Clustering

Permalink to this page:

Preface

This page discusses cluster- and clustering-related questions. Please make sure to read the Clustering HowTo page in the main Tomcat documentation bundle as well.

Questions

Answers

Can I configure a cluster at the Engine level?

Yes, beginning with Tomcat 5.5.10 you can configure clusters at both the Engine and Host levels. This helps support clustering for web hosting companies.

Show me a simple cluster configuration example.

For Tomcat 5.5.10 and later:

```
<Cluster className="org.apache.catalina.cluster.tcp.SimpleTcpCluster" defaultMode="true" />
```

How do I turn on transport logging?

(**FIXME**: The text below needs an update, The logging categories are `org.apache.catalina.ha` and `org.apache.catalina.tribes` nowadays. What else is needed? There no "clusterLog" attribute.)

* Use "org.apache.catalina.cluster" as logger category and switch to info, debug or trace as log level.
* Configure the clusterLog attribute (logging category) to get and send and receive message log.

How do I use JMX to monitor the cluster?

Since Java 5 you can use the jconsole application to look inside the runnnig cluster: please see the JMX configuration section in the Clustering HowTo document. In fastasyncmode replication mode you can get more information with sender attributes doProcessingStats="true" and queueDoStats="true". Finally, with the new JMX remote Ant task you can access the state and call operations.

Can I pause the message sending?

Yes, the async senders buffer the messages, but make sure the membership ping is active. With fastasyncqueue mode you can limit the max queue size.

Can I add more senders (pooled mode)?

Yes, with sender attribute maxPoolSocketLimit="40" you can have more than the default 25 sockets to transfer more parallel messages.

What happens when I pull the network cable?

The other members will remove the instance from the cluster, but when you insert the cable again, the Tomcat instance might have completely flipped out. This is because the OS might start using 100% of the CPU when a multicast message is sent. There has not yet been a good solution for this, I will let you know when I have come up with one. (pero: I test this and I works correct with java 5 and exists when you use the cluster with JDK 1.4.x)

On my windows laptop without network my cluster doesn't work.

The Membership attribute mcastBindAddress="127.0.0.1" must be set!

The cluster doesn't work under Linux with two nodes on two boxes.

Check the the following:

- Is your network interface enabled for multicast? ifconfig eth0 MULTICAST
- Exists a multicast route to your network interface? route add -host 228.0.0.4 dev eth0
- Is your firewall active? Then check that multicast port is on your UDP open list and the receiver TCP port is also for both machines open!

I get "localhost" rather than "eth0" or another interface when using tcpListenAddress="auto".

Change /etc/hosts so that the localhost domain resolves to the actual IP address of the NIC, eth0. Please see Bugzilla for more.