Paper SAS415-2017

Twelve Cluster Technologies Available in SAS® 9.4

Rob Collum, SAS Institute Inc.

ABSTRACT

We are always looking for ways to improve the performance, efficiency, and availability of our investment in SAS solutions. To address those needs, SAS offers the ability to cluster many of its constituent software components. This session identifies twelve components of SAS software components that can be clustered and describes how they are designed to boost the capabilities of SAS to function in the enterprise.

INTRODUCTION

SAS software is frequently used to support critical analytics tasks that drive day-to-day business decisions. With successful implementation of those tasks, they take on heightened importance and popularity. Over time, the enterprise becomes increasingly reliant on SAS software to support running more tasks for a growing community of users. This growth amplifies the need for SAS software to deliver those results in an efficient and dependable manner.

As the software use grows and expands, the SAS solution needs to scale to meet the demand. And as the business becomes ever more reliant on the solution, then the SAS software needs a robust deployment to ensure its continued availability even in the face of outages, planned or not. Clustering of software components provides the ability to support these needs, offering improved scalability and/or availability of the solution.

This paper will briefly introduce the primary concepts to describe the different cluster technologies used by SAS software. It will also provide key architectural requirements as well as provide a rating of the difficulty of deployment based on this scale:

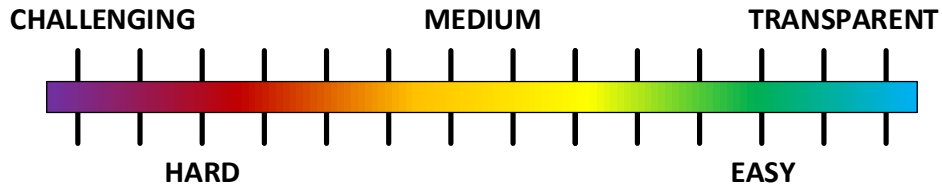


Figure 1. Deployment Difficult Rating Scale

Deployment of the cluster technologies used by SAS range from **TRANSPARENT** – where zero effort is required to enable the cluster – all the way to **CHALLENGING** – where significant effort to deploy and maintain the cluster is necessary.

CLUSTERING 101

A cluster is a set of systems that work together with the goal of providing a single service. Planning the deployment of clustered software systems requires the system architects to understand the specific capabilities and requirements of each type of clustered component.

SCALABILITY

Scalability is an assessment of a system's potential to change in size. From a computing perspective this means that a system is scalable when there is a well-defined understanding of how we change the size of our computing environment to either increase or decrease its capacity to do work. As most computing work tends to grow in size over time, we are most often concerned with increasing our systems' ability to process work.

When designed for scalability beyond a single host machine, the components of a software cluster are deployed across multiple hosts. Incoming tasks are distributed (that is, load balanced) across the software's cluster nodes on those host machines.

AVAILABILITY

Hardware failures in a computing environment are inevitable. Hard drives fail; power supplies overload; motherboards burn out; accidents happen. Software clusters designed for improved availability are resilient in the face of these challenges. The first goal of any available system is to remove potential single points of failure (SPOF). A single point of failure is a problem because if that particular item ceases to function properly, then other inter-dependent components won't be able to function either.

To achieve improved availability, a software cluster is deployed across multiple host machines to remove single points of failure. This concept can be extended beyond just the server hosts, but also to storage solutions, racks, data centers, and even geography.

CLUSTER TECHNOLOGIES AVAILABLE IN SAS

SAS solutions consist of many discrete software elements. Each of these elements is responsible for a set of tasks to enable the fast, efficient delivery of analytics and results to users. The users do not need to know about the cluster technologies used at their site – they just want their SAS jobs to run. SAS architects and administrators, on the other hand, need to be familiar with these cluster technologies to properly plan, deploy, and support SAS solutions.

CLUSTER DEPLOYMENT CONCEPTS

The various components of SAS solutions offer different cluster implementations. Some are optimized for scalability only, or for availability only. Some offer both improved scalability and availability. Furthermore, to fully realize the software cluster's potential, third-party software and/or hardware might be required.

There are two computing architectures referred to in this paper:

- **SMP** – symmetric multi-processing:
  Most software we work with regularly operates using the SMP model. It executes on a single host machine alongside other software simultaneously. It can take advantage of multi-threaded processing on one or more CPU cores. In a clustered environment, a new task is assigned by a load-balancing algorithm to one host machine for processing.
- **MPP** – massively parallel processing:
  Some software is designed to operate across multiple host machines and work together to provide a single logical service. This approach allows for massive scalability since individual host machines can be improved, but more importantly, new hosts can be added to the cluster as needed. Typically, a task provided to this software is broken up into smaller chunks and distributed across the hosts of the cluster. Each host processes its chunk of the problem in parallel with the others to radically reduce the time needed to complete the overall task.

For each cluster discussed here, this paper will:

- Identify the cluster's objective: scalability, availability, or both.
- Explain the cluster's functions in basic terms.
- Provide a simple rating to gauge the difficulty of the cluster's deployment. A rating of "easy" assumes expertise in planned SAS software deployments. A rating of "hard" implies that experience with third-party technologies, system administrative privileges, and/or complex manual configuration is required.
- Describe required deployment considerations.

01 – SAS® METADATA SERVER

The SAS® Metadata Server fulfills numerous roles in a SAS solution environment, including authentication, authorization, service definition, content management, and much more. It is one of the most critical components of the SAS® Intelligence Platform architecture.

Clustering Functions

The SAS Metadata Server can be clustered to run as multiple instances on separate host machines to provide much higher availability since access to metadata is so crucial to the operations of all SAS solutions. The cluster technology used by SAS Metadata Server requires at least 3 configured instances and offers improved availability through active-active operation of all of the clustered nodes.

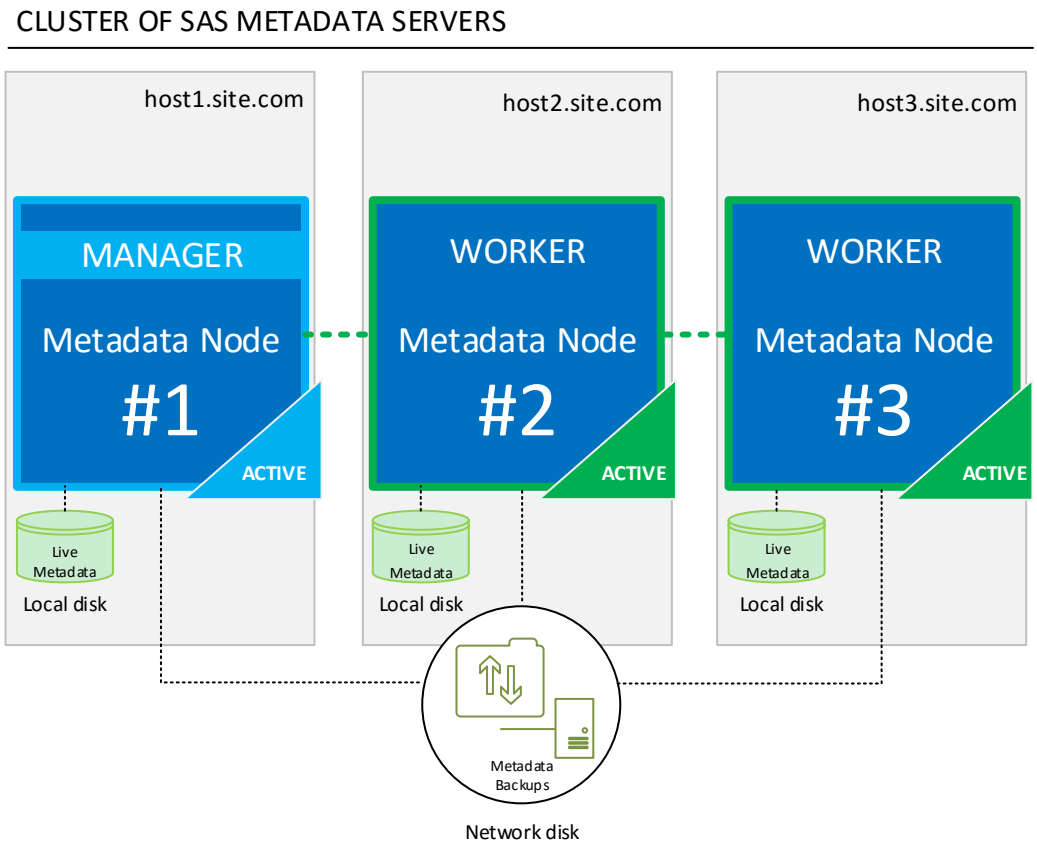


Figure 2. SAS Metadata Server Clustered to Operate across 3 Hosts

Workload and data integrity are managed by one SAS Metadata Server Node, which operates as the Manager of the cluster. The remaining instances act as Worker Nodes. If one Worker fails, then the service will continue to operate uninterrupted with only 2 (or down to half of the configured number of instances, depending on certain conditions). If the Manager Node goes down, then the Workers will elect a new Manager and operations will resume.

There are three kinds of clients that rely on the SAS Metadata Server. And how they discover where the active cluster nodes are running varies a bit:

1. Managed desktop clients:
    - SAS® Enterprise Guide®, SAS® Office Analytics, and so on
    - Connection profile maintained by the client application itself
    - Successful connection to any node of a clustered SAS Metadata Server automatically updates

the connection profile (in case of added or removed nodes)

2. Server processes:
    - SAS Object Spawner, SAS Application Servers (Workspace, OLAP Server, others), and so on
    - A connection profile for each service is maintained in the SAS configuration directory which is automatically created at install time. Use the *sas-update-metadata-profile* utility to make changes to the profile if the metadata cluster configuration changes.
3. Programmatic interfaces:
    - SAS Display Manager, User-authored SAS programs, and so on
    - Users can create their own metadata connection profile to describe individual or clustered metadata servers. Useful for dev/test/prod environments.

When deployed in a clustered configuration, the SAS Metadata Server cannot support normal user operations with less than half of the configured hosts online. The reasons for this are based in the concept of quorum and preventing the potential computing problem known as “split-brain” where separate data sets get out of sync.

In the illustration above, notice that each metadata node maintains its own local copy of metadata files. The Manager Node coordinates the Update operations of the cluster. A separate network file share is also required for the backup and recovery process.

Clustering Deployment Considerations:

Deploying a clustered SAS Metadata Server is **EASY** since it’s a standard option of a planned SAS software deployment.

Summary:

Objective	Improves availability
Operation	Active-Active failover Manager coordinates update actions Manager delegates all user operations to the Workers
Cluster Architecture	SMP Minimum 3 server hosts, 4 is the maximum recommended No geographic separation
Client discovery	Preferred: Use SAS Metadata Server Connection Profile Also: Direct metadata connection to any host (deprecated)
Deployment difficulty	Easy Standard option of a planned SAS deployment

02 – SAS OBJECT SPAWNER

The SAS Object Spawner provides a mechanism to launch and manage SAS Workspace Servers, SAS Stored Process Servers, and SAS Pooled Workspace Servers. If running multiple instances of the object spawner, then they can be configured to work together. This effectively creates a cluster of object spawners which then provides load balancing and higher availability of these services. There are several choices of load-balancing algorithms to choose from to meet your customer’s workload requirements.

Clustering Functions

When an object spawner first starts up, it communicates with the SAS Metadata Server to get its configuration properties, including references to any other object spawners in the environment. The object spawners will elect one to act as the parent-peer which acts as a load balancer for the cluster. The parent-peer node will also accept job assignments to launch SAS server processes on its host machine.

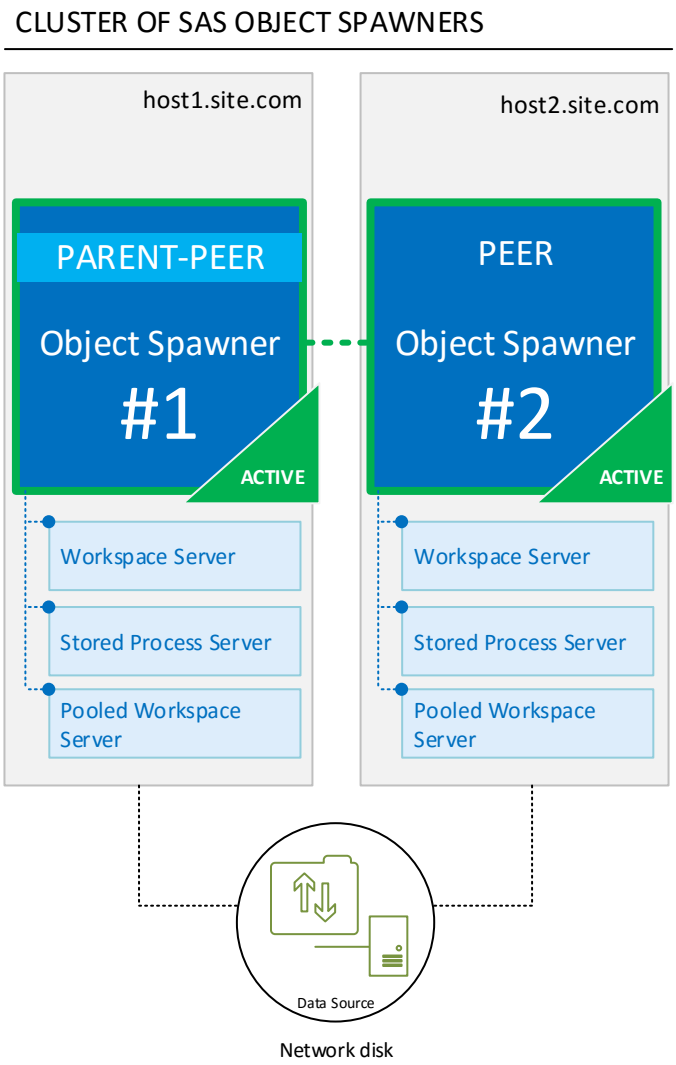


Figure 3. SAS Object Spawners Clustered to Launch SAS Server Processes across Multiple Hosts

In the illustration above, notice the reference to shared network disk. The object spawner cluster does not require this (although it is a convenient way to deploy a shared configuration). However, users of the SAS server processes will not know which host is running the associated SAS Workspace Server (or Stored Process Server or Pooled Workspace Server). The proven practice is to place SAS data locations on a common shared filesystem that is mounted at the same directory location on each host.

SAS clients that communicate with the object spawner include SAS Enterprise Guide, SAS® Studio, SAS® Enterprise Miner™, and many more. These clients discover the connection information to communicate with the object spawner cluster from the SAS Metadata Server.

Clustering Deployment Considerations:

Deploying SAS Object Spawners in a clustered configuration is **EASY** since the components are installed as part of a planned SAS software deployment. Some manual modification of metadata as well as the associated configuration files on disk is required.

Summary:

Objective	Improve scalability and availability of SAS server processes
Operation	Active-Active failover Parent-peer load balances requests, all nodes participate in job execution
Cluster Architecture	SMP Minimum 2 server hosts Excellent integration with SAS® Grid Manager
Client discovery	Object spawner definitions reside in SAS Metadata Server
Deployment difficulty	Easy Standard option of a planned SAS deployment with a few manual configuration items

03 – SAS® OLAP SERVER

The SAS® OLAP Server has much of the same technology as the object spawner baked right into it. So while a single OLAP instance can run alone, you have the option of deploying several and then configuring them to work together. Again, this creates a cluster of OLAP services that are able to load balance incoming requests and provide higher availability.

Clustering Functions

Like the object spawner the OLAP server requests its cluster configuration information at start-up from the SAS Metadata Server. One node acts as the parent-peer and handles load balancing for the cluster. All nodes participate in responding to tasks.

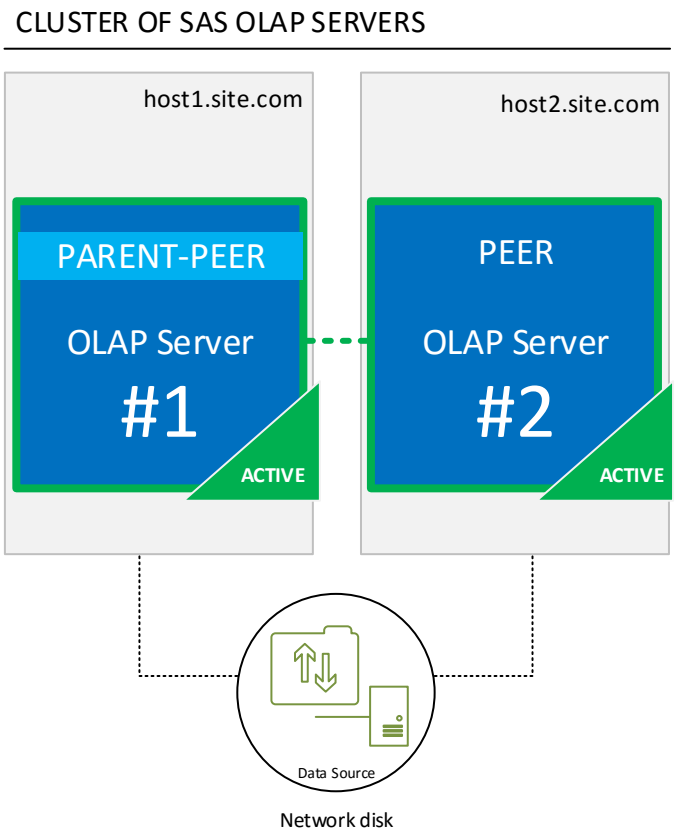


Figure 4. A Cluster of SAS OLAP Servers to Distribute Tasks across Multiple Host Machines

Similar to the SAS server processes associated with the object spawner, the illustration above shows a shared file system at a common mount point on both hosts where the OLAP servers can find data.

Clustering Deployment Considerations:

Deploying SAS OLAP Server in a clustered configuration is **EASY** since the components are installed as part of a planned SAS software deployment. Some manual modification of metadata as well as the associated configuration files on disk is required.

Summary:

Objective	Improve scalability and availability of SAS OLAP Server
Operation	Active-Active failover Parent-peer load balances requests, all nodes participate in job execution
Cluster Architecture	SMP Minimum 2 server hosts Excellent integration with SAS Grid Manager
Client discovery	OLAP Server definitions reside in SAS Metadata Server
Deployment difficulty	Easy Standard option of a planned SAS deployment with a few manual configuration items

04 – SAS® WEB APPLICATION SERVER

The SAS® Web Application Server is a specialized implementation of Pivotal tc Server – which is itself a robustly configured implementation of Apache Tomcat suitable for enterprise use. The web app server is responsible for hosting the SAS web applications that are server-side Java programs that provide a web interface for access to SAS analytics and administration capabilities.

Clustering Functions

The SAS Web Application Server can be configured to provide vertical clustering with multiple Java virtual machine (JVM) instances on a single host machine as well as horizontal clustering with JVM instances deployed across multiple hosts. The SAS Deployment Wizard asks some simple questions about the host machines in the environment and then handles all of the tricky clustering details for you at installation time.

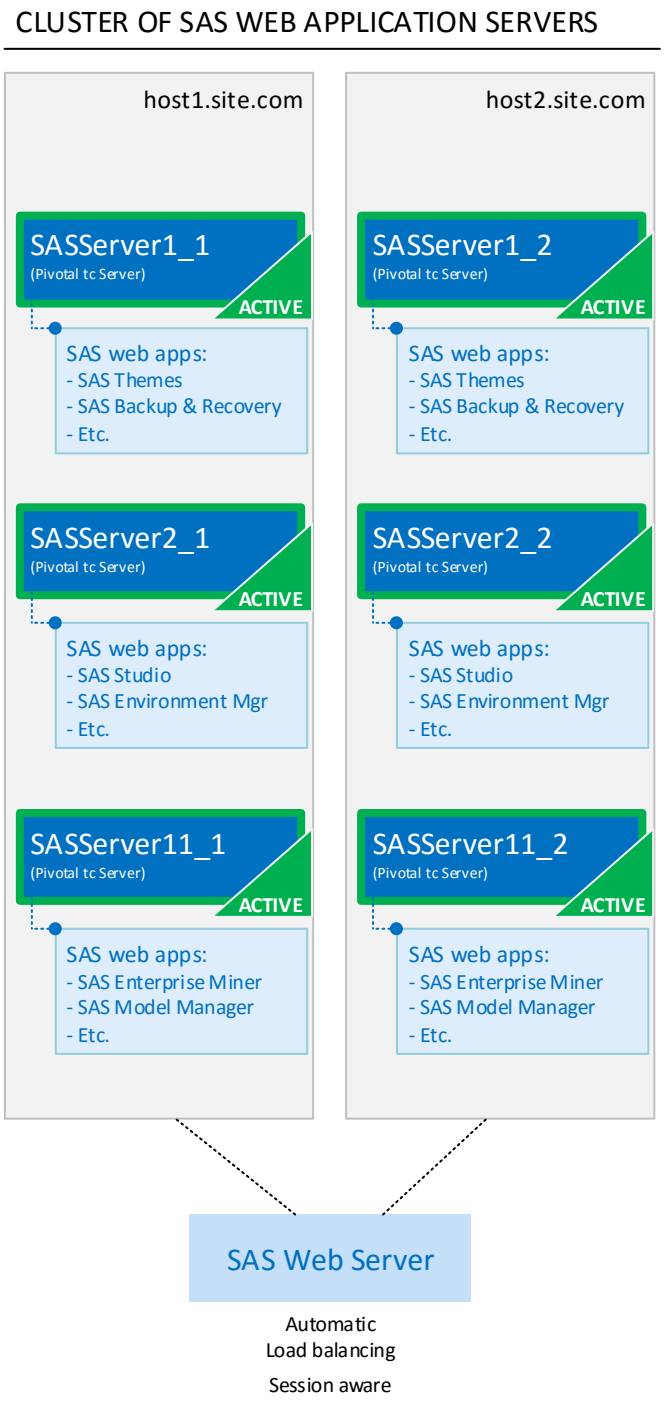


Figure 5. The SAS Web Application Server with JVMs Clustered Both Vertically and Horizontally

Depending on the SAS solutions licensed and deployed at your site, there could be as many as fifteen different JVMs used, named SASServer1 to SASServer15. The various web applications associated with those solutions are automatically placed in specific JVMs based on function.

The SAS Web Server is automatically configured at installation time to act as a reverse proxy for a clustered SAS Web Application Server. Simply accessing a solution's portion of the URL via the SAS Web Server's address will automatically route to the appropriate SAS web application in a specific JVM ensuring proper load balancing.

User requests are associated with sessions that are proxied to always return to the same SAS Web Application Server instance where the session was initiated – there is no way to replicate a user's session across multiple SAS Web Application Server instances.

Clustering Deployment Considerations:

Deploying SAS Web Application Server in a clustered configuration is **EASY** since the components are installed and configured as part of a planned SAS software deployment.

Not all SAS web applications, however, are actually clusterable. Some can only run on a single JVM in the deployment. The SAS Deployment Wizard is aware of these web apps and configures services appropriately.

Summary:

Objective	Improve scalability and availability of SAS web apps
Operation	Active-Active failover All JVMs participate in user activity as needed. Load balancing is automatically handled by the SAS Web Server
Cluster Architecture	SMP Minimum 2 server hosts
Client discovery	SAS Web App Server cluster configuration is stored in the SAS Metadata Server
Deployment difficulty	Easy Standard option of a planned SAS deployment

05 – SAS WEB SERVER

The SAS Web Server is a specialized implementation of Pivotal Web Server – which is itself a robustly configured implementation of Apache HTTP Server suitable for enterprise use. The web server provides a single point of contact for all your SAS solution content available at your site. It acts as a reverse proxy and directs requests for content or web apps to the appropriate destination.

Clustering Functions

The SAS Web Server has no built-in functionality to support clustering. However, the HTTP protocol is designed to operate statelessly – and if we take care to handle user sessions, then we can deploy a second SAS Web Server to provide improved availability.

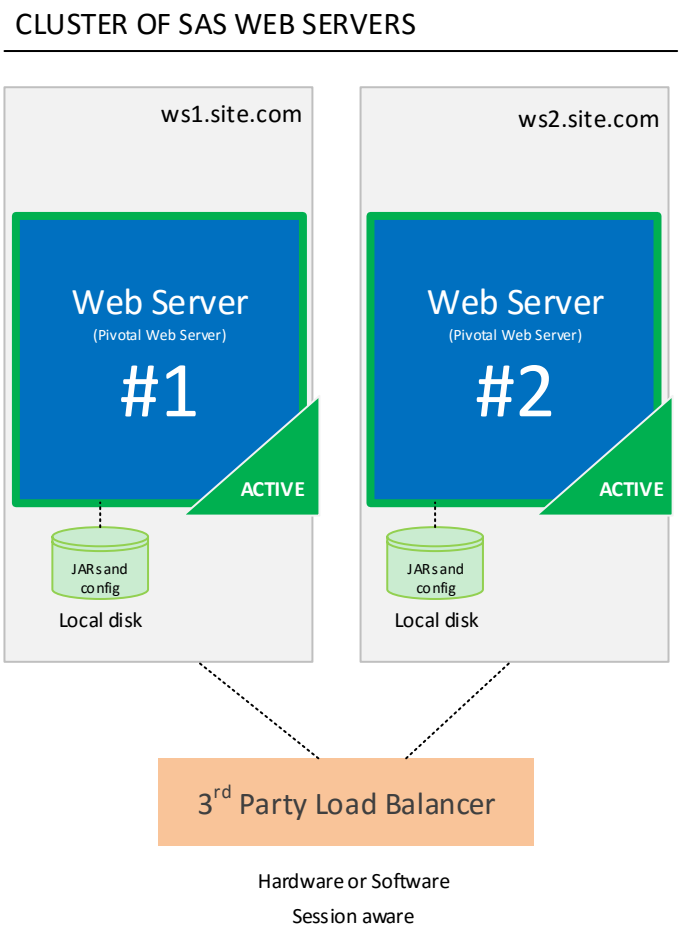


Figure 6. SAS Web Servers Clustered behind a Third-party Load Balancer

In this type of deployment – which is typical for Apache HTTP Servers – the web servers do not communicate nor coordinate with each other. They operate independently. Coordination is handled by a third-party load balancer. You and your IT team can select the load balancing product that works best at your site. The key requirement is to ensure that the load balancer implements the common feature to direct a user's web session requests to the same web server every time – sessions are load balanced, not just individual HTTP requests. Remember that the SAS Web Server will then forward the request on to the SAS Web Application Server where the session information resides to ensure that the user can work within the scope of her session.

If one of the web servers fails, then any associated user sessions will be abandoned – those affected users must log on to create a new session. All user-generated HTTP traffic should be directed to the load balancer, not directly to the clustered web servers. This will ensure that if one web server goes down, then the load balancer will direct all traffic to the remaining SAS Web Servers.

Of course, to properly ensure high availability, the IT organization must configure the load balancer for improved availability as well.

Clustering Deployment Considerations:

Deploying the SAS Web Server in a clustered configuration is rated as **MEDIUM** difficulty.

The SAS Deployment Wizard will only deploy a single SAS Web Server. To create a web server cluster, a second SAS Web Server must be created on a new host as a copy of the software and parameter files from the first with some manual configuration changes. Also, the site's IT team must deploy their choice of

a third-party load balancer that is configured to distribute requests and sessions across the SAS Web Servers.

Summary:

Objective	Improve availability
Operation	Active-Active failover Load balancing is provided by a third-party load balancer
Cluster Architecture	SMP Minimum 2 server hosts
Client discovery	HTTP clients connect to the third-party load balancer
Deployment difficulty	Medium Manual deployment of additional SAS Web Servers Requires third-party hardware and/or software support as well as environmental and system administration privileges

06 – SAS® LASR™ ANALYTIC SERVER

The SAS® LASR™ Analytic Server is an in-memory analytic engine, which can be operated in two modes: distributed (also known as massively parallel processing or MPP) and non-distributed (also known as symmetric multi-processing or SMP). Distributed SAS LASR Analytic Server is of primary interest here and operates as a cluster of multiple instances, each of which runs on a separate host machine. This MPP approach gives SAS LASR Analytic Server the ability to scale up and handle extremely large volumes. SAS LASR Analytic Server, however, even as MPP, does not offer high availability or predictive load balancing. Scalability to tackle “big data” at blazing fast speeds is its primary objective.

Clustering Functions

The Distributed SAS LASR Analytic Server operates using the MPP paradigm. Analytic tasks in SAS LASR Analytic Server are processed by distributing the data for analysis evenly across multiple machines. Each worker instance of the distributed SAS LASR Analytic Server is given a subset of data to work with. After the workers have completed their assigned tasks on their respective sets of data, their results are collected by the SAS LASR Analytic Server Root Node and assembled into the final answer.

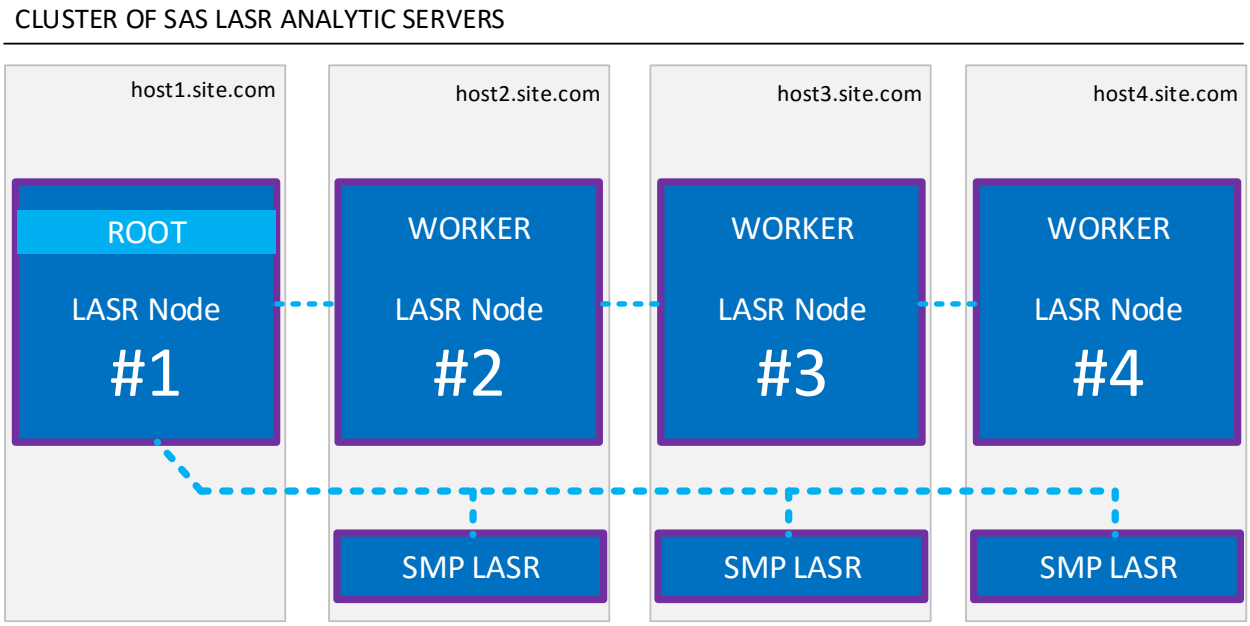


Figure 7. Distributed SAS® LASR™ Analytic Server for Large Data Tasks alongside SMP LASR Instances for Small Data Jobs

A SAS LASR Analytic Server cluster is designed for ultimate scalability – and is not highly available. Even co-locating your SAS LASR Analytic Server cluster alongside a Hadoop cluster which has high availability features enabled will not improve the availability of SAS LASR Analytic Server itself.

If the LASR Root or any LASR Worker goes offline, then the SAS LASR Analytic Server cluster must be restarted and any required tables reloaded. If a LASR host machine must remain offline, then SAS LASR Analytic Server can be started with the remaining nodes (possibly requiring manual configuration). If SAS LASR Analytic Server works with SASHDAT files from a co-located Hadoop provider, then symmetry between the LASR nodes and the HDFS nodes must be maintained.

A recent improvement in load balancing came with the release of SAS 9.4 M3. A distributed deployment of SAS LASR Analytic Server can also work with a number of individual SMP LASR instances in the same clustered environment to host small tables that could not be efficiently stored in a distributed manner.

Clustering Deployment Considerations:

Deploying the Distributed SAS LASR Analytic Server is rated as **MEDIUM** difficulty.

The SAS Deployment Wizard does not deploy LASR. Instead, installation is performed at the Linux command-line and requires root-level privileges to complete. Furthermore, if SASHDAT is desired for staging data for rapidly (re-)loading into LASR, then a supported Hadoop distribution must be deployed alongside LASR symmetrically – where the LASR Root Node is hosted on the same machine as the HDFS NameNode and the LASR Workers are placed alongside each of the HDFS DataNodes.

Summary:

Objective	Massive scalability
Operation	All nodes critical LASR Root manages requests and directs the LASR Workers
Cluster Architecture	MPP (plus some SMP if needed) Minimum 4 server hosts

Client discovery	LASR Root Node is defined in the SAS Metadata Server
Deployment difficulty	Medium Requires system administration privileges in Linux (and possibly Hadoop) Also possibly third-party hardware and/or software support

07 – SAS® HIGH-PERFORMANCE ANALYTICS SERVER

Before there was SAS LASR Analytic Server, we had SAS® High-Performance Analytics Server. And today, it still offers unique and valuable analytics capabilities not found in SAS LASR Analytic Server.

Clustering Functions

The SAS High-Performance Analytics Server also functions as an in-memory analytics engine in much the same way as SAS LASR Analytic Server. However, the main difference is that where SAS LASR Analytic Server persists – that is, remains continuously available with resident data for incoming requests – the SAS High-Performance Analytics Server instead will instantiate across all machines, load data and run the one job requested, and then shutdown. Each subsequent request repeats that process: start up a new server, load data, run analysis, return results, and quit.

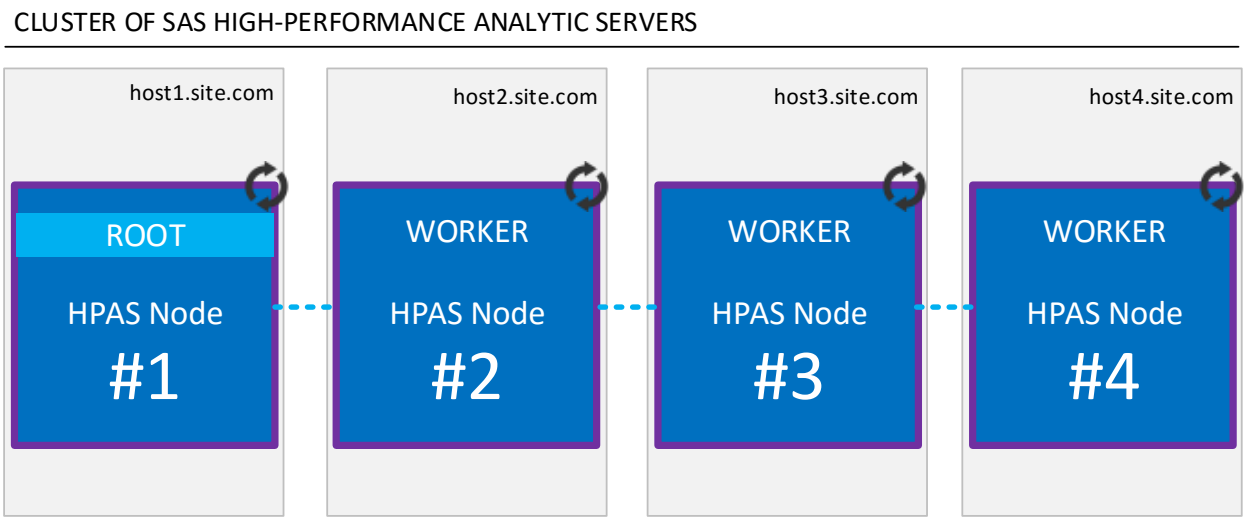


Figure 8. Distributed SAS High-Performance Analytics Server for Large Data Tasks

Scalability is also the primary objective here. The availability situation might be considered slightly better than SAS LASR Analytic Server, however. That's because each new request instantiates a new in-memory service. So it's possible that the loss of a host machine could go unnoticed if it occurs between requests. However, just like SAS LASR Analytic Server, if an SAS High-Performance Analytics Server host machine must remain offline, then manual configuration might be required so that a new instance of SAS High-Performance Analytics Server can run on the remaining hosts.

Clustering Deployment Considerations:

Deploying the Distributed SAS High-Performance Analytic Server software is rated as **MEDIUM** difficulty.

The SAS Deployment Wizard does not deploy SAS High-Performance Analytics Server. Instead, installation is performed at the Linux command-line and requires root-level privileges to complete. SAS LASR Analytic Server and SAS High-Performance Analytics Server rely on the same underlying software to operate in MPP mode - TKGrid. So a single deployment of TKGrid with proper licensing can host both SAS LASR Analytic Server and SAS High-Performance Analytics Server running at the same time. Be

aware that SAS LASR Analytic Server and SAS High-Performance Analytics Server use memory independently of each other. And while you can process a LASR table with SAS High-Performance Analytics Server, realize that execution of the HP PROC will make its own in-memory copy of that data before processing.

Summary:

Objective	Massive scalability
Operation	All nodes critical SAS High-Performance Analytics Server Root Node manages requests and directs the Worker Nodes
Cluster Architecture	MPP Minimum 4 server hosts
Client discovery	SAS High-Performance Analytics Server Root Node host, port, and other connection information are specified in your SAS program code
Deployment difficulty	Medium Requires system administration privileges in Linux (and possibly Hadoop) Also possibly third-party hardware and/or software support

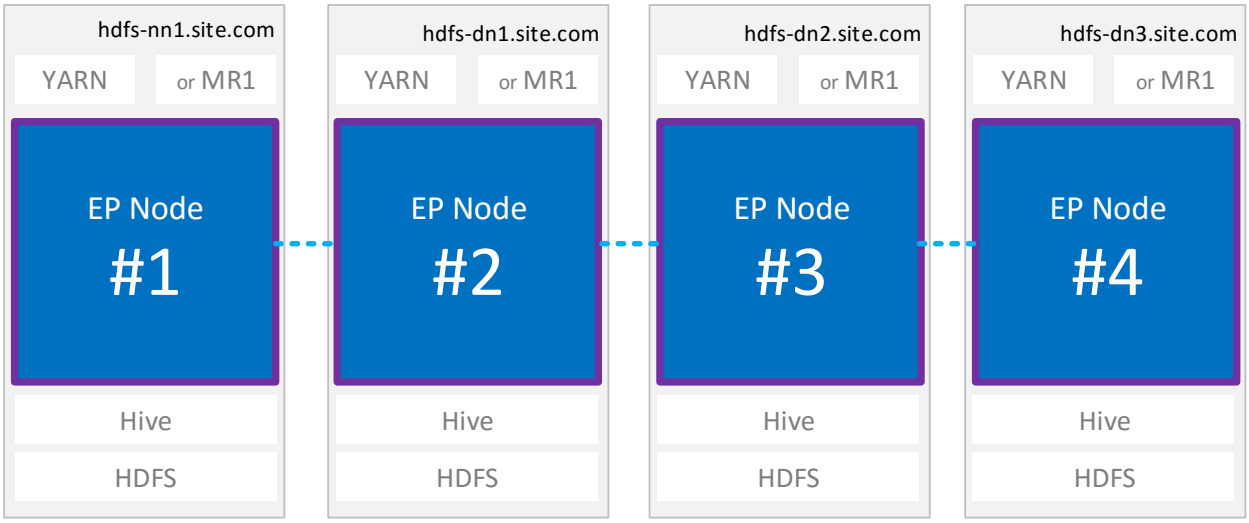
08 – SAS IN-DATABASE EMBEDDED PROCESS (for Hadoop)

The SAS Embedded Process is part of the SAS In-Database technology offerings. The idea behind SAS In-Database is to move analytic capabilities to where the data resides already (versus copying data from a central store over to analytic services hosted elsewhere). The embedded process achieves this goal beautifully.

Clustering Functions

SAS offers the embedded process for several third-party data providers. In this paper, we will look at the SAS Embedded Process for Hadoop. The embedded process software is deployed to an existing Hadoop cluster. An instance of the embedded process runs on each and every Hadoop node responsible for executing MapReduce jobs. In this way, the embedded process is automatically in a position to leverage the existing Hadoop abilities supporting availability and scalability while participating in the Hadoop load balancing scheme via YARN.

MPP CLUSTER OF SAS EMBEDDED PROCESS



Beyond performing analytics within the Hadoop cluster, the embedded process can also perform data transfer from each embedded process node directly to each node of a SAS LASR Analytic Server or SAS High-Performance Analytics Server cluster. This is known as parallel data transfer and provides a fast and efficient means to copy data over to your in-memory analytics solution.

The SAS Embedded Process also supports deployment to other database technology vendors – and in those situations, relies on specific requirements to deploy and run properly into each of those environments.

Clustering Deployment Considerations:

Deploying the SAS Embedded Process software is rated as **MEDIUM** difficulty.

The SAS Deployment Wizard does not deploy the embedded process. Instead, installation is performed one of three ways:

1. At the Linux command-line on the Hadoop cluster and requires root-level privileges to complete.
2. Use SAS Deployment Manager to create a Cloudera parcel for the embedded process and then Cloudera Manager can deploy to the Cloudera Distribution Hadoop cluster.
3. Use SAS Deployment Manager to create a stack for Hortonworks for the embedded process and then Ambari can deploy to the Hortonworks Data Platform cluster.

In order to install the embedded process in Hadoop, you must have administration-level privileges to manage HDFS. Database permissions to access data in Hive (or another supported DBMS based on Hadoop) are needed as well.

Summary:

Objective	Massive scalability while relying on the Hadoop load balancing and availability model
Operation	All nodes are used The EP instantiates on each node as a MapReduce job and works with the data locally available
Cluster Architecture	MPP The EP deploys to all MapReduce hosts

Client discovery	Host, port, and other connection information for the SAS Embedded Process are specified in your SAS program code
Deployment difficulty	Medium Requires system administration privileges in Linux and Hadoop Also possibly third-party hardware and/or software support

09 – SAS CACHE LOCATOR

The SAS Cache Locator is a specialized implementation of Pivotal GemFire. It provides service discovery functionality and informs new, connecting members (like the SAS Web Application Server) where other running members are located.

Clustering Functions

The members of a SAS Cache Locator cluster act as peers and provide failover support for each other. If one goes down, the other can continue doing all of the work.

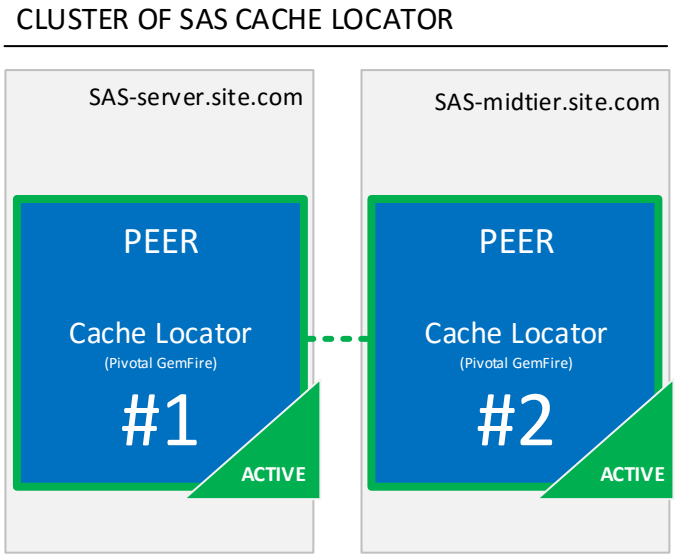


Figure 9. Automatically Deployed Cluster of SAS Cache Locator

Clustering Deployment Considerations:

Deploying the SAS Cache Locator as a cluster is rated as **TRANSPARENT** difficulty.

For multi-machine deployments, the SAS Deployment Wizard will automatically deploy two SAS Cache Locators in a clustered arrangement. One is placed on the primary middle-tier host machine and the other on the primary server-tier host.

Summary:

Objective	Improved scalability and availability of the Cache Locator service
Operation	Active-Active failover Both nodes participate in task execution
Cluster Architecture	SMP Minimum 2 server hosts

Client discovery	Host and port of cluster nodes automatically configured in participating members
Deployment difficulty	Transparent Cluster deployment is fully automatic in multi-tier deployments

10 – SAS JMS BROKER

The SAS JMS Broker is a SAS deployment of Apache ActiveMQ software. It provides Java message services for SAS solutions.

Clustering Functions

Clustering is achieved through the use of a master-slave paradigm where all instances rely on a shared file system to access common files. To ensure high availability of the JMS Broker, then the shared filesystem for common files must also be hardened against failure.

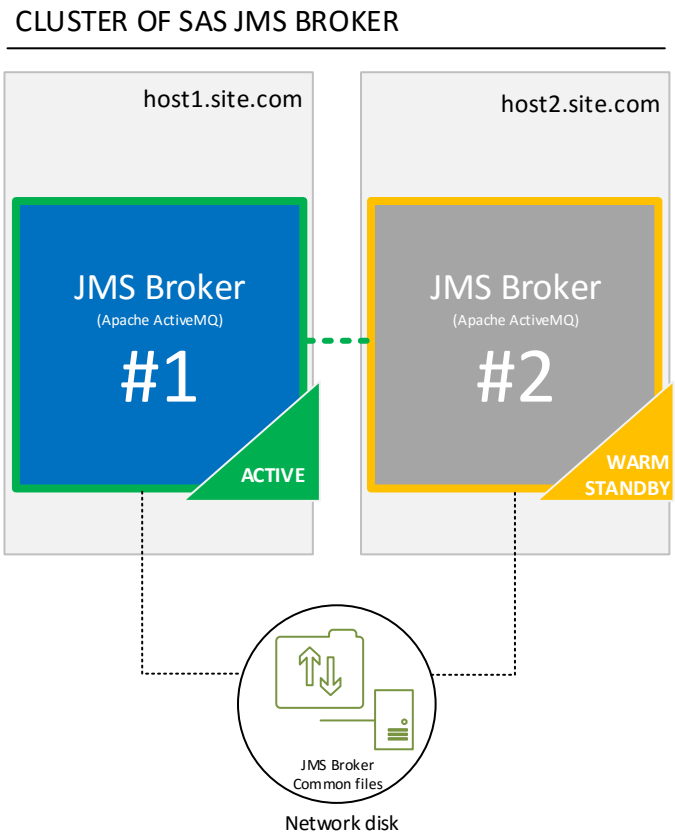


Figure 10. SAS JMS Broker after Manual Deployment with a Failover Node

Clustering Deployment Considerations:

Deploying the SAS JMS Broker as a cluster is rated as **HARD** difficulty.

The SAS Deployment Wizard will only deploy a single JMS Broker instance. The broker can be clustered through additional manual steps to eliminate the single point of failure and improve availability.

Summary:

Objective	Improved availability of the JMS Broker service
Operation	Active-Passive failover with a Warm Standby node Only one node is active to execute tasks
Cluster Architecture	SMP Minimum 2 server hosts
Client discovery	Host and port of cluster nodes automatically configured in participating members
Deployment difficulty	Hard Requires environment and system administration privileges as well as using third-party documentation.

11 – SAS® ENVIRONMENT MANAGER SERVER

The SAS® Environment Manager is a web application that provides an administrative and monitoring interface for your SAS solution through your web browser.

The SAS Environment Manager interface relies on an infrastructure built using VMware vFabric Hyperic software, which consists of a SAS Environment Manager *Server* and SAS Environment Manager *Agents.* An agent is automatically deployed by the SAS Deployment Wizard to each host machine in your SAS solution and collects operational metrics, resource utilization, and so on. The agents then transmit their data to the SAS Environment Manager Server, which stores the data in a central database – the SAS Web Infrastructure Data Server – for monitoring and reporting purposes.

In its way then, the SAS Environment Manager functionality requires its own complex, multi-host architecture. Improving the availability of SAS Environment Manager means enabling clustering for those other pieces as well. However, for this part of our discussion about clustering, we focus only on the SAS Environment Manager Server.

Clustering Functions

Clustering the SAS Environment Manager Server improves availability of the service by mitigating the single node of failure. It requires manual effort to deploy a second node. A third-party load-balancer must also be used to automatically redirect traffic to the warm standby node if the primary node goes offline.

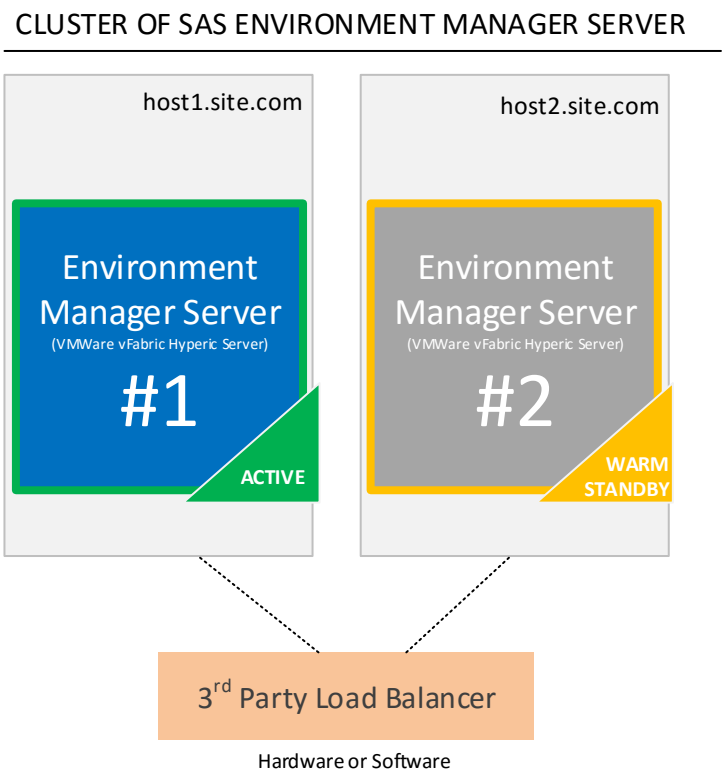


Figure 11. SAS Environment Manager Server Clustered for Improved Availability

Clustering Deployment Considerations:

Deploying the SAS Environment Manager Server as a cluster is rated as **HARD** difficulty.

The SAS Deployment Wizard will only deploy a single SAS Environment Manager Server. The Server component can be clustered through additional manual steps to eliminate the single point of failure and improve availability.

Summary:

Objective	Improved availability of the SAS Environment Manager Server
Operation	Active-Passive failover with a Warm Standby node Only one node is active to execute tasks
Cluster Architecture	SMP Minimum 2 server hosts
Client discovery	Clients connect to the third-party load balancer
Deployment difficulty	Hard Requires environment and system administration privileges as well as using third-party documentation.

12 – SAS® WEB INFRASTRUCTURE PLATFORM DATA SERVER

The SAS® Web Infrastructure Platform Data Server is built using a PostgreSQL 9.1.9 RDBMS and Pgpool-II 3.3.4 middleware. It provides transactional data storage for use by SAS middle-tier and SAS solutions software.

Clustering Functions

The SAS Web Infrastructure Platform Data Server can be clustered to mitigate a single point of failure to improve availability. The Postgres software provides a standard approach to this type of cluster where a second instance of the PostgreSQL server runs as a warm standby node with its own copy of the database in Read-Only mode. Streaming replication is used so that the contents of the live database are copied in near real time to the standby database.

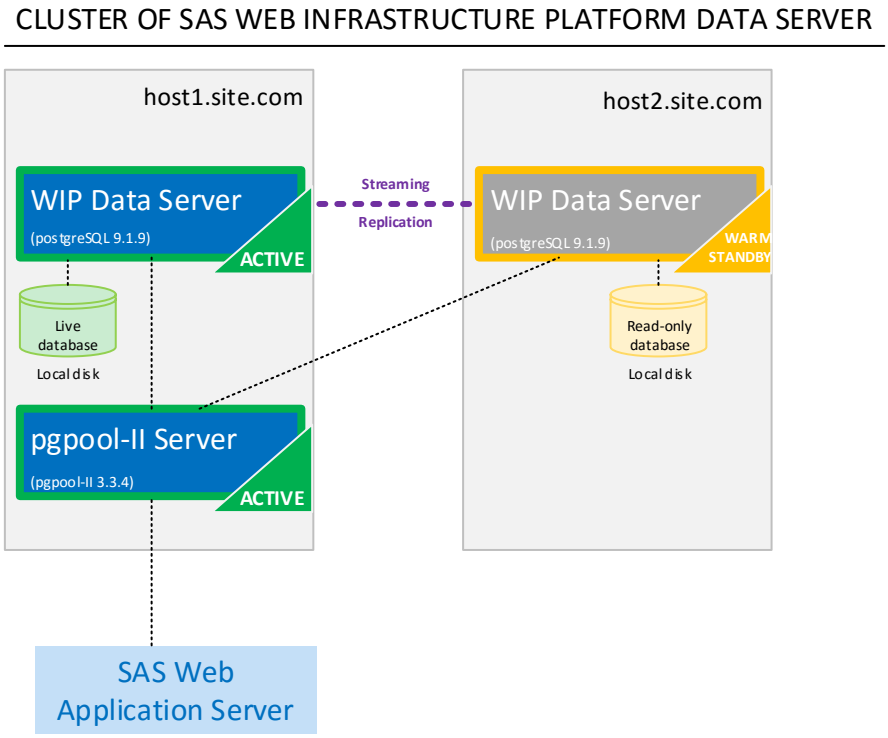


Figure 12. SAS Web Infrastructure Platform Data Server Clustered for Improved Availability

Clustering Deployment Considerations:

Deploying the SAS Web Infrastructure Platform Data Server as a cluster is rated as **CHALLENGING** difficulty.

The SAS Deployment Wizard will only deploy a single SAS Web Infrastructure Platform Data Server service. The SAS Web Infrastructure Platform Data Server can be clustered through additional manual steps to eliminate the single point of failure and improve availability.

Clustering of the SAS Web Infrastructure Platform Data Server must be enabled at initial deployment of the SAS solution. A single instance cannot be retro-fitted to operate as a cluster.

By default, the SAS Deployment Wizard configures a different PostgreSQL Server instance for each SAS database. So enabling high availability must be considered for the SharedServices database, the SAS Environment Manager database, any SAS solution-specific databases, and so on.

In this explanation, we did not eliminate the one instance of Pgpool-II Server as a single point of failure. SAS Technical Support is working with SAS R&D to evaluate additional steps for that as well.

Never configure more than one primary SAS Web Infrastructure Platform Data Server for use simultaneously. Doing so can result in data loss or corruption that can be very difficult to recover from

without full backups of the database(s). Make regular backups of the database(s) as a best practice. The high-availability configuration is not a substitute for making regular backups.

Summary:

Objective	Improved availability of the SAS databases in SAS Web Infrastructure Platform Data Server
Operation	Active-Passive failover with a Warm Standby node Only one node is active to execute tasks
Cluster Architecture	SMP Minimum 2 server hosts
Client discovery	SAS High-Availability JDBC shim, which redirects queries and connections to the database on the running host
Deployment difficulty	Challenging Limited applicability; involves manifold expert configurations; requires environment and system administration privileges as well as using third-party documentation.

CONCLUSION

There are twelve different cluster technologies used by the SAS 9.4 platform.

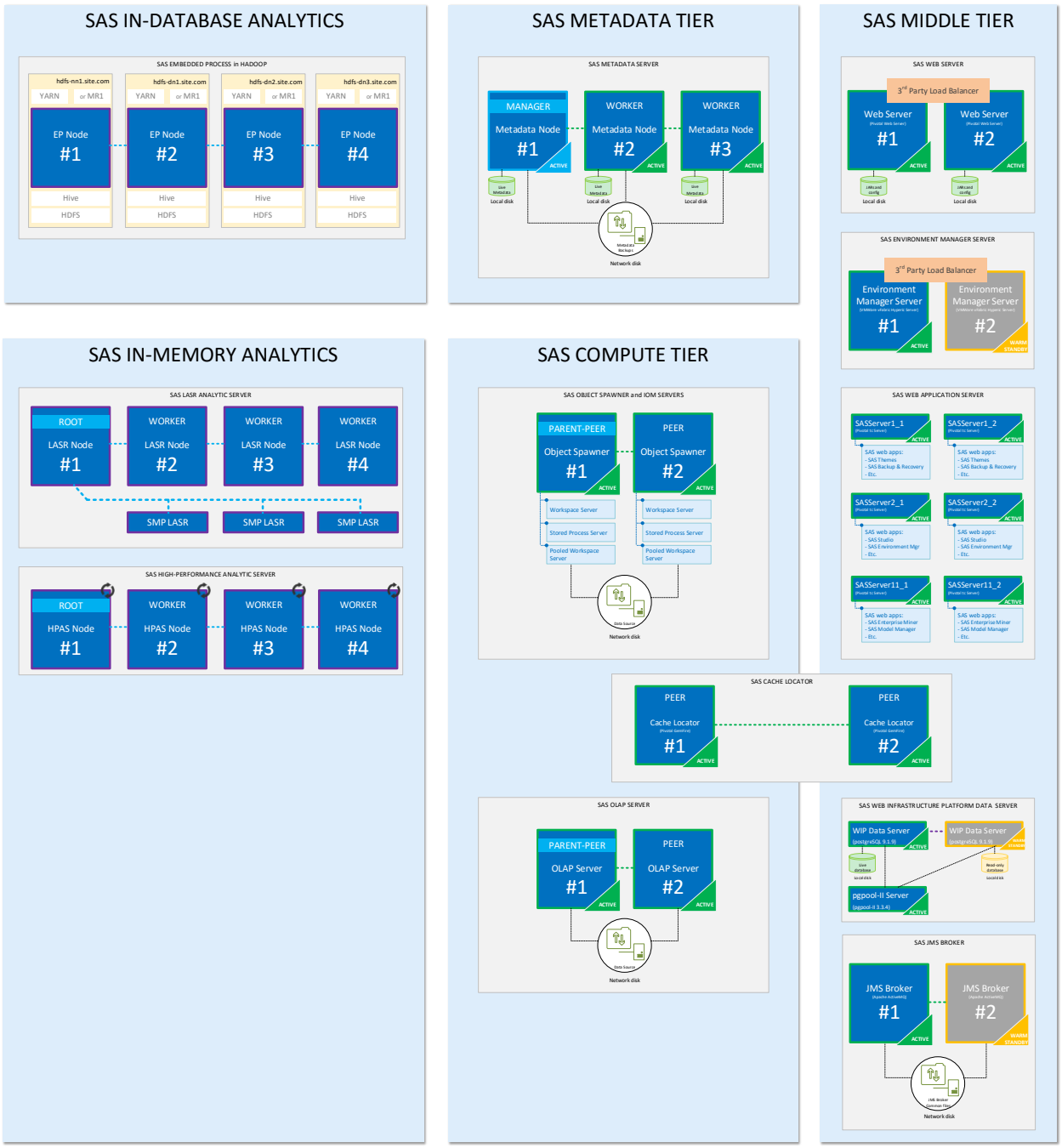


Figure 13. Twelve Cluster Technologies Available in SAS 9.4

Clustering the components of the SAS software solution offerings improves the services' ability to scale up for greater load, eliminates any single points of failure to improve availability of the services, and in some cases, can do both.

The techniques for clustering vary from one software component to another – as does the considerations around deployment and usage. In many cases, the installation utility for each component of the SAS software can automatically configure individual services to operate as a cluster. However, a few SAS products will require manual deployment and configuration to provide a cluster.

ACKNOWLEDGMENTS

I would like to thank the following people for taking the time to review and contribute to this paper:

- Scott McCauley
- Edoardo Riva
- Mark Thomas
- Simon Williams

RECOMMENDED READING

For more information about the different clustering technologies used by SAS 9.4 software components, read the following.

SAS METADATA SERVER

- *SAS 9.4 Intelligence Platform: System Administration Guide, Fourth Edition*

SAS OBJECT SPAWNER

- *SAS 9.4 Intelligence Platform: Application Server Administration Guide*

SAS OLAP SERVER

- *SAS 9.4 Intelligence Platform: Application Server Administration Guide*

SAS WEB APPLICATION SERVER

- *SAS 9.4 Intelligence Platform: Middle-Tier Administration Guide, Fourth Edition*
- *SAS 9.4 Guide to Software Updates*
  - Exceptions to Middle-Tier Clustering Support

SAS WEB SERVER

- *SAS 9.4 Intelligence Platform: Middle-Tier Administration Guide, Fourth Edition*

SAS LASR ANALYTIC SERVER

- *SAS High-Performance Analytics Infrastructure 3.5 Installation and Configuration Guide*
- *SAS LASR Analytic Server 2.8: Reference Guide*

SAS HIGH-PERFORMANCE ANALYTIC SERVER

- *SAS High-Performance Analytics Infrastructure 3.5 Installation and Configuration Guide*

SAS IN-DATABASE EMBEDDED PROCESS

- *SAS In-Database Products: Administrator's Guide, Eighth Edition*

SAS CACHE LOCATOR

- *SAS 9.4 Intelligence Platform: Middle-Tier Administration Guide, Fourth Edition*

SAS JMS BROKER

- *SAS 9.4 Intelligence Platform: Middle-Tier Administration Guide, Fourth Edition*
- Original ApacheMQ documentation
  - Features > Clustering > MasterSlave > **Shared File System Master Slave**
    http://activemq.apache.org/shared-file-system-master-slave.html

SAS ENVIRONMENT MANAGER

- *SAS 9.4 Intelligence Platform: Middle-Tier Administration Guide, Fourth Edition*
- Original VMware vFabric Hyperic documentation
  - vFabric Hyperic Configuration > Configure and Run the Hyperic Server > **Clustering Hyperic Servers for Failover**
    http://pubs.vmware.com/vfabricHyperic50/index.jsp?topic=/com.vmware.vfabric.hyperic.5.0/Clustering_Hyperic_Servers_for_Failover.html

SAS WEB INFRASTRUCTURE PLATFORM DATA SERVER

- *SAS 9.4 Intelligence Platform: Middle-Tier Administration Guide, Fourth Edition*
- Technical papers on support.sas.com
  - *Configuring the SAS Web Infrastructure Platform Data Server for High Availability*
  - *Managing SAS Web Infrastructure Platform Data Server High-Availability Clusters*

CONTACT INFORMATION

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

Rob Collum
SAS Institute Inc.
rob.collum@sas.com
http://www.sas.com