Best Practices for Upgrading from SAS® 9.1.3 to SAS® 9.4

David Stern, SAS Institute Inc., Cary, NC

ABSTRACT

We regularly speak with organizations running established SAS® 9.1.3 systems that have not yet upgraded to a later version of SAS®. Often this is because their current SAS 9.1.3 environment is working fine, and no compelling event to upgrade has materialized. Now that SAS 9.1.3 has moved to a lower level of support and some very exciting technologies (Hadoop, cloud, ever-better scalability) are more accessible than ever using SAS® 9.4, the case for migrating from SAS 9.1.3 is strong. Upgrading a large SAS ecosystem with multiple environments, an active development stream, and a busy production environment can seem daunting. This paper aims to demystify the process, suggesting outline migration approaches for a variety of the most common scenarios in SAS 9.1.3 to SAS 9.4 upgrades, and a scalable template project plan that has been proven at a range of organizations.

INTRODUCTION

Now is a great time to upgrade from SAS 9.1.3 to SAS 9.4. If you own, manage or use a SAS 9.1.3 deployment, SAS encourages you to upgrade to the current release. The benefits to be gained from enhanced capabilities in SAS which become available to you when you upgrade from SAS 9.1.3 are so significant that the effort required is worthwhile. See "Why should you upgrade from SAS 9.1.3?" on the next page.

The process of upgrading a SAS deployment has acquired a reputation for being hard. While upgrading is complex, SAS has been continually strengthening and streamlining the process. This paper highlights the benefits of moving to the current release, and outlines the key steps that you should include in your overall upgrade approach, to help you plan and implement a smooth transition. In doing so, it provides examples and guidelines based on some of the most common upgrade scenarios that SAS Consulting has supported.

The core message I want to get across is that upgrading from SAS 9.1.3 to SAS 9.4 can be an intricate process, and can take some time, especially if your existing SAS 9.1.3 deployment has been heavily customized, but you should not be apprehensive about it. The benefits are so great, and the challenges so manageable, that I hope to convince you go for it and upgrade now. Using the ideas outlined in this paper, you can achieve a very satisfying outcome from your upgrade.

Who should read this paper

This paper is for owners of SAS 9.1.3 deployments, and the consultants and project managers who work for them. Because it is mostly non-technical, you will get value from this paper even if you have only a passing familiarity with SAS. You might understand more of the detail if you have seen or used SAS migration and upgrade tools. But even if you have never seen them, you should have no difficulty in understanding this paper.

Scope of this paper

In this paper you will learn the best approaches to migrating and upgrading SAS 9.1.3 deployments that have *a SAS® Metadata Server* to SAS 9.4.

The process of upgrading SAS deployments *without* a metadata server (i.e., those whose main SAS product is SAS® Foundation, SAS® Enterprise Guide or similar) is typically much simpler than the one we will explore in this paper.

The great majority of this paper also applies to SAS deployments which include specific industry solutions. It is therefore also relevant if there are one or more SAS solutions in your deployment. However, many solutions include additional components that must be considered when planning and implementing your upgrade. If you have industry-specific SAS 9.1.3 solutions in your deployment, contact your SAS Consulting team for support, and review the available "What's New" documentation for your

solution. If you have multiple SAS solutions, you may need to consider additional dependencies *between those solutions* during your upgrade planning and execution. Again, SAS Consulting can help you with this.

There should also be plenty in this paper to interest you, even if your SAS upgrade will not be from SAS 9.1.3 to SAS 9.4. Parts of the approach described in this paper are applicable to any SAS upgrade.

The executive summary, which follows, highlights the main recommendations in this paper, and contains the key points to remember if you do not have time to read it all.

EXECUTIVE SUMMARY

SAS 9.4 is a tremendous advance beyond SAS 9.1.3. Whether you are an owner, a user, or an administrator of a SAS 9.1.3 deployment, you will find something great in the latest release of SAS. The next section lists some of the best new features.

To move from SAS 9.1.3 to SAS 9.4, you must upgrade in two stages via an interim SAS 9.3 environment, because it is not possible to import application content from SAS 9.1.3 directly into SAS 9.4 using SAS migration tools. However, while it is necessary to migrate and upgrade metadata (plus some closely-related content) in two stages, much of your other application content can be migrated and upgraded directly from your source SAS 9.1.3 deployment to SAS 9.4.

This paper presents a scalable template project plan for performing such a migration and upgrade. The next several pages explore the recommended upgrade approach in detail.

Next, this paper examines an approach to planning and executing a set of migrations from SAS 9.1.3 to SAS 9.4 for an *ecosystem* of related SAS deployments – e.g., development, test, and production. High-level approaches for this are presented, with consideration for the *route to live* for new development work and bug fixes at all stages in the migration.

Finally, this paper explores techniques that help you avoid some of the difficulties of upgrading very busy development and production environments.

WHY SHOULD YOU UPGRADE FROM SAS 9.1.3?

You are entitled to have the most recent versions of each SAS product for which you are currently licensed, at no extra cost. Upgrading like-for-like costs you nothing in SAS software fees.

Despite an increasingly competitive market for analytics, big data and reporting, SAS 9.4 is the clear leader in delivering big data analytics. And it keeps getting better. See the "What's New" page, available at http://support.sas.com/documentation/whatsnew/, the "Maintenance Releases and Product Upgrades" page at http://support.sas.com/software/updates/, and the regularly-updated "Guide To Software Updates" documentation, available at http://support.sas.com/documentation/cdl/en/whatsdiff/66129/HTML/default/viewer.htm#titlepage.htm.

If you are the owner of a SAS deployment, you may want to upgrade because SAS 9.4 enables you to deploy SAS environments with much greater scalability and higher availability than are possible using SAS 9.1.3. You can also host SAS 9.4 on your own hardware using physical or virtual machines, or in any of the several types of cloud. SAS 9.4 comes with a self-contained middle tier web application server, removing the dependency on a third-party web application server.

Administrators will like that SAS 9.4 comes with enterprise class administration and monitoring, along with a large set of standard audit reports in SAS® Environment Manager. SAS 9.4 also provides additional security options, for example the LOCKDOWN system option which can limit access to files and specific SAS features in batch or server processing.

If you use SAS for analytics, reporting or data management, there are dozens of enhancements, small and large which make an upgrade compelling. One is that SAS 9.4 works well with several big data storage and processing systems, including Hadoop. These systems provide access to state-of-the-art technology for distributing both processing and storage over a large array of low-cost servers. For certain types of tasks, Hadoop delivers truly astonishing improvements in performance over other traditional

processing techniques. Many other "big data" providers are also supported. SAS 9.4 is great for deriving analytics insight into data stored in its own formats, or in other vendor's databases, and SAS has tremendously enhanced its ability to execute SAS analytics inside those databases, right next to the data.

SAS 9.1.3 was released in August 2004, and was superseded by SAS 9.2 in March 2008. SAS 9.1.3 deployments which are still in use today have been in use for a very long time – something between six and nine years – without being upgraded!

However, SAS 9.1.3 was moved to Level C support on September 30, 2013. By the time this paper will be presented at SAS Global Forum 2015, SAS 9.1.3 will have been at this reduced level of support for 19 months. The significance of this is explained on the SAS Support web page "SAS TECHNICAL SUPPORT SERVICES AND POLICIES", available at http://support.sas.com/techsup/support.html. If your business relies on SAS 9.1.3, you should upgrade to a version which has a higher support level. To maximize the benefits, you should move to the latest version of SAS, and I would encourage you to come along with SAS as we release new versions of our software.

YOU MUST UPGRADE FROM SAS 9.1.3 TO SAS 9.4 IN TWO STAGES

When upgrading your application content from SAS 9.1.3 to 9.4, the key technical consideration is that if your SAS 9.1.3 deployment contains metadata, you **have to upgrade in two stages**.

If you have a choice of SAS® 9.2 or SAS 9.3 as your intermediate deployment, you should choose SAS 9.3 because SAS 9.2 transitioned to level B support on September 30, 2013, as shown at http://support.sas.com/techsup/support.html#non-current.

YOU CAN UPGRADE DEPLOYMENTS WITHOUT METADATA FROM SAS 9.1.3 DIRECTLY TO SAS 9.4

This paper is mostly concerned with deployments that have metadata. But SAS 9.1.3 deployments exist that either:

- do not have metadata, or

- have very little metadata, or the metadata is of modest business value, or can be easily re-created

In the first case, there is no metadata to migrate. In the second, it is reasonable for you to opt *not to migrate metadata* to SAS 9.4 and instead to re-create it manually. In either case, this means you do not need to upgrade your application content via SAS 9.3, and instead you should migrate your content to SAS 9.4 directly.

WHY YOU MUST UPGRADE DEPLOYMENTS WITH METADATA FROM SAS 9.1.3 TO SAS 9.4 IN TWO STAGES

Because your metadata cannot be upgraded from SAS 9.1.3 to SAS 9.4 directly, you will have to upgrade it via an interim deployment, which as discussed above should be at SAS 9.3. From that interim deployment you can then export the metadata again, and this time import it into your new SAS 9.4 deployment.

This is because the migration tools included in SAS 9.4 are designed to import and upgrade metadata (plus some closely-related content, e.g. deployed and external code files, WebDAV assets for SAS Web Report Studio Reports etc.) from either of the two previous major versions of SAS 9 (that is, from SAS 9.3 and SAS® 9.2), but not from earlier versions. SAS 9.1.3 is three major versions earlier than SAS 9.4. See the official SAS statement at http://support.sas.com/rnd/migration/utility/upgrade.html#from913.

Figure 1 represents the path that metadata, and all other application content will take between SAS 9.1.3, SAS 9.3 and SAS 9.4 during the upgrade process.

| 9.1.3 SP4 | 9.3 | 9.4 |
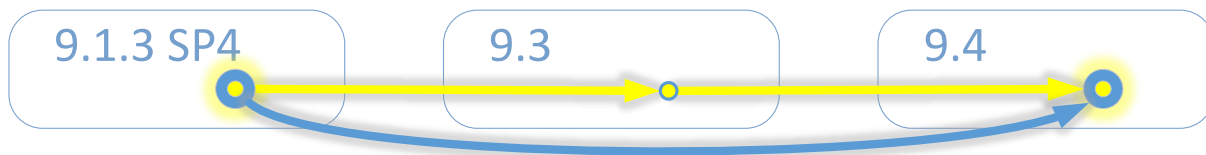
○ Metadata (plus some closely-related content) in SAS Package Files (.spk)

◉ All other application content, such as data, code stored outside the configuration directory, SAS® Enterprise Guide Project files (.egp), text files, and so on

Figure 1. Path taken by metadata and other application content in migration and upgrade from SAS 9.1.3, via SAS 9.3 to SAS 9.4.

There are three ways to move your application content between environments. Let's look at each one.

The SAS® Migration Utility and SAS® Deployment Wizard

The amazing SAS Migration Utility does two jobs. First, it analyses the configuration of software components in your current deployment, and reports any issues that you need to resolve before creating a migration package, or which you might need to resolve in your target deployment after you have migrated your metadata and configuration content to it. Examples of these issues include directory paths for libraries that will need to be updated, and passwords for outbound login credentials will need to be re-entered, but there are many other issues that the SAS Migration Utility can help you identify and resolve, which you might otherwise not recognize by yourself – its analysis mode is a tremendous tool.

Second, it can also package up all the metadata, and the content of the configuration directory into a file archive called a migration package. If there is data in your Config/Lev*n*/Data directory, it will also package that, but there are few cases where storing data of any size or significance there is a good idea - it is far better to keep large tables somewhere else, or move them somewhere else temporarily while you create the migration package, in order to keep the size of the package small.

The SAS Deployment Wizard completes this part of the SAS migration tools. It can import a migration package into a new SAS deployment as it is deployed. It upgrades the content of that migration package to the newer version of SAS as it does so.

However, there are some constraints which apply to this method of upgrading SAS application content. The SAS Deployment Wizard can only accept a migration package created on a deployment with the same topology, and with the same operating system family as the new SAS instance being deployed. See "High-Level SAS Migration Requirements" in the *SAS(R) 9.3 Intelligence Platform: Migration Guide*, http://support.sas.com/documentation/cdl/en/bimig/62613/HTML/default/viewer.htm#n01019intelplatform00migrate.htm. Also the SAS Migration Utility and the SAS Deployment Wizard offer no means of selecting which content is packaged, or which content in the migration package is deployed – they package and deploy everything. There is no time other than during initial deployment that you can use a migration package.

Promotion with the Export SAS Package Feature in SAS® Data Integration Studio or SAS® Management Console

A second way to move your application-specific metadata is promotion. You do this using the Export SAS Package wizard in SAS® Management Console or SAS® Data Integration Studio to select metadata objects and export them to a SAS Package (.spk) file. You can then import this SAS Package file using the Import SAS Package wizard in SAS Data Integration Studio or SAS Management Console in the target deployment. It offers you the option to select all, or only a subset of the objects in the package to be imported.

When importing a SAS Package File, the Import SAS Package wizard asks you to confirm or update metadata and other associations for the objects that are being imported.

Re-create metadata manually in SAS 9.4

As a way to move metadata between deployments this might seem a bit of a tenuous recommendation, but bear with me. It is clearly not practical to manually re-create large amounts of metadata in SAS 9.4 by hand. The tools described above are far better choices if you have a lot of metadata.

But, if you *do not* have large amounts of metadata to migrate, do not dismiss the simplicity of just creating, altering, or re-creating that metadata in your new SAS deployment.

Choose a Migration Approach to Suit Your Circumstances

For the two-stage migration needed to take metadata from SAS 9.1.3 to SAS 9.4, which technique is best for each stage and why?

For the most common scenario, in which you have an existing SAS 9.1.3 deployment with metadata, running on Windows or UNIX hosts, and you begin your migration project having no intermediate SAS 9.3 environment, and no host machines for your SAS 9.4 environment, the best practice is to:

1. Create an interim SAS 9.3 deployment on temporary hardware (or a virtual machine). In most cases, it's best to use a *single machine* to host all the tiers in your interim SAS 9.3 deployment, because a single machine has a lower cost than multiple machines, and it takes less effort to deploy SAS on a single machine. You will need this machine only for the duration of the migration, and it does not need to be as powerful as the 'real' source or target hardware.

2. Create a carefully designed and fully architected SAS 9.4 deployment to act as your final target for the migrated application content. You should use whatever number of host machines, and whatever server topology your SAS architect recommends in this deployment in order to ensure that the deployment will best meet the needs of your business.

3. Migrate and upgrade all the application content in your source SAS 9.1.3 deployment to the temporary SAS 9.3 deployment using promotion.

   o Note: If your source SAS 9.1.3 deployment is on a single machine, and the other constraints for using the SAS Migration Utility for this stage are met, you could use the SAS Migration Utility for the first stage of your migration and upgrade to SAS 9.3. The SAS Migration Utility approach is very effective and is easier to use than the promotion approach recommended in this paper. However, the SAS Migration Utility it is not selective in what it upgrades, and you cannot use it to upgrade from a multi-machine deployments of SAS 9.1.3 to a single-machine deployment of SAS 9.3. Much as I like SAS Migration Utility, the second of those two reasons is why using it for this stage of the upgrade is not the primary recommendation in this paper.

4. Migrate and upgrade selected parts (or all) of your application content from the interim SAS 9.3 deployment to your final SAS 9.4 deployment using promotion.

   o Note: If your SAS architect recommends a single machine to host the server component of your SAS 9.4 deployment, and the other constraints are also met so that using the SAS Migration Utility is feasible, you might opt to use the SAS Migration Utility for the second stage of the migration and upgrade, from SAS 9.3 to SAS 9.4. I do not recommend it as the best practice in this paper, because it limits your architectural options in SAS 9.4 too much. But it is a reasonable choice if you can meet your business needs with a single host machine for SAS 9.4, running the same OS family as your interim SAS 9.3 deployment.

Not all SAS 9.1.3 upgrades will fit this typical scenario. However, you will most likely find the outline migration process described below useful, even if your particular upgrade's circumstances mean that your best choice is to deviate from it to some degree.

SCALEABLE TEMPLATE PROJECT PLAN

Figure 2 shows the approximate flow of tasks in the approach outlined in this paper, for an ecosystem migration pattern like that shown in Figure 7. In that pattern, only one upgrade of the application content is performed from the current development SAS 9.1.3 deployment, via SAS 9.3 for the metadata (and some closely-related content), to the new SAS 9.4 development deployment. Other application content is

copied directly from SAS 9.1.3 to SAS 9.4. From there, the upgraded application content is promoted to the other SAS 9.4 deployments.

The tasks in each darker rectangle in the plan correspond closely with the tasks set out in this paper, though some are necessarily omitted to make the diagram clearer.

Note that Figure 2 under-represents the amount of effort that you should spend on testing, fixing issues, and resynchronizing multiple times. These activities, and in particular testing, should be the largest by effort spent, and scaling should take up something like 40-50% of the total effort.
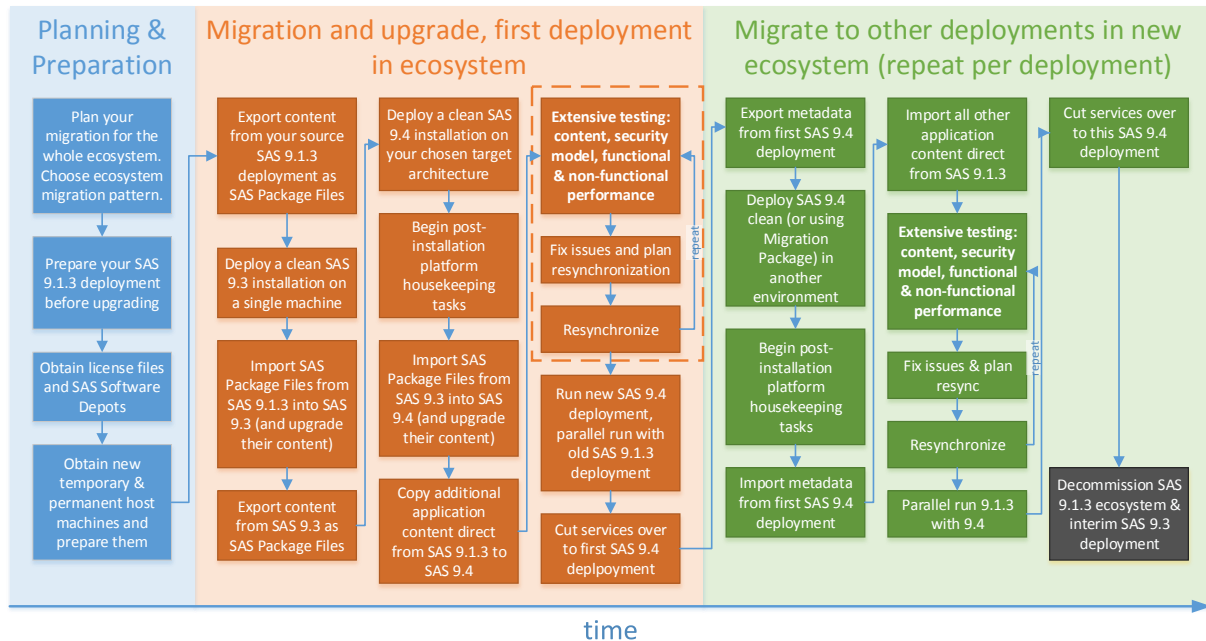
Figure 2. High level project plan, showing broad division of ecosystem migration and upgrade into 3 high-level phases: 1. planning & preparation, 2. migration and upgrade of the first deployment, and 3. migration to the other deployments in the ecosystem.

PREPARE YOUR SAS 9.1.3 DEPLOYMENT BEFORE UPGRADING

Your SAS 9.1.3 deployment must have certain updates applied to it before you will be able to successfully migrate content from it. Meeting this requirement can be one of the more demanding tasks in upgrading a SAS 9.1.3 deployment, so allow time for it in your plan, and ask SAS Customer Support for assistance if you need it. You should also read the indicated sections of the two documents referenced below.

Read the brief notes below the headings "Ensuring the Correct Software Version When Exporting from SAS 9.1.3" and "Moving SAS 9.1.3 Tables and Libraries to Folders" in the section "Special Considerations for Promoting Metadata from SAS 9.1.3 to SAS 9.2" in the SAS® Intelligence Platform: 9.1.3 to 9.2 Migration Guide, available at http://support.sas.com/documentation/cdl/en/bimig/61696/HTML/default/viewer.htm#mig023.htm. Though written for upgrades to SAS 9.2, everything that section describes remains relevant if you are upgrading to SAS 9.3.

If you will use the SAS Migration Utility for your first step, you should also carefully read "Migrating from SAS 9.1.3" in the SAS Migration Utility documentation, available at http://support.sas.com/rnd/migration/utility/upgrade93.html#from913.

Here are the steps I have most often had to take to prepare a SAS 9.1.3 deployment for upgrade:

- The SAS 9.1.3 deployment must have SAS 9.1.3 (9.1 TS1M3) and SAS 9.1.3 Service Pack 4 installed.

- The SAS 9.1.3 instance of SAS Management Console must have SAS® Foundation Services 1.4 installed, which includes BI Manager.

- Hotfix 913SMC02 for SAS Management Console must be installed, if it is not installed already. Installing this hotfix includes manual steps that are described in the hotfix instructions.

- Once BI Manager and hotfix 913SMC02 are both installed, right-click the BI Manager node in SAS Management Console, and select Collect Ungrouped Library Metadata, which ensures that all Tables and Libraries are in folders, which is necessary before they are exported.

- If you have SAS® Data Integration Studio 3.4 (N.B. that version specifically – this does not apply to other versions) at SAS 9.1.3, you must also apply hotfix E9BC59. Applying this hotfix **requires you to restart your Metadata Server**, and requires further manual steps that are described in the hotfix instructions.

- If your SAS 9.1.3 deployment includes a Xythos database for a WebDAV, ensure that it is at version 4.0.48.

However, those steps might not be *all* you need to do. Carefully review all the notes in the two sections of the documents that are referenced in this section before proceeding. They contain preparatory steps that are absolutely required for *some* deployments, and which I have not mentioned above.

RUN THE SAS MIGRATION UTILITY IN ANALYZE MODE

Rather than recommending you use the SAS® Migration Utility to create a migration package for either of its two stages in most situations, this paper recommends you use promotion for both stages. However, even if you use promotion to move your metadata (and closely-related content) from SAS 9.1.3 to SAS 9.3, it is still a good idea to run the SAS Migration Utility *in analyze mode* on your source SAS 9.1.3 deployment. This is because it is easy to use, quick, and because it can detect and report a number of issues which are relevant even when you upgrade using promotion. It is one of the best diagnostic tools I have ever seen. Use it to examine the installed SAS products and platform configuration of your source SAS 9.1.3 deployment, and pay close attention to the analysis report that it generates. It runs checks that have been added to a knowledge base of known issues over an extended period of widespread use in the field.

Use the version of the SAS Migration Utility for SAS 9.1.3 on your operating system. The Windows version is called smu.exe, the UNIX version is called smu.sh, and the z/OS version is called smu.zos. If you do not yet have your SAS 9.3 software depot, you can download a copy of the SAS Migration Utility from http://support.sas.com/downloads/package.htm?pid=492. However, if you have your SAS 9.3 Software Depot, you should use the version of the SAS Migration Utility in the utilities folder of the depot instead. Examine the AnalysisReport (FullReport.html) that it creates. If you will use promotion for both stages of the migration and upgrade, you may exercise some judgment in deciding whether or not to act on any error messages (indicated by the symbol ✗) in the AnalysisReport. If you do intend to use the SAS Migration Utility for the first stage of the upgrade and migration, you must act on those error messages. Fixing them is normally required before a migration package can be created – nearly all of those I have seen personally 'in the field' were straightforward to fix.

Similarly, pay close attention to any warning or advisory messages in the report (indicated by the symbol 🟡). These might suggest that you perform further investigation on the source deployment, or indicate that remedial action will be required in the target deployment if you decide to use the SAS Migration Utility for the first stage of the upgrade and migration. Similar remedial action may be required if you use promotion for this stage. However, the prompts to choose new values for references in items being imported by the Import SAS Package wizard might enable you to address many of the warnings in the AnalysisReport as you import them into SAS 9.3. SAS Customer Support, or the SAS Professional Services Division can assist you with this!

CLEAN UP ANY UNUSED APPLICATION CONTENT IF YOU CAN DO SO SAFELY

If you can identify application content in your metadata, or on your file system, that you are certain is unused and no longer required, delete it before upgrading. This reduces the amount of unnecessary migration and especially testing effort expended on the unused content.

However, if you are in doubt as to whether an object or file is used, as has often been the case in my experience, I suggest that it might require less effort and be safer for the integrity of your application content to migrate everything, and test it on the target deployment, than to pause to investigate whether it is used. This is a judgment call, guided by your organization's preferences for speed versus rigor.

INVENTORY YOUR APPLICATION CONTENT AND CHECK FOR DUPLICATE TABLES

Next, inventory all the application content that you need to move from your existing SAS 9.1.3 deployments to your SAS 9.4 deployments, in consultation with your deployment's stakeholders.

In this paper, the term **application content** means any type of artifact or change made in your existing environment that was not created when SAS was deployed. It includes custom metadata, custom application artefacts, code, reports, data, and configuration changes created specifically for an organization. While *data* is part of your application content, you might migrate most of the data and some platform configuration settings to a given target environment from a *different source environment* than the rest of the application content. We'll come back to this later, in the section "Source of data and platform configuration settings for upgrades and migrations."

You can get that inventory of application content manually, by visual inspection, but this takes a long time if you have a lot of content, and it is error prone. One way to get a count of the metadata objects in your application content is to run the SAS Management Console's Export SAS Package wizard. In SAS Management Console, logged in as an unrestricted user, export all the metadata under the root folder as a SAS Package File. As a by-product, the Export SAS Package wizard creates an export log that includes an INFO message that lists the number of exported metadata objects of each type:

```
INFO  - Exporting the following objects: 41 Folders, 989 Tables, 7
Libraries, 130 Jobs, 7 External Files, 81 Deployed Jobs, 23 Generated
Transforms
INFO  - ******** Exporting Metadata ********
INFO  - ******** Exporting Content ********
```

From the overall list of metadata objects found, agree with your stakeholders on which objects you will migrate and upgrade.

Also, check the export log for messages regarding duplicate tables folders:

```
Duplicate tables in \shared data...
```

Clean up any duplicate table definitions in the metadata before upgrading, because they can potentially cause errors during upgrade.

Inventory the file paths that are used by each library and each external file in your source SAS 9.1.3 deployment's metadata. Make a list of the new library and file paths on the SAS 9.4 environments (if you know what they will be). You will need these library file paths when you import library metadata into SAS 9.4 later. Similarly, make a note of the database management system (DBMS) server definitions and database connection information on both your source SAS 9.1.3 environment and your target SAS 9.4 environment. You will need to update references to the DBMS and to the library and file paths for the new environment later.

Customizations to your current SAS 9.1.3 deployment's configuration files, start-up scripts, format and macro libraries and autoexec.sas files and so on are an intrinsic part of your application content. These things might be less obviously important to someone casually inspecting the environment, but the administrator who made the changes will appreciate their importance. Take time to review all the SAS configuration and autoexec files (and so on) in SAS 9.1.3 to look for things that will, or might need to be replicated in the new SAS 9.4 environment.

If you are not sure whether a configuration setting is custom to your organization or part of the SAS standard configuration, I suggest you include it at this stage, rather than spend valuable time trying to figure out whether it is standard or custom. The cost of noting a configuration setting, which you later learn is standard (in SAS 9.4), is likely to be much smaller than the cost of researching what is standard, and thus separating that configuration setting from your customizations. It might take you a couple of attempts to get the overall upgrade process perfected, so it is fastest to assume that something is nonstandard for now if you are not sure, and to try to figure out later whether it actually is or not, in the SAS 9.4 ecosystem.

Be detailed and precise about the content that you and your stakeholders want to have in your target SAS 9.4 deployment, and about each item's name and location. This list is the set of application content items that you will later need to confirm are present in the target SAS 9.4 deployment when you have finished, and should be one input into the test plan for your application content in SAS 9.4.

COMPLETE A THOROUGH ARCHITECTURE AND SIZING DESIGN FOR YOUR NEW DEPLOYMENTS

Complete the architecture designs for both your interim SAS 9.3 deployment and for each of your final target SAS 9.4 deployments. Depending on your organization's IT standards, you might be able to follow a less rigorous process for the interim SAS 9.3 deployment's architecture than you will follow for the target SAS 9.4 deployments. Irrespective of the level of rigor you apply, designing a single-machine deployment for SAS 9.3 is simpler than designing a multi-machine one. I have had success in several upgrade projects using an off-site location for the interim deployment, as a practical way to avoid the rigor that is necessary when hosting even a temporary SAS deployment in a production data center.

However, complete a thorough architecture design for your final SAS 9.4 deployments. If you are upgrading multiple deployments (such as development, test, and production) at the same time, design the architecture for all these deployments together, to ensure that they are consistent with each other, and that they can handle the data and processing load they will be required to support for the foreseeable future.

The optimal architecture design for your SAS products and solutions at SAS 9.4 will differ from what worked at SAS 9.1.3. SAS 9.4 is different from SAS 9.1.3 in many ways. You might have learned of limitations in the architecture of your existing SAS 9.1.3 deployments, now that you have had them for several years. The hardware market has evolved considerably since SAS 9.1.3 was deployed, your data has probably evolved, and your use of it has likely evolved too. Consult with SAS or partner experts if you need to: they have a great deal of experience in SAS architecture that few other companies have the opportunity to gain.

With your organization's stakeholders, agree and get approval to spend the budget required for your temporary and new hardware, and for the migration and upgrade effort.

REQUEST LICENSE FILES FOR SAS 9.3 AND SAS 9.4

To install the two new versions of SAS (SAS 9.3 and SAS 9.4), you will need two new license files and software depots. Discuss your need for two short-term SAS software license files with your SAS account representative, and explain that you would like to run SAS 9.1.3, SAS 9.3, and SAS 9.4 all at the same time, for a short period, in order to upgrade and migrate to SAS 9.4. I emphasize that **this document does not form part of any agreement between you and SAS Institute Inc. ("SAS") or any SAS companies or affiliates**. If your account representative agrees to provide you with temporary license files, the following may be typical of what you might expect.

You will only need a SAS 9.3 license file for a short period, as the interim SAS 9.3 deployment should not need to be used for very long. Even if you purchase new SAS products for use at SAS 9.4, you should ask for a temporary license file for your SAS 9.3 deployment with only the products that are equivalent to those you have at SAS 9.1.3. The intention is to make your SAS 9.3 interim deployment as much like the source SAS 9.1.3 deployment as possible, in terms of OS and installed products, but with all server tiers of the SAS Intelligence Platform deployed on a single host machine regardless of how many server hosts there are in the source SAS 9.1.3 deployment.

Your new SAS 9.4 license file may also initially have a short-term expiration date. On completion of your upgrade, when you have switched over live business services to the new version of SAS, you will cease using the older versions of SAS. When this happens, you should ask for a new SAS 9.4 license file which is valid until a date defined by your SAS license agreement. These short-term license files would normally be granted to you on condition that you use them for migration, and not to run more SAS deployments for an extended period for real production use. SAS might need to charge you for the additional licenses if your migration and upgrade takes too long, lasting beyond the period you agree with your SAS account representative. So please keep in contact with your SAS account representative throughout this process, and let them know about your progress and about any delays.

PREPARING THE HOSTS FOR YOUR INTERIM SAS 9.3 DEPLOYMENT

Following the approach in this paper, you need to prepare a physical or virtual host for the interim SAS 9.3 deployment.

Work with your architect, IT department, and SAS to choose a specification for the host in this environment. It does not need to be a powerful machine. If you intend promoting your content using SAS Package Files, you need only ensure the host meets the requirements for running your products at SAS 9.3. However, if you will be using a migration package created by the SAS Migration Utility, you must ensure that the interim host environment meets all of the constraints for an upgrade performed this way. See "High-Level SAS Migration Requirements" in the *SAS(R) 9.3 Intelligence Platform: Migration Guide*, http://support.sas.com/documentation/cdl/en/bimig/62613/HTML/default/viewer.htm#n01019intelplatform00migrate.htm.

Note that SAS installation, especially of multi-machine deployments, is a skilled task. Full documentation explaining how to install SAS is on our public SAS Support website, but you might still prefer to have your installations done by, or in consultation with, an expert SAS installation consultant. Please seek assistance from SAS Professional Services if you need it.

PREPARING THE HOSTS FOR YOUR FINAL SAS 9.4 DEPLOYMENT

You also need to prepare the hosts for the final environment in which SAS 9.4 will be deployed. Again, work with your architect and IT department to determine the specifications and other requirements for the hosts, storage, networking, etc. in this environment. Be thorough about completing an architecture design for all relevant aspects of the target SAS 9.4 environment to ensure that it will be able to function adequately and will meet the customer's expectations for installed products, performance, I/O throughput, storage capacity, and so on.

If you follow the upgrade process below, you will use promotion to move your application content from SAS 9.3 to SAS 9.4. This means you do not have to keep the same deployment topology as your source SAS 9.1.3 or interim SAS 9.3 deployment for SAS 9.4. But if you instead choose to use the SAS Migration Utility for the second stage of the migration, you must keep the same topology and operating system family for SAS 9.4 as you used for your interim SAS 9.3. For details, see the section "High-Level SAS Migration Requirements" in the introduction of the *SAS(R) 9.4 Intelligence Platform: Migration Guide*, http://support.sas.com/documentation/cdl/en/bimig/63853/HTML/default/viewer.htm#n01019intelplatform00migrate.htm.

DOWNLOAD SAS SOFTWARE DEPOTS FOR SAS 9.3 AND SAS 9.4

When you request a newer release of SAS software, your SAS account representative will review that everything is in order, and will ensure that the requested SAS software release will work in your proposed environment. If all is well, they will start the software order process. When the software is ready to be downloaded, the nominated SAS Installation Representative in your organization should receive a SAS® ESD Software Order E-mail. Follow the instructions in that email to download first the SAS® Download Manager, and then each of the SAS Software Orders. Put each order in a separate depot, each of which might be in the range of 20GB to 70GB in size. The emails will indicate what size you should expect each to be.

DETAILED STEPS DESCRIBING HOW TO UPGRADE THE APPLICATION CONTENT IN A SINGLE SAS 9.1.3 DEPLOYMENT TO SAS 9.4

To upgrade a single typical SAS 9.1.3 Intelligence Platform deployment to SAS 9.4, using promotion for both stages, you should follow the steps below. You might find that some steps can be performed in a slightly different order, or may overlap with each other and run in parallel in some cases, rather than being strictly sequential, but the order of the following 22 steps is a reasonable guide.

STAGE 1: SAS 9.1.3 TO SAS 9.3

1. Inside your SAS 9.3 software depot, **find the variant of the SAS Migration Utility** intended to be run against SAS 9.1.3 SP4. You should find it in the *utilities/smu* folder (just *smu*, not *smu92* or *smu93*). If you previously ran a copy of the SAS Migration Utility from the SAS Support website, discard that copy and use only the copy in the SAS 9.3 depot.

2. Before running the SAS Migration Utility in analyze mode, **carefully read the instructions** in the SAS Migration Utility Reference section in the "*SAS(R) 9.3 Intelligence Platform: Migration Guide.*"

    Below is an **example command to run the SAS Migration Utility** in analyze mode. In this example, the SAS Migration Utility has been copied to a *migration* directory under the home directory of a user called '*sas*' - i.e., to */home/sas/migration* - on a UNIX machine, might look something like the following. It would look quite similar on a Windows host, though you don't need the `--` on Windows; the command name changes to `smu.bat` and the file paths would obviously need to be changed to Windows-style paths):

    > */home/sas/migration/* `smu.sh -- -analyze -localhost` *thishostname.company.com* `-`
    > `sasconfigdir` */sas/config/Lev1* `-sasproductdir` */sas/software/913SP4* `-metadatahost`
    > *metadatahostname.company.com* `-metadataport` *8561* `-user` *sasadm* `-password`
    > *{sas001}XYZABCDEFGH=* `-outputdir` */home/sas/migration/smu_output* `-tier` *metadata_tier* `-`
    > `replace`

3. If you have multiple machines in your SAS 9.1.3 deployment, **run the SAS Migration Utility first on the Metadata Server,** and **then on each of the other machines**, in the sequence recommended in the documentation. Pay close attention to the content of each AnalysisReport it creates. If any of them contain any errors, you should NOT proceed, and should instead correct those errors and then run the SAS Migration Utility again. When it has run on each tier in your source SAS 9.1.3 deployment, with no errors in the log file, and no errors in the AnalysisReport, you are ready to promote your metadata using SAS Management Console.

4. Open SAS 9.1.3 Management Console, and log in to the SAS 9.1.3 Metadata Server as an unrestricted user. From the Folders tab, export *all the metadata under the root folder* to a SAS Package File in a directory accessible to your desktop computer. Inspect the export log, and pay particular attention to any warnings or errors in that log. If you find warnings or errors, attempt to remediate them and try this step again.

5. An experienced installation consultant should then **run the SAS 9.3 Deployment Wizard** on the machine which will act as the host for your SAS 9.3 Metadata Server. Have him or her deploy the server tier of a two-machine installation plan. Then, he or she should proceed to deploy the client tier, containing the SAS 9.3 client applications, onto a desktop PC.

6. Open SAS 9.3 Management Console, log in to the SAS 9.3 Metadata Server as an unrestricted user, and in the Folders tab, to import *everything* from the SAS Package File you created in step 4 into the root folder. Remap associations when prompted to do so by the Import SAS Package wizard, according to the mappings you previously noted for file paths, library paths etc.

7. **Complete basic post-installation testing** and validation of the SAS 9.3 deployment. Follow the instructions in the post-installation instructions.html document. Validate that you can connect to the new SAS 9.3 deployment with your SAS 9.3 client applications. In this SAS 9.3 deployment, there is no need to perform extensive validation of the migrated content to test that it works – we do not expect it to work well because we will not migrate data, or spend effort connecting SAS to any data

sources. At most, check that the metadata objects which make up your application content, are present: jobs, tables, libraries, transforms, flows, stored processes, etc.

STAGE 2: SAS 9.3 TO SAS 9.4

8. If you do not have one open already, **open an instance of the SAS 9.3 Management Console**, and log in to SAS 9.3 Metadata Server as an unrestricted user. Export *all the metadata under the root folder*, as one or more SAS Package (.spk) files. If SAS needs to access files on the disk in order to export your content, which is common, you will be prompted for alternative credentials – use an account that can access those files via SAS.

9. Have an experienced installation consultant **run the SAS 9.4 Deployment Wizard** on the machine that will host your SAS 9.4 Metadata Server. Have him or her, perform a 'clean' installation of SAS 9.4. If there are multiple machines in your SAS 9.4 deployment plan, install SAS 9.4 on each machine in the appropriate order.

10. Similarly to step 7 above, **complete basic post-installation testing** and validation of the SAS 9.4 deployment. Follow the instructions in the post-installation instructions.html document. Validate that you can connect to the new SAS 9.4 deployment with your SAS 9.4 client applications. You might also want to complete validation guides published by the SAS Global Solution & Technology Innovation Center (STIC) where relevant. Since this is a permanent deployment, be diligent in validating the install.

11. Define and begin executing a list of **post-installation platform configuration tasks** to set up configuration, services, and scheduled housekeeping tasks which are not part of the standard SAS deployment. You should aim to complete them by, or before, the time you reach step 18 below, and begin parallel running your new SAS 9.4 deployment alongside your existing SAS 9.1.3 deployment.

12. **Import the SAS Package Files** that you exported from SAS 9.3 in step 8 above, in a valid order if there is more than one package. You may want to be selective about what you import, for example to only import metadata that you and your stakeholders wanted to have in SAS 9.4. If you discover at this step that any of the objects you wanted to import depend on other objects which you did not previously choose to import, there should be no difficulty in importing those additional objects too. This is because you upgraded all the metadata objects you could export from SAS 9.1.3 to SAS 9.3, and exported one or several packages containing all those objects from SAS 9.3. It's better to be selective about which objects you upgrade at the last available opportunity, rather than at the first.

13. Having done that, **migrate any other content** (such as customizations to configuration files, macros, formats, and possibly data) to SAS 9.4. For nearly all SAS platform migrations, there will be a significant quantity of application content on your SAS 9.1.3 platform (such as data, code, file system files and directory structures outside the configuration directory, scripts, customizations to user's profiles and umask settings, etc.) which are not migrated even if you do use the SAS Migration Utility. Wherever you have a choice of taking this content from SAS 9.3, or SAS 9.1.3, take it from SAS 9.1.3. Adapt or convert it if necessary on the way.

14. **Test assets are present in SAS 9.4.** You may want to use (or write) code that can report the content of metadata in both SAS 9.1.3 and SAS 9.4, and compare the lists, to assist you in this. For programmers, see the *SAS(R) 9.4 Language Interfaces to Metadata, Second Edition*, which describes among other things, the "DATA Step Functions for Reading and Writing Metadata." The functions I find most useful here are METADATA_GETATTR, METADATA_GETNASN, METADATA_GETNATR, METADATA_GETNOBJ. Don't use functions that will change your metadata unless you are sure you know what you are doing. These functions are also in SAS 9.1.3 and SAS 9.3. Here is an example call, used in a DATA Step, which returns the number of jobs in the metadata repository:

```
Count = metadata_getnobj("omsobj:Job?@Id contains '.'",1,'');
```

15. **Test your security model** in SAS 9.4. Examine the configuration and integrity of the implemented security model in your new environment, to verify that users who should have access to metadata, data, and functionality have it, and that users who should be denied access are denied it.

16. **Perform system integrity, functional, and operational acceptance** testing to validate that your new SAS 9.4 deployment works. Consider testing for some or all of the following:

- Test that **migrated data and code are present**, and accessible to accounts that should have access to them.

- Test that **jobs, programs, and stored processes run** with no more errors than they had in the source deployment.

- For a small but representative sample of data, test that **data in SAS 9.4 compares well (that it 'reconciles') with data in the source SAS 9.1.3 deployment**.

- Test that **job flows,** scheduled sequences of **batch jobs**, and **stored processes run**.

- Test that **results** output by the source SAS 9.1.3 deployment and by the new SAS 9.4 deployment compare favorably - that they **reconcile well** with each other.

Testing is likely to be the area on which you spend the largest effort during your migration. That is why I recommend you to spend no more effort testing your interim SAS 9.3 deployment than is absolutely necessary. Save your effort to test in SAS 9.4 (vs SAS 9.1.3).

17. **Remediate issues and, if needed, plan a resynchronization** following a lighter upgrade process.

If necessary, **repeat some parts of the migration** using promotion to resynchronize your migrated content and fix issues caused by errors in the previous migration and upgrade run. Identify any new content that was not previously migrated, content that was migrated incorrectly (e.g. with incorrectly adjusted reference mappings), and application content that has changed on the source SAS 9.1.3 deployment since you first packaged it up with the SAS Migration Utility or first copied it to SAS 9.4. With each repeated resynchronization, improve your migration process.

18. When your upgrade looks like it has been successful, **run ETL and data processing in your new SAS 9.4 deployment, regularly and in parallel with your exiting SAS 9.1.3 deployment.** If your deployment does not have ETL jobs, do whatever else is appropriate to populate it with data, and use it for reporting, analytics or its other main uses, in the same way as your existing deployment. Each time the processing runs, or daily, or at some other appropriate interval, run your custom **reconciliation program** to compare the results. If necessary, because you identify an issue and its cause, resynchronize the deployments again, repeating selected parts of the migration to correct errors in the migrated content or to pick up new changes made to the source deployment, and fix the content on the target deployment.

Also, if you have not yet begun a change freeze for the upgrade, you should plan for one to start at the moment you complete your last resynchronization. From that moment onwards, any changes made to the source SAS 9.1.3 deployment – that risk making the parallel run and reconciliation results different between SAS 9.1.3 and SAS 9.4 – will have no means of getting into the target SAS 9.4 deployment before business services are switched over to it.

19. **Prepare business users for the switch over** to SAS 9.4. While you are testing, resynchronizing, running regular reconciliation reports and fixing issues in the new SAS 9.4 deployment as you identify them, deploy the new SAS 9.4 client applications (such as SAS Enterprise Guide, SAS Data Integration Studio, the SAS Visual Analytics web interfaces, etc.) to business end users. If appropriate, produce orientation material for end users in each major constituent group, to help make their initial experiences of using it smooth and free from frustrations.

When you have a sufficient body of good reconciliation reports showing that both deployments are behaving similarly well, you are synchronized between SAS 9.1.3 and SAS 9.4, your change freeze has begun, and your end users are as prepared as they need to be to use the new SAS 9.4 platform, you are ready to switch business services over from SAS 9.1.3 to SAS 9.4.

20. You can now **switch business services over from the old SAS 9.1.3 deployment to the new SAS 9.4 one**. Your switch over plan should revolve around the end users experience as much as it does around the technical procedure and the system and data integrity. It should involve as little change as

possible to your actual deployments. Having planned the switch over, follow your plan, keep communicating with the team involved throughout the process, and **execute the switch over**.

21. After you have switched over, **keep the old deployment running for a short period of time**. Keep feeding new data into it if you have ETL processes. Keep all the services running. It's valuable to have the option to switch back to the existing deployment if you discover a serious problem in your new deployment that testing before the switch over did not reveal. Also, in the event that you have inadvertently left a dependency between something in your new deployment and something in the old deployment, it's better to **choose a time when the new deployment is not under heavy use to turn off the old deployment**.

22. When you have stopped processes running on the old deployment, you can **begin decommissioning**. The time you can do this might be different if you are upgrading your application content from SAS 9.1.3 to SAS 9.4 in multiple related deployments at (roughly) the same time. You might prefer to wait until business services for *all* your deployments in your ecosystem have been switched over before beginning the decommissioning process.

In this section, we have looked in detail at the typical steps that you should follow to upgrade a single SAS deployment running core SAS Intelligence Platform products.

While there can be lot of steps, none of them should individually be difficult or risky. With a good plan, you should be quite confident about approaching your Intelligence Platform upgrade from SAS 9.1.3 to SAS 9.4. The benefits you can get outweigh this effort, making the upgrade hugely worthwhile.

Next, we'll look at what to do for customers who have multiple related SAS deployments, e.g., a development, test, and production deployment, and what you can do if one of your deployments is extremely busy and must be only minimally impacted while you migrate and prepare for switch over.

HOW TO UPGRADE MULTIPLE BUSY DEPLOYMENTS

I find the term **ecosystem** is a useful way to describe multiple related deployments. Figure 3 shows a representation of an ecosystem consisting of a development, a test, and a production deployment.

Note that one deployment may be distributed across several hosts, but the separate hosts in each deployment are not shown in Figure 3. Deployments in an ecosystem do not necessarily have the same topology as each other; production may often need more host machines than development does for the same tiers (metadata, compute, middle-tier), to provide appropriate processing power for each one. To put it another way, the same tiers will usually exist in all deployments in an ecosystem, but those tiers are not necessarily deployed over the same number of hosts, nor in the same arrangement.

Figure 3 A representation of an ecosystem, consisting of three SAS deployments. The individual hosts for each deployment (where there is more than one) are not shown in this figure.

If you have multiple SAS deployments in an ecosystem, you should upgrade all of them roughly together. This avoids two problems:

1. It limits the period of time during which you have to promote content between dev, test, and prod, and upgrade it to a later version of SAS at the same time, and

2. It avoids the situation in which you are unable to promote content between the deployments because prod is at an *earlier* version of SAS than dev or test.

If you have several related deployments in your source SAS 9.1.3 ecosystem, the best approach is to migrate and upgrade application content from your source SAS 9.1.3 ecosystem to your target SAS 9.4 ecosystem, using a single interim SAS 9.3 deployment for metadata migration, even if you upgrade content from more than one of the source SAS 9.1.3 deployments. For a brief period of time during the overall migration process, both (or all three) ecosystems are running at the same time.

In planning to upgrade the content in an ecosystem of deployments to run on a new version of SAS (on new hosts), there are four important considerations that we have not yet discussed:

- The route to live between your deployments, along which changes and fixes move as they are developed, tested, and put into production.

- Whether you should migrate your deployments sequentially or simultaneously.

- Which deployment in the source ecosystem – or in the target ecosystem – that you will use as the source for the application content that is migrated to each deployment in the target ecosystem.

- How to plan your upgrade if one or more of your deployments is extremely busy.

PLANNING TRANSITIONS IN YOUR ROUTE TO LIVE

The usual route to live for a typical ecosystem might look like that shown in Figure 4.
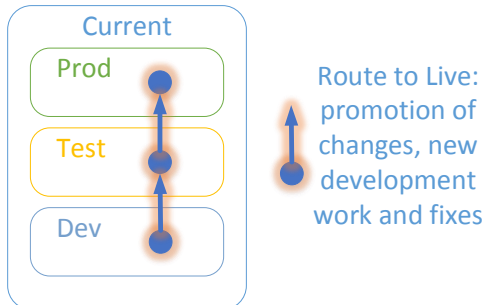
Figure 4. The route to live in a typical ecosystem of deployments is the path from development, to test, and on to production, along which changes, new development work, and bug fixes travel.

But as Figure 5 shows, if you switch only your developers over to using the new development deployment, it is impossible to promote their changes (at least, any changes that include changes to metadata) to the test deployment, because the test deployment would still be running the older version of SAS. To put it another way, switching over a deployment to the new version of SAS, when you have not yet switched over a deployment which is later on the route to live, causes the route to live to become temporarily obstructed.
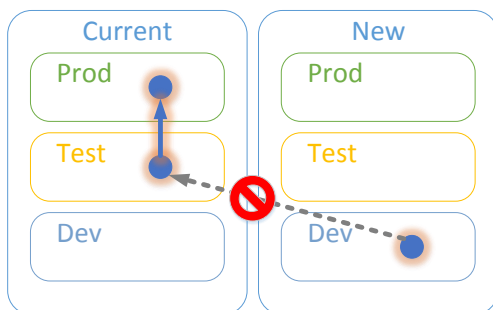
Figure 5. In this transitional state in the overall upgrade of the SAS ecosystem, development has been switched over to the new version of SAS, but test and production have not. Changes to metadata in the new dev deployment cannot be promoted up to the old test deployment. The route to live is temporarily obstructed.

There are two potential solutions to this problem, but I don't strongly recommend either one; instead I would recommend you plan to allow for the temporary obstruction in the route to live.

One potential solution is to upgrade production to the new version of SAS first, as illustrated in Figure 6. Any changes, new developments, or bug fixes that are urgent enough can be promoted to the new version of SAS at any time. However, this carries a greater risk of unexpected behavior in production, than promoting to the next deployment along the route to live when it is running the same version of SAS. I think the high risk of introducing a defect into production makes this approach ill-advised. You should aim to perfect the migration and upgrade process in dev and test first.
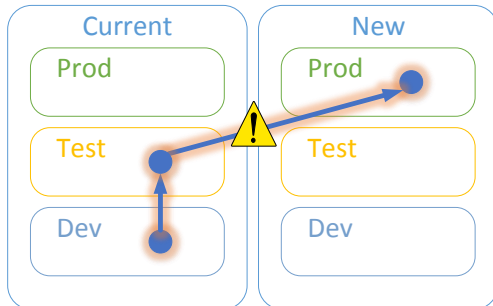
Figure 6. By upgrading production to the new version of SAS first, you keep a potential route to live open. But the risks involved in promoting changes not just between deployments, but between different versions of SAS, make this inadvisable.

The second potential solution is to migrate content to the new SAS deployments, but not to switch over business services to them, until you are ready to switch over services to all your deployments in the ecosystem at the same time. There are two major concerns with this approach which mean it is also difficult to recommend, unless you are prepared to allow additional time and effort to address both concerns. The first concern is that the content in the new development deployment will be significantly out of date by the time you have migrated content to the new test and production deployments. To address this concern, you must perform additional resynchronization before switching over to the new dev deployment. The second concern is that committing to switch over services to all the deployments in quick succession may leave you with insufficient time in your plan to address any unforeseen issues which arise during or after each switchover. If you choose this approach, expect a more 'bumpy ride' in the cutover than with other approaches where the switch from the old deployments to the new are slightly more spaced out over time.

I believe it is most often best to upgrade dev, and switch developers over to it first, accepting the obstruction in the route to live for a time while you allow them to continue developing against the new features in the new version of SAS. The same goes for the switch over for test. If you can tolerate a short period of time in the transitional state, such as the one shown in Figure 5, choose this approach.

SHOULD YOU MIGRATE SEQUENTIALLY OR IN PARALLEL?

You should migrate sequentially.

Allow yourself and your project team the opportunity to learn from your experiences in one migration before performing, or progressing very far, in the next. There are some preparatory steps that you can safely perform in the next deployment before you have completely finished and cut over the last. But the really substantial steps of performing the actual migration are experiences from which you should learn before proceeding.

ECOSYSTEM MIGRATION PATTERNS

The pattern of an ecosystem migration should indicate which deployment is the source of application content for each target deployment at the new version of SAS, and in what sequence should you migrate and upgrade application content to each of those target deployments. They are usually simple.

Supposing an ecosystem has development, test, and production deployments. Some organizations allow no changes to be made initially anywhere other than development. From there, changes are promoted to test, and once ready, to production. During periods of minimal development effort, the development test and production deployments are in close alignment with each other.

If you are in the fortunate position of upgrading one of these ecosystems from SAS 9.1.3 to SAS 9.4, then you should decide whether you would prefer your development (Figure 7), or your production deployment (Figure 8) in SAS 9.1.3 to be considered the 'best' copy from which your application content will be migrated. If the development deployment is the 'best' copy, upgrade and migrate application content from that deployment to the new version of SAS first, and then to promote the application content up to the new test and production deployments.
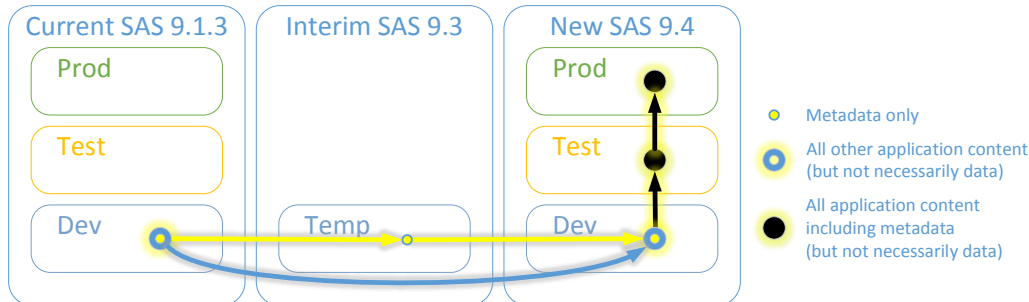
Figure 7. Here, the development deployment is judged to have the 'best' copy of your application content, so it is used as the source to upgrade to the new version of SAS. From there, content is promoted to test and production.
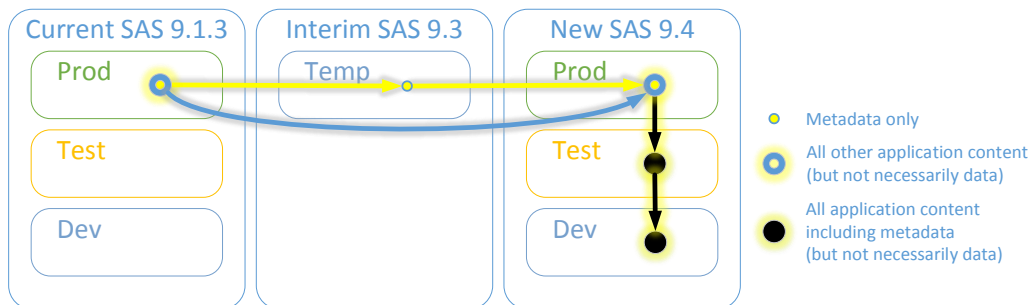
Figure 8. Alternatively, here the production deployment is judged to have the 'best' content, so it is used as the source to upgrade to the new version of SAS. From there, content is 'demoted' to test and development.

However, not all ecosystems are so neatly managed. Other patterns of ecosystem migration may be appropriate if you have content which must be upgraded in more than one of your source deployments, as shown in Figure 9. (The interim SAS 9.3 deployment has been omitted from these diagrams).
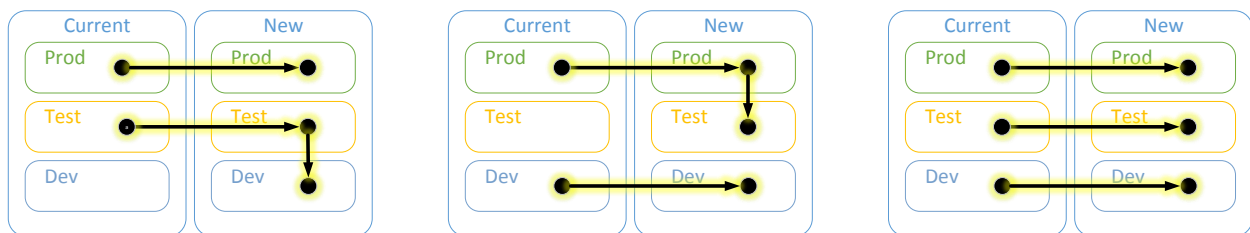
Figure 9. Examples of other valid ecosystem upgrade and migration patterns.

SOURCE OF DATA AND PLATFORM CONFIGURATION SETTINGS FOR UPGRADES AND MIGRATIONS

While it might be desirable to promote code, most configuration settings, and metadata from your new SAS 9.4 test deployment to your new SAS 9.4 production deployment, it is unlikely that you will want to migrate **data**, and all of your **platform configuration settings** to the new production SAS 9.4 deployment from the SAS 9.4 test deployment. I do not mean to suggest in Figure 7, Figure 8, and Figure 9 that data, and all platform configuration settings necessarily follow the same migration path as other content. You would almost always migrate data to the new production deployment from the current production deployment. Certain platform configuration settings (e.g. an alias held in a SAS macro variable for the

deployment name, such as "Production") would similarly be migrated from the current production deployment, rather than from a new test environment. Please forgive this intentional simplification.

UPGRADING CONTENT FROM A BUSY DEVELOPMENT DEPLOYMENT

It's inconvenient to have a change freeze in a busy development deployment, because a change freeze prevents developers from doing much of their work. So, as already described, attempt the upgrade multiple times. During all but the last attempt, your developers can carry on working – though you should have them work on minor bug fixes, and small scale changes, rather than on something really big. Each of these attempts should be more successful than the last. You only need a change freeze during the last attempt, by which time your migration process has been perfected, as illustrated in Figure 10.
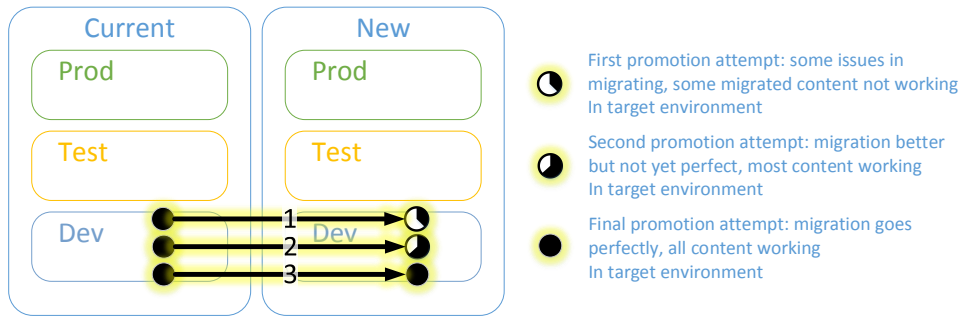
Figure 10. Multiple shots at an upgrade and migration, each successively more complete and successful.

This is particularly valuable when the upgrade is a very complex procedure, that requires practice and refinement before you can make it quick and efficient. The two-stage upgrade from SAS 9.1.3 to SAS 9.4 is an example of a moderately complex upgrade. But there's really nothing about this part of the approach that's specific to upgrades from SAS 9.1.3 to SAS 9.4. I prefer to do this for almost any upgrade.

MIGRATING CONTENT FROM A BUSY PRODUCTION DEPLOYMENT

Some SAS production deployments run at or very near their full processing capacity for a large proportion of every workday. Running the SAS Migration Utility, custom code to report on the content of metadata, or the Export SAS Package wizard all add extra load on the CPU, memory, and disk, especially of the SAS Metadata Server. Such overloading can make a deployment unresponsive until the process completes. This has an unacceptable impact on other users of the deployment.

One way to safely upgrade the application content in a single, over-stressed source deployment (production in this case) is illustrated in Figure 11 below.
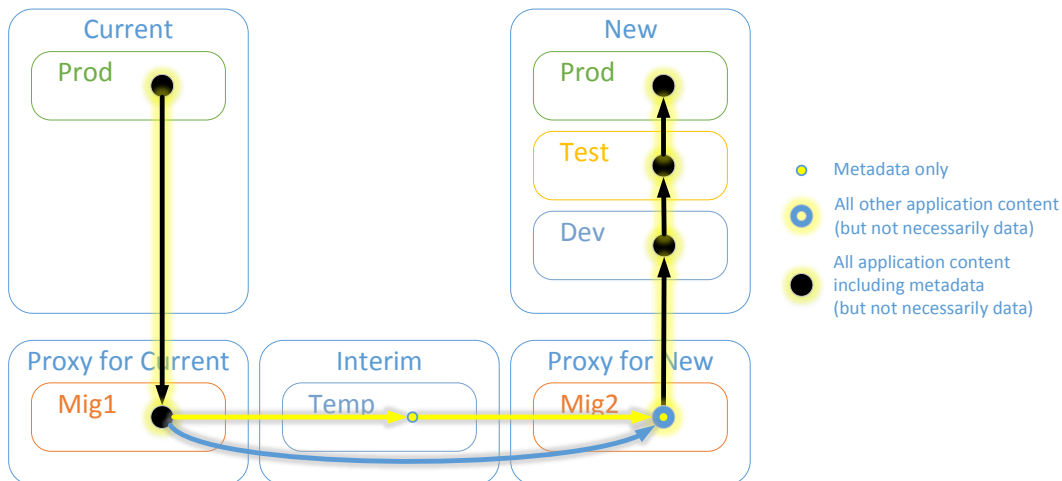

Figure 11. Illustration of the route through which metadata is moved to a proxy deployment using a backup and restore, upgraded to the new version of SAS (in two stages), and then migrated/promoted back to the new SAS deployments.

In this approach, the bulk of the upgrade work is conducted on small, temporary machines in a lab deployment, which may be off-site somewhere else. These deployments serve as proxies for the real current and new SAS deployments. Once the migration is complete, their temporary SAS installations must be removed, and then they can be released and reused for other purposes.

You still need to conduct some basic architecture design work for the proxy deployments, but the main challenges to address in the architecture are how to make small, temporary servers look (to SAS software) as similar as the real deployments as is necessary. Configure the hosts so that their OS version, hostname, and key file system directory structures closely match the source deployment. The hostname is likely to be a challenge for some because you cannot easily have two hosts with the same hostname in the same network. It may only be possible to create these deployments in an off-site lab.

Next, you will need a SAS license file and software depot for each proxy deployment, very much like those for the SAS 9.3 interim deployment and the SAS 9.4 final deployment (before its full-term license is issued) as discussed earlier. If you are upgrading from SAS 9.1.3 to SAS 9.4, you need all **THREE** temporary deployments: two to act as proxies for the current and new real deployments, and a third for the interim SAS 9.3 deployment (which is not a proxy). The version of SAS on each proxy deployment should match the versions on the real source and new SAS deployments, down to having the same SAS ship event.

Again, discuss your proposed approach in detail with your SAS account representative, and keep him or her closely involved in the whole migration and upgrade project. Just as discussed earlier, you will need his or her support in order to be granted temporary license files for the three temporary deployments, and you should keep him or her updated on your progress and estimated completion date, to avoid accidentally incurring a cost for requiring an additional license for longer than you agree with them is a reasonable time. I recommend you also discuss this approach with SAS Professional Services.

Next, obtain the latest metadata backup from the source ecosystem's production metadata server. Restore the backup to your proxy of the current deployment. This step resembles a disaster recovery scenario. Copy code and macros from the SASEnvironment directory in the real source deployment's configuration directory (i.e., …/config/Lev$n$/SASEnvironment/…) to this deployment. There is no need to copy data, external code, or other non-metadata application content to this deployment.

You should at least start the SAS Metadata Server and the Object Spawner on your SAS 9.1.3 proxy deployment. Export metadata and associated content as SAS Package Files from the proxy current deployment using SAS Management Console. I would not expect jobs or other SAS programs to run very successfully in this deployment, because it lacks data and other configuration, but they don't need to work here. All you need is for the SAS Metadata Server and the Object Spawner to start, so that you can log in using SAS Management Console, and launch a Workspace Server.

This proxy SAS 9.1.3 deployment is then used to migrate and upgrade metadata, via an interim SAS 9.3 deployment as described above, to a proxy of the new deployment. From there it can be promoted to the real new SAS 9.4 deployments. The rest of the upgrade procedure is as we have seen already.

In this way you can take multiple attempts at migrating and upgrading, without having to negotiate planned outages, to export metadata from a deployment that contains the same metadata content as your real current production server, but is not used by anyone else.

CONCLUSION

You have seen that upgrading from SAS 9.1.3 to SAS 9.4 is entirely feasible, and that no one step in the very flexible process outlined is unreasonably difficult. If your application content is especially complicated, or in a questionably well-maintained state, there will likely be some work involved in making it work well at SAS 9.4, but doing so is entirely possible.

If you own, use, or look after a SAS 9.1.3 deployment, ask the other stakeholders in your deployment to work with you to upgrade your deployments to SAS 9.4. The benefits are hugely worthwhile.

REFERENCES

SAS Institute Inc. 2015. "Upgrading from SAS 9.1.3 to SAS 9.4." Cary, NC: SAS Institute Inc. Available at http://support.sas.com/rnd/migration/utility/upgrade.html#from913.

SAS Institute Inc. 2015. "Support Services for Current and Prior Releases of Software." Cary, NC: SAS Institute Inc. Available at http://support.sas.com/techsup/support.html#non-current.

SAS Institute Inc. 2015. "Migrating from SAS 9.1.3." Cary, NC: SAS Institute Inc. Available at http://support.sas.com/rnd/migration/utility/upgrade93.html#from913.

SAS Institute Inc. 2015. "Planning Your Migration" Cary, NC: SAS Institute Inc. Available at http://support.sas.com/rnd/migration/planning/.

SAS Institute Inc. 2015. "Inventory of Migration Tools" Cary, NC: SAS Institute Inc. Available at http://support.sas.com/rnd/migration/planning/files/tools_inventory.html.

SAS Institute Inc. 2013. *SAS® 9.4 Intelligence Platform Overview.* Cary, NC: SAS Institute Inc.

SAS Institute Inc. 2011. "SAS Migration Utility Reference." *SAS® 9.3 Intelligence Platform: Migration Guide.* Cary, NC: SAS Institute Inc. Available at http://support.sas.com/documentation/cdl/en/bimig/62613/HTML/default/viewer.htm#p08intellplatform00migrategd.htm

SAS Institute Inc. 2009. "Special Considerations for Promoting Metadata from SAS 9.1.3 to SAS 9.2." *SAS® Intelligence Platform: 9.1.3 to 9.2 Migration Guide.* Cary, NC: SAS Institute Inc. Available at http://support.sas.com/documentation/cdl/en/bimig/61696/HTML/default/viewer.htm#mig023.htm

Wicklin, Rick. 2013. "How old is your version of SAS? Release dates for SAS software." Cary, NC: SAS Institute Inc. Available at http://blogs.sas.com/content/iml/2013/08/02/how-old-is-your-version-of-sas-release-dates-for-sas-software/

ACKNOWLEDGMENTS

CONTACT INFORMATION

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

David Stern
SAS Global Enablement and Learning
+44 1628 490851
David.Stern@sas.com
@SASDavidStern