Cluster Server Agent for Informatica Informatica Installation and Configuration Guide

AIX, Linux, Solaris

7.0

March 2017

VERITAS™

Cluster Server Agent for Informatica Informatica Installation and Configuration Guide

Last updated: 2017-03-29

Legal Notice

Veritas Technologies LLC
500 E Middlefield Road
Mountain View, CA 94043

http://www.veritas.com

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

https://www.veritas.com/support

You can manage your Veritas account information at the following URL:

https://my.veritas.com

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

Worldwide (except Japan)	CustomerCare@veritas.com
Japan	CustomerCare_Japan@veritas.com

Documentation

Make sure that you have the current version of the documentation. Each document displays the date of the last update on page 2. The latest documentation is available on the Veritas website:

https://sort.veritas.com/documents

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

xyz@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

http://www.veritas.com/community/

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

Chapter 1

Introducing the agent for Informatica

This chapter includes the following topics:

About the Cluster Server agent for Informatica

Cluster Server (VCS) agents monitor specific resources within an enterprise application. They determine the status of resources and start or stop them according to external events.

The Cluster Server agent for Informatica provides high availability for the Informatica B2B Data Exchange and PowerCenter applications. The agent monitors specific PowerCenter components, and Informatica B2B Data Exchange components such as:

- B2B Data Exchange Server (dxserver)
- B2B Data Exchange Operation Console (dxconsole)
- B2B Data Exchange JMS Broker (activemq)

- B2B Managed File Transfer (mft)

The agent brings these components online, takes them offline and, in case of a failure, shuts Informatica down.

Supported software

For information on the software versions that the Cluster Server agent for Informatica supports, see the Veritas Services and Operations Readiness Tools (SORT) site: https://sort.veritas.com/agents.

Features of the agent

The following are the features of the Cluster Server agent for Informatica:

- Support for validation of attributes that are based on the agent functions
  The agent can validate attributes in each agent function before the actual data processing starts.
- Support for First Failure Data Capture (FFDC)
  In case of a fault, the agent generates a huge volume of the debug logs that enable troubleshooting of the fault.
- Support for Fast First Level Monitor (FFLM)
  The agent maintains PID files based on search patterns to expedite the monitoring process.
- Support for external user-supplied monitor utilities
  The agent enables user-specified monitor utilities to be plugged in, in addition to the built-in monitoring logic. This enables administrators to completely customize the monitoring of the application.
- Support for intelligent resource monitoring and poll-based monitoring
  The agent supports the Cluster Server Intelligent Monitoring Framework (IMF) feature. IMF allows the agent to register the resources to be monitored with the IMF notification module so as to receive immediate notification of resource state changes without having to periodically poll the resources.

How the agent makes Informatica highly available

The agent provides the following levels of application monitoring:

- Primary or Basic monitoring
  This mode has Process check and Health check monitoring options. With the default Process check option, the agent verifies that the Informatica component

processes are present in the process table. Process check cannot detect whether processes are in the hung or stopped states.

- Secondary or Detail monitoring
  In this mode, the agent runs a utility to verify the status of the Informatica component. The agent detects application failure if the monitoring routine reports an improper function of the Informatica component processes. When this application failure occurs, the Informatica component service group fails over to another node in the cluster.
  In addition to these levels of application monitoring, the agent for Informatica is IMF-aware and uses asynchronous monitoring framework (AMF) kernel driver for IMF notification.
  Thus, the agent ensures high availability for Informatica components.

How the agent supports intelligent resource monitoring

With Intelligent Monitoring Framework (IMF), VCS supports intelligent resource monitoring in addition to the poll-based monitoring. Poll-based monitoring polls the resources periodically whereas intelligent monitoring performs asynchronous monitoring.

When an IMF-enabled agent starts up, the agent initializes the Asynchronous Monitoring Framework (AMF) kernel driver. After the resource is in a steady state, the agent registers with the AMF kernel driver, the details of the resource that are required to monitor the resource. For example, the agent for Informatica registers the PIDs of the Informatica processes with the AMF kernel driver. The agent's imf_getnotification function waits for any resource state changes. When the AMF kernel driver module notifies the imf_getnotification function about a resource state change, the agent framework runs the monitor agent function to ascertain the state of that resource. The agent notifies the state change to VCS, which then takes appropriate action.

You can enable or disable the intelligent resource monitoring functionality of the VCS agent for Cluster Server manually. See "Enabling and disabling intelligent resource monitoring manually" on page 29.

Refer to the *Cluster Server Administrator's Guide* for more information.

Informatica agent functions

The agent consists of resource type declarations and agent executables. The agent executables are organized into online, offline, monitor, and clean functions.

Online

The online function performs the following tasks:

- Verifies that the required attributes are set correctly.
- Performs a preliminary check to ensure that the Informatica component is not already online.
  If the component is online, the online operation exits immediately.
- Depending upon the Informatica component that is configured, the agent starts the component by running the following command:

activemq	*DXHome*/bin/activemq.sh start
dxconsole	*DXHome*/bin/dxconsole.sh start
dxserver	*DXHome*/bin/dxserver.sh start *Port*
mft	*DXHome*/bin/mft.sh start
powercenter	*InfaHome*/server/tomcat/bin/infaservice.sh startup

- Ensures that the component is up and running successfully. The agent function uses the wait period that the OnlineTimeout attribute specifies, to enable the Informatica component to initialize fully before allowing the monitor function to probe the newly running component.
- Returns the control to HAD.

Offline

The offline function performs the following tasks:

- Verifies that the required attributes are set correctly.
- Performs a preliminary check to ensure that the Informatica component is not already offline.
  If the component is offline, the offline operation kills any existing processes that belong to this component and exits
- Depending upon the Informatica component that is configured, the agent stops the component gracefully by running the following commands:

activemq	*DXHome*/bin/activemq.sh stop

dxconsole	*DXHome*/bin/dxconsole.sh stop
dxserver	*DXHome*/bin/dxserver.sh stop *Hostname Port*
mft	*DXHome*/bin/mft.sh stop
powercenter	*InfaHome*/server/tomcat/bin/infaservice.sh shutdown

- Ensures that the resource is given enough time to go offline successfully. The agent function uses a wait period that the OfflineTimeout attribute specifies, to allow the Informatica component to complete the offline sequence before allowing further probing of the resource.
- Returns the control to HAD.

Monitor

The monitor function performs the following tasks:

- Conducts a first-level check on the Informatica component to ensure that the relevant process is running. The agent identifies the process for the component by applying the pattern matching on command lines of processes running in the system.
  The agent for Informatica also supports Intelligent Monitoring Framework (IMF) in the first-level check. IMF enables intelligent resource monitoring. See "How the agent supports intelligent resource monitoring" on page 8.
  You can use the MonitorFreq key of the IMF attribute to specify the frequency at which the agent invokes the monitor function.
- Depending on the Informatica component that is configured, the monitor function can conduct an in-depth, second-level check on the component.
  For the B2B Data Exchange JMS Broker (activemq) and B2B Data Exchange Server (dxserver) components, the second-level check runs the following commands:

activemq	*ActiveMQHome*/bin/activemq-admin list --jmxurl service :jmx:rmi:///jndi/rmi://localhost:$iPort/jmxrmi
dxserver	*DXHome*/bin/dxserver.sh ping *Hostname iPort*

- For the B2B Data Exchange Operation Console (dxconsole) and PowerCenter (powercenter) components, the Monitor function queries the webserver running on the specified port and checks the HTTP return status.
- If the B2B Managed File Transfer (mft) component is configured, the Monitor function uses a connect (3c) method to check for the Informatica component to listen to the port defined by the Port attribute. The host name needed to perform this check is derived from the InfaHome agent attribute.
- Depending upon the value of the MonitorProgram attribute, the monitor function can perform a customized check using a user-supplied monitoring utility.

Clean

The clean function performs the following tasks:

- Attempts to gracefully shut down the Informatica component.
  Depending upon the Informatica component that is configured, the agent stops the component gracefully by running the following commands:

activemq	*DXHome*/bin/activemq.sh stop
dxconsole	*DXHome*/bin/dxconsole.sh stop
dxserver	*DXHome*/bin/dxserver.sh stop *Hostname Port*
mft	*DXHome*/bin/mft.sh stop
powercenter	*InfaHome*/server/tomcat/bin/infaservice.sh shutdown

- Identifies the process for the Informatica component and kills it.
- Returns the control to HAD.

Setting up Informatica in a VCS cluster

Follow the steps below to set up Informatica in a cluster:

- Set up a VCS cluster.
  For more information on installing and configuring Cluster Server, refer to the Cluster Server installation and configuration guides.
- Install and configure Informatica for High Availability.
  See “About configuring the Cluster Server agent for Informatica” on page 22.

- Install the Cluster Server agent for Informatica.
- Configure the service groups for Informatica.

Chapter 2

Installing, upgrading, and removing the agent for Informatica

This chapter includes the following topics:

Before you install the Cluster Server agent for Informatica

You must install the Cluster Server agent for Informatica on all the systems that will host Informatica service groups.

Before you install the agent for Informatica, ensure that the following prerequisites are met.

- Install and configure Cluster Server.
  For more information on installing and configuring Cluster Server, refer to the Cluster Server installation and configuration guides.
- Install the latest version of ACC Library.

To install or update the ACC Library package, locate the library and related documentation in the Agent Pack tarball,
See "Installing the ACC library" on page 14.

About the ACC library

The operations of a Cluster Server agent depend on a set of Perl modules known as the ACC library. The library must be installed on each system in the cluster that runs the agent. The ACC library contains common, reusable functions that perform tasks, such as process identification, logging, and system calls.

Instructions to install or remove the ACC library on a single system in the cluster are given in the following sections. The instructions assume that the ACCLib tar file has already been extracted.

Note: The LogDbg attribute should be used to enable debug logs for the ACCLib-based agents when the ACCLib version is 6.2.0.0 or later and VCS version is 6.2 or later.

Installing the ACC library

Install the ACC library on each system in the cluster that runs an agent that depends on the ACC library.

To install the ACC library

1. Log in as a superuser.
2. Download ACC Library.

   You can download either the complete Agent Pack tar file or the individual ACCLib tar file from the Veritas Services and Operations Readiness Tools (SORT) site (https://sort.veritas.com/agents).
3. If you downloaded the complete Agent Pack tar file, navigate to the directory containing the package for the platform running in your environment.

AIX	*cd1*/aix/vcs/application/acc_library/*version*_library/pkgs
Linux	*cd1*/linux/generic/vcs/application/acc_library/*version*_library/rpms
Solaris	*cd1*/solaris/*dist_arch*/vcs/application/acc_library/*version*_library/pkgs where *dist_arch* is sol_sparc.

4 If you downloaded the individual ACCLib tar file, navigate to the pkgs directory (for AIX and Solaris), or rpms directory (for Linux).

5 Install the package. Enter **Yes**, if asked to confirm overwriting of files in the existing package.

AIX
```
# installp -ac -d VRTSacclib.bff VRTSacclib
```

Linux
```
# rpm -i \
VRTSacclib-VersionNumber-GA_GENERIC.noarch.rpm
```

Solaris
```
# pkgadd -d VRTSacclib.pkg
```

Note: The LogDbg attribute should be used to enable debug logs for the ACCLib-based agents when the ACCLib version is 6.2.0.0 or later and VCS version is 6.2 or later.

Installing the ACC library IPS package on Oracle Solaris 11 systems

Install the ACC library IPS package on an Oracle Solaris 11 system.

To install the ACC library IPS package on Oracle Solaris 11 systems

1 Copy the `VRTSacclib.p5p` package from the pkgs directory to the system in the `/tmp/install` directory.

2 Disable the publishers that are not reachable as package install may fail, if any, of the already added repositories are unreachable.

```
# pkg set-publisher --disable <publisher name>
```

3 Add a file-based repository in the system.

```
# pkg set-publisher -g /tmp/install/VRTSacclib.p5p Veritas
```

4 Install the package.

```
# pkg install --accept VRTSacclib
```

5 Remove the publisher from the system.

```
# pkg unset-publisher Veritas
```

6 Enable the publishers that were disabled earlier.

```
# pkg set-publisher --enable <publisher name>
```

Installing the ACC library package on Solaris brand non-global zones

With Oracle Solaris 11, you must install the ACC library package inside non-global zones. The native non-global zones are called Solaris brand zones.

To install the ACC library package on Solaris brand non-global zones

1. Ensure that the SMF services, `svc:/application/pkg/system-repository:default` and `svc:/application/pkg/zones-proxyd:default`, are online on the global zone.

   ```
   # svcs svc:/application/pkg/system-repository:default
   ```

   ```
   # svcs svc:/application/pkg/zones-proxyd:default
   ```

2. Log on to the non-global zone as a superuser.
3. Ensure that the SMF service `svc:/application/pkg/zones-proxy-client:default` is online inside the non-global zone:

   ```
   # svcs svc:/application/pkg/zones-proxy-client:default
   ```

4. Copy the `VRTSacclib.p5p` package from the `pkgs` directory to the non-global zone (for example, at the `/tmp/install` directory).
5. Disable the publishers that are not reachable, as package install may fail, if any of the already added repositories are unreachable.

   ```
   # pkg set-publisher --disable <publisher name>
   ```

6. Add a file-based repository in the non-global zone.

   ```
   # pkg set-publisher -g/tmp/install/VRTSacclib.p5p Veritas
   ```

7. Install the package.

   ```
   # pkg install --accept VRTSacclib
   ```

8. Remove the publisher on the non-global zone.

   ```
   # pkg unset-publisher Veritas
   ```

9. Clear the state of the SMF service, as setting the file-based repository causes the SMF service `svc:/application/pkg/system-repository:default` to go into the maintenance state.

   ```
   # svcadm clear svc:/application/pkg/system-repository:default
   ```

10. Enable the publishers that were disabled earlier.

    ```
    # pkg set-publisher --enable <publisher>
    ```

Note: Perform steps 2 through 10 on each non-global zone.

Installing the agent in a VCS environment

Install the agent for Informatica on each node in the cluster.

To install the agent in a VCS environment

1. Download the agent from the Veritas Services and Operations Readiness Tools (SORT) site: https://sort.veritas.com/agents.

   You can download either the complete Agent Pack tar file or an individual agent tar file.

2. Uncompress the file to a temporary location, say /tmp.

3. If you downloaded the complete Agent Pack tar file, navigate to the directory containing the package for the platform running in your environment.

   AIX `cd1/aix/vcs/application/informatica_agent/vcs_version/version_agent/pkgs`

   Linux `cd1/linux/generic/vcs/application/informatica_agent/vcs_version/version_agent/rpms`

   Solaris `cd1/solaris/dist_arch/vcs/application/informatica_agent/vcs_version/version_agent/pkgs`

   where, *dist_arch* is sol_sparc.

   If you downloaded the individual agent tar file, navigate to the pkgs directory (for AIX and Solaris), or rpms directory (for Linux).

4. Log in as a superuser.

5 Install the package.

AIX
```
# installp -ac -d
VRTSinformatica.rte.bff VRTSinformatica.rte
```

Linux
```
# rpm -ihv \
VRTSinformatica-AgentVersion-GA_GENERIC.noarch.rpm
```

Solaris
```
# pkgadd -d . VRTSinformatica
```

6 After installing the agent package, you must import the agent type configuration file.

Installing the agent IPS package on Oracle Solaris 11 systems

To install the agent IPS package on an Oracle Solaris 11 system

1 Copy the `VRTSinformatica.p5p` package from the `pkgs` directory to the system in the `/tmp/install` directory.

2 Disable the publishers that are not reachable as package install may fail, if any of the already added repositories are unreachable.

```
# pkg set-publisher --disable <publisher name>
```

where the publisher name is obtained using the `pkg publisher` command.

3 Add a file-based repository in the system.

```
# pkg set-publisher -g /tmp/install/VRTSinformatica.p5p Veritas
```

4 Install the package.

```
# pkg install --accept VRTSinformatica
```

5 Remove the publisher from the system.

```
# pkg unset-publisher Veritas
```

6 Enable the publishers that were disabled earlier.

```
# pkg set-publisher --enable <publisher name>
```

Installing agent packages on Solaris brand non-global zones

With Oracle Solaris 11, you must install the agent package inside non-global zones. The native non-global zones are called Solaris brand zones.

To install the agent package on Solaris brand non-global zones

1. Ensure that the SMF services, `svc:/application/pkg/system-repository:default` and `svc:/application/pkg/zones-proxyd:default`, are online on the global zone.

   ```
   # svcs svc:/application/pkg/system-repository:default
   ```

   ```
   # svcs svc:/application/pkg/zones-proxyd:default
   ```

2. Log on to the non-global zone as a superuser.
3. Ensure that the SMF service `svc:/application/pkg/zones-proxy-client:default` is online inside non-global zone:

   ```
   # svcs svc:/application/pkg/zones-proxy-client:default
   ```

4. Copy the `VRTSinformatica.p5p` package from the `pkgs` directory to the non-global zone (for example, at the `/tmp/install` directory).
5. Disable the publishers that are not reachable, as package install may fail, if any of the already added repositories are unreachable.

   ```
   # pkg set-publisher --disable <publisher name>
   ```

6. Add a file-based repository in the non-global zone.

   ```
   # pkg set-publisher -g/tmp/install/VRTSinformatica.p5p Veritas
   ```

7. Install the package.

   ```
   # pkg install --accept VRTSinformatica
   ```

8. Remove the publisher on the non-global zone.

   ```
   # pkg unset-publisher Veritas
   ```

9. Clear the state of the SMF service, as setting the file-based repository causes the SMF service `svc:/application/pkg/system-repository:default` to go into the maintenance state.

   ```
   # svcadm clear svc:/application/pkg/system-repository:default
   ```

10. Enable the publishers that were disabled earlier.

    ```
    # pkg set-publisher --enable <publisher>
    ```

Note: Perform steps 2 through 10 on each non-global zone.

Installing the agent in a Solaris 10 brand zone

To install the Informatica agent in a Solaris 10 brand zone:

- Ensure that the ACC library package, VRTSacclib, is installed in the non-global zone.
  To install VRTSacclib in the non-global zone, run the following command from the global zone:
  ```
  # pkgadd -R /zones/zone1/root -d VRTSacclib.pkg
  ```
- To install the agent package in the non-global zone, run the following command from the global zone:
  ```
  # pkgadd -R zone-root/root -d . VRTSinformatica
  ```
  For example: `# pkgadd -R /zones/zone1/root -d . VRTSinformatica`

Uninstalling the agent in a VCS environment

You must uninstall the agent for Informatica from a cluster while the cluster is active.

To uninstall the agent in a VCS environment

1. Log in as a superuser.
2. Set the cluster configuration mode to read/write by running the following command from any node in the cluster:
   ```
   # haconf -makerw
   ```
3. Remove all Informatica resources from the cluster. Run the following command to verify that all resources have been removed:
   ```
   # hares -list Type=Informatica
   ```
4. Remove the agent type from the cluster configuration by running the following command from any node in the cluster:
   ```
   # hatype -delete Informatica
   ```
   Removing the agent's type file from the cluster removes the include statement for the agent from the `main.cf` file, but the agent's type file is not removed from the cluster configuration directory. You can remove the agent's type file later from the cluster configuration directory.

5 Save these changes. Then set the cluster configuration mode to read-only by running the following command from any node in the cluster:

```
# haconf -dump -makero
```

6 Use the platform's native software management program to remove the agent for Informatica from each node in the cluster.

Run the following command to uninstall the agent:

AIX	`# installp -u VRTSinformatica.rte`
Linux	`# rpm -e VRTSinformatica`

Solaris	`# pkgrm VRTSinformatica`
	Note: To uninstall the agent IPS package on a Solaris 11 system, run the following command:
	`# pkg uninstall VRTSinformatica`

Removing the ACC library

Perform the following steps to remove the ACC library.

To remove the ACC library

1 Ensure that all agents that use ACC library are removed.

2 Run the following command to remove the ACC library package:

AIX	`# installp -u VRTSacclib`
Linux	`# rpm -e VRTSacclib`

Solaris	`# pkgrm VRTSacclib`
	Note: To uninstall the ACCLib IPS package on a Solaris 11 system, run the following command:
	`# pkg uninstall VRTSacclib`

Chapter 3

Configuring the agent for Informatica

This chapter includes the following topics:

About configuring the Cluster Server agent for Informatica

After installing the Cluster Server agent for Informatica, you must import the agent type configuration file. After importing this file, review the attributes table that describes the resource type and its attributes, and then create and configure Informatica resources and service groups.

You must have administrator privileges to create and configure a service group. You can configure the service groups using one of the following:

- The Cluster Manager (Java console)
- Veritas Infoscale Operations Manager
- The command-line

To view the sample agent type definition and service groups configuration:

See “About sample configurations for the agents for Informatica” on page 38.

Before configuring the resources for Informatica

Before you configure the Informatica resources, you must:

- Verify that Cluster Server is installed and configured on all nodes in the cluster where you will configure the service group.
  For more information on installing and configuring Cluster Server, refer to the Cluster Server installation and configuration guides.
- Verify that the agent for Informatica is installed on all nodes in the cluster.
  See "Installing the agent in a VCS environment" on page 17.
- If you are configuring resources for any of the supported B2B Data Exchange components, you must correct the B2B Data Exchange start script before you start any of the components. To correct the start script, you must replace the front slash (/) with a back slash (\) in line 105 of the script:

```
$ diff -u dxruntime.sh.orig dxruntime.sh
--- dxruntime.sh.orig   2012-09-21 12:35:46.000000000 +0530
+++ dxruntime.sh        2012-07-20 20:52:21.000000000 +0530
@@ -102,7 +102,7 @@
 echo ""

 if [ "${ACTION}" = "start" ]; then
-  "$JAVA" $JAVA_OPTS -classpath $RUN_CLASS_PATH com.informatica
   .b2b.dx.broker.DXControl $* /
+  "$JAVA" $JAVA_OPTS -classpath $RUN_CLASS_PATH com.informatica
   .b2b.dx.broker.DXControl $* \
   >> $DX_HOME/logs/dxserver.out 2>&1 &
 else
   "$JAVA" $JAVA_OPTS -classpath $RUN_CLASS_PATH com.informatica
   .b2b.dx.broker.DXControl $*
```

Importing the agent types files in a VCS environment

To use the agent for Informatica, you must import the agent types file into the cluster.

You can import the agent types file using the VCS graphical user interface or using the command line interface.

To import the agent types file using the VCS graphical user interface

1. Start the Cluster Manager (Java Console) and connect to the cluster on which the agent is installed.
2. Click **File > Import Types**.
3. In the **Import Types** dialog box, select the following file:

   ```
   /etc/VRTSagents/ha/conf/Informatica/InformaticaTypes.cf
   ```

4. Click **Import**.
5. Save the VCS configuration.

   You can now create Informatica resources. For additional information about using the VCS GUI, refer to the *Cluster Server Administrator's Guide*.

To import the agent types file using the command line interface (CLI):

1. Log on to any one of the systems in the cluster as the superuser.
2. Run the following command:

   ```
   # sh /etc/VRTSagents/ha/conf/Informatica/InformaticaTypes.cmd
   ```

3. To verify that the agent types file is successfully imported to the VCS engine, run the following command:

   ```
   # hatype -display Informatica
   ```

   You can now create Informatica resources.

Informatica agent attributes

Refer to the following required and optional attributes while configuring the agent for Informatica.

Table 3-1 lists the required attributes for the agent for Informatica.

Table 3-1 Required attributes

Required attribute	Description
ResLogLevel	The logging detail performed by the agent for Informatica for the resource. Valid values are: ERROR: Only logs error messages. WARN: Logs above plus warning messages. INFO: Logs above plus informational messages. TRACE: Logs above plus trace messages. TRACE is very verbose and should only be used during initial configuration or for troubleshooting and diagnostic operations. Default: INFO Example: TRACE
Component	The components of B2B Data Exchange or PowerCenter for which the resource must be configured. Default: none Examples: activemq, dxconsole, dxserver, mft, powercenter
User	The UNIX user name used to start and stop the Informatica processes. If MonitorProgram is specified, the agent for Informatica uses this user's credentials to run the defined program. You must synchronize the user name across the systems within the cluster. This user name must resolve to the same UID and have the same default shell on each system in the cluster. The agent operations use the getpwnam (3C) function system call to obtain UNIX user attributes. Hence you can define the user name locally or in a common repository such as NIS, NIS+, or LDAP. Example: "infa", "informatica"
Hostname	The hostname or IP address of the system where the resource is configured to be running. Depending on the component, the hostname is: ■ For B2B Managed File Transfer (mft): the host name or the IP address of the system where the MFT process runs. ■ For B2B Data Exchange JMS Broker (activemq): localhost ■ For B2B Data Exchange Operation Console (dxconsole): the host name or the IP address of the webserver. ■ For B2B Data Exchange Server (dxserver): the host name or the IP address of the system where the dxserver process runs. Default: localhost Examples: localhost, myhostname.mydomain.com, 192.168.1.100

Table 3-1 Required attributes *(continued)*

Required attribute	Description
Port	The port that is used to query the status of a component. Specify a unique port for each component. Default: none Example: 18095
DXHome	The installation directory that is used while installing B2B Data Exchange. The start and stop scripts of the B2B Data Exchange components reside in this directory. You must configure this attribute if any of the following B2B Data Exchange components are configured: B2B Data Exchange JMS Broker (activemq), B2B Data Exchange Operation Console (dxconsole), B2B Data Exchange Server (dxserver), or B2B Managed File Transfer (mft). Default: none Example: `/opt/infahome/B2B/DataExchange`
ActiveMQHome	The complete path of the fuse-message-broker directory. Typically, this directory is inside the DXHome directory, as <*DX_HOME*>/ fuse-message-broker-5.4.2/. You must configure this attribute if the B2B Data Exchange JMS Broker (activemq) component is configured. Default: none Example: `/opt/infahome/B2B/DataExchange /fuse-message-broker-5.4.2`
MFTHome	The complete path of the B2B Managed File Transfer (mft) component. The mft component is usually present along with the B2B Data Exchange installation. You must configure this attribute if the B2B Managed File Transfer (mft) component is configured. Default: none Example: `/opt/infahome/B2B/ManagedFileTransfer`
InfaHome	The complete path of the PowerCenter directory. You must configure this attribute if the PowerCenter (powercenter) component is configured. Default: none Example: `/opt/infahome/Informatica/9.1.0`

Note: For more information about the attributes that the agent uses for the different components: See “Sample resource configurations for Informatica” on page 40.

Table 3-2 lists the optional attributes.

Table 3-2 Optional attributes

Optional attribute	Description
EnvFile	The complete path of the env file that the agent sources to set the environment before starting the server configuration. Veritas recommends storing the file on shared disk. The following shell environments are supported: ksh, sh, and csh. Default: "" Example:""
MonitorProgram	The full path name and command-line arguments for an externally provided monitor program. Default: "" Example: " "
LevelTwoMonitorFreq	This is a type-level attribute that specifies the frequency at which the agent for this resource type must perform second-level or detailed monitoring. You can also override the value of this attribute at the resource level. The value indicates the number of monitor cycles after which the agent will monitor the Informatica in detail. For example, the value 5 indicates that the agent will monitor the Informatica in detail after every five online monitor intervals. Default: 0
IMF	This type-level attribute determines if the agent must perform intelligent resource monitoring. You can also override the value of this attribute at the resource level. See "About the keys of the IMF attribute" on page 27.
IMFRegList	An ordered list of attributes whose values are registered with the IMF notification module. The attribute values can be overriden at the resource level.

About the keys of the IMF attribute

The IMF type-level attribute uses the following keys:

Table 3-3 IMF attribute keys

Key	Description
Mode	Define this attribute to enable or disable intelligent resource monitoring. Valid values are as follows: ■ 0—Does not perform intelligent resource monitoring ■ 1—Performs intelligent resource monitoring for offline resources and performs poll-based monitoring for online resources ■ 2—Performs intelligent resource monitoring for online resources and performs poll-based monitoring for offline resources ■ 3—Performs intelligent resource monitoring for both online and for offline resources. **Note:** The agent for Informatica supports intelligent resource monitoring for online resources only. Hence, Mode should be set to either 0 or 2. Default: 2
MonitorFreq	This key value specifies the frequency at which the agent invokes the monitor agent function. The value of this key is an integer. You can set this key to a non-zero value for cases where the agent requires to perform both poll-based and intelligent resource monitoring. If the value is 0, the agent does not perform poll-based process check monitoring. After the resource registers with the AMF kernel driver, the agent calls the monitor agent function as follows: ■ After every (MonitorFreq x MonitorInterval) number of seconds for online resources ■ After every (MonitorFreq x OfflineMonitorInterval) number of seconds for offline resources Default: 5
RegisterRetryLimit	If you enable intelligent resource monitoring, the agent invokes the imf_register agent function to register the resource with the AMF kernel driver. The value of the RegisterRetryLimit key determines the number of times the agent must retry registration for a resource. If the agent cannot register the resource within the limit that is specified, then intelligent monitoring is disabled until the resource state changes or the value of the Mode key changes. Default: 3

Attributes used in different resource configurations

For each resource configuration, some attributes may be used by the agent and others may not be used. Use the following tables to figure out which attributes must be configured for your resource depending on the required configuration for your resource.

In these tables, the following conventions hold true:

- "Yes" implies that the attribute is mandatory for the given configuration.
- "Opt" implies that configuring the attribute is optional for the given configuration.
- "-" implies that the attribute is not used by the agent for the given configuration.

Table 3-4 shows the attributes used in different resource configurations.

Table 3-4 Attributes used in different resource configurations

Attribute	activemq	dxconsole	dxserver	mft	powercenter
ResLogLevel	Yes	Yes	Yes	Yes	Yes
Component	Yes	Yes	Yes	Yes	Yes
User	Yes	Yes	Yes	Yes	Yes
EnvFile	Opt	Opt	Opt	Opt	Opt
MonitorProgram	Opt	Opt	Opt	Opt	Opt
Hostname	Yes	Yes	Yes	Yes	Yes
Port	Yes	Yes	Yes	Yes	Yes
DXHome	Yes	Yes	Yes	Yes	-
ActiveMQHome	Yes	-	-	-	-
MFTHome	-	-	-	Yes	-
InfaHome	-	-	-	-	Yes
LevelTwoMonitorFreq	Opt	Opt	Opt	Opt	Opt

Enabling and disabling intelligent resource monitoring manually

The intelligent resource monitoring feature is enabled by default. Review the following procedures to enable or disable intelligent resource monitoring manually.

The IMF resource type attribute determines whether an IMF-aware agent must perform intelligent resource monitoring.

To enable intelligent resource monitoring manually

1. Make the VCS configuration writable.

   ```
   # haconf -makerw
   ```

2. Run the following command to enable intelligent resource monitoring of online resources:

   ```
   # hatype -modify Informatica IMF -update Mode 2
   ```

3. If required, change the values of the MonitorFreq key and the RegisterRetryLimit key of the IMF attribute.

4. Save the VCS configuration.

   ```
   # haconf -dump -makero
   ```

5. Restart the agent. Run the following commands on each node.

   ```
   # haagent -stop agent_name -force -sys sys_name
   ```

   ```
   # haagent -start agent_name -sys sys_name
   ```

To disable intelligent resource monitoring manually

1. Make the VCS configuration writable.

   ```
   # haconf -makerw
   ```

2. To disable intelligent resource monitoring for all the resources of a certain type, run the following command:

   ```
   # hatype -modify Informatica IMF -update Mode 0
   ```

3. To disable intelligent resource monitoring for a specific resource, run the following command:

   ```
   # hares -override resource_name IMF
   ```

   ```
   # hares -modify resource_name IMF -update Mode 0
   ```

4. Save the VCS configuration.

   ```
   # haconf -dump -makero
   ```

Chapter 4

Troubleshooting the agent for Informatica

This chapter includes the following topics:

Using the correct software and operating system versions

Ensure that you use correct software and operating system versions.

For information on the software versions that the agent for Informatica supports, see the Veritas Services and Operations Readiness Tools (SORT) site: https://sort.veritas.com/agents.

Meeting prerequisites

Before installing the agent for Informatica, ensure that the following prerequisites are met.

For example, you must install the ACC library on VCS before installing the agent for Informatica.

See "Before you install the Cluster Server agent for Informatica" on page 13.

Starting the Informatica instance outside a cluster

If you face problems while working with a resource, you must disable the resource within the cluster framework. A disabled resource is not under the control of the cluster framework, and so you can test the Informatica instance independent of the cluster framework. Refer to the cluster documentation for information about disabling a resource.

You can then restart the Informatica instance outside the cluster framework.

Note: Use the same parameters that the resource attributes define within the cluster framework while restarting the resource outside the cluster framework.

A sample procedure to start a Informatica instance outside the cluster framework, is illustrated as follows.

To restart Informatica outside the cluster framework

1. Log in as superuser onto the host on which the Informatica component is to run.
2. Use the values defined in the agent attributes to initiate the Informatica start program.

   For example, assume that the B2B Data Exchange JMS Broker (activemq) component is configured, and the following values are assigned:

Component	activemq
User	infa
Hostname	localhost
Port	18095
ActiveMQHome	`/opt/infahome/B2B/DataExchange /fuse-message-broker-5.4.2`
DXHome	`/opt/infahome/B2B/DataExchange`

3. Log in to the Informatica component B2B Data Exchange JMS Broker (activemq) by using the user name specified in the User attribute:

   ```
   # su - infa
   ```

4 Go to the directory specified in the DXHome attribute:

```
# cd /opt/infahome/B2B/DataExchange
```

5 Start the Informatica component:

For B2B Data Exchange JMS Broker (activemq):

```
DXHome/bin/activemq.sh start
```

If the Informatica component starts successfully, an appropriate message is displayed.

6 Enter this command:

```
# exit
```

If the Informatica component works properly outside the cluster framework, you can then attempt to implement the Informatica component within the cluster framework.

Reviewing error log files

If you face problems while using Informatica or the agent for Informatica, use the log files described in this section to investigate the problems.

Using Informatica log files

If an instance of the Informatica component type faces problems, you can access the server log files to further diagnose the problem. The log files are located in `/var/VRTSvcs/log/`.

Reviewing cluster log files

In case of problems while using the agent for Informatica, you can access the engine log file for more information about a particular resource. The engine log file is located at `/var/VRTSvcs/log/engine_A.log`.

For a long running cluster, the log files are rotated as engine_B.log, engine_C.log, and so on. The most-recent engine logs are present in the engine_A.log file.

Reviewing agent log files

In case of problems while using the agent for Informatica, you can access the agent log, `/var/VRTSvcs/log/Informatica_A.log`.

For a long running resource, the logs are rotated as Informatica_B.log, Informatica_C.log, and so on. The most-recent resource logs are present in the `Informatica_A.log` file.

The agent saves the output of every operation process in the temporary folder of the resource system. If the temporary folder is /tmp, the log files are saved using the following naming format:

`/tmp/.VRTS`*`AgentName`*`/`*`ResourceName_EntryPointName`*`.out`

For example:

- `/tmp/.VRTSInformatica/MyDxServer_online.out`
  The output of the start or online operation is redirected to this file.
- `/tmp/.VRTSInformatica/MyDxServer_clean.out`
  The output of the clean operation is redirected to this file.
- `/tmp/.VRTSInformatica/MyDxServer_offline.out`
  The output of the stop or offline operation is redirected to this file.

Note: These files are overwritten each time you execute the corresponding operation process. If you want to save the information, make a copy of the files to another location.

Using trace level logging

The ResLogLevel attribute controls the level of logging that is written in a cluster log file for each Informatica resource. You can set this attribute to TRACE, which enables very detailed and verbose logging.

The LogDbg attribute should be used to enable the debug logs for the ACCLib-based agents when the ACCLIB version is 6.2.0.0 or later and the VCS version is 6.2 or later.

To localize ResLogLevel attribute for a resource

1. Identify the resource for which you want to enable detailed logging.
2. Localize the ResLogLevel attribute for the identified resource:

   ```
   # hares -local Resource_Name ResLogLevel
   ```

3. Set the ResLogLevel attribute to TRACE for the identified resource:

   ```
   # hares -modify Resource_Name ResLogLevel TRACE -sys SysA
   ```

4 Test the identified resource. The function reproduces the problem that you are attempting to diagnose.

5 Set the ResLogLevel attribute back to INFO for the identified resource:

```
# hares -modify Resource_Name ResLogLevel INFO -sys SysA
```

6 Save the configuration changes.

```
# haconf -dump
```

7 Review the contents of the log file.

You can also contact Veritas support for more help.

To enable debug logs for all resources of type Informatica

- Enable the debug log.

```
# hatype -modify Informatica LogDbg DBG_5
```

To override the LogDbg attribute at resource level

- Override the LogDbg attribute at the resource level and enable the debug logs for the specific resource.

```
# hares -override Informatica LogDbg
# hares -modify Informatica LogDbg DBG_5
```

Troubleshooting the configuration for IMF

If you face problems with the IMF configuration or functionality, consider the following:

- Ensure that the following attributes are configured with appropriate values.
  - AgentFile
  - IMF
  - IMFRegList
    If IMFRegList is not configured correctly, the Informatica resources that have been registered for IMF get unregistered every time the monitor function is run.
- If you have configured the required attributes to enable the Informatica agent for IMF, but the agent is still not IMF-enabled, restart the agent. The imf_init function runs only when the agent starts up, so when you restart the agent,

imf_init runs and initializes the Informatica agent to interface with the AMF kernel driver.

- You can run the following command to check the value of the MonitorMethod attribute and to verify that a resource is registered for IMF.

  ```
  # hares -value resource MonitorMethod system
  ```

  The MonitorMethod attribute specifies the monitoring method that the agent uses to monitor the resource:

  - Traditional—Poll-based resource monitoring
  - IMF—Intelligent resource monitoring

- You can use the `amfstat` to see a list of registered PIDs for an Informatica resource.

  The `amfstat` command shows the PIDs monitored by the Informatica agent.

  ```
  Registered Reapers (1):
  =======================
   RID   PID     MONITOR   TRIGG   REAPER
   0     1665    1          0      Informatica

  Process ONLINE Monitors (1):
  ============================
   RID     R_RID   PID    GROUP
   2       0       2896   MyDxServer
  ```

  The agent identifies the process for the Informatica component by applying pattern matching on the output of the `ps -ef` command. The patterns for the different Informatica component processes are:

  activemq `java .*-jar ActiveMQHome/bin/run.jar start`

  dxconsole `java .*-classpath.*DXHome\b.*-Dcatalina.home=DXHome/apache-tomcat-\S+\b.*Bootstrap start`

  dxserver `java .*-classpath .:DXHome:.*com.informatica.b2b.dx.broker.DXControl start`

  mft `java .* com.zerog.lax.LAX MFTHome/VLTraderc.lax`

  powercenter `java.*java\.awt\.headless=true\b.*-DINFA_HOME=$sInfaHome\b.*Bootstrap start\b`

- Run the following command to set the ResLogLevel attribute to TRACE. When you set ResLogLevel to TRACE, the agent logs messages in the Informatica_A.log file.

  ```
  # hares -modify ResourceName ResLogLevel TRACE
  ```

- Run the following command to view the content of the AMF in-memory trace buffer.

  ```
  # amfconfig -p dbglog
  ```

The agent may fail to detect the correct status of the `dxconsole` component

After an online operation, if second-level monitor is enabled, the VCS agent for Informatica may not detect the `dxconsole` component as online. To work around this issue, increase the value of the OnlineWaitLimit attribute so as to allow `dxconsole` sufficient time to come up.

1. Make the VCS configuration writable.

   ```
   # haconf -makerw
   ```

2. Increase the OnlineWaitLimit attribute value for the appropriate resources.

   For example:

   ```
   # hatype -modify Informatica OnlineWaitLimit 4
   ```

3. Save the VCS configuration.

   ```
   # haconf -dump -makero
   ```

Appendix A

Sample Configurations

This appendix includes the following topics:

About sample configurations for the agents for Informatica

The sample configuration graphically depicts the resource types, resources, and resource dependencies within the service group. Review these dependencies carefully before configuring the agents for Informatica. For more information about these resource types, refer to the *Cluster Server Bundled Agents Reference Guide*.

Sample agent type definition for Informatica

This section provides an example of the Informatica agent type definition file:

```
type Informatica (
    static int IMF{} = { Mode=2, MonitorFreq=5, RegisterRetryLimit=3 }
    static str IMFRegList[] = { User, Component, Hostname, Port }
    static boolean AEPTimeout = 1
    static str AgentFile = "/opt/VRTSvcs/bin/Script51Agent"
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Informatica"
    static str ArgList[] = { ResLogLevel, State, IState, User, Component,
```

```
    EnvFil }
    str ResLogLevel = INFO
    str User
    str Component
    str EnvFile
    str MonitorProgram
    str Hostname = localhost
    int Port
    str DXHome
    str ActiveMQHome
    str MFTHome
    str InfaHome
)
```

Sample service group configuration for Informatica

An Informatica resource consists of the following:

Disk Group: Veritas Volume Manager disk group contains information required by the DiskGroup agent to import and export the shared disk object used in support of a clustered Informatica component.

Mount: This resource mounts, monitors, and unmounts the file system that is dedicated to the Informatica installation and configuration files. Use the resource type Mount to create this resource.

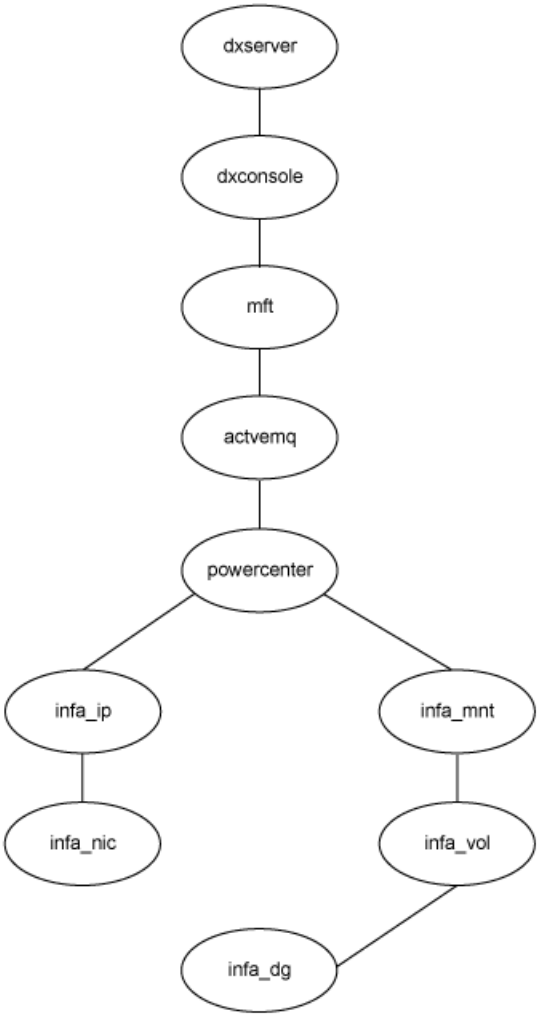
Network Interface: This resource monitors the network interface card through which the Informatica communicates with other services.

Virtual IP: This resource configures the virtual IP address dedicated to the Informatica. External services, programs, and clients use this address to communicate with this Informatica component.

Informatica: This resource starts, stops, and monitors the Informatica component. Use the Informatica resource type to create this resource.

Figure A-1 shows an example of an Informatica service group

Figure A-1 Sample service group configuration



Sample resource configurations for Informatica

The following tables show the sample resource configurations for the different Informatica components:

Table A-1 Sample resource configurations for B2B Data Exchange JMS Broker (activemq)

Attribute	Sample Value
Component	activemq
User	infa
Hostname	localhost
Port	18098
ActiveMQHome	`/opt/infahome/B2B/DataExchange /fuse-message-broker-5.4.2`
DXHome	`/opt/infahome/B2B/DataExchange`
MFTHome	-
InfaHome	-

Table A-2 Sample resource configurations for B2B Data Exchange Operation Console (dxconsole)

Attribute	Sample Value
Component	dxconsole
User	infa
Hostname	localhost
Port	18080
ActiveMQHome	-
DXHome	`/opt/infahome/B2B/DataExchange`
MFTHome	-
InfaHome	-

Table A-3 Sample resource configurations for B2B Data Exchange Server (dxserver)

Attribute	Sample Value
Component	dxserver

Table A-3 Sample resource configurations for B2B Data Exchange Server (dxserver) *(continued)*

Attribute	Sample Value
User	infa
Hostname	localhost
Port	18095
ActiveMQHome	-
DXHome	`/opt/infahome/B2B/DataExchange`
MFTHome	-
InfaHome	-

Table A-4 Sample resource configurations for B2B Managed File Transfer (mft)

Attribute	Sample Value
Component	mft
User	infa
Hostname	localhost
Port	5080
ActiveMQHome	-
DXHome	`/opt/infahome/B2B/DataExchange`
MFTHome	`/opt/infahome/B2B/ManagedFileTransfer`
InfaHome	-

Table A-5 Sample resource configurations for PowerCenter (powercenter)

Attribute	Sample Value
Component	powercenter
User	infa
Hostname	localhost

Table A-5 Sample resource configurations for PowerCenter (powercenter) *(continued)*

Attribute	Sample Value
Port	6005
ActiveMQHome	-
DXHome	-
MFTHome	-
InfaHome	/opt/infahome/Informatica/9.1.0

Sample configuration in a VCS environment

```
include "types.cf"
include "InformaticaTypes.cf"

cluster infa_clus (
 UserNames = { admin = dOPhOJoLPkPPnXPjOM, a = GPPj }
 Administrators = { admin, a }
 )

system nodeA (
 )

system nodeB (
 )

group SG1 (
 SystemList = { nodeA = 0, nodeB = 1  }
 )

 DiskGroup dg_res (
  Critical = 0
  DiskGroup = B2B_dg
  )

 IP ip_res (
  Critical = 0
  Device = eth0
  Address = "10.209.76.140"
```

```
 NetMask = "255.255.252.0"
 )

Informatica activemq (
 Critical = 0
 User = infa
 Component = activemq
 Port = 18098
 DXHome = "/opt/Informatica/B2B/DataExchange"
 ActiveMQHome = "/opt/Informatica/B2B/DataExchange/
 fuse-message-broker-5.4.2"
 )

Informatica dxconsole (
 Critical = 0
 User = infa
 Component = dxconsole
 Port = 18080
 DXHome = "/opt/Informatica/B2B/DataExchange"
 )

Informatica dxserver (
 Critical = 0
 User = infa
 Component = dxserver
 Port = 18095
 DXHome = "/opt/Informatica/B2B/DataExchange"
 )

Informatica mft (
 Critical = 0
 User = infa
 Component = mft
 Port = 5080
 DXHome = "/opt/Informatica/B2B/DataExchange"
 MFTHome = "/opt/Informatica/B2B/ManagedFileTransfer"
 )

Informatica powercenter (
 Critical = 0
 User = infa
 Component = powercenter
 Port = 6005
```

```
 InfaHome = "/home/informatica/Informatica/9.1.0"
 )

Mount mount_res (
 Critical = 0
 MountPoint = "/opt/Informatica"
 BlockDevice = "/dev/vx/dsk/B2B_dg/B2B_dg_vol"
 FSType = vxfs
 FsckOpt = "-y"
 )

NIC nic_res (
 Critical = 0
 Device = eth0
 )

Volume vol_res (
 Critical = 0
 DiskGroup = B2B_dg
 Volume = B2B_dg_vol
 )

activemq requires powercenter
dxconsole requires mft
dxserver requires dxconsole
ip_res requires nic_res
mft requires activemq
mount_res requires vol_res
powercenter requires ip_res
powercenter requires mount_res
vol_res requires dg_res


// resource dependency tree
//
// group SG1
// {
// Informatica dxserver
//     {
//     Informatica dxconsole
//         {
//         Informatica mft
//             {
```

```
//             Informatica activemq
//                 {
//                 Informatica powercenter
//                     {
//                     IP ip_res
//                         {
//                         NIC nic_res
//                         }
//                     Mount mount_res
//                         {
//                         Volume vol_res
//                             {
//                             DiskGroup dg_res
//                             }
//                         }
//                     }
//                 }
//             }
//         }
// }
```

Index