Agenda

- About me
- Tomcat Clustering Overview
- Session Replication
- Cluster Channel Component
- Monitoring Cluster components
About me

• Keiichi Fujino
• I live in Japan
• Software Enginner since 2002
• Apache Tomcat committer since 2010

• kfujino@apache.org
Clustering Overview

What is Cluster?
  • Performance improvement
  • High availability

Tomcat Clustering
  • Cluster membership/Grouping
  • Session Replication

Load balancing is not a Tomcat features
  • Use mod_jk / mod_proxy_balancer
Cluster Architecture

Cluster

Channel
- ChannelListener
- MembershipListener
- ChannelInterceptor
- MembershipService
- ChannelSender
- ChannelReceiver

ClusterManager (template)
- Valve(ClusterValve)
- ClusterListener
- ClusterDeployer

Create from template

Manager

Engine

Host

Context

APACHECON North America
MAY 16-18, 2017 MIAMI, FL
Cluster Architecture

- Cluster
  - The main component of Tomcat Cluster

- Cluster Manager
  - The session manager for the session replication

- Valve (Cluster Valve)
  - The same as usual Valve
  - Added to the request processing pipeline automatically
Cluster Architecture

- Cluster Deployer
  - Sharing of WAR files among cluster nodes

- ClusterListener
  - Listen cluster messages and events

- Channel
  - Performs messaging and grouping among the cluster nodes
Session Replication
Session Replication

Implementations
• All-to-All session replication
• DeltaManager (Default)

• Primary-Secondary session replication
  • BackupManager

Diagram:
- Tomcat1 → Tomcat2
- Tomcat1 → Tomcat3
- Tomcat3 → Tomcat4
- Replicate to all nodes
- Replicate to backup node
Use constraints

• Make sure that your web.xml describe the <distributable/> element

• Session attributes must implement java.io.Serializable

• sticky session
  • If you use the BackupManager, This is required
How to configure

- Define `<Cluster>` element inside The Engine or Host element.

- Configure Cluster Manager
  - DeltaManager or BackupManager

- Configure Channel components

- Enable `org.apache.catalina.ha.tcp.ReplicationValve`

- Enable `org.apache.catalina.ha.session.ClusterSessionListener`
  - If DeltaManager use
Delta Replication

• Replicate only the changes of session
  • Not all session data

• Replicate all changes of session at the time of end of request
  • Not replication per change of session
**Register Delta Info**

- Add `ATTRIBUTE(Attr_A, Value_A)`
- Add `ATTRIBUTE(Attr_B, Value_B)`
- Remove `ATTRIBUTE(Attr_A)`
- Add `ATTRIBUTE(Attr_B, Value_BB)`

Default: `recordAllActions=false`

<table>
<thead>
<tr>
<th>TYPE</th>
<th>ACTION</th>
<th>NAME</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATTRIBUTE</td>
<td>SET</td>
<td>Attr_A</td>
<td>Value_A</td>
</tr>
<tr>
<td>ATTRIBUTE</td>
<td>SET</td>
<td>Attr_B</td>
<td>Value_B</td>
</tr>
<tr>
<td>ATTRIBUTE</td>
<td>REMOVE</td>
<td>Attr_A</td>
<td>null</td>
</tr>
<tr>
<td>ATTRIBUTE</td>
<td>SET</td>
<td>Attr_B</td>
<td>Value_BB</td>
</tr>
</tbody>
</table>

record `recordAllActions=true`

<table>
<thead>
<tr>
<th>TYPE</th>
<th>ACTION</th>
<th>NAME</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATTRIBUTE</td>
<td>SET</td>
<td>Attr_A</td>
<td>Value_A</td>
</tr>
<tr>
<td>ATTRIBUTE</td>
<td>SET</td>
<td>Attr_B</td>
<td>Value_B</td>
</tr>
<tr>
<td>ATTRIBUTE</td>
<td>REMOVE</td>
<td>Attr_A</td>
<td>null</td>
</tr>
<tr>
<td>ATTRIBUTE</td>
<td>SET</td>
<td>Attr_B</td>
<td>Value_BB</td>
</tr>
</tbody>
</table>
DeltaManager

- All-to-All session replication

- Default Session Manager in Cluster environment

- For small cluster
Architecture

Tomcat1

- DeltaManager
  - create/remove
- ReplicationValve
  - Delta replication

Cluster channel

Replication

Tomcat2

Cluster channel

ClusterSessionListener

Receive replication Message

DeltaManager
DeltaManager

Behavior

- Tomcat1
  - Session (Primary)
- Tomcat3
  - Session (Non-Primary)
- Tomcat2
  - Session (Non-Primary)
- Tomcat4
  - Session (Non-Primary)

Replicate create/delta/remove
Node Failure

- Tomcat1
  - Node Failure
- Tomcat2
  - Session (Non-Primary)
- Tomcat3
  - Session (Non-Primary)
- Tomcat4
  - Session (Non-Primary)
DeltaManager

Node Recovery

Tomcat1
- No Session Data
- Session(Non-Pri)
- Session(Non-Pri)

Get All session Message

Tomcat2
- Session(Pri)
- Session(Non-Pri)

All Session Data Message

Complete Message

Tomcat3
- Session(Non-Pri)
- Session(Pri)

Tomcat4
- Session(Non-Pri)
- Session(Non-Pri)

All Session Data Message
BackupManager

• Primary-Secondary session replication

• For large cluster
Architecture

Tomcat1

ReplicationValve

↓ Delta replication

BackupManager

↓ create/remove/delta

ReplicatedMap

channel

Tomcat2

BackupManager

Replication

ReplicatedMap

channel
BackupManager

Behavior of Create

- **Create session**
  - Tomcat1: Session (Primary)
  - Replicate session data
    - Tomcat2: Session (Backup)
  - Replicate sessionId
    - Tomcat3: Session (Proxy)
    - Tomcat4: Session (Proxy)
BackupManager

Behavior of Create

- Tomcat1:
  - Session (Primary)
  - Session (Proxy)

- Tomcat2:
  - Session (Backup)
  - Session (Proxy)

- Tomcat3:
  - Session (Proxy)
  - Session (Primary)

- Tomcat4:
  - Session (Proxy)
  - Session (Backup)

Create session

Replicate sessionId

Replicate session data
Behavior of Delta

Tomcat1
Session (Primary)

Replicate delta

Tomcat2
Session (Backup)

Tomcat3
Session (Proxy)

Tomcat4
Session (Proxy)
BackupManager

Behavior of Remove

- **Tomcat1**: Session (Primary)
  - Remove session

- **Tomcat2**: Session (Backup)

- **Tomcat3**: Session (Proxy)
  - Remove session

- **Tomcat4**: Session (Proxy)
BackupManager

Node Failure

Tomcat1
- SessionA (Primary)
- SessionB (Backup)
- SessionC (Proxy)

Tomcat2
- Promote & Select new backup & publish
  - SessionA (Backup)
  - SessionB (Primary)
  - SessionC (Proxy)
  - NOP

Tomcat3
- SessionA (Proxy) → newly backup → SessionA (Backup)
- SessionB (Proxy) → newly proxy → SessionB (Proxy)
- SessionC (Primary) → NOP

Tomcat4
- SessionA (Proxy) → newly proxy → SessionA (Proxy)
- SessionB (Proxy) → newly backup → SessionB (Backup)
- SessionC (Backup) → NOP
BackupManager

Node Recovery

Tomcat1

- No Session Data
- SessionA(Proxy)
- SessionB(Proxy)

Send/Receive RPC Message

Tomcat2

- SessionA(Primary)
- SessionB(Proxy)

Send/Receive RPC Message

Tomcat3

- SessionA(Backup)
- SessionB(Primary)

Send/Receive RPC Message

Tomcat4

- SessionA(Proxy)
- SessionB(Backup)

Send/Receive RPC Message

- INIT Message
- STATE Message
- START Message
What is Channel?
- Messaging & Grouping component

Responsibility
- Build Membership
- Send channel messages
- Receive channel messages
Channel Components

Channel

- MembershipService
- ChannelSender
- ChannelReceiver
- ChannelListener
- MembershipListener
- ChannelInterceptor
- ChannelCoordinator
Channel Components

- **Channel**
  - The main component of channel
  - `org.apache.catalina.tribes.group.GroupChannel` only

- **MembershipService**
  - The component which build a cluster group
  - Start a multicast receiver thread and a multicast sender thread
Channel Components

- **ChannelSender**
  - Send channel messages to other nodes
  - Sender Queue size is specified in poolSize attribute

- **ChannelReceiver**
  - Receive channel messages from other nodes
  - Tuning of the maxThreads attribute depends on send option of channel message
    - Synchronization mode
      - Need to align maxThreads with poolSize
    - Asynchronous mode
      - do not need to align the maxThreads with poolSize.
Channel Components

- ChannelListener
  - Listen received channel messages

- MembershipListener
  - Listen add/remove of cluster members
Channel Components

• ChannelInterceptor
  • Intercept a channel message and a member detection
  • There are many configurable implementations

• ChannelCoordinator
  • Special ChannelInterceptor
  • Coordinates the ChannelInterceptors
Sample config

```xml
<Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster">
    
    <Channel className="org.apache.catalina.tribes.group.GroupChannel">
        <Membership className="org.apache.catalina.tribes.membership.McastService"
                        address="229.0.0.61" port="45564" frequency="500" dropTime="4000" />

        <Receiver className="org.apache.catalina.tribes.transport.nio.NioReceiver"
                  address="auto" port="4004" autoBind="100" maxThreads="25"/>

        <Sender className="org.apache.catalina.tribes.transport.ReplicationTransmitter">
            <Transport className="org.apache.catalina.tribes.transport.nio.PooledParallelSender"
                        timeout="5000" poolSize="25"/>
        </Sender>

        <Interceptor className="org.apache.catalina.tribes.group.interceptors.MessageDispatch15Interceptor" />
        <Interceptor className="org.apache.catalina.tribes.group.interceptors.TcpFailureDetector"/>
    </Channel>

    
</Cluster>
```
TCPFailureDetector

- **Main Features**
  - Intercept memberDisappeared events.
  - Member check when send errors.

- **Other Features**
  - Manage membership in static membership
**ChannelInterceptor**

**MessageDispatchInterceptor**

- Asynchronous send message

- Use Thread Pooling
  - maxThreads
  - maxSpareThreads

- You must set the send options asynchronous.
  - Cluster’s channelSendOptions to asyn mode(8)
  - BackupManager’s mapSendOptions to asyn mode(8)
StaticMembershipInterceptor

- The static membership instead of multicast

- Make sure
  - Disable multicast membership
    - Cluster’s channelStartOptions attribute = 3
  - Enable TcpPingInterceptor for nodes failure detection
  - Enable TcpFailureDetector for membership management
  - The order is
    - TcpPingInterceptor
    - TcpFailureDetector
    - StaticMembershipInterceptor
<Interceptor className="org.apache.catalina.tribes.group.interceptors.TcpPingInterceptor"/>
<Interceptor className="org.apache.catalina.tribes.group.interceptors.TcpFailureDetector"/>
<Interceptor className="org.apache.catalina.tribes.group.interceptors.StaticMembershipInterceptor">
  <LocalMember className="org.apache.catalina.tribes.membership.StaticMember"
    uniqueId="{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,1}"/>
  <Member className="org.apache.catalina.tribes.membership.StaticMember"
    port="4010" host="hostA"
    uniqueId="{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,2}" />
  <Member className="org.apache.catalina.tribes.membership.StaticMember"
    port="4010" host="hostB"
    uniqueId="{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,3}" />
</Interceptor>
ChannelInterceptor

ThroughputInterceptor
• Measuring the send and receive of channel messages

DomainFilterInterceptor
• Filter the members that join cluster group by the domain
Monitoring Cluster
How to monitor Tomcat cluster.

- Logging
- JMX
Monitoring your Cluster with Logging

- To track channel messages for Debugging
  - Use the Tomcat JULI
  - Key: org.apache.catalina.tribes.MESSAGES
  - Level: FINEST

```java
# FOR DEBUG
10catalina.org.apache.juli.AsyncFileHandler.level = FINEST
10catalina.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
10catalina.org.apache.juli.AsyncFileHandler.prefix = MESSAGES.
10catalina.org.apache.juli.AsyncFileHandler.bufferSize = -1
org.apache.catalina.tribes.MESSAGES.level = FINEST
org.apache.catalina.tribes.MESSAGES.handlers = 10catalina.org.apache.juli.AsyncFileHandler
```
Monitoring your Cluster with Logging

- ThroughputInterceptor
  - Report the throughput statistics.
  - Default interval is to report every 10000 messages.
  - Key: org.apache.catalina.tribes.group.interceptors.ThroughputInterceptor
  - Level: INFO

```java
org.apache.catalina.tribes.group.interceptors.ThroughputInterceptor.report
ThroughputInterceptor Report[
  Tx Msg:60,024 messages
  Sent:53.30 MB (total)
  Sent:53.31 MB (application)
  Time:16.84 seconds
  Tx Speed:3.16 MB/sec (total)
  TxSpeed:3.16 MB/sec (application)
  Error Msg:0
  Rx Msg:60,048 messages
  Rx Speed:0.23 MB/sec (since 1st msg)
  Received:53.28 MB]
```
Monitoring Cluster

Monitoring your Cluster with JMX

• Catalina Domain
  • Cluster Mbeans
    • Cluster Mbean
    • Deployer Mbean
    • Member MBeans
  • (Cluster )Manager Mbean
  • (Cluster )Valve MBean
Monitoring Cluster

Cluster Mbean

• Cluster settings
Monitoring Cluster

Member Mbeans

- All Cluster members that have been joining same cluster group
Monitoring Cluster

(Cluster )Manager Mbean

• Session Information
Monitoring Cluster

(Cluster)Valve MBean

• ReplicationValve settings
Monitoring Cluster

Monitoring your Cluster with JMX

- ClusterChannel Domain
  - Channel Mbeans
    - Channel Mbean
    - Interceptor Mbeans
    - Membership Mbean
    - Receiver Mbean
    - Sender MBean
Monitoring Cluster

Channel Mbean
- Channel settings
Monitoring Cluster

Membership Mbean

• Membership Settings

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>128.0.8.61</td>
</tr>
<tr>
<td>Bind</td>
<td></td>
</tr>
<tr>
<td>Domain</td>
<td>byte[]</td>
</tr>
<tr>
<td>DomainName</td>
<td>[1,2,3,4]</td>
</tr>
<tr>
<td>DropTime</td>
<td>4000</td>
</tr>
<tr>
<td>Frequency</td>
<td>500</td>
</tr>
<tr>
<td>LocalLoopbackDisabled</td>
<td>false</td>
</tr>
<tr>
<td>LocalMemberName</td>
<td>org.apache.catalina.tribes.membership.MemberImpl[<a href="http://129.129.129.129">http://129.129.129.129</a>]</td>
</tr>
<tr>
<td>MembersBName</td>
<td>java.lang.String[]</td>
</tr>
<tr>
<td>Port</td>
<td>45654</td>
</tr>
<tr>
<td>Properties</td>
<td>[topListenHost=129.56.240.82, udpListenPort=1, ...]</td>
</tr>
<tr>
<td>RecoveryCounter</td>
<td>-1</td>
</tr>
<tr>
<td>RecoveryEnabled</td>
<td>false</td>
</tr>
<tr>
<td>RecoveryRooted</td>
<td>false</td>
</tr>
<tr>
<td>SoTimeout</td>
<td>0</td>
</tr>
<tr>
<td>Tl</td>
<td>0</td>
</tr>
</tbody>
</table>
Monitoring Cluster

Membership Mbean

- Membership Operations
Monitoring Cluster

Sender MBean

- Settings and Stats info
Monitoring Cluster

Receiver Mbean

• Settings and Stats

<table>
<thead>
<tr>
<th>Attribute values</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>ActiveCount</td>
<td>0</td>
</tr>
<tr>
<td>Address</td>
<td>129.60.246.02</td>
</tr>
<tr>
<td>AutoBind</td>
<td>100</td>
</tr>
<tr>
<td>CompletedTaskCount</td>
<td>1510</td>
</tr>
<tr>
<td>Direct</td>
<td>false</td>
</tr>
<tr>
<td>Listening</td>
<td>true</td>
</tr>
<tr>
<td>MaxIdleTime</td>
<td>30000</td>
</tr>
<tr>
<td>MaxThreads</td>
<td>25</td>
</tr>
<tr>
<td>MinThreads</td>
<td>25</td>
</tr>
<tr>
<td>OnAllInLine</td>
<td>true</td>
</tr>
<tr>
<td>PoolSize</td>
<td>25</td>
</tr>
<tr>
<td>Port</td>
<td>40004</td>
</tr>
<tr>
<td>RxBufSize</td>
<td>43800</td>
</tr>
<tr>
<td>SecurePort</td>
<td>1</td>
</tr>
<tr>
<td>SelectorTimeout</td>
<td>5000</td>
</tr>
<tr>
<td>SoKeepAlive</td>
<td>false</td>
</tr>
<tr>
<td>SoLingerOn</td>
<td>true</td>
</tr>
<tr>
<td>SoLingerTimeout</td>
<td>2</td>
</tr>
<tr>
<td>SoReuseAddress</td>
<td>true</td>
</tr>
<tr>
<td>TaskCount</td>
<td>1510</td>
</tr>
<tr>
<td>TopNoDelay</td>
<td>true</td>
</tr>
<tr>
<td>Timeout</td>
<td>30000</td>
</tr>
<tr>
<td>TxBufSize</td>
<td>25188</td>
</tr>
<tr>
<td>UdpPort</td>
<td>1</td>
</tr>
<tr>
<td>UdpRxBufSize</td>
<td>43800</td>
</tr>
<tr>
<td>UdpTxBufSize</td>
<td>25188</td>
</tr>
<tr>
<td>UseBufferPool</td>
<td>true</td>
</tr>
</tbody>
</table>
Interceptor Mbeans

- All interceptors that are used in channel
- Implements MBeans of all commonly used Interceptors but it does not implement all Interceptor MBeans.
Monitoring Cluster

ThroughputInterceptor

- The throughput statistics
Monitoring Cluster

TcpFailureDetector

- Settings and Member check by TCP
Monitoring Cluster

MessageDispatchInterceptor

- Settings and Stats info
It's important to note that

- Channel Mbeans are supported in 9.0.0.M20 and later
- This feature has not back-ported into Tomcat8.5 yet.
- The ReplicationMap MBean is T.B.D
Questions?
Thank You
Tomcat Cluster

Keiichi Fujino