Universal Data Access: A Success Story
Bruce Babbitt, New England Power Service Company, Westborough, MA
Marianna Suciu, SAS Institute Inc., Cary, NC
Emily Wallace, SAS Institute Inc., Cary, NC

ABSTRACT
Data are a valuable asset to a corporation but data are stored in many different forms on many different hardware platforms. Getting to all of the data needed to make an informed decision can be a challenge because it can require the use of several different tools on several different operating systems. By the time the data are gathered and the decision is made, the business opportunity could be lost.

SAS/ACCESS® software provides a direct and transparent interface to your database data and can be used in conjunction with other SAS® software to access all of the data in your enterprise. The same set of programming statements are used to access data in SAS files as well as the data in databases so you use one set of tools for all of your applications. It is very easy to gather information quickly when you need to make important decisions.

This paper demonstrates the ways that SAS/ACCESS products can be used to build applications that retrieve data from a variety of sources. It shows how seamlessly the SAS/ACCESS products are integrated into the SAS System to enable you to build powerful applications that cross hardware types and database systems.

INFORMATION DELIVERY
The SAS System offers a complete strategy for information delivery. Information delivery is the process of using technology to bring people together with the information they need to make better decisions. This in turn helps people in an organization achieve their business goals.

BARRIERS
But there are certain obstacles or barriers we have to overcome in order to turn data into useful meaningful information and get it to the right people at the right time.

We have summarized these into four categories centering around the diversity found in today’s organization: diverse data sources, diverse application needs, diverse users needs, and diverse hardware environment.

In this presentation, we will concentrate on the diverse data sources; discussing 1) the evolution of business computing and the SAS System to meet changing user and environmental needs, 2) a user success story, and 3) the technology of the SAS System to access enterprise-wide data.

THE EVOLUTION OF ENTERPRISE COMPUTING
Until the early eighties, most organizational computing was occurring on mainframe and minicomputer processors. Data were stored in proprietary file structures such as flat files, system files, and hierarchical database management systems. Applications were written by trained programmers, using third generation languages such as COBOL. In fact, other than data entry staff who may have used front-end to a CICS® application, programmers were the sole users of the system, cranking out code that would enhance financial applications or that would become batch production reports. Management and other decision makers used these production reports to help them make day-to-day operational decisions as well as longer-term, more strategic decisions.

As we moved through the eighties, many factors influenced the changes that have molded enterprise computing as we know it today. Perhaps the two most important, the advent of personal computers and increased global competition, pushed the drive for technology that could deliver timely information to decision makers.

As members of the organization pressed the Information Systems personnel to provide a new report or enhance another, application backlogs formed. As a solution, the need for programmer productivity tools arose. The SAS System, widely known for its superior data analysis and management capabilities, along with its library of rich functions, was adopted as a new standard in business computing. This confirmed that the reputation that SAS software had earned in the scientific community also applied to more general business computing tasks.

The SAS System’s fourth generation language was touted by the industry as a major productivity tool. In fact, early advertisements for the SAS System positioned the time savings and code compression in applications programming compared with more older, more traditional approaches such as COBOL. Also, giving momentum to SAS Institute’s growth in the area of general business computing was the ease in which SAS software could access almost any file structure, even the most complex. Recognizing the growing need for more sophisticated user productivity tools, SAS Institute introduced an interactive programming environment, an interface to IMS/DB to facilitate easier access and reporting, application development facilities, and various decision support modules.

As we reached the mid-eighties, demands on Information Services staff continued to skyrocket. Surveys of data professionals showed applications backlog as their primary issue. Realizing that the “end-users” of information knew their business problems better than an IS staff person and that additional IS staff, as corporate overhead, at some point would no longer be tolerated,
IS created the concept of the Information Center. The Information Center concept was adopted by many organizations as a way to empower the end-user. The success of the Information Center varied from site to site, depending on how attuned the Information Center staff was to providing access to corporate data, appropriate computing tools, and proper training and support.

Database vendors also recognized they could make a positive impact on productivity by providing products that simplified the creation and maintenance of database management systems. As a result, the standards embraced by the relational data model including Structured Query Language (SQL), found their way into mainstream computing. With these relational database management systems (RDBMS), many new corporate databases began to move away from the older style data files. As time passed, the industry began to acknowledge the need for coexistence of older "legacy" data types with the newer RDBMS. This was due to the amount of resources that would be consumed to convert older database management systems as well as the possibility of losing performance if large scale production databases were made available to end-user reporting systems.

As the use of these newer database management systems grew, users of the SAS System desired the ability to interact with these proprietary data structures. SAS/ACCESS software came into existence, allowing data to be extracted from popular DBMS's and then made available for data management, analysis, reporting, and presentation using the SAS System's tools. SAS/ACCESS software was very popular since it gave the database administrator an easier way of making desired corporate data available.

As the decade ended, global competitive pressures continued to build. Organizations realized that building a competitive advantage would mean bringing together their most important assets: their people and data. Effective utilization of technology was the vehicle for empowering the end-user, or knowledge worker.

SAS Institute recognized these evolving challenges of computing and identified the barriers that must be addressed in order to provide true information delivery. During the late eighties to present, SAS Institute built tools to address those barriers: transparent data access methodologies; a variety of user interfaces (GUI's), each designed to provide the right combination of ease-of-use and flexibility for the target user; a myriad of application types, all seamlessly integrated; and finally, software that is truly portable yet exploits the particular hardware environment.

With the advent of Version 6 of the SAS System, SAS Institute introduced a comprehensive strategy for data access. Multiple Engine Architecture, building on the capabilities already found in the software. This strategy includes the SAS/ACCESS product line for transparent data access to popular relational and hierarchical database management systems as well as host files; support of industry standards, such as SQL, for passing queries directly to the relational database; support of industry initiatives, such as the IBM® Information Warehouse and the Microsoft® ODBC; and information database capabilities to enhance ad-hoc applications.

Therefore, the SAS System treats data as a generalized and available resource by providing transparent access to those data wherever they reside. The technology is implemented in such a way that recognizes the impact that various styles of computing can have on performance. This comprehensive data access strategy provides all the components for fine-tuning any user and/or application need.

The following user story is representative of the issues that most large organizations in the U.S. face regarding data. Data are scattered throughout New England Power Service Company in a variety of data sources including DB2®, the SAS information database, and legacy data in older production IMS databases on the mainframe; INGRES® on departmental machines; and various data on the PC. Using the SAS System, New England Power Service Company delivers timely information to the user population.

DATA ACCESS USING THE SAS SYSTEM AT NEW ENGLAND POWER SERVICE COMPANY

New England Power Service Company (NEPSCO) is a subsidiary of the New England Electric System which owns three retail electric companies in Massachusetts, Rhode Island and New Hampshire. NEPSCO is running MVS/ESA on an IBM ES/9000-580, VMS on a VAX® 8550 and 6610, and several Novell® local area networks at our main office in Westborough, Massachusetts. In addition, there are smaller VAX systems, ULTRIX® workstations and more local area networks in our power plants and district offices throughout New England. These are all interconnected through a wide area network.

There is a need for users on any LAN to be able to access data from and communicate with any other location and platform. These data exist in IMS, DB2 and VSAM databases on the IBM mainframe, INGRES data bases on the VAX systems and workstations, and Paradox® on the LANs. On the IBM system, users can access any type of data with the SAS System. INGRES and Paradox users are presently using the access tools that are delivered with the database.

NEPSCO licenses the following SAS products on the IBM mainframe: Base SAS, SAS/AF®, SAS/ASSIST®, SAS/FSP®, SAS/GRAPH®, SAS/SHARE®, SAS/STAT®, SAS/CONNECT®, SAS/ACCESS® Interface to DB2, and SAS/ACCESS Interface to IMS. On the PC DOS platform, we have base SAS, SAS/AF, SAS/FSP, SAS/GRAPH, SAS/STAT and SAS/ASSIST software. Users can log onto the mainframe from their LAN and run a SAS session under TSO/E, or run their SAS session from DOS on the PC and access mainframe data when needed by invoking SAS/CONNECT software. While past development of SAS applications has been primarily on the mainframe, we can now have users develop and test applications on the PC and port them to the mainframe later.

History of SAS Software at NEPSCO

SAS software has been in use at NEPSCO since 1981. At first, we used SAS software to do system performance analysis, capacity planning, problem and change management, and assisting users with ad-hoc requests. Our database at that time was IMS. We used the SAS/IMS-DL/I® product to extract data into SAS files for
analysis. The DLITEST feature was used to test DL/I call patterns and fix problems on IMS databases.

We found that SAS software could be used to access any type of data that we had. The Technical Support group used SAS software for data analysis, reporting, prototyping, and fixing damaged data. We built a system to generate MFS code (for building IMS online screens) through interactive panels that could be developed by end-users. We built problem and change management data base systems, and a software inventory database. The engineering, demand planning and rate departments have used SAS programs to collect, analyze and report on many types of data.

With the development of relational databases and the SAS/ACCESS engines, we were able to develop applications that could be prototyped and developed using SAS data sets and ported to DB2 without changing the application. Any data that can be accessed through one of the SAS database engines can be accessed as if it were a SAS data set; the program, the developer and the user do not need to learn a new architecture. Equally important, our older, hierarchical IMS databases can also be accessed through a SAS database engine, providing the ability to use relational type queries (SQL) against IMS data.

Two of our notable success stories include the Safety database and reporting system and the Street Light database and maintenance system. The Safety system, originally projected to take 18 man-months using the traditional COBOL-IMS database development approach, was completed in six weeks by a person with no prior SAS programming experience, with assistance from a more experienced SAS programmer. The database was put into a SAS data set and the application was developed using SAS/AF software. We use SAS/SHARE software to allow simultaneous update by multiple users. This system keeps track of all on-the-job injuries and illnesses, and is used for OSHA reporting and our corporate safety goals program. Personal information on injured employees is imported by the SAS System from a payroll IMS database as the user enters the accident information.

The Street Light system was installed in less than two months versus an estimate exceeding 12 months. The data were put into a SAS data set and a prototype was built to get approval on the approach. The developer was able to work with the user to incorporate changes and new ideas that were generated by the initial model before a significant amount of time was invested in the initial design. When the system was ready for production, the data were ported to DB2 with only a few changes to handle record locking. The users were very excited about having the ability to participate in the development of the application, as opposed to older systems where the system was developed according to the original design specs and the user could not see the results until the project was nearly complete.

SAS has been in use in place of QMF and other DB2 access tools. The ease of building user interfaces to the data can provide for queries by users with no knowledge of the data or the platform where the data exist. At the same time, power users have all of the tools that they need to read, update, join and analyze data from any database. These same tools can be used for IMS databases as well. Most of the ease and power of the SQL language is available to access IMS data. To ease the process of building access descriptors for IMS, we wrote a SAS program which reads the IMS Data Base Descriptors (DBDs) and creates SAS access descriptors.

The SAS/FSP product, originally designed to edit SAS data sets, now can be used with any of the SAS/ACCESS engines. This allows us to do inquiries and updates against the databases, presenting data one record per screen or in list mode, with no programming. This has been very useful for providing quick access to data, fixing problems in databases and testing call patterns before coding them into a program. Customized FSP screens have been built into several of our applications.

Replacing older applications usually requires a large data conversion effort. With SAS software we have been able to significantly reduce the amount of time that is required to accomplish this. Since SAS programs can read or write data in any format, easily changing data formats along the way, some conversion efforts have been reduced by over 50% from early estimates.

NEPSCO DIRECTIONS

We are in the process of determining how data will be accessed across the organization. We want people to be able to get to data in any format, on any platform, without regard to how the data are stored. To this point, SAS software is the only product that we have reviewed that can access almost any of our corporate data on any platform, with the same interface. SAS/ASSIST users can do sophisticated reporting and analysis on relational databases, hierarchical databases, SAS data sets, VSAM files and sequential files through the delivered panels, or they may have customized panels built that are more specific to their needs. We have recently begun an evaluation of the SAS/ACCESS interface to INGRES on the VAX. As on our other databases, it is so transparent that the users do not know whether they are looking at DB2, INGRES, SAS data or some other data.

With SAS/CONNECT Software, the SAS System can be run on any of our systems and exchange data and programs with any other. We have built applications on a PC and run it without changes on the IBM mainframe, and vice versa. By automating the connections between systems (which is greatly facilitated by scripts provided with the product) we can have users at a desktop device select the data that they want to access, and SAS will determine where that data resides, make the necessary connection and bring the data, a subset of the data or a report from that data, back to their device. We have found SAS/CONNECT software to be able to handle many more types of data transfers than our other file transfer products. Besides using it to transfer SAS data sets and SAS catalogs, which the other programs cannot handle, we also use it to transfer all types of other data between platforms.

The ability to prototype applications, the almost identical user interface on every platform, connectivity, and the ability to access almost any type of data make the SAS System a solid choice for enterprise data access.
**SAS/ACCESS Technology**

The configuration at NEPSCO is representative of many companies in the industry today. They have a mixture of hardware platforms and their data are stored in many different files and databases. This is the kind of company we had in mind when we designed the Multiple Engine Architecture and implemented our SAS/ACCESS products.

When we originally designed the Multiple Engine Architecture, our primary goals were:

- transparency for applications and users
- ease of use
- extensibility.

We wanted to provide our SAS users with the same kinds of access to database data that they have always had to their SAS data. We wanted them to be able to use the same programming steps and application building tools no matter whether the data were stored in a SAS data set or in a database table or file.

The information that Bruce provided about his company’s use of SAS Software and the SAS/ACCESS products shows that we are achieving our goals. He states that users at his site have been able to access different types of data with the same SAS programs. They have moved applications from one type of data to another without changing their code.

They found that engines are easy to use because an inexperienced SAS programmer coded the Safety light application which gets data from an IMS database.

They have also benefitted from the extensibility of the SAS System. Because the IMS engine works with the other components of the SAS System, the SQL procedure features are available to IMS users. As new features are added to our software, they immediately work with our engines so that they can be used with all types of databases and files.

With the SAS/ACCESS products, our development goals were:

- to provide interfaces to all of the popular databases
- to make the database access transparent so that SAS users could use database data as easily as they use SAS data
- to make the interfaces powerful so that users could build SAS applications for their database systems instead of having to develop them in COBOL or other programming language
- to make the interfaces integrate seamlessly with the other SAS products
- to make the interfaces as similar as possible so that it is easy to switch from one product to another
- to provide additional functionality where possible
- to provide security and good performance.

We currently have interfaces for the following databases:

- DB2, IMS, SYSTEM 2000 Data Management Software (DMS), ADABASE, and CA-DATACOM/DB on MVS; SQL/DS and SYSTEM 2000 DMS on CMS; ORACLE, Rdb/VMS, INGRES and SYBASE on VMS; DBF, DIF, OS/2 Database Manager, and AS/400 on OS/2; DBF, DIF, and SQL Server on Windows; ORACLE and SYBASE on SUN; and ORACLE on RS/6000. More interfaces are under development for release later this year. We will continue to expand our interface product offerings in response to our user demands.

One of our highest priorities in developing the SAS/ACCESS products has been to make the access to the database transparent so that a SAS programmer would not require extensive database knowledge or training. NEPSCO programmers were able to build an application involving access to IMS, which is a relatively complicated database system, with an inexperienced SAS programmer and were able to develop the application very quickly.

All of our interfaces with the exception of the DIF engine update the database data as well. This gives users the ability to develop many different types of complex applications quickly as NEPSCO discovered. The interface is simple enough to be used by an inexperienced SAS user and is robust enough to allow a database administrator to fix a problem in a database. Applications can be prototyped with SAS data sets and migrated to the database once the application is ready for production. This means that the development and testing of the application will not adversely impact the database system performance. Applications can be developed and modified quickly and this makes it easy for users to be involved in the design process. This results in better applications and happier users.

Since our SAS/ACCESS interfaces allow full read and update support and integrate with the other SAS products, they can be used in place of IBM’s QMF and other query tools because they can do the same queries. In addition, users can do analytical, reporting, graphics and other application functions so they use one set of tools for all of their database access.

Because of the Multiple Engine Architecture, engines interact with the SAS System in the Core and Host layers so all SAS procedures and the DATA step are available to the database users. This means that SAS products such as SAS/FS and SAS/ASSIST can be used equally well with databases as well as with SAS data sets. This seamless integration makes database data very easy to use.

As Bruce noted, his site is just starting to use our SAS/ACCESS interface to INGRES and has found that it is easy to use because it is very similar to our DB2 and IMS interfaces which his site is using. We have made every effort to keep our interfaces as similar as possible to make it easy to switch from one to another.

We have also built additional functionality into engines where possible. The BDLOAD procedure is available in many products to create and load a database or table from SAS data set input. We have also implemented the PROC SQL passsthrough feature in many of our SQL-based engines to allow sophisticated database users to enter statements that are sent directly to the database system for execution. These features can be incorporated into applications so that they are even more powerful.
Conclusion

It is very gratifying to listen to Bruce's description of the way that his company uses our products. It shows that we are achieving the goals we set for our engine architecture and our SAS/ACCESS products. But we are not going to rest on our laurels. Right now several new interface products are under development and others are on the drawing board. We are working very hard to add interface products in the PC and UNIX areas. We are also continuing to enhance our existing products by supporting new releases of the database systems and the new functionality they provide. We want our products to meet all of your needs and we welcome any comments, suggestions and other feedback that you have.


AS/400, DB2, IBM, MVS/ESA, OS/2, QMF, and SQL/DS are registered trademarks or trademarks of International Business Machines Corporation.

ORACLE is a registered trademark or trademark of Oracle Corporation.

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