Migration of SAS® Software From VMS® to Windows® NT®: A Real Life Story

Paul Gilbert & Steve Light at DataCeutics, Inc.,
John Scott Grainger at ClinTrials Research

Introduction

At ClinTrials Research, Inc. clinical data is maintained in Ingress and Oracle data bases and extracted to SAS for reporting and analysis. An Open VMS SAS mainframe-based configuration is used on DEC VAX and AXP machines. In the VMS SAS programming environment data, programs, output, and work libraries are resident on the VAX cluster and processing is performed on the DEC machines. With approximately 30 active programmers, the burden on the VAX system generated by SAS programming is significant. To reduce the processing load on the VAX it was decided to move SAS processing off the mainframe while retaining central storage of programs and data on the VAX cluster.

The target environment that was selected for the migration was a network of Intel-based PCs running Windows NT 4.0 over a 100 MB LAN.

This paper describes the specific goals of the project to move SAS processing into NT, the selection of a client/server configuration, and the selection and implementation of tools supporting the distributed SAS environment.

Goals

The primary goals of the project were as follows:

1) Implement a Windows-based SAS programming environment, with desktop processors running SAS locally to maximize processing distribution.

2) Maintain or reduce the cost of licensing the SAS System software.

3) Provide the ability to easily, quickly and cheaply add more computing power to the SAS programming environment.

4) Distribute processing across multiple locations to level the work load. This was addressed by including additional Pentium Pro dual processor servers dedicated to running SAS programs.

5) Facilitate code and data management through storage of all code and data in shared, centralized directory structures.

6) Allow for a configuration which is scaleable for a small or large work group.

7) To maximize flexibility during the transition from VMS to NT, allow programmers to edit and run programs from either NT or VMS, using one set of data and programs.

Migration Strategy

The migration to NT was planned to occur in stages starting with implementation of the full NT infrastructure, including installation of a 100 MB network, NT workstations, PathWorks drives and software configuration. Once the NT environment was in place a single project involving a small number of SAS programmers was implemented in NT. At the time of completion of this paper the initial project was underway, with a review planned following project completion. Any problems encountered in the initial project will be addressed before further implementation of the NT programming environment. Results of the initial project will be reported in the conference presentation of this paper.

Following the initial project migration from VMS to NT will take place on a project-by-project basis. Projects already underway in VMS may be retained in VMS or moved over to NT, based on the expected duration or project-specific requirements. New projects will be implemented in NT.

Selecting the distributed computing configuration

A number of client/server configurations were considered before final selection of a two-tiered model with work group process servers, shown in Figure 1.

In this configuration SAS data libraries, catalogs, and associated code and output are stored in Windows format on PathWorks drives on the VAX cluster. PathWorks drives can be accessed from either NT or VMS, allowing programmers to work with the data and programs from either platform. SAS/CONNECT® Software Remote Library Services is used to access the NT format data from VMS. Programs can be edited and run, output printed, etc. from either VMS or NT. SAS/ACCESS® is used to access Oracle and Ingress data.

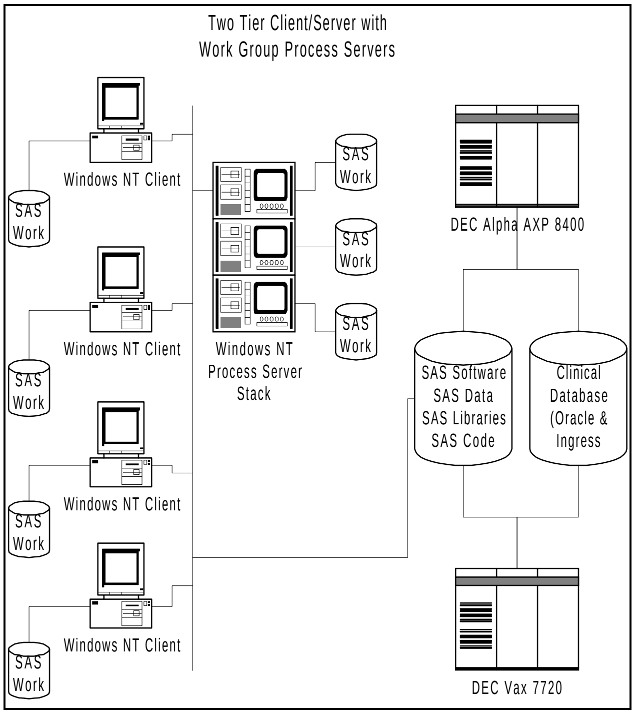

Figure 1

Specifications of the selected configuration include:

Client Workstation

- 200 MHz Pentium Pro, single processor
- 128 MB RAM
- Microsoft Windows NT Workstation 4.x
- SAS Software
- SAS Work library on dedicated 2 GB SCSI drive

Workgroup Process Server

- 200 MHz Pentium Pro, dual processor
- 128 MB RAM
- Microsoft Windows NT Server 4.x
- SAS Software including SAS/Connect and SAS/Access
- SAS Work library on 10 GB RAID level 3 drive

VMS Server Specifications

- DEC Alpha AXP 8400 and DEC VAX 7720
- Open VMS
- SAS Software including SAS/Access and SAS/Connect
- PathWorks drives where all SAS data, programs and output are stored

The distinctive feature of this configuration is the use of workgroup process servers. These servers were introduced into a conventional two-tiered configuration to provide a platform for execution of resource-intensive programs that may exceed the capabilities of the standard workstation. The process servers have dual processors, increased memory and a large amount of disk space available for SAS work libraries.

SAS programs are submitted for remote execution on the process servers using a third party scheduling tool, Seagate Ashwin. Ashwin provides the capability to essentially submit programs in manner analogous to a ‘batch queue’ in VMS. Ashwin supports scheduling of programs for delayed execution, or repeated execution according to a calendar-based schedule. Ashwin automatically balances the load of executing programs across the pool of process servers.

Extracting Ingress and Oracle Data into SAS Datasets

A combination of SAS/Access and SAS/Connect Remote Library Services is used to extract data. SAS/Access for Oracle and and Ingress allow use of pass-through SQL directly to Oracle via SQL*NET. The resulting datasets are stored on the VAX PathWorks drives in Windows format.

Accessing data and programs from NT and VMS

SAS data and programs are centrally located on a set of VAX-based PathWorks drives. These drives can be accessed from Windows NT or from VMS, allowing use of a single set of data and programs from the two platforms. This provides for maximum flexibility during the period of transition from VMS to NT. Either environment may be used to edit or execute programs, and to print output.

The SAS datasets are stored in Windows format. Using the SAS datasets in Windows format from

VMS requires use of SAS/Connect Remote Library Services. To allow use of individual programs from either NT or VMS, SAS data libraries are allocated via a standard SAS macro that assigns all libname statements. *Programmers do not write libname statements in their programs*.

For programs executed from NT the data access macro determines the current operating environment and assigns libname statements with standard syntax, there are no special requirements.

If the program is being executed from VMS an active SAS/Connect session and a current remote system option are required. The 'remote' option is assigned as part of initiation of a SAS/Connect session and holds the name or address of the remote session being connected to. If 'remote' has no current value the data access macro is terminated and the user is notified that a SAS/Connect session must be active to access data from VMS. If the 'remote' option has a valid value, libname statements are assigned to use Remote Library Services using the 'server=<remote>' qualifier.

Remote Library Services does not allow cross-platform access of SAS Catalogs. This is an issue at CTR primarily for SAS format libraries. To support execution of SAS programs in both NT and VMS it has been necessary to maintain format libraries in both Windows and VMS formats. Maintenance and synchronization of the two copies of the catalogs is an ongoing issue.

Tools supporting the NT SAS programming environment

Whereas VMS provides adequate support for editing, batch job execution and scheduling, Windows NT does not meet the requirements for a large SAS programming shop. Part of the move to NT required selection of tools for use by SAS programmers.

SAS Program Editors

A text editor for SAS program development is the most basic requirement for the NT programming environment. The text editor supplied as part of Windows NT, Notepad, is not adequate for use as a SAS program editor. Fortunately, there is a wealth of third party program editors available for NT. Several options were considered including:

- the SAS AWS environment, formerly referred to as Display Manager
- the SAS System Viewer
- full featured 'professional' programmer's editors such as CodeWright
- low priced text editors, generally shareware products

The SAS AWS was not considered to be the best program development environment for several reasons, including limited editing features and a restriction to a single program editor window. While the AWS can be a convenient way to code and run 'quickie' programs, it was felt that a more robust and flexible editor was desirable.

The SAS System Viewer, supplied as a no cost standalone application, can be used as a text editor. This was not considered to be adequate due to a very limited number of editing features.

'Professional' programmer's editors were found to be very powerful and loaded with features, but were expensive and somewhat complex to work with. Many of the included features were not of interest to SAS programmers and only made them more difficult to use.

Many low priced text editors are available for use in NT. These editors are generally very powerful and easy to use, providing many editing features which were not available at all in VMS. Significant editing features available in several of these editors include:

- unlimited number of open edit windows
- syntax highlighting with color
- drag and drop from the Windows Explorer
- auto-indent of code
- bookmarking of file positions
- file comparison
- multiple undo/redo of last edit action
- code template storage/insertion

Three editors were selected for evaluation with the goal of identifying a single standard editor for use by all programmers. The editors evaluated included: WinEdit (Wilson Windowware), UltraEdit (IDMComp) and KEDIT (Mansfield Software). At the time of completion of this paper a final selection had not been made.

Batch Processing

Production jobs at CTR require execution of sequences of SAS programs in a controlled order. These jobs can become quite large, and sometimes require several days for execution. Within the VMS environment, scheduling and control of production jobs has been handled by use of DCL scripts and batch execution queues.

Our strategy for handling the large production batch jobs in NT was through use of the workgroup process servers. Programmers submit production jobs to execute on the servers. This keeps their workstations clear and provides greater, dedicated resources for processing the jobs. One of the goals of the migration was to introduce a mechanism for balancing the server processing load which could be easily extended as additional servers were added to the system.

NT 4.0 does not provide powerful scripting tools or scheduling tools such as the programmers were accustomed to using in VMS. Additionally, there is no equivalent to VMS batch queues which would allow programmers to submit large programs to run on remote processors. These are significant gaps in the NT environment which require third party solutions.

As part of the initial project in NT Seagate Ashwin is being evaluated as a tool to provide job scheduling, batch processing and process load balancing.

Ashwin enables users to submit (or schedule) individual programs or groups of programs for batch execution on the workgroup process servers. It provides scheduling facilities that allow users to define dependencies between jobs or schedules to control the sequence of program execution, and can provide email notification on success or failure of jobs. This functionality approximately substitutes for the batch queue facility and DCL programming done in VMS.

Ashwin is capable of managing executable programs such as SAS, as well as NT command files (BAT or CMD files). Command files are useful for automating time consuming tasks such as copying files or printing large volumes of output. Using Ashwin features command files can be set to run conditionally based on the success or failure of a production job. For example, a command file which prints all output from a large production job can be set to execute only upon successful completion of all components.

Ashwin also provides load balancing by monitoring a defined set of networked machines and directing scheduled jobs to execute on the first machine available. Load levels allowed on each machine can be adjusted as needed and servers can be easily added or removed from the list of available machines. This provides an easily scaleable batch processing configuration, meeting one of the primary goals of the project.

Printing SAS Output

SAS generates text output as unformatted text files. Printing SAS output for production work at CTR requires standardization of font, font size, page orientation, margins, etc. In cases where only a few files need to be printed it is a relatively easy matter to load each file into an editor or word processor, assign the formatting information and print from that application.

However, the large number of files generated at CTR makes this method impractical. A batch method of printing files with accurate formatting information (that does not require a programmer to load each file into an interactive application) is required. NT does not provide good support for this kind of 'direct' printing, leading us once again to look for third party solutions.

The features of the ideal printing solution for SAS text output in NT might be as follows:

- Allows command line or interactive execution.
- Uses an INI file to assign the target printer and page formatting information. This would allow one to store separate INI files with different page formats.
- Capable of printing multiple files as a single job in an order specified by the user.
- Capable of attaching network resources (printers).
- Allows interactive selection of files to be printed in an Explorer-like interface.
- Capable of reading a text file with an ordered list of files to print.
- Generates log of print history including files printed, date and time, target printer, INI file in use and success or failure of the print job.

Unfortunately, if such a utility exists we have not found it. At the time of completion of this paper a shareware utility, CodePrint for Windows 95 and NT (Micro Software Solutions) is being evaluated. This utility provides most of the desired functionality, but does not have the capability of reading a text file with an ordered list of files to print and does not generate log files.

SAS Code Management

Part of the initial project in NT included an evaluation of use of a code management system for

SAS programs. In an effort to gain greater control over the SAS program development process and to track project status Microsoft SourceSafe® was implemented to manage all SAS programs. SourceSafe stores source code in a central repository and tracks changes made to program files. This application provides a number of benefits to the SAS programming environment including

- control of write-access to SAS files
- tracking of modifications to files
- review file history
- revert to earlier file versions
- track which files are checked out for editing, current owner of file
- ability to label a project with a version number and to later restore entire project as a single operation

Conference Presentation

The conference presentation of this paper will discuss ongoing developments, results of our initial project and our experiences with moving more of the production environment from VMS into NT.

SAS, SAS/ACCESS and SAS/CONNECT are registered trademarks or trademarks of SAS/ Institute in the USA and other countries.

® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

Biographies

Paul Gilbert is Vice President of DataCeutics, Inc., a consulting group specializing in software solutions, system integration, programming and support in the areas of clinical data management and statistical reporting. His fifteen years of experience includes clinical data management, implementing and maintaining BBN/Clintrial, designing SAS based biostatistics reporting systems, managing SAS based NDA programming support, SAS/PH-Clinical implementation and CANDA integration.

Steve Light, a Project Manager at DataCeutics, Inc., is responsible for SAS report macro development and biostatistics/clinical SAS project management. His twelve years of experience with clinical information systems include SAS systems development and validation, NDA driven project management, SAS/PH-Clinical implementation, programming and clinical data management.

Paul and Steve can be reached at

DataCeutics, Inc.
244 High St
Suite 302
Pottstown, PA 19464

or email
gilbertp@dataceutics.com
lights@dataceutics.com

Scott Grainger is Vice President, Information Systems for ClinTrials Research, Inc., an international contract research services organization with approximately 1500 employees providing comprehensive preclinical and clinical research services including monitoring, data management, biostatistics, medical and regulatory to pharmaceutical, biotechnology and medical device clients. His 25 years of experience is in operations, systems management, application development, system deployment, and information technology planning in a variety of clinical environments.