Paper 312-2008

Backup and Disaster Recovery: When Disaster Strikes, What Will You Do? What Will You Do?

Arthur Hunt, SAS Institute Inc., Cary, NC
Tanya Kalich, SAS Institute Inc., Cary, NC
Billy Dickerson, SAS Institute Inc., Cary, NC
Chuck Antle, SAS Institute Inc., Cary, NC

ABSTRACT

This presentation provides a conceptual framework for creating a backup-and-disaster recovery plan (also known as a Service Continuity Management plan) for your SAS® solution. It covers often-overlooked subtleties in backing up the SAS® Metadata Repository, the WebDAV repository, and the associated configuration files, as well as ensures synchronization of the pieces that are required to work together. This discussion will help administrators avoid the common pitfalls and ensure that their site is able to be up and running as soon as possible in the event of small or large disasters.

INTRODUCTION

The subject of this discussion, backup and recovery of SAS solutions, is for the following groups of people:

- everyone who cannot afford to lose their SAS environment and the work that they have done on it
- IT professionals, in particular, those who are familiar with backup and recovery options, but might not be familiar with SAS requirements specifically.
- SAS professionals who are unfamiliar with backup and recovery
- SAS Business Intelligence Platform administrators who need to know more about the considerations specific to that platform.

This paper covers the general considerations for backing up and recovering your SAS environment. It also describes the authors' own experiences and conclusions reached in creating a backup-and-recovery process for their internal environment at SAS headquarters.

CONSIDERATIONS FOR CREATING A DISASTER RECOVERY PLAN FOR YOUR SAS ENVIRONMENT

In an ideal world, nothing would ever happen to your environment. Hardware problems, viruses, and user errors would never happen, so disaster planning would be unnecessary. In an *almost-ideal* world, if any disasters did occur, you would have a perfect up-to-the-minute backup that could be restored instantly, making recovery almost transparent to all system users. Given that even this more modest goal is prohibitively expensive in most cases, what should the goals be for backup and recovery of your SAS environment?

In general, the goal of an information technology (IT) backup-and-recovery plan is to create a process that restores business-critical processes and data within the following parameters:

- within an acceptable time-frame
- within the given budget
- without undue impact on normal day-to-day operations

Creating such an IT recovery plan helps to ensure the survival of your business by reducing the impact of a disaster or major failure. Focusing on effective risk analysis and risk management enables you to reduce that risk, preventing a loss of confidence by users and customers. But how do you do it? In the process of creating a plan for SAS, our team realized that there are five or six steps that you have to go through to create a disaster recovery plan:

1. Define what software and data your organization cannot do without.
2. Evaluate how much time will be required to recover from a disaster.
3. Set a budget for disaster recovery.
4. Define acceptable levels of inconvenience and productivity associated with implementing your backup strategy.
5. Consider the different recovery paths for different types of failures.
6. Determine any *upstream* data sources that you need to account for in the case of a failure. *Upstream data sources* are those data or processes that happen before or after your primary data processing.

STEP 1: DEFINE WHAT SOFTWARE AND DATA YOU CANNOT DO WITHOUT

First, you need to determine what business functions have the highest priority for recovery. In the event of a widespread failure, prioritize the order and complexity of what needs to come back online first. Inventory your business's software and data, and then prioritize that list according to how critical each system is to the performance of your business. This inventory and categorization of your software systems will provide the map for you to begin to define the physical backup processes and procedures.

STEP 2: EVALUATE THE TIME REQUIRED TO RECOVER FROM A DISASTER

Consider the time it will take to recover if a disaster does occur. What is the maximum amount of time your business could survive without each of your critical hardware and software systems? The answer to this question will determine what extra hardware you will need to have available at an off-site location. It also determines the type of backups you need to make. If downtime will be very costly, it might be worthwhile to create a mirror image of your entire system, or at least some subset of it. Disk mirroring of various sorts can greatly decrease recovery time. A number of hardware and software products are available, under the category Redundant Arrays of Independent/Inexpensive Disks (RAID) that you can use to create a mirror image.

STEP 3: SET A BUDGET FOR DISASTER RECOVERY

Decide how much time, energy, and money you will need to expend in order to recover from disaster. The dollar figure is likely to be closely related to how much capital you might lose in the case of a disaster and to how easily your business could absorb that loss. Other major factors that will drive the cost of your backup-and-recovery solution are the sheer volume and complexity of the material that needs to be backed up. These factors can also determine what media type you use for backup; for example, disk-to-tape, disk-to-disk, satellite, or a combination of types for short-term versus long-term storage.

STEP 4: DEFINE ACCEPTABLE LEVELS OF INCONVENIENCE AND PRODUCTIVITY ASSOCIATED WITH IMPLEMENTING A BACKUP STRATEGY

Determine how much inconvenience and loss-of-processing time you are able to accept in order to implement your backup strategy. The task of creating the copies of your current working environment will take time and resources away from the production processes that are already in place.

You will also need to schedule your backups to occur at regularly scheduled intervals. But at what intervals should you schedule backups? Weekly? Daily? The timing of the backups is based on how dynamic the environment is; how expensive it would be to re-create the environment in the event of a failure; and your required service levels and commitments to your users, customers, and other businesses. These and other factors can affect when you schedule your backups to run, and some of these factors compete with each other in priority. Examples of such factors include:

- data volume
- data complexity (relational data, snapshots, longitudinal matching, adjudication, and so on)
- the RAID level of what you are backing up
- users' schedule needs

You will also want to make sure that your schedule does not coincide with data ETL schedules. Indeed, many backups occur as part of the ETL process, so some data might already be covered.

In addition to the regularly scheduled backups, we highly recommend creating backups before and after any significant changes, such as installation of a new product, a new release, or even a new hot fix.
You will also need to document the resources (administrators, storage, and system downtime) that will be required to create the backups, and you will need to consider these resources as you create your plan.

STEP 5: CONSIDER THE DIFFERENT RECOVERY PATHS FOR DIFFERENT TYPES OF FAILURES

You need to consider the variety of different failures from which you might need to recover:

- server failures: full or partial server failure
- storage failures: full or partial storage failure (device, file system, volume, logical unit number [LUN], array)
- application or middle-tier failures: full or partial failure of the application or middle-tier software or servers

Each failure comes with a different recovery path. Plan for and practice the recovery path for each of these types of failure.

STEP 6: DETERMINE ANY UPSTREAM DATA SOURCES THAT YOU NEED TO CONSIDER IN THE CASE OF A FAILURE

Many industries, such as finance, insurance, and retail, have huge volumes of user-generated data coming into their systems daily, 24 hours a day. If your organization has this situation, it is crucial to consider how you will handle the data if a system failure means that it cannot be processed in a timely fashion, and you need to make sure you have adequate storage to accumulate the data while your system is in recovery. This step did not apply in our situation because SAS does not have upstream data that is generated by users being fed into our system.

CREATING A BACKUP STRATEGY

Once you have created your plan, consider some of the physical aspects for creating your backups. Unfortunately, there are far too many aspects than can be covered here, so this paper touches on some of the most critical, or often overlooked, aspects. These aspects are divided into two categories: general system considerations and considerations that are specific to your SAS deployment.

GENERAL SYSTEM CONSIDERATIONS

The first consideration is one that we call *the elephant question*. As the old story goes, if you try to eat an elephant, your only hope of finishing it is to divide the elephant into smaller, more easily digested portions. The same is true when you tackle an *elephant* as large as the effort of backing up all of your critical IT systems.

Start by looking at your backup methodology with a business-function approach. First, list and prioritize the business units that use your system and focus first on the highest priority units. Once you create that list, for each business unit, examine the components that those units use and focus on the administrative domains across which those components are distributed. Many of these distributed components might be shared across multiple business units, so you should focus on your top business priorities first.

Then, you need to subset things even further by administrative domain. *Administrative domains* comprise collections of networks, computers, and databases that are administered together. They generally share authentication information, and they might share network or operating-system resources. These domains can be as small as a user's database or as large as a wide-area network (WAN), but they are the basic chunks that your system administrators recognize. For simple configurations, each separate host can be its own administrative domain. Web servers that perform authentication are often their own administrative domains.

Your task is to list these administrative domains for each of the business functional units that work in your SAS BI deployment. Once you have created this list, go through it one domain at a time and consider all the issues that might arise with synchronization, both within that domain and in interaction with other domains. This step, along with others discussed later, is particularly crucial for SAS Business Intelligence systems. Basic questions you need to consider with regard to the administrative domains:

- What applications use data from multiple domains?
- What applications use data from this domain?
- Could any applications try to write to this domain while we are creating a backup?
- When do these applications run?
- Are the applications interactive or batch?
- Who has access to the applications?
- How and when can you restrict access without impacting essential business processes?
- What are the consequences if an attempt by each application to read or write during the backup process is unsuccessful?
- What are the consequences if an attempt by each application to read or write during the backup process is successful?
- Is there data on this domain that must be synchronized with data on another domain?
- What are the consequences of unsynchronized domains?
- Will both domains need to be taken down at the same time to ensure synchronization?

Once you have answered the questions about synchronizing and *quiescing* (rendering inactive) the administrative domains, you can:

- identify the critical pieces within each domain that you need to copy and store
- identify the important operating-system, application, and data files to back up
- decide where to store the backups, how many backups to keep, how long to keep backups, and how to archive the backups.

As is the case with the frequency of backups, these decisions depend on how dynamic the system is and how much your business can afford to lose.

Even a very well-organized backup-and-restore process is only as good as your ability to restore, and there will almost always be occasional backups that are not usable. As part of your backup plan, you should schedule and execute tests of your ability to restore your systems from one of the backups you have created. You should also keep a number of backups in reserve in case the most recent backups are unusable.

Documentation is the final consideration in the list of critical or overlooked aspects of the physical backup. A well-documented backup-and-recovery plan is even more essential than lining up the proper emergency hardware and off-site storage. It is also more essential than creating the correct schedule for your backups. This documentation is so important because without detailed instructions for what is being done, why it is being done, and how to make use of the results, all of the work and expense is pointless. Your backups must be usable, which means the instructions for their creation and use must be foolproof. Time spent writing, reviewing, and updating this documentation will enable you to see holes in your process that you might not discover any other way. That detailed plan will save you in multiple ways when the time comes to implement the plan.

SPECIFIC SAS DEPLOYMENT CONSIDERATIONS

What specific aspects do you need to consider with regard to your SAS deployment, and even more particularly, your SAS business intelligence deployment? For a start, you need to consider what process components and data sources you have that are specific to your SAS BI deployment. The following list provides all of the main components of a SAS BI deployment, organized by the tier in which they appear. Note that each of these tiers can include additional components that are not listed here.

Server Tier

- SAS® Metadata Server
- Workspace server
- SAS® Stored Process Server
- SAS®OLAP Server
- SAS/CONNECT® server
- SAS/SHARE® server
- batch SAS server
- database server
- SAS® Scalable Performance Data Server® and the SAS® Scalable Performance Data Engine
- single-user SAS invocations
- LSF Scheduler (Platform Process Manager)
- SAS/ACCESS® software products

Middle Tier

- Web Application Servers and all middle-tier applications that are installed on them, such as the following:
    - SAS® Web Report Studio
    - SAS® Information Delivery Portal
    - SAS® Web OLAP Viewer for Java
    - SAS solution applications (such as SAS® Marketing Automation)
- WebDAV server for 9.1.3 Xythos (recommended)
- SAS Remote Services/SAS Services Application

Client Tier

- SAS® Enterprise Guide®
- SAS® Add-In for Microsoft Office
- SAS® Stored Processes client
- SAS Solutions desktop clients, such as SAS® Customer Intelligence Studio

You should make a similar list of the components in your SAS deployment because each of these components can have special considerations for backup and recovery. For example, the SAS® Metadata Server is a process that keeps metadata in resident memory while it is executing. To create a complete and consistent backup of the content within the SAS Metadata Server, you must pause the server and use one of the tools provided with it to flush the contents of its memory to disk. These tools include the %OMABAKUP macro, the METAOPERATE procedure, and the **PAUSE** action in the SAS Metadata Manager plug-in of SAS® Management Console. A comprehensive list of your SAS® System components enables you to identify any special considerations for each of the components on your list.

We also recommend that you list your components by tiers as well, primarily because it is a logical way to organize the inventory of your SAS system component processes. This structure will also help you to identify dependencies and interactions when you are creating your backup-and-recovery plan. Each of the tiers (and even each of the components) can be distributed across different administrative domains, and having this inventory list, categorized by tier, helps you quickly see and organize those dependencies.

TAKING INVENTORY OF YOUR DATA RESOURCES AND THEIR DEPENDENCIES

Once you have a complete record of the SAS process components, you need to create an inventory of your data resources and any dependencies that those data resources might have across the system. As is done in the following example inventory, you should document the location for each data resource, which processes make use of it, and any considerations specific to that resource:

- Metadata repositories:
  - You must quiesce the server in order to create a backup.
  - The use of the %OMABAKUP macro is recommended in order to force metadata to the disk.
  - You must have permission (privileges) in order to perform a backup and to manage the server.
  - The process needs to be quick in order to minimize downtime of SAS applications that use the metadata server.
  - Schedule the backup during low usage hours.
- SAS data sets: Must not be accessed in Write mode or some backup software will not back them up. Other backup software might back up the data sets. However, when they are restored, SAS might flag them as corrupt, so the data sets will be unusable.
- SAS catalogs: Must not be accessed in Write mode or some backup software will not back them up. Other backup software might back up the catalogs. However, when they are restored, SAS might flag them as corrupt, so the catalogs will be unusable.
- SAS Scalable Performance Data Server (SPD Server) and the SAS Scalable Performance Data Engine (SPD Engine) table data:
  - This data requires special utilities in order for it to be backed up.
  - These tables consist of many partitions that can be accessed only as a single large table. Standard utilities cannot perform incremental backups. But the special utilities provided with the SPD Server and the SPD Engine can handle incremental backup
  - See *SAS®Scalable Performance Data Server® 4.43: Administrator's Guide* for details on the following SPD Server utilities:
    - SPDSLS
    - SPDSBKUP
    - SPDSRSTR
  - Permissions (privileges) are required in order to perform backups of SPD Server and SPD Engine data.
- OLAP cubes:
  - OLAP cubes are typically large.
  - They cannot be backed up incrementally.
  - They can be regenerated from other data sources.
  - They use SPD Engine table formats and tend to be large in size
- SAS source code directories: Are they centrally located, or are they distributed across machines and administrative domains?
- SAS installation directories for servers, middle-tier applications, clients, and solutions:
  - If the installation directories of the various tiers of your SAS deployment span multiple administrative domains, you need to coordinate the backups across those domains.
  - The maintenance levels need to be the same across all tiers.
- SAS configuration directories for servers, middle-tier applications, clients, and solutions: Require special privileges (permissions) to access them.
- Script and configuration:
  - Some operating systems might require that the server start-up primitives and some SAS configuration mechanisms be stored outside of the default SAS configuration directories.
  - There might be third-party configuration components as well.
  - You must know where all of these components are located, and you must be certain that they are being backed up according to their individual requirements, in a synchronized manner.
  - These script and configuration components might require special privileges (permissions) to access them.

- Raw data sources:
    - flat files
    - spread sheets
    - tables
    - system logs

  These sources can take almost any form and reside almost anywhere on the system.
- Operating-system automatic start mechanisms: Solaris Management Facility (SMF), Lightweight Directory Access Protocol (LDAP), cron jobs, and other scheduling and automation pieces
- User directories
- Departmental directories
- Middle-tier application server directories (especially the configuration directory): In many cases, products can be reinstalled quickly if the configuration directory can be recovered.
- Middle-tier data sources
- Third-party entities such as Xythos, WebSphere, and WebLogic: Refer to third-party documentation for specifics on backup and restoration.
- Databases:
    - Refer to the specific database administration guide for details on backup and restoration.
    - If you are using a database as a backing store for distributed authoring and versioning (DAV), synchronization is important.
    - Databases might be under a separate administrative domain.
- Clients data:
    - The data might be distributed across desktop machines
    - The data might be on something like a Citrix Server
    - The data might be in a separate administrative domain
    - Maintenance level must be synchronized with the middle and server tiers.
- Scheduler (Platform Process Manager) work area
- Windows Registry
- Other: Any other components you think of that are not listed here.

The backup of your SAS deployment will work only if all of the dependent data sources are synchronized. They can be synchronized only if you have made a correct and complete list of your data sources and all their dependencies.

Some of the most common dependencies in a SAS BI deployment include:

- **Metadata repositories:** Metadata repositories within the same metadata server are almost always dependent upon one another. The standard process for backing up your metadata server keeps these synchronized by quiescing the server and using %OMABAKUP to flush the repositories to the disk in a consistent state. The mechanics of this process are documented in the SAS Intelligence Platform Administration Guides. Note that there are other approved mechanisms for flushing your metadata repositories to disk, such as the use of Proc Meta Operate, or the use of SAS Management Console.
- **Stored processes and their metadata:** Ensure that the backup of your stored processes is synchronized with the backup of their associated metadata.
- **WebDAV server (for example, Xythos) and the WebDAV metadata stored in the database:** Many WebDAV servers store metadata about their data content in a separate database. These back-end databases must also be kept in synch for the WebDAV to work correctly when it's restored.
- **WebDAV content and content in the SAS Metadata Server:** These must be synchronized for many SAS Solutions because of the tight coupling between WebDAV content and metadata.
- **Clients, Middle Tier, and Servers:** These must be at the same maintenance levels to work correctly together.

Once you have identified these dependencies, you need to define a strategy to back up each group of interdependent data sources. Make sure that during the backup of the group, every one of the data sources within the group is inaccessible to processes and users that could affect that backup. Preventing this contention for every data source in a group is the only way to guarantee that the interconnected elements will stay synchronized. If even one of the data sources within an interdependent group is out of synch, all of the data sources might be unusable.

To do this, we recommend that you have a synchronized stoppage of all of the SAS components. The metadata server must be quiesced in order to create a backup of the repositories. Other SAS BI applications must also be stopped in order to get complete backups of their components and data sources. But these steps are only a part of the contention that needs to be addressed. Individual users are the other part.

On most operating systems, you can create or purchase tools that enable system administrators to identify which users and processes are using what data sources. Once these users and processes are identified, their activities

must be stopped before the backup takes place. This process can involve anything from requesting that users free up those resources to intentionally removing users' access to the resources and stopping particular processes. To determine the appropriate path, you need to ask yourself how important it is to be able to reliably restore your SAS® System in a consistent state.

SCHEDULING BACKUPS

Create a schedule for your backups so that users can plan their activities around the times that the system will be unavailable. Having a set schedule helps to minimize users' frustration as well as contention for resources during the backup process. You should also schedule your backups to occur during a time when the SAS BI system usage is low. Ask yourself the following questions:

- How much activity is there on the system? The more activity there is, the more frequently you will need to back up your system.
- How much work can we afford to lose if there is a failure? Having longer periods between backups means more work that could be lost in the event of a failure.
- Are there any facilities or tricks available that might enable me to shorten the window of time that the system is unavailable?

TYPES OF BACKUPS

You can have the following types of backups:

- full
- incremental
- differential
- some combination of all three of the types

Full backups facilitate the quickest path to recovery because in the case of a failure, only one backup needs to be restored in order to recover the entire system. However, this type of backup takes the longest time to produce.

Incremental backups only back up the information that has changed since the last backup process ran. As a result, this type of backup runs faster. The downside is that in order to restore a system, you need to start with the last full backup and then restore all subsequent incremental backups until you have the system back to its last known useful state. This process can be very time-consuming.

Differential backups save everything that has changed since the last full backup was performed. As more of the deployment changes, the differential backups will take longer and longer to perform. The benefit of this type of backup is that when a recovery is needed in the case of a disaster, you have to restore, at most, only two backups: the last-known good full backup and the subsequent, good differential backup.

Generally, you should schedule a full backup to run weekly (over the weekend or at some other convenient time) and then use incremental or differential backups the rest of the week. This process minimizes the amount of work that can be lost through the week and, at the same time, it minimizes the daily impact of the backup process to the users.

You will need to design a backup strategy for your site and application that has the appropriate mix of acceptable values for recovery time, potential for loss of work, system downtime for the backup process to run, and financial cost.

CREATING A BACKUP STRATEGY FOR SAS DEVELOPMENT AND TESTING

Defining the authentication domains, documenting the contents of each tier in your SAS deployment, figuring out how to avoid resource contention, and documenting the entire process are not easy tasks! We know from experience. To give you an idea of what this process is like, this section discusses the backup-and-recovery process that we implemented for development and testing at SAS World Headquarters in Cary, North Carolina.

At SAS, we have a shared testing environment where we integrate as many SAS offerings as possible into a single SAS enterprise BI deployment. Such an environment offers many advantages. One is that we get to manage the environment in the same way that our customers do, enabling us to experience the challenges that IT administrators and SAS administrators have in managing our products. Not surprisingly, we discovered that one of the bigger challenges is backing up and recovering a SAS enterprise BI deployment.

A large number of different product teams work in the shared testing environment. Some of these teams also work from remote locations around the globe. In addition to the teams working from SAS World Headquarters in Cary, NC, we also have teams in Austin, Texas; Pune, India; Beijing, China; and Helsinki, Finland. These global teams are

active in this environment on a regular basis. Therefore, it is important to be able to back up the environment regularly in order to limit the amount of work that could be lost by these teams if a catastrophic failure occurred. It is equally important to minimize the length of time in which the system is unavailable to the teams. Because of the geographic diversity of our users, selecting the best time to make the system unavailable is a challenge.

In the event of a catastrophic failure, we need to recover the entire system. However, if an isolated area of the system becomes corrupted, we might need to recover only a small portion of the system. To ensure that we can recover completely from a failure, whether it is a complete failure or a simple corruption of departmental data, we created an inventory of all of the processes and data sources that are used by the system.

Following the guidelines defined in the section "Creating a Backup Strategy", we defined the scope of our backup-and-recovery process. Based on the activity on our system and the number of teams that use this configuration daily, we targeted a 24- to-72 hour recovery time.

We have a basic assumption that if a critical data source is in use, then our backup software will not perform backup processing on that resource. Because limited work can be done on the middle tier or the other server components while the metadata server is paused, we shut down all of the interconnected components simultaneously. This action ensures that the components remain synchronized. The tools that we have in place enable us to perform an orderly, synchronized shut-down and restart. The tools also work well with Solaris 10 zone functionality, which can be used to remove user access and process contention on all potentially critical data resources.

The particular strategies our team used were determined, at least in part, by the resources that were available to us. The following discussion describes specific hardware and software we are using in our environment. We mention specific products to demonstrate how we exploited the facilities of a particular system; you can extrapolate these techniques for use on your own system. This information is not intended to serve as a recommendation for any vendor's offerings.

At SAS, we use a Sun Fire E2900 that runs the Solaris 10 (Update 3) operating system. This machine is partitioned into different zones that are used by various SAS deployments. Not all of these SAS deployments have a significant level of activity on a daily basis. Some of these environments are playpen areas where groups can try things before moving them into the larger common environment. Another deployment is a debugging environment in which developers can examine problems in the larger common environment without impacting other people's work.

This machine is partitioned in this manner to manage the processes that are running on it more easily. We have deployed server tiers and middle tiers for current SAS releases in the field, as well as for SAS releases that are still under development. These tiers are distributed across the different zones to enable us to run and manage our software in a manner as similar as possible to that of our customers. An 8-terabyte storage area network (SAN) is our primary disk storage. The SAN is configured to run the Solaris ZFS file system. The SAN is shared across all of the zones on the machine. Most of the SAN is configured as a RAID5 device.

For our backup program, we use Veritas NetBackup, which is already in use for many of our strategic systems at SAS. We also take advantage of some of the base functionalities of the ZFS file system in order to minimize the system downtime required to get a backup of the system.

To verify that our backup-and-recovery strategy is working correctly, we executed a trial run of the documented recovery process onto a backup machine.

CASE STUDY: DETAILING THE BACKUP PROCESS FOR THE SAS® 9.2 TESTING ENVIRONMENT

This discussion focuses on our SAS 9.2 testing environment because it is our largest, most complex, and most active image on the machine. This makes it the most difficult to back up effectively.

The 9.2 test environment uses five of the zones mentioned previously. The five zones are for the following items:

- the metadata server
- the WebSphere application server
- the stored process servers, the SAS Workspace servers, the object spawner, the SAS/CONNECT® server, and the SAS/SHARE® server (all of these servers share one zone)
- the SAS® OLAP Server
- a reverse proxy server, which is composed of IBM Corporation's Tivoli Access Manager and WebSEAL.

With so many groups and customers in many different time zones, we had to select a time that would be the least disruptive. To further minimize the disruption, we also decided to take a consistent, synchronized copy of the system and then back up the copy rather than the system itself. We create this copy daily at 7:00pm using some facilities of

the ZFS file system that are called ZFS snapshots and ZFS clones. You can read more about these facilities in the Solaris ZFS Administration Guide that is available on the Sun Microsystems Web page. Other platforms also have similar offerings that you can leverage to minimize the downtime of your system. SAS performs full backups of these copies every weekend and runs incremental backups on the copies daily. After an incremental backup is run, our backup software then reconstructs the full images. This reconstruction is a huge advantage in minimizing the time to recovery. Without the reconstruction technology, we would have to restore the last full backup and then restore each subsequent incremental backup to get a full restore of our environment.

Fifteen minutes before the backup image is to be created, an automated notification is sent to the general user population reminding them that a backup is about to start. We then check any users or processes that are running on the system for any contention on resources that need to be backed up. An additional, more urgent notification is sent to the owners of processes that are using resources that need to be backed up. The notification informs users that they have critical resources allocated and instructs them to free those resources before the backup process takes place.

At exactly 7:00 p.m. Eastern Standard Time, the automated process to quiesce (render inactive) the SAS servers begins. Before the servers are quiesced, we run the Solaris sync command to ensure that the data on disk is in a consistent state. The servers that need to be quiesced are:

- **SAS Metadata Server**: Quiesce this server and flush the metadata to disk.
- **SAS Workspace Server:** Terminate this server to free allocated data sources.
- **SAS Stored Process Server:** Terminate this server to free allocated data sources.
- **SAS object spawner:** Terminate the spawner.
- **SAS OLAP Server:** Terminate this server to free allocated data sources.
- **SAS/CONNECT server:** Terminate this server to free allocated data sources.
- **SAS/SHARE Server:** Terminate this server to free allocated data sources.
- **Web Application Server:** Terminate this server to back up configuration information.
- **Xythos Server:** Terminate this server to free allocated data sources.
- **Database for Xythos metadata:** The database containing the metadata for our Xythos DAV server runs on a separate machine that is in a different administrative domain. We coordinate with the administrator of this machine to ensure that he creates a backup of the database content at the correct time so that it will be synchronized with the content of our Xythos DAV.

The shared test environment administration team has produced scripts that will quiesce and restart servers in an orderly fashion via Solaris Management Facility. However, these scripts can be adapted easily to any UNIX implementation and can be run through other automation mechanisms. A generic UNIX version of these scripts is available from the Enterprise Integration Management Web page at support.sas.com/rnd/emi.

One nice feature of Solaris Zones is that they can be put into a quiesced state and have only a single user (superuser) running in the zone. With only a single user running in the zone, it becomes more difficult, though not impossible, for other users to cause data contention issues. Further, the process of bringing the zone to a single user state causes the termination scripts for any processes that are managed by SMF to be shut down also. In the case of the scripts that we produced to automatically shut down and restart the SAS servers, these SAS processes are terminated in a way that prevents data loss. You can automate all of these zone management tasks.

As mentioned earlier, when you run a zone in single-user mode, it is more difficult for users and processes to cause contention among data sources. That is true only if a user or a process attempts the contention through the zone or zones that are in single-user mode. If a user or a process is running in another zone or environment that shares the disk with the zone that is shut down, contention for resources can still occur. Because of this, we have created scripts that will change the permissions on directories that contain critical data to Read-only mode. Limiting access to critical directories removes the possibility that a resource is put into an inconsistent state as the result of an update.

After quiescing the servers and restricting access to data sources, most of the critical data sources on the server and the middle tier are frozen and in a consistent state. However, if we were to run our backups directly against this image of the data, our SAS configuration might be unavailable to users for an unacceptable length of time. In practice, it has taken more than 39 hours to create a full backup of this data. To minimize this problem, we use the ZFS snapshot and ZFS clone facilities that are available in the Solaris ZFS file system.

Once the data sources are frozen, we use the **`zfs snapshot`** and **`zfs clone`** commands to clone the data in our ZFS file system. Then, we run the backup against the cloned data image.

The file system contains over a terabyte of data. Therefore, creating a full backup directly from the real file system would take days. However, using the **`zfs snapshot`** and **`zfs clone`** commands limits the unavailability of the system to about 15 minutes. We also have these clones available on disk for a quick turnaround recovery of the system, if needed.

Before the system can be restarted, we run the SAS **`cleanwork`** command to clean up any SAS work data sets that might be left in an inconsistent state when SAS processes are terminated. Then we restart the zones, the servers, and the middle-tier components in the zones that use the automated SMF processing. When these processes are available again, the general user community receives an automated "all clear" notification stating that the system is available again. The time from the initial quiescing of the servers to the announcement that users can get back on the system is less than 15 minutes.

IDENTIFYING WHAT TO BACK UP

To identify what to back up, we inventoried all of the processes and data on our system. We needed to know what data sources were used and which of these data sources need to remain synchronized. Data sources were grouped logically under a small and manageable number of directories so that the bulk of the data we needed to back up would be found in the configuration directory, the SAS installation directory, and the SAS data tree. We also cataloged all raw data sources, system directories that contain startup or system management mechanisms, Web application server directories, and other third-party software directories (such as WebDAVs, databases, the LSF Scheduler, and other miscellaneous directories).

In addition to the SAS configuration directory, the SAS installation directory, and our SAS data tree, we backed up the following directories:

- SMF-dependent items
  - SMF scripts: **`/users/prdsas/SMF`** on all zones. Note: Special permissions are required of the user ID under which these scripts.
  - SMF logs: **`/var/svc/log`**
- WebSphere ND installation and profiles—***`zone`*`:/opt/IBM/WebSphere`**
- WebSphere MQ installation—***`zone`*`:/opt/mqm`**
- All database clients
  - **`/dbi`** on all zones
  - Backup files of Xythos and Oracle content: **`/dbi/backups/files`**
- General management tools
  - **`/users/prdsas/bin`**
  - **`/users/`*`user-name`*`/bin`**

SPECIAL BACKUP CONSIDERATIONS

For the backup strategy at SAS, we had to keep the following special considerations in mind:

- **SAS Metadata Server:** As mentioned earlier, the metadata server keeps its data in memory while the server is running. To accommodate this, we first stop the SAS BI servers, which are dependent on the metadata server. Then, we flush the metadata server's data to the disk using %OMABAKUP, which pauses and flushes each metadata repository.
- **Web Application Server:** The critical piece for us is the configuration directory, although our backup enables a full restoration of the entire WebSphere directory tree.
- **SPD Server and SPD Engine servers:** Though we are not running SPD Server or SPD Engine in our environment, keep in mind that you must use the special set of utilities that are provided by the SPD Engine and the SPD Server software.
- **WebDAV servers:** Xythos servers are used by a number of different client applications in our SAS business intelligence suite. The WFSDump command that is included with Xythos is used to back up the WebDAV metadata and the content simultaneously. The WFSDump and WFSRestore commands both have a dependency on the SAS Metadata Server.
- **Stored processes**: Stored processes and their associated metadata are also synchronized to avoid issues with errant definitions of processes in the metadata.

In our process, we follow these best practices:

- The system is backed up before we apply any maintenance, before and after deploying new applications, and before and after making any significant configuration changes to the system. These backups do not have to be our usual Veritas backups.
- We take advantage of zone cloning, another feature of Solaris Zones. Clones are created for any affected zones so that if anything unexpected occurs to our configuration during the application of updates, we can recover quickly to our previous state.

One trick that aids us greatly in restoring our system is to use aliases or canonical name (CNAME) records for the host names when we configure our SAS business intelligence platform. This trick enables us to move our SAS deployment from one machine to another (that is a similar platform, of course) by simply updating our alias table on our Domain Name Server (DNS). To do this successfully, we have to use the **`-dnsMatch`** option for the SAS object spawner.

Another variation on this trick is the use of virtual IP addresses. A number of different variations of virtual IP addressing are available across the vendors on which SAS is supported. Use of virtual addressing makes the process of restoring your SAS enterprise BI deployment onto new or different hardware much easier. Consider taking advantage of these new technologies when you first deploy your SAS solution to simplify the management of your deployment in the long term.

TESTING THE RESTORATION PROCESS

To be absolutely sure we can successfully recover our system in the event of catastrophic failure, we did a trial restoration of our SAS enterprise business intelligence deployment. This trial helped us discover issues with synchronization of data sources and helped us catch some omissions in our backup plan. We have created a schedule for periodic testing of our ability to restore the system so we can be sure that our backup process continues to be correct and complete.

CONCLUSION

The ability to successfully back up and recover your SAS BI deployment does not just happen without some careful consideration and planning. A full-complement SAS deployment has many functioning components. Each of these components needs to be understood both individually and as a part of the whole in order to obtain a complete, synchronized, and usable restore image. At SAS, we try to simulate a customer environment with our shared testing environment, but the real world simply has too many possible configurations and business scenarios for us to be able to simulate all of them. No two deployments are the same, and neither are the needs of the various business units that might be using the system. It is imperative that your SAS administrators work with your IT team to establish a backup-and-recovery plan that matches the needs of your specific business and SAS deployment needs.

In this presentation, we outlined the process for creating a backup strategy:

1. Start by listing and prioritizing the various business units that use your system.
2. Focus on the needs of the highest priority business units first, detailing the parts of the system that are required by those units in order for them to be operational as soon as possible and within any service level agreements that you might have with those business units.
3. Set a budget for your backup-and-recovery efforts. Remember that you will not value those efforts as highly until your system has crashed and your users are asking when it will be back online. Schedule your backups to run on a regular timetable so that your users know what to expect.
4. Evaluate the time that will be required for recovery for the different types of failures that might occur.
5. Schedule your backup process. The time you choose to run your backups will inevitably have to be a balance of needs and priorities.
6. Have a well-documented plan in place that addresses each type of failure, and practice your plan. Make sure that your plan is adequate for your business needs and any service level agreements to which you are committed.
7. Make plans for any upstream data and processing that might back up while your system is being recovered.

8. Be sure that you understand and exploit any functionality that is available in the environment on which you are running:

    - disk mirroring
    - logical partitioning
    - virtual IP addressing
    - host-name aliasing
    - any other functionality that might lessen the time required to create backups or that help minimize the requirements for a test restoration of the system.

9. Work closely with your IT backup-and-recovery team to fully exploit the mechanisms that are available to you at your site. They know the available tools, but they need you to help them with the subtleties that apply to SAS.

In addition to providing the steps for creating a backup strategy, we have consolidated the knowledge gained from our own experience in creating a backup-and-recovery process to use at SAS. This paper documents those procedures in a repeatable process that you can use as a model for your own strategy. That process includes the following procedures:

- stopping and starting all of the various components of a SAS BI deployment
- documenting the locations of all of the content that you need to back up
- documenting the locations of where backed-up data will be stored and the procedures you need to retrieve that data
- using specific utilities and tools that are required to back up and recover the system
- scheduling when the backup process should occur and maintaining related service levels for archiving and restoration of the system.
- handling problems in backup and recovery of the system

RESOURCES

Backup and Restoration of Zone Content

Cotton, Penny. *Backup, Restore, and Disaster Recovery on a System With Zones Installed*. 2005. Santa Clara, California: Sun Microsystems Inc. Available at www.sun.com/bigadmin/features/articles/backup_zones.jsp.

Database Backup and Recovery

IBM Corporation 2008. *Informix Library*. Armonk, New York: IBM Corporation. Available at www-306.ibm.com/software/data/informix/pubs/library/.

IBM Corporation 2008. *Library [DB2*]. Armonk, New York: IBM Corporation. Available at www-306.ibm.com/software/sw-library/en_US/products/J441045L92289N69/.

MySQL AB. 2008. *MySQL 5.0 Reference Manual*. Uppsala, Sweden and Cupertino, California, USA: MySQL AB. Available at dev.mysql.com/doc/refman/5.0/en/.

Sybase Inc. 2008. *Adaptive Server Enterprise 12.5 (Core Documentation Set)*. Dublin, California: Sybase Inc. Available at manuals.sybase.com/onlinebooks/group-as/asg1250e/.

Teradata Corporation. *Teradata Information Products*. Miamisburg, OH: Teradata Corporation. Available at www.info.teradata.com/.

SAS `cleanwork` COMMAND

SAS Institute Inc. 2003. "Tools for the System Administrator: cleanwork Command" in *SAS OnlineDoc 9.1.3*. Cary, NC: SAS Institute Inc. Available at support.sas.com/onlinedoc/913/getDoc/en/hostunx.hlp/a000350996.htm.

SAS Content Backup for the SAS®9 Enterprise Intelligence Platform

SAS Institute Inc. 2006. *Backing Up SAS Content in Your SAS®9 Enterprise Intelligence Platform: Considerations for Creating Backups of Your SAS Content.* Cary, NC: SAS Institute Inc. Available at support.sas.com/resources/papers/contentbackup.pdf.

SAS Scalable Performance Data Server Utilities

SAS Institute Inc. 2007. *SAS® Scalable Performance Data Server® 4.43: Administrator's Guide*. Cary, NC: SAS Institute Inc. Available at support.sas.com/documentation/onlinedoc/spds/admin443.pdf.

Solaris ZFS File System

Sun Microsystems Inc. 2008. *Solaris ZFS Administration Guide*. 1994-2008. Santa Clara, California: Sun Microsystems Inc. Available at docs.sun.com/app/docs/doc/819-5461/819-5461?a=browse.

SunONE LDAP Server Back and Restoration

Sun Microsystems Inc. 2003. "Backing Up Data" in *Chapter 4: Populating Directory Contents.* Santa Clara, California: Sun Microsystems Inc. Available at docs.sun.com/source/816-6698-10/populate.html#14962

Sun Microsystems Inc. 2003, "Restoring Data from Backups" in *Chapter 4: Populating Directory Contents*. Santa Clara, California: Sun Microsystems Inc. Available at docs.sun.com/source/816-6698-10/populate.html#15115.

WebLogic Failed Server Recovery

BEA Systems, Inc. 2008. *Recovering Failed Servers*. San Jose, California: BEA Systems. Available at e-docs.bea.com/wls/docs81/adminguide/failures.html.

WebSphere Backup and Recovery

IBM Corporation 2007. *Backing up and recovering the application serving environment*. Armonk, New York: IBM Corporation. Available at publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.base.doc/info/aes/ae/tadm_snrmain.html.

IBM Corporation 2007. *backupConfig command*. Armonk, New York: IBM Corporation. Available at publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.wsfep.multiplatform.doc/info/ae/ae/rxml_backupconfig.html

CONTACT INFORMATION

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

Arthur Hunt
SAS Institute Inc.
SAS Circle Drive
Cary, NC 27513
Arthur.Hunt@sas.com

Tanya Kalich
SAS Institute Inc.
SAS Circle Drive
Cary, NC 27513
Tanya.Kalich@sas.com

Billy Dickerson
SAS Institute Inc.
SAS Circle Drive
Cary, NC 27513
Billy.Dickerson@sas.com

Chuck Antle
SAS Institute Inc
SAS Circle Drive
Cary, NC 27513
Chuck.Antle@sas.com