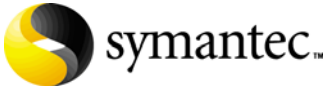
Application Note: Symantec High Availability Solution for WebSphere Commerce

UNIX



Application Note: Symantec High Availability Solution for WebSphere Commerce

Symantec Corporation
20330 Stevens Creek Blvd.
Cupertino, CA 95014
www.symantec.com

Third-party legal notices

Third-party software may be recommended, distributed, embedded, or bundled with this Symantec product. Such third-party software is licensed separately by its copyright holder.

Technical support

Visit http://www.symantec.com/business/support/assistance_care.jsp for product assistance. Use the Knowledge Base search feature to access resources such as TechNotes, product alerts, software downloads, hardware compatibility lists, and our customer email notification service. If you encounter an error when using a product, include the error number preceding the message when contacting Technical Services. You can also use the error number to search for information in TechNotes or documents on the Web site.

Symantec High Availability Solution for WebSphere Commerce

Introduction

The Symantec High Availability Solution utilizes the following products: Veritas™ Storage Foundation, Veritas™ Cluster Server, and Cluster Server agents, which are designed specifically for WebSphere Commerce and WebSphere Application Server. The result is a solution that you can quickly deploy, which protects critical WebSphere Commerce applications from either planned or unplanned downtime.

This document describes the overall WebSphere Commerce availability environment and the Symantec High Availability Solution. It also offers a technical overview of local availability in the WebSphere Commerce environment, and describes a few scenarios using the Symantec solution.

Symantec offers an end-to-end, fully integrated solution for ensuring highly available WebSphere Commerce environments. The Symantec solution simplifies the administration of the complex environment with a single interface, and supports local failover for business continuity purposes.

Components of the Solution

The Symantec High Availability Solution integrates the following software products:

- Veritas Storage Foundation
  This product combines Symantec's industry-leading file system and volume management solutions to create a highly available, robust foundation for WebSphere Commerce data. The journal file system restarts in seconds for fast failovers. Logical volumes support highly available, high performance storage configurations. Database-specific components such as direct I/O accelerate database read and write performance while simplifying the manageability of database data. Veritas Storage Foundation provides database-specific optimizations for Oracle, DB/2, Sybase, Windows, and Oracle RAC databases.
- Veritas Cluster Server (VCS)
  VCS eliminates planned and unplanned downtime by clustering critical applications and the resources they require. Specific agents for WebSphere Commerce, the underlying database, and the HTTP server ensure that all of the critical components of your WebSphere Commerce environment are monitored and managed centrally to ensure maximum application availability.

- Veritas High Availability agent for WebSphere Application Server
  The agent starts the WebSphere Commerce Application Server during the online operation, stops the Server during the offline operation, monitors the Server for critical processes, and removes processes and IPC resources that may remain from an ungraceful or incomplete shutdown of the Server instance. For more information, refer to *Veritas High Availability Agent for WebSphere Application Server Installation and Configuration Guide*.
- Veritas Agent for Apache
  The agent starts the IBM HTTP server during the online operation, stops the server during the offline operation, monitors the server for critical processes, and removes processes and IPC resources that may remain from an ungraceful or incomplete shutdown of the Apache instance. For more information, refer to documentation for the Veritas agent for Apache.
- Veritas Agent Builder tool
  This tool can be used to generate and deploy a high availability agent for WebSphere Commerce Information Center. Refer to "Clustering WebSphere Commerce Information Center using the Agent Builder tool" on page 17 on how to create the agent for the WebSphere Commerce Information Center using the Agent Builder tool.

Working together, the above software components help IT organizations improve WebSphere Commerce availability on a daily basis while offering significant protection from the loss of service and data in the case of an application failure.

Architecture of WebSphere Commerce systems

This section describes the architecture of WebSphere Commerce systems.

WebSphere Commerce components

A WebSphere Commerce instance has multiple services which are typically deployed across multiple servers. WebSphere Commerce identifies the following services as critical to the application environment, representing potential single points of failure:

- WebSphere Commerce Server (Application Server)
- IBM HTTP Server
- Database instance

Database layer

WebSphere Commerce uses the DB2 or Oracle relational database management systems (RDBMS) as the database server. Administering the database itself requires experience in the specific database used.

The database represents a single point of failure for the WebSphere Commerce system, unless the database itself is already clustered. The Symantec High Availability Solution has database-specific components for the DB2 and Oracle databases.

Application layer

WebSphere Commerce uses WebSphere Application Server at the application layer level.

Dependencies

The following components have strict dependencies in the order in which they are restarted in case of a failure.

For example:

- The database server must be available before the WebSphere Commerce instance is started.
- The HTTP server must be available before the WebSphere Commerce instance is started.
- The nodeagent (if configured) must be available before starting the WebSphere Application Server.

Configuring Veritas Cluster Server for WebSphere Commerce

Veritas Cluster Server (VCS) uses application-specific agents to start, stop, monitor, and switch over different applications and infrastructure components. A VCS environment running WebSphere Commerce uses VCS agents to monitor and track the WebSphere Commerce instance, the database, and the HTTP server.

The VCS agent for WebSphere monitors essential Application Server Java processes. The agent also provides the ability to start, stop, and monitor the WebSphere Application Server instance. Additionally, if a service fails, the agent cleans the node of any remaining system processes and resources.

VCS provides multiple levels of monitoring for WebSphere Commerce services:

- The first (default) level confirms the existence of essential processes in the process table.
- The second level, which is optional and additional to the first, runs `serverStatus.sh` to check the state of the WebSphere Commerce instance.
- The third level, which is also optional, invokes an external monitoring program, enabling you to provide your own scripts for application monitoring.

For more information, refer to documentation for the Veritas agent for WebSphere.

Setting up the cluster

Veritas Cluster Server provides highly flexible, scalable clustering configurations. A WebSphere Commerce environment might typically include the following system components in the VCS cluster:

- WebSphere Application Server
- WebSphere Deployment Manager
- WebSphere Node Agent
- Database Server
- HTTP Server

Figure 1-1 A VCS configuration screen for a WebSphere instance

About service groups

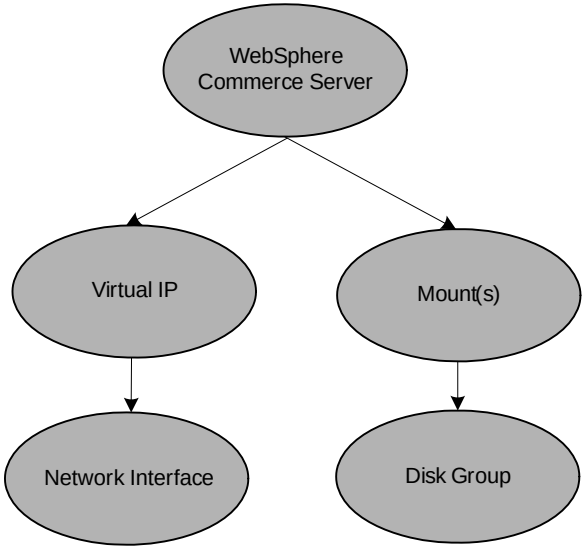
VCS provides application failover by encapsulating the resources required for each application into a service group — creating virtualized application services that can be moved among cluster nodes. The operations staff can operate on the cluster itself, on the service group (starting, stopping, switching over, and so on), or on the specific resources within the service group.

The WebSphere Commerce service group contains a set of dependent resources — the lower-level components that an application requires to operate successfully. Resources include disk groups, disk volumes, file systems, IP addresses, and dependent application processes.

VCS starts, stops, monitors, and switches service groups on any server in the cluster in response to server or resource faults. In addition, an administrator can proactively move a service group between cluster nodes to perform preventative maintenance or apply patches. The service group includes the logic about the dependencies between application components.

For example, the following diagram illustrates the relationship between the VCS resources required to support a WebSphere Commerce instance:

Figure 1-2 Relationship between the VCS resources for supporting a WebSphere Commerce instance



About disk groups

Each service group requires a dedicated file system, volume, and disk group to store the service group's data and programs. By importing and deporting this set of storage objects on different servers in the cluster without affecting other service groups, Veritas Cluster Server allows the service groups themselves to be independent of the underlying architecture and mobile across the cluster.

For example, when Veritas Cluster Server shuts down a service group, it starts at the top of the resource hierarchy:

- As the file system resource is shut down, VCS unmounts the file system
- As the volume is shut down, VCS stops the volume
- As the disk group is shut down, VCS deports the disk group

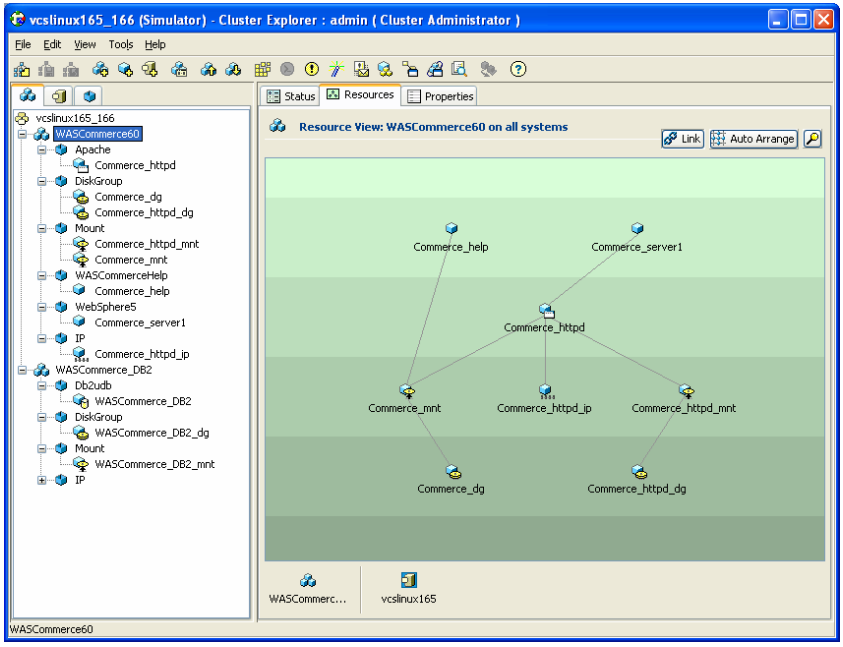
Veritas Cluster Server initiates a startup of the service group on another system in the cluster where each resource is started in dependent order. On the new system, VCS imports the disk group, starts the volume, and mounts the file system. This entire process happens automatically in the event of a failure, significantly reducing the downtime associated with a failure or outage. This sequence would not have been accomplished if the WebSphere Commerce component was installed on a local system disk on one node in the cluster. For more information on the DiskGroup agent, refer to *Veritas Cluster Server Bundled Agents Reference Guide.*

About network resources

In addition to disk groups, applications also require specific network resources, such as Network Interface Cards (NICs) and IP addresses.

The following figure illustrates both the network and disk group dependencies of WebSphere Commerce Server.

Figure 1-3 Network and disk group dependencies of WebSphere Commerce Server



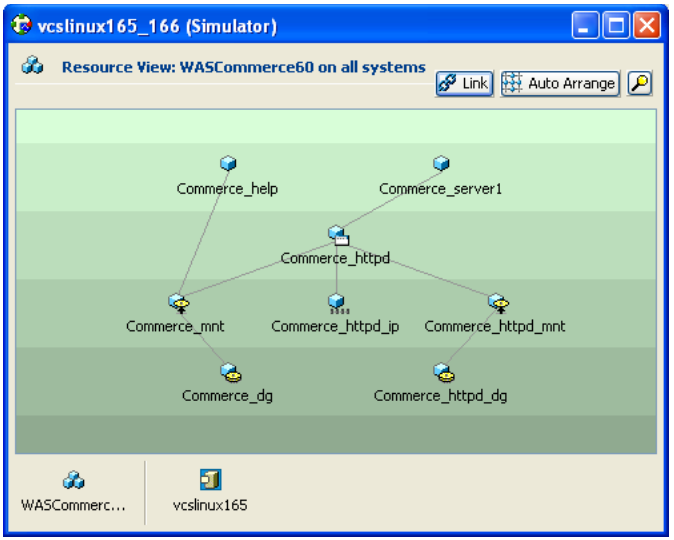
To support automated failover, WebSphere Commerce components must be configured with virtual IP addresses. If a service group becomes unavailable, VCS frees the virtual IP address so that it can be re-configured on the failover host. In this way, users connect to the application without regard for its physical location.

Sample configuration 1

This sample configuration considers the following scenario:

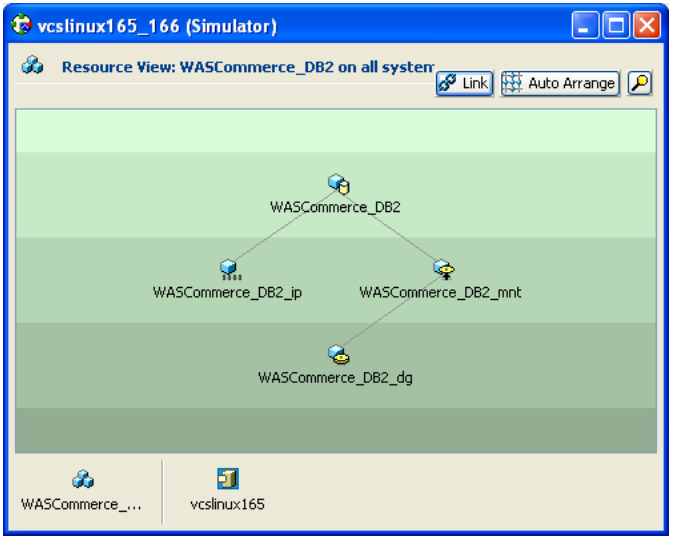
- A standalone WebSphere Commerce instance
- IBM HTTP Server
- DB2 Database
- WebSphere Commerce Information Center

Figure 1-4 Resource view for WASCommerce60 on all systems



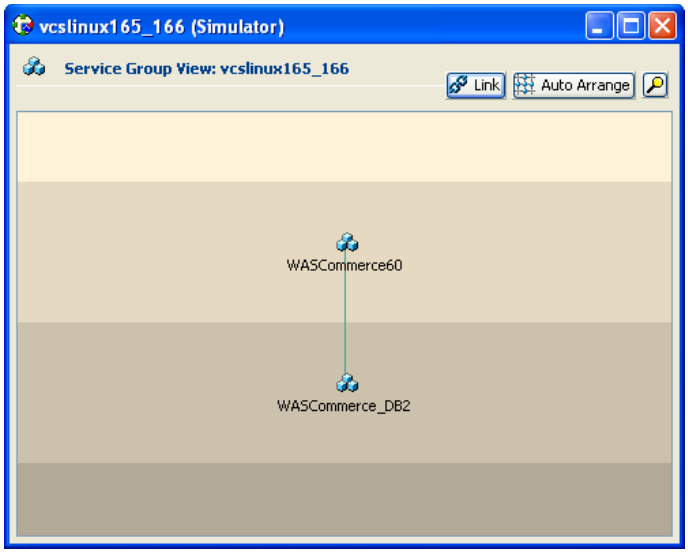
The service group WASCommerce60 consists of the HTTP Server resource, WebSphere Commerce Server resource, and the WC Information Center resource. The dependency makes sure that the HTTP server is started before starting the Commerce Server instance.

Figure 1-5 Resource view for WASCommerce_DB2 on all systems



Service group WASCommerce_DB2 consists of the VCS resource for the DB2 instance and the underlying infrastructure resources for IP, Mount, DiskGroup.

Figure 1-6 Service group view



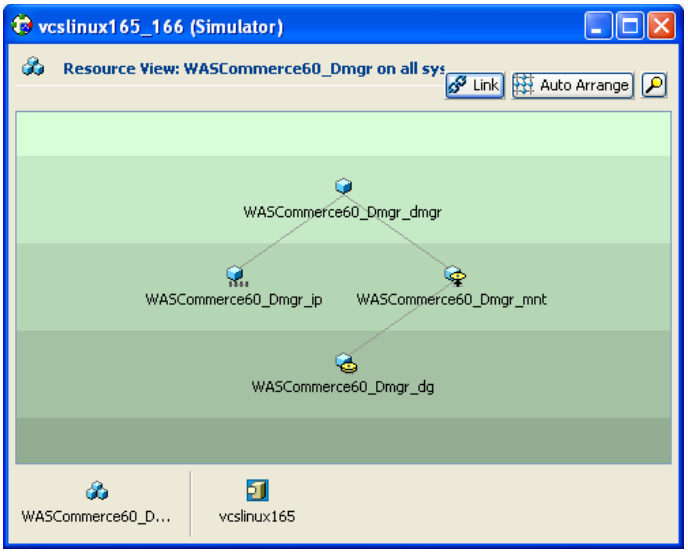
The group dependency ensures that the Database Server group must be brought online before the WebSphere Commerce service group is brought online.

Sample configuration 2

This sample configuration considers the following scenario:

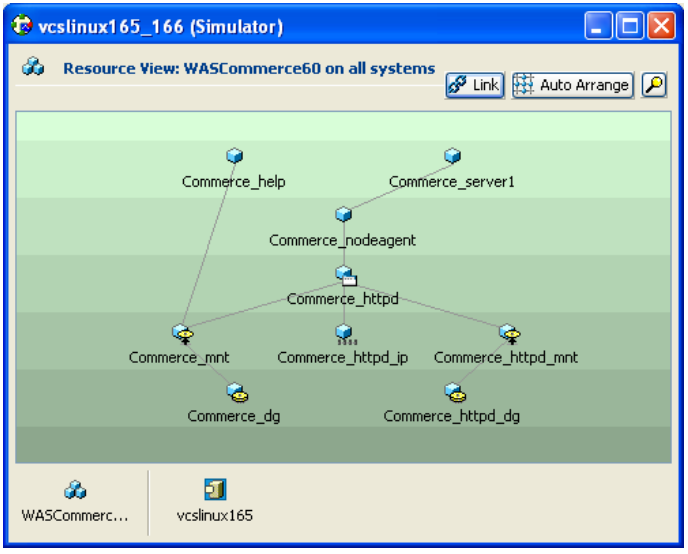
- Federated WebSphere Commerce Server
- Nodeagent for the WebSphere Commerce Server
- Deployment Manager
- IBM HTTP Server
- DB2 Database
- WebSphere Commerce Information Center

Figure 1-7 Resource view for WASCommerce60_Dmgr on all systems



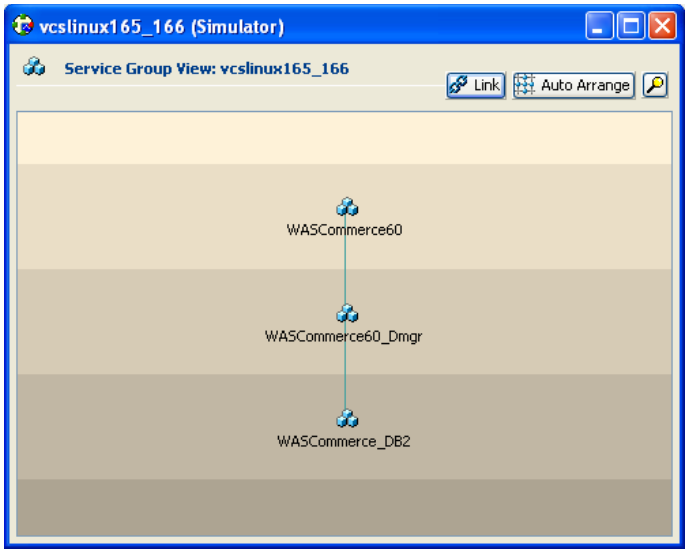
Service group WASCommerce60_Dmgr consists of WebSphere Deployment Manager and the underlying infrastructure VCS resources.

Figure 1-8 Resource view for WASCommerce60 on all systems



Service group WASCommerce60 consists of WebSphere Deployment Manager and the underlying infrastructure VCS resources.

Figure 1-9 Service group view



The group dependency ensures that the Database Server group must be brought online before the WebSphere Deployment Manager is brought online. The WebSphere Deployment Manager must be brought online before the WebSphere Commerce Server is brought online.

Clustering WebSphere Commerce Information Center using the Agent Builder tool

This section describes how to cluster WebSphere Commerce Information Center using the Agent Builder tool.

Installing the Agent Builder tool

Before proceeding with the steps in this section, make sure that you have the latest Agent Pack.

Perform the following steps on each system in the cluster

1. Log in as root.
2. Navigate to the directory containing the Agent Builder tool.
   For example:
   `<agentpack topdir>/<platform>/application/agentbuilder/5.0/5.1_agent`

3 Review the agent builder Application Note in the `docs` directory.

4 Install the Agent Builder and ACC Library package on each node as described in the document.

5 Identify the cluster nodes on which the WebSphere Commerce Information Center will be running as a VCS resource.

6 Navigate to the installed location of the Agent Builder tool and execute it.
For example:
VCS 5.0

```
# cd /opt/VRTSagents/ha/bin/AgentBuilder
# ./agentbuilder WASCommerceHelp -base vcs5 -platform
<platform> -rsh -system vcssun1 -system vcssun2
```

VCS 4.1

```
# cd /opt/VRTSvcs/bin/AgentBuilder
# ./agentbuilder WASCommerceHelp -base vcs4 -platform
<platform> -rsh -system vcssun1 -system vcssun2
```

The above command deploys the WASCommerceHelp agent on cluster nodes vcssun1 and vcssun2 through rsh. Make sure that rsh (without a password) or ssh is configured across the cluster nodes where you want to deploy the agent.

WASCommerceHelp attributes

In the following example configuration the WebSphere Commerce Server is installed in `/opt/IBM/WebSphere/CommerceServer60` directory.

Table 1-1 WASCommerceHelp attributes

Attribute	Description
User	OS user running the WebSphere Commerce Information Center
StartProgram	Start script provided by WebSphere Commerce Information Center
StopProgram	Stop script provided by WebSphere Commerce Information Center
CleanProgram	Stop script provided by WebSphere Commerce Information Center
MonitorSequence	MonitorProcessPatterns PidFilesPatterns ListenAddressPort

Note: The default values need not be changed.

Configuring the MonitorProcessPatterns attribute

You can configure the `MonitorProcessPatterns` attribute of the WebSphere Commerce Information Center VCS resource by identifying WebSphere Commerce Information Center processes.

For example:

```
# ps -ef | grep java | grep HelpSystem
wasuser   4187     1  0 16:14 ?        00:00:00
/opt/IBM/WebSphere/AppServer/java/jre/bin/java
-DWAS_HOME=/opt/IBM/WebSphere/AppServer
-DWAS_INST=/opt/IBM/WebSphere/AppServer
-DWAS_PLUGIN_HOME=/opt/IBM/WebSphere/Plugins -classpath
/opt/IBM/WebSphere/CommerceServer60/CommerceHelpSystem/eclipse/plug
ins/org.eclipse.help.base_3.1.0/helpbase.jar
org.eclipse.help.standalone.Infocenter -eclipsehome
/opt/IBM/WebSphere/CommerceServer60/CommerceHelpSystem/eclipse
-command start -plugincustomization
plugins/com.ibm.commerce.base.doc/plugin_customization.ini -port
8001 -noexec -clean -vmargs -Xmx256M

wasuser   4198  4187  5 16:14 ?        00:00:08
/opt/IBM/WebSphere/AppServer/java/jre/bin/java -Xmx256M
-Dserver_port=8001 -cp startup.jar org.eclipse.core.launcher.Main
-nosplash -application org.eclipse.help.base.infocenterApplication
-data
/opt/IBM/WebSphere/CommerceServer60/CommerceHelpSystem/eclipse/work
space -plugincustomization
plugins/com.ibm.commerce.base.doc/plugin_customization.ini -clean
```

The processes matching the above pattern can be used to configure the `MonitorProcessPatterns` attribute of the WebSphere Commerce Information Center VCS resource.

Sample WebSphere Commerce Information Center resource attributes

Table 1-2 Sample WebSphere Commerce Information Center resource attributes

Attribute	Value
CleanProgram	/opt/IBM/WebSphere/CommerceServer60/bin/stopHelp.sh
MonitorProcessPatterns	"/opt/IBM/WebSphere/AppServer/java/jre/bin/java -DWAS_HOME=/opt/IBM/WebSphere/AppServer -DWAS_INST=/opt/IBM/WebSphere/AppServer -DWAS_PLUGIN_HOME=/opt/IBM/WebSphere/Plugins -classpath /opt/IBM/WebSphere/CommerceServer60/CommerceHelpSystem/eclipse/plugins/org.eclipse.help.base_3.1.0/helpbase.jar org.eclipse.help.standalone.Infocenter -eclipsehome /opt/IBM/WebSphere/CommerceServer60/CommerceHelpSystem/eclipse -command start -plugincustomization plugins/com.ibm.commerce.base.doc/plugin_customization.ini -port 8001 -noexec -clean -vmargs -Xmx256M" "/opt/IBM/WebSphere/AppServer/java/jre/bin/java -Xmx256M -Dserver_port=8001 -cp startup.jar org.eclipse.core.launcher.Main -nosplash -application org.eclipse.help.base.infocenterApplication -data /opt/IBM/WebSphere/CommerceServer60/CommerceHelpSystem/eclipse/workspace -plugincustomization plugins/com.ibm.commerce.base.doc/plugin_customization.ini -clean"
MonitorSequence	MonitorProcessPatterns PidFilesPatterns ListenAddressPort
ResLogLevel	INFO
StartProgram	/opt/IBM/WebSphere/CommerceServer60/bin/startHelp.sh
StopProgram	/opt/IBM/WebSphere/CommerceServer60/bin/stopHelp.sh