Data Warehouse:

How to use SAS from Beginning to Almost Ending for Your Warehouse

Akbar Golmirzaie, University of Arkansas, Fayetteville, AR

January 7, 1998

Abstract

The University of Arkansas at Fayetteville first began exploring the development and implementation of data warehousing in November, 1995. In under two years, the Computing Services Department has moved from conceptual modeling, through prototyping, to a working, evolving production system for data storage and retrieval – all at a substantial savings in time, money and resources.

Paramount in this endeavor has been the use of SAS® software. From the system administration tools to those maintaining the secure server, the University of Arkansas development team found in a single software suite what previously had only been available as separate, much more expensive packages. In addition, the new SAS® ODBC drivers insure that end-users can directly retrieve, massage and report from the now-centralized data with any ODBC-compliant software package, including of course, SAS® itself.

This paper will present a review of the data warehouse implementation on the Fayetteville campus, as well as an outlook for the future.

Introduction

The Department of Computing Services provides computing facilities in support of instruction, research, and administration to University of Arkansas students, faculty, and staff. An important component of this mission is to develop comprehensive information systems and pursue reengineering of ongoing systems in support of university information needs.

Initially, this support came in the form of mainframe on-line and batch-reporting systems. Then PC, Macintosh® and networked desktop applications came on-line. These were and are still in use, a vital part of day-to-day functioning within individual departments. However, as computing needs of the campus expanded, it soon became apparent that what was needed was the ability to truly share meaningful data campus-wide in such a way that end users – from the departmental level to senior administration – could independently access, manipulate and produce reports from the data when and how needed.

Heretofore, such operations had required the scheduling and running of batch report jobs, taking anywhere from several hours to several days. In an ever-increasing rush to acquire information, this was simply no longer sufficient. Most importantly, with the increasing demands of departments calling upon the growing data pools, it became critical to find a way of reconciling that data so that reporting from every point could be made easy, clear, accurate, and above all credible. To this end, issues of how, when and by what criteria data should be extracted became of crucial importance.

A case in point was the Student Admission, Financial Aid and Records Information (SAFARI) system. The concept of data warehousing was first tentatively explored in the early 1990's, during the development period of the SAFARI Project.

The main concern at the time was that there were so many tables required of this system, and the queries against them were so complex, that mainframe resource utilization may be negatively impacted. This, indeed, proved to be the case. Generating reports based on increasingly complex queries using Information Builders, Inc.'s FOCUS® software, the functionality of SAFARI and similar systems was becoming increasingly cumbersome and time consuming.

Another key factor was the need to distribute across the user community the ability to create and run required documents and reports from the desktop level. Not only were mainframe hardware resources being taxed severely; Computing Services manpower, too, was being increasingly consumed in the scheduling, preparation and running of ever-growing numbers of ad-hoc departmental batch report jobs. While a good number of mainframe-based batch-run reports continue to exist, it was felt that the bulk of these data-reporting tasks belonged in the hands of the end-users or their departments.

Early in 1995, Computing Services realized that warehousing needed to be developed and made the commitment to give it priority attention and assign personnel. As people and new skills were combined, it was found that in order to properly analyze business – and especially financial – information, new software and a much more user-friendly front-end was needed.

In addition to redesigning the user interface, Computing Services faced an entire overhaul of system architecture and processing. What existed at this point were transaction-based production systems which were not a good environment for warehousing. They were set up to deal with limited amounts of transactions in particular systems over isolated and relatively short time frames. What was needed was a process-based system, capable of accessing large amounts of data for very narrow, quantifiable time frames: an analytical tool capable of revealing trends and the "what-ifs" necessary in making better business decisions. Quite simply, the existing information systems were not getting the job done.

By November of 1995, with the technology of both hardware and software having improved substantially, the University of Arkansas entered into an analysis phase, exploring their options. At this point, the Data Warehouse Project officially began.

Prototyping the Warehouse

Initially, this was a period in which the newly-assembled team tried to understand what data warehousing was all about. Approximately three or four months were spent doing nothing but research – from reading all the available material to consulting list servers and entering into discussions with personnel at other universities who had themselves undertaken the Herculean task of creating data warehouses. From these sources, the University of Arkansas was able to begin formulating an idea of what tools were available and how to adequately structure the data warehouse.

Data Warehouse Platform Issues

The Data Server and Peripheral Hardware

The platform selected to act as the server for the UAF Data Warehouse was a SUN SPARCcenter™ 2000 UNIX® server. In October of 1996, most users and applications were migrated off of the SPARCcenter 2000 to a new UNIX server – a SUN Ultra™ Enterprise™ 5000, thus freeing up the 2000 for other uses. SUN worked quite effectively with the University to increase our computational power while keeping our investment reasonable.

As presently configured, the Data Warehouse Server contains two 60MHz SPARC CPU modules, 512Mb of random-access memory, and 25.2GB of disk storage space, partitioned into twelve 2.1GB drives operating on two different SCSI channels. The unit uses an FDDI network attachment and is equipped with a CD ROM and 8mm tape. The operating system in use is Solaris version 2.5.1, which has proven to be a good, robust operating system. The Online DiskSuite affords the data warehouse with RAID 5 support for large file system access across the multiple 2.1GB drives. It is anticipated that around 100 people could simultaneously access the warehouse server with acceptable throughput. With the actual processing of data being done on the workstation level, the demands on the SPARC CPU should be well within its capacity to meet our service requirements.

Workstation Recommendations

The way in which the University of Arkansas data warehouse has been configured places the processing and memory-intensive number-crunching power not on the server, but on the end-user's workstations. While it is felt that most 486 machines (or their Macintosh counterparts) available could be used with the warehouse, performance at the workstation level is of paramount concern. For that reason, the development team is recommending that anyone wanting to access the data warehouse have at least the following workstation configuration: a Pentium processor, at least 16Mb of Random Access Memory, a 1Gb hard drive, and 1Mb of Video memory.

Evaluating the Available Warehousing Development Software

Price, compatibility, versatility and user-friendliness were the main criteria considered when evaluating software for use in the construction of the University of Arkansas data warehouse. The evaluation process itself involved both those products we had in-house, as well as information garnered from interviews and demonstrations from other campuses around the country who had already begun work on their own warehousing systems.

In the course of our evaluation we looked at such packages as Platinum Technology, Oracle and Informix. While each had some features which we found intriguing, none combined them as seamlessly and completely as did the SAS software suite.

Platinum Technology, as we discovered in interviews with Platinum representatives in Little Rock, was merely a front-end which orchestrated the functionality of various 3rd-party products. This would not do for our purposes. What we were after was something which would require us to deal with one vendor if technical assistance was required.

Oracle, as demonstrated to us by the people at Arizona State, looked quite intriguing at first, until we learned that for the use and deployment of ODBC (Open Database Connectivity) drivers we would again have to go through an additional 3rd-party vendor at a cost of close to $100 per driver per workstation. Assuming one ODBC driver per station and 100 stations, the price tag of this would be around $10,000 – just for the ability to access the data. This, too, was not palatable. In addition, to filter out duplicate keys required the writing of specialty modules. This, it was discovered, was not necessary in SAS as a duplicate key filter was a built-in function.

We then looked at the Informix product we had in-house. In many respects, it looked like a viable platform on which to construct our data warehouse. It was robust, had many of the desired features "under one hood" and was already part of our in-house software set. But as we started looking deeper, we discovered a failing which made Informix, too, a "no-go": It required all the data to be in flat, unpacked, completely "vanilla" ASCII format. Also – like Oracle and other packages – Informix required the purchase of 3rd-party ODBC drivers on a per driver/per workstation basis.

In contrast to the above packages, we found that the SAS® software suite combined the essentials of a user-friendly interface, robust development tools, and the ability to access data from multiple data sets all in one package. Other solid points in SAS®'s favor are upward and downward compatibility between versions, ease of use and maintenance. Additionally, the ODBC drivers – capable of accessing 17 different data sources – are included in the SAS base package.

Armed with this information, the warehouse team set about evaluating the SAS software against – as we saw it – its only competition for their purposes: Informix.

The University of Arkansas Development Team tested SAS® against Informix for about a month. Initially, we were not entirely sure which software would give us the results we wanted. But, once we got some test data on-line and put the different products through a speed trial, it was no contest. SAS® won on speed, hands down.

For this test, an unindexed General Ledger data set of 100,000 records in flat ASCII form was used. However, given the nature of the ADABAS format from which the data was drawn, the numeric values were in packed format. As cited above, Informix could not interpret these on its own. We were forced to write Assembler routines to unpack the affected data and massage it into a form acceptable to Informix before processing could be done. Once the source data was put into a form acceptable to both packages, the benchmarking began.

The initial test was a timing of how long it would take each package to download the selected records from the mainframe to the UNIX server and log out of the TSO session. The SAS software did this in two minutes and 34 seconds. Informix required an astounding 69 minutes! The results were so staggering and unexpected that we repeated the test several additional times. Each bore out the results of the first.

At this point, the development team looked no further. We had found our development platform. In fact, the SAS® software is so complete that right now about 90 percent of the University of Arkansas Data Warehouse is constructed from that one product. The other 10 percent is limited to front-end tools and GUI interfaces from 3rd-party sources, using the new SAS® ODBC drivers.

Installation Issues

Thus far, installation – both on the server and workstation side – have gone relatively smoothly. However there has been one recurrent and bothersome problem faced by the UAF warehouse development team. This is the use across campus of multiple vendors' flavors of file transfer protocol (FTP) software. Fortunately, this has been on a department by department basis. Within departments, at least, the users have standardized on a single platform. Still, the implementation of a campus-wide warehouse is occasionally being hampered by having to circumvent problems caused by disparate software. It is hoped that now this problem has been realized and there will be a campus-wide move to standardize the FTP and related packages to allow for a uniform "look and feel" functionality of the warehouse.

Loading the Data

The loading and fielding of data to the UAF warehouse currently incorporates both automated and manual processes. The automated routines are based upon two sets of UNIX scripts which, in turn, call sets of SAS programs.

The first UNIX scripts are load procedures which interrogate MVS data sets and extracts to produce raw SAS tables. Once the UNIX scripts are executed, the SAS routines they call establish a connection between the SUN2000 and the mainframe using SAS/Connect. Following a login routine, the connection is checked and a SAS session is begun on the TSO side, using a TCP connection. At this point, one of two things happens: If the source data to be read is a flat ASCII file, the SAS program uses an INPUT statement, following which the data is fielded internally to the program module, read in, and the resultant SAS data set created. If, however, the source is a non-flat file – such as ADABAS – a bit of pre-definition is involved. This requires defining a permanent OS data set in the TSO session using the SAS Access module specific to the model of the source data used. In our case, it was ADABAS. The developer or system administrator must then describe each field required of the final data set into library members. Once this is done, the appropriate library name is added into the SAS load script. When executed, the result is a seamless porting of data from the mainframe to the UNIX server.

Following the load scripts, a second series of UNIX scripts and SAS code is run to massage the SAS data in order to make it generic for use with various workstation query tools such as SAS, BrioQuery, and the Microsoft Office suite of software. Upon completion of this process, data is moved from the raw data sets created by the load routines to the permanent data warehouse file structures.

At this point, given the data which is being used in the warehouse, there are six sets of LOAD and RUN modules. These handle the transfer of common, daily transaction, FOCUS extract, historical, payment detail, and projected information from the MVS systems to the UNIX data warehouse server.

Currently, manual intervention is required to execute the load and run SAS programs. The operator must first log onto the UNIX server. Next, the appropriate UNIX scripts are executed in a predefined sequence to create the required SAS data sets for the UAF Warehouse. Upon successful completion of the load and run process, human intervention into the loading of the SAS data is at an end. If a failure occurs, error messages are generated and the operator must step back through the process, diagnose and fix the problems which generated the error(s) and re-run the jobs.

Designing User Interfaces

Just as important as the server-side issues of data warehouse development is the creation of flexible, user-friendly interfaces to enable end-users access to and reporting capabilities from the data of the data warehouse. In this regard, the UAF Data Warehouse Development team has been quite pleased with several of the 3rd-party query tools and development products on the market. Having looked at such packages as Powerplay, Forest and Trees, BrioQuery, the SAS GUI tools and the Microsoft Office suite of software, we were impressed with the features we saw in each. However, in terms of user-friendliness, three stood out: BrioQuery, Microsoft and SAS.

In fact, based on ease of use, functionality, flexibility, and performance, the Warehouse Development Team is currently recommending BrioQuery® as their product of choice for an ad-hoc query tool. Following this, they suggest the use of the Microsoft Office® suite of products. Additionally, we are in the process of further reviewing the latest of the SAS® GUI tools. Depending on what is learned, the SAS® tools may potentially replace BrioQuery® as Computing Service's number one recommendation for a user front end. From what has been learned thus far, using SAS at the workstation level requires no additional tools. Whether the platform be PC or Macintosh, users will find a noticeable performance increase with SAS over the other products as the SAS software will directly interrogate the warehouse data without the need of interpretation through ODBC drivers.

Initial Deployment

One of the first bottlenecks encountered was common to the other campuses the development team had consulted: the end-users by and large were not sure of what the data warehouse could do for them and, more importantly, *what they wanted it to do for them*.

Thus, the most expedient method was chosen to illustrate the nature and benefit of a data warehouse. Based on the available user input, as well as consultations with accounting and senior administration personnel, a prototype was created and put on-line.

This prototype operated from April to September of 1996, during which time both end-users and the development team learned a great deal about what not only should, but could be done.

With all this end-user simplicity, speed and flexibility, one would think that the data warehouse would have been next to impossible to accomplish in the short length of time the University of Arkansas team has been working on it.

This has not been the case. In fact, the Development Team has experienced no problems to speak of. Even in the setting up of the secure server, we had no difficulty. SAS® has proven to be reliable and a more complete product than any other in the field thus far reviewed. The support people are always available, very cooperative and helpful. Even the price is right. Putting all this together SAS® has given organizations such as the University of Arkansas the ability to develop data warehouses completely in-house resulting in huge savings.

End-User Feedback

The prototype phase of the data warehouse garnered an abundance of user input and comment, giving the development team the criticism and feedback they needed. Development proceeded at a fast pace. Then, in September, the first production elements of the warehouse went "live".

Subsequent to the prototype going "live", the Data Warehouse Development Team continues to solicit input and feedback from both hands-on and decision-making users so that better tools could be built for them. The goal is to make the data warehouse very user-friendly and as non-technical as possible by putting the data out there in a manner the users can understand.

At first, people on the Fayetteville campus were reluctant to use the warehouse. But when they began to understand what it was and what it could do for them, they soon learned how indispensable it could be to them. When they access the warehouse, end-users now have a great deal more available to them – both in data sets and retrieval/reporting capabilities – than were available to them before. Added to this, data access is much easier and faster than it used to be. Users can get directly at what they want and report against it in their own way. This is especially important for decision-makers. It gives the Administration the ability to see who is overspending, the status of departmental budgets, and so on – all instantly, on demand.

One of the most gratifying user responses came from the University of Arkansas auditors. Once the data warehouse had loaded in the University's core financial data, they had asked to be set up so that data sampling could be done. What the auditors found was that not only was the data extremely accurate; but they were able to get in a couple hours what previously had taken them a couple of weeks to obtain. They were quite impressed.

Conclusion

Automation is a Priority

When first implemented, the process of collecting, massaging and loading data to the warehouse file structures was a heavily manual intensive process. In the past several months the Development Team has found ways to reduce the overhead of coding to only two external batch processes, consisting of only a few lines each. However all the team members acknowledge that impressive as this is, it is merely a mile post, not an end in itself.

There are two categories of tables stored within the data warehouse: data which is overwritten and that which is appended to. The handling of the update procedures – especially as regards this second type of file – is of much concern to the development team. While the non-accumulating files can be easily replaced by re-running the extract and file creation routines, the accumulating files pose a special problem. If a production run terminates in an error, all the records which were appended to the accumulator must be removed, the integrity of that file must then be verified, and the append process begun anew.

At this point, key steps in the data load and update procedures are – as cited above – manual in nature. This has resulted in occasional production run errors caused by inadvertent multiple running of load/run routines, as well as missing the load or run of certain SAS programs.

During the initial start-up phases of the data warehouse, these manual operations were necessary. However, as the development progresses and more users come on-line, it is essential that this manual interaction be reduced or – if at all possible – eliminated so as to increase the integrity and reliability of the data.

The warehouse development team is currently working on a method of automatically scheduling the loading of data from the MVS system to the Data Warehouse, thereby eliminating the need for human intervention.

Performing Updates to the Warehouse Data

One of the biggest problems which the development team has had is that some users fail to log off their session. This results in the locking of the warehouse datasets, thereby preventing subsequent updating of the data tables during the nightly update run. To counteract this, the development team is currently testing a script file which will log off all current users and restart the server so that nightly production can be run properly.

Performance Issues

When the completed Data Warehouse application goes "live" there is another substantial difference which is of concern: The other tools – Informix®, Oracle®, Sybase® – all need to maintain indexes. Given the amount of data which will potentially be stored in the warehouse, these will take up huge amounts of storage space. Thus far, it appears that SAS® does not have this problem. The University of Arkansas Development Team hasn't yet found any real need for indexes. Even though some of the data files currently have over a million observations in them, there have been no noticeable performance problems.

In order to validate the timing issues of the end-user front ends we are reviewing, the data warehouse team did a head-to-head comparison between SAS/Desktop, Microsoft Access 97 and BrioQuery 5.0. All three were tested on a Pentium 200 MMX laptop with 48 Mb of RAM, a 2.1Gb hard disk and 2 Mb of video memory, running against a cumulative General Ledger data table containing 4,289,105 records. In all cases, we selected every table field for our query and set a filter on company name and department. SAS/Desktop retrieved the selected table data in 6.75 seconds as opposed to 7.15 seconds for Microsoft Access. BrioQuery lagged far behind, requiring an astounding 4 minutes and 4 seconds to gather the same data to the screen. In the case of SAS and Access, there was an added test performed: that of opening the entire data set. In this, Access clocked in with an amazing 1.21 seconds, surpassing the 4.19 seconds required of the SAS/Desktop application. Overall, both the SAS and Microsoft products were quite impressive in their speed of retrieval. If the data grows into the terrabyte realm, it is conceded that perhaps indexing of the SAS® files will become necessary. But for now, indexing appears not to be needed.

At this point, it should be noted that indexing only affects non-ODBC data access. Whereas indexing does indeed assist performance of data acquisition through native SAS, ODBC connections seem unaffected and oblivious to any indexing used.

Another item currently under development concerns the passing of 40-character names and labels from SAS® to the end-user front-end packages through the ODBC drivers. Work on this has just recently started. However, great progress has been made and it is hoped to see this feature implemented soon.

Capitalizing on Desktop Legacy Data

General ledger data has been on-line within the Data Warehouse since the middle of 1996. Since then, the Development Team has added other financial and administrative data to it. Most recently, this past May, SAFARI information was added.

One of the most fundamental benefit of the Data Warehouse is that it extends the lives of the legacy systems and makes the data they contain more accessible and user-friendly. So instead of scrapping things and starting from scratch, Computing Services is able to capitalize on what has already been built and use it advantageously.

In the months ahead, the users on the Fayetteville campus can look forward to yet more accounting and financial data moving to the data warehouse, mostly from legacy systems.

With regard to new interfaces and retrieval mechanisms, the Development Team is exploring alternatives to the use of FOCUS® as a query and reporting tool. This is not to say that it will necessarily be phased out. But, with the advances in front-end and connectivity software over the time the UAF Data Warehouse has been in development, several products look quite favorable right now – GUI products like Microsoft Access®, SAS/Assist®, SAS/Desktop®, SAS/EIS® and BrioQuery®.

Summary

Even in the short time of its existence, the University of Arkansas data warehouse has already built up a goodly number of clients. Here on the Fayetteville campus, there are currently over 100 end-users in several departments, from Agronomy to Human Resources. Most recently, the Systems Office in Little Rock has come on-line.

From all indications, this is just the tip of the iceberg.