

How to Migrate Your SAS Environment from Tru64 UNIX to HP-UX 11i on HP Integrity Servers



| | |
|--|----|
| Introduction | 2 |
| Terminology | 2 |
| Do I need to migrate? | 3 |
| Migration overview and planning for activities | 3 |
| Organizational model for SAS data and programs | 6 |
| Transferring files to a target environment | 7 |
| Suggested migration method – PROC MIGRATE | 8 |
| System or project migration | 8 |
| PROC MIGRATE syntax | 9 |
| Source library member types | 10 |
| Data attributes | 11 |
| Validation | 11 |
| Summary | 13 |
| Acknowledgements | 13 |
| Appendix A: SAS V6 files migration | 14 |
| Appendix B: Traditional conversion methods | 15 |
| For more information | 17 |

Introduction

HP will be phasing out Tru64 UNIX® and the AlphaServer enterprise system line in favor of the HP Integrity server line based on the Intel® Itanium® 2 processor architecture.

With the release of SAS V9.1, the SAS System is now available for the HP-UX 11i v2 operating system on HP Integrity servers.¹ This marks a key trigger in the migration from Tru64 UNIX to HP-UX 11i for your SAS environment. See [For more information](#) for the link to the white paper *When to migrate your SAS environment from Tru64 UNIX to HP-UX Integrity servers*. That document discusses key considerations in determining the proper time frame for migrating.

It is important to develop a plan for migrating to HP-UX 11i on HP Integrity servers. This “How to” paper builds on the “When to” paper referenced above by helping you plan for the migration and by explaining various “how to” techniques. The examples and techniques discussed throughout this paper are based on a source environment of SAS 8.2 or SAS 9 on Tru64 UNIX and a target environment of SAS 9.1.2 (or later) on HP-UX 11i v2. Although some of the techniques in this document might work with previous versions of SAS, they have not been tested. The technical details in this paper may not be applicable to other platforms.

The target audience is members of your migration team, who know SAS and are familiar with your overall SAS project environment. Also, the migration team must work closely with the UNIX system administrator of the Tru64 UNIX and HP-UX 11i environments.

Terminology

It is important to establish a common use of terminology before delving into the specifics of migration strategies. Below are key terms used in this document.

- **SAS migration** – the process that converts SAS files and programs currently running on Tru64 UNIX to equivalents that run on HP-UX 11i.
- **Conversion** – the implementation of the SAS migration.
- **Pilot project** – a test migration of at least one project.
- **SAS files** – all file types that can be processed by the SAS System, including SAS data sets, catalogs, SQL views, ACCESS views, and ACCESS descriptors.
- **Source** or **source environment** – the SAS System on the Tru64 UNIX environment.
- **Target** or **target environment** – the SAS System on the new HP-UX 11i on HP Integrity server environment.
- **Conversion in place** – SAS files to be migrated are first copied from the Tru64 UNIX environment to the HP-UX 11i environment and then converted by reading the Tru64 UNIX files and creating new SAS files on the HP-UX 11i system as converted files. As explained later, this is not a recommended migration method.

¹ Throughout this document, HP-UX 11i will be used to mean HP-UX 11i on HP Integrity servers, as separate and distinct from HP-UX 11i on HP 9000 (PA-RISC-based) systems.

Do I need to migrate?

Yes. Although both systems are 64-bit Virtual Address environments and both have the same IEEE floating point representation, SAS data files have a different internal structure on HP-UX 11i than on Tru64 UNIX. The difference is mainly due to HP-UX 11i being big-endian and Tru64 UNIX being little-endian². As a result of this difference, Tru64 UNIX members are considered “foreign” when accessed by HP-UX 11i.

When you use SAS files that are native to the environment, you increase processing speed and reduce resource consumption. Performance increases because computer resources are not busy with translating and transcoding via Cross-Environment Data Access (CEDA). CEDA is a Base SAS feature that enables foreign Version 7 and 8 files to be processed in a SAS 9 session automatically and transparently. CEDA does not provide access to catalogs or indexes. This is another reason to migrate. You can set the SAS system option `MSGLEVEL=I` to generate a log note that indicates whether a file is being processed with CEDA. That is:

```
OPTIONS FULLSTIMER MSGLEVEL=I ;
```

Here is an example of the message that displays in the log:

```
INFO: Data file SOURCE.ACCTYPE.DATA is in a format native
to another host or the file encoding does not match the
session encoding. Cross Environment Data Access will be
used, which may require additional CPU resources and reduce
performance.
```

The data format is converted as non-native data files are read by SAS. This conversion occurs every time a non-native data object is read, consuming extra resources. After all SAS files are migrated to the HP-UX 11i environment, conversion overhead is no longer a consideration.

Migration overview and planning for activities

Experience has established that any or all of the following activities can be part of the migration effort. Thinking these through helps you create and carry out a better and more successful migration plan.

- Migrating SAS files from one platform to another

This represents the bulk of the effort. The key to successfully converting SAS files from one platform to another is understanding the number of SAS programs and SAS data sets that must be migrated as a unit. (This is called a **project** after the system level is migrated; see the section [Organizational model for SAS data and programs](#).) It is important to pay attention to SQL and ACCESS views because source code may not have been saved for them and source regeneration (reverse engineering) might be required before they can be recreated on the target platform.

² Endianism refers to the way in which data is stored and defines how bytes are addressed in multi-byte data types. Tru64 UNIX is least significant first (little-endian) while HP-UX 11i is most significant first (big-endian). A database that tries to interpret multi-byte binary data on a machine that is of a different endianism will have different results. Therefore, moving binary data from Tru64 UNIX to HP-UX 11i is not allowed unless the move is endian neutral.

- Migrating SAS files from one release to another
It is almost always advisable that, if possible, applications be at the same level on both platforms before conducting a transition. How to move from one release to another, to conduct such upgrades, is standard knowledge available from SAS (see the link to the SAS support website in [For more information](#)). This activity would be required even if you were not migrating from one platform to another. However, there may be a few additional changes needed in SAS programs. For instance, there may be a few exceptional cases of certain PROCs where an option has changed or been discontinued. You can set up automated conversion routines or scripts to detect and change code in these cases, which would be specific to your site.
- Moving or converting files that are not SAS files
Moving third-party (non-SAS) files and host scripts from one system to another is not detailed in this document. However, it is likely that some UNIX scripts will require changes to file system and directory names. These changes can be automated with tools similar to those used to convert SAS programs. Information on processes and tools can also be found in the Planning module of the Platform Infrastructure Transition Module (see [For more information](#) for the link to the Tru64 UNIX to HP-UX 11i transition website).
- Converting Base SAS programs (DATA step, procedures, macros, and so on)
This should be minimal because of SAS language portability. Changes are usually limited to LIBNAME statements. This can be minimized by using directory naming conventions. That is, make project directory names the same on the HP-UX 11i environment as they are on the Tru64 UNIX environment.
- Converting SAS product applications
An example is a web-based application, where the user interface is an HTML browser window. You must research the requirements of any third-party software used in the application.
- Changing programs to maintain access to databases and data sources (remote or local) outside SAS
All SAS/ACCESS views to third-party relational databases must be converted, wherever they are used. Views cannot be migrated; they must be recreated from source code. Most of the access descriptor and access views may have been created through the SAS/ACCESS window. This means the source code must be regenerated from the Tru64 UNIX platform views in order to recreate the views on the HP-UX 11i platform. SQL pass-through code should not require any changes.
- Establishing file permissions for conversion routines
Automated tools used to convert SAS files within system and project directories require write access to the entire directory and all subdirectories on the target environment. Note that the tools might be set up to allow full or selective recursive processing of subdirectories. Conventions for setting file protections should be discussed and considered. It might be necessary to run the conversion tools under an account with access privileges to these directories or to have each directory owner run the conversion tools under the owner's personal account.

- Migrating very large data sets or catalogs

Migrating large SAS data sets and catalogs (greater than 1 gigabyte) can pose obvious challenges in managing both time and space constraints. Just considering space requirements, after you run a CPOR/CIMPORT conversion, the unconverted SAS files, converted SAS files, and transport file itself all consume disk space³. This can be minimized by using the new MIGRATE procedure and having the source (Tru64 UNIX) files NFS mounted on the HP-UX 11i system. See [Suggested migration method – PROC MIGRATE](#) later in this document. Using this method, HP has measured 3 to 5 megabytes per second for the PROC MIGRATE migration of non-catalog member types over a 100 megabit per second LAN.
- Tracking migrated entities and verification of converted SAS files and programs

Experience has demonstrated the importance of maintaining a central log of all SAS entities as they are transitioned to the new environment. This is where the organizational model of **system** level, **project** level, and **subproject** level can help. (See the section [Organizational model for SAS data and programs](#)). Identify and migrate all system level SAS files as the first step, then all SAS files together for a project or subproject. It has been suggested that a good way to document the conversion would be to take CONTENTS procedure snapshots of the SAS data sets before and after the conversion. See the discussion below about PROC MIGRATE ([Suggested migration method – PROC MIGRATE](#)) and validation tools ([Validation](#)). The validation tools compare PROC CONTENTS output before and after the migration.
- Establishing a well-defined backup process, supported by tools

Experience has shown the importance of backing up the originating platform, either prior to or as part of the migration process. The ideal solution would be to maintain the original version of the converted programs online until the converted versions have been verified. In the event that the directory is purged before verification is obtained, having the recourse of the backup version could prove vital. Also, some thought should be given to the recovery of data should it be needed after the Tru64 UNIX environment is no longer available.
- Training users on differences

The user training should be minimal because of the similarities between Tru64 UNIX and HP-UX 11i. Training can therefore focus on what is different, either from a UNIX view or a business process change during the migration project. Some suggested training focus areas might be:

 - SAS 8.2 versus SAS 9 interfaces

Check the following website for “What’s New in SAS 9”:
<http://support.sas.com/documentation/onlinedoc/index.html>
 - Tru64 UNIX versus HP-UX 11i interfaces.

Check the following website for training:
<http://h30097.www3.hp.com/training.html>
- Recognizing constraints on migration scheduling

Consider time constraints on migration scheduling. After the system level is migrated, each project can decide on its migration timeline and other specific issues. Use test migrations of known amounts of data to estimate how long migration takes for your environment.

³ A substantial part of which is needed only temporarily.

- Stabilizing the software environment during migration

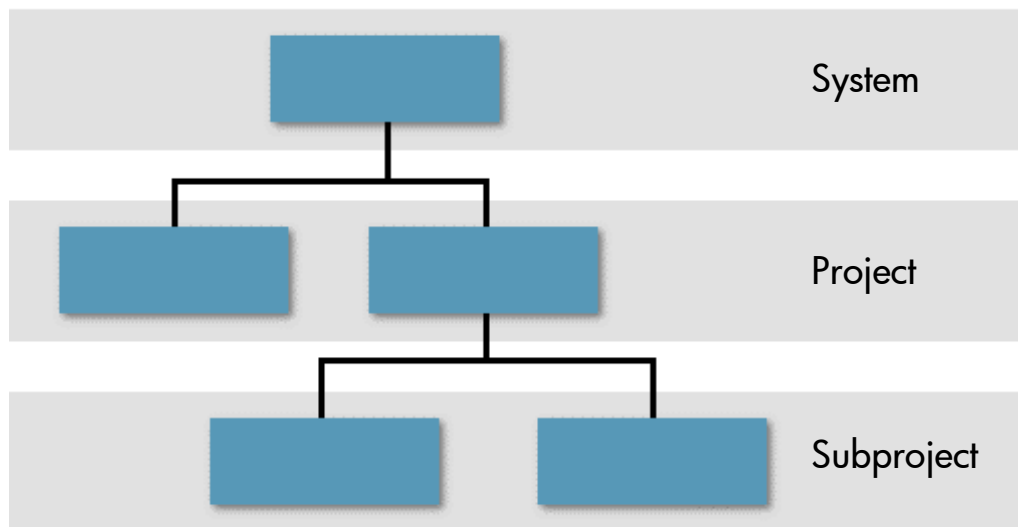
Before the migration project begins, decide which versions of Tru64 UNIX, HP-UX 11i, and SAS to use on each platform. Keep the versions stable until the migration is completed. Tru64 UNIX V5.1A, V5.1B, and V5.1B-1 are good with SAS 8.2. HP-UX 11i v2 with SAS 9.1.2 (or later) is good for the target environment. The Tru64 UNIX environment could also be SAS 9.1.2 or 9.1.3, but from the point of view of this document, those versions are not required for migration.

Organizational model for SAS data and programs

An important aspect of production SAS programs and data sets is that they can be organized into a multi-tiered hierarchy that is modeled after the business organization and maps to the physical hierarchy of the directories in which they reside (see Figure 1). This structure serves to utilize common programs and common SAS tables across the entire business. If you currently have multiple projects with supporting system-wide structure, then everything can be migrated as one project.

The top level of the hierarchy is designated as the **system** level. It contains programs and data that are used across multiple projects. System-level entities (such as macro libraries) are generally stored at the top level of a directory structure which is dedicated to the projects it serves. The next level of the hierarchy is the **project** level, containing programs and data that are specific to a particular business project. The lowest levels of the hierarchy are the **subprojects** for particular major projects. Subprojects might not exist in actual practice. They are created as a discussion abstraction for this paper.

Figure 1. Organizational model



One of the reasons this structure is important to the migration effort is that it establishes the UNIX directory (which would normally map to a single SAS library) as the natural source of a conversion run. Another reason is that it imposes certain requirements on the migration, as follows:

1. Migration of system-level data and programs must happen first. If the system-level data is not migrated first, then when a particular project migrates, the system-level data it needs (in other words, a dependency) might not be available.

Converted and unconverted system-level directories must co-exist. Because projects may migrate at different times⁴, parallel versions of the system-level directories must exist, one for use by the converted projects on the HP Integrity platform and the other for the unconverted projects on the Tru64 UNIX platform. This condition must persist until all projects complete migration and validation, at which time the unconverted system-level directories can be removed. While system-level directories exist on both environments, special attention must be given to keeping them synchronized on both environments as required.

2. If possible, all SAS files under a project should be migrated at the same time as the project. This is where the subproject hierarchy can be helpful. The main project can be migrated, then each subproject can be migrated for each different `LIBNAME` directory. Ideally, completing the entire project and migrating all subprojects simultaneously makes the migration the cleanest and easiest and avoids synchronization issues. However, this might not be possible with large and complex projects.

Transferring files to a target environment

You can use the following methods to make files available for access.⁵ NFS is the preferred method because it avoids the extra disk space used for the duplicate copies of the data when `rsh` or FTP is used.

- NFS (Network File Services)

Use NFS to mount a remote file system or directory on a target environment and then read or write the files as though they were local. This has the advantage that the remote file system (located on the Tru64 UNIX platform) can be mounted on the HP Integrity platform running HP-UX 11i (say on the `/net` mount point) and used as the input *source* for the examples to follow. The input files do not need to be copied and can be read directly from the Tru64 UNIX system. The output destination (referred to as *target* later in the document) can (and should) be a local file system on the HP-UX 11i environment. The storage on the HP Integrity environment is only required for the migrated SAS files. NFS mounting of the remote (Tru64 UNIX) file system is the preferred method to use during the migration process.

- `rsh` UNIX command

```
rsh [-pr] source destination
```

For a basic connection, the user's home directory on the remote host must contain a `$HOME/.rhosts` file that lists the local host name and user name pairs on a line. The `$HOME/.rhosts` file must be owned by the remote user and have permissions set to 600 (read and write by owner only). Read the UNIX documentation for a secure connection environment. If you have any security concerns with the `.rhost`, then adapt for your needs.

⁴ Not all projects will find simultaneous migration possible or convenient.

⁵ Throughout this document, the use of the UNIX root account is minimized and should not be used for the actual SAS migration work. Standard user accounts should be established for the migration projects per your company's policies.

- FTP (File Transfer Protocol)

Use FTP services to copy a file to a specific target environment. Copy text files (for example .sas programs) in **ASCII** transfer mode. Copy all others (data sets, catalogs, transport files, and so on) in **binary** transfer mode.

Best Practice: Using the NFS method will avoid extra disk space that would be required for duplicate copies of the data if `rsh` or FTP were used.

Use the following link for documentation on the use of NFS, `rsh`, and FTP:

<http://docs.hp.com/>

Suggested migration method – PROC MIGRATE

Starting with the SAS 9.1 release, the migration process is simplified by the availability of a new `MIGRATE` procedure and validation tools. `PROC MIGRATE` not only copies or moves a library to the target platform; it also changes the members' file format to that of the most recent release of SAS. In addition, `PROC MIGRATE` provides some new functionality that the `COPY`, `CPORT`, `CIMPORT`, and `CATALOG` procedures do not. The `MIGRATE` procedure migrates members in a SAS data library to SAS 9.1 and completely transforms the members to the native format of the SAS HP-UX 11i environment. `PROC MIGRATE` can migrate SAS V6, V7, or V8 data structures to SAS 9, but it does not support V6 tape format. V6 tape formats should be read to disk file systems and then migrated from the file system environment.

System or project migration

Figure 2. How SAS file member types will be migrated in one step

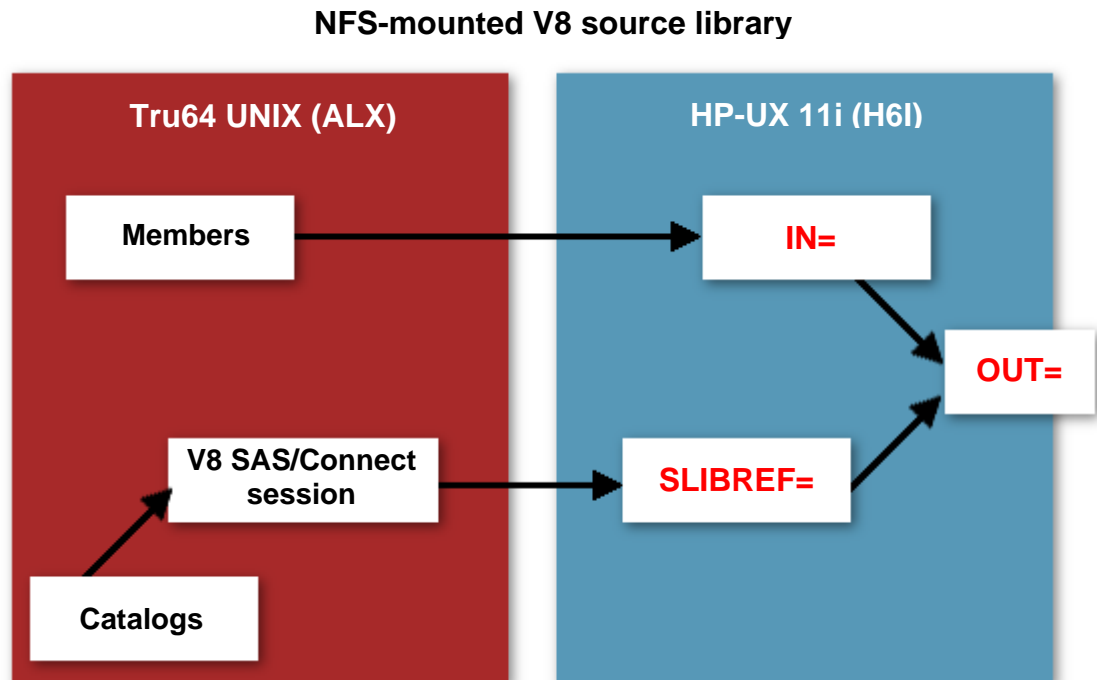


Figure 2 depicts how SAS file member types will be migrated in one step. The ALX and H6I in the diagram are the three character codes SAS uses to uniquely identify the platform. NFS is the preferred method to make files available for access. The data type members (non-catalogs) will be migrated directly thru the NFS mount point (IN=). The catalogs will be migrated using SAS/Connect and the SLIBREF option of PROC MIGRATE.

Under UNIX platforms, when you migrate Tru64 UNIX SAS catalogs across machines to an HP-UX 11i target library, it is strongly suggested that you use RLS (Remote Library Services) together with PROC MIGRATE. When catalogs exist in the library, you must specify a SAS 9.1 server for the source library and a SAS 8 server for the SLIBREF= option. This means you must have access to a SAS 8 server. The specifics depend on hardware and setup at your site, but the general method is as follows.

1. On a Version 8 SAS/CONNECT Tru64 UNIX server, assign a libref to the project source library:

```
libname v8srv 'source-library-pathname' ;
```

2. In the HP-UX 11i session, associate the in libref to the nfs mount point on HP-UX 11i of the Tru64 UNIX project data, matching the project location of step 1:

```
libname source /nfs/data/project ;
```

3. In the HP-UX 11i session, associate a second libref to the Version 8 RLS server (SERVER=V8-SESSION-ID):

```
libname v8srv server=V8-session-ID ;
```

4. Define the target library (where the migrated members will exist on the HP-UX 11i system):

```
libname target 'target-library-pathname' ;
```

5. You're ready to PROC MIGRATE:

```
proc migrate in=source out=target slibref=v8srv ; run ;
```

See the SAS online documentation for RLS, which is a feature of SAS/SHARE and SAS/CONNECT software. If you don't have access to SAS/CONNECT software, contact the Customer Interaction Center (phone 1-800-727-0025; e-mail CIcenter@sas.com).

If you have SAS V6 files on Tru64 UNIX that need to be migrated, see [Appendix A: SAS V6 files migration](#).

PROC MIGRATE syntax

```
LIBNAME source <engine> 'path' ;  
LIBNAME target BASE 'path' ;  
PROC MIGRATE IN=source OUT=target  
  <MOVE> <BUFSIZE=#> <KEEPNODUPKEY> <SLIBREF=slibref> ;  
RUN ;
```

You must define a LIBNAME statement to indicate the source library—the library that is migrated. This is on (or from) the Tru64 UNIX environment. Note that an engine name is optional here. If you have a location that contains both V6 and V8 files, you should indicate which engine's members you want to migrate. Without the engine name, the V8 files are migrated by default. PROC CONTENTS will tell you the SAS engine name that created the member.

The second LIBNAME statement indicates the target library—the location where you want your new SAS 9.1 library to be located on the HP-UX 11i environment. Make sure that the target location is an empty location, to contain only the SAS 9.1 files. If a file already exists in the target library that has the same name and member type as one in the source library, the file is not migrated. You are migrating forward to the new release; an engine name of BASE is specified.

With the source and target LIBNAMEs defined, the simplest case of a migration only requires the following:

```
PROC MIGRATE IN=source OUT=target ; RUN ;
```

The PROC MIGRATE command has four options:

- MOVE deletes any members migrated without errors from the source library. Using the MOVE option significantly limits the information that can be produced by the PROC MIGRATE validation tools (see the [Validation](#) section). It is preferable to wait and delete the source library on the Tru64 UNIX later—after a successful validation has confirmed that the library has been migrated successfully.
- BUFSIZE specifies the buffer page size to use when creating the members in the target library. If you do not specify BUFSIZE, the default is the original buffer page size used to create the source member.
- KEEPNOUPKEY specifies to retain the NODUPKEY assertion. Because a data set created before SAS 9 may have the assertion set erroneously, the default is to migrate without the NODUPKEY sort assertion.
- SLIBREF is an option to aid in the migration of catalogs. In the suggested migration, this option is used to migrate the catalogs from Tru64 UNIX to HP-UX 11i using SAS/Connect.

Source library member types

ACCESS views, catalogs, data sets, DATA step views, item stores, MDDBs, programs (.sas files), and SQL views are all the member types that you can store in a SAS library. All of these member types migrate except the program files. To migrate a program file, use the original source to create it while running SAS 9.1 on the target platform.

There are a few exceptions to the member types that can migrate:

- PROC MIGRATE internally calls PROC CPORT and PROC CIMPORT to migrate catalogs. (You might notice CPORT and CIMPORT notes written to the SAS log during migration.) Therefore, migration of catalogs falls under the rules of CPORT and CIMPORT. As such, a catalog cannot be migrated directly from a sequential library (for example, tape library). You can migrate from a sequential library by first restoring the tape library to disk and then migrating from the disk copy.
- DATA step views gained the ability to store source with the view in Version 8. It is the default to store the source. Any DATA step views with the source stored are migrated to the SAS 9.1 library. The first time the view in the target library is accessed, it is regenerated and the view is stored. If a view does not contain the source, it cannot be migrated.
- V7 MDDBs are not migrated. This should not be an issue because V7 was not widely used among the Tru64 UNIX AlphaServer community.

Data attributes

Data set attributes migrate along with the data set. Indexes, integrity constraints, encryption, compression, generations, audit trail, data representation, and encoding are migrated. No other facility in SAS can migrate an audit trail, because other conversion routines clean up deleted observations. PROC MIGRATE retains these deleted observations (still deleted) so that observation numbers match with the migrated audit trail. The created/last modified dates and times are also retained.

Two attributes that might change during a migration are data representation and encoding. These attributes are surfaced in SAS 9 PROC CONTENTS "Attribute" information. The data representation value reports the operating system on which the data sets were created. This value changes from Tru64 UNIX to HP-UX 11i (see tables below). Note that data sets created prior to Version 8 do not store this value, so SAS 9 PROC CONTENTS reports this value as "default". Encoding is a new SAS 9 data attribute that reports the internal international data format in which the data is stored. This value can change if the Tru64 UNIX data files were created in a different region from the target HP-UX 11i environment (for example, Tru64 UNIX in Eastern Europe, HP-UX 11i in Asia Pacific.).

Table 1. Expected data representation and encoding values (SAS 9.1 PROC CONTENTS, North America) for Tru64 UNIX

| | V6 | V8.2 | SAS 9.1 |
|----------------------------|---------|-------------|---|
| Data representation | default | ALPHA_TRU64 | ALPHA_TRU64 |
| Encoding | default | default | Latin 1 Western (ISO) (this value may change) |

Table 2. Expected data representation and encoding Values (SAS 9.1 PROC CONTENTS, North America) for HP-UX 11i

| | V6 | V8.2 | SAS 9.1 |
|----------------------------|---------|---|---|
| Data representation | default | HP_UX_64, RS_6000_AIX_64, SOLARIS_64, HP_IA64 | HP_UX_64, RS_6000_AIX_64, SOLARIS_64, HP_IA64 |
| Encoding | default | default | Latin 1 Western (ISO) (this value may change) |

Validation

It is important to know whether the migration was successful. Validation is an essential part of a successful migration. SAS 9 offers new tools to validate the migration of SAS data libraries. You can run PROC MIGRATE with validation tools that are available to download from SAS Institute:

<http://support.sas.com/rnd/migration/resources/procmigrate/validtools.html>

Copy `migrate_macros.sas` and `migrate_template.sas` to a migration user directory on your HP-UX 11i environment. These are SAS programs, neither of which is shipped with the SAS software. The first program, `migrate_macros.sas`, contains 13 SAS macros designed to output validation information in the SAS Output Window. This file should not be modified. The second program, `migrate_template.sas`, uses the macros in the first program to simplify the migration and validation process. `migrate_template.sas` can be used as is, but you are free to modify it to better suit your individual needs.

Interactive session

Follow these steps to use the validation tools from an interactive SAS session:

1. To compile the macros, copy and paste the text in the `migrate_macros.sas` file into an interactive SAS session and submit it.
2. Copy and paste the test in the `migrate_template.sas` file into an interactive SAS session, define the source library, target library, and library to contain the ODS (Output Delivery System) output data sets. The ODS output area is used by the validation tools during the validation process so that the source and target libraries do not contain files used specifically for the validation process.
3. Modify the PROC MIGRATE step in the `migrate_template.sas` file with the appropriate options.
4. Submit the `migrate_template.sas` file.

Example `migrate_template.sas` file without many comments:

```
OPTIONS NOCENTER FORMDLIM='- ' ;
LIBNAME source '/net/project1' ; *NFS mount to Tru64 UNIX library
LIBNAME target '/data/project1' ; * empty HP-UX target library
LIBNAME ods '/saswrk/ods' ; * path used by macros
%before ;
PROC MIGRATE in=source out=target ; RUN;
%after ;
%checkem ;
RUN;
```

The `%before` macro inventories the contents of the source library before migration. The `%after` macro inventories the target library and produces the first validation output. This report lists the results of the migration sorted by member type of the migrated library.

The `%checkem` macro uses information collected during the `%before` macro to drive the validation of individual members. By default, `%checkem` only reports differences. "Host Created" and "Release Created" are expected differences.

For the latest how-to steps to execute an interactive session migration, check the following link:

<http://support.sas.com/rnd/migration/resources/procmigrate/validtools.html>

Batch session

The above interactive steps can be modified to do the work in a *batch* or non-interactive session.

1. Make a copy of the `migrate_template.sas` file and call it `migrate_batch.sas`.
2. Edit `migrate_batch.sas` to make the macros available by including the following line before the `libname` statements:

```
%INCLUDE 'migrate_macros.sas' ;
```
3. Define the source library, target library, and library to contain the ODS output data sets. Make any other modifications to this file that you chose.
4. Run the program with `sas migrate_batch.sas`.
5. Examine the output (`.log` and `.lst` files) as you would interactively.

Sample output from validation tools

The first report produced in the SAS Output Window is a comparison of the members in the source library before migration to the members in the target library after migration. In this example, four members were present in the source library, but only three were migrated to the target library (result is OK). One member, PROGRAM has a result of not migrated. In this situation, it is recommended that you check the SAS Log Window or log file for PROC MIGRATE NOTES, WARNINGS, and ERRORS for clues. However, since PROC MIGRATE does not migrate program files, this is an expected difference.

Output 1: "What files were migrated?"

Target library after migrate (sorted by MemType)

| Obs | name | memType | result |
|-----|----------|---------|--------------|
| 1 | DATA_SET | AUDIT | OK |
| 2 | FORMATS | CATALOG | OK |
| 3 | DATA_SET | DATA | OK |
| 4 | PROGRAM | PROGRAM | not migrated |

Output 2: "Were my files migrated correctly?"

```
MY_DATA_SET
Number 7 of 9 data sets in source library
PROC COMPARE of data
Number 3 of 3 reports for this data set
NOTE: No unequal values were found.
```

Summary

Proper planning using some of the ideas and considerations presented in this paper can help make the migration of the SAS environment from Tru64 UNIX to HP-UX 11i for HP Integrity servers a straightforward process. Using PROC MIGRATE with validation tools can greatly increase the probability of success. The migrated SAS environment can be used in a timely manner and with confidence.

Acknowledgements

The author would like to acknowledge the SAS migration developer and tester for review and support of this document.

Appendix A: SAS V6 files migration

The migration of SAS V6 files requires additional steps. The SAS V6 Tru64 UNIX files have *.s*04 file extensions. If a V6 library is assigned in an HP-UX 11i SAS session which references V6 Tru64 UNIX SAS files, the library will appear empty, even if members actually exist. The HP-UX 11i SAS session is searching for files with a .s*02 extension. Since no files with that extension exist in the directory, PROC DATASET returns WARNING: No matching files in directory. Due to the requirements of Remote Library Services and PROC MIGRATE, you must migrate through an SAS 9 server. The example below uses two different servers to migrate V6 members. The solution is a “nested rsubmit”:

```
/* signon to remote SAS 9 HP-UX 11i session */
rsubmit sas9serv ;
    /* signon to remote V8 Tru64 UNIX session */
    rsubmit v8serv ;
        libname lib1 'path to source library' ;
    endrsubmit ;
libname lib1 server=v8serv ;
endrsubmit ;

libname lib1 server=sas9serv ;
libname target 'path to target library' ; /* on HP-UX */
```

If the source library contains V6 catalogs, submit:

```
/* signon to another remote V8 Tru64 UNIX session separate
from the session running in the nested rsubmit */
rsubmit v8new ;
    libname catlib V6 'path to source library' ;
    endrsubmit ;
libname catlib server=v8new ;
proc migrate in=lib1 out=target slibref=catlib ;
run ;
```

Appendix B: Traditional conversion methods

For completeness, this appendix discusses several traditional conversion methods. These traditional methods were not specifically designed for migration to a new version of SAS. The new PROC MIGRATE (discussed above) is the recommended method to migrate.

PROC COPY

The COPY procedure copies one or more SAS files from a SAS data library.

PROC COPY ignores explicit concatenations with catalogs. In fact, if you use PROC COPY to bring forward a catalog to a new version, only the shell of the catalog is changed to the new version and the internal entries are left in their old format until they are further processed.

Conversely, PROC MIGRATE completely transforms all the entries of a library to the new version. This is what is meant by migration of a SAS library—the library members become native HP-UX 11i.

PROC CPORT and CIMPORT

The CPORT procedure writes individual SAS files or entire SAS libraries to sequential file formats (transport files). Use PROC CPORT and PROC CIMPORT together to move files from one environment to another. The transport format that PROC CPORT writes is the same for all environments and for many releases of SAS. In PROC CPORT, **export** means to put an SAS data library, catalog, or data set into transport format. PROC CIMPORT restores (**imports**) the transport file to its original form as an SAS data library, catalog, or data set. Only PROC CIMPORT can read the transport files that PROC CPORT creates.

The disadvantage of the CPORT and CIMPORT method is the transport file format encoding can lose precision of the floating point numbers.

General file transport process

To export and import files, follow these steps:

1. Use PROC CPORT to export the SAS files on your Tru64 UNIX platform that you want to transport to your HP Integrity server.
2. Copy the transport files to the new HP Integrity server using either communications software or a magnetic medium. Be sure the transfer treats the transport file as a binary file. (See the section [Transferring files to a target environment](#).)
3. Use PROC CIMPORT to translate the transported files into the format appropriate for the HP Integrity server environment.

Example: Exporting a single SAS data set

This example shows how to use PROC CPORT to export a single SAS data set.

```
LIBNAME source 'SAS-data-library' ;
FILENAME tranfile 'transport-file' ;
PROC CPORT DATA=source.onedata FILE=tranfile ;
RUN;
```

Example: Exporting multiple catalogs

This example shows how to use PROC CPORT to export entries from all of the SAS catalogs in the SAS data library you specify.

```
LIBNAME source 'SAS-project-library' ; * on Tru64 UNIX
FILENAME tranfile 'transport-file' ;
PROC CPORT LIBRARY=source FILE=tranfile MEMTYPE=CATALOG ;
RUN;
```

The log file will list the catalogs and the entries in each catalog that have been transported.

Example: Importing multiple catalogs

This example shows how to use PROC CIMPORT to import the SAS catalogs done in the above example.

```
LIBNAME target 'SAS-project-library' ; * on HP-UX
FILENAME tranfile 'transport-file' ;
PROC CIMPORT LIBRARY=target INFILE=tranfile ;
RUN;
```

The log file will list the catalogs and the entries in each catalog that have been transported.

PROC UPLOAD/DOWNLOAD

Using SAS/CONNECT, you can transfer files between a client session and a server session by using the UPLOAD procedure. The DOWNLOAD procedure copies files that are stored on the server to the client.

For more information

For additional information and assistance, contact:

transition.modules@hp.com

For more information about the Alpha RetainTrust program and other sources of support, go to:

<http://www.hp.com/go/alpha-retaintrust/>

For technical and planning information about transition subjects, go to:

<http://www.hp.com/go/tru64transition/>

<http://www.hp.com/go/transition-modules/>

For information regarding the Tru64 UNIX roadmap, go to:

http://h30097.www3.hp.com/unix/downloads/tru64_unix_roadmap.ppt

For the SAS Migration Community:

<http://support.sas.com/rnd/migration/index.html>

For SAS PROC MIGRATE validation tools:

<http://support.sas.com/rnd/migration/resources/procmigrate/validtools.html>

For the white paper *When to migrate your SAS environment from Tru64 UNIX to HP-UX Integrity servers* (and the SAS HP partner directory):

<http://www.sas.com/partners/directory/hp/papers.html>

For technical information on UNIX and HP-UX 11i, go to:

<http://docs.hp.com>

© 2005 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Intel and Itanium are registered trademarks of Intel Corporation in the U.S. and other countries and are used under license. Oracle is a registered U.S. trademark of Oracle Corporation, Redwood City, California. UNIX is a trademark of The Open Group in the U.S. and/or other countries. All other product names mentioned herein may be the trademarks of their respective companies.

5982-7332EN, 1/2005



i n v e n t