

Paper 002-31

## Achieving High Availability for the SAS®9 Metadata Server

Steve Holzworth, SAS Institute Inc., Cary, NC

Clarke Thacher, SAS Institute Inc., Cary, NC

### ABSTRACT

The SAS® Metadata Server is a critical component of the SAS® Intelligence Platform. To ensure that information is available to the people who need it, system managers can leverage high-availability solutions to monitor and protect the SAS Metadata Server.

SAS has been working with its technology partners to implement high-availability solutions for the SAS Business Intelligence Architecture. This presentation reports on the technologies that SAS has used in this process, and discusses the steps that are necessary to implement high availability for your SAS Metadata Server.

### INTRODUCTION

When information is power, access to information is critical. In today's competitive marketplace, users need confidence that the information and analysis they use to drive their businesses will be available when they need it. IT organizations face the challenge of ensuring that key information systems are robust and resilient to failure.

The SAS Intelligence Platform and SAS® Business Solutions provide intelligence that drives many business activities. SAS understands the importance of ensuring that the software is available when users want to ask questions and review information. We also understand that problems occur. That's why SAS works with its SAS Alliance Technology Partners (SAS Alliance Technology Program 2006) to describe how to use high-availability tools in the SAS environment.

The SAS Intelligence Platform provides a suite of servers and services that deliver The Power to Know® throughout the enterprise. SAS Solutions and custom applications are typically deployed in multi-tier environments where Web browsers, application servers, and SAS Foundation Servers operate to create a dynamic and powerful environment to meet today's requirements.

The architecture of the SAS Intelligence Platform spans four logical tiers: data sources, SAS servers, middle-tier Web and Java components, and clients (Figure 1). At the heart of the SAS Intelligence Platform is the SAS Metadata Server. Protecting the SAS environment focuses first on keeping the metadata server available and accessible across the architecture.

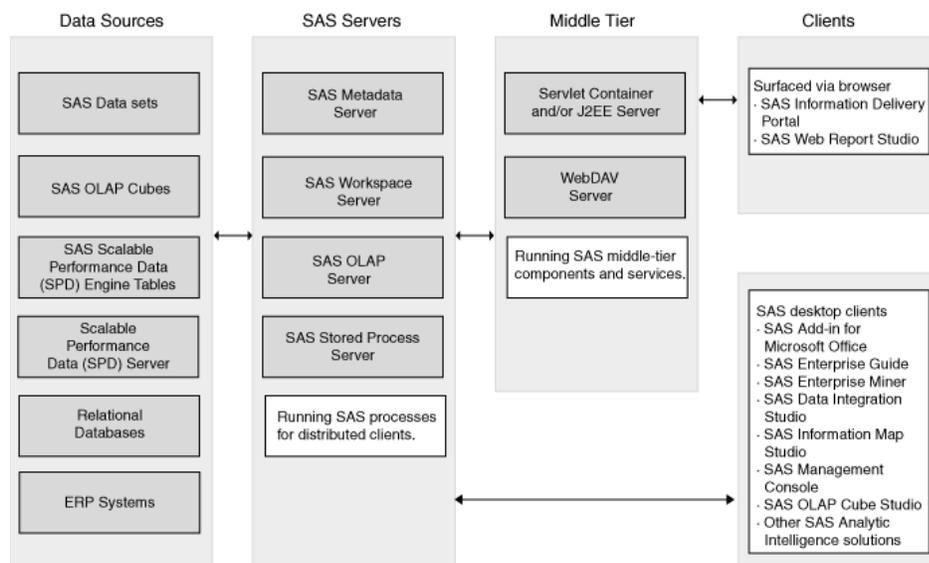


Figure 1. Architecture of the SAS Intelligence Platform (SAS Institute (a) 2005) (Note: SAS Data Integration Studio was formerly known as SAS ETL Studio.)

## SAS METADATA SERVER

The SAS Metadata Server is a multi-user software server that surfaces metadata from one or more repositories to applications via the SAS Open Metadata Architecture. With the ability to import, export, reconcile, and update metadata, and document those actions, the server manages technical, process, and administrative metadata across all applications.

Integrated metadata (information about data sources, how it was derived, business rules and access authorizations) is crucial for producing accurate, consistent information. By storing metadata from all applications in an open, centralized, and integrated repository, data changes need to be documented in only one place, there are fewer systems to support, and business users can count on high-quality, consistent information. A single version of “the truth” is available to all, and better use of staff time lowers the total cost of ownership for IT infrastructures.

## WHAT IS HIGH AVAILABILITY?

*High availability* (HA) generally means ensuring that the software is there when you need it. Related concepts include *clustering*, *fault tolerance*, *continuous availability*, and *disaster recovery*. The common thread among these is the focus on *business continuity*. The distinctions are based on how much down time you can tolerate, and how much money you are willing to spend to limit that down time. Whether implemented in hardware, software, or a combination of both, the general process of protecting the environment is the same:

- Monitor processes and resources.
- Restart/recover from failures when they are detected.
- Synchronize with other components and resources.
- Re-establish client connections.

## COMPONENTS OF HIGH AVAILABILITY TO SUPPORT THE SAS®9 BUSINESS INTELLIGENCE ARCHITECTURE

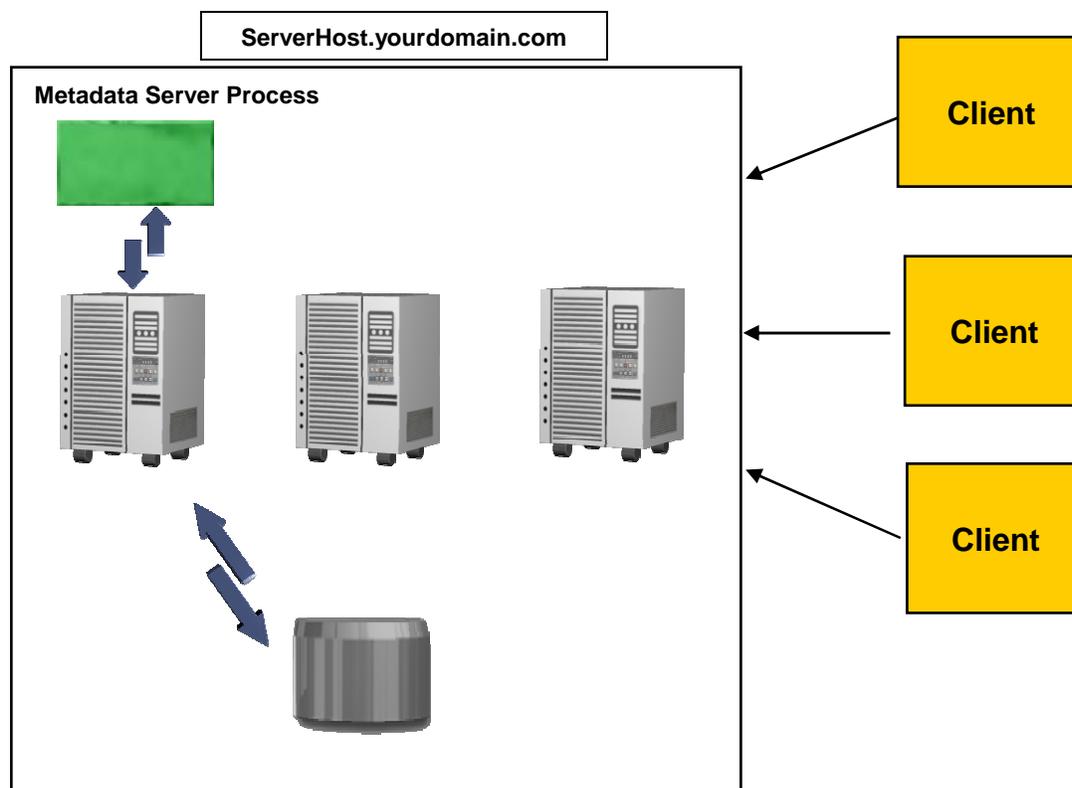
A system which implements high availability must offer the means to monitor running server processes or services and restart them on the same or an alternate machine (at minimum). From the client application perspective, the restarted service must appear to exist at the same network name and/or IP address, and must have access to required data storage, services, and other resources through the same naming paths as the original instance of the service. In some cases, the client application or user might need to re-authenticate themselves to the new service instance following a restart or failover.

In addition to these minimum requirements, HA systems can also support additional monitoring and failover capabilities such as supporting redundant network connections or alternate communications paths, heartbeat monitoring of machines in the HA cluster, redundant disk or network-addressed storage, operator alerts, and controlled shutdown of services in response to unrecoverable failure modes.

## HA DEPLOYMENT OF SAS METADATA SERVER

A typical deployment of SAS in a high-availability configuration (Figure 2) includes a cluster of two or more similar hardware/OS machines, each with access to the data storage used by the SAS Metadata Server and other SAS server processes. A third-party HA system such as Hewlett-Packard's Serviceguard, Sun Microsystems' Sun Cluster, Microsoft's Cluster Server (MSCS), or IBM's HACMP (High Availability Cluster Multiprocessing) monitors the SAS Metadata Server and any of its auxiliary processes. If one of the monitored processes dies, the HA system attempts to restart the SAS Metadata Server.

One machine in the cluster is the primary host for the SAS Metadata Server while the other machines work together to serve as a failover resource pool. The system administrator can configure the HA cluster to either try to restart a failed SAS Metadata Server on the same machine, or to restart the Server on an alternate machine from the failover resource pool.



**Figure 2. Sample SAS Metadata Server HA Cluster**

The SAS Metadata Server is stateless. That is, requests to the server are independent and do not require the retention of information from previous requests. This allows the HA system to redirect metadata requests to a new, restarted server-instance in response to detecting a failure of the SAS Metadata service without impacting or relying on previous requests.

Some HA systems allow configurations of monitored services to be defined as single, independent units, or “packages.” Each package is treated as a discrete entity, and can be monitored and run on any machine in the HA cluster. This gives an administrator the ability to utilize each machine in a cluster as an active resource, with each serving as a potential failover resource for the other HA packages. For example, in a two-machine cluster, SAS Metadata Server can run on one machine while SAS OLAP Server (SAS OLAP Server 2006) is running on the other, with each server package monitored for high availability. If a fault is detected with the SAS Metadata Server, it can failover to the SAS OLAP Server machine, allowing continued service to the clients of each at the expense of additional processor load on the failover machine. This configuration would yield better utilization of the cluster while supplying failover resources.

Typical HA monitoring and failover configurations require that the SAS server applications and server data sources (for example, SAS Metadata repositories) reside on storage media that is accessible by all nodes in the HA cluster. Combinations of direct-attached or network-attached shared storage can be supported by the particular vendor’s HA implementation. Access to, and contention for, these shared resources is controlled by the vendor’s HA system.

Client applications such as SAS ETL Studio expect SAS servers to stay at constant, predefined network locations and port numbers. Those values are defined by the SAS system administrator when the particular SAS application is installed and configured. Rather than associating those values with a particular physical machine, the SAS administrator should define them relative to the logical location or address of the HA cluster or package, as determined by the particular HA vendor’s implementation. Use of a logical address enables the HA system to automatically redirect service requests to alternate physical machines in the cluster if a failure is detected on a particular node. From the perspective of a client application such as SAS ETL Studio, the related SAS server still resides at a constant location and port, irrespective of which physical machine in the cluster it is actually running on.

The HA cluster or package location is normally defined by the network administrator as a symbolic name that is resolved into an IP address by the local network's Domain Name System (DNS) (for example, `myserver.mycompany.com`). This results in another layer of indirection, allowing the network administrator to efficiently reassign IP addresses within the local network without requiring reconfiguration of the SAS servers or client applications.

HA systems directly monitor for hardware failures in the cluster through OS-specific features. For typical UNIX-based systems, applications running on the cluster are monitored for failures by invoking user-supplied scripts at set intervals to check the status of related application processes. One approach used is to check for the continued existence of the application's process ID. Another approach is to directly issue a query to a server application for a data item, insuring that the server does respond to client requests.

### A SAMPLE IMPLEMENTATION WITH HEWLETT-PACKARD SERVICEGUARD

SAS Institute and Hewlett-Packard engineers worked together to validate SAS Metadata Server as compatible with Hewlett-Packard Serviceguard on the HP-UX Itanium platform (SAS Institute (b) 2005). In the HP Serviceguard example cited below, the SAS Metadata Server was typically restored to service in ten seconds or less after detection of a failure.

HP Serviceguard is a system that allows administrators to create high-availability clusters of HP 9000 or HP Integrity servers. A Serviceguard high-availability cluster protects against software failure and hardware component failure such as failure of a system processing unit, local area network, or disk. If such a hardware failure occurs, a redundant component takes over. Serviceguard uses the concept of a *package* to group resources of an application service. The typical package is a *failover package* which is configured to run on multiple nodes in a cluster, but actively runs on only one node at a time. If a package resource such as a service, node, or network fails on the current node, Serviceguard automatically transfers control of the package to another node in the cluster, resulting in minimal interruption of service (Hewlett-Packard (a) 2005).

The test configuration was composed of an HP DS2100 disk device and a cluster of two HP Integrity rx5670 server nodes. One of the rx5670 units served as the primary node used to run the SAS Metadata Server package, with the second unit used as a dedicated failover node. The DS2100 disk device was shared between the two cluster nodes and contained all shared executables, data repositories, and configuration files. One disk mechanism in the DS2100 served as a cluster lock disk for purposes of cluster arbitration.

The Serviceguard system uses packages to start and halt configured applications on nodes in the cluster. "A package is a collection of services, disk volumes and IP addresses that are managed by Serviceguard to ensure that they are available" (Hewlett-Packard (b) 2005). When a package is started on (or transferred to) a node in the cluster, all associated package resources including the package IP address are directed to the new node. As a result, the identity of the node executing the package within the cluster is irrelevant to the clients that access the related service. Even when the package is restarted as the result of a failover to another node, the effective IP address of the service remains the same from the client's perspective.

Each package has a run script and a halt script associated with it. These scripts control the configured application. SAS and HP engineers adapted existing SAS Metadata Server startup scripts to the Serviceguard model, developed a monitoring script to verify execution of related applications, and created a Serviceguard package configuration file to define the restart/failover behavior of the SAS Metadata Server application. A package configuration file specifies which nodes in the cluster that the package can be run on, the failover and possible failback behaviors of the package when a service failure is detected, and whether the associated application should be run automatically at cluster startup time.

After the configuration files and scripts were completed, the SAS Metadata Server package (`saspkg`) was started on the cluster, a test client was created and invoked to rapidly and repeatedly query the Metadata Server, and various failure modes were tested for proper recovery. Serviceguard commands were used to manually start, stop, and force failover of the Metadata Server package. Various types of hardware failures were tested, including disconnecting the primary network communications cable, disconnecting the primary cluster heartbeat cable, physically powering-off the active node, and "killing" the running process via the UNIX signal mechanism. The system performed as expected in all of these scenarios, restarting or failing over to a backup node as specified by the package configuration.

The test client detected the failure of the SAS Metadata service by a loss of the TCP/IP connection. The client then initiated a series of attempts to reconnect to the service at the original service network address and port number.

When the SAS Metadata Server was restarted by Serviceguard, either on the same or an alternate node, the test client successfully continued querying the metadata repository.

## CONCLUSION

The primary requirements for operating the SAS Metadata Server in a high-availability server configuration are similar across HA implementations offered by hardware and system management vendors. While the specific steps vary among different vendors' HA implementations, the HP Serviceguard example demonstrates the required aspects of operating a high-availability server configuration:

- The server application processes are monitored for availability along with any required resources such as network interfaces and disk volumes.
- If any monitored element fails, the HA system switches to a backup resource, such as a secondary network interface, or restarts the service on the same or an alternate node.
- Clients might need to reconnect, but should not need to reconfigure. From the client perspective, the restarted service should retain the same network address, logical filenames and paths, and service port numbers of the original instance of the service.
- The above steps must occur rapidly to minimize service interruption.

In the contemporary business environment, ready access to information is vital to decision makers within the enterprise. You can minimize downtime of the SAS Intelligence Platform in the event of a failure by leveraging system management tools to provide rapid, transparent failover and resumption of the metadata service to client applications and data users without operator intervention.

## REFERENCES

Hewlett-Packard (a). October 2005. *Managing Serviceguard (Twelfth Edition)*. Palo Alto, CA: Hewlett-Packard Development Company, L.P. Available [docs.hp.com/en/B3936-90095/B3936-90095.pdf](https://docs.hp.com/en/B3936-90095/B3936-90095.pdf).

Hewlett-Packard (b). 2005. *Managing Serviceguard Version A.11.16* (Eleventh Edition, Second Printing). Palo Alto, CA: Hewlett-Packard Development Company, L.P.

SAS Institute Inc. *The SAS Alliance Technology Program*. (2006). Available [www.sas.com/partners/programs/platform/index.html](http://www.sas.com/partners/programs/platform/index.html).

SAS Institute (a). 2005. *SAS® Intelligence Platform: Overview*. Cary, NC: SAS Institute Inc.

SAS Institute (b). 2005. "SAS® Metadata Server Compatibility with HP Serviceguard Suite Validated." News/Press Release. Cary, NC: SAS Institute Inc. Available [www.sas.com/news/preleases/082905/news1.html](http://www.sas.com/news/preleases/082905/news1.html).

SAS® *OLAP Server*. (2006). Cary, NC: SAS Institute Inc. Available [www.sas.com/technologies/dw/storage/mddb/](http://www.sas.com/technologies/dw/storage/mddb/).

## CONTACT INFORMATION

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

Steve Holzworth  
SAS Institute Inc.  
SAS Campus Drive  
Cary, NC 27513  
[Steve.Holzworth@sas.com](mailto:Steve.Holzworth@sas.com)

Clarke Thacher  
SAS Institute Inc.  
SAS Campus Drive  
Cary, NC 27513  
[Clarke.Thacher@sas.com](mailto:Clarke.Thacher@sas.com)

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.