

Home

ActiveCluster is a framework for writing cluster-based applications, monitoring the available nodes in a cluster and communicating with them. ActiveCluster consists of an API together with a default implementation.

The API is simple and small which reuses a few of the standard JMS APIs for Message, Destination, MessageListener and adds a few simple interfaces for working with clusters representing nodes in the cluster, replicated node state and listeners to nodes joining, leaving or updating their state.

Features

The current features include

- viewing the current nodes in the cluster and their current state
- listening to nodes arrive, update their state, shut down or fail
- communicate with the cluster as a whole or with any specific node in the cluster
- elect a leader of the cluster to make decisions on behalf of the cluster

Implementation

It is easy to implement the ActiveCluster API on top of any networking library such as raw sockets, JMS, JGroups or Jabber. The default implementation uses JMS though we should be easily be able to add other implementations further down the line, like a JGroups implementation etc

Use cases

ActiveCluster can be used to build a variety of different network protocols like...

- buddy-groups for Web based or EJB based session replication
- master / slave protocols for High Availability (HA) protocols with hot-standby
- controller-voting protocols to elect a single node in a cluster to be the controller along with defining who will be the next node if the controller falls over as well as detecting the loss of the controller
- distributed lock manager

Related projects

Currently ActiveCluster is used by

- [ActiveMQ](#) to perform its cluster management features
- [WADI](#) to help organise buddy groups for HTTP/EJB session replication

News

[James Strachan](#) posted on Feb 14, 2005

[ActiveCluster 1.0 Released](#)

I'm pleased to announce the 1.0 release of [ActiveCluster](#), the open source cluster management & control framework.

This release completes the core API and reference implementation which includes the following features

- viewing the current nodes in the cluster and their state

- listening to nodes arrive, update their state, shut down or fail
- communicate with the cluster as a whole or with any specific node in the cluster
- elect a leader of the cluster to make decisions on behalf of the cluster

Future versions will include more complex higher level protocols like buddy-group arrangement protocols and HA features.

To download the binary or source versions of this release, please view the [download page](#)

[More News](#)