

# Cluster Server Agent for Sybase IQ Installation and Configuration Guide

Linux, Solaris

7.0

Last updated: 2020-07-03

## Legal Notice

Copyright © 2020 Veritas Technologies LLC. All rights reserved.

Veritas and the Veritas Logo are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

This product may contain third-party software for which Veritas is required to provide attribution to the third-party ("Third-Party Programs"). Some of the Third-Party Programs are available under open source or free software licenses. The License Agreement accompanying the Software does not alter any rights or obligations you may have under those open source or free software licenses. Refer to the third-party legal notices document accompanying this Veritas product or available at:

<https://www.veritas.com/about/legal/license-agreements>

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Veritas Technologies LLC and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. VERITAS TECHNOLOGIES LLC SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, et seq. "Commercial Computer Software and Commercial Computer Software Documentation," as applicable, and any successor regulations, whether delivered by Veritas as on premises or hosted services. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Veritas Technologies LLC  
2625 Augustine Drive  
Santa Clara, CA 95054  
<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](mailto:CustomerCare@veritas.com)

Japan

[CustomerCare\\_Japan@veritas.com](mailto: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:

[infoscaledocs@veritas.com](mailto:infoscaledocs@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](https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf)

# Contents

<b>Chapter 1</b>	<b>Introducing the agent for Sybase IQ</b>	<b>6</b>
	About the Cluster Server agent for Sybase IQ	6
	Supported software	6
	Features of the agent	7
	How the agent supports intelligent resource monitoring	7
	Sybase IQ agent functions	8
	Online	8
	Offline	9
	Monitor	9
	Clean	10
<b>Chapter 2</b>	<b>Installing, upgrading, and removing the agent for Sybase IQ</b>	<b>11</b>
	Before you install the Cluster Server agent for Sybase IQ	11
	About the ACC library	12
	Installing the ACC library	12
	Installing the agent in a VCS environment	13
	Installing the agent in a Solaris 10 brand zone	14
	Uninstalling the agent in a VCS environment	15
	Removing the ACC library	15
	Upgrading the agent in a VCS environment	16
<b>Chapter 3</b>	<b>Configuring the agent for Sybase IQ</b>	<b>18</b>
	About configuring the Cluster Server agent for Sybase IQ	18
	Importing the agent types files in a VCS environment	18
	Sybase IQ agent attributes	19
	Executing a customized monitoring program	23
<b>Chapter 4</b>	<b>Enabling the agent for Sybase IQ to support IMF</b>	<b>25</b>
	.....	25
	About Intelligent Monitoring Framework	25
	Benefits of IMF	26
	Agent functions for the IMF functionality	26

	imf_init .....	26
	imf_getnotification .....	26
	imf_register .....	26
	Attributes that enable IMF .....	27
	IMF .....	27
	IMFRegList .....	28
	Before you enable the agent to support IMF .....	28
	Enabling the agent to support IMF .....	28
	If VCS is in a running state .....	29
	If VCS is not in a running state .....	31
	Disabling intelligent resource monitoring .....	31
	Sample IMF configurations .....	32
<b>Chapter 5</b>	<b>Configuring the service groups for Sybase IQ using the CLI .....</b>	<b>34</b>
	About configuring service groups for Sybase IQ .....	34
	Before configuring the service groups for Sybase IQ .....	35
	Sybase IQ entities in a clustered environment .....	35
	Virtualizing Sybase IQ .....	36
<b>Chapter 6</b>	<b>Troubleshooting the agent for Sybase IQ .....</b>	<b>37</b>
	Using the correct software and operating system versions .....	37
	Meeting prerequisites .....	37
	Verifying virtualization .....	38
	Starting the Sybase IQ server outside a cluster .....	38
	Reviewing error log files .....	39
	Using trace level logging .....	39
	Troubleshooting the configuration for IMF .....	41
	Known issues .....	42
<b>Appendix A</b>	<b>Sample Configurations .....</b>	<b>43</b>
	About sample configurations for the agents for Sybase IQ .....	43
	Sample agent type definition for Sybase IQ .....	43
	Sample configuration files .....	44
	Sample service group configurations for SybaseIQ .....	45

# Introducing the agent for Sybase IQ

This chapter includes the following topics:

- [About the Cluster Server agent for Sybase IQ](#)
- [Supported software](#)
- [Features of the agent](#)
- [How the agent supports intelligent resource monitoring](#)
- [Sybase IQ agent functions](#)

## About the Cluster Server agent for Sybase IQ

The Cluster Server (VCS) agent for Sybase IQ manages the Sybase IQ servers in a clustered environment and provides high availability for all the Sybase IQ servers in the cluster. The agent can bring a specific Sybase IQ server instance online and monitor the state of the Sybase IQ server. The agent can also detect failures and can shut down the instance in case of a failure.

## Supported software

For information on the software versions that the Cluster Server agent for Sybase IQ 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 Sybase IQ:

- 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.  
See [“About Intelligent Monitoring Framework”](#) on page 25.
- Delayed agent function  
The agent manages the first monitor after online for slow initializing applications.

## 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 Sybase IQ registers the PIDs of the Sybase IQ 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.

For more information, see the *Cluster Server Administrator's Guide*.

## Sybase IQ agent functions

The operations or functions that the Cluster Server agent for Sybase IQ can perform are as follows:

### Online

The online function performs the following tasks:

- Verifies that the required attributes are set correctly.
- Verifies that the Sybase IQ server instance is not already online. If the instance is online, the online operation exits immediately.
- Kills any Sybase IQ processes that remain, by using the user name associated with the specific resource.
- Attempts to start the Sybase IQ server instance with the command:

```
SybIQBinDir/start_iq StartOpts ServerName SybIQDBConfFile
SybIQDB
```

The command always gets executed in the context of `SyblQUser`.

- Depending on the attribute values provided, attempts to start the SybaseIQ server instance with any of the following commands:
  - If the values of the `StartOpts`, `SybIQDBConfFile`, and `SybIQDB` attributes are provided: `SybIQBinDir/start_iq StartOpts -n ServerName @SybIQDBConfFile SybIQDB`.
  - If only the value of the `StartOpts` attribute is provided: `SybIQBinDir/start_iq StartOpts -n ServerName`.
- Starts the Sybase IQ server in `system.slice`, if the `UseSystemD` attribute is set on the supported Linux platforms, by using the following command:
 

```
# systemctl start serviceFileName
```
- Checks whether the server has started up completely.
- Gives the control back to HAD.



## Offline

The offline function performs the following tasks:

- Verifies that the required attributes are set correctly.
- Verifies that the Sybase IQ server instance is not offline.
- If the instance is already offline, the operation verifies if any processes belonging to this Sybase IQ resource exist.
- Attempts to stop the Sybase IQ server instance with the command:

- If the SybIQDB attribute is configured:

```
SybIQBinDir/dbstop ServerName SybIQDBAdmin  
SybIQDBAdminPasswd SybIQDB
```

- If the SybIQDB attribute is not configured:

```
SybIQBinDir/stop_iq -stop one
```

These commands always gets executed in the context of SybIQUser.

- Gives the control back to HAD.

## Monitor

The monitor function monitors the states of the Sybase IQ servers on all nodes within the cluster. The operation performs the following tasks:

- The monitor function conducts a first-level check to determine that the Sybase IQ server processes are running on the system in the cluster. If the first-level check does not find these processes running on the node, the check exits immediately and reports the instance as OFFLINE.

The agent for Sybase IQ also supports Intelligent Monitoring Framework (IMF) in the first-level check. IMF enables intelligent resource monitoring. The agent for Sybase IQ is IMF-aware and uses the asynchronous monitoring framework (AMF) kernel driver for resource state change notifications. See [“How the agent supports intelligent resource monitoring”](#) on page 7. You can use the MonitorFreq key of the IMF attribute to specify the frequency at which the agent invokes the monitor function. See [“MonitorFreq”](#) on page 27.

- If the SecondLevelMonitor attribute is set to greater than 0, the monitor operation conducts a second level check. During Second Level Monitoring, the agent uses the monitor command to verify that the SybaseIQ server is really up:

```
/DBISQL/bin/dbisql -c  
"uid=SybIQDBAdmin;pwd=SybIQDBAdminPasswd;eng=ServerName;  
dbf=SybIQDb;links=tcip"-nogui "describe SYSIQINFO"
```

The command is executed in the context of a SybaseIQ user, who has the privilege to monitor the SybaseIQ process.

---

**Note:** The attribute used to configure the second-level check and its frequency depends on the software versions of VCS you have installed: For VCS 5.1 SP1 or later with SybaseIQ agent version 5.1.0.0, use the `LevelTwoMonitorFreq` attribute. For VCS 5.1 or earlier with SybaseIQ agent 5.1.0.0, use the `SecondLevelMonitor` attribute.

---

- Depending upon the value of the `MonitorProgram` attribute, the monitor operation can perform a customized check using a user-supplied monitoring utility.

## Clean

In case of a failure or after an unsuccessful attempt to bring a Sybase IQ server instance online or take a Sybase IQ server instance offline, the clean operation performs the following tasks:

- Attempts to gracefully shut down the Sybase IQ server instance with the command:

```
SybIQBinDir/dbstop ServerName SybIQDBAdmin  
SybIQDBAdminPasswd SybIQDB
```

The command always gets executed in the context of `SybIQUser`.

- The clean operation kills any remaining processes pertaining to this Sybase IQ instance.
- Gives the control back to HAD.

---

**Note:** For information about the additional functions of the agent for Sybase IQ when IMF is enabled: See [“Agent functions for the IMF functionality”](#) on page 26.

---

# Installing, upgrading, and removing the agent for Sybase IQ

This chapter includes the following topics:

- [Before you install the Cluster Server agent for Sybase IQ](#)
- [About the ACC library](#)
- [Installing the ACC library](#)
- [Installing the agent in a VCS environment](#)
- [Uninstalling the agent in a VCS environment](#)
- [Removing the ACC library](#)
- [Upgrading the agent in a VCS environment](#)

## Before you install the Cluster Server agent for Sybase IQ

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

Before you install the agent for Sybase IQ, 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 “[About the ACC library](#)” on page 12.

## 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.

Linux            `cd1/linux/generic/vcs/application/acc_library/version_library/rpms`

Solaris        `cd1/solaris/dist_arch/vcs/application/acc_library/version_library/pkggs`

- 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.

```
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 agent in a VCS environment

Install the agent for Sybase IQ 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.

```
Linux      cd1/linux/generic/vcs/application/sybase_agent/  
           vcs_version/version_agent/rpms  
  
Solaris    cd1/solaris/dist_arch/vcs/application/sybase_agent/  
           vcs_version/version_agent/pkg
```

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.

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

Solaris        # pkgadd -d . VRTSsybiq
```

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

See [“Importing the agent types files in a VCS environment”](#) on page 18.

## Installing the agent in a Solaris 10 brand zone

To install the Sybase IQ 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 . VRTSsybiq
```

For example: # pkgadd -R /zones/zone1/root -d . VRTSsybiq

---

**Note:** You can ignore the following messages that might appear:

```
## Executing postinstall script.
```

```
ln: cannot create /opt/VRTSagents/ha/bin/SybaseIQ/imf_getnotification:
File exists
```

```
ln: cannot create /opt/VRTSagents/ha/bin/SybaseIQ/imf_register: File
exists
```

```
or ## Executing postinstall script.
```

```
ln: cannot create /opt/VRTSagents/ha/bin/SybaseIQ/imf_getnotification:
No such file or directory
```

```
ln: cannot create /opt/VRTSagents/ha/bin/SybaseIQ/imf_register: No
such file or directory
```

---

# Uninstalling the agent in a VCS environment

You must uninstall the agent for Sybase IQ 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 Sybase IQ resources from the cluster. Run the following command to verify that all resources have been removed:

```
# hares -list Type=SybaseIQ
```

- 4 Remove the agent type from the cluster configuration by running the following command from any node in the cluster:

```
# hatype -delete SybaseIQ
```

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 Sybase IQ from each node in the cluster.

Run the following command to uninstall the agent:

```
Linux          # rpm -e VRTSsybiq
```

```
Solaris        # pkgrm VRTSsybiq
```

## 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:

```
Linux          # rpm -e VRTSacclib
Solaris        # pkgrm VRTSacclib
```

## Upgrading the agent in a VCS environment

Perform the following steps to upgrade the agent with minimal disruption, in a VCS environment.

**To upgrade the agent in a VCS environment**

- 1 Persistently freeze the service groups that host the application.  

```
# hagr -freeze groupName -persistent
```
- 2 Stop the Sybase IQ agent on all the nodes.  

```
# haagent -stop SybaseIQ -force -sys systemName
```
- 3 Ensure that the agent is stopped on all nodes:  

```
# haagent -display SybaseIQ -sys systemName
```
- 4 Uninstall the agent package from all the nodes. Use the platform's native software management program to remove the agent for Sybase IQ from each node in the cluster.

Run the following command to uninstall the agent:

```
Linux          # rpm -e VRTSsybiq
Solaris        For Solaris 10:
                # pkgrm VRTSsybiq
```

- 5 Install the new agent on all the nodes.

See [“Installing the agent in a VCS environment”](#) on page 13.

The `UpdateTypes.pl` script runs automatically on successful installation of the agent.



- 6 Check for the changes in the resource values required, if any, due to the new agent types file.

---

**Note:** To note the list of changed attributes, compare the new type definition file with the old type definition file.

---

- 7 Start the agent on all nodes, if not started.

```
# haagent -start SybaseIQ -sys systemName
```

- 8 Ensure that the agent is started on all the nodes.

```
# haagent -display SybaseIQ -sys systemName
```

- 9 Unfreeze the service groups after all the resources come to an online steady state.

```
# hagrps -unfreeze groupName -persistent
```

# Configuring the agent for Sybase IQ

This chapter includes the following topics:

- [About configuring the Cluster Server agent for Sybase IQ](#)
- [Importing the agent types files in a VCS environment](#)
- [Sybase IQ agent attributes](#)
- [Executing a customized monitoring program](#)

## About configuring the Cluster Server agent for Sybase IQ

After installing the Cluster Server agent for Sybase IQ, 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 Sybase IQ resources.

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

See [“About sample configurations for the agents for Sybase IQ”](#) on page 43.

## Importing the agent types files in a VCS environment

To use the agent for Sybase IQ, 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 Java GUI

- 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:

VCS 5.1 or later	Linux	/etc/VRTSagents/ha/conf/SybaseIQ/ SybaseIQTypes.cf
VCS 5.1 or later	Solaris	/etc/VRTSagents/ha/conf/SybaseIQ/ SybaseIQTypes51.cf

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

You can now create Sybase IQ 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 CLI

- 1 If VCS is running, run the  
/etc/VRTSagents/ha/conf/SybaseIQ/SybaseIQTypes.cmd file from the command line.
- 2 If VCS is not running, perform the following steps sequentially:
  - Copy the agent types file from /etc/VRTSagents/ha/conf/SybaseIQ/SybaseIQTypes.cf to the /etc/VRTSvcsvcs/conf/config directory.
  - Include the agent types file in the main.cf file.
  - Start HAD.

# Sybase IQ agent attributes

Refer to the required and optional attributes while configuring the agent for Sybase IQ.

Table 3-1 lists the required attributes for the Sybase IQ agent.

**Table 3-1** Required attributes

Attribute	Description
ResLogLevel	<p>Specifies the logging detail that the agent performs for the resource.</p> <p>The valid values are as follows:</p> <ul style="list-style-type: none"> <li>■ ERROR: Only logs error messages.</li> <li>■ WARN: Logs error messages and warning messages.</li> <li>■ INFO: Logs error messages, warning messages, and informational messages</li> <li>■ TRACE: Logs error messages, warning messages, informational messages, and trace messages. TRACE is very verbose and should be used only during initial configuration or for troubleshooting and diagnostic operations.</li> </ul> <p>Default Value: INFO</p> <p>Example: <b>INFO</b></p>
SyblQUser	<p>The user name that is required to run Sybase IQ commands.</p> <p>Default value: No default value</p> <p>Example: <b>root, sybase</b></p>
SyblQBinDir	<p>Specifies the directory for SybaseIQ15.x where the IQ Server binary start_iq is located.</p> <p>Default value: No default value</p> <p>Example: <b>/opt/sybase/IQ-15_3/bin64</b></p>
ServerName	<p>Specifies the logical name or the nickname which uniquely identifies the SybaseIQ server instance.</p> <p>Default value: No default value</p> <p>Example: <b>myserver_iqdemo</b></p>
EnvFile	<p>Full path to the env file that the agent sources to set the environment before executing any SybaseIQ server commands.</p> <p>Veritas recommends storing the file on shared disk. The following shell environments are supported: ksh, sh, and csh.</p> <p>This attribute is required only if second-level monitoring is enabled.</p> <p>Default value: No default value</p> <p>Example: <b>/opt/sybase/IQ-15_3/ IQ-15_3.sh</b></p>

**Table 3-1** Required attributes (*continued*)

Attribute	Description
SyblQDBAdmin	<p>The Sybase database administrator that has all the privileges to the database and can stop the SybaseIQ database server instance using the <code>dbstop</code> command.</p> <p>Default value: No default value</p> <p>Example: <b>DBA</b></p>
SyblQDBAdminPasswd	<p>The Sybase database administrator password.</p> <p>Default value: No default value</p> <p>Example: <b>sql</b> (The password is encrypted.).</p>

[Table 3-2](#) lists the optional attributes for the Sybase IQ agent.

**Table 3-2** Optional attributes

Attribute	Description
StartOpts	<p>Specifies the start-up option, other than <code>-n</code>, that is used to start the SybaseIQ database server outside of VCS control.</p> <p>Configure this attribute if you start the Sybase IQ database server outside of VCS control with options such as <code>-r</code> or <code>-m</code>.</p> <p>If the Sybase IQ server is started outside VCS, make sure that the order of options provided for StartOpts is the same as the order seen in the SybaseIQ process.</p> <p>Default value: No default value</p> <p>Example: <b>-r</b> starts the SybaseIQ server instance in read-only mode. If no option is specified, the server starts in write mode.</p>
SyblQDB	<p>Specifies the complete path of the Sybase database for in-depth monitoring.</p> <p>Default value: No default value</p> <p>Example: <b>/opt/sybase/IQ-15_3/demo/iqdemo.db</b></p>
SyblQDBConfFile	<p>Complete path of the Sybase database configuration file containing the server-name and the TCP-IP Port .</p> <p>Default value: No default value</p> <p>Example: <b>/opt/sybase/IQ-15_3/demo/iqdemo.cfg</b></p>

**Table 3-2** Optional attributes (*continued*)

Attribute	Description
MonitorProgram	<p>Absolute path name of an external, user-supplied monitor executable.</p> <p>For information about setting this attribute:</p> <p>See <a href="#">“Executing a customized monitoring program”</a> on page 23.</p> <p>Default value: No default value</p> <p>Example 1.: <b>ServerRoot/bin/myMonitor.pl</b></p> <p>Example 2.: <b>ServerRoot/bin/myMonitor.sh arg1 arg2</b></p>
SecondLevelMonitor	<p>Used to enable second-level monitoring and specify how often it is run. Second-level monitoring is a deeper, more thorough state check of the configured SybaseIQ instance.</p> <p>The numeric value specifies how often the second-level monitoring routines are run. For example, if the MonitorInterval is set to 60 seconds, and the SecondLevelMonitor is set to 100, then the second level check would only get performed every 100 minutes. To provide maximum flexibility, the value set is not checked for an upper limit. You can set the second level check to occur once a month, if that is desired.</p> <p>Care should be taken when setting this attribute to large numbers.</p> <p><b>Note:</b> The SecondLevelMonitor attribute is applicable to VCS versions earlier than VCS 5.1 SP1. From VCS version 5.1 SP1 onwards, the SecondLevelMonitor attribute is deprecated. Instead, a resource type level attribute LevelTwoMonitorFreq should be used to specify the frequency of in-depth monitoring.</p> <p>Default value: 0</p> <p>Example: <b>1</b></p>
LevelTwoMonitorFreq	<p>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 Sybase IQ server in detail.</p> <p>For example, the value 5 indicates that the agent will monitor the Sybase IQ server in detail after every five online monitor intervals.</p> <p>Type and dimension: integer-scalar</p> <p>Default value: 0</p> <p>Example: <b>2</b></p>

**Table 3-2** Optional attributes (*continued*)

Attribute	Description
UseSystemD	<p>SystemD is a system and a service manager for Linux operating systems. It helps manage applications across Linux distributions that support the SystemD feature. The UseSystemD attribute is applicable only on those systems that use the supported distributions of RHEL and SLES.</p> <ul style="list-style-type: none"> <li>■ When this attribute is set to 1, the Sybase IQ agent uses the Sybase IQ service file for the online operation. The SybaseIQ resource comes online as a service in system.slice.</li> <li>■ When this attribute is set to 0, a typical online function starts the resource in user.slice.</li> </ul> <p>Type and dimension: boolean-scalar</p> <p>Default value: 0</p> <p>Example: 1</p>

**Note:** For information about the additional attributes of the agent for Sybase IQ when IMF is enabled: See [“Attributes that enable IMF”](#) on page 27.

## Executing a customized monitoring program

You can configure the monitor function to execute MonitorProgram. MonitorProgram is a custom monitor utility to perform a user-defined Sybase IQ server state check.

The utility is executed in the context of the UNIX user that is defined in the SybIQUser attribute.

The monitor operation executes MonitorProgram if:

- The MonitorProgram attribute value is set to a valid executable utility.
- The first-level process check indicates that the Sybase IQ server instance is online.
- The second-level monitor check indicates that the SybaseIQ server instance is online.
- The second-level monitor check is deferred for this monitoring cycle.

The monitor operation interprets the program exit code as follows:

110 or 0

Sybase IQ server is online

100 or 1	Sybase IQ server is offline
Any other value	Sybase IQ server state is unknown



# Enabling the agent for Sybase IQ to support IMF

This chapter includes the following topics:

- [About Intelligent Monitoring Framework](#)
- [Agent functions for the IMF functionality](#)
- [Attributes that enable IMF](#)
- [Before you enable the agent to support IMF](#)
- [Enabling the agent to support IMF](#)
- [Disabling intelligent resource monitoring](#)
- [Sample IMF configurations](#)

## About Intelligent Monitoring Framework

With the IMF feature, 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. You can enable or disable the intelligent resource monitoring functionality of the Sybase IQ agent.

VCS process and mount-based agents use the AMF kernel driver that provides asynchronous event notifications to the agents that are enabled for IMF.

You can enable the Sybase IQ agent for IMF, provided the following software versions are installed:

- Cluster Server (VCS) 5.1 SP1 or later
- Cluster Server agent for Sybase IQ version 5.1.0.0 or later

Refer to the *Cluster Server Administrator's Guide* for more information about IMF notification module functions and administering the AMF kernel driver.

## Benefits of IMF

IMF offers the following benefits:

- **Performance**  
Enhances performance by reducing the monitoring of each resource at a default of 60 seconds for online resources, and 300 seconds for offline resources. IMF enables the agent to monitor a large number of resources with a minimal effect on performance.
- **Faster detection**  
Asynchronous notifications would detect a change in the resource state as soon as it happens. Immediate notification enables the agent to take action at the time of the event.

## Agent functions for the IMF functionality

If the Sybase IQ agent is enabled for IMF support, the agent supports the following functions, in addition to the functions mentioned in the Sybase IQ agent functions topic.

### imf\_init

This function initializes the Sybase IQ agent to interface with the AMF kernel driver, which is the IMF notification module for the agent for Sybase IQ. This function runs when the agent starts up.

### imf\_getnotification

This function gets notifications about resource state changes. This function runs after the agent initializes with the AMF kernel module. This function continuously waits for notification and takes action on the resource upon notification.

### imf\_register

This function registers or unregisters resource entities with the AMF kernel module. This function runs for each resource after the resource goes into a steady state—online or offline.

# Attributes that enable IMF

If the agent for Sybase IQ is enabled for IMF support, the agent uses type-level attributes in addition to the agent-specific attributes.

## IMF

This resource type-level attribute determines whether the Sybase IQ agent must perform intelligent resource monitoring. You can also override the value of this attribute at the resource level.

This attribute includes the following keys:

### 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 Sybase IQ supports intelligent resource monitoring for online resources only. Hence, Mode should be set to either 0 or 2.

---

Type and dimension: integer-association

Default: 0 for VCS 5.1 SP1, 3 for VCS 6.0 and later.

### MonitorFreq

This key value specifies the frequency at which the agent invokes the monitor agent function. The value of this key is an integer.

Default: 1

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

## 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.

## IMFRegList

An ordered list of attributes whose values are registered with the IMF notification module.

Type and dimension: string-vector

Default: No default value

---

**Note:** The attribute values can be overridden at the resource level.

---

# Before you enable the agent to support IMF

Before you enable the Sybase IQ agent to support IMF, ensure that the AMF kernel module is loaded and AMF is configured. For details, refer to the 'Administering the AMF kernel driver' section of the *Cluster Server Administrator's Guide*. For details about the commands you can configure AMF using the `amfconfig -h` command.

## Enabling the agent to support IMF

In order to enable the Sybase IQ agent to support IMF, you must make the following configuration changes to the attributes of the agent:

- `AgentFile`: Set the `AgentFile` attribute to **Script51Agent**

- IMF Mode: Set the IMF Mode attribute to **2**
- IMFRegList: Update the IMFRegList attribute

The following sections provide more information about the commands you can use to make these configuration changes, depending on whether VCS is in a running state or not.

---

**Note:** If you have upgraded VCS from an earlier version to version 5.1 SP1 or later, and you already have Sybase IQ agent 5.1.00 or later installed, ensure that you run the following commands to create appropriate symbolic links:

```
# cd /opt/VRTSagents/ha/bin/SybaseIQ
# ln -s /opt/VRTSamf/imf/imf_getnotification imf_getnotification
# ln -s /opt/VRTSagents/ha/bin/SybaseIQ/monitor imf_register
```

---

## If VCS is in a running state

**To enable the Sybase IQ resource for IMF when VCS is in a running state:**

- 1** Make the VCS configuration writable.

```
# haconf -makerw
```

- 2** Run the following command to update the AgentFile attribute.

```
# hatype -modify SybaseIQ AgentFile\
/opt/VRTSvc/bin/Script51Agent
```

- 3** For VCS version 6.0 or later, run the following commands to add the IMF attributes:

```
# haattr -add -static SybaseIQ IMF -integer -assoc Mode 0 \
MonitorFreq 1 RegisterRetryLimit 3

# haattr -add -static SybaseIQ IMFRegList -string -vector
```

---

**Note:** Run these commands only once after you first enable IMF support for the agent.

---

- 4 Run the following command to update the IMF attribute.

```
# hatype -modify SybaseIQ IMF Mode num MonitorFreq num  
RegisterRetryLimit num
```

For example, to enable intelligent monitoring of online resources, with the MonitorFreq key set to 5, and the RegisterRetryLimit key is set to 3, run the following command:

```
# hatype -modify SybaseIQ IMF Mode 2 MonitorFreq 5 \  
RegisterRetryLimit 3
```

---

**Note:** The valid values for the Mode key of the IMF attribute are 0 (disabled) and 2 (online monitoring).

---

- 5 Run the following command to update the IMFRegList attribute:

```
# hatype -modify SybaseIQ IMFRegList ServerName EnvFile SybIQUser  
SybIQBinDir
```

- 6 Save the VCS configuration.

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

- 7 If the Sybase IQ agent is running, restart the agent.

For information on the commands you can use to restart the agent, see [Restarting the agent](#).

## Restarting the agent

To restart the agent:

- 1 Run the following command to stop the agent forcefully:

```
# haagent -stop SybaseIQ -force -sys <system>
```

---

**Note:** Stopping the agent forcefully eliminates the need to take the resource offline.

---

- 2 Run the following command to start the agent:

```
# haagent -start SybaseIQ -sys <system>.
```

## If VCS is not in a running state

**To change the SybaseIQ type definition file when VCS is not in a running state:**

- 1 Update the AgentFile attribute.

```
static str AgentFile = "/opt/VRTSvcs/bin/Script51Agent"
```

- 2 Update the IMF attribute.

The valid values for the Mode key of the IMF attribute are 0 (disabled) and 2 (online monitoring).

```
static int IMF{} = { Mode=num, MonitorFreq=num,  
RegisterRetryLimit=num }
```

For example, to update the IMF attribute such that the Mode key is set to 2, the MonitorFreq key is set to 5, and the RegisterRetryLimit key is set to 3:

```
static int IMF{} = { Mode=2, MonitorFreq=5, RegisterRetryLimit=3  
}
```

- 3 Update the IMFRegList attribute.

```
static str IMFRegList[] = { ServerName, EnvFile, SybIQUser,  
SybIQBinDir }
```

## Disabling intelligent resource monitoring

**To disable intelligent resource monitoring**

- 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 SybaseIQ 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
```

## Sample IMF configurations

An example of a type definition file for a Sybase IQ agent that is IMF-enabled is as follows. In this example, the IMF-related attributes are set to the following values:

- AgentFile /opt/VRTSvcs/bin/Script51Agent
- IMF{ { Mode=2, MonitorFreq=5, RegisterRetryLimit RegisterRetryLimit=3 }
- IMFRegList[] { ServerName EnvFile SybIQUser SybIQBinDir }
- LevelTwoMonitorFreq 25

```
type SybaseIQ (
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/SybaseIQ"
    static str AgentFile = "/opt/VRTSvcs/bin/Script51Agent"
    static str ArgList[] = { ResLogLevel, State, IState, SybIQUser,
        EnvFile, SybIQBinDir, ServerName, StartOpts, SybIQDB,
        SybIQDBConfFile, SybIQDBAdmin, SybIQDBAdminPasswd,
        SecondLevelMonitor, MonitorProgram }
    static boolean AEPTIMEOUT = 1
    static int IMF{ } = { Mode=2, MonitorFreq=5, RegisterRetryLimit=3 }
    static str IMFRegList[] = { ServerName, EnvFile, SybIQBinDir,
        SybIQUser }
    str ResLogLevel = INFO
    str SybIQUser
    str EnvFile
    str SybIQBinDir
    str ServerName
    str StartOpts
    str SybIQDB
    str SybIQDBConfFile
    str SybIQDBAdmin
    str SybIQDBAdminPasswd
    int SecondLevelMonitor
    str MonitorProgram
)
```

A sample resource configuration from the /etc/VRTSvcs/conf/config/main.cf file is as follows:

```
group svg_SybaseIQ (
    SystemList = { galaxy = 0, nebula = 1 }
)

SybaseIQ SybaseIQ_Res1 (
```



```
Critical = 0
ResLogLevel = TRACE
SybIQUser = root
EnvFile = "/opt/sybase/IQ-15_3/IQ-15_3.sh"
SybIQBinDir = "/opt/sybase/IQ-15_3/bin64"
ServerName = ABC
StartOpts = "-r"
SybIQDB = "/opt/sybase/IQ-15_3/demo/iqdemo.db"
SybIQDBConfFile = "/opt/sybase/IQ-15_3/demo/iqdemo.cfg"
SybIQDBAdmin = DBA
SybIQDBAdminPasswd = GUJuHSs
```

)

# Configuring the service groups for Sybase IQ using the CLI

This chapter includes the following topics:

- [About configuring service groups for Sybase IQ](#)
- [Before configuring the service groups for Sybase IQ](#)
- [Sybase IQ entities in a clustered environment](#)
- [Virtualizing Sybase IQ](#)

## About configuring service groups for Sybase IQ

Configuring the Sybase IQ service group involves creating the Sybase IQ service group, its resources, and defining attribute values for the configured resources. 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

# Before configuring the service groups for Sybase IQ

Before you configure the Sybase IQ service group, you must:

- Verify that the Cluster Server components are installed and configured on all nodes in the cluster where you will configure the service group.  
For more information on installing the components, refer to the *InfoScale Availability Installation Guide*.
- Verify that the Cluster Server agent for Sybase IQ is installed on all nodes in the cluster.

## Sybase IQ entities in a clustered environment

A service group is a logical setup containing all resources that can support a Sybase IQ instance in a clustered environment.

The required resources are as follows.

Disk group	<p>Contains a volume and a file system, which is a mount resource containing the Sybase IQ installation files.</p> <p>Use the DiskGroup resource type to create this resource. Create the disk group from the shared disk so that you can import the group into any system in the cluster.</p>
Mount	<p>Mounts, monitors, and unmounts the file system that is dedicated to the Sybase IQ installation files.</p> <p>Use the Mount resource type to create this resource.</p>
Network interface	<p>Monitors the network interface card through which the Sybase IQ instance communicates with other services.</p> <p>Use the NIC resource type to create this resource.</p>
Virtual IP	<p>Configures the virtual IP address dedicated to the Sybase IQ instance. The external services, programs, and clients use this address to communicate with this instance.</p> <p>Use the IP resource type to create this resource.</p>
Sybase IQ server	<p>Starts, stops, and monitors the Sybase IQ server instance.</p> <p>Use the Sybase IQ server resource type to create this resource.</p>

# Virtualizing Sybase IQ

To ensure that your Sybase IQ machine can function properly on any node of the cluster, you need to virtualize all the parameters that could be dependent on a particular node.

Review the following basic notes for virtualization:

Host names	When installing and configuring the Sybase IQ machine, ensure that you enter the virtual host name associated with the IP address used to configure the IP resource. This ensures that if the application needs to be migrated, you are not tied down by the physical IP address given to the Sybase IQ machine.
Path names	Ensure that your application gets installed on a shared disk so that it is not constrained by anything that is local to the node. If this is not possible every time, make sure that the local data is available on each configured node.

# Troubleshooting the agent for Sybase IQ

This chapter includes the following topics:

- [Using the correct software and operating system versions](#)
- [Meeting prerequisites](#)
- [Verifying virtualization](#)
- [Starting the Sybase IQ server outside a cluster](#)
- [Reviewing error log files](#)
- [Troubleshooting the configuration for IMF](#)

## 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 Sybase IQ supports, see the Veritas Services and Operations Readiness Tools (SORT) site:

<https://sort.veritas.com/agents>.

## Meeting prerequisites

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

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

See [“Before you install the Cluster Server agent for Sybase IQ”](#) on page 11.

## Verifying virtualization

Verify that your application does not use anything that ties it down to a particular node of the cluster.

See [“Virtualizing Sybase IQ ”](#) on page 36.

## Starting the Sybase IQ server 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 Sybase IQ database server independent of the cluster framework. Refer to the cluster documentation for information about disabling a resource.

---

**Note:** Use the same parameters that the resource attributes defined within the cluster framework while restarting the resource outside the framework, like the owner of the application, the environment file etc.

---

- Starting the Sybase IQ server

To start the Sybase IQ server outside cluster, execute:

```
$ SybIQBinDir/start_iq startOptions serverName
SybIQDBConfFile SybIQDB
```

- Stopping the Sybase IQ server

To stop the Sybase IQ server outside cluster, execute:

```
$ SybIQBinDir/dbstop ServerName SybIQDBAdmin
SybIQDBAdminPasswd SybIQDB
```

- Monitoring the Sybase IQ server

First verify that the Sybase IQ processes are running as SybIQUser.

- The agent uses the following monitor command to verify that the Sybase IQ server is up.

```
$ SybIQBinDir/dbisql -c "uid=SybIQDBAdmin;pwd=SybIQDBAdminPasswd;
eng=serverName;links=tcPIP" -nogui "describe SYSIQINFO"
```

## Reviewing error log files

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

The common reasons for issues are as follows:

Incorrect port, environment or parameter settings	<p>Verify that ports have been properly configured and declared. Typically, ports from 1 through 1024 are reserved for the superuser.</p> <p>Ensure that parameters to the agent are correctly defined.</p>
Expired licenses	<p>Check the application log files for any error messages related to expired licenses.</p> <p>Ensure that the license keys/files have been placed at the appropriate location, as needed by the application.</p>
Broken symlinks, missing files, and libraries	<p>Verify your installation.</p> <p>Make sure that nothing is broken, and all dependencies for the executables are met.</p>
Insufficient disk space or system parameters	<p>Ensure that the file-system has sufficient space for creation of temporary files that the application might need.</p> <p>Verify that the kernel has been tuned for sufficient IPC resources, file descriptors and meets the hardware requirement. Consult your product documentation for these details.</p>

## Using trace level logging

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

If you set ResLogLevel to TRACE, a very high volume of messages are produced. Veritas recommends that you localize the ResLogLevel attribute for a particular resource.

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 Note the time before you begin to operate the identified resource.
- 5 Test the identified resource. The function reproduces the problem that you are attempting to diagnose.
- 6 Note the time when the problem is reproduced.
- 7 Set the ResLogLevel attribute back to INFO for the identified resource:

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

- 8 Save the configuration changes.

```
# haconf -dump
```

- 9 Review the contents of the log file.

Use the time noted in the previous steps to diagnose the problem.

You can also contact Veritas support for more help.

### To enable debug logs for all resources of type SybaseIQ

- ◆ Enable the debug log.

```
# hatype -modify SybaseIQ 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 SybaseIQ LogDbg
```

```
# hares -modify SybaseIQ 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 Sybase IQ 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 Sybase IQ 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 Sybase IQ 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 a SybaseIQ resource. A sample output of the `ps -ef` command for the SybaseIQ processes is as follows:

```
sybase  28592      1  0 10:13 ?          00:00:00 /home/sybase
/base/IQ-15_3/bin64/iqsrv15 -r -n Oxford_Server -c 48m -gc 20 -gd all
-g1 all -gm 10 -gp 4096 -ti 4400 -gn 25 -o /home/sybase/base/IQ-15_3/
logfiles/Oxford_Server.0001.srvlog -hn 7
sybase  29058      1  2 10:13 ?          00:00:01 /home/sybase
/base/IQ-15_3/bin64/iqsrv15 -r -n ABC_Demo @/home/sybase/base/IQ-15_3/
demo/iqdemo.cfg /home/sybase/base/IQ-15_3/demo/iqdemo.db -ti 4400 -gn
25 -o /home/sybase/base/IQ-15_3/logfiles/ABC_Demo.0001.srvlog -hn 7
```

The `amfstat` command shows the PIDs monitored by the SybaseIQ Server agent.

```
Registered Reapers (2):
=====
RID      PID      EVENT    REAPER
1         28715    2         0         SybaseIQ

Process ONLINE Monitors (2):
=====
RID      R_RID    PID      GROUP
19        1        29058    Syb_Res2
20        1        28592    Syb_Res1
```

- Run the following command to set the ResLogLevel attribute to TRACE. When you set ResLogLevel to TRACE, the agent logs messages in the SybaseIQ\_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
```

## Known issues

This release of the agent for Sybase IQ has the following known issues:

### Problem

An error message might appear when you run the `hares -offline` command to take a resource offline.

### Description

When a resource is taken offline, it is unregistered from the AMF module. However, the `imf_register` function attempts to unregister the resource again.

### Workaround

It is safe to ignore this error message.

# Sample Configurations

This appendix includes the following topics:

- [About sample configurations for the agents for Sybase IQ](#)
- [Sample agent type definition for Sybase IQ](#)
- [Sample configuration files](#)
- [Sample service group configurations for SybaseIQ](#)

## About sample configurations for the agents for Sybase IQ

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 Sybase IQ. For more information about these resource types, refer to the *Cluster Server Bundled Agents Reference Guide*.

## Sample agent type definition for Sybase IQ

```
type SybaseIQ (
static str AgentDirectory = "/opt/VRTSagents/ha/bin/SybaseIQ"
static str AgentFile = "/opt/VRTSvcs/bin/Script51Agent"
static str ArgList[] = { ResLogLevel, State, IState, SybIQUser,
EnvFile, SybIQBinDir, ServerName, StartOpts, SybIQDB,
SybIQDBConfFile, SybIQDBAdmin, SybIQDBAdminPasswd,
SecondLevelMonitor, MonitorProgram, UseSystemD }
static boolean AEPTIMEOUT = 1
static int IMF{} = { Mode=2, MonitorFreq=5,
RegisterRetryLimit=3 }
```

```
static str IMFRegList[] = { ServerName, EnvFile, SybIQBinDir,  
SybIQUser }  
str ResLogLevel = INFO  
str SybIQUser  
str EnvFile  
str SybIQBinDir  
str ServerName  
str StartOpts  
str SybIQDB  
str SybIQDBConfFile  
str SybIQDBAdmin  
str SybIQDBAdminPasswd  
int SecondLevelMonitor  
str MonitorProgram  
boolean UseSystemD = 0  
)
```

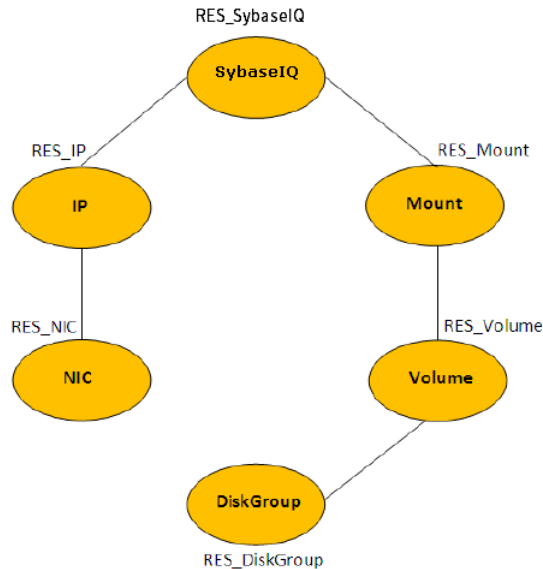
## Sample configuration files

A sample main.cf file is as follows:

```
group svg_SybaseIQ (  
SystemList = { galaxy = 0, nebula = 1 }  
)  
  
SybaseIQ SybaseIQ_Res1 (  
Critical = 0  
ResLogLevel = TRACE  
SybIQUser = root  
EnvFile = "/opt/sybase/IQ-15_3/IQ-15_3.sh"  
SybIQBinDir = "/opt/sybase/IQ-15_3/bin64"  
ServerName = ABC  
StartOpts = "-r"  
SybIQDB = "/opt/sybase/IQ-15_3/demo/iqdemo.db"  
SybIQDBConfFile = "/opt/sybase/IQ-15_3/demo/  
iqdemo.cfg"  
SybIQDBAdmin = DBA  
SybIQDBAdminPasswd = GUJuHSS  
UseSystemD = 1  
)
```

# Sample service group configurations for SybaseIQ

The following figure shows a service group with a SybaseIQ instance running in a VCS environment.



```
group sybaseiq (  
    SystemList = { galaxy = 0, nebula = 1 }  
)  
  
DiskGroup iq_dg (  
    DiskGroup = sybaseiq  
)  
  
IP ip_res (  
    Device @galaxy = eth3  
    Device @nebula = eth4  
    Address = "10.209.69.62"  
    NetMask = "255.255.252.0"  
)  
  
Mount iq_mount (  
    MountPoint = "/sybaseiq"  
    BlockDevice = "/dev/vx/dsk/sybaseiq/sybaseiq_vol"  
    FSType = vxfs  
)
```

```
FsckOpt = "-y"
)

NIC nic_res (
    Device @galaxy = eth3
    Device @nebula = eth4
)

SybaseIQ sybvaseiq_res (
    ResLogLevel = TRACE
    SybIQUser = sybaseiq
    EnvFile = "/sybaseiq/installation/IQ-16_0/IQ-16_0.sh"
    SybIQBinDir = "/sybaseiq/installation/IQ-16_0/bin64"
    ServerName = iq_server
    SybIQDBAdmin = DBA
    SybIQDBAdminPasswd = hvkVitT
)

Volume iq_vol (
    DiskGroup = sybaseiq
    Volume = sybaseiq_vol
)

ip_res requires nic_res
iq_mount requires iq_vol
iq_vol requires iq_dg
sybvaseiq_res requires ip_res
sybvaseiq_res requires iq_mount

// resource dependency tree
//
// group sybaseiq
// {
//   SybaseIQ sybvaseiq_res
//   {
//     IP ip_res
//     {
//       NIC nic_res
//     }
//     Mount iq_mount
//     {
//       Volume iq_vol
```

```
//      {  
//      DiskGroup iq_dg  
//      }  
//      }  
//      }  
// }
```