

SAS Migration from Unix to Windows and Back

Valerie Williams, ICON Clinical Research

ABSTRACT

SAS® software has been a mainstay for performing analysis and reporting (A&R) in the pharmaceutical industry, for the past four decades. SAS occasionally releases new versions of their software with code enhancements and bug fixes to SAS® functions, procedures, and/or elements. In order to exploit those changes, SAS users may need to upgrade their programming environment and migrate their SAS programs and data to a new platform. Migrating A&R directories across operating systems has proven to be the most challenging, of any migration process, but much can be learned from migrating to a different operating system twice. Resources from SAS, IT, and all business units that use or plan to use SAS must work together to perform the appropriate level of testing and documentation, to ensure that proper directory permissions are in place and that the new environment is fit for purpose, without affecting business unit delivery timelines.

This paper will endeavor describe to a mid-level SAS programmer the process and pitfalls of migrating SAS programs and data across SAS versions, SAS platforms (native SAS to SAS GRID), and operating systems and provide a few helpful tips to avoid problems, when migrating.

INTRODUCTION

In information technology (IT), migration is the process of moving from the use of one operating environment to another operating environment that is, in most cases, thought to be a better one. Migration can involve upgrading to new hardware, new software or both. The migration process includes making sure the new environment's features are exploited, old settings do not require changing and that current applications continue to work. According to Techopedia.com, software migration is the transfer of a company's programs used to operate computers and related devices into a new, upgraded operating system. Conversely, system migration is the transfer of business process IT resources to a newer hardware infrastructure or a different software platform.

Why migrate from one operating system to another? There can be many reasons that an organization is motivated to take this step. A few reasons are to reduce licensing costs, the need for a better platform (i.e., architecture, security and file management of directory structure and permissions, user experience), for improved performance or because the current platform such as Citrix is becoming obsolete or is being phased out. Key factors to consider when migrating are hardware, operating systems, software versions, directory structures, applications, and program updates. A change in any of these elements will affect the time required to plan, execute, and close a migration project. A comprehensive migration assessment will identify what needs to be addressed. Teams should take a moderately conservative, yet risk-averse approach to migration, according to their business needs. In the pharmaceutical industry, different research business units such as Data Management, Biostatistics, Statistical Programming, Pre-Clinical/Early Phase, Real World Data, Data Analytics, and Clinical Risk Management work differently and will need to use migration strategies that are suitable for their own way of working.

PROGRAMMING ENVIRONMENTS

This scenario describes two separate migrations that were conducted, over a matter of months. For the first migration, the initial operating system and software platform was 64-bit Solaris Unix with SAS 9.1.3/9.3, which was located in a single data center. IT and the company's business units decided to migrate directories containing SAS catalogs, programs, and data sets from Unix to two 64-bit Windows servers with SAS 9.3 and eventually, SAS 9.4, in separate data centers within two different time zones. The Unix SAS 9.1.3 migration also included some SAS 9.3 Windows studies. In the second migration project, directories were migrated to 64-bit Red Hat Enterprise Linux SAS GRID/SAS 9.4 servers in a single data center, as shown in the graphic, in Figure 1.

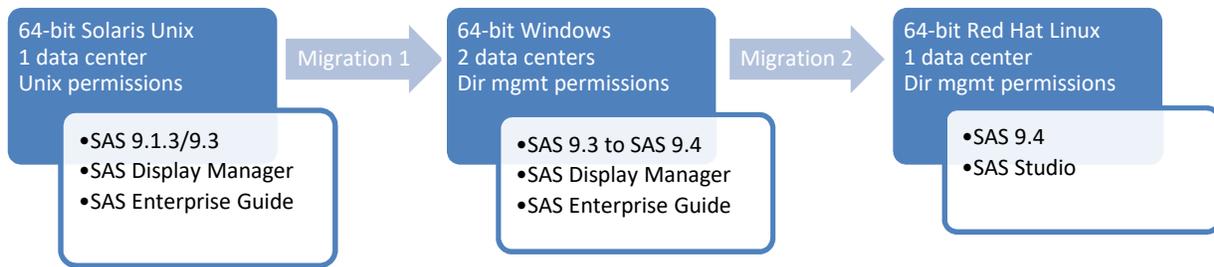


Figure 1. Migration across operating systems

STORAGE, MEMORY, AND PROCESSING

The initial amount of storage that you will need on the new platform depends upon the amount of space that is being utilized on the current platform, plus an estimated amount of storage that that will be required in the first one to two years on the new platform. Assess the amount of terabytes or gigabytes of storage you will need, as shown in Table 1, and then expect to need more than you have anticipated. Conduct a similar exercise to determine what type and how much memory such as Java Virtual Machine Heap Memory that you will need, taking system requirements into account.

Summary of Unix SAS 9.1.3 Files by Type			Summary of Windows SAS 9.3/SAS 9.4 Files				
File type	Extension	Count	Storage Virtual Machine	Share	Volume	Volume Size	Used
SAS Program	.sas	995,363	am-01	SAS_data_1	cifs_data_vol_1	20TB	15.89TB
SAS Dataset	.sas7bdat	604,377	am-01	SAS_data_2	cifs_data_vol_2	1TB	704.5GB
SAS Listing	.lst	318,121	am-01	SAS_data_3	cifs_data_vol_3	675GB	574.6GB
SAS transport file	cpo, xport, xpt	58,454	am-01	SAS_data_4	cifs_data_vol_4	4TB	3.22TB
SAS Catalog	.sas7bcat	3,534	eu-01	SAS_data_5	cifs_data_vol_5	14.50TB	13.44TB
SAS item store file	.sas7bitm	1,742	eu-01	SAS_data_6	cifs_data_vol_6	800GB	163.7GB
SAS Index	.sas7bndx	29	eu-01	SAS_data_7	cifs_data_vol_7	0	0
SAS View	.sas7bview	3	eu-01	SAS_data_8	cifs_data_vol_8	400GB	95.59GB
			eu-01	SAS_data_9	cifs_data_vol_9	16.80TB	15.18TB

Table 1. Unix and Windows Storage

In a SAS GRID computing environment, SAS computing tasks are distributed among multiple computers on a network, under the control of SAS Grid Manager and workloads are distributed across a grid cluster of computers. This workload distribution enables workload balancing, accelerated processing, and job scheduling. SAS GRID is a thin client, which is software that is primarily designed to communicate with a server. A SAS GRID Linux manager production environment could consist of clustered Metadata Servers with 32GB Cores, clustered Mid-Tier Servers with 64GB Cores, and multiple grid nodes with several 64GB cores. SAS Web Infrastructure Data Base JDBC Drivers 9.41 and SAS Web Infrastructure Platform 9.4 are on the Mid-Tier Linux for x64 platform. Redundant hardware components, fault tolerant server chassis, and dual network pathing are also beneficial.

SAS GRID Computing's Enterprise Grid Orchestrator (EGO) monitors the key services and will restart/relocate the service, and will start the service on the failover host, when needed. EGO is included in the Platform Load Sharing Facility (LSF) installation and is part of the SAS Grid Manager for Platform. A sample architecture diagram is shown, in Figure 2.

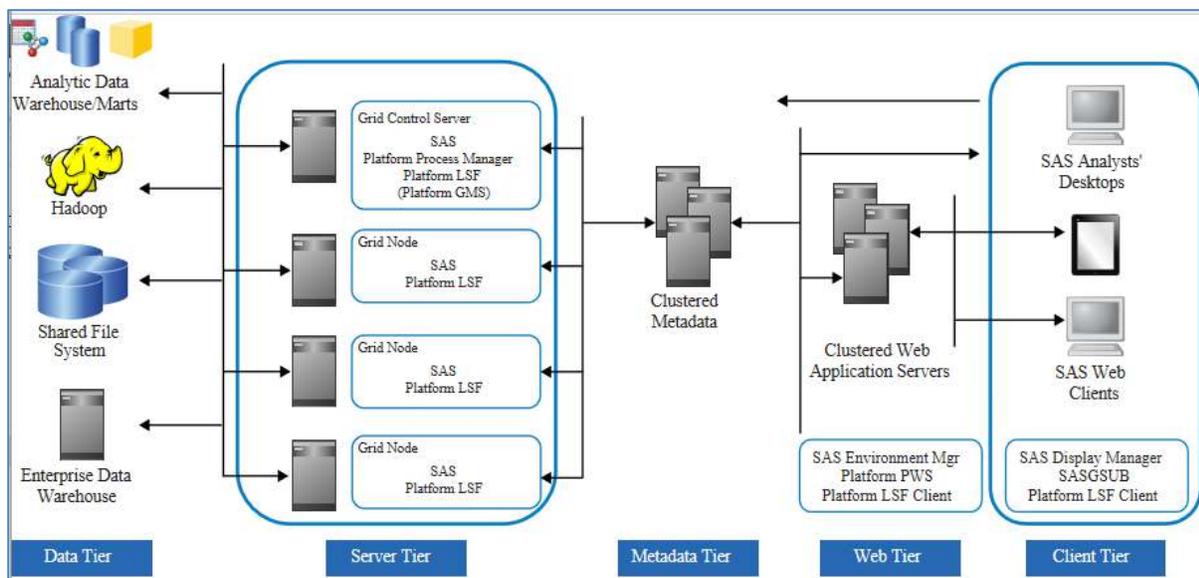


Figure 2. Example of SAS GRID architecture.

Which interface should you use? On SAS GRID, SAS Studio or SAS Enterprise Guide (EG) are the likely choices. Before SAS GRID was available, native SAS Display Manager was popular, in addition to SAS EG. Many non-programmers prefer the SAS EG user experience to SAS Studio because they believe that SAS EG is easier to use. Some programmers have found running programs in SAS EG to be faster than in SAS Studio. SAS Studio is a web-based browser that *connects to a SAS server in order to process SAS user commands. The SAS server can be a hosted server in a cloud environment, a server in your local environment, or a copy of SAS on your local machine. After the code is processed, the results are returned to SAS Studio. SAS Studio supports multiple web browsers, such as Microsoft Internet Explorer, Apple Safari, Mozilla Firefox, and Google Chrome.* (SAS Institute). This option is less expensive, as it avoids the need to have copies of the graphical user interface on multiple servers and personal computers. Keeping only one SAS session open at a time prevents a significant slowdown in performance. Instead, use a different editor such as Ultra Edit to edit multiple programs, at once.

A migration assessment from ThatWave consulting noted that the user experience will change, when migrating across operating systems. Unix super users will miss the power and ease of using a command line interface, for their work, and will often feel powerless when confronted with how to perform the same functions in Windows. In addition, automating routine command sequences through scripts can be a tremendous asset to a super user. The Windows environment supports viewing interim data, during a stepwise development cycle. One unwelcome change for our users occurred, when my company migrated from Windows to Linux and did not allow access to the Unix command prompt, for security reasons. Table 2 contains comparison of the user experience on each operating.

Feature	Windows	Unix	Linux
Interface	Windows interface through native Windows machines or Citrix remotely connected virtual desktops	X-Window Graphical User Interface (such as Gnome or KDE) for file navigation, printing and basic navigation around the system as well as potential SAS/AF applications	SAS Studio web browser, SAS Enterprise Guide Access to SAS GRID directories via Windows Explorer

Command Prompt	Windows DOS prompt and PowerShell	C shell, Perl, UNIX scripting languages via the UNIX command line interface	SAS GRID file access through Windows Explorer - no command line access through SAS GRID. Windows DOS prompt and PowerShell
Editor	Windows clients: SAS Display Manager, SAS Enterprise Guide	SAS file editing and log viewing via "vi" or other UNIX editors	SAS Studio web browser, SAS Enterprise Guide, UltraEdit with syntax highlighting for editing/browsing files
Scheduling	Scheduler	Scheduler	Custom Scheduler for connection to external sources (i.e., databases)
Directory Permissions	Directory file management system	Unix permissions	Directory file management system

Table 2. Operating System User Experience Comparison

MIGRATION STRATEGY

The business units come to an agreement upon an overall migration strategy. Migration can be performed either in bulk (whole cloth) or stepwise, over a period of time, depending upon the platform updates. The best candidates for bulk migrations are those where the operating system remains the same (i.e., Windows-to-Windows or Unix-to-Unix) and there are no or only minor software versions changes. When operating systems, software versions, and directory structures are changing, an incremental migration should be done, to allow for adequate testing and verification. Simply copying files from one location to another without any updating or testing is not considered to be a sufficient migration, as the programs may not function and users will not know what to expect, when they try to run their code on the new platform.

Build Teams

Establish a Migration Core Team consisting of a business project lead and business process subject matter experts who will carry out testing, during the pilot, and will act as consultants, during the actual migration. Decide if a dedicated migration team and study team blend or an individual programmer will perform the migration. If there is statistical code, statisticians will need be on the team and test their own programs because of the complexity of the analysis that they perform. IT Team members may include an IT Business Partner, a Project Manager, a Business Analyst, a Linux or Windows Team Lead, a Storage Lead, an Application support, an IT Service Owner, a Server Team, and a Quality Assurance Technology Analyst. IT can execute a bulk migration over a series of a few weekends, after a few months of preparation, but stepwise migrations are more complex and will probably take more time to complete.

Determine Overall Scope

Data security (i.e., directory permissions), delivery timelines, availability of resources (business units and migration team) are factors that should be considered, when planning a migration. In the regulated pharmaceutical industry, it is of utmost importance to retain the original permissions on directories, when migrating, for compliance reasons and avoidance of rework. When migrating clinical studies, special care should be taken to retain permissions on unblinded directories, to prevent study unblinding.

Elements of migration project that we considered are:

- Number of clinical studies and their complexity
- Resources – dedicated migration team or study team

- Timeframe – based on resources, number/complexity of studies, timing of client deliveries, license expiration date
- Average time to migrate a study
- Consultants – identify types of files; hard-coded paths; and amount of storage needed; initiate and run pilot

MIGRATION PROCESS

The Migration Core Team for each business unit defines a migration process that will have minimal impact on A&R for new studies and future deliveries. They determine the scope and duration, define the migration approach, forms the migration pilot team, train the migration pilot team, execute the migration pilot, and train the teams who will perform the actual migrations. QA performs a Risk Assessment to identify probability, impact, and mitigation of migration risk factors such as security breach, failed audits, downtime, data loss, poor performance, overlapping with business priorities, lack of resources, inadequate migration testing. The remainder of this paper will cover the stepwise migration process.

The Pilot

Prior to the migration pilot, the test environment should be set up where teams can work on pilot studies. Once IT has the system architecture in place and the migration pilot team has been assembled, the migration pilot is ready to start. The migration team selects a few representative studies to pilot and begins the team follows the migration process. The technical lead for the pilot will write a Migration Technical User Guide containing details about how files will be copied to the new environment and tools for updating programs.

Use tools to copy files from the legacy environment to the new environment and to apply some updates to programs for platform independence and to handle the upgrade for newer versions of SAS. We used FastCopy freeware, shown in Figure 3, to copy files and generate a log of the copied files. This served as migration documentation.

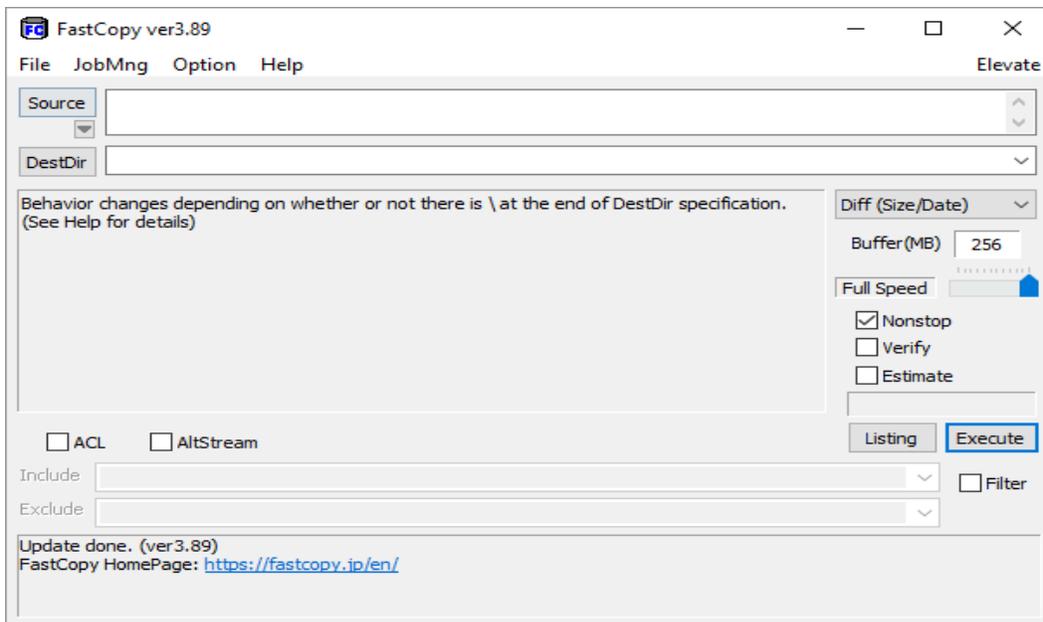


Figure 3. FastCopy Tool

For the migration to SAS GRID, a SAS consultant developed a script to run Proc Migrate on data sets and convert back slashes to forward slashes in SAS programs and also provided a series of macros to run, monitor, and kill SASGSUB batch jobs. In addition, we developed our own in-house tools for making further updates to SAS programs. It required some updating, over the course of the migration. Some of the coding challenges to be aware of during a migration are:

- Non-SAS Applications
- Shell scripts, perl scripts
- ODS, Statistical Graphics devices, positions, fonts
- Hardcoded paths in %include statements, filename and/or libname references.
- Handling macros for titles and footnotes metadata files.
- Non-standard directories
- Opening .pdf files
- Dataset encoding of 32-bit to 64-bit (validvarname=7 before proc import and transpose for SAS GRID)
- Data base connections
- Batch submission
- Saving log to a specific destination
- SAS catalogs

D-Wise makes the following additional recommendations for migration from Windows to Linux and has developed a useful SAS GRID Migration Roadmap (see Figure 4):

- Test out scripting languages such as VBS vs. Linux Shell. Even AIS scripts will not work the same on Linux.
- Replace any “Windows-specific” cleverness, such as DDE and VBS integrations with MS Word.
- Although Excel import and export do work fine, some of the LIBNAME engines are not supported on all operating systems, such as EXCEL and XLSX engines.
- Migrate SAS catalogs. These usually need to be CPORTed on the source system and CIMPORTED into the new system.



Figure 4. d-Wise SAS GRID Migration Roadmap

IT makes updates to the architecture and configuration, as needed and the migration pilot team will adjust the migration process. Some of the Windows to Linux migration challenges were synchronizing the GID and UID (group id and user id) and batch processing. The pilot is considered to be finished, when the business units have a process in place and agree with the results of their migration pilot testing. System back up and disaster recovery measures and procedures should be in place, before the migration begins.

Sample Code to change from back slash to forward slash for SAS GRID - Linux:

```
%macro slash;
  %if %sysfunc(indexc(&path.,'\')) %then %do; %let slash= /; %end;
  %else %let slash = /;
%mend;
%slash;
```

Saving Logs and Output without SASGSUB

```
%let sasloc=%sysget(SASROOT);
%put &=sasloc;
%let PNAME=myspath;
%let PSRCPATH=/opt/sas/data/sas_bu/sas_bu_grp/study/1234;
%let POUTPTH=/opt/sas/data/sas_bu/sas_bu_grp/study/1234;
%sysexec(%sysget(SASROOT)/sas -sysin "&PSRCPATH./&PNAME..sas" -
noautoexec -rsasuser -altlog "&POUTPTH/&PNAME..log" -print
"&POUTPTH/&PNAME..lst");
```

Saving Logs and Output with SASGSUB

Note that logs and output are written to the shared directory named "gridwork", by default.

```
options mprint mlogic symbolgen NOQUOTELENMAX;
%let gsub_location =
/opt/sas/sascompute/config/xyz/Applications/SASGridManagerClientUtility
/9.4;
%let testing=/opt/sas/data/sas_bu/sas_bu_grp/ study/testing;
%sysexec %str(&gsub_location./sasgsub -gridsubmitpgm
"&testing./mytest.sas;" -gridwatchoutput);
```

The Migration

IT releases the production environment, for the start of the production migration work. Table 3 shows a bird's eye view of migrations from Unix to Windows and Windows to Linux. For the Unix to Windows migration, IT implemented a new file management system and a standard directory structure which took several months to complete and involved requirements definition and user acceptance testing by the business units.

Unix to Windows	Windows to Linux
Duration: 18 months for 80 clinical studies <ul style="list-style-type: none"> • Consultant Migration Assessment • Move from one to two data centers • Test directory permissions • Kick- off meetings and training were held for each study team • Dedicated Migration Team plus study teams 	Duration: 8 months for 86 clinical studies <ul style="list-style-type: none"> • Consultant Migration Assessment • Move back to one data center • Test directory permissions • A single Kick- off meeting and Just-in-Time training sessions were held • Dedicated Migration Team plus study teams

<ul style="list-style-type: none"> • Allow one month between regular deliveries • Testing Scope – compared Unix and Windows output and Windows dev and val output • Implemented a new file management system and a standard directory structure 	<ul style="list-style-type: none"> • Allow two weeks between regular deliveries • Testing Scope – compared Linux development and validation output
--	--

Table 3. Migration Overview

Basic steps for a stepwise migration of a clinical study are outlined, in Table 4. Migration needs to be performed at least two weeks ahead of the next planned regular deliveries. The study programmer completes a program inventory checklist of the files that will be copied to the new environment, for the migration programmer to use, as a reference. Documentation is an important part of the migration process, for future audit purposes. Each migration can be documented in a Migration Study Summary and Findings Report, which is signed off by the study team. The migration lead can maintain a Migration Schedule of all upcoming, ongoing, and completed migrations.

General Stepwise Migration Process
<ol style="list-style-type: none"> 1. Identify a study to be migrated 2. Assess size/complexity of study and estimate duration of study migration 3. Use Program Inventory Checklist <ul style="list-style-type: none"> ➤ Take inventory of programs, datasets, and other files for that study 4. Setup the study with the correct permissions, in the new environment 5. Transfer study programs, datasets, and other files <ul style="list-style-type: none"> ➤ FTP or FastCopy files to new environment ➤ Confirm that all files have been copied 5. Update Programs for new server <ul style="list-style-type: none"> ➤ Run script to update hard-coded directory paths (forward or back slashes), native OS commands, and OS environment variables ➤ Replace operating system-specific and SAS version-specific code <ol style="list-style-type: none"> 1. Convert data sets to 64-bit using Proc Migrate ➤ Manually update program, as needed 6. QC Code <ul style="list-style-type: none"> ➤ Run top priority programs for migrated studies ➤ If errors, correct programs and rerun until no errors and acceptable warnings 7. Compare development to validation output (and across operating systems, optional) <ul style="list-style-type: none"> ➤ Use in-house VisualBasic tool to compare WORD output files (reports) <p style="margin-left: 40px;">Note: no 'diff' command equivalent exists on Windows</p>

Table 4. Migration Process

CONCLUSION / RECOMMENDATIONS

Migrations require careful planning. We were able to apply some lessons learned from the migration from Unix to Windows, when we migrated from Windows to Linux migration, and achieved a better outcome. It is recommended that you use consultants with proven experience in executing successful migration projects, in your industry. A good quality upfront migration assessment and a knowledge of both source and destination platforms (e.g., SAS GRID) makes the process go much more smoothly. During the assessment phase, ensure that the new environment will have the same needed functionality as the

legacy environment. SAS batch submission using SASGSUB on SAS GRID was a particular pain point that took quite some time to resolve. It is best to have a dedicated migration team to perform the migration and have it reviewed by the study programmer so that the latter can focus on their regular assignments. To save time, have only one Kick-off meeting at the beginning of the migration project, for the second migration with regularly scheduled training session throughout the project and streamline the testing approach, as the migration project progresses, if possible. Interact with other business units to share knowledge and work through issues.

REFERENCES

D-Wise. 2020. "Migrating to a SAS Grid Platform: It's easy if you have a Roadmap." Accessed April 2, 2020.

<https://www.d-wise.com/blog/migrating-to-a-sas-grid-platform-its-easy-if-you-have-a-roadmap>

SAS Institute. 2020. "Grid Computing in SAS® 9.4, Fifth Edition, High Availability and SAS Grid Manager for Platform." Accessed April 2, 2020.

<https://documentation.sas.com/?docsetId=gridref&docsetTarget=titlepage.htm&docsetVersion=9.4&locale=en>

SAS Institute. 2020. "System Requirements, SAS® Grid Manager 9.46." Accessed April 2, 2020.

<https://support.sas.com/documentation/installcenter/en/ikgridmgrsasser/72745/HTML/default/index.html>

Techopedia Inc. 2020. "Definition - What does *Thin Client* mean?" Accessed April 2, 2020.

<https://www.techopedia.com/definition/462/thin-client>

TechTarget, Inc. 2020. "Definition Migration." Accessed April 2, 2020.

<https://searchcio.techtarget.com/definition/migration>

Nelson, Greg. 2015. ThotWave Technologies. "SAS® Migration Assessment."

ACKNOWLEDGMENTS

I would like to thank Ganesh Prasad, Randi McFarland, and Chris Hill, for their support.

RECOMMENDED READING

Brinsfield, Eric C. 2017. Meridian Analytics. "SAS® Programmer's Guide to Life on the SAS Grid." PharmaSUG.

<https://www.pharmasug.org/proceedings/2017/BB/PharmaSUG-2017-BB11.pdf>

CONTACT INFORMATION

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

Valerie Williams
ICON Clinical Research
+1 215 583 2107
Valerie.Williams@iconplc.com

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

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