

CHAPTER 14: HIGH AVAILABILITY

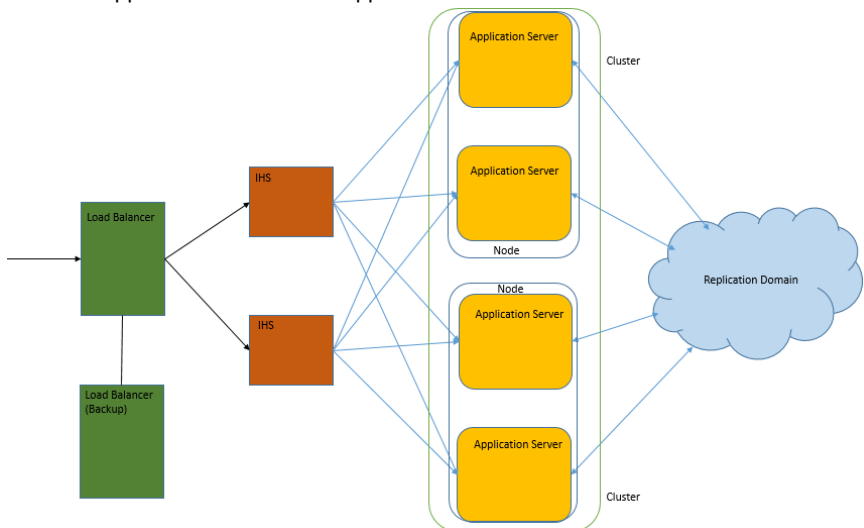
Theory

High Availability is the ability to tolerate a certain amount of failures without any interruption to the service. It is not a consideration point only for WebSphere Application Server and it can be achieved by only having more redundancy in your architecture. WebSphere Application Server Network Deployment gives you different options to have a highly available environment for your business critical applications.

In order to avoid single point of failure (SPOF), WebSphere Application Server provides vertical and horizontal scaling across different systems. Adding multiple load balancers and web servers can increase the redundancy of the environment.

In a WebSphere Application Server environment, there are multiple areas that are critical to data availability:

- Databases: For read-only data, having multiple copies of synchronized databases behind a load balancer can be a solution. For read/write data access, it can be better to have a hardware cluster for the database node.
- HTTP session state: The session manager creates HTTP sessions and manages the life cycles of HTTP sessions that are associated with the application.
- EJB session state: EJB session state caching that can be replicated across a cluster and persisted to a variety of data stores for failover purposes.
- EJB persistence: It enables to monitor dynamically of deployed applications to add JPA support.



In WebSphere Application Server high availability manager (HA Manager) provides singleton processes to have process high availability. A singleton process can exist in only one location at any given instance, or multiple instances of this function operate independently of one another.

A core groups is the high available part of a WebSphere Application Server cell. Each HA Manager creates connectivity with all the other HA Manager instances in the same core group. HA Manager periodically runs number of tasks in background to provide following services:

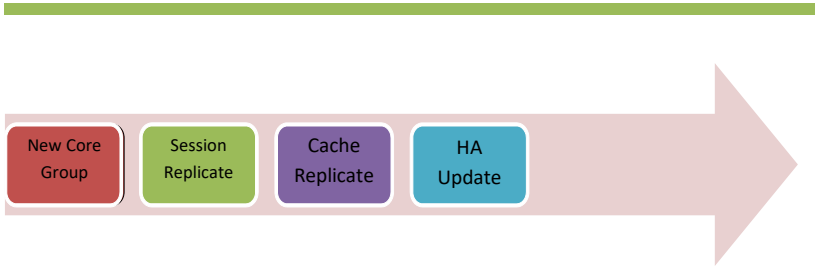
- Memory-to-memory replication is provided by Data Replication Service (DRS) which is part of WebSphere Application Server. It is used to replicate HTTP session data, EJB stateful session, and dynamic cache data among cluster members by using the transport channels to pass information between cluster members.
- Singleton failover is a cluster based service that includes transaction managers for cluster members and the default messaging provider, service integration bus.
- Workload management routing has following types:
 - Routing of the default messaging bus. (SIB)
 - Routing of HTTP requests through WebSphere Application Server proxy server
 - Routing of Web Services Addressing requests through WebSphere Application Server proxy server
 - Routing of Session Initiation Protocol requests
- On-demand configuration routing is used for WebSphere Application Server proxy server routing.

AIM

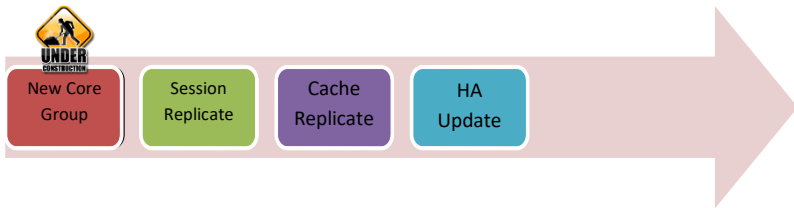
In this lab exercise, you will be able to configure high availability and fail over capabilities of WebSphere Application Server. In order to achieve this goal, you need to complete following tasks:

- Create a new core group
- Configure session replication
- Configure cache replication
- Configure high availability application update

Lab Exercise 14: HIGH AVAILABILITY



1. **Create a new core group**
2. **Configure session replication**
3. **Configure cache replication**
4. **Configure high availability application update**



Task 1: Create a new core group

Step 1: Navigate to “Servers>Core Groups>Core Group Settings” and click “New”.

The screenshot shows the WebSphere Integrated Solutions console interface. The left-hand navigation pane has 'Servers' and 'Core Groups' highlighted. The main content area is titled 'Core Groups' and contains the following information:

Core Groups
Use this page to define a core group. A core group is a grouping of WebSphere(R) Application Server cell processes. A core group can contain stand-alone servers, cluster members, node agents, and the deployment manager. A core group must contain at least one node agent or the deployment manager. A core group must be empty before it can be deleted. Connected core groups are core groups that can communicate with each other. Access point groups must be defined to establish communication between core groups.

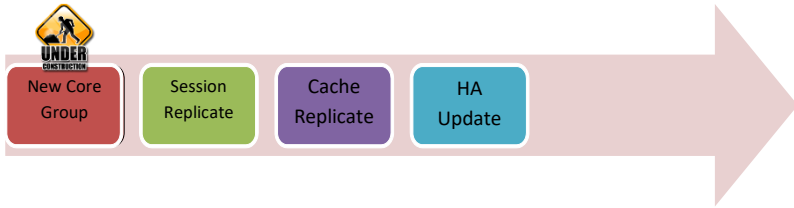
Preferences
New Delete

You can administer the following resources:

Select	Name	Description	Connected core groups
<input type="checkbox"/>	DefaultCoreGroup	Default Core Group. The default core group cannot be deleted.	

Total 1

Help
Field help: For field help information, select a field label or list marker when the help cursor is displayed.
Page help: More information about this page.
Command Assistance: View administrative scanning command for last action.



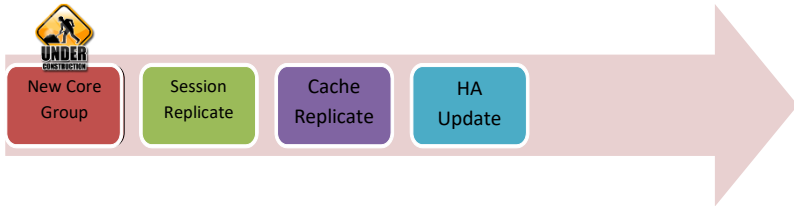
Step 2: Enter “WAS_CoreGroup” as name and click “OK”.

The screenshot shows the 'Core Groups > New...' configuration page in the WebSphere Integration Developer console. The 'General Properties' section is highlighted with a red box, and the 'Name' field is set to 'WAS_CoreGroup'. The 'Description' field contains 'A non-default core group'. The 'Number of coordinators' is set to '1', and the 'Transport memory size' is set to '100 megabytes'. The 'Transport type' is set to 'Channel framework', and the 'Transport chain' is set to 'DCS'. The 'OK' button is highlighted with a red box.

Step 3: Click “Save” to write changes to the master configuration.

The screenshot shows the 'Core Groups' page in the WebSphere Integration Developer console. A message box is displayed, indicating that changes have been made to the local configuration and can be saved directly to the master configuration. The 'Save' button is highlighted with a red box. Below the message box, the 'Core Groups' section shows a table of core groups. The table has columns for 'Name', 'Description', and 'Connected core groups'. The 'DefaultCoreGroup' is listed as the default core group, and 'WAS_CoreGroup' is listed as a non-default core group.

Select	Name	Description	Connected core groups
<input type="checkbox"/>	DefaultCoreGroup	Default Core Group. The default core group cannot be deleted.	WAS_CoreGroup
<input type="checkbox"/>	WAS_CoreGroup	A non-default core group	DefaultCoreGroup

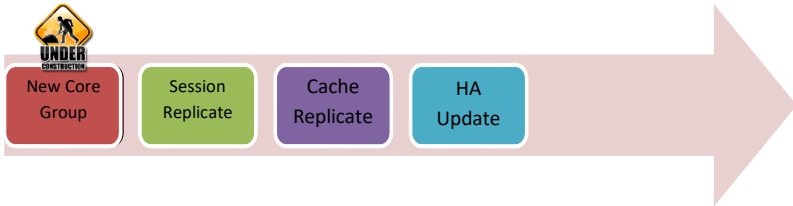


Step 4: Click on “DefaultCoreGroup”.

The screenshot shows the WebSphere Integrated Solutions console. The left sidebar contains a navigation tree with categories like Welcome, Servers, Applications, Resources, and Monitoring and Tuning. The main content area is titled 'Core Groups' and includes a description of core groups. Below the description is a table with columns for 'Name', 'Description', and 'Connected core groups'. The table lists two core groups: 'DefaultCoreGroup' and 'WAS_CoreGroup'. The 'DefaultCoreGroup' row is highlighted with a red box. The 'WAS_CoreGroup' row is also highlighted with a red box. The table shows that 'DefaultCoreGroup' is connected to 'WAS_CoreGroup' and vice versa.

Step 5: Click “Core group servers” under “Additional Properties”.

The screenshot shows the WebSphere Integrated Solutions console. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Core Groups > DefaultCoreGroup'. It has three tabs: 'Runtime', 'Configuration', and 'Operations'. The 'Configuration' tab is selected. Under the 'Configuration' tab, there are two sections: 'General Properties' and 'Additional Properties'. The 'Additional Properties' section is highlighted with a red box. Under 'Additional Properties', there is a link labeled 'Core group servers' which is also highlighted with a red box. Other links in this section include 'Properties and failure detection', 'Policies', 'Preferred coordinator servers', and 'Custom properties'. The 'General Properties' section shows fields for 'Name' (DefaultCoreGroup), 'Description' (Default Core Group. The default core group cannot be deleted.), 'Number of coordinators' (1), and 'Transport memory size' (100 megabytes). The 'Transport type' section shows 'Channel Framework' and 'Transport chain'.



Step 6: Select application servers to move to the new core group and click “Move”.

The screenshot shows the 'Core Groups' page in the WebSphere software console. The 'Move' button is highlighted with a red box. Below it, a table lists application servers. The first two rows, 'App_Server01' and 'App_Server02', are selected with checkboxes. The table has columns for 'Select', 'Name', 'Node', 'Version', 'Type', and 'Cluster Name'.

Select	Name	Node	Version	Type	Cluster Name
<input checked="" type="checkbox"/>	App_Server01	wasv90Node01	ND 9.0.0.0	Application Server	WAS_CLUSTER
<input checked="" type="checkbox"/>	App_Server02	wasv90Node02	ND 9.0.0.0	Application Server	WAS_CLUSTER
<input type="checkbox"/>	CDR	wasv90Node02	ND 9.0.0.0	Custom (ONDEMAND, ROUTER)	

Step 7: Select the target core group under “To core group” and click “OK”.

The screenshot shows the 'Move' configuration page in the WebSphere software console. The 'To core group' dropdown menu is highlighted with a red box, showing 'WAS_CoreGroup' selected. The page includes a 'Configuration' section with 'General Properties' and a 'Move selected servers' section.

Configuration

General Properties

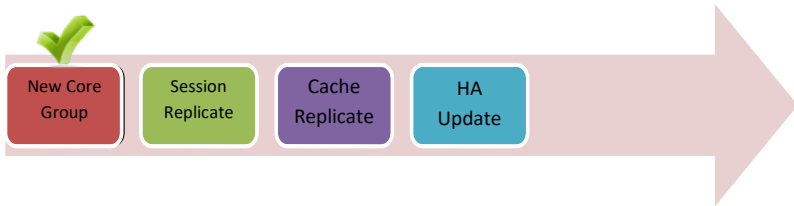
Move selected servers

Cluster name WAS_CLUSTER App_Server01 App_Server02

From core group
DefaultCoreGroup

To core group
WAS_CoreGroup

App OK Reset Cancel



Step 8: Click “Save” to write changes to the master configuration file.

Messages

The servers were successfully moved to core group WAS_CoreGroup.
 Changes have been made to your local configuration. You can:

- Save directly to the master configuration.
- Review changes before saving or discarding.

An option to synchronize the configuration across multiple nodes after saving can be enabled in [Preferences](#).
 The server may need to be restarted for these changes to take effect.

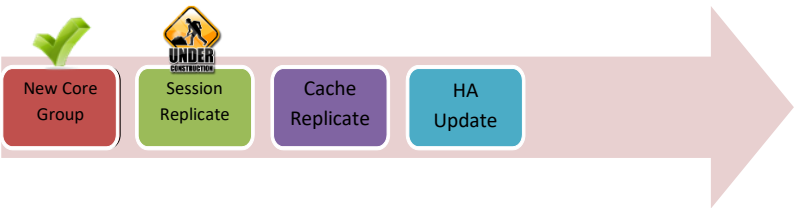
Core Groups - DefaultCoreGroup - Core group servers

Use this page to view and manage the servers that belong to a core group. A core group server can be an application server, a deployment manager, or a node agent that is a member of a high availability core group.

Preferences

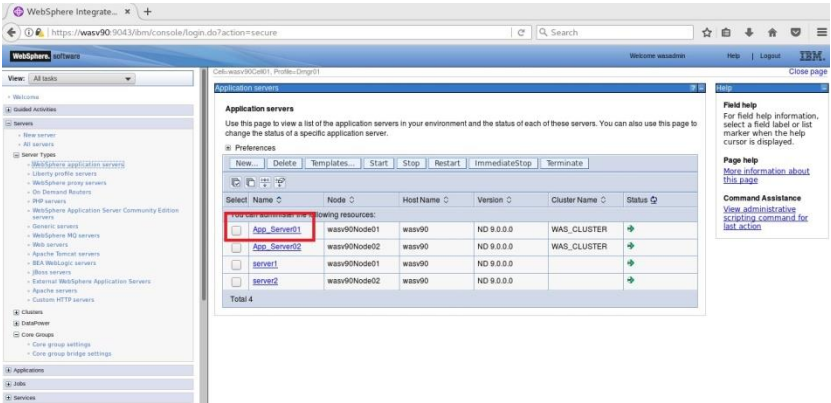
Select	Name	Node	Version	Type	Cluster Name
<input type="checkbox"/>	CDR	wasv90Node02	ND 9.0.0.0	Custom (ONDEMAND_ROUTER)	

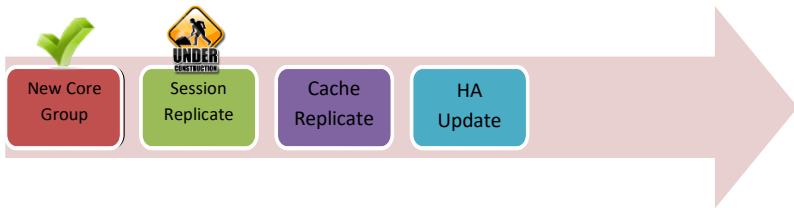
Task 1 is complete!



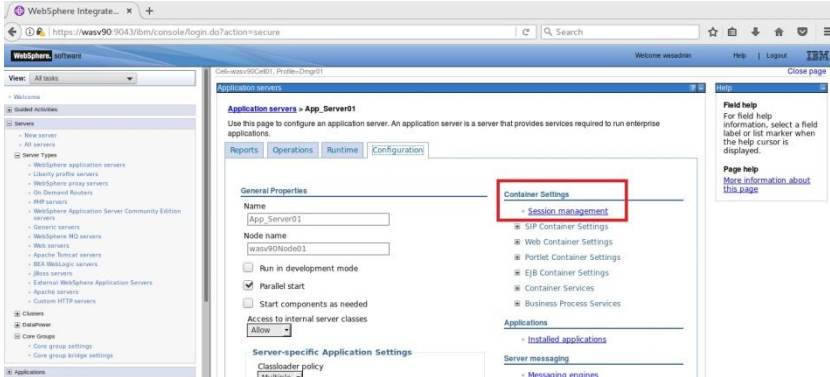
Task 2: Configure session replication

Step 1: Navigate to “Servers>Server Types>WebSphere application servers” and click on the application server name “App_Server01”.

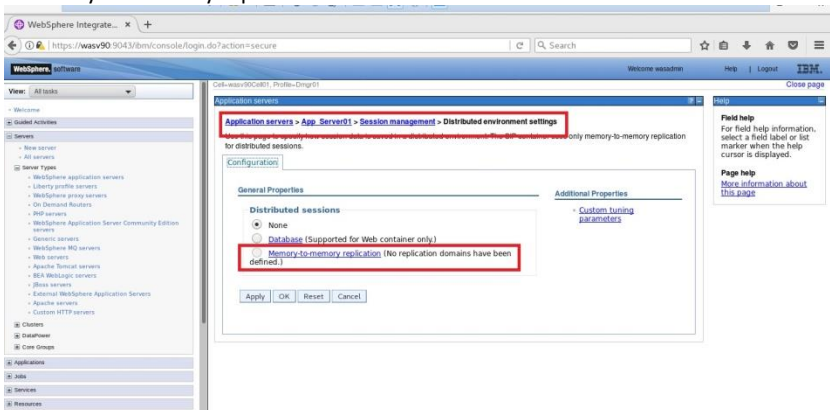


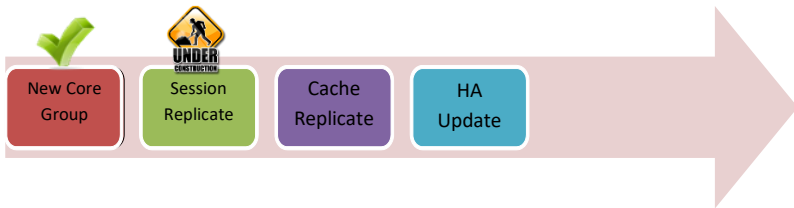


Step 2: Click on “Session management” under “Container Settings”.

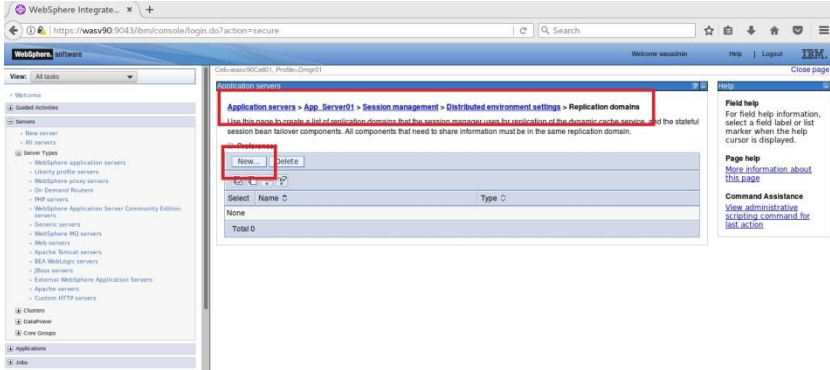


Step 3: Under Additional Properties, Distributed Environment Setting. Then click on “Memory-to-memory replication”.

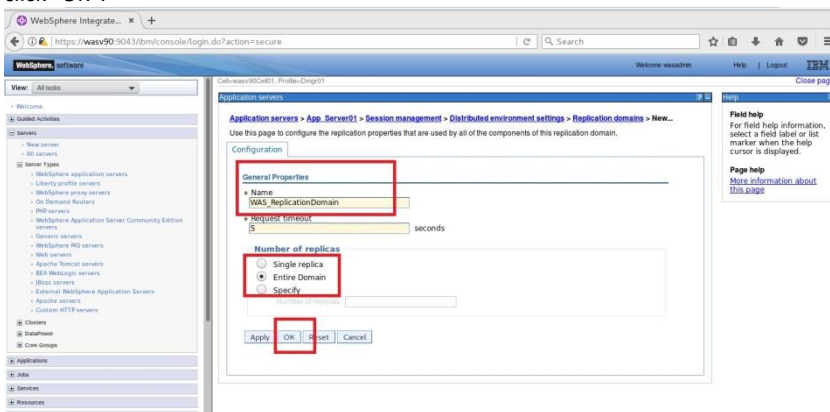


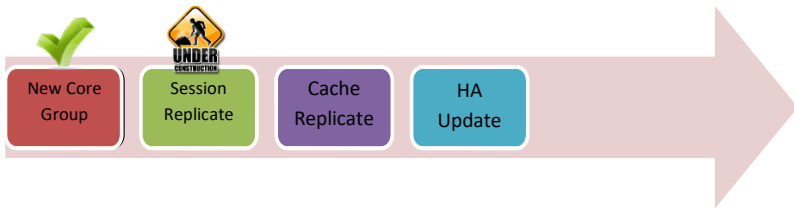


Step 4: Since we don't have any replication domain, click "New" to create one.

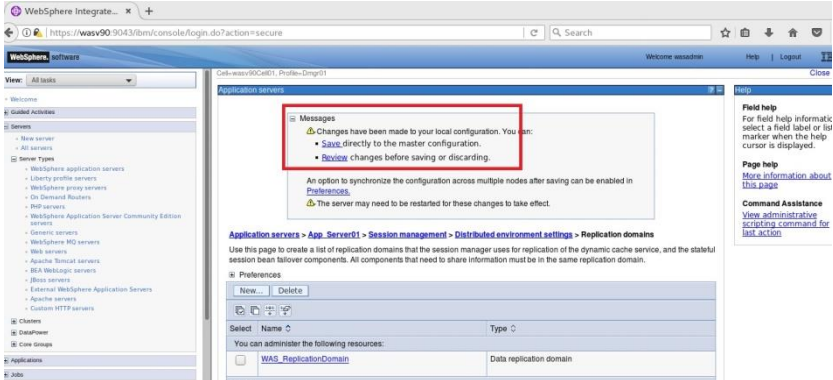


Step 5: Enter "WAS_ReplicationDomain" as name and select "Entire Domain" then click "OK".

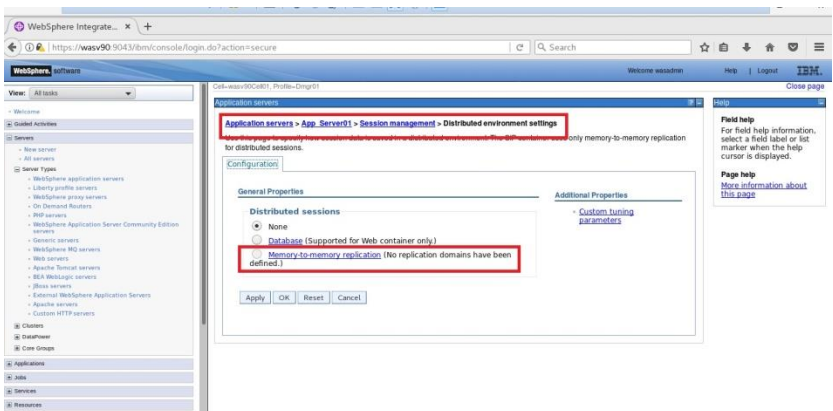


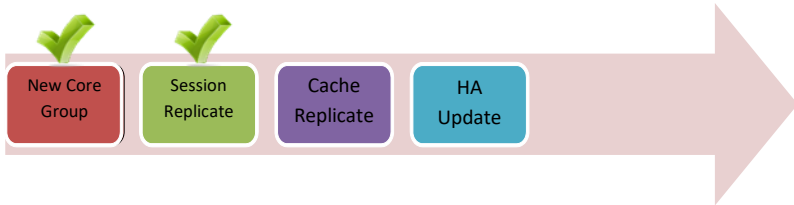


Step 6: Click “Save” to write changes.

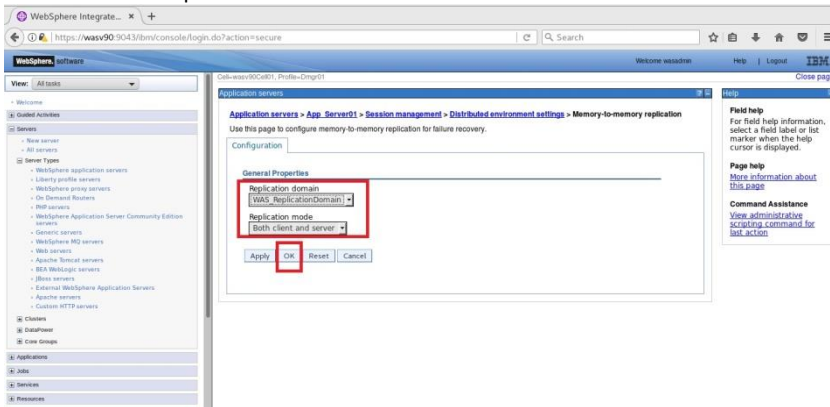


Step 7: Click on “Memory-to-memory replication”.

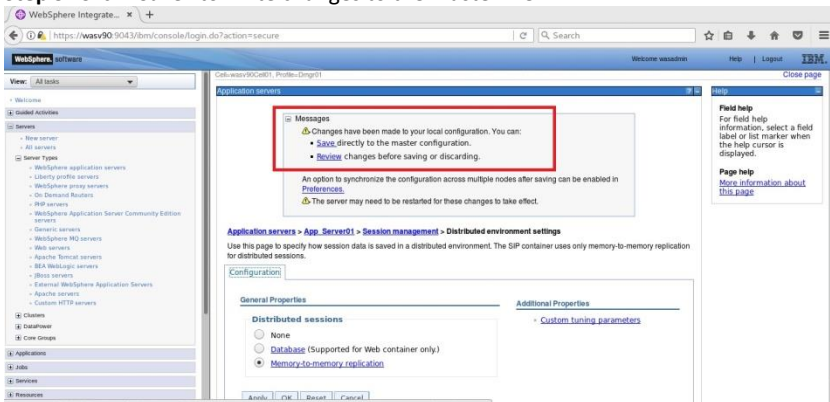




Step 8: Select “WAS_ReplicationDomain” as “Replication domain” and “Both client and server” as “Replication mode” and click “OK”.

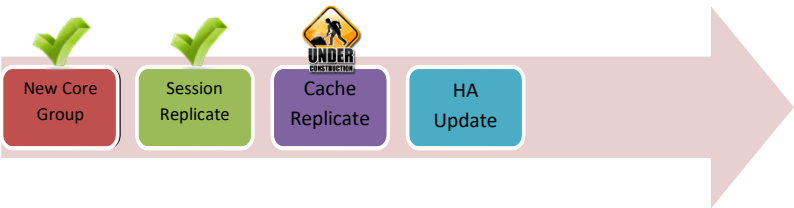


Step 9: Click “Save” to write changes to the master file.



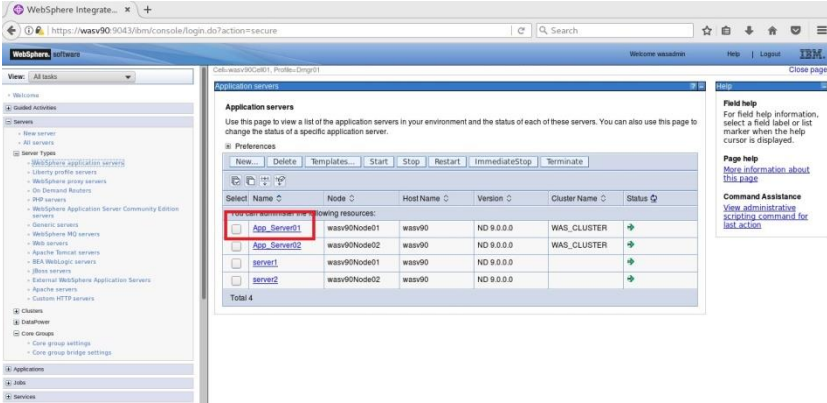
Step 10: Repeat the steps 7 to 9 for the application server “App_Server02”.

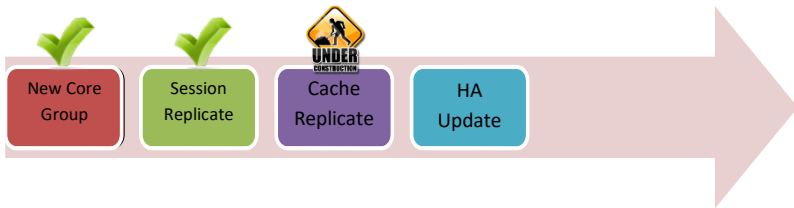
Task 2 is complete!



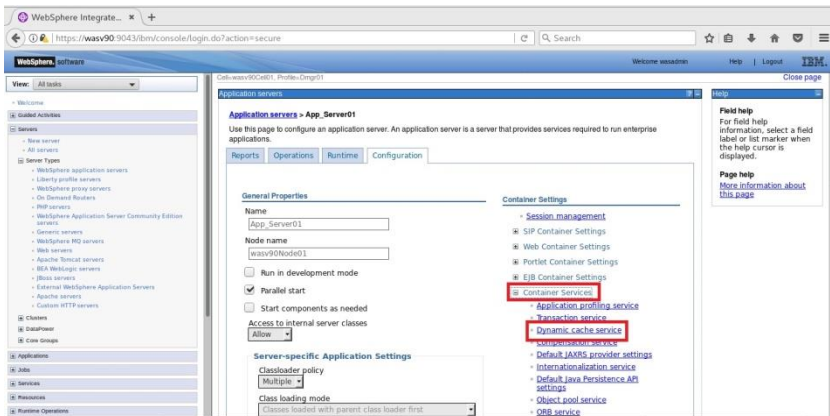
Task 3: Configure cache replication

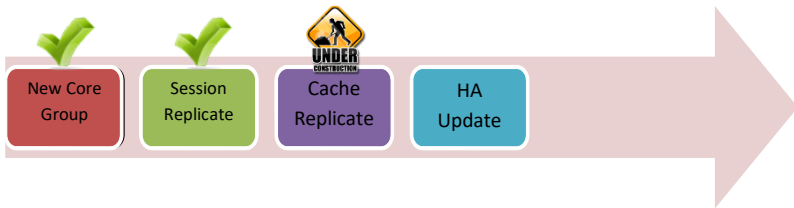
Step 1: Navigate to “Servers>Server Types>WebSphere application servers” and click on the application server name “App_Server01”.



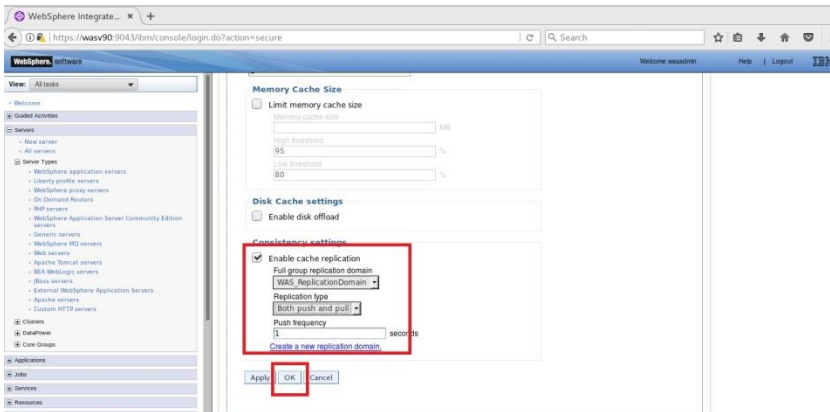


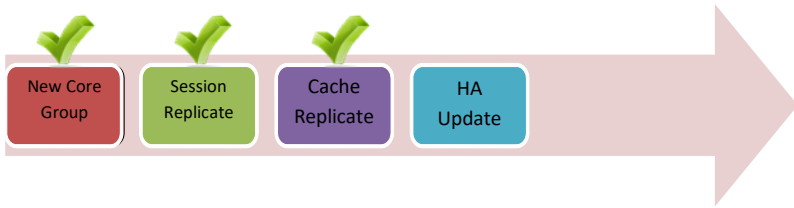
Step 2: Click “Dynamic cache service” under “Container Settings>Container Services”.



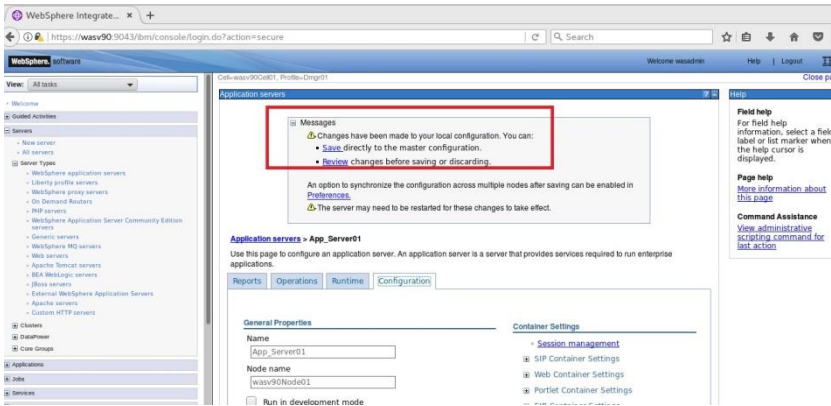


Step 3: Check “Enable cache replication” and select “WAS_ReplicationDomain” for “Full group replication name”, “Both push and pull” as “Replication type” and “1” for “Push frequency” then click “OK”.



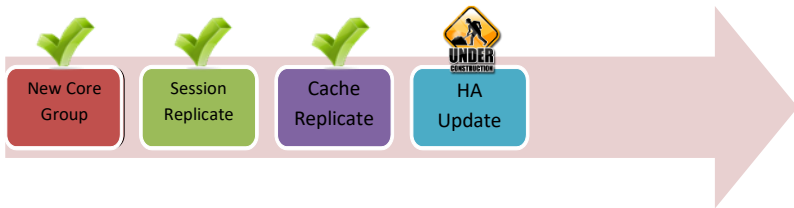


Step 4: Click “Save” to write changes directly to the master configuration file.



Step 5: Repeat the steps 1 to 4 for the application server “App_Server02”.

Task 3 is complete!



Task 4: Configure High Availability Application Update

Step 1: Navigate to “System administration>Node agents” and click on “nodeagent” for the first node.

Node agents

Use this page to manage node agents and application servers on the node that a node agent manages. The node agent process serves as an intermediary between the application servers on the node and the deployment manager. The node agent process runs on every node and is specialized to perform node-specific administration functions, such as server process monitoring, configuration synchronization, file transfer, and request routing.

Preferences

Stop Restart Restart all Servers on Node

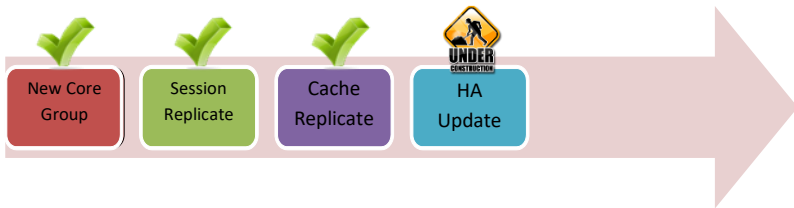
Select: Name Node Host Name Version Status

You can administer the following resources:

	Name	Node	Host Name	Version	Status
<input checked="" type="checkbox"/>	nodeagent	wasv90Node01	wasv90	ND 9.0.0.0	➔
<input type="checkbox"/>	nodeagent	wasv90Node02	wasv90	ND 9.0.0.0	➔
Total 2					

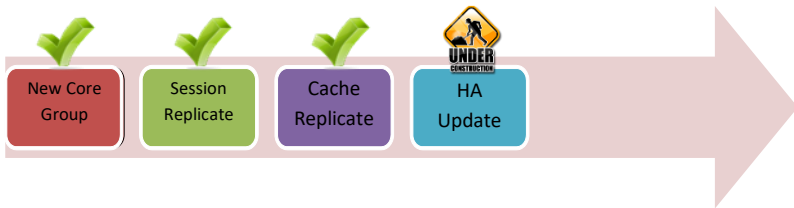
Field help
For field help information, select a field label or list marker when the help cursor is displayed.

Page help
More information about this page

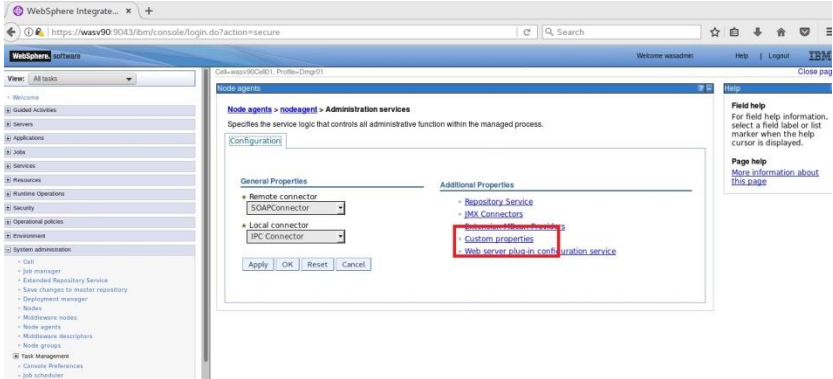


Step 2: Navigate to “Additional Properties>Administration services”.

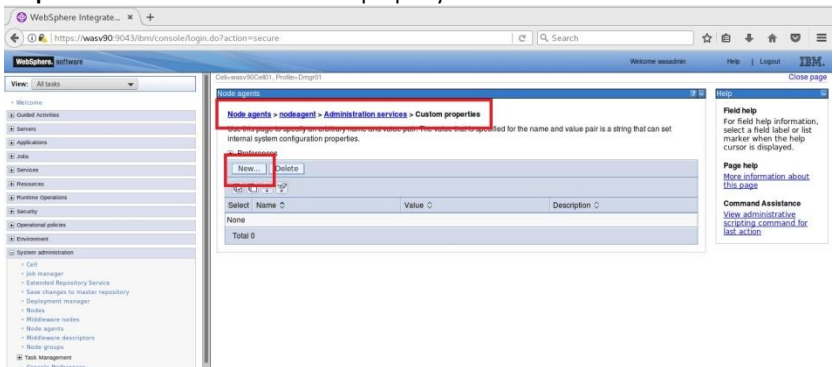
The screenshot shows the WebSphere software console interface. The left sidebar contains a navigation tree with categories like 'Overview', 'Related resources', 'Servers', 'Applications', 'JMS', 'Services', 'Resources', 'Runtime Operations', 'Security', 'Operational policies', 'Environment', and 'System administration'. The main content area is titled 'Node agents > nodeagent' and includes a description of the node agent process. Below this, there are tabs for 'Runtime' and 'Configuration'. The 'Configuration' tab is active, showing 'General Properties' and 'Server Infrastructure'. In the 'Server Infrastructure' section, the 'Additional Properties' sub-section is expanded, and 'Administration services' is highlighted with a red box. Other visible options include 'Core group service', 'Ports', 'File synchronization service', 'File transfer service', 'Performance monitoring infrastructure (PMI)', 'Custom services', 'QRS service', 'Diagnostic trace service', and 'Change log detail levels'.

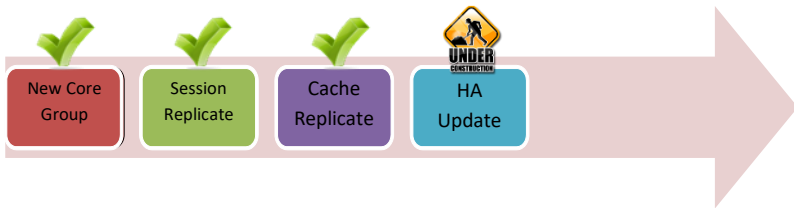


Step 3: Click on “Custom properties”.



Step 4: Click “New” to add a custom property.



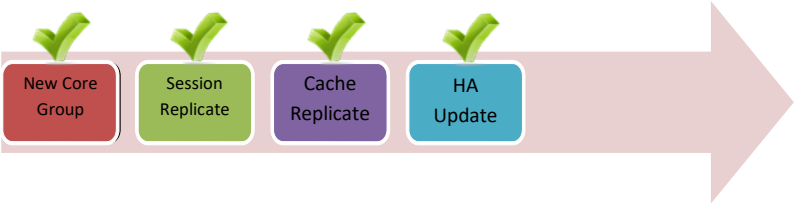


Step 5: Enter “*com.ibm.websphere.zos.rollout.pauseresume*” as “Name” and “true” as “Value” then click “OK”.

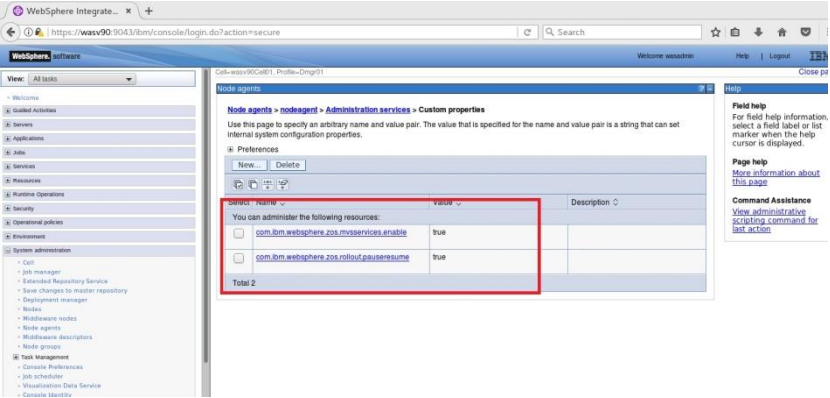
WebSphere Integration console screenshot showing the configuration of a custom property. The breadcrumb path is: Node agents > nodeagent > Administration services > Custom properties > New... The 'Name' field contains 'com.ibm.websphere.zos.rollout.pauseresume' and the 'Value' field contains 'true'. The 'OK' button is highlighted with a red box.

Step 6: Add another custom property that has name “*com.ibm.websphere.zos.mvsservices.enable*” and value “true” then click “OK”.

WebSphere Integration console screenshot showing the configuration of a custom property. The breadcrumb path is: Node agents > nodeagent > Administration services > Custom properties > New... The 'Name' field contains 'com.ibm.websphere.zos.mvsservices.enable' and the 'Value' field contains 'true'. The 'OK' button is highlighted with a red box.



Step 7: Repeat the steps 1 to 6 for the other node.



Task 4 is complete!

SUMMARY

High Availability is the ability to tolerate a certain amount of failures without any interruption to the service. WebSphere Application Server Network Deployment gives you different options to have a highly available environment for your business critical applications. In order to avoid single point of failure (SPOF), WebSphere Application Server provides vertical and horizontal scaling across different systems. WebSphere Application Server high availability manager (HA Manager) provides singleton processes to have process high availability.

REFERENCES

- http://www-01.ibm.com/support/knowledgecenter/SSZIPZ_8.5.0/com.ibm.swg.im.iis.production.iisinfo.v8r5.install.doc/topics/wsisinst_ha_wasclusteringtop.html?lang=en
- http://publib.boulder.ibm.com/infocenter/wsdm400/v6r0/index.jsp?topic=/com.ibm.websphere.iseries.doc/info/ae/ae/cprs_memory2memory.html
- <http://pic.dhe.ibm.com/infocenter/wxinfo/v8r5/index.jsp?topic=%2Fcom.ibm.websphere.extremescale.doc%2Ftxsdynache.html>

INDEX

core group	455
data availability.....	454
DRS.....	455
EJB persistence	454
EJB session state	454
High Availability	454
high availability manager	455
HTTP session state	454
load balancers.....	454
Memory-to-memory replication.....	455
redundancy	454
Singleton	455
SPOF.....	454
Workload management	455

•