

Building a Graphical Network Analysis System Using SAS/EIS

Mahdi Al-Haery

MCI Telecommunications Corporation

In any company's effort to provide the highest quality of service, up to date information is not a luxury but a necessity. This philosophy lead management at MCI in a quest to make available the huge amount of data that is generated by telecommunication equipment to the engineers who need it. A method was needed to present the information in an easy, yet powerful and informative way. MCI engineers designed and successfully implemented a system to provide this information. SAS/EIS was chosen due to its simplicity and mainframe-to-PC portability and compatibility. This paper discusses the transformation of raw unformatted data on a mainframe into easy to use graphs on a PC. It goes through this process of transformation discussing issues such as data handling, connectivity and interface development. It also provides solutions to some encountered problems and it discusses the capabilities and limitations of SAS/EIS and SAS/Connect in relation to the project. The developed system is applicable whenever there exists a huge amount of raw data which needs to be filtered, processed, analyzed and presented in a graphical and easy to use method.

Introduction

State-of-the-art telecommunications equipment provide a great quantity of information about their performance. The raw data generated by the equipment consists of self diagnosis and status messages on the equipment in the field. When starting this project, this data was being generated, gathered and stored daily. Nevertheless, it was not being looked at, analyzed or examined. From an

operator's perspective, data from a single piece of equipment can be viewed and examined rather quickly, but when there are hundreds of such machines, developing a comprehensive overview of the network is virtually impossible. In order to monitor the overall health of the network, a group of engineers would need hours to examine one day's worth of data. Due to time constraints and lack of man power, this potential source of information was largely being ignored. Nevertheless, in an effort to proactively solve potential problems, management determined that this data is essential in network maintenance and expansion operations; that by observing trends in the data, one can detect a critical event before it becomes a problem. The goal, naturally, was to develop a system which network engineers can utilize in order to acquire an accurate overview of the network. The system had to give information on a variety of equipment and let the users choose the level of detail they wishes to explore.

Data Manipulation

The amount of raw data which is collected daily for all equipment is very large, in the order of 1.5 Gigabytes. But for the purposes of this project, not all equipment was going to be monitored. Furthermore, the data for the chosen equipment needed to be filtered to provide only key information on the equipment. As a result, many status message were truncated and some additional information was not included. The criteria for this selection was based upon recommendations made by maintenance and network operations personnel. They are the ones who determined what parts of the messages they would like to see. For some equipment they selected the whole message, for others it consisted of a small of part of the message. This was largely dependent on the type of machine. This filtering is essential because it reduces the data in the mainframe to a manageable size to be placed in the database.

In order to accomplish this task, six programs read this data from the mainframe raw data files. Figure 1 shows such a program. These programs place the filtered data in SAS databases. In order to migrate these datasets to a PC, SAS/Connect had to be invoked. SAS/Connect contains excellent scripts for communication with virtually any system. With little modification and expert help, a TCP/IP connection was possible. The SAS databases on the mainframe are further processed by programs on the PC which download the summary statistics of those data sets. Figure 2 shows a sample of such programs.

After all this filtering, there remains 25 megabytes of data which needs to be downloaded daily. When the system was first implemented, an Ehllapi/Irma board connection was used. Using this method of downloading, it took 10 hours to download one day's worth of data. But now using TCP/IP it takes around 30 minutes.

The data for some machines had to be divided by the type of information. This was necessary due to the fact that SAS/Connect could not handle data sets that are too large. The data was first divided into regions and then into the type of information. This division was performed at the SAS downloading level (Figure 2). Once the datasets are downloaded, then EIS can access them. By creating the metalist, the metabase and the EIS database, one can create business graphs. In addition, EIS was used to create desktops using the Graphical Menu Builder which either call other desktops or call a business graph.

Interface

The goal was to make the interface as user friendly as possible. The obvious advantage is that the users are going to be engineers with various backgrounds and training time had to be kept to a minimum. The second major consideration is that the sequence of information being presented has to match the way that the users

```

DATA SNAS.STPSWIC;
FORMAT SITECODE$7. ALMCOD$12. D1$8. D2$8. D3$8. D4$8.
      D5$8. D6$8. D7$8. D8$8. ;
INFILE STPSWC MISSOVER LENGTH=LENG;
INPUT @;
VLENG=LENG-73;
INPUT @1 TEST $VARYING200. VLENG @;
SIG=0; SIG2=0;
SIG=INDEX(TEST,'REPT');
IF SIG NE 0 THEN
INPUT @10      SITECODE $3.
      @SIG+6 ALMCOD $12.
      #3 @30 D1 $8.
      @39 D2 $8.
      @48 D3 $8.
      @57 D4 $8.
      @66 D5 $8.
      @65 D6 $8. ;
      @74 D7 $8. ;
      @83 D8 $8. ;
RUN;

PROC SORT DATA=SNAS.STPSWIC;
BY SITECODE ALMCOD D1 D2 D3 D4 D5 D6 D7 D8 ;
RUN;

```

FIGURE 1. A mainframe SAS program showing the reading of an external flat file and its conversion to a SAS database.

```

libname switch 'k8mxal.dms';

data switx;
set switch.Stpswic;
if almcod<'CAAA';
test=index(d1,'0');
if test = 0 then delete;
test2=index(sitecode,'-');
if test2=0 then delete;
run;

proc sort data=switx out=switxs;
by sitecode almcod d1 d2 d3 d4 d5 d6 d7 ;
run;

proc summary data=switxs(drop= sig );
by sitecode almcod d1 d2 d3 d4 d5 d6 d7 ;
output out=switx1;
run;

proc download data=work.switx1(keep=_freq_ sitecode almcod d1 d2 d3 d4 d5 d6 d7 )
out=downnew.Stp1;
run;

```

FIGURE 2. The SAS program on the PC utilizing SAS/Connect to download all of the important information from the main frame to the PC.

want to receive that information. For example, information about all of the sites combined preceded detailed information about a particular machine.

The interface also has to look familiar to what the users are already used to. SAS/EIS gives the developer the capability to build 100% MS Windows compatible software. The interface for the system is made up of icons, button and business graphs which represent a hierarchy of information regarding the different types of equipment and more detailed information about the equipment.

One problem which presented itself in the course of development was the fact that there were forty different sites for one type of equipment, the DMS-250. Representing the information on one screen makes the screen very hard to read. In fact, if there are too many items to show on a business graph the description of the items will not be written. Even when using the smallest font, Line Printer 8, Business Graphs can not handle more then thirty items when they are represented horizontally. Horizontal representation was necessary because the item descriptions were too large to represent vertically. The decision was made to divide up the forty items into three operational regions, North, South and West. By doing so, not only did the information become easier to read, but also the response time for a drill down decreased because the size of the data files became smaller. The same was true for other equipment. The sites were divided into their corresponding operational regions. Data for one type of equipment, the DEX600E switch, represented a new problem: Even though there fewer sites in each region, the number of types of status messages was too many to display on one screen. For some equipment types, there are as many as 80 different messages on any given day. This number is too high the SAS/EIS to show in a standard Business Graph, so I went ahead and divided the types of IPRs into five sections (Figure 3). For another type of equipment, the STPs, there are fewer types of IPRs, but

nevertheless, they were also divided into three regions. Table one shows the division of the data by date, type of equipment, region and type of message.

Data Organization and Presentation

The question arose on how to show different levels of information to the user. There are up to 8 data words which are associated with each IPR. For the other equipment there were two levels of information, the sites and the IPR types. The fact that there are ten different drill levels which need to be displayed presented a new problem: EIS can not handle more than 9 drill levels properly. As a result, the decision was made to drop the last drill level due to it being the least significant. We also needed information about the previous days network performance so the system was set up with seven days worth of data, each days worth of EIS files (Figure 4). The SAS data files were stored in a separate sub directory on the PC. This separation eases automation of the system. Automation still remains a problem. SAS is not designed to handle 100% automation of any application.

Concluding Remarks

There were many unanticipated problems with this project. The key was working around the problems and being flexible with the design and the requirements. The important issue remains the presentation of up to date accurate information in an easy to use friendly system. The system is now fully operational. The SAS data files are begin generated daily and are downloaded via TCP/IP and send via the same method to another location.

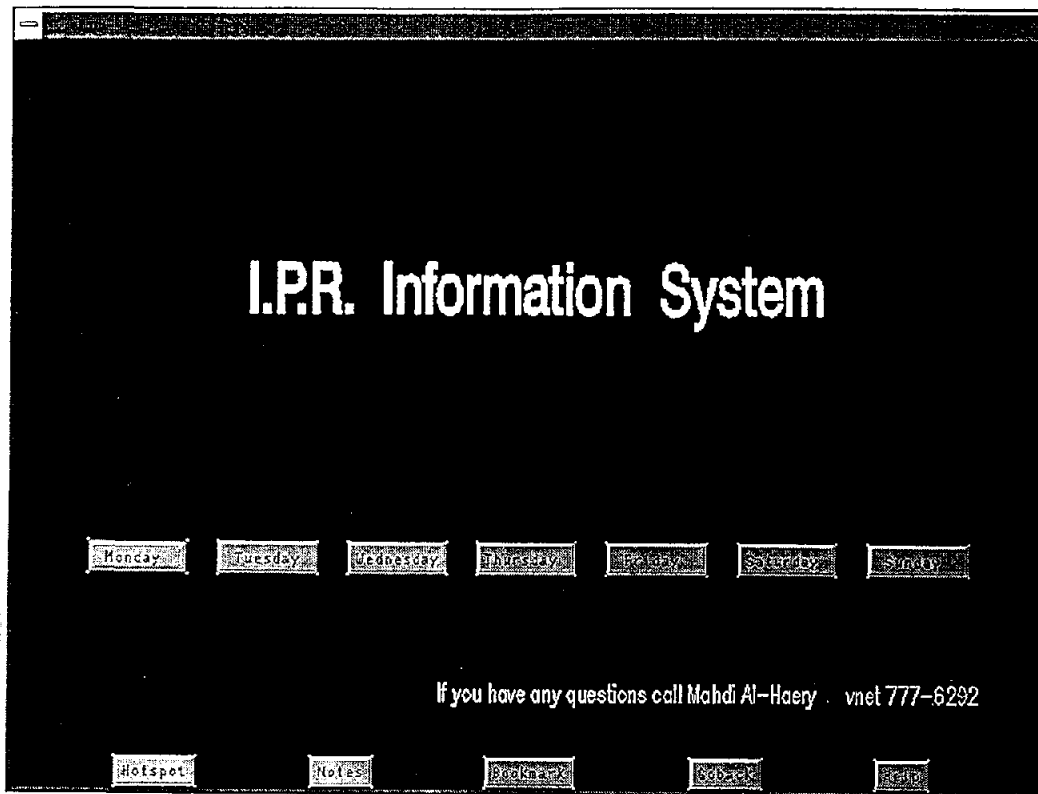


FIGURE 3. An EIS interface showing the choice for different days of the week.

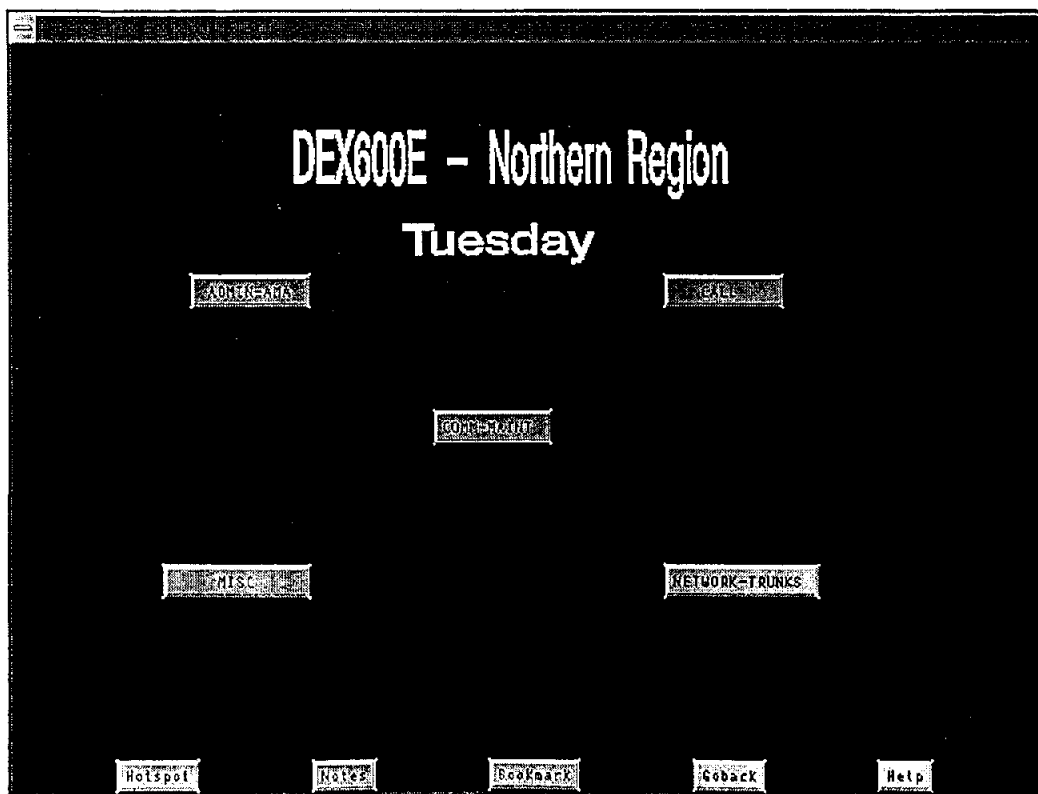


FIGURE 4. An EIS interface illustrating the need to divide the data into five sections