Paper 469-2013

Best Practices for Deploying Your SAS® Applications in a High-Availability Cluster

Helen Pan, SAS Institute Inc., Beijing, China

ABSTRACT

Are you frustrated when software or hardware failures interrupt your SAS® system? A high-availability cluster with high-availability (HA) cluster with HA software can help you by providing failover protection for your SAS applications, thus reduce your system downtime.

This paper discusses what you should consider when deploying your SAS applications or servers in an HA cluster, as well as the following best practices of the deployment process:

- HA cluster architecture with a SAS deployment.
- SAS installation and deployment in an HA cluster.
- Dependency configuration of the SAS servers.
- Consideration about the SAS clients when a failover happens.

INTRODUCTION

"High-availability clusters (also known as HA clusters or failover clusters) are groups of computers that support server applications that can be reliably utilized with a minimum of down-time. They operate by harnessing redundant computers in groups or clusters that provide continued service when system components fail."[1]

SAS applications can be deployed in a high-availability cluster for reducing downtime and improving availability. When a SAS application is out of service or a computer on which SAS applications are running goes down, other computers in the cluster can take over and bring the SAS applications back online.

This paper introduces best practices for deploying your SAS applications or servers in a high-availability cluster and highlights important considerations to keep in mind during the deploying process. The process is as follows.

1. Planning hardware and software components for a high-availability cluster.
2. Preparing hardware components for a high-availability cluster, such as network, storage, and quorum and / or fencing device.
3. Creating a high-availability cluster and configuring the cluster with quorum and / or fencing device.
4. Installing and configuring SAS software on the cluster nodes with SAS Deployment Wizard.
5. Deploying SAS applications into the high-availability cluster with resource dependencies considerations and resource and resource group creation.
6. Validating that SAS applications failover successfully across the cluster nodes.

In addition, this paper also discusses what will happen to SAS client sessions and running SAS jobs when the SAS applications in a high-availability cluster failover from one computer to another.

STEP 1: PLANNING YOUR HIGH-AVAILABILITY CLUSTER

A high-availability cluster is an integrated system of hardware and specialized software. Before a high-availability cluster is created, you need to carefully plan your high-availability cluster with the hardware and software components. Figure 1 shows a typical high-availability cluster architecture with a SAS deployment. Each component of the architecture and its considerations are described below. After you make a specific plan for your cluster, it is strongly recommended to prepare a checklist to document the key components and related notes, which will be referred to during the deployment process.

Cluster nodes

High-availability clusters have a minimum of two nodes or computers, a primary and a secondary. Each node is responsible for and must be capable of running the same set of SAS applications.

[1] "High-availability cluster," Wikipedia, available at http://en.wikipedia.org/wiki/High-availability cluster (accessed January 5, 2013)..

Logical host name or IP address

This provides the access point to the applications or service in a cluster for clients.

Shared storage

This is storage accessible to all of the cluster nodes. It is generally used to store shared data and/ or as a quorum device. In many configurations, a storage area network (SAN) is the most popular solution for the shared storage.

Public network

This allows nodes to communicate outside the cluster to the clients. Each host in the cluster configuration must have at least one public-network connection to the same set of public sub-nets for the clients to access the cluster.

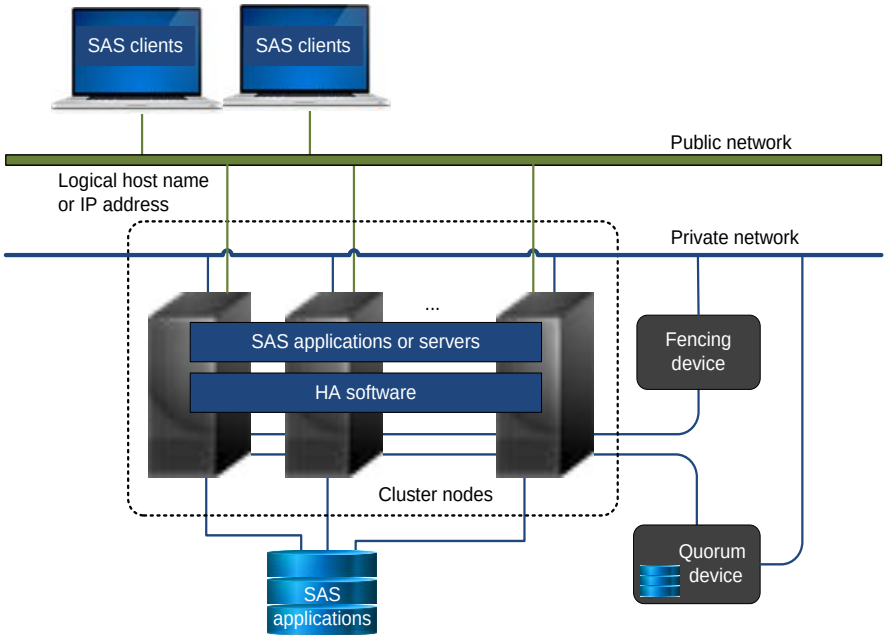

Figure 1. High-availability cluster architecture with a SAS deployment

Private network

This provides communication among the cluster nodes (heartbeat) and to other cluster hardware such as fencing devices and shared storage.

Quorum device

This component helps determine which node will be used to run the applications after a failure happens or the cluster-interconnect between nodes is lost. After failover, one of the working nodes must be responsible for running the protected applications. This is often implemented using a voting mechanism in which each node votes for itself if it believes it can run the applications. But what happens when both nodes in a two-node cluster vote for themselves? You have a tie and both might attempt to start the applications, which will not work. Including a quorum device eliminates the threat of a tie. As part of a majority vote mechanism, this guarantees only one node will have the votes needed and eliminates the risk of the application being started on multiple nodes.

The quorum device can be a partition on the shared storage, shared file system, or a special server depending on your high-availability software requirement. Most of time, a partition of shared storage is used as quorum device.

Fencing device

These are devices or services that isolate a node or protect shared resources such as shared storage when a node appears to be malfunctioning. The isolation or protection process is called fencing. Using fencing is more reliable than simply relying on an errant node to stop using resources on its own. A fencing device is required for some high-availability software on a Linux operating system such as Red Hat Cluster Suite.

High-availability software

High-availability software must be installed on each cluster node to create a high-availability cluster.

SAS software works with various high-availability software, including: Platform Enterprise Grid Orchestrator, which is a component of SAS® Grid Manager, Microsoft Windows Server Clustering, Red Hat Cluster Suite, Veritas Cluster Server, Oracle Solaris Cluster, and so on.

SAS software (SAS applications and clients)

The SAS® Metadata Server is an important part of SAS environment. Failure of the SAS Metadata Server leads to failure of the whole SAS System. The high-availability cluster is designed to provide failover protection for critical applications. Deploying the SAS Metadata Server into a high-availability cluster is strongly recommended. The high-availability cluster can also provide the failover protection for other SAS servers, such as the object spawner and the SAS® OLAP Server.

When the SAS plan file is created, the recommendation is to place the SAS applications for which you want to provide failover protection on a separate machine. Certainly the plan should conform to the requirement of SAS deployment. See **SAS 9.3 Intelligence Platform: Installation and Configuration Guide** for the details of making a plan.

In a typical SAS® Visual Analytics deployment, all components of the SAS Visual Analytics bundle (other than the software for the cluster node) are installed and configured onto a single machine. Thus, you need to deploy all servers including SAS servers and web application server into the high-availability cluster.

When the operating system for each cluster node is Windows, the SAS applications or components must be installed on each cluster node. When that is an operating system that is like UNIX, the SAS applications or components can be installed on the local disk of each cluster node or onto shared storage. That is, duplicate SAS software executables can exist on more than one of the cluster nodes. However, it is recommended that you install the SAS applications on shared storage for the convenience of maintenance on an operating system that is like UNIX. On the other hand, the SAS applications must be configured in a single shared storage location to ensure data integrity. The SAS applications will run on only one cluster node at any given point in time.

STEP 2: PREPARING HARDWARE COMPONENTS

The hardware components need to be configured according to the requirement of the high-availability software. The following description introduces the key hardware components and provides their configuration. The selection of hardware device selection is not covered in this paper.

- Configure both a public network and a private heartbeat network as described in the documentation for the high-availability software.
- Partition local disk according to the requirement of the high-availability software. Make sure the local disk of each cluster node has enough space to accommodate the operating system, the high-availability software, and the SAS applications if you plan to install your SAS applications on local disk.
- Create file systems on the partitions of shared storage that will be used to stored data and make sure the file systems are accessible to all cluster nodes. Make sure shared storage has enough space to accommodate SAS configurations, SASWORK, and other SAS files or data sets. If it is a cluster of operating systems that are like UNIX and you plan to install your SAS software on shared storage, additional space will be needed for the SAS installations in addition to what was mentioned above.
- Reserve a partition on shared storage if you plan to use a shared partition as quorum device. Otherwise, configure some other quorum device according to your high-availability software. For example, Microsoft® Windows Failover Clustering can use a shared file system and Oracle Solaris Cluster can use a dedicated quorum server as their quorum device.
- Prepare the hardware fencing devices if they are part of your plan.

STEP 3: CREATING AND CONFIGURING A HIGH-AVAILABILITY CLUSTER

This step is to install high-availability software, create a cluster, configure the quorum device for the cluster and / or the fencing device for each cluster node and validate the cluster. Configure these things according to the documentation of the high-availability software you have selected.

During the cluster creation process, you will be asked if you would like to enable the failback capability. Enabling failback indicates the SAS applications should switch back to the preferred node (which you specify) automatically when the preferred node comes back online after a failure. The failover provisions of a SAS software license generally require you to enable this failback capability.

Validating the configuration of your high-availability cluster is highly recommended before installing SAS software. The validation should include the following items:

- Switchover of shared storage: confirm shared storage remains accessible to the expected cluster node after a failover.
- Switchover of the logical host name or IP address: confirm that a connection to the logical host name is redirected to the expected cluster node after a failover.
- Failback functionality: confirm that the logical host name or IP address should automatically switch to the preferred node when the preferred node is back online and the shared storage is accessible again as well.
- The functionality of quorum and fencing devices if any exists: confirm that the shared storage and logical host name remain online even after more than a half of the cluster nodes fail.

STEP 4: SAS SOFTWARE INSTALLATION AND CONFIGURATION

Whether the SAS software and applications are installed on local disk or shared storage has been discussed earlier. This section discusses only the precautions for the installation and configuration of the SAS applications that will be installed on the cluster nodes.

Regardless whether you install the SAS software on either local disk or shared storage, the following should be done or confirmed:

- On an operating system that is like UNIX, remember that you should not install or configure SAS with the `root` user, and the suggestion is to use a special SAS installer account instead.
- During the SAS configuration process: when SAS® Deployment Wizard prompts the host name of the SAS applications you are configuring, you must supply the logical host name instead of host name of any cluster node. To make the logical host name reachable during the configuration process, you can map the IP address of the current node to the logical host name in the operating system hosts file `c:\windows\system32\drivers\etc\hosts` (for a Windows operating system) or `/etc/hosts` (for an operating system that is like UNIX) of this node. Remember to change this back after the configuration is complete.
- On a Windows operating system, after the SAS configuration, the Windows services for the SAS servers will have been created on the current node only. You need to bring shared storage online for each of the other cluster nodes, being careful to mount it with the same drive name used on the first node, and install the necessary local Windows services on the other nodes by using the SAS server batch files and the `-install` parameter (for example, `<SAS_CONFIG>/Lev1/SASMeta/MetadataServer/MetadataServer.bat -install`).

In addition to these general considerations, there are some special considerations for depending on which scenario you choose use for installing SAS software.

Scenario 1: Install and configure SAS software on shared storage

The SAS software can be installed and configured on shared storage by launching the SAS Deployment Wizard from any cluster node.

- Be sure to mount shared storage to the appropriate node before launching the SAS Deployment Wizard.
- Both the SAS installation and configuration directory paths should point to locations on shared storage during the SAS configuration process.

Scenario 2: Install SAS software onto local disk and configure it on shared disk

SAS software must be installed on local disk of each node and configured on shared storage from any cluster node once.

- You need to mount shared storage on each node before starting the configuration process.
- The SAS Deployment Wizard must be launched on each node and used to install SAS software on local disk of each node. Make sure the installation paths are the same on all cluster nodes.
- On operating systems that are like UNIX, after the SAS installation is completed, execute the following command as the `root` user on each node to make sure the proper permissions are set on SAS binaries: `/sas/SASHome/SASFoundation/9.3/utilities/bin/setuid.sh`.
- The configuration directory paths should point to the locations on shared storage when the SAS Deployment Wizard is configuring SAS software.

STEP 5: CONFIGURING THE HIGH-AVAILABILITY CLUSTER FOR SAS APPLICATIONS

To provide failover protection for the SAS applications, you need to configure the high-availability cluster for the SAS applications that have been installed and configured on the cluster nodes in above step. High-availability software manages applications in a service or resource group that include one or more resources. In this step you primarily need to consider the following two points:

- Dependencies between resources
- Definition of resources and resource groups

DEPENDENCY CONSIDERATIONS

When the starting or running of an application is dependent on the running of another application, an application must be started before the dependent application is started. The dependency between these two applications must be defined to the high-availability software. For SAS applications, it is necessary to consider the dependencies within the SAS applications and between external resources and SAS applications.

External resources include shared storage and logical host name or IP address. Some high-availability does not support configuring SAS with two direct dependent resources. If that is the case, you can make one of the external resources dependent on the other.

The internal dependencies between the SAS servers are given in Table 1. There are three approached to defining these internal dependencies:

- Each SAS server is defined as an operating system service, and the internal dependencies are defined the operating system level. This approach requires that the operating system support defining such dependencies.
- Each SAS server is defined as a high-availability cluster resource, and the internal dependencies are defined at the level of high-availability software.
- The internal dependencies between SAS servers are handled in an operating system-level script and the script is defined as a resource in the cluster.

You can also combine these three approaches to manage the internal dependencies.

RESOURCES AND RESOURCE GROUP DEFINITION

You need to create resources for your SAS applications or servers, create resource groups to include all applications resources and their dependencies and finally bring the resource group online.

Windows operating system

On Windows operating systems, a local Windows services for each SAS server is created when the SAS software is configured. In Microsoft® Windows Server Failover Clustering (WSFC) software, you can add a resource of the "Windows services" type for each SAS server. The dependencies can then be managed by defining the parent-child relationship between resources in a service.

For other high-availability software, see the corresponding document for more information.

Operating system that is like UNIX

On operating systems that are like UNIX, the SAS servers are usually added into the HA cluster as script resources. High-availability software requires a script to start, stop, and check status of a server. SAS provides a set of SAS control script files and they can handle starting, stopping, and checking the status of the SAS servers. Table 2 lists SAS officially supported script files for the typical SAS servers and web application server, and the dependency between all servers.

SAS Server / Spawner	ConsolidatedServer Script	Server Script	Dependency
Metadata Server	<SAS_CONFIG>/Lev1/sas.servers	<SAS_CONFIG>/Lev1/SASMeta/MetadataServer/MetadataServer.sh	None
Object spawner		<SAS_CONFIG>/Lev1/ObjectSpawner/ObjectSpawner.sh	Metadata Server
OLAP Server		<SAS_CONFIG>/Lev1/SASApp/OLAPServer/OLAPServer.sh	Metadata Server
Connect spawner		<SAS_CONFIG>/Lev1/ConnectSpawner/ConnectSpawner.sh	Metadata Server

SAS Server / Spawner	ConsolidatedServer Script	Server Script	Dependency
Share server		<SAS_CONFIG>/Lev1/ShareServer/ShareServer.sh	Metadata Server
Framework data server		<SAS_CONFIG>/Lev1/FrameworkServer/dffedsvrcfg/dffedsvrcfg.sh	Metadata Server
Remote services		<SAS_CONFIG>/Lev1/Web/Applications/RemoteServices.sh	Metadata Server
Information Retrieval Studio*	None	<SAS_CONFIG>/Lev1/Applications/SASInformationRetrievalStudioforSAS1.41/IRStudio.sh	None
LASR Analytic Server Monitor*	None	<SAS_CONFIG>/Lev1/Applications/SASVisualAnalytics5.2/HighPerformanceConfiguration/LASRMonitor.sh	None
Web application server	None	<WEBAPP_SERVER_DIR>/bin/SASServer1.sh	Framework data server, Remote services

Table 3. SAS Officially Provided Script Files for SAS Servers and Dependency

* The SAS® Information Retrieval Studio and SAS LASR® Analytic Server Monitor are special components of SAS Visual Analytics.

As shown in Table 1, SAS supplies both individual control scripts that handle a single SAS server and a consolidated script that handles many of the lower level scripts. This consolidated script handles the dependency of these lower level servers and their dependencies.

You have several options when defining resources for your SAS servers:

- Use the consolidated script `<SAS_CONFIG>/Lev1/sas.servers` and add logic of monitoring other servers into this script. It is important to remember to incorporate the dependencies of the new added server scripts. This approach requires additional script development but reduces the efforts of creating and maintaining multiple resources and script files. You must also remember to double-check the script, and potentially make the same modifications again, after you update your SAS software.
- Define a resource (within your high-availability software) for each single script and manage the dependencies by defining the parent-child relationship between resources in a service group. In this way, you have to create a resource for each SAS server and specify the child-parent relationships of all SAS servers when creating the SAS server service group.
- Use the consolidated script `<SAS_CONFIG>/Lev1/sas.servers` to control the SAS servers it already controls and create individual script for each of other SAS servers.

Some high-availability software, such as Red Hat Cluster Suite and Veritas Cluster Server, requires that a script return a specific value or range when it is executed successfully or fails. You might need to copy each of original script and modify the return codes to return the expected values. Or, you can write wrapper script files that handle setting the correct return code.

Testing your scripts, outside of the high-availability software, before using them as part of a high-availability resource is strongly recommended. Make sure that each script file can start, stop and check the health status of the corresponding SAS server correctly, and that it returns the correct return code.

STEP 6: VALIDATING FAILOVER OF YOUR SAS SERVERS

Generally, the validation of SAS applications after configuring them for high-availability should be executed under the following three scenarios.

- Manual switchover

  The high-availability software usually provides a command or mechanism to manually switch a service group, or application, from a host to another. Use this command to validate to confirm that the SAS applications can be switched correctly. Refer to the documentation of the high-availability software for the details about the mechanism.

- Software failure

  Manually stop a protected SAS application or server (perhaps by using the kill command on operating systems that are like UNIX) and verify that the SAS applications are switched automatically to one of other nodes.

- Hardware failure

  Power off the node where the SAS applications or servers are running to verify that the SAS applications are switched automatically to one of other nodes.

  Power off and power on the preferred node to verify that the SAS applications are switched automatically to one of other nodes and then will failback to the preferred node, if you have enabled the failback function in a previous step.

The following techniques can be used to validate the SAS applications or servers are running on a node after failover or failback:

- SAS Metadata Server validation

  Launch the SAS® Management Console client and connect the metadata server with the logical host name. The connection should be successful.

- SAS application server validation

  1. In SAS Management Console, on the **Plug-ins** tab, find the SAS server under the **Server Manager** node or its sub-nodes.
  2. Right-click this server and select **Validate**. You should see a Validation Successful message.

- Web application server validation

  Attempt to log in to one or more of the SAS web applications; if failover is working properly, you should be able to log on successfully. Keep in mind that it might take time for the web application server to restart and make the SAS web applications available.

CONSIDERATIONS WHEN FAILOVER HAPPENS

The failover of SAS servers will affect SAS client sessions and running SAS jobs. The effect varies depending on the SAS servers or web application server that failover and the SAS clients involved.

SAS METADATA SERVER

If the SAS Metadata Server fails, for the SAS desktop client applications (such as SAS Management Console, SAS® Enterprise Guide®, SAS® Data Integration Studio, and SAS® Information Map Studio), a message box is displayed when such a failure happens. After the user confirms this message,

- any open SAS Management Console sessions lose their connection to the metadata server and all unsaved work is lost. These users need to reestablish a connection to the server manually after the failover process has finished.
- the other client applications try to automatically reconnect. After the failover process has finished, these client applications are able to reconnect and re-authenticate themselves to the new SAS Metadata Server instance. If the failure happens while a user is in the middle of a client wizard in a SAS client application, the previous work in the wizard is not lost and the user is able to continue through the wizard.

If the SAS Metadata Server fails during the execution of a task, in SAS Enterprise Guide or SAS Data Integration Studio, for example, the task fails and the user needs to re-execute it. If the task relies on data stored in temporary libraries, such as SASWORK, all tasks that generate the temporary data need to be re-executed, because this data has been lost.

In many cases, the SAS web applications automatically reconnect when the SAS Metadata Server resumes from a failure. However, as a best practice, it is recommended to restart SAS Remote Services and the web application server in the middle tier after the SAS Metadata Server fails over successfully, if these two servers are not already configured in a high-availability cluster. If the user session is inactive during the downtime, the re-authentication process is transparent to the user. However, if there is a user request pending when the SAS Metadata Server fails, the work in process could be lost.

SAS APPLICATION SERVERS

The failure of a SAS object spawner results in any subsequent requests to the SAS application servers that were managed by the failed object spawner not being processed until the object spawner is restarted or fails over. The failure of a SAS OLAP server process results in the failure of any cube query tasks.

When a failure happens during the SAS application server processing a client request, an exception message is returned to the SAS client application. The user can resubmit his or her work that is executed by a new SAS process running on a healthy server.

However, when a running SAS process is terminated, the temporary data associated with the process (including the contents of the SAS WORK directory) is lost and unavailable even if a SAS client application later reestablishes the connection to the server.

WEB APPLICATION SERVER

If the web application server fails over, users will be re-directed to the SAS logon page after the failover completes. There might be some error messages displayed to the user when the user session is active when the failure happens. Users can log on again and work on a new server, but any previous unsaved work is lost.

CONCLUSION

A number of leading high-availability software products can be used to deploy SAS software into high-availability clusters. The specific high-availability architecture for a SAS deployment is slightly different depending on the clustering software involved. However, the SAS installation and configuration practices are generally the same. This paper provides information about the important architectural considerations you should keep in mind when deploying SAS with high-availability software and documents best practices that have been developed over time. When configuring a high-availability cluster for SAS servers, the dependencies within SAS applications and between SAS applications and external resources are an especially important consideration. Deploying SAS with high-availability technology allows you to reduce downtime and increase availability of your SAS applications.

REFERENCES

Three papers prepared by Enterprise Excellence Center, SAS Institute Inc., and available from your SAS representative:

“Availability Framework within the SAS® Architecture.”

“Deploying SAS® Visual Analytics Servers in a High Availability Cluster with Red Hat® Enterprise Linux Cluster Suite.”

“Deploying SAS® Metadata Server with Windows 2008 Server Failover Clustering.”

“SAS® 9.3 Intelligence Platform: Installation and Configuration Guide.” SAS Institute Inc. Available at http://support.sas.com/documentation/cdl/en/biig/62611/PDF/default/biig.pdf.

“High-availability cluster.” Wikipedia. Available at http://en.wikipedia.org/wiki/High-availability_cluster.

ACKNOWLEDGMENTS

I would like to express my gratitude to my manager Kansun Xia, Greg Smith, and my teammates for their full support and great suggestions on this paper.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Helen Pan
SAS Research and Development (Beijing) Co., Ltd
19/F North Wing, Central Office Tower, Junefield Plaza, No.10 Xuanwumenwai Dajie Xuanwu District
100052
Work Phone: +86 10 8319-3355 ext. 3689
Fax: +86 10 6310-9130
E-mail: honglian.pan@sas.com