenza di calcolo massima di un nodo. Si definisca, nei controller, una politica di accettazione/redistribuzione dei job che favorisca il bilanciamento del carico sui nodi e che tenga conto del carico computazionale sui nodi vicini.