

BroadVision 8.1 Tuning

1. Enable text cache

If a program retrieve content items from a category, then use text cache. For Portal only.

```
sqlplus <user>/<password>@<sid>
update BV_KM_PROGRAM set CACHEABLE=1 where PROGRAM_SRC_TYPE=1;
commit;
quit
```

If you're running the Kukini-commerce sample app, some pages default to using the text cache and some pages do not. To force all pages in the Kukini-commerce sample app to use the text cache, edit the web.xml file in shell.war and add the following:

```
<context-param>
  <param-name>forceUseTextCache</param-name>
  <param-value>true</param-value>
</context-param>
```

2. Enable Web Membership Cache and set read from cache

A new cache is available in SuperPortal 8.1AA, the Web Membership Cache. It caches a category's children in the web tier. A new parameter is also available, named Read_through, which if set to false, would change the default for obtaining visitor profile information to read from cache.

i) Open \$BV1TO1/baseSuites/bv_services/modules/svc_clt/bv-param.xml:

ii) Add *WebMembershipCache* to *CACHE_NAME_LIST*:

```
<param name="CACHE_NAME_LIST">
<description>data area property </description>
<type>string</type>
<value>RuleSetCache RuleCache IfCache ThenCache CommunityCache TextCache
VisitorGroupCache ScheduleParamsCache LocaleRuntimeCache ACLAccessGroupCache
DelegationCache ACLPermittedRolesCache SearchCache DefaultSequenceCache
ContentCache WebMembershipCache</value>
</param>
```

iii) Add a section for *WebMembershipCache* in the Cache definitions section:

```
<context name="bv.core.webcache.WebMembershipCache">
<param name="CACHE_FACTORY">
<description>cache factory class</description>
<type>string</type>
<value>com.broadvision.data.cache.WebMembershipCacheFactory</value>
</param>
<param name="CACHE_TIME_OUT">
<description>cache time out in seconds</description>
<type>integer</type>
<value>900</value>
</param>
<param name="CACHE_SIZE">
```

```

<description>cache size </description>
<type>integer</type>
<value>50000</value>
</param>
<param name="CACHE_AUTO_CREATE">
<description>create cache automatically on cache manager
initialization</description>
<type>boolean</type>
<value>true</value>
</param>
</context>

```

iv) Add the READ_THROUGH parameter to the bv.core.visitor context:

```

<!-- context: bv.core.visitor -->
...
<param name="READ_THROUGH">
<description>false indicates read from cache</description>
<type>boolean</type>
<value>FALSE</value>
</param>

```

v) Save the file.

vi) Repeat steps ii) through v) for the file, \$BV1TO1/baseSuites/commerce/modules/svc_clt/bv-param.xml.

vii) Run bvtool deploy-config -no-res. Type *No* at the prompt.

viii) Restart JBoss.

3. Set JBoss to use call by reference

Update Tomcat's jboss-service.xml, ear-deployer.xml and bv-bootstrap-service.xml to configure **call by reference**.

i) Update \$JBOSS_HOME/server/<config_name>/deploy/jbossweb-tomcat55.sar/META-INF/jboss-service.xml:

```

<attribute name="Java2ClassLoadingCompliance">true</attribute>

```

ii) Update \$JBOSS_HOME/server/<config_name>/deploy/ear-deployer.xml:

```

<attribute name="CallByValue">false</attribute>

```

iii) If you also deployed bvaxis.war, you would get an exception when JBoss is starting up with the above settings. To prevent this, add the following line to \$JBOSS_HOME/server/<config_name>/deploy/bv-bootstrapservice.xml:

```

<classpath archives="commons-discovery.jar"
codebase="file:/install_dir/CLASSES/wsrp/" />

```

Using CallByValue=false and Java2ClassLoadingCompliance=true also causes Single Sign-on to fail in Portal (BVNqa47759). To make Single Sign-on work, add an entry to \$BV1TO1/JBoss/server/bv_framework/deploy/bv-bootstrap-service.xml:

```

<classpath archives="jakarta-regexp-1.2.jar" codebase="file:/$BV1TO1/CLASSES/wsrp/"
/>

```

4. Set development to false in Tomcat

Update Tomcat to a production environment by adding four new lines to
\$JBOSS_HOME/server/<config_name>/deploy/jbossweb-tomcat55.sar/conf/web.xml:

```
<servlet>
  <servlet-name>jsp</servlet-name>
  <servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class>
  ...
<init-param>
<param-name>development</param-name>
<param-value>false</param-value>
</init-param>
</servlet>
```

5. Disable JBoss Hot Deploy

\$JBOSS_HOME/server/<config_name>/conf/jboss-service.xml

```
<mbean code="org.jboss.deployment.scanner.URLDeploymentScanner"
name="jboss.deployment:type=DeploymentScanner,flavor=URL">
...
<!-- A flag to disable the scans -->
<attribute name="ScanEnabled">false</attribute>
```

Note: When hot deploy is disabled, you will not be able to deploy any new files without server restart.

6. Comment out unused Connector in Tomcat

Comment out the HTTP or AJP connector that is not used, by editing
\$JBOSS_HOME/server/<config_name>/deploy/jbossweb-tomcat55.sar/server.xml

```
<!-- A AJP 1.3 Connector on port 7113 -->
<!--
<Connector port="7113" address="{jboss.bind.address}"
emptySessionPath="true" enableLookups="false" redirectPort="7111"
protocol="AJP/1.3"/>
-->
```

7. Increase database connection pool size

\$JBOSS_HOME/server/<config_name>/deploy/bv_framework.BVRuntimeDBPool-service.xml

```
<server>
...
<mbean code="org.jboss.resource.connectionmanager.JBossManagedConnectionPool"
display-name="Broadvision connection pool: bv_framework.BVRuntimeDBPool"
name="bv.ds:service=ManagedConnectionPool,name=bv_framework.BVRuntimeDBPool">
  <depends optional-attribute-
name="ManagedConnectionFactoryName">bv.ds:service=ManagedConnectionFactory,name=bv_
framework.BVRuntimeDBPool</depends>
```

```

    <attribute name="Criteria">ByNothing</attribute>
    <attribute name="MaxSize">60</attribute>
    <attribute name="MinSize">40</attribute>
    ...
  </mbean>
</server>

```

8. Modify JVM heap size

`$JBOSS_HOME/server/<config_name>/run_BV.cmd`

```

set JAVA_OPTS=%JAVA_OPTS% -server -Xms1024m -Xmx1024m -XX:NewRatio=2 -
XX:PermSize=128m -XX:+DisableExplicitGC

```

`$JBOSS_HOME/server/<config_name>/run_BV.sh`

```

JAVA_OPTS="$JAVA_OPTS -Xms1024m -Xmx1024m -XX:PermSize=128m -XX:NewRatio=2 -
XX:+DisableExplicitGC"

```

9. Limit JBoss and BroadVision logging

i) Limit JBoss logging to file only and to WARN level.

`$JBOSS_HOME/server/<config_name>/conf/log4j.xml`

```

<appender name="FILE" class="org.jboss.logging.appender.DailyRollingFileAppender">
  ...
  <param name="Append" value="false"/>
  <param name="Threshold" value="INFO"/>
  ...

<root>
  <priority value="INFO" />
  <!-- appender-ref ref=CONSOLE"/ -->
  <appender-ref ref="FILE"/>
</root>

```

ii) Limit BroadVision logging to WARN

`$BV1TO1_VAR/appConfig/<config_name>/etc/log-config.xml`

```

<root>
  <level value = "WARN" class="com.broadvision.basic.log.LogLevel"/>
  <appender-ref ref="FileAppender"/>
</root>

```

10. Use new Linux threading model (Linux only)

By default, JBoss uses the old threading model of Linux. Each thread on Linux is a process and `ps`, `pstree` or `top` does not understand this. The clue is that when you run `top`, they all have the same memory allocation.

Linux 2.6 kernels use a new threading model called NPTL. This behaves the same way as other operating systems. You only see one process. This threading model is more scalable. Configure JBoss to use this new model by commenting out a line in `$JBOSS_HOME/server/<config_name>/run_BV.sh`:

```
# Linux OS
if [ `uname` = "Linux" ]; then
    LD_ASSUME_KERNEL=2.4.1
fi
#export LD_ASSUME_KERNEL
```