Native Web Server Integration

Mladen Turk
Developer & Consultant
JBoss Inc.

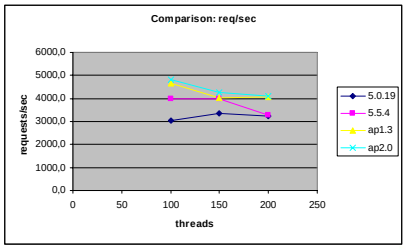
March 8, 2005

Agenda

- Native Web Server Integration
- Getting the most out of Integration
- Integration using mod_jk
- Integration using mod_proxy
- Future and AJP14
- Q&A

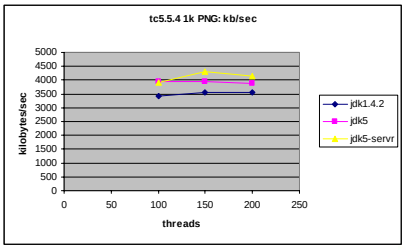
Native Web Server Integration

- Performance is no more an issue



Native Web Server Integration

- Modern JVMs offer higher performance

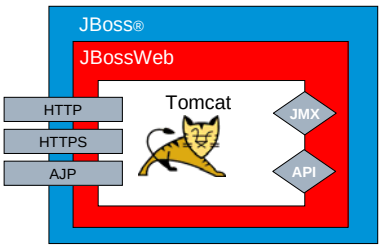


Native Web Server Integration

- Why use a Native Web Server in front of Tomcat?
  - ✓ Scalability
  - ✓ Load Balancing
  - ✓ Existing legacy applications
    - Perl
    - PHP
  - ✓ Custom Security
    - LDAP
    - NTLM

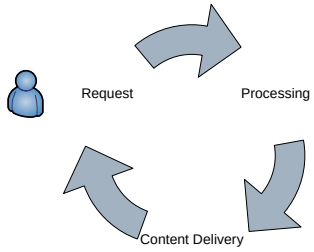
Native Web Server Integration

- JBossWeb Architecture



Native Web Server Integration

- Tomact request processing
  - ✓ Each client connection consumes one or more threads

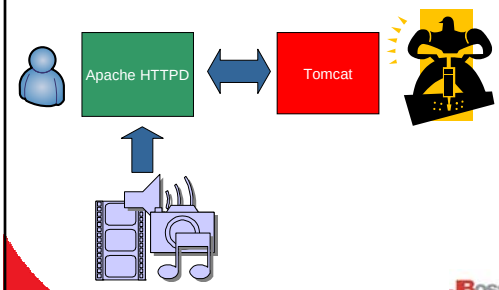


Native Web Server Integration

- Context switch latency
  - ✓ Limits the number of concurrent connections
  - ✓ Limit 200 per CPU – (empirical)

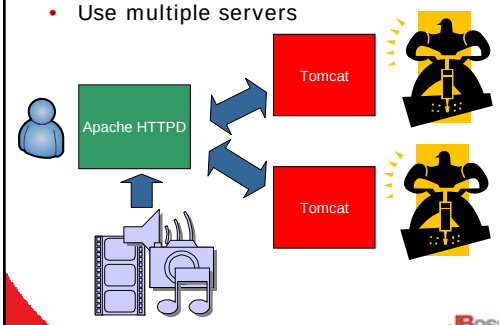
Native Web Server Integration

- Use native web server for static content



Native Web Server Integration

- Use multiple servers



Getting the most out of Integration

- Do you need that?
  - ✓ Additional configuration
  - ✓ Additional maintenance
  - ✓ Need to build the connector on your own
  - ✓ Additional point of failure
- Define your needs
  - ✓ Application response time
  - ✓ Number of concurrent users
  - ✓ Network throughput

Getting the most out of Integration

- Define your topology
  - ✓ Application servers
  - ✓ Web servers
  - ✓ Static content delivery
  - ✓ Clustering
  - ✓ Sessions
  - ✓ Caching
  - ✓ Failover
  - ✓ Load balancing

Getting the most out of Integration

- Separate content
  - ✓ Do not mix application and native web server content
- Separate applications
  - ✓ Do not mount /*
  - ✓ Do not mount /*.jsp
- Use native SSL
  - ✓ Much faster then JSSE
  - ✓ Can use hardware accelerators

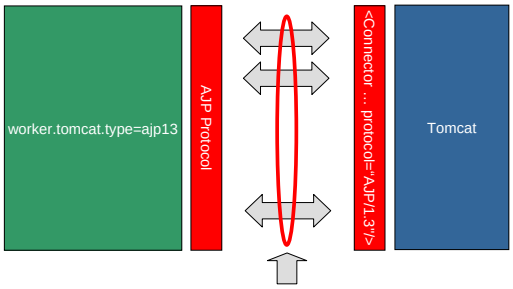
Integration using mod_jk

- Apache Jakarta Project
  http://jakarta.apache.org/tomcat/connectors-doc/
- Fully supported Web servers
  - ✓ Apache 1.33+
  - ✓ Apache 2.0.52+
  - ✓ Apache 2.1/2.2 (*)
  - ✓ IIS 5/5.1/6.0
- Supported Web servers
  - ✓ iPlanet (SunOne)
  - ✓ Lotus Domino

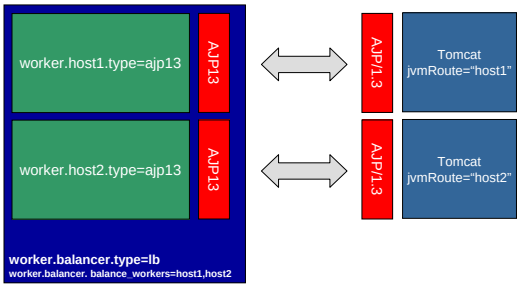
JK Architecture

- Uses AJP 1.3 protocol
  - ✓ Constant connection pool
  - ✓ No need to parse HTTP protocol
- Integrated load balancer
  - ✓ Sticky sessions
  - ✓ Failover
  - ✓ Domain model clustering

JK Architecture



JK Load balancer Architecture



JK Architecture

- Understand configuration

httpd.conf

```
<IfModule worker.c>
  StartServers        2
  MaxClients        150
  MinSpareThreads    25
  MaxSpareThreads    75
  ThreadsPerChild    25
  MaxRequestsPerChild  0
</IfModule>
```

workers.properties

```
worker.host1.type=ajp13
worker.host1.port=8009
worker.host1.host=109.101.102.103
worker.host1.cache_size=25
```

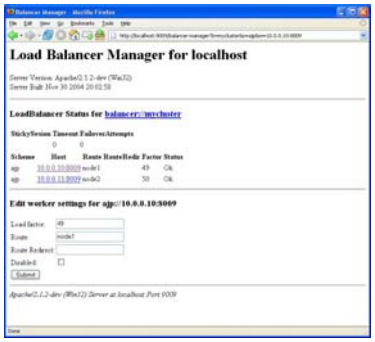
server.xml

```
<Service name="jboss.web"
   className="org.jboss.web.tomcat.tc5.StandardService">

   <!-- A AJP 1.3 Connector on port 8009 -->
   <Connector port="8009" address="${jboss.bind.address}"
      enableLookups="false" redirectPort="8443" debug="0"
      maxThreads="150" minSpareThreads="25"
      protocol="AJP/1.3"/>

   <Engine name="jboss.web" defaultHost="localhost"
    jvmRoute="host1">
```

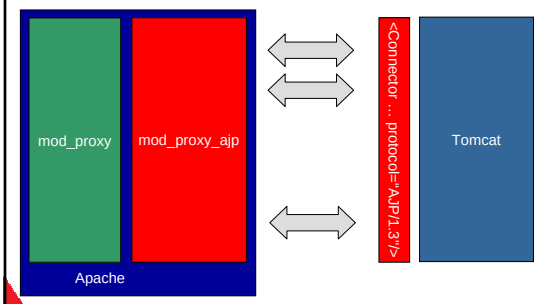
Integration using mod_proxy

- Apache 2.1/2.2
  - ✓ mod_proxy_ajp
  - ✓ mod_proxy_balancer
- Integrated within core
  - ✓ Rewrite
  - ✓ Cache
  - ✓ Compression
- No need to build additional module

Web based management



Mod_proxy Architecture



Mod_proxy load balancing

- Multiple algorithms
  - ✓ Requests
  - ✓ Transfer rate
  - ✓ Custom (CPU load?)

```
<Proxy balancer://mycluster>
    BalancerMember ajp://10.0.0.10:8009 min=10 max=100 route=node1 loadfactor=1
    BalancerMember ajp://10.0.0.11:8009 min=20 max=200 route=node2 loadfactor=2
</Proxy>
ProxyPass /servlets-examples balancer://mycluster/servlets-examples
```


Future and AJP14

- Missing features in AJP13
  - ✓ Feedback from Tomcat
- Missing features in JK
  - ✓ Proper way to remove the node from the cluster
  - ✓ Dynamic management
  - ✓ Shared memory

AJP14

- Multicast support
  - ✓ Join node to exiting cluster
- Single and two phase authentication
- Encryption
- Compression
- Node topology discovery
  - ✓ Find deployed applications

AJP14

- Node topology status
  - ✓ I'm going to stop
  - ✓ I do not accept any new sessions
  - ✓ Please redirect to node XXX instead
  - ✓ I'm stopped, don't bother me
  - ✓ I'll shut down in a moment
  - ✓ I'm too busy, take it easy
  - ✓ I have new application deployed

AJP14

- Node topology management
  - ✓ Change connection load
  - ✓ Change session routes
- Cluster topology discovery
  - ✓ Find nodes
  - ✓ Find node balance factors
  - ✓ Uses multicast
- Q3 2005

Mod_jk 1.2.10

- March 2005
- Shared memory
  - ✓ Resolves equal load presumption
- Management Web based console
  - ✓ Change balance factors
  - ✓ Change route
  - ✓ Put node offline
- Put node in maintenace resolved!

Q&A

- Tomcat Developer List
  - ✓ See jakarta.apache.org
- mturk@jboss.com