JavaOne™

JavaOne™

50 Tips in 50 Minutes for GlassFish Fans

Arun Gupta
Chris Kasso

MAKE THE
FUTURE
JAVA

ORACLE®

Program Agenda

- Introduction
- 50 Tips (50 minutes)
- Q & A

Introduction

- About Arun and Chris
- About GlassFish
  - It is the Java EE reference implementation.
  - It is freely available (and commercially supported).
  - Go to glassfish.java.net for details.
- Goals today:
  - Fast pace session covering a wide range of topics related to GlassFish.
  - Prioritizing quantity of tips over depth of information.
  - Tips are interesting to new and seasoned users.
  - Your response: “Ah, I didn’t know I can do that.”

Introduction

- This presentation will be a good reference.
- For more information about the tips visit:

  http://wikis.oracle.com/display/GlassFish/50+Tips

Tip #1: Using the login Command

- To execute a remote command with asadmin you must provide the admin's user name and password.
- The login command allows you to store the login credentials to be reused in subsequent commands.
- Can be logged into multiple servers (distinguish by host and port).

Tip #1: Using the login Command

Example:

```
% asadmin --host ouch login
Enter admin user name [default: admin]>
Enter admin password>
Login information relevant to admin user name [admin]
for host [ouch] and admin port [4848] stored at
[/Users/ckasso/.asadminpass] successfully.
Make sure that this file remains protected.
Information stored in this file will be used by
asadmin commands to manage the associated domain.
Command login executed successfully.
% asadmin --host ouch list-clusters
c1 not running
Command list-clusters executed successfully.
```

Tip #2: Use Multimode for Batch Processing

- Multiple Command Mode (a.k.a multimode)
- Ideal for batch processing
- Can invoke multimode from within multimode
- Any options used with multimode apply to subcommands

Example:

```
% cat batchfile.txt
create-cluster c1
create-local-instance --cluster c1 i1
create-local-instance --cluster c1 i2
% asadmin --terse --user admin multimode --file ./batchfile.txt
```

Tip #3: Use Shell Pipelines With Multimode

- Can pipe commands to asadmin
- Useful when incorporating asadmin into scripts
- Useful for repeated commands

Example: (create 5 server instances)

```
 % for i in 0 1 2 3 4; do echo "create-local-instance server$i";
done | asadmin
```

Tip #4: Using the AS_DEBUG Env Variable

- Environment variable to control client side debug output
- Exposes:
  - command processing info
  - URL used to access the command:
    - http://localhost:4848/__asadmin/uptime
  - Raw response from the server

Tip #4: Using the AS_DEBUG Env Variable

Example:

```
% export AS_DEBUG=true
% asadmin uptime
CLASSPATH= ./../glassfish/modules/admin-cli.jar
Commands: [uptime]
asadmin extension directory: /work/gf-3.1.2/glassfish3/glassfish/lib/asadm
------- RAW RESPONSE  ---------
 Signature-Version: 1.0
 message: Up 7 mins 10 secs
 milliseconds_value: 430194
 keys: milliseconds
 milliseconds_name: milliseconds
 use-main-children-attribute: false
 exit-code: SUCCESS
------- RAW RESPONSE  ---------
```

Tip #5: The AS_LOGFILE Environment Variable

- Environment variable to control client side log file of executed commands.
- Records subcommands which are run.
- Works with multimode too.

Tip #5: The AS_LOGFILE Environment Variable

Example:

```
% export AS_LOGFILE=/tmp/aslog.txt
% <run a bunch of asadmin commands>
% cat /tmp/aslog.txt
08/20/2012 14:17:12 EXIT: 0 asadmin login
08/20/2012 14:06:00 EXIT: 0 asadmin uptime
08/20/2012 14:06:23 EXIT: 0 asadmin list-clusters
08/20/2012 14:06:38 EXIT: 0 asadmin create-cluster c1
08/20/2012 14:08:44 EXIT: 1 asadmin create-cluster c1
08/20/2012 14:14:28 EXIT: 0 asadmin list-commands
08/20/2012 14:14:37 EXIT: 0 asadmin create-jvm-options --help
08/20/2012 16:05:19 EXIT: 0 asadmin version
```

Tip #6: How to Build the Workspace

```
svn checkout
https://svn.java.net/svn/glassfish~svn/trunk/main glassfish
export MAVEN_OPTS=-Xmx512m
cd glassfish
mvn

[INFO] Total time: 5:34.593s
[INFO] Finished at: Tue Sep 18 08:12:47 PDT 2012
[INFO] Final Memory: 333M/508M
```

JDK 1.6 U22+

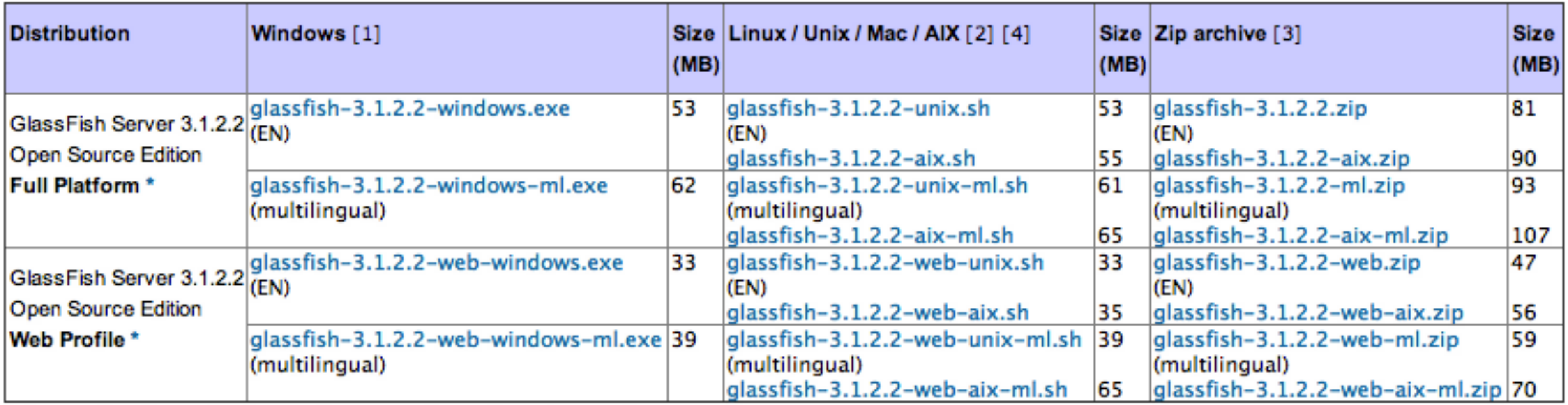
maven

Trunk: 3.0.3+

3.x: 2.2.1+

Tags: https://svn.java.net/svn/glassfish~svn/tags/
For example: https://svn.java.net/svn/glassfish~svn/tags/3.1.2

Tip #7: Installers: Zip, GUI, Silent



Distribution	Windows [1]	Size (MB)	Linux / Unix / Mac / AIX [2] [4]	Size (MB)	Zip archive [3]	Size (MB)
GlassFish Server 3.1.2.2 Open Source Edition **Full Platform** *	glassfish-3.1.2.2-windows.exe (EN)	53	glassfish-3.1.2.2-unix.sh (EN) glassfish-3.1.2.2-aix.sh	53 55	glassfish-3.1.2.2.zip (EN) glassfish-3.1.2.2-aix.zip	81 90
	glassfish-3.1.2.2-windows-ml.exe (multilingual)	62	glassfish-3.1.2.2-unix-ml.sh (multilingual) glassfish-3.1.2.2-aix-ml.sh	61 65	glassfish-3.1.2.2-ml.zip (multilingual) glassfish-3.1.2.2-aix-ml.zip	93 107
GlassFish Server 3.1.2.2 Open Source Edition **Web Profile** *	glassfish-3.1.2.2-web-windows.exe (EN)	33	glassfish-3.1.2.2-web-unix.sh (EN) glassfish-3.1.2.2-web-aix.sh	33 35	glassfish-3.1.2.2-web.zip (EN) glassfish-3.1.2.2-web-aix.zip	47 56
	glassfish-3.1.2.2-web-windows-ml.exe (multilingual)	39	glassfish-3.1.2.2-web-unix-ml.sh (multilingual) glassfish-3.1.2.2-web-aix-ml.sh	39 65	glassfish-3.1.2.2-web-ml.zip (multilingual) glassfish-3.1.2.2-web-aix-ml.zip	59 70

Same configurations for Oracle GlassFish Server

`–n`: Dry run to create *answer file*
`–a`: Use the *answer file*

```
glassfish-3.1.2.2-unix.sh —a myAnswer
glassfish-3.1.2.2-unix.sh —a myAnswer -s
```

Tip #8: asadmin Command Help

```
./bin/asadmin list-commands
********** Local Commands **********
backup-domain
change-admin-password
change-master-password
. . .
********** Remote Commands **********
add-resources
apply-http-lb-changes
change-master-broker
. . .
```

```
./bin/asadmin start --localonly=true
********** Local Commands **********
backup-domain
change-admin-password
. . .
```

```
./bin/asadmin start --remoteonly=true
********** Remote Commands **********
add-resources
apply-http-lb-changes
. . .
```

Tip #8: asadmin Command Help

```
./bin/asadmin start
Command start not found.
Check the entry of command name. This
command may be provided by a package
that is not installed.
Closest matching local and remote
command(s):
    restart-domain
    restart-instance
    restart-local-instance
    start-cluster
    start-database
    start-domain
    start-instance
    start-local-instance
Command start failed.
```

Tip #9: Secure Administration

- Remote administration is disabled by default
  - Reduce exposure to network attack
- Change admin password (non-empty)
  - `asadmin change-admin-password`
- Enable remote administration (restart DAS & instances)
  - `asadmin enable-secure-admin`
  - Also encrypts admin traffic (using self-signed certs and TLS, <3.1.1)
- Disable secure administration
  - `asadmin disable-secure-admin`

Tip #10: Configure Log4J

- Global
  - Copy `log4j.jar` in `glassfish/lib`
  - Copy `log4j.properties` in `glassfish/domains/domain1/config`
  - `asadmin create-jvm-options -Dlog4j.configuration=file\`
    `\://${com.sun.aas.instanceRoot}/config/log4j.properties`
  - `asadmin restart-domain`
- Per application: add `log4j.xml` in `WEB-INF/classes` or root of `ejb-beans.jar`

Tip #11: Using Password Aliases

- Some resources require a password to access (e.g. DB, JMS, etc.).
- The resource connector is defined in the domain.xml.

Example:

Suppose the DB resource you wish to access requires an entry like this in the domain.xml:

```
<property name="password" value="secretp@ssword"/>
```

But company policies do not allow you to store the password in the clear.

Tip #11: Using Password Aliases

- Use password aliases to avoid storing the password in the domain.xml
- Create a password alias:

```
% asadmin create-password-alias DB_pw_alias
Enter the alias password>
Enter the alias password again>
Command create-password-alias executed successfully.
```

- The password is stored in domain's encrypted keystore.
- Now update the password value in the domain.xml:

```
<property name="password" value="${ALIAS=DB_pw_alias}"/>
```

Tip #12: Using install-node to Create a Cluster

- Before a cluster can be created the GlassFish software must be installed on the remote nodes - all of them.
- Use install-node from the DAS to install GlassFish on the target nodes.
- Use install-node-dcom for Windows hosts (DCOM must be enabled on target).
- The command creates a glassfish.zip or uses an existing zip.
- Accepts a list of hosts and a install path.
- Can specify an alternate ssh user.
- uninstall-node[-dcom] is available.

Example:

```
     % asadmin install-node --installdir /export/glassfish
host1.foo.com host2.foo.com
```

Tip #13: Validating Cluster Networking

- GlassFish supports dynamic clustering.
- Uses the Group Management Service (GMS) which depends on multicast.
- GMS is used for cluster shape change events, p-to-p messaging, etc.
- Network must support mutlicast.
- Use validate-multicast to verify clustering will work properly.
- Can be used to diagnose problems where the DAS and instances can not see each other.
- Run the command on all the hosts that will be part of the cluster

Tip #13: Validating Cluster Networking

Example:

```
% asadmin validate-multicast
Will use port 2048
Will use address 228.9.3.1
Will use bind interface null
Will use wait period 2,000 (in milliseconds)

Listening for data...
Sending message with content "host1" every 2,000 milliseconds
Received data from host1 (loopback)
Received data from host2
Exiting after 20 seconds. To change this timeout, use the
--timeout command line option.
Command validate-multicast executed successfully.
```

Tip #14: Checking the Domain’s Status

- What is the status of all the domains?

```
% asadmin list-domains
domain1 running
mydomaintest not running
Command list-domains executed successfully.
```

- How long has a domain been running?

```
% asadmin –host domainhost --port 12345 uptime
Up 4 days
Command uptime executed successfully.
```

Tip #15: Checking the Cluster’s Status

- What's the status of my cluster?

```
% asadmin get-health mycluster
instance1 started since Mon Sep 10 17:31:06 PDT 2012
instance2 stopped since Mon Sep 10 18:11:21 PDT 2012
Command get-health executed successfully.
```

Tip #16: Clustering with DCOM

- Alternative to SSH-based clustering on Windows machines
- Minimal configuration, if any, required
- Windows DAS ➔Windows remote
- New commands
  - `setup-local-dcom:` Configure DCOM
  - `validate-dcom:` Sanity check for DCOM
  - `create`/`delete`/`install`/`uninstall-node-dcom`
- Integrated in Admin Console

Tip #17: Clusters without SSH

- May not want to deal with SSH
- Need JDK 1.6 U24+
- Start DAS: `asadmin start-domain`
- Enable Secure Admin: `asadmin enable-secure-admin`
- Create cluster: `asadmin create-cluster c1`
- Create instances on each remote machine: `asadmin --host dashost create-local-instance --cluster c1 i1`
- Verify: `asadmin list-clusters, list-instances, list-nodes`
- Start local instances: `asadmin start-local-instance`

Tip #18: Lifecycle Modules

- Modules automatically initiated at server startup
- Notified at different phases of the server lifecycle
  - Implement `com.sun.appserv.server.LifecycleListener` (in `glassfish/modules/glassfish-api.jar`)

```
@Override
public void handleEvent(LifecycleEvent le) throws ServerLifecycleException {
    switch (le.getEventType()) {
        case INIT_EVENT:
        case READY_EVENT:
        case SHUTDOWN_EVENT:
        case STARTUP_EVENT:
        case SHUTDOWN_EVENT:
        default:
    }
}
```

Tip #19: Application Versioning

- Allows multiple versions of same module and application
- At most one version is active
  - Enabled using `--enabled` (default `true`)
- Simplifies upgrade and rollbacks
- Examples
  - `asadmin deploy --name=myApp:RC1 --enabled=false myApp.war`
  - `asadmin enable myApp:RC1`
  - `asadmin undeploy myApp:RC*`

Tip #20: Application Scoped Resources

- Available only to module/application that define it
- Created/destroyed/recreated during deployment/undeployment/ redeployment
- Free from *resource starvation*
- Applies to: JDBC/Connector Connection Pools/Resources, Resource Adapters, JavaMail Resources, …
- `glassfish-resources.xml` in `WEB-INF`
- `redeploy --properties preserveAppScopedResources=true myApp.war`

```
<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <jdbc-resource enabled="true"
      jndi-name="jdbc/myDatasource"
      object-type="user"
      pool-name="myConnectionPool">
    <description/>
  </jdbc-resource>
</resources>
```

Tip #21: How to Start GlassFish as a Service

- Configuring a server to automatically start at boot can be tedious.
- Each platform does it differently.
- The create-service command makes this easy.
  - Windows: creates a Windows service
  - Linux: /etc/init.d script
  - Solaris: Service Management Facility (SMF) service
- Must execute create-service with admin privileges.
- Can be used for the DAS or instances
- Try it first with the --dry-run option.
- There is a (unsupported) _delete-server

Tip #21: How to Start GlassFish as a Service

Example:

```
     # asadmin create-service domain1
     The Service was created successfully. Here are the details:
     Name of the service:application/GlassFish/domain1
     Type of the service:Domain
     Configuration location of the service:/work/gf-3.1.2.2/
glassfish3/glassfish/domains
     Manifest file location on the system:/var/svc/manifest/
application/GlassFish/
domain1_work_gf-3.1.2.2_glassfish3_glassfish_domains/Domain-
service-smf.xml.
     You have created the service but you need to start it
yourself.  Here are the most typical Solaris commands of interest:
     * /usr/bin/svcs  -a | grep domain1  // status
     * /usr/sbin/svcadm enable domain1 // start
     * /usr/sbin/svcadm disable domain1 // stop
     * /usr/sbin/svccfg delete domain1 // uninstall
```

Tip #22: How To Enable Server Monitoring

- Extensive monitoring is built into the server.
- CLI, REST and JMX access.
- Console provides a nice UI to enable discrete monitoring.
- Monitoring data maintained in a tree.
- Monitoring Levels:
  - LOW. Simple statistics, such as create count, byte count, and so on.
  - HIGH. Simple statistics plus method statistics, such as method count, duration, and so on.
  - OFF. No monitoring, no impact on performance.

Tip #22: How To Enable Server Monitoring

- Monitoring is not enabled by default.
- Can be enabled without restarting the server.

Example

```
% asadmin get -m “*"
i1:
No monitoring data to report.
i2:
No monitoring data to report.
```

Tip #22: How To Enable Server Monitoring

Example: Enabling monitoring of the Web Container on the DAS

```
  % asadmin enable-monitoring --modules web-container=HIGH
```

Example: Enabling monitoring of the EJB Container on a cluster

```
  % asadmin enable-monitoring --target c1 --modules ejb-
container=HIGH
```

Example: Enabling monitoring of the Web Container on an instance

```
   % asadmin enable-monitoring --target i1 --modules web-
container=HIGH
```

- Use `asadmin get server.monitoring-service.*` to list the module names and their associated monitoring levels.

Tip #23: How To View Monitoring Data (CLI)

- Monitoring data is maintained in a hierarchical tree.
- The tree changes based on:
  - What modules are loaded.
  - What modules have monitoring enabled.
  - Applications that are deployed.

Example: List the monitor elements

```
% asadmin list -m "*"
server.security
server.security.web
server.security.realm
server.applications
server.applications.hello
...
```

Tip #23: How To View Monitoring Data (CLI)

Example: Get every monitor value

```
% asadmin get -m "*"
<too much to show>
```

Example: Who implemented the JVM the server is running on?

```
% asadmin get -m "server.jvm.runtime.vmvendor-current-current"
server.jvm.runtime.vmvendor-current-current = Apple Inc.
```

Tip #23: How To View Monitoring Data (CLI)

Example: How much heap is used?

```
    % asadmin get -m "server.jvm.memory.usedheapsize*"
    server.jvm.memory.usedheapsize-count-count = 60066392
    server.jvm.memory.usedheapsize-count-description = Amount of
used memory in bytes
    server.jvm.memory.usedheapsize-count-lastsampletime =
1347410288602
    server.jvm.memory.usedheapsize-count-name = UsedHeapSize
    server.jvm.memory.usedheapsize-count-unit = bytes
```

Example: How many active session for a deployed application?

```
    % asadmin get -m
"server.applications.mywebapp.server.activesessionscurrent-current"
    server.applications.mywebapp.server.activesessionscurrent-
current = 5
```

Tip #24: Using the monitor Command

- Output is displayed continuously in a tabular format.
- Interval (in seconds) is controlled by –interval.

Example:

```
% asadmin monitor --interval 5 --type jvm server
                           JVM Monitoring
UpTime(ms)                      Heap and NonHeap Memory(bytes)
current   min        max        low        high       count
1345760   67276800   753860608  0          0          193273856
1350761   67276800   753860608  0          0          193372160
1355758   67276800   753860608  0          0          194846720
1360764   67276800   753860608  0          0          201973760
```

Tip #24: Using the monitor Command

Example

```
% asadmin monitor --interval=5 --type webmodule server
asc   ast   rst   st    ajlc  mjlc  tjlc  aslc     mslc      tslc
0     0     0     0     0     0     0     8        0         8
0     0     0     0     0     0     0     8        0         8
```

- Type “h” and return at anytime to see column header descriptions.

```
* ajlc = Number of active JSP pages
* asc  = Number of currently active sessions
* aslc = Number of active servlets that are loaded
```

- See the "run-script" subcommand for information about using JavaScript to monitor the server.

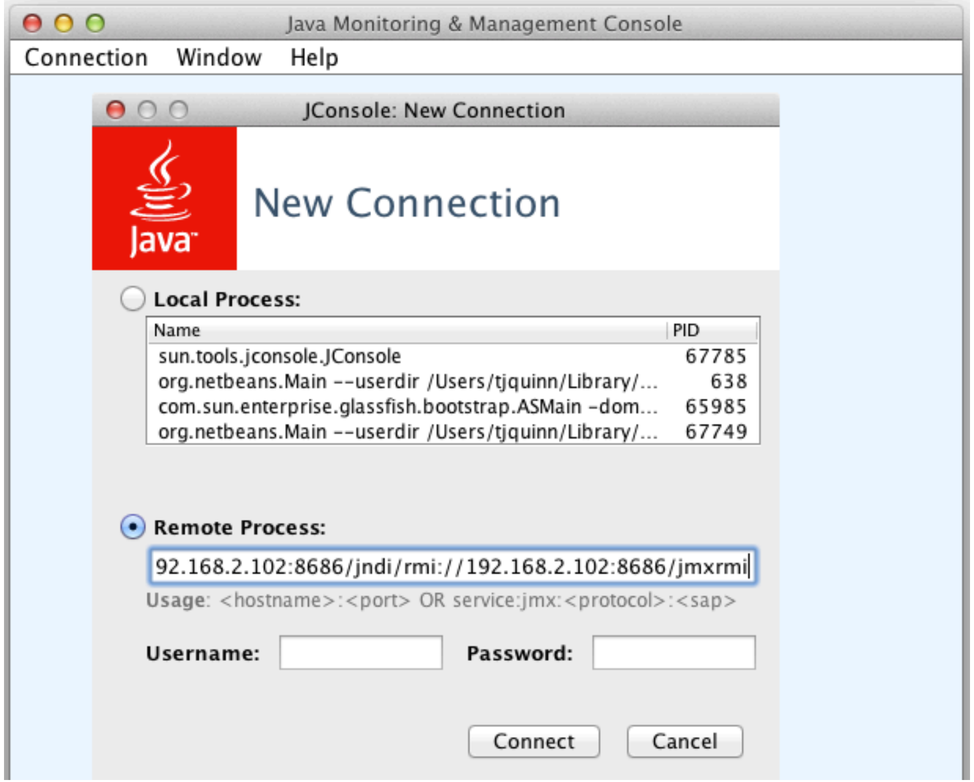
Tip #25: Using JMX to Access Monitoring

- Have you ever seen this odd message in your server log:

```
    JMXStartupService has started JMXConnector on JMXService URL
    service:jmx:rmi://192.168.2.102:8686/jndi/rmi://
192.168.2.102:8686/jmxrmi
```

- It means you can monitor GlassFish using a Java Management Extensions (JMX) client.
- Can use clients such as jconsole or jvisualvm to attach to the server.
- Works for both the DAS and instances. (See the instance's log file for the string)
- Copy the address starting with “service:” into your JMX console.

Tip #25: Using JMX to Access Monitoring



○ **Local Process:**

Name	PID
sun.tools.jconsole.JConsole	67785
org.netbeans.Main --userdir /Users/tjquinn/Library/...	638
com.sun.enterprise.glassfish.bootstrap.ASMain -dom...	65985
org.netbeans.Main --userdir /Users/tjquinn/Library/...	67749

● **Remote Process:**

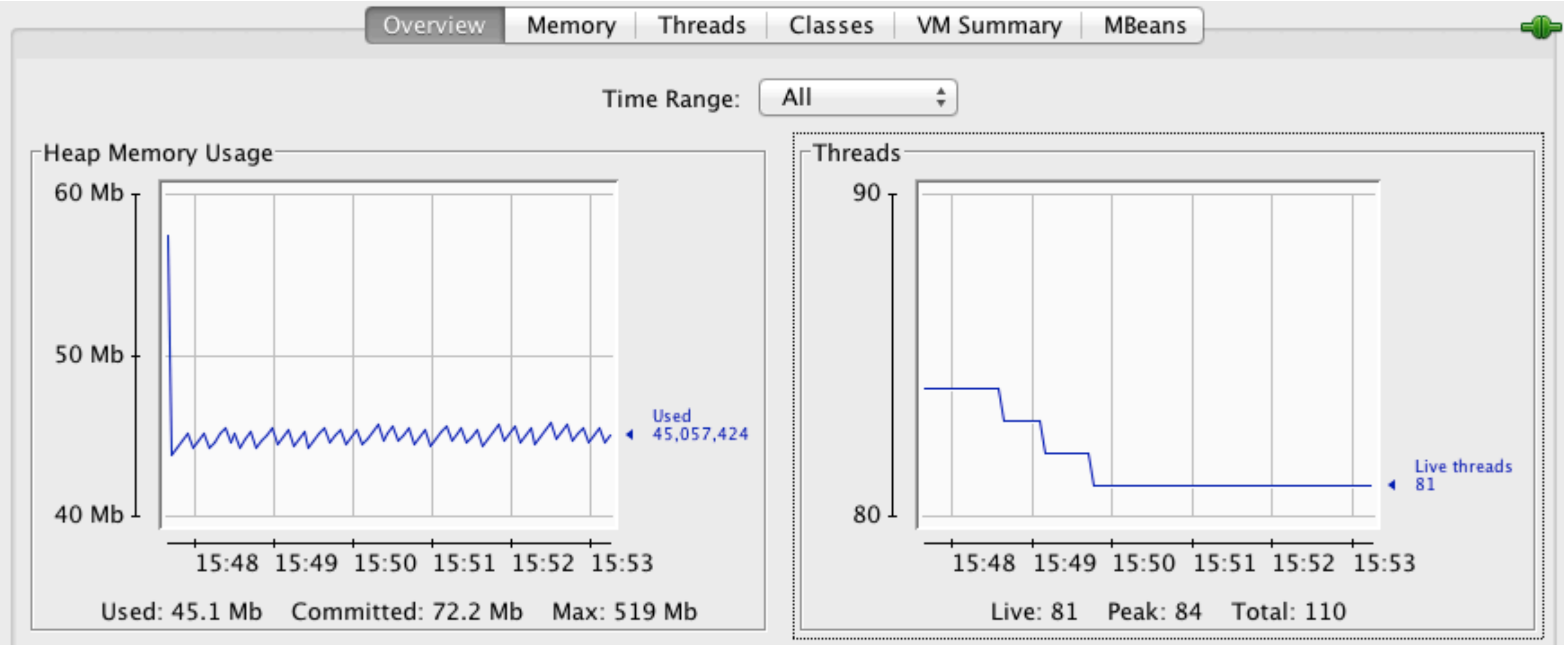
92.168.2.102:8686/jndi/rmi://192.168.2.102:8686/jmxrmi

Usage: <hostname>:<port> OR service:jmx:<protocol>:<sap>

Username:

Password:

Tip #25: Using JMX to Access Monitoring



Time Range: All

Used: 45.1 Mb Committed: 72.2 Mb Max: 519 Mb

Live: 81 Peak: 84 Total: 110

Tip #26: Inject OSGi Service in Java EE Apps

- OSGi/Java EE *Hybrid Applications* can be deployed
- Typesafe resolution of an OSGi service in a Java EE Application
- Built as CDI portable extension
- Intercepts deployment of hybrid applications with components that have expressed dependencies on OSGi services

```
@Inject @OSGiService(dynamic=true) Hello hello;
```

Tip #27: Debugging OSGi Bundles using Gogo

- Enable Apache Felix Gogo
  - Change the value of `glassfish.osgi.start.level.final` in `glassfish/config/osgi.properties` to 3
  - `asadmin create-jvm-options --target server-config -Dglassfish.osgi.start.level.final=3`

Tip #27: Debugging OSGi Bundles using Gogo

```
telnet localhost 6666
Trying ::1...
telnet: connect to address ::1: Connection refused
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

____________________________
Welcome to Apache Felix Gogo

g! help
felix:bundlelevel
felix:cd
felix:frameworklevel
. . .
```

Tip #27: Debugging OSGi Bundles using Gogo

- `lb`: List bundles
- `bundle` <##>: Show details about a bundle
- `install` <##>: Install a bundle from a URL
- `start/stop/uninstall/update` <##>: Start/Stop/Uninstall/Update a bundle
- `inspect`: Inspect capabilities and requirements of a bundle
  - `inspect p c 1` (Packages exported by bundle 1)
- `which`: Determines where a bundle loads a class

Tip #28: Run GlassFish inside an OSGi Shell

- Download GlassFish Full or Web profile
- Option 1

```
BundleContext context = FrameworkUtil.;
Bundle bundle = context
       .install(“file:///…/glassfish/modules/glassfish.jar”);
bundle.start();
bundle.stop();
```

GlassFish install and directory location are deduced from Activator

Tip #28: Run GlassFish inside an OSGi Shell

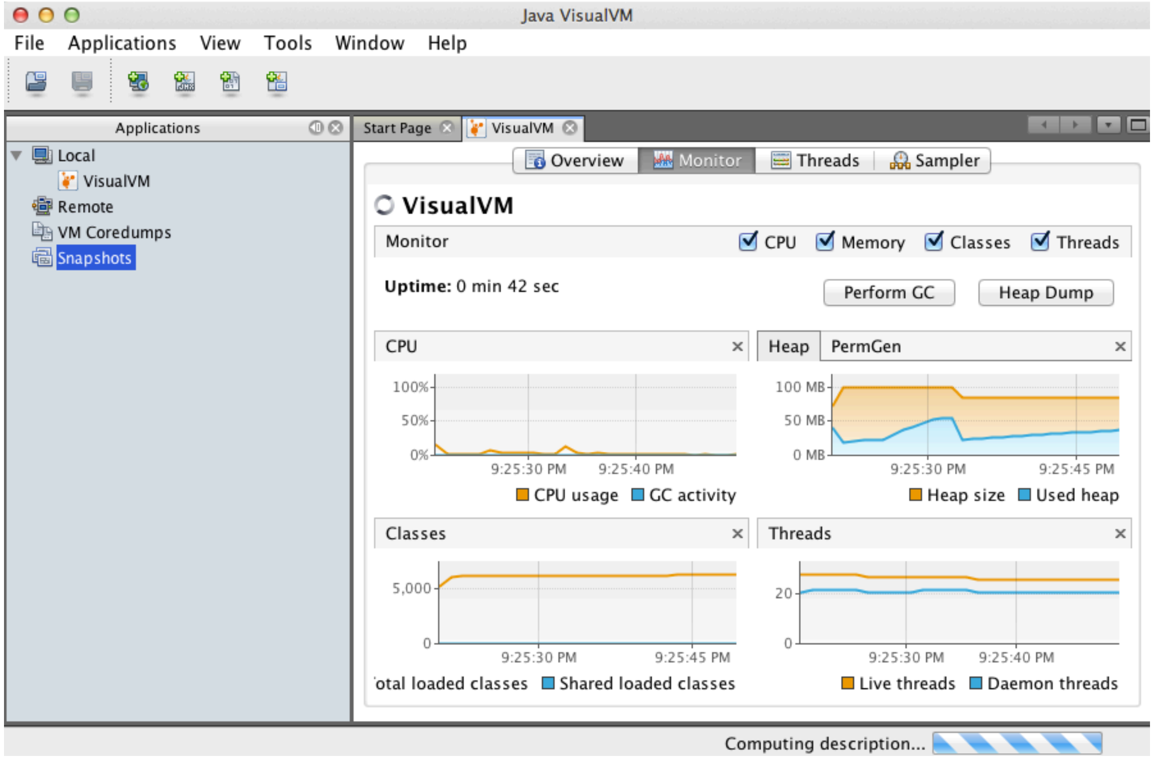
- Option 2
  - Start your OSGi shell
  - Install the bundle
  - Start the bundle

Tip #29: What's in your JVM ? – Troubleshoot

- Server thread dump
  - `asadmin generate-jvm-report –type=thread|class|memory|`**`summary`**`|log`
- JVM Command Line Tools
  - `jps`: List instrumented JVMs
  - `jstack`: stack traces of Java threads for a given process
  - `jstat`: performance statistics
    - `-class` (class loader), `-compiler` (JIT), `-gcutil` (GC stats)
    - `jstat –gcutil <VMID>`
  - `jinfo`, `jmap`, `jhat`, …

Tip #29: What’s in your JVM ? – Troubleshoot

Visual VM (`jvisualvm`)



Java VisualVM

VisualVM

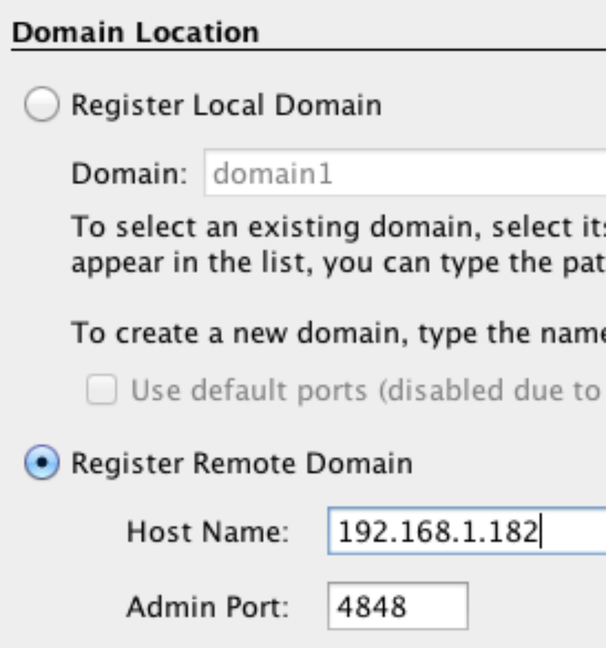
Monitor ☑ CPU ☑ Memory ☑ Classes ☑ Threads

Uptime: 0 min 42 sec

Computing description...

Tip #30: Remote Deployment in NetBeans

- Enable secure administration
- Require local GlassFish as well for libraries
- Regular operations work
  - Develop, Deploy, Debug
- Features that don’t work
  - Deploy-on-Save



Domain Location

○ Register Local Domain

Domain: domain1

To select an existing domain, select it: appear in the list, you can type the pat

To create a new domain, type the name

□ Use default ports (disabled due to

● Register Remote Domain

Host Name: 192.168.1.182

Admin Port: 4848

Tip #31: Using the GlassFish REST Interfaces

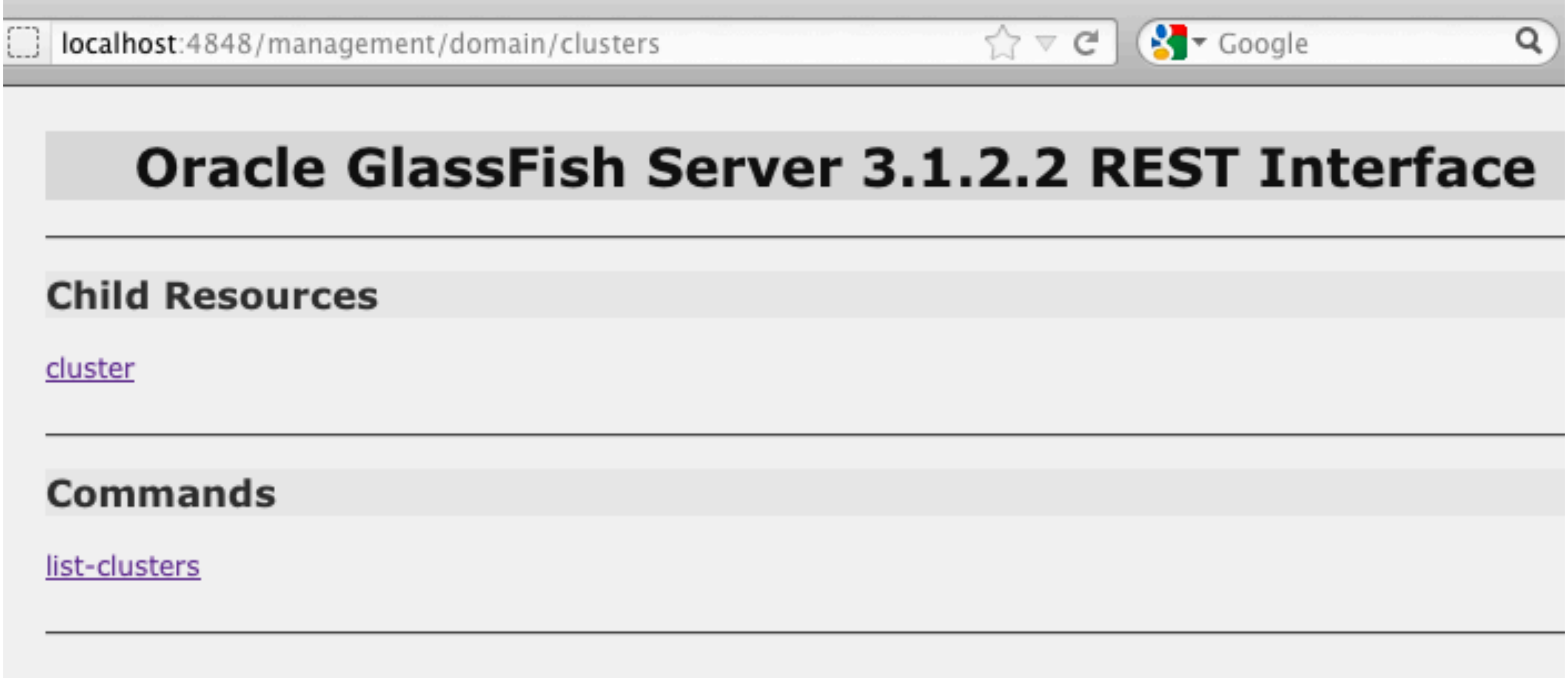
- The GlassFish Server exposes a REST interface.
- Implementation based on project Jersey.
- Management and Monitoring trees are available:
  - Configuration: http://host:port/management/domain/path
  - Monitoring: http://host:port/monitoring/domain/path

Example (viewing log entries):

```
%curl http://localhost:4848/management/domain/view-log?start=322660
```

Bonus Tip: See X-Text-Append-Next in the HTTP response for a URL to use to get latest log messages.

Tip #31: Using the GlassFish REST Interfaces



localhost:4848/management/domain/clusters

Oracle GlassFish Server 3.1.2.2 REST Interface

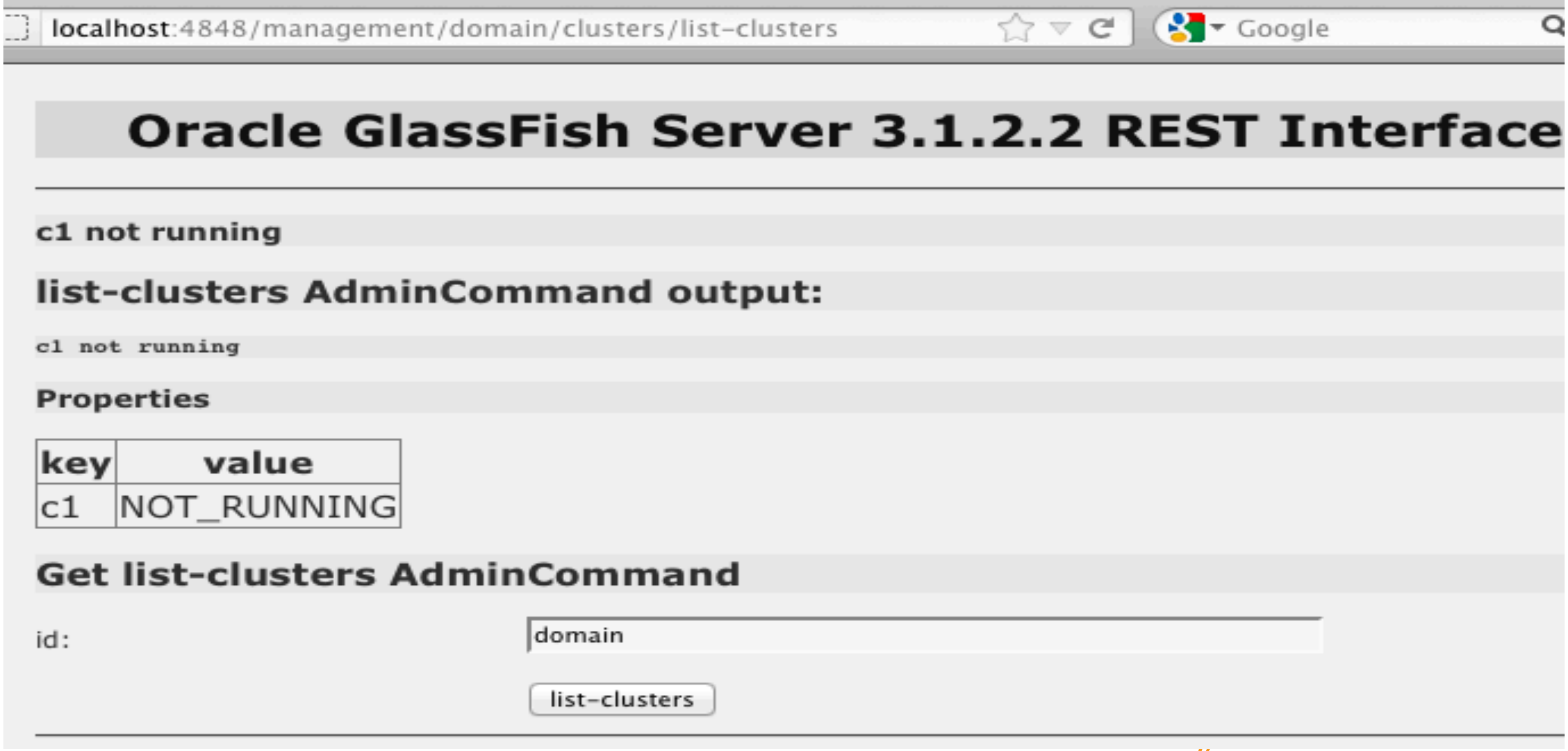
Child Resources

cluster

Commands

list-clusters

Tip #31: Using the GlassFish REST Interfaces



Oracle GlassFish Server 3.1.2.2 REST Interface

c1 not running

list-clusters AdminCommand output:

```
c1 not running
```

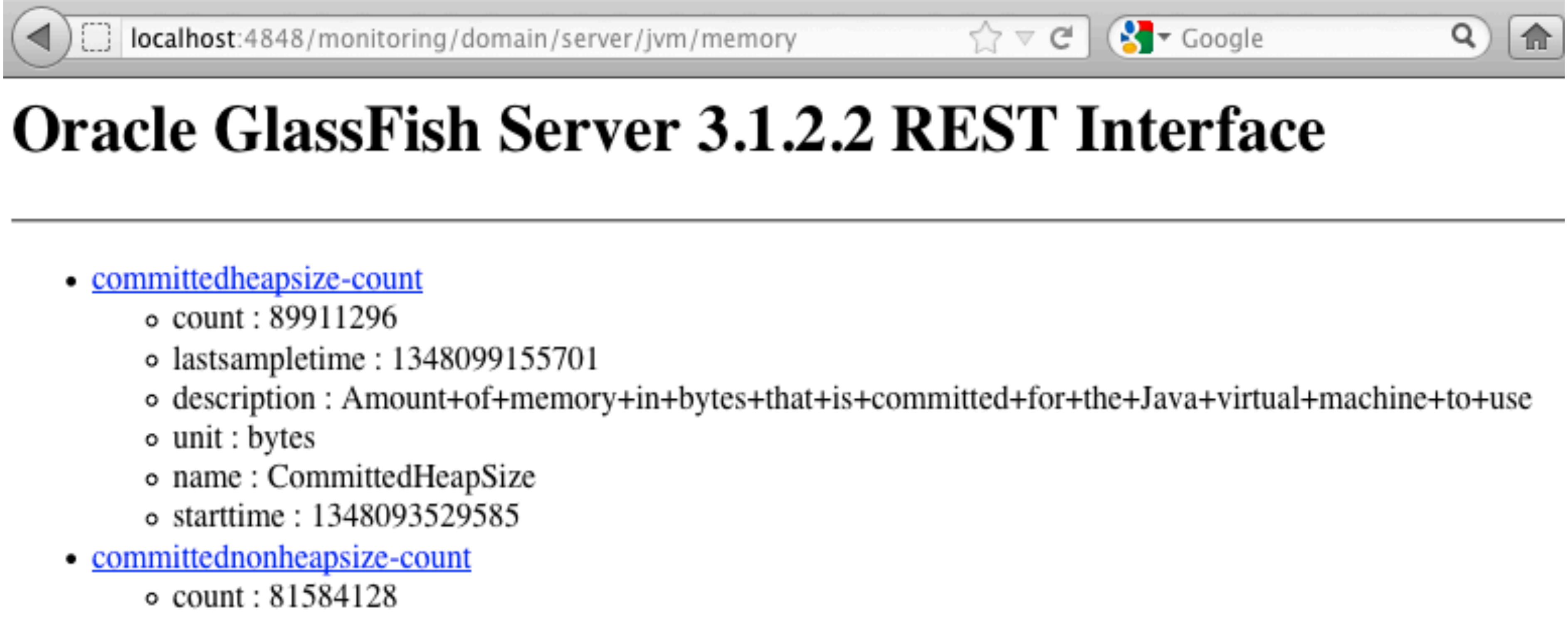
Properties

key	value
c1	NOT_RUNNING

Get list-clusters AdminCommand

id: domain

Tip #31: Using the GlassFish REST Interfaces



Oracle GlassFish Server 3.1.2.2 REST Interface

- committedheapsize-count
  - count : 89911296
  - lastsampletime : 1348099155701
  - description : Amount+of+memory+in+bytes+that+is+committed+for+the+Java+virtual+machine+to+use
  - unit : bytes
  - name : CommittedHeapSize
  - starttime : 1348093529585
- committednonheapsize-count
  - count : 81584128

Tip #32: Changing the REST Response Format

- Three formats are supported:
  - HTML
  - JSON
  - XML
- Controlled by the suffix provided in the URL:
  - http://localhost:4848/monitoring/domain/server/jvm/memory.html
  - http://localhost:4848/monitoring/domain/server/jvm/memory.json
  - http://localhost:4848/monitoring/domain/server/jvm/memory.xml

Example:

```
% curl http://localhost:4848/monitoring/domain/server/jvm/
memory.json
```

Tip #32: Changing the REST Response Format

(whitespace and newlines added for clarity)

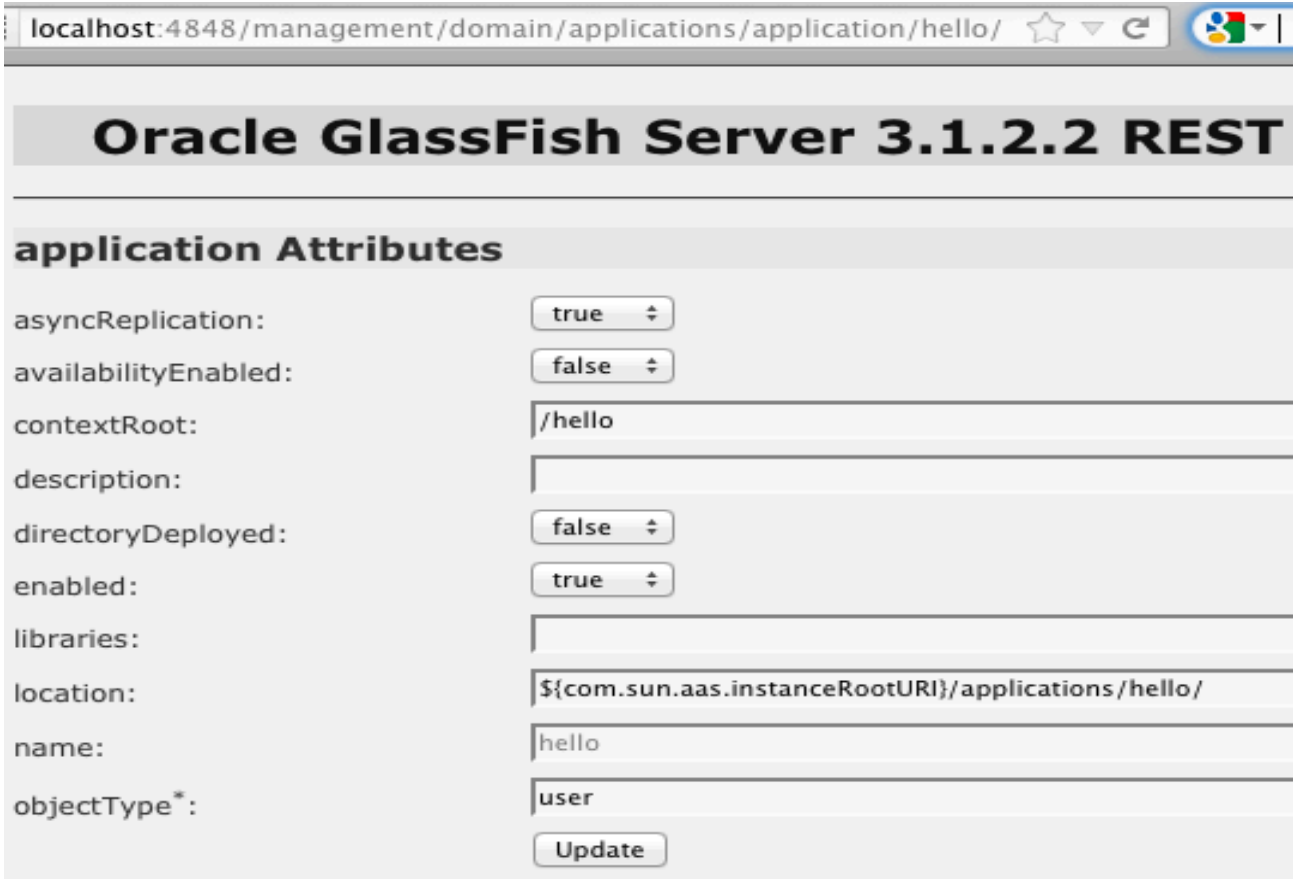
```
{
   "message":””,
   "command":"Monitoring Data”,
   "exit_code":"SUCCESS",
   "extraProperties":{
      "entity":{
          "committedheapsize-count":{
          "count":76869632,
          "lastsampletime":1348096984910,
          "description":"Amount of memory in bytes that is
   committed for the Java virtual machine to use”
   …
   …
```

Tip #33: Changing Configuration via REST

- Access the management tree via a browser to view and modify the configuration.
- Example: Modify the configuration of the deployed hello application:

http://localhost:4848/management/domain/applications/application/hello/

Tip #33: Changing Configuration via REST



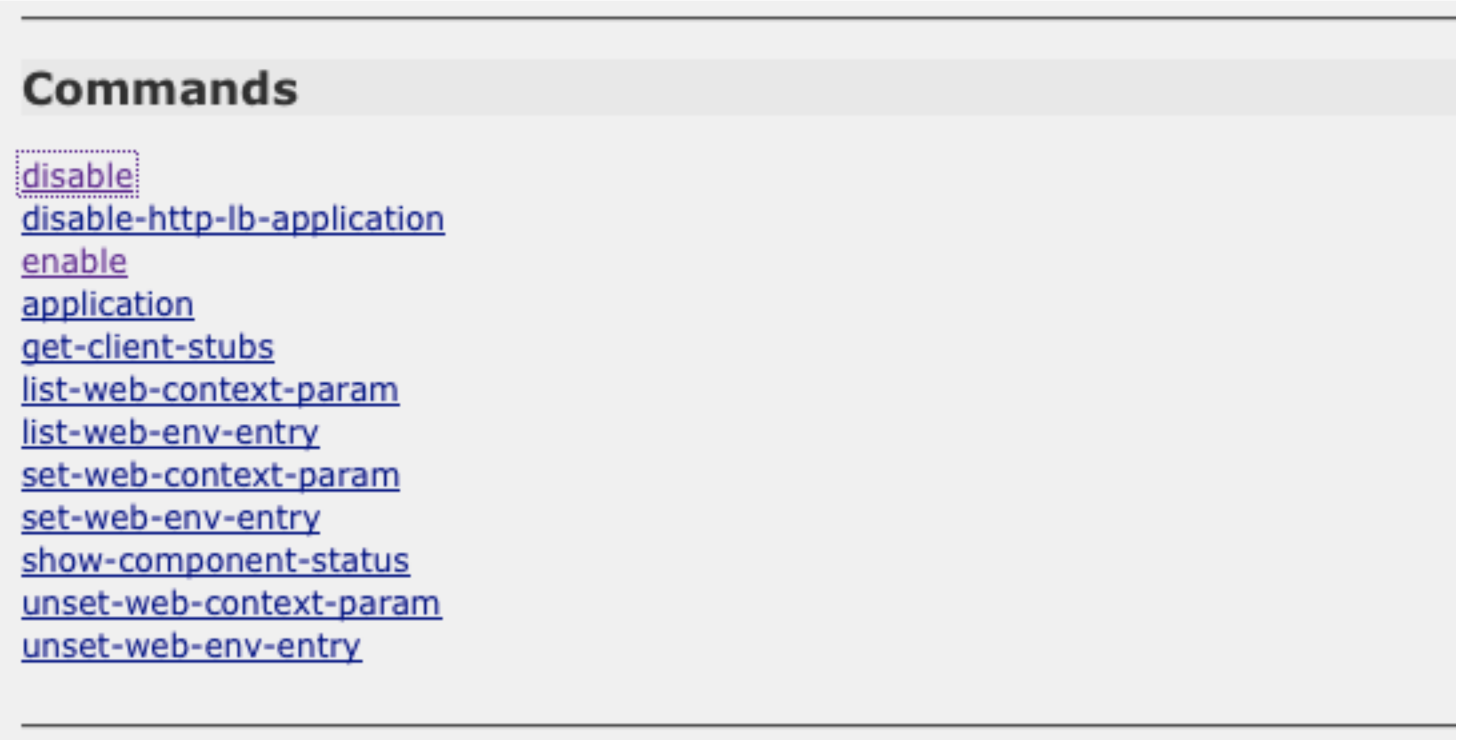
localhost:4848/management/domain/applications/application/hello/

Oracle GlassFish Server 3.1.2.2 REST

application Attributes

asyncReplication:	true
availabilityEnabled:	false
contextRoot:	/hello
description:	
directoryDeployed:	false
enabled:	true
libraries:	
location:	${com.sun.aas.instanceRootURI}/applications/hello/
name:	hello
objectType*:	user

Tip #33: Changing Configuration via REST



Commands

disable
disable-http-lb-application
enable
application
get-client-stubs
list-web-context-param
list-web-env-entry
set-web-context-param
set-web-env-entry
show-component-status
unset-web-context-param
unset-web-env-entry

Tip #34: Posting a Command via REST

- Use wget/curl to execute commands on the DAS.

Example: Deploying an application

```
    % curl -s -S \
        -H 'Accept: application/json' -X POST \
        -H 'X-Requested-By: anyvalue' \
        -F id=@/path/to/application.war \
        -F force=true http://localhost:4848/management/domain/
applications/application
```

- Use @ before a file name to tell curl to send the file's contents.
- The force option tells GlassFish to force the deployment in case the application is already deployed.

Tip #34: Posting a Command via REST

Example: Undeploy the same application

```
    % curl -s -S \
         -H 'Accept: application/json' -X DELETE \
         -H 'X-Requested-By: anyvalue' \
         http://localhost:4848/management/domain/applications/
application/hello
```

Example: Get the GlassFish version

```
    % curl -s –S -X GET http://localhost:4848/__asadmin/
version.json
```

```
    { "name":"Oracle GlassFish Server 3.1.2.2 (build 5)"
    , "command":"version AdminCommand"
    , "exit_code":"SUCCESS"
    }
```

Tip #35: Using Domain Templates

- Use domain templates to create customized domains.
- Templates live in <INSTALLDIR>/glassfish/lib/templates.
- Copy domain.xml to mydomain.xml.
- Modify mydomain.xml (e.g. change initial JVM options).

Example: create a domain via domain template

```
% asadmin create-domain --template mydomain.xml mynewdomain
...
Command create-domain executed successfully.
% asadmin verify-domain-xml mynewdomain
All tests passed; domain.xml is valid.
Command verify-domain-xml executed successfully.
```

Tip #36: GlassFish Hosting Providers

rimuhosting
eApps
PAIR ONE NETWORKS
enciva
JAVA-HOSTER
KATTARE
eroute.net
VISION WEBHOSTING
prominic .NET
EOPEN SOLUTIONS
OXXUS.NET Java hosting since 2003
[DE] SYSTEMS ENGINEERING Ltd.
glassfish hosting .com
Jelastic
HIOX India
JSP zone
SANVER E-Solutions
1stRentalServer

Tip #37: Backup and Restore Domain

- Manual
  - `backup-domain`, `restore-domain`, `list-backups`
- Automatic
  - Only in Oracle GlassFish Server
  - Create backup configuration (`create/delete-backup-config`)
  - Associate with a cron-based schedule (`create/delete-schedule`)
  - Full or Configuration only

Tip #38: Transparent JDBC Pool Reconfiguration

- No need to redeploy applications if JDBC connection pool properties/ attributes are changed.
  - Host, Port, Username, Password, …
- New requests are queued till in-flight requests are completed.
- `asadmin set server.resources.jdbc-connection-pool.<POOL_NAME>.property.dynamic-reconfiguration-wait-timeout-in-seconds=30`

Tip #39: Tracing SQL queries

- Trace SQL statements executed by a JDBC connection pool.
  - JPA or non-JPA applications
- `javax.enterpise.resource.sqltrace` log level to `FINE` (default)
- Implement `org.glassfish.api.jdbc.SQLTrceListener` to record `SQLTraceRecord` objects
  - `asadmin set server.resources.jdbc-connection-pool.DerbyPool.sql-trace-listeners=MyListener`
- Top ‘n’ most frequently used queries
  - `number-of-top-queries-to-report` (10)
  - `time-to-keep-queries-in-minutes` (5)

Tip #40: Detecting JDBC Statement and Connection Leaks and Reclaim

```
} finally {
  try {
    if (s != null)
      s.close();
    if (c != null)
      c.close();
  } catch (SQLException ex) {
    //. . .
  }
}
```

statement-leak-timeout-in-seconds="2”
statement-timeout-in-seconds="6"
connection-leak-timeout-in-seconds="10"

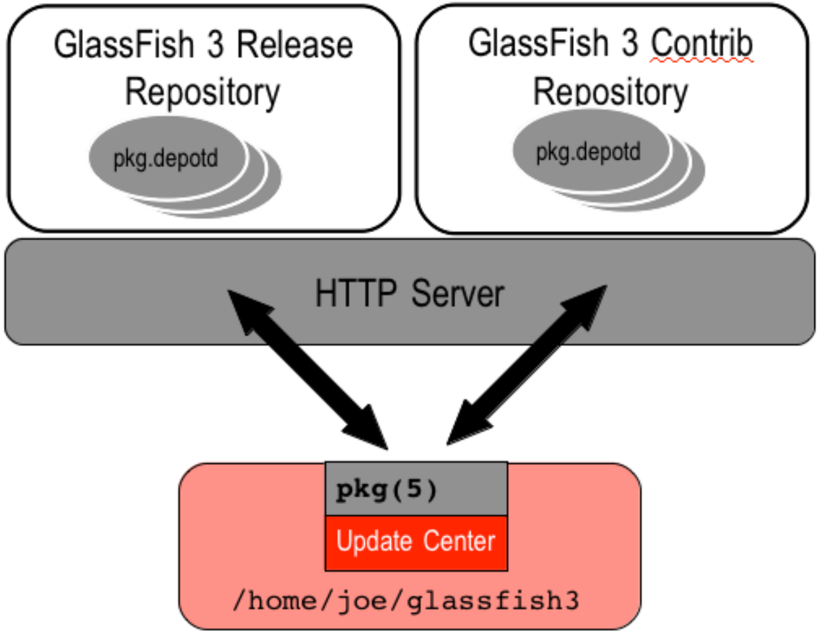
statement-leak-reclaim=“true”
connection-leak-reclaim=“true”

```
asadmin set server.resources.jdbc-connection-pool.DerbyPool.xxx=VALUE
```

Tip #41: Using Update Center to Keep Current

- Cross platform tools for software add-on / update from a network repository.
- Built on the Image Packaging System from Solaris 11.
- CLI (pkg(1)) and GUI Desktop Applications.
  - Update Tool
  - Software Update
  - Desktop balloon notification

Tip #41: Using Update Center to Keep Current



Tip #41: Using Update Center to Keep Current

- Example: Listing installed packages:

```
% pkg list
NAME (PUBLISHER)                 VERSION         STATE      UFIX
felix                            3.0.8-0         installed  ----
glassfish-appcl                  3.1.1-12        installed  ----
glassfish-clust                  3.1.1-12        installed  ----
glassfish-cmp                    3.1.1-12        installed  ----
glassfish-comm                   3.1.1-12        installed  ----
```

Tip #41: Using Update Center to Keep Current

- Example: Listing with a useful name:

```
% pkg list -s
NAME (PUBLISHER)               SUMMARY
felix                          Apache Felix
glassfish-appclient            GlassFish Appclient
glassfish-cluster              GlassFish Clustering
glassfish-cmp                  GlassFish CMP
glassfish-common               GlassFish Common Components
glassfish-common-full          GlassFish Commons Full Profile
```

Tip #41: Using Update Center to Keep Current

- Example: List installed packages that have pending updates:

```
% pkg list -u
NAME (PUBLISHE                    VERSION         STATE      UFIX
glassfish-corb                    3.1.0-30        installed  u---
glassfish-corba-base              3.1.0-30        installed  u---
glassfish-ejb                     3.1.1-12        installed  u---
glassfish-ejb-lite                3.1.1-12        installed  u---
```

Tip #41: Using Update Center to Keep Current

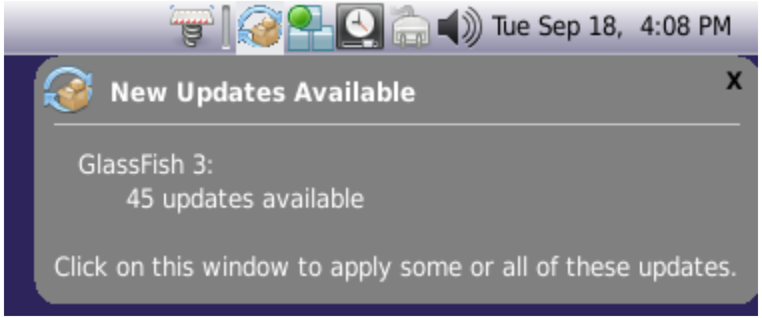
- Example: Update the GlassFish installation in seconds:

```
% pkg image-update
DOWNLOAD                          PKGS        FILES    XFER (MB)
Completed                         47/47       405/405  79.2/79.2

PHASE                                         ACTIONS
Removal Phase                                   96/96
Install Phase                                 111/111
Update Phase                                  636/636
```

Tip #42: Update Center GUI Simplifies Updates

- Desktop “notifier” installed when GlassFish is installed.
- Informs you when new updates are available.
- Use “updatetool” to display or change frequency.

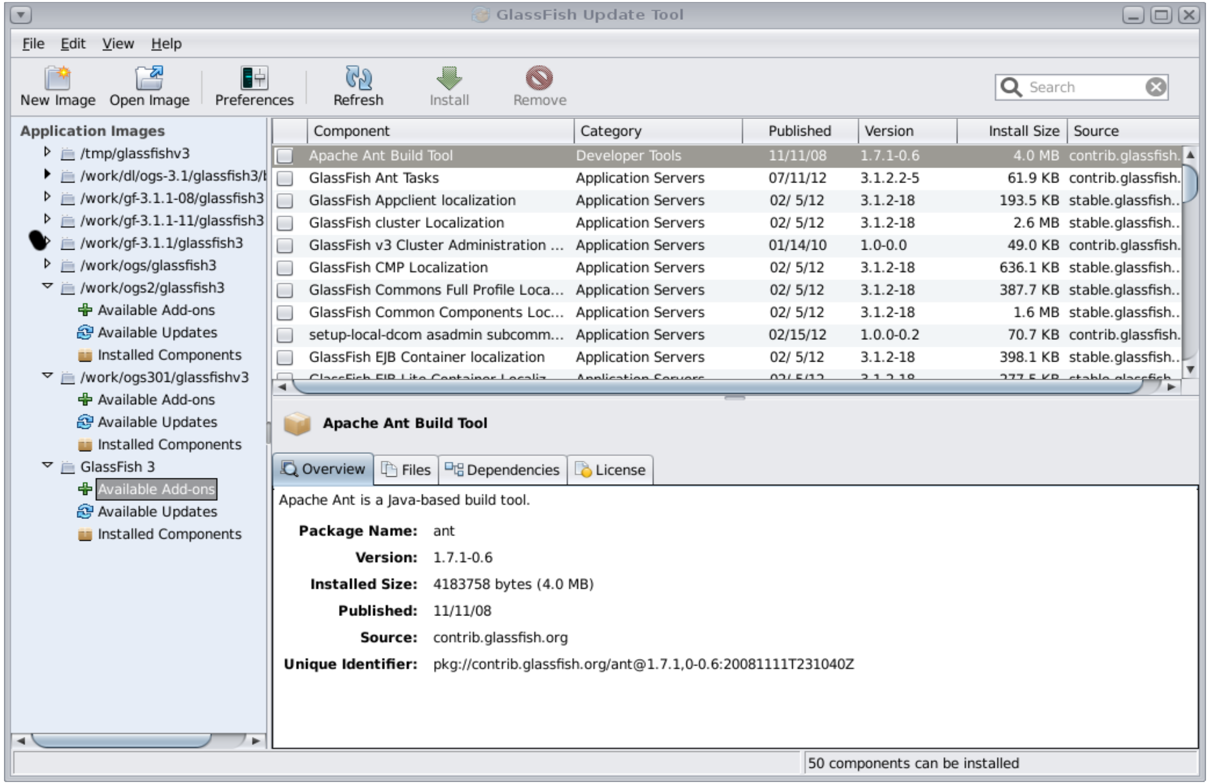


New Updates Available

GlassFish 3:
45 updates available

Click on this window to apply some or all of these updates.

Tip #42: Update Center GUI Simplifies Updates



	Component	Category	Published	Version	Install Size	Source
□	Apache Ant Build Tool	Developer Tools	11/11/08	1.7.1-0.6	4.0 MB	contrib.glassfish.
□	GlassFish Ant Tasks	Application Servers	07/11/12	3.1.2.2-5	61.9 KB	contrib.glassfish.
□	GlassFish Appclient localization	Application Servers	02/ 5/12	3.1.2-18	193.5 KB	stable.glassfish..
□	GlassFish cluster Localization	Application Servers	02/ 5/12	3.1.2-18	2.6 MB	stable.glassfish..
□	GlassFish v3 Cluster Administration ...	Application Servers	01/14/10	1.0-0.0	49.0 KB	contrib.glassfish.
□	GlassFish CMP Localization	Application Servers	02/ 5/12	3.1.2-18	636.1 KB	stable.glassfish..
□	GlassFish Commons Full Profile Loca...	Application Servers	02/ 5/12	3.1.2-18	387.7 KB	stable.glassfish..
□	GlassFish Common Components Loc...	Application Servers	02/ 5/12	3.1.2-18	1.6 MB	stable.glassfish..
□	setup-local-dcom asadmin subcomm...	Application Servers	02/15/12	1.0.0-0.2	70.7 KB	contrib.glassfish.
□	GlassFish EJB Container localization	Application Servers	02/ 5/12	3.1.2-18	398.1 KB	stable.glassfish..
	GlassFish EJB Lite Container Localiz	Application Servers	02/ 5/12	3.1.2-18	277.5 KB	stable.glassfish.

Apache Ant Build Tool

Apache Ant is a Java-based build tool.

Package Name: ant
Version: 1.7.1-0.6
Installed Size: 4183758 bytes (4.0 MB)
Published: 11/11/08
Source: contrib.glassfish.org
Unique Identifier: pkg://contrib.glassfish.org/ant@1.7.1,0-0.6:20081111T231040Z

50 components can be installed

Tip #43: How to Write a Custom Command

- Extending asadmin to support a sub-command is easy.
- Use an extensive admin framework to construct commands in a consistent way.
- Common operations are made easy.
- Can focus on the logic of the command.
- The Oracle GlassFish Server Add-On Component Development Guide has all the details.
- Decorators for CRUD based commands available too.

Tip #43: How to Write a Custom Command

Example:

```
asadmin create-mycontainer --originator any-character-string
  [--description any-character-string]
  [--enabled {true|false}] container-name
```

```
@Service(name="create-mycontainer")
@Scoped(PerLookup.class)
public Class CreateMycontainer implements AdminCommand {
```

Tip #43: How to Write a Custom Command

```
@Inject
Domain domain;

@Param
String originator;

@Param(name="description", optional=true)
@I18n("mydesc")
String mycontainerDescription

@Param (acceptableValues="true,false", defaultValue="false", optional=true)
String enabled

@Param(primary=true)
String containername;
```

Tip #43: How to Write a Custom Command

```
/**
 * Executes the subcommand with the subcommand parameters passed as Properties
 * where the keys are the paramter names and the values the parameter values
 * @param context information
 */
public void execute(AdminCommandContext context) {
    ActionReport report = context.getActionReport();

    // Do command specific work.

    if (successful) {
        report.setActionExitCode(ExitCode.SUCCESS);
    } else {
        report.setActionExitCode(ExitCode.FAILURE);
    }
}
```

Tip #44: Controlling the Server's JVM

- How does `asadmin start-domain` determine which JVM to use?
- The precedence is:
  1. domain.xml: use the java-home attribute in <java-config>.
  2. AS_JAVA is set in glassfish/config/asenv.{conf, bat}
  3. The JAVA_HOME environment variable is set.
  4. The java.home system property (from the JVM executing asadmin)

- How do we validate the path? We look for {TARGET-PATH}/bin/java[.exe]

Tip #45: Controlling the JVM Options

- The JVM options used to start the server are maintained in the domain.xml.
- See the <java-config> element.   But which one?
  - default-config, server-config, cluster1-config???
- Use list-jvm-options --target <server, cluster-name, instance-name>
- Use create-jvm-options --target <target> <JVM option> to add options.
- Use delete-jvm-options --target <target> <JVM option> to delete options.
- Adding Java system properties does not require a server restart.

Tip #45: Controlling the JVM Options

Example:

```
% asadmin create-jvm-options --target c1 --Xmx1024m
Created 1 option(s)
Command create-jvm-options executed successfully.

% asadmin list-jvm-options
-XX:MaxPermSize=192m
-XX:PermSize=64m
-client
-Djava.awt.headless=true
…
```

Tip #46: Upgrading to a Newer Version

- Upgrade applications and configuration from an earlier version
- Upgrade Tool: Side-by-side upgrade
  - GUI: `asupgrade`
  - CLI: `asupgrade --c`
  - What happens ?
    - Copies older source domain -> target domain directory
    - asadmin start-domain --upgrade

Tip #46: Upgrading to a Newer Version

- Update Tool and `pkg`: In-place upgrade
  - GUI: `updatetool`, install all Available Updates
  - CLI: `pkg image-update`
  - Upgrade the domain
    - `asadmin start-domain --upgrade`

Tip #47: Upgrade from Open Source to Commercial

- Use pkg or UpdateTool
- Remove repositories
  - pkg unset-publisher **dev.glassfish.org stable.glassfish.org release.javaeesdk.oracle.com**
- Add repositories for Oracle GlassFish Server
  - pkg set-publisher -P -O http://pkg.oracle.com/glassfish/v3/release **release.glassfish.oracle.com**
  - pkg set-publisher –O http://pkg.oracle.com/glassfish/v3/dev/ **dev.glassfish.oracle.com**

Tip #47: Upgrade from Open Source to Commercial

- Install add-on components
  - `pkg install glassfish-enterprise-web-profile`
  - `pkg install glassfish-enterprise-full-profile`

Tip #48: Extending and Updating Inside a Closed Network

- Server without an Internet connection or isolated from other networks
- How ?

1. Install the pre-installed toolkit image (~4MB) + starter repository (~20MB) inside closed network
2. Download repository from support.oracle.com
   - `Unzip 145095-01.zip`
   - `Unzip ogs-3.1.1-repo-mac-i386.zip`
3. Start local repository daemon
   - `./pkg.depotd –d <DIR> -p <PORT>`

Tip #48: Extending and Updating Inside a Closed Network

4. Configure GlassFish Server to use Local Repository
   - `pkg -R <GlassFish> set-publisher -Pe -O http://<repo-host>:<port> <publisher>`

Tip #49: GlassFish Docs

glassfish.org/docs

- Oracle GlassFish Server 3.1.2.2
  - http://docs.oracle.com/cd/E26576_01/index.htm
    - Getting Started (4)
    - Installing and Upgrading (2)
    - Administering and Deploying Applications (5)
    - Troubleshooting (2)
    - Scaling and Tuning the Performance (2)
    - Developing Applications (6)
    - Extending and Embedding (2)
    - Reference (2)

Tip #49: GlassFish Docs

glassfish.org/docs

- GlassFish Server Open Source Edition
  - http://download.java.net/glassfish/3.1.2/release/glassfish-ose-3.1.2-docs-pdf.zip

Tip #50: How to reach us?

- GlassFish Forum: http://www.java.net/forums/glassfish/glassfish
- users@glassfish.java.net
- @glassfish
- facebook.com/glassfish
- youtube.com/GlassFishVideos
- blogs.oracle.com/theaquarium

Thanks!

http://wikis.oracle.com/display/GlassFish/50+Tips

MAKE THE FUTURE JAVA

JavaOne™

ORACLE®

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.

JavaOne™