SESUG Paper ###-2025

Yippee Ki-Yay! Successfully Deploying SAS Financial Management with SAS 9.4 M8 Without Losing Your Mind

Osmel Brito-Bigott; Yenireth Gil Villegas, Carlos Arellano Brito, DATANALITICA;
Sergio Llopis, SAS Institute Spain

ABSTRACT

Deploying SAS Financial Management (FM) 5.62 on SAS 9.4 Maintenance 8 (M8) can feel like navigating a high-stakes mission—one that requires precision, planning, and a solid understanding of SAS's multi-tier architecture. This paper provides a practical, experience-based guide to installing and configuring SAS FM in a modern M8 environment, helping administrators avoid common pitfalls and reduce deployment time.

The session covers strategies for improving reliability and maintainability post-deployment, including environment validation techniques and the use of scripted automation to reduce manual errors and ensure repeatability. Configuration tips, validation checklists, and real-world lessons learned are shared to help ensure a smoother, more predictable implementation experience.

Whether you're deploying SAS FM for the first time or upgrading in an existing SAS 9.4 ecosystem, this paper will provide practical insights and confidence to approach the process effectively.

Yippee Ki-Yay, SAS Admins! Let's deploy SAS FM 5.62 the smart way.

INTRODUCTION

One of the most critical aspects of managing a SAS architecture is staying current with the latest maintenance release. This is not only relevant for installations that rely primarily on SAS Enterprise Guide, but it becomes absolutely vital in more complex environments—particularly those that include specialized solutions such as SAS Financial Management. Keeping your platform on the most recent release, such as SAS 9.4 M8, ensures that you benefit from bug fixes straight from the factory, performance optimizations, and, most importantly, the ongoing support of platform experts.

Just as John McClane in Die Hard 4.0 had to adapt quickly to evolving threats in order to survive, administrators and architects must adapt to evolving software environments to keep systems stable and secure.

This paper will shows the process of upgrading to an M8 environment, with a focus on architectures where SAS Financial Management plays a central role, ensuring you not only survive the upgrade but come out stronger—Yippee Ki-Yay!..

WHAT UPGRADE OPTIONS DO WE HAVE?

When planning an upgrade in SAS, there are three main approaches available. The choice of method is always determined by the time and resources you have at your disposal:

- Upgrade In Place
- Upgrade on New Machines
- Upgrade on a Separate Level (Lev2, Lev3, etc.)

We are going to focus into describe scenarios 2 and 3.

UPGRADE IN NEW MACHINES

This method, often preferred for its safety and flexibility, involves setting up a completely new environment on separate hardware and then migrating the content and configuration from the old environment into the new one.

General Description:

SAS 9.4 M8 and SAS Financial Management 5.62 are installed from scratch on a new set of servers. Once the new environment has been validated and is operational, data and metadata are migrated from the old production system.

Detailed Migration Process:

Step 1: Prepare the File Structure

Replicate the original compute server's directory structure on the new machines. This ensures that metadata paths and SAS program references remain valid without modification.

Step 2: Transfer Compute Data

Perform a complete transfer of all assets (.sas programs, flat files, etc.) into the identical directory paths created in Step 1.

Step 3: Migrate SAS Metadata

Use SAS Management Console to export the complete metadata tree from the old environment. It is crucial to ensure that the export includes the Financial Management content located at:

Products/SAS Financial Management/StageFM.

This copy is then imported into SAS Management Console in the new environment.

Step 4: Redeploy Metadata Objects

After import, redeploy solution objects. This step reassociates imported objects (libraries, servers, etc.) with the new resources in the M8 environment.

Step 5: Migrate Financial Management Data (SASSDM Database)

This critical manual process uses PostgreSQL utilities (pg_dump and pg_restore) to move the sassdm database from the old environment to the new one.

Advantages:

Security: The original production environment remains intact and fully operational during the entire installation and configuration process.

Comprehensive Validation: Full testing and validation can be conducted in the new environment before go-live.

Simple Rollback: If issues arise, the new environment can simply be discarded while continuing to use the old one.

Key Considerations:

Resource Requirements: Requires additional hardware, leading to higher upfront costs.

Migration Complexity: Transferring data and metadata introduces additional steps and complexity.

UPGRADE IN A SEPARATED LEVEL

This is a hybrid technique that creates a new SAS configuration (for example, Lev2) on the same hardware, coexisting with the old configuration (Lev1).

General Description:

The SAS Deployment Manager is used to install a new SAS configuration instance in a different directory (e.g., SASCONFIG/Lev2) on the same server. Lev1 and Lev2 can coexist, although they are typically not run simultaneously.

Detailed Migration Process:

The approach depends on where the compute data resides:

- Case A: Data Outside the Configuration Directory (Best Practice): If compute data and SAS code are stored in generic paths (e.g., `/sas/data or /sas/code`) outside of `SASCONFIG/Lev1`, then no file movement is necessary. The new Lev2 configuration can simply point to these existing directories.
- Case B: Data Inside the Configuration Directory: If compute data resides within SASCONFIG/Lev1, then copying is required. Create the same subfolders under SASCONFIG/Lev2 and transfer all files (code, tables, etc.) from Lev1 to Lev2.

Common Steps for Both Cases:

- Metadata Migration: Export the metadata tree from Lev1 (including the StageFM folder) and import it into Lev2 via SAS Management Console.
- Object Redeployment: Redeploy metadata objects so they align with Lev2 configuration.
- SASSDM Database Migration: Use pg_dump and pg_restore to migrate the Financial Management database, taking care to adjust for port differences since Lev2 often uses a different PostgreSQL port.

Advantages:

- No New Hardware Required: A more cost-effective option.
- Isolation: Lev2 can be tested independently without affecting Lev1, and rollback is simple (restart Lev1 services).

Key Considerations:

- Resource Management: CPU, memory, and network ports must be carefully managed to prevent conflicts between Lev1 and Lev2.
- Configuration Complexity: Maintaining two SAS configurations on the same machine can complicate administration if not managed properly over time.

WHAT HAPPENS WITH MY DATA?

The approach to data migration depends heavily on the type of upgrade being performed. In the case of an Upgrade In Place, the process is relatively simple: data is migrated automatically as part of the Deployment Manager's step-by-step execution. No additional manual interventions are required.

However, things become more complex when performing an Upgrade on New Machines or setting up a Second Level configuration (Lev2). These scenarios require careful planning, precise execution, and strict attention to detail. Much like Thomas Gabriel in Die Hard 4.0—who exploited every overlooked system weakness—failing to account for the small technical details in this process can lead to significant disruptions.Crear los mismos folders del servidor de cómputo en las nuevas máquinas.

Upgrade on New Machines

In this scenario, it is essential to plan for a downtime window to transfer data between servers. The key steps include:

1. **Replicate Directory Structures:** Create the same compute server folders on the new machines.
2. **Transfer SAS Assets:** Move .sas programs, SAS tables, and flat files to the same paths used in the original installation.
3. **Migrate Metadata:** Export a complete metadata tree from SAS Management Console, ensuring that the **Financial Management folder** at Products/SAS Financial Management/StageFM is included, and import it into the new environment.
4. **Redeploy Metadata Objects:** Re-associate imported objects (libraries, servers, etc.) with the new M8 resources.

Upgrade on a Second Level (Lev2)

For a second-level configuration, whether or not data transfer is required depends on where compute data resides:

- **If Data Resides Outside Lev1 (Best Practice):** No copying is necessary. Lev2 can point directly to the existing directories.
- **If Data Resides Inside Lev1:** Replicate the Lev1 directory structure in Lev2 and transfer .sas programs, tables, and flat files to their corresponding paths.

Regardless of data location, two critical steps must always be completed:

1. **Migrate Metadata:** Export from Lev1 (including StageFM) and import into Lev2.
2. **Redeploy Metadata Objects:** Align imported objects with Lev2 resources.

Financial Management Data and Dimensions

But what happens with the **data and dimensions already loaded into SAS Financial Management**? At first glance, one might assume everything must be rebuilt from scratch—a daunting prospect. Fortunately, SAS Financial Management uses its own **PostgreSQL database (SASSDM)**, which allows a clean **export/import process** similar to standard database migrations.

The steps are straightforward if executed carefully:

1. **Create Database Backups:** Use pg_dump to create backups of both the original and new SASSDM databases. For Linux, an example command is:

   ```
   /opt/sas/SASHome/SASWebInfrastructurePlatformDataServer/9.4/bin/pg_dump \
   --host localhost --port 9632 --username "sassdmdbadm" \
   --format custom --verbose \
   --file "/home/sas/FileName.backup" "sassdm"
   ```

   - Replace port and path values according to your installation.
   - You will be prompted for the sassdmdbadm password.
   - This backup ensures you have both a blank installation copy and a copy of your production database.

   The process is identical in Windows using the installed pg_dump.exe.

2. **Validate Access to the New Environment:** From a client machine, log into SAS Financial Management 5.62 (associated with SAS 9.4 M8) to ensure there are no errors before proceeding with the restore.
3. Restore the Database:Use pg_restore in the new environment (Lev2 or new servers):

   ```
   /opt/sas/SASHome/SASWebInfrastructurePlatformDataServer/9.4/bin/pg_restore \
   --host localhost --port 9632 --username "sassdmdbadm" \
   --dbname "sassdm" --verbose "/home/sas/FileName.backup"
   ```

   - Adjust port, paths, and folder names according to your installation.
   - If restoring to Lev2, expect possible errors about the existing *public* schema. This can be resolved using **PgAdmin** with a `DROP CASCADE` before repeating the process.

4. **Apply Structural Adjustments (Due to Version Changes):**SAS 9.4 M8 introduces modifications to certain SASSDM table structures. For example, the comment_id column in the sas_audit table requires an adjustment:

   ```
   ALTER TABLE sas_audit
   ALTER COLUMN comment_id TYPE character varying(255)
   USING comment_id::character varying(255);
   ```

   Validate that the values remain intact after applying the new data type.

Final Note

Performing an upgrade on new machines or a second level requires vigilance, precision, and preparation. As Thomas Gabriel demonstrated, one overlooked detail can bring down an entire system. Following these steps ensures that your SAS Financial Management data, metadata, and custom developments survive the upgrade intact—leaving you in control, not at the mercy of hidden risks.

AND THE CLIENT MACHINES?

With this new maintenance release, several solutions have undergone version changes, which means that the **entire depot software must be reinstalled**. The recommended approach is to work initially with **two environments**: one machine running the current installation and another where the new depot is deployed.

This is especially important because **SAS Financial Management 5.62 introduces incompatibilities** with form packages from earlier versions. In other words, forms that worked fine in your previous setup will not run as-is in the new release. Ignoring this step is like ignoring Matt Farrell's warnings in *Die Hard 4.0*: the system will break, and you'll be left wondering why it collapsed.

Required Steps

1. **Validate Python Installation:**After installing SAS Financial Management 5.62 on client machines, ensure that Python is properly installed.
2. **Clean Up Old Directories:**Delete the folder (if it exists):

   ```
   {Drive}\Users\{youruser}\SAS\FinancialManagementStudio\5.62
   ```

3. **Run the Update Utility:**From the Command Prompt (as Administrator), run the vjr_update_manifest program, pointing initially to:

   ```
   C:\Program Files\SASHome\SASFinancialManagementStudio\5.62
   ```

4. **Validate the New Installation:**Open SAS Financial Management 5.62 and verify consistency against your previous environment.
5. **Migrate Forms Correctly:**Only after completing these steps should you migrate forms for each cycle. This requires importing **SPK packages** generated from a machine connected to the original installation. Before doing so, remove the placeholder forms visible in the phases of the new installation.

Critical Considerations

It is important to understand that—even though the database migration process appears to copy all solution objects—**forms are the one element that do not transfer automatically** into the new database. If you attempt to open or import forms without following the procedure above, the system will throw errors.

Think of it as Farrell's golden rule: *"If you don't update the manifest and clean up the old paths, the system won't know what to do with your forms."* These precautions ensure your forms survive the upgrade and integrate properly into SAS Financial Management 5.62.

CONCLUSION

Upgrading to SAS 9.4 M8 with SAS Financial Management 5.62 is not a straightforward mission—it requires planning, discipline, and attention to every technical detail. Along the way, there are pitfalls that can derail the process: incompatible form packages, metadata alignment issues, PostgreSQL migrations, and the hidden complexities of multi-level configurations. Much like the digital battlefield in *Die Hard 4.0*, each of these challenges can appear overwhelming when taken together.

But just as John McClane ultimately triumphed over Thomas Gabriel's cyber-siege through persistence, adaptability, and sheer determination, we too can overcome the difficulties of this upgrade process. By combining a clear strategy, rigorous validation, and proven migration techniques, the "bad guys" of broken deployments, failed validations, and corrupted forms can be defeated.

In the end, the lesson is clear: upgrades are not only about moving to the latest software version—they are about ensuring stability, protecting existing investments, and enabling new capabilities for the future. With the right methodology, even the most complex SAS Financial Management upgrade becomes not just survivable, but winnable. And like McClane standing victorious against impossible odds, we can proudly say after the process is complete: **Yippee Ki-Yay, mission accomplished.**

ACKNOWLEDGMENTS

We would like to express our deepest gratitude to **Raúl Romero Pecho from SAS Spain**, whose guidance and expertise in **SAS Financial Management** have been invaluable across multiple projects. His dedication and willingness to share knowledge not only helped us successfully navigate the complexities of this solution, but also inspired us to approach challenges with confidence and precision.

Raúl's mentorship has been a cornerstone of our work, and this paper would not have been possible without the lessons learned through his support. Beyond the technical achievements, we owe him an invitation: Raúl, it's time to come to the Caribbean, to Santo Domingo, and enjoy a cold **Presidente beer** with us—a well-deserved toast to celebrate both the victories in analytics and the friendships forged along the way.

CONTACT INFORMATION

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

Osmel Brito-Bigott
DATANALITICA
osmel.b@datanalitica.com

Yenireth Gil Villegas
DATANALITICA
yenireth.g@datanalitica.com

Sergio Llopis
SAS Institute Spain
segio.llopis@sas.com

Carlos Arellano Brito
DATANALITICA
carlos.ab@datanalitica.com