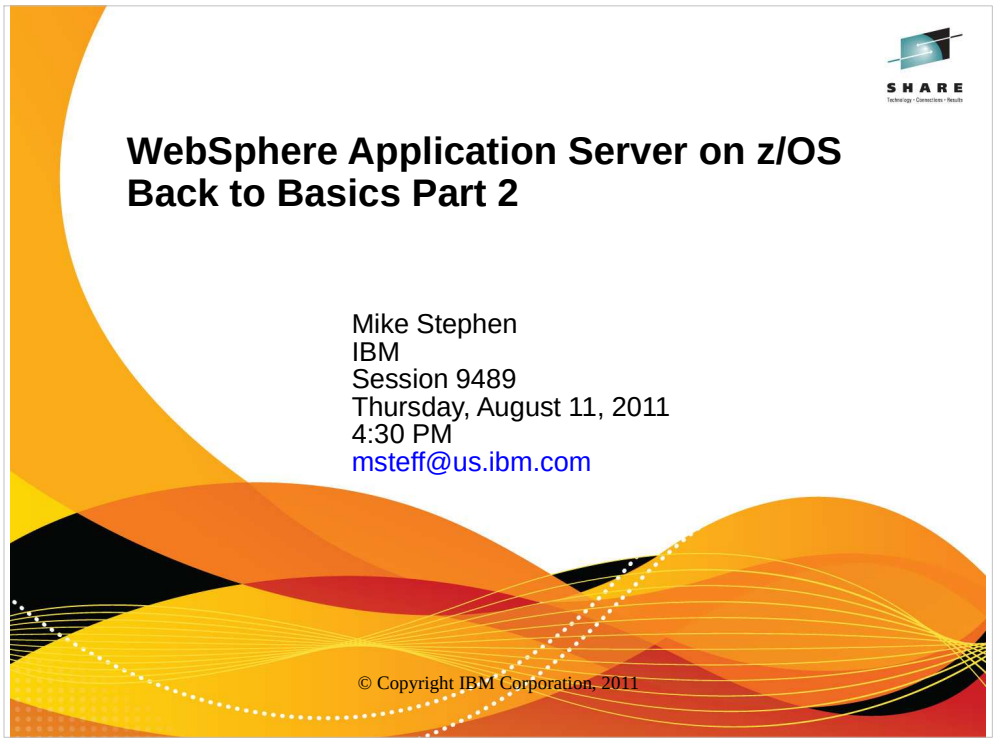


This is part 2 of 2.

WebSphere Application Server Sessions

Day	Time	#	Title	Speaker	Room
Wednesday	3:00	9483	Using IBM's New Cross-Platform Installer on z/OS	Mierzejewski	Oceanic 5
Thursday	8:00	9482	WAS Version 8 – Overview	Follis	Europe 2
Thursday	9:30	9486	WAS Version 8 – Batch Update	Hutchinson	Europe 2
Thursday	11:00	9485	WAS Version 8 – New z/OS Exploitation/Differentiation Features	Follis	Europe 2
Thursday	1:30	9484	WAS Version 8 – High Availability Enhancements	Follis	Europe 2
Thursday	3:00	9488	WAS - Back to Basics Part 1	Loos	Europe 2
Thursday	4:30	9489	WAS - Back to Basics Part 2	Stephen	Europe 2
Friday	8:00	9490	WAS for z/OS - Level 2 Update	Stephen	Europe 2
Friday	9:30	9487	WAS for z/OS – PotPourri	Follis, Hutchinson, Loos, Mierzejewski, Stephen, etc.	Europe 2

Part 1...

- Introduction.
- Adding an Additional Node.
- Creating a Server.
- Creating to a Cluster.
- Some Useful Stuff (Config tweaking).
  - Setting some variables.
  - Setting console preferences.
  - Use of the RMI connector.
  - Etc...
- Q & A?

This presentation is the accumulation of experiences installing WebSphere on z/OS at customer locations across the US in combination with those acquired working with the folks from the WSC, with input from WebSphere on z/OS level 2 support.

Adding an additional Managed Node

- Using the Configuration spreadsheet.
- Using the zPMT (WCT).
- Running the generated jobs.

Slide 4

Creating the additional Managed Node involves basically three major steps, each with a few “substeps”.

First (if it is being used) the information from the configuration spreadsheet must be extracted.

Then the zPMT (WCT) must be run and the output (generated jobs) must be uploaded to the target system.

Then the generated jobs need to be run in accordance with the generated instructions.

Creating an additional Managed Node

- Using the spreadsheet.
  - s8cell.ods
- Using the zPMT.
  - Start the WCT.
  - Etc…

Slide 5

The spreadsheet provides a starting point for the PMT, i.e. a response file so that you don't have to enter all values manually.

When starting the WCT, you'll need to choose a "location" for your definitions. Use a path that is indicative of what you are doing. A possible suggestion would be: drive:\WCT Workspace*cellname.* Then when you create your customization definitions for that location, you can choose names for them that further define what you are doing: dmgr, emptynode, etc.

The order in which you create the jobs with the PMT is not particularly important, but it seems to work well to start with the Deployment Manager, then create an Empty Managed node, then to add servers either with scripting or the AdminConsole.

Since we are working with an existing cell and adding an additional empty node, we'll be adding a definition to an existing location.

The Location used to create the jobs for the cell (s8cell) used for this example were:

Location Name: s8Cell
Location Path: D:\WAS V8 WCT Workspaces\s8cell

The customization definitions that were created previously were:

Deployment Manger: s8dmgr
1st Empty Managed Node: s8emptyc

Now we'll be adding:

2nd Empty Managed Node: s8emptyd

Creating an additional Managed Node

- For this section of the PMT (Empty Managed Node):
  - Load the correct “response file”.
  - Step through the panels.
    - Make any changes that seem necessary.
  - Create the jobs.
  - Upload the jobs.

If you are using the zPMT version of the spreadsheet shown earlier, you will have created “response files”. For each section (dmgr, empty managed node), you will need to specify the created response file in the zPMT.

Using "Simplified RACF definitions…

- Two Jobs in the .CNTL datasets run a REXX Exec in the .DATA datasets.
  - BBOSBRAK ➔ BBOSBRAC
  - BBOMBRAK ➔ BBOMBRAC
- The two REXX execs may be modified or replaced.
  - We used a set of modifications when this cell was originally created, illustrated here.
    - BBOMBRAK ➔ DORAC700
- Some documentation on the modifications…
  - Using Generic RACF Profiles with WebSphere on z/OS V7
  - http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101427

Slide 7

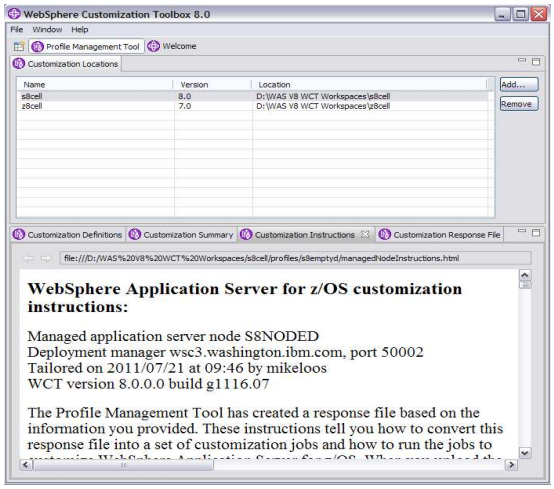
This all assumes the use of RACF for the security product. If you are an ACF2 or Top Secret installation, then you definitely need to make significant changes.

If you are NOT using simplified RACF definitions, then nothing needs changing.

There is an alternate template on the techdoc site that provides a full set of RACF definitions if you are using the naming convention supported by the spreadsheet and prefer the simplified generic RACF definitions. Simply copy the template into the .DATA dataset using a new member name, and modify it, either according to the comments at the beginning of the template or in conjunction with the edit macro provided as part of the same techdoc. Then modify the BBOMBRAK member of the .CNTL dataset to point to the new member of the .DATA dataset and run the job. There is no need to run any other RACF jobs after that point.

Creating an additional Managed Node

- Read the instructions.



- Generated Instructions.

Clicking on View allows you to look at or print the generated instructions.

Creating an additional Managed Node

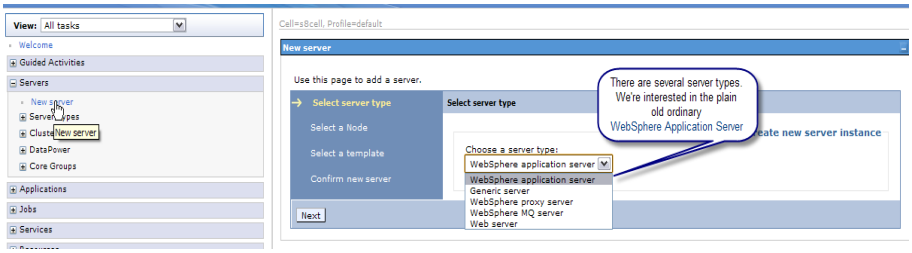
- Running the generated JOBs.
  - BBOMCFS
  - BBOMHFSA
  - BBOWWPFM
  - BBOMPROC
  - BBOWMNAN
- Start the nodeagent.

Note the lack of any security related jobs…

We used the generic RACF definitions described in the white paper and everything was defined when the cell was first created so nothing needs to be done at this time.

Creating a new server

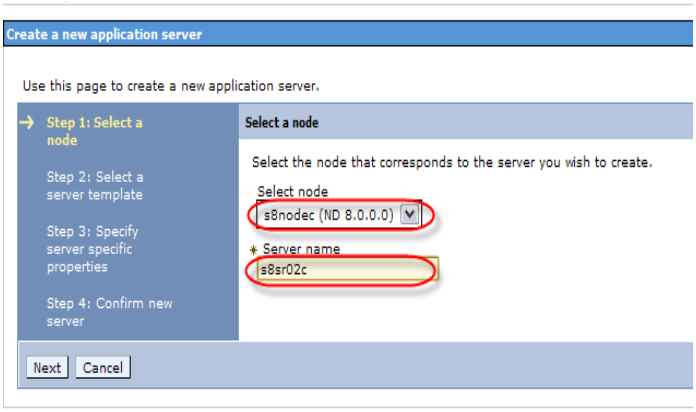
- Create a new server…



To create a new server.

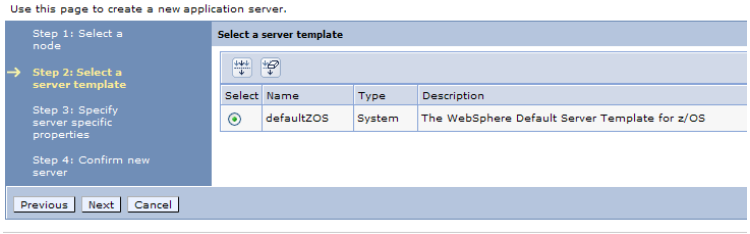
Click on Servers >> New Server, select the server type (the default of WebSphere Application Server is what we want for this exercise), and click on Next.

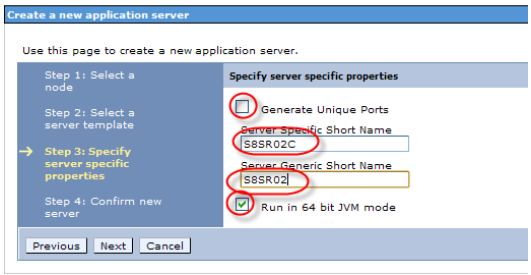
Creating a new server



Select the proper node from the dropdown box (we only have one at this point so it is an easy decision) and specify the new server name. Then click on Next.

Creating a new server



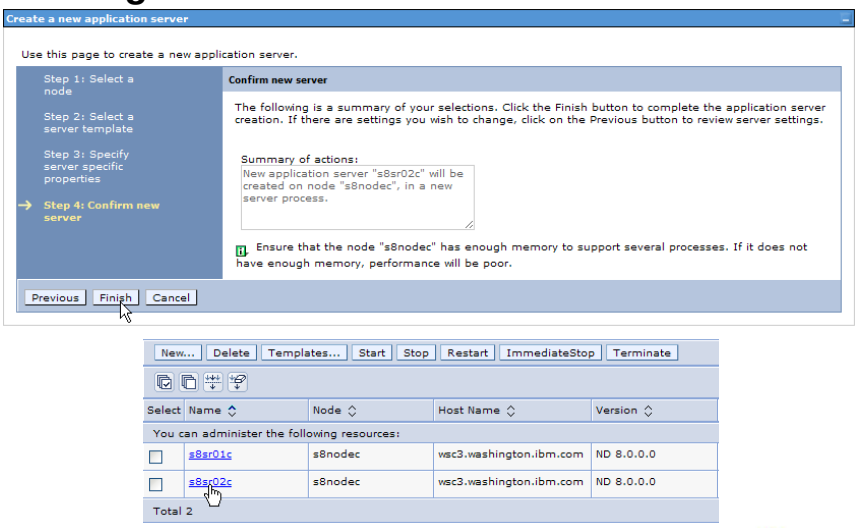


Slide 12

Select a template (the only one available at this point is the default template). Click Next.

Uncheck the “Generate Unique Ports” box, specify the Server Short names (Specific and Generic), and if you don’t want to run in 64 bit mode (which is the V7 default) uncheck the box. We’ll leave it checked. Click Next.

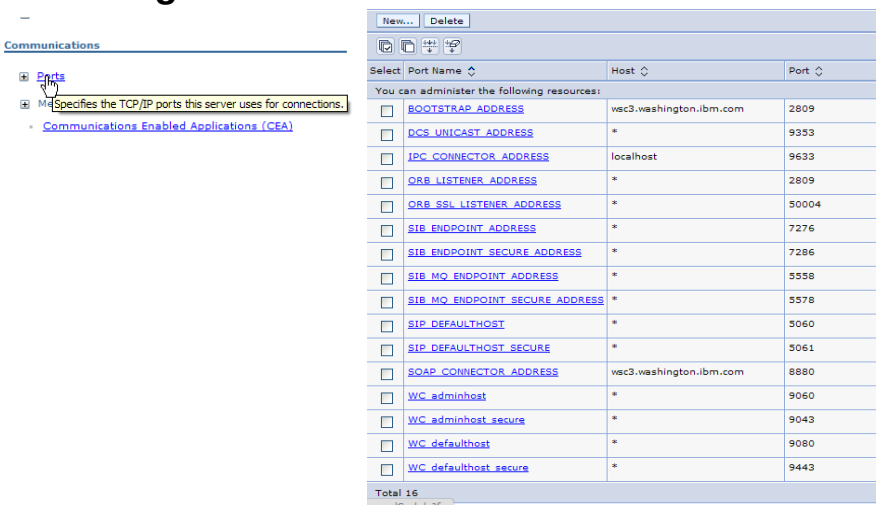
Creating a new server



You will next be presented with a confirmation panel. Click on Finish.

Then click on your new server.

Creating a new server



On the next screen, locate the section for Ports and click on Ports.

You will be presented with a screen listing all of the ports for the server. You now need to set each port to the proper number based on your numbering scheme.

You may also have to add some additional host alias entries to the default virtual host for the http ports.

You may then click on Save, and synchronize and your new server is ready to start.

Creating a new server

- As an alternative to setting the ports manually…
- A script!

  updNewServerv8.py

As an alternative to setting the ports manually, you may wish to create and run a script to set them all manually after the server had been saved and the configuration synchronized. The script shown uses the WSC Naming convention, but could easily be adapted to other naming and numbering conventions…

As an added benefit, it adds the necessary virtual host host alias entries as well.

Creating a new server

- And if you really want to avoid the adminconsole…

createNewServerv8.py

As an alternative to creating the server manually, you can run a script to do the entire process. The script shown requires that you tell it as arguments the servername, the nodename, and the originating (or low) port for the server. If you were using the WSC naming convention, the script would require no modification as it would properly set the specific and generic server short names and the port specified would be the soap port for the server. Again, if you are using a different naming convention the script should be fairly easy to modify to fit your needs.

Creating a new server

- Security updates.
  - Using Generics.
  - Using non-Generics.
    - Userids
      - *Controller.*
      - *Servant.*
      - *Adjunct (optional).*
    - STARTED class profile.
    - CBIND class profiles.
    - SERVER class profiles.
    - Keyrings and certificates.

Slide 17

Security profiles may need to be updated. If you have defined a generic set of RACF definitions which will cover these new servers, you may not need to change anything at this point.

In a cell where the RACF definitions are generic, it is common to use one USERID for all controllers and one for all servants and adjuncts. In this case the keyring and certificate that is defined for those USERIDs will suffice.

However, if you used the default RACF definitions, which are specific (not generic), as provided by the BBOWBRAK Rexx exec in the generated .DATA file, you will need to add the following:

New USERID for each of the address spaces (controller, servant, and optionally adjunct).

STARTED profile for the new server controller, servant, and optionally the adjunct regions

CBIND and SERVER classes profiles to include the new Cluster Transition Name and the new server

Keyrings for the controller and servant (and possibly adjunct) regions, and new certificate for the controller (and possibly adjunct), and connect the CERTAUTH certificate to the keyrings.

Creating a cluster

- What is a cluster?

Artifact	Name	Comment
Cell	s8cell	Existing
Node on SYSC	s8nodec	Existing
Node on SYSD	s8noded	Existing, Federated, Empty
Cluster	s8sr02	To be created
Server on SYSC	s8sr02c	Existing
Server on SYSD	s8sr02d	To be created

Slide 18

A cluster, in its most simple form is a server. And every server is a cluster. But a cluster composed of a single server isn't of much interest. Usually when discussing a cluster, we are talking about a cluster containing two or more servers, generally in more than one node.

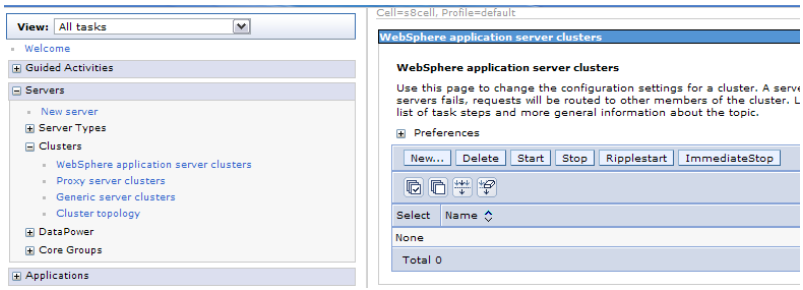
Creation of a cluster can start with zero, or one servers or templates. A cluster can never be created from more than one server.

Our starting point for this exercise is the table shown.

We have created so far in this presentation a multinode, mutisystem cell. The cell is the s8cell, and we have a node s8nodec on SYSC and a node s8noded on SYSD (which we have just created). The s8nodec node has a server s8sr02c (also recently created).

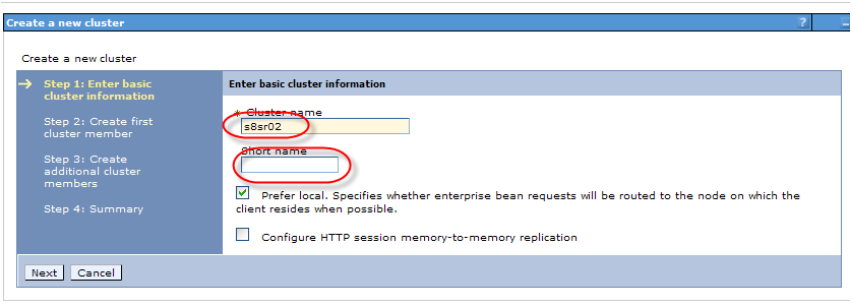
We will create a cluster s81sr02 that will consist of s8sr02c which will be converted into a cluster member, and we will add one additional member s8sr02d in node s8noded.

Creating a cluster



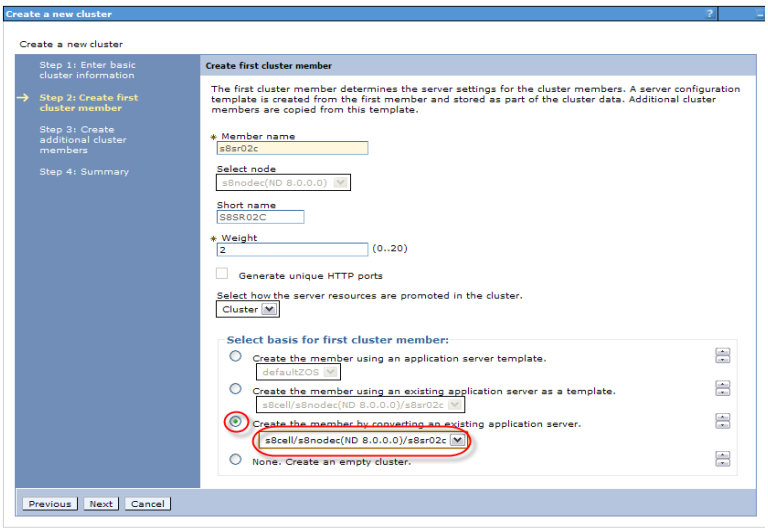
As always, we start by logging on to the adminconsole. Then we navigate to Servers >> Clusters >> WebSphere application server clusters and then click New.

Creating a cluster



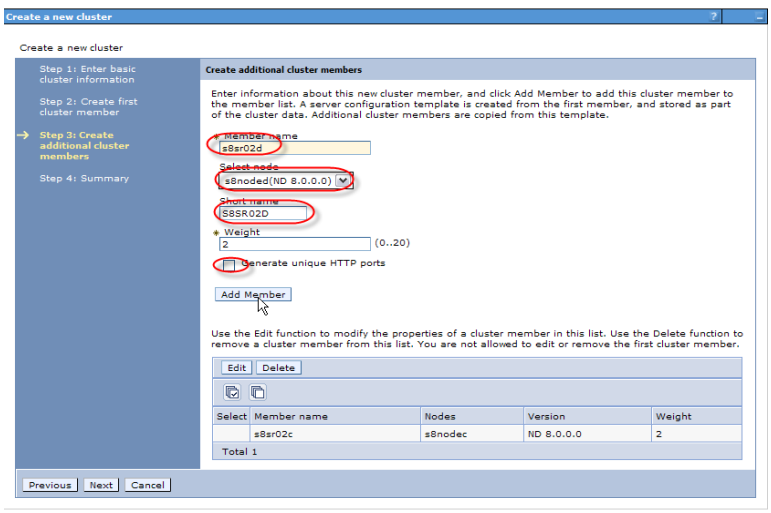
You'll see a screen similar to the one shown where you can fill in the values for Cluster name. We leave the cluster Short name blank. This will make more sense after the next foil. Then click Next.

Creating a cluster



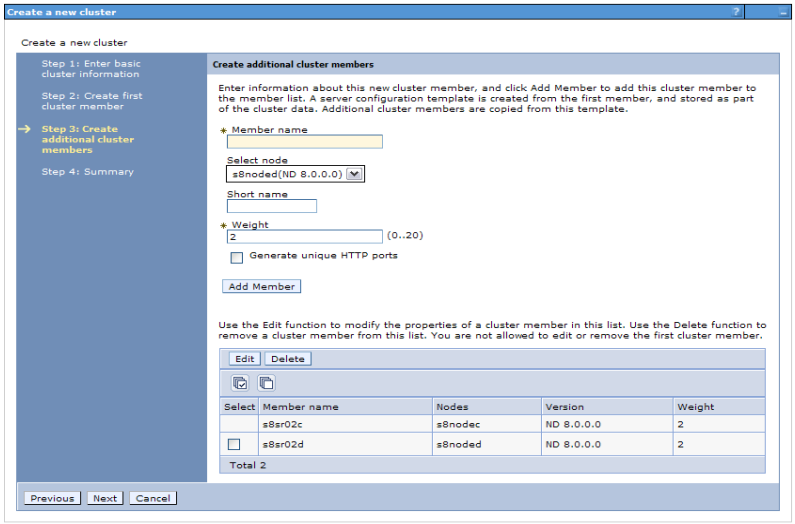
Now we're ready to create the first cluster member. We're going to go the route of creating the member by converting an existing application server. As soon as we select that option, many of the other fields "grey out". We select s8sr02c in node s8nodec from the drop down box after checking the radio button. The cluster we are creating will inherit it's short name (left blank on the previous screen) from the server we are converting to a cluster member. Then click Next.

Creating a cluster



This is the next screen you'll see. Fill in the member name of the new cluster member (the one that doesn't yet exist), s8sr02d, select the appropriate node from the drop down box, in this case s8noded, and fill in the short name, S8SR02D in our example. Uncheck the Generate unique http ports (in most cases). Then click on Add Member

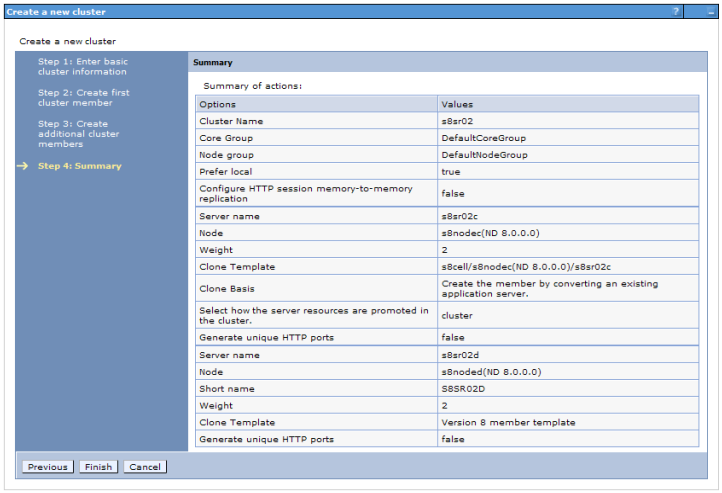
Creating a cluster



Slide 23

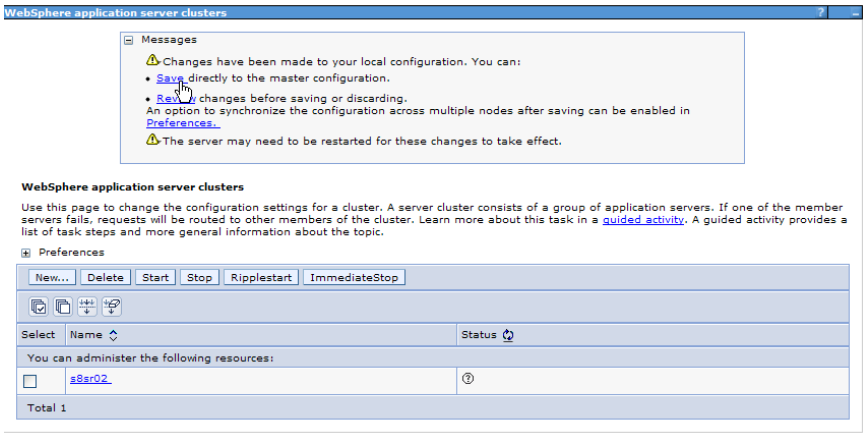
You'll be presented with a panel similar to this one. You'll notice that there is an additional cluster member in the table at the bottom of the panel. You now have an opportunity to add additional cluster members, simply by filling in the blanks as we did on the previous screen and clicking on add member. Two members is enough for our example, so we'll simply click on Next.

Creating a cluster



You'll see a summary page describing the to be created cluster. All you need to do is to click on Finish.

Creating a cluster



Simply Save and synchronize and you've created the cluster.

You may have to update the Host Aliases for the Virtual Host. If you have been using an asterisk(*) for the host name on the original server, this will not be necessary.

All that remains is to start the cluster members (servers) and verify that they start and run properly.

Creating a cluster

- A script?
- s8id = AdminConfig.getid('/Server:s8sr02c/')
- c8id = AdminConfig.convertToCluster(s8id, 's8sr02')
- AdminTask.createClusterMember('[-clusterName s8sr02 -memberConfig [-memberNode s8noded -memberName s8sr02d -memberWeight 2 -genUniquePorts false -specificShortName S8SR02D]]')
- wsadmin>AdminConfig.save()

If you'd rather create your cluster using a script, here are the commands (bare bones) that you'd need.

The first gets the id of the server you choose to convert to a cluster.

The second converts the server with long name s8sr02c to a cluster with cluster name of s8sr02 with a single member named s8sr02c

You're essentially done at this point, but since the main reason for a cluster is to have two or more members…

The third command will create a new cluster member in cluster s8sr02 with member name (long name) of s8sr02d on node s8noded with a shortname of S8SR02D. This command also leaves the ports alone. You'll have to do something about that later.

The last command saves the configuration.

Simple? Yes!

Now you should probably update the port assignments of the new cluster member, possibly using the previously mentioned script updNewServerv8.py.

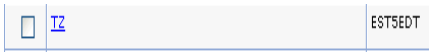
You may have to update the Host Aliases for the Virtual Host. If you have been using an asterisk(*) for the host name on the original server, this will not be necessary.

All that remains is to start the cluster members (servers) and verify that they start and run properly.

Configuration tweaking…

- Time zone(s).
  - Trace.

  - Application level.



Slide 27

The default for timestamps is for all of them to appear in GMT. Many installations would prefer to have the timestamps appear in local time. There are three variables which you may alter to change the behavior to match your desires. They are:

DAEMON_ras_time_local

ras_time_local

And **TZ**

DAEMON_ras_time_local and **ras_time_local** are set to **0** (GMT) by default. Actually the variables do not even exist. If you wish to change them, you have to add them and change their value to **1** (local).

The **TZ** variable also does not exist, and in its absence, all application time will be in GMT. This variable can be set to one of the standard timezone values, such as **CST6CDT** or **EST5EDT** or whatever is appropriate for your installation.

All of these variables are set by logging on to the adminconsole and navigating to the Environment >> WebSphere variables >> and adding them. I would suggest setting them at the cell scope and if you need to vary from that for a particular server you can add another instance at that server's level.

Then save the config and synchronize and the next time a component is started it will take effect.

Configuration tweaking…

- Message Routing.
  - Procs are “pre-loaded” with proper DDNAMES

```
000035 //*
000036 //* Output DDs
000037 //*
000038 //DEFALTDD   DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
000039 //HRDCPYDD   DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
000040 //SYSOUT     DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
000041 //CEEDUMP    DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
000042 //SYSPRINT   DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
```

  - Addition of environment variables.

☐	DAEMON_ras_default_msg_dd	DEFALTDD	Cell=s2cell
☐	DAEMON_ras_hardcopy_msg_dd	HRDCPYDD	Cell=s2cell
☐	ras_default_msg_dd	DEFALTDD	Cell=s2cell
☐	ras_hardcopy_msg_dd	HRDCPYDD	Cell=s2cell

Techdoc: TD103695
Managing Operator Message Routing in WebSphere for z/OS Servers
http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TD103695

Slide 28

As delivered, many messages are routed to the system log via a WTO. They also appear in the JESMSGLG and JESYSMSG DDs for the started tasks. The volume of messages on the syslog and coming across the console is annoying (to say the least). Fortunately it is a simple matter to change this behavior.

The routing of these messages is controlled by the following variables:

DAEMON_ras_default_msg_dd
DAEMON_ras_hardcopy_msg_dd
ras_default_msg_dd
ras_hardcopy_msg_dd

These variables may be set to the DDNAME that you wish to have them routed to and they will no longer be issued as a WTO. Again, the suggestion is to set them at the cell level and then if you need to change them on a finer grained level you can do so by adding a copy of the variable at a “lower” scope.

The variables are set by logging on to the adminconsole and navigating to the Environment >> WebSphere variables >> and adding them.

Then save, synchronize, and as each component is restarted, the change will take effect.

The techdoc describes additional controls that may be added that apply to JES2 systems.

Configuration tweaking…

- An alternative to all of the editing…
- A script!

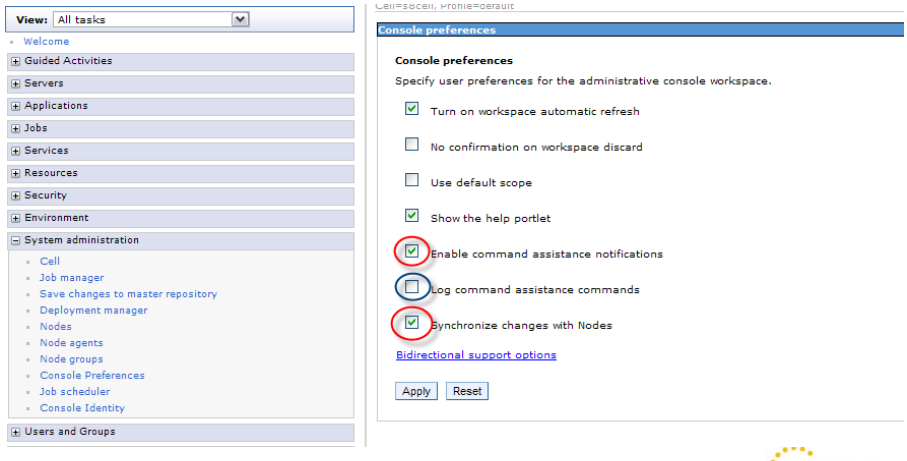
initsetvarsv8.py

Script is available at the following URL:
http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100963

If you are adverse to doing all of that manual editing, or if you create a lot of cells and wish to make them all look alike, a script might be a good alternative to doing all of the manual edits. The script I am showing, **initsetvarsv8.py** is a jython script which does all of the changes we made in the adminconsole (with the exception of the console preferences, since they are “per userid”).

Configuration tweaking…

- Console preferences.



One of the first things you may want to do when you first log on to the adminconsole is set your default console preferences. These tend to be individual preferences and I'm showing how I like them set. Your selections may of course vary…

You set them by logging on to the adminconsole, and navigating to: Systems administration >> Console Preferences

I suggest setting the Synchronize changes with Nodes on, and also like to have the command assistance notifications enabled.

You may also consider turning on the Log command assistance commands option.

Use of RMI Connector with wsadmin.sh

- wsadmin.sh uses either a SOAP connection or an RMI connection.
- The SOAP connection necessitates the passing of a clear text userid and password.
- The RMI connection can either prompt or use the userid and password of the logged on user.
- The RMI connection has been “observed” to be a little quicker than the soap connection (anecdotal).

Use of RMI Connector with wsadmin.sh

- To switch from the SOAP (default) connection to RMI…
- From somewhere that supports editing ASCII encoded files.
  - SSH, Telnet, etc.

```
cd /wasv8config/s8cell/s8mnode/DeploymentManager/profiles/default/properties
```

Use of RMI Connector with wsadmin.sh

Use an ASCII editor to change:

sas.client.props

From:

```
com.ibm.CORBA.securityServerHost=
com.ibm.CORBA.securityServerPort=
com.ibm.CORBA.loginSource=prompt
```

To:

```
com.ibm.CORBA.securityServerHost=wsc3.washington.ibm.com
com.ibm.CORBA.securityServerPort=50003
com.ibm.CORBA.loginSource=none
```

Use of RMI Connector with wsadmin.sh

Optionally, (these could be specified on the wsadmin.sh command line as parameters) use an ASCII editor to change:

wsadmin.properties

From:

```
com.ibm.ws.scripting.connectionType=SOAP
#com.ibm.ws.scripting.connectionType=RMI
com.ibm.ws.scripting.port=15610       /* from SOAP port */
com.ibm.ws.scripting.defaultLang=jacl
```

To:

```
#com.ibm.ws.scripting.connectionType=SOAP
com.ibm.ws.scripting.connectionType=RMI
com.ibm.ws.scripting.port=50003      /* to ORB port */
com.ibm.ws.scripting.defaultLang=jython
```

Use of RMI Connector with wsadmin.sh

- Now you can:

```
cd /wasv8config/s8cell/s8dmnode/DeploymentManager/profiles/default/bin
./wsadmin.sh
```

If you have any questions, now is the time...