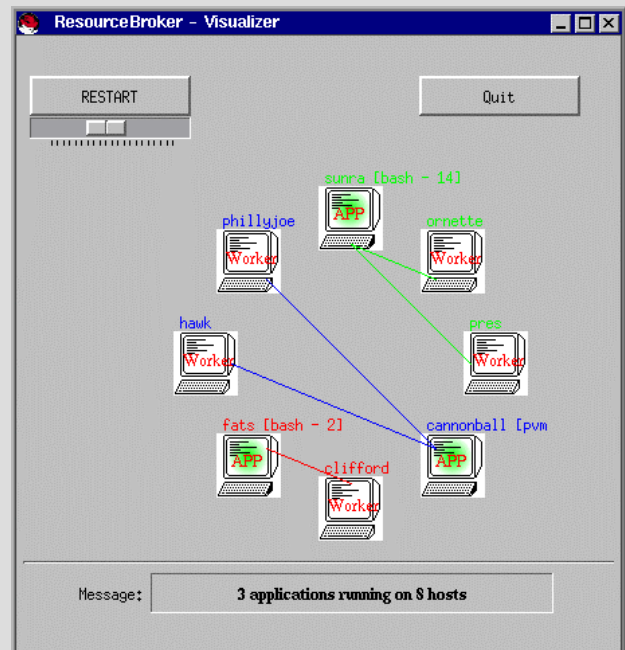
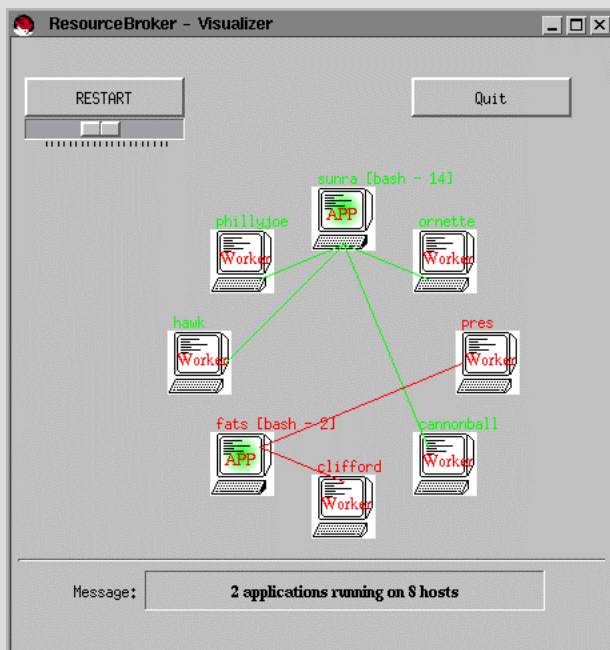
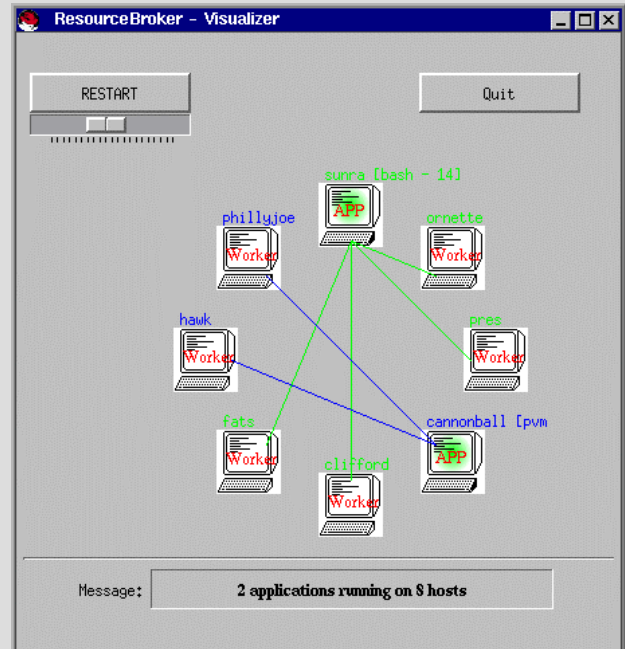
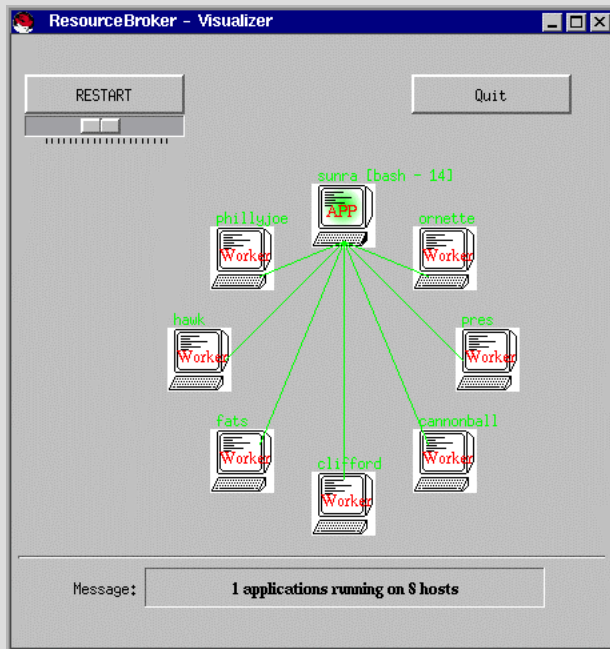


ResourceBroker

A resource manager for distributed platforms



These four screen snapshots, arranged clockwise from the top left corner, show a trace of the system's behavior. (1) A scalable Calypso computation, depicted in green, is running on all eight available machines. (2) A PVM computation, depicted in blue, asks for, and is given three of the eight machines. The Calypso computation scaled down to five machines. (3) An MPI computation, depicted in red, asks for three machines, but is given only two. The Calypso computation scaled down to three machines. (4) The PVM computation completed. The newly available three machines are redistributed. The MPI computation scales up to three machines it originally wanted and the Calypso computation scales up to five machines.

Goals

ResourceBroker is a resource management system for monitoring computing resources in a distributed platform and for dynamically assigning them to concurrently executing com-

putations. It is, at present, partially designed and implemented. Although applicable to a wide variety of computations, including sequential ones, it especially benefits adaptive parallel computations.

A computation is *adaptive* if it exhibits at least one of these two properties: (1) it can statically (at start time) and/or dynamically (during the execution) *ask* for resources satisfying certain characteristics and incorporate such resources when they are given to it; (2) it can continue executing if some resources are taken away from it.

Functionality

ResourceBroker can dynamically select and assign resources to computations even if those computations were not developed to work with external resource managers. It simultaneously supports computations written for several unmodified parallel programming systems, including PVM, MPI, and Calypso. ResourceBroker can be viewed as supporting arbitration among concurrently running virtual machines, allocating, deallocating, and reallocating resources among the machines, and growing and shrinking the virtual machines as needed.

ResourceBroker is a service, optionally usable at each invocation of each computation. It runs with user-level privilege, never compromising the security of the platform.

The system consists of two weakly coupled layers. The *resource broker layer* is platform-wide. When fully developed, it will be able to bind resources to virtual machines based on the QoS requirements of the individual computations, with the resource needs specifiable in a language based on that used in Globus.

The *agent layer* consists of sets of *agent pro-*

cesses, each associated with a computation. A user who wants to use ResourceBroker submits the computation through an agent process. This agent process communicates with a set of *subagent processes* on other machines participating in the computation.

An agent facilitates the interaction between the resource manager and a computation and actively intervenes in the execution of the computation, as directed by the resource manager.

Bibliography (available from MILAN web sites at NYU and/or ASU)

A. Baratloo, A. Itzkovitz, Z. Kedem, and Y. Zhao. Just-in-time resource management in distributed systems.

**MILAN is a joint project of
New York University
Arizona State University**

Zvi M. Kedem
New York University
+1 212 998 3101 (phone)
+1 212 477 3265 (fax)
kedem@cs.nyu.edu
<http://www.cs.nyu.edu/milan>

Partha Dasgupta
Arizona State University
+1 602 965 5583 (phone)
+1 602 965 2751 (fax)
partha@asu.edu
<http://milan.eas.asu.edu>

