

MXG GOES ON HOLIDAY

David Vasey and Steve Jack

St George Bank Limited and SJCC Pty Ltd

Why run MXG on the mainframe?

Merrill's MXG has been the cornerstone of Capacity Planning for many organisations. It decodes data from many sources, while SAS provides the flexibility to store almost any data from any platform.

With the increasing costs of running SAS on ever larger mainframes what options are there?

This paper reviews the planning process behind moving SAS and MXG from a mainframe platform and looks at problems and successes in actually making the move from one site's experience.

1. INTRODUCTION

Have you ever thought of running SAS and MXG on another platform? What if you had no choice? Would you be worried about database size, run times, database integrity? Well, we were!

2. THE STARTING POINT

2.1 Business Architecture

St George Bank in a business sense is a merger of several smaller banking institutions, the two largest being St George Bank and Bank of South Australia, and the mainframe infrastructure still reflects those origins. Following this last key merger the best applications were selected for each business area from the constituent banks, and all data migrated to the winning applications. With the focus squarely on application migration and integration it was decided to leave the systems environments more or less intact.

2.2 Systems Architecture

The mainframe environment reflects the business requirements closely, and (prior to the zSeries) consisted of two machines, development and production, with each machine split into two LPARs servicing the applications selected as best to retain following business integration. There were also several very small LPARs for systems testing, disaster recovery, etc. The development machine acted as DR machine for the production images, with production data mirrored to disk at the development site by SRDF. The following graph is a snapshot of the configuration showing CPU utilisations during a busy mid-morning interval.

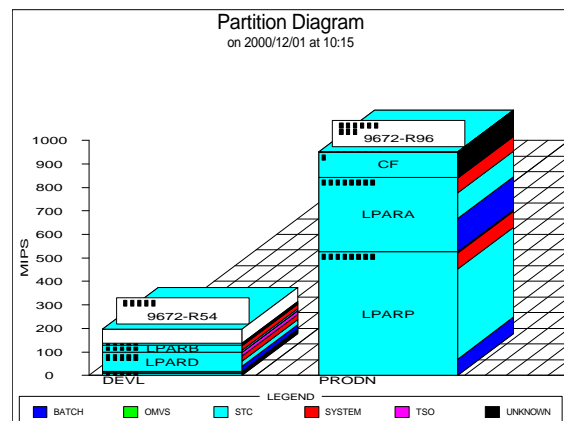


Figure 1 - Original St George hardware

2.3 Performance Architecture

St George assumes it is a fairly typical site in terms of Capacity and Performance requirements. There is a small team responsible for Capacity Planning. A Technical Services Group of systems programmers is responsible for all MVS systems functions, and also includes a separate team responsible for Storage Administration. Yet another team concentrates on Service Level Management. Finally, everyone worries about Performance - whenever a performance problem is identified, that is.

In terms of Performance Management, SMF records are collected on all 4 main LPARs, with the data archived on a development image (via shared DASD where necessary) for post-processing. Typical quantities of data across the 4 images are outlined in the following table:

SMF RECD	TYPICAL DAY		BUSY DAY	
	COUNT	BYTES	COUNT	BYTES
30	230,000	610MB	314,000	750MB
74	11,000	190MB	11,000	190MB
110	406,000	10.4GB	447,000	13.2GB
ALL	7.3M	14.0GB	5.0M	15.5GB

Notes:

- The typical SMF110 record contains details of 33.7 CICS transactions.
- All intervals are 15 minutes, and SMF30's are synchronised with RMF.

2.4 Performance Products in Use

St George systems personnel have the usual range of tools ranging from online monitors to performance databases to modelling software.

2.4.1 OMEGAMON & EPILOG

Used by: Systems Programmers, Capacity Planners, some developers.

Used for: Problem resolution, the main systems programmers' performance tool.

2.4.2 SAS AND MXG

Used by: Capacity Planners, Storage Administrators, TPNS Testing.

Used for: Legacy, unmaintained MXG PDB (data from 1 production and 1 development LPARS). Several small but important "boutique" performance databases. Some non-MXG use of SAS.

2.4.3 EXPETUNE

Used by: Testing team.

Used for: Providing top-twenty and exception reports using SAS/MXG databases as starting point.

2.4.3 MAINVIEW (WAS BEST/1) VISUALIZER

Used by: Capacity Planners, Testing.

Used for: Exception reporting, trending (2 years history in SQL-Server databases), monthly management reporting. Data collected from MVS, UNIX, and NT servers.

2.4.4 MAINVIEW (WAS BEST/1) PREDICT

Used by: Capacity Planning.

Used for: Prediction of (development machine based) TPNS test results in a production environment. Capacity Planning from forecast business growth. Other "what-ifs" concerning hardware and workloads.

2.5 SAS Products Licensed

Perhaps like many other MXG sites a "minimum" of SAS products were licensed on the mainframe; at St George this included: SAS/Base, SAS/FSP, and SAS/Graph. The important point is that SAS was installed on a single development LPAR, and was therefore priced for just the development machine.

3. WHY CHANGE?

The announcement of the IBM zSeries 900 was the catalyst. The zSeries offered a number of hardware benefits, and the possibility of only paying for software (IBM's at the very least) by defined capacity requirements – not by total machine capacity. This all sounded attractive.¹

St George negotiated acceptable Enterprise License Agreements with all vendors except SAS. If St George combined their development and production machines into a single box, SAS would require an upgrade fee and maintenance charge based on total machine size. For St George this would mean licensing based on 1260 MIPS instead of 196 MIPS. Any reader attempting to gauge the upgrade fee required is advised to have a strong cup of tea and several aspirins to hand.

4. OUR ALTERNATIVES

What could we do?

- Drop SAS and MXG entirely?
- Run SAS on the new zSeries?
- Run SAS on a UNIX machine?
- Run SAS on an NT machine?

4.1 Drop SAS and MXG

SAS is a useful and powerful tool – it would be a loss to not have it. SAS would be an indispensable tool if Capacity Planning were ever to increase its involvement in the Performance Analysis area. Also, MXG is the industry standard tool for Performance Analysis and thus known to most visiting consultants, vendors, and any prospective Capacity/Performance staff - losing MXG would marginalise the entire function.

4.2 Run SAS on the zSeries

Price forced us to consider other alternatives. Dropping SAS/GRAPH made little difference to the magnitude of the upgrade fee.

4.3 Run SAS on a UNIX machine

Possible, but we had no UNIX experience, and we assumed UNIX would be more expensive than a PC platform.

4.4 Run SAS on an NT machine

Possible. Capacity Planning already had an NT server to host the BEST/1 Visualizer database. If additional disk space were purchased then MXG

¹ See CMGA 2001 paper "Is Z 4 U?" for a further discussion on zSeries machine usage.

could possibly be accommodated on this server with minimal hardware cost. Software costs were also appealingly low - so much lower that we could afford many additional bells, whistles, toys, and even some analysis options that would greatly enhance the Capacity Planning function.

However, we were worried because we hadn't seen PC-SAS since 1986 (version 6.02), and we didn't know whether processing times would be acceptable.

5. THE CHANGE PROCESS

5.1 User Review

We thought we already knew the main users of SAS, and a few emails soon produced the following list:

User area	Products	Usage
Security	SAS	Adhoc reporting: "Who deleted that dataset?" "When was that contractor here?"
Applications (various)	SAS	Various reports: PROC PRINT; PROC TABULATE;
Storage Admin	SAS, MXG	Disk, tape analysis
Tech Services	SAS, MXG	Special studies only, rarely
Cap Planning	SAS, MXG	Special studies, eg. job class usage
Cap Planning	SAS	End to end response time dbase
Testing	SAS, MXG	CPT ExpeTune exception reporting

A few users used SAS so little that they were happy to simply stop using it. The good people at CPT agreed to port their ExpeTune code to PL/I, thus ensuring continuation of service to the Testing Team. Other users' requirements could be easily answered using REXX or PILOT. Some were already planning to move their data to other products, eg. data warehousing systems. Finally, some users might need *encouragement* to speed their migration ...

5.2 User Exit

The main issue was isolated SAS reporting embedded within applications. SAS was used to identify the use of SAS, and to quantify the effort in moving reports to another solution. The "Divide and Conquer" tactic was adopted: "If your application is the only one continuing to use SAS then your department will have to bear the total costs of the increased SAS license." This worked a treat!

Soon we were able to report that all SAS code could

be ported, converted, or dropped before the delivery of the zSeries machine.

But we still didn't know whether SAS and MXG would run on an NT server ...

5.3 SAS on a PC - research

Would it be viable to run SAS and MXG on another platform? St George cuts over 14 gigabytes of SMF data a day, with 10 gigabytes of SMF 110 CICS data alone. These sounded like huge volumes of data to process with a PC.

Searching the MXG documentation we found Newsletter Twenty-Five that outlines moving MXG to a PC system. While Barry Merrill did not encourage the move he said it was possible, and given the age of the newsletter, 1994, and being optimistic we thought a lot has changed since then.

We then searched through all the CMG proceedings - on CDs and on paper looking for any references of anyone else who had been through this process before us. There was no point in starting if we were following in the steps of Burke and Wills! We found CMG papers written in 1995, with 486 processors.

5.4 SAS on a PC - a trial

In parallel with trying to move others off SAS completely, we decided to run a trial that would give us:

- A quick comparison to mainframe processing speeds
- An indication of download speeds
- Early warning of NT server sizing requirements
- Any problems in converting data

We selected a fairly easy application to migrate: our Branch end to end ("B20") response time database. The code was small (500 lines), the overnight feeds were small (3 MB), and the database was a medium size (600 MB) that would be a test of download capabilities.

6. WHAT IF WE COULDN'T DO IT?

We identified two major risks:

- NT server processing speed; and
- Download (FTP) time.

6.1 NT Server processing speed

Briefly, the NT Server hardware consisted of:

- Dual 866 MHZ PIII processors (not XEON)

- ServerWorks mainboard, 64-bit PCI bus
- 256 MB PC133 memory
- 2x 18GB SCSI disks, 7200RPM, RAID-1 (mirrored) configuration

Assuming that processing times were not excessive, we were already thinking of ways to improve them.

Firstly, the hardware is no slouch, but just like a home PC, it could always be improved! We thought that changing from RAID-1 to RAID-0 (striping) might provide an I/O throughput improvement. We could also experiment with SAS's data compression options. Would more memory speed sort times?

Secondly, we could modify the PDB Build process: reducing data variables kept, filtering records with the IMAC and EXTY members, or splitting into several concurrent builds. The ultimate goal would be near real-time builds occurring hourly throughout the day.

Finally, the trial database builds would be conducted on a quiet machine, but we had to remember that in production the MXG database builds would probably overlap the Visualiser database load and summarisation jobs.

We should also point out that we were also prepared to trial SAS on a UNIX server. If we had to buy a UNIX box it would still be considerably cheaper than running SAS on the new zSeries mainframe.

6.2 Download speed

If the time it took to download data to the server was excessive then we had the following options:

- Compress data prior to download.
- Limit the amount of data that was downloaded – reduce the CICS SMF 110 records.
- Download data in segments throughout the day.
- Consider the use of the newly installed Storage Area Network (SAN) to remove the need to download - both the mainframe and the NT server would see the disk as locally attached.

7. THE TRIAL RESULTS

7.1 Processing Speed

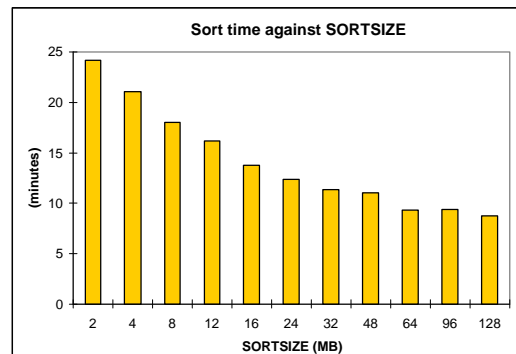
The following table summarises the results:

Task	Mainframe		PC	
	CPU	Elapsed	CPU	Elapsed

Update job	1.3s	2.6s	3.5s	14s
Report job	1.2m	2.7m	3.2m	6.5m

To our great relief (because time was running out) the results were excellent! They showed that at least for small to intermediate data sizes, and for fairly simple processing the PC was (almost) the equal of the mainframe.

An additional test was to sort the database looking for duplicates caused by multiple updates of the same days data (something that shouldn't happen, but that only means that it does). This would be a better test of PC-SAS disk and memory throughputs, and it would also provide a feel for the impact of OPTIONS SORTSIZE. We tested the same code against the downloaded database with several SORTSIZE settings, and recorded the following times:



At this point we have increased the default SORTSIZE to 24MB, and expect users to code larger SORTSIZE up to 64MB, always bearing in mind expected contention with other users, and the 512MB physically available.

7.2 Download Times

Downloads proved to be our biggest nightmare. Normally obtaining 800MB/s across the LAN we were surprised to see speeds between only 20 and 600KB/s from the development images down to the LAN. The existing link is only 10Mb/s and suffers from ADSM traffic overnight. We are eagerly awaiting the implementation of a Gigabit link, which should eliminate this constraint.

Our first and most easily tested solution was to compress the data prior to download. Experiments using ISPZIP, by Australian Systems Engineering, delivered remarkable compression ratios:

Peak hour, on a typical day:			
SMF recd	VBS recfm	Compressed	Saving
CICS	950MB	147MB	84%
non-CICS	5MB	1MB	79%

ALL	955MB	147MB	84%
-----	-------	-------	-----

Transferring 150MB at St George can take as little as 5 minutes during the day, but because of the contentions outlined above, can stretch to 2 hours overnight!

8. A NEW HOME FOR MXG

Following the successful trial we ensured all users of mainframe SAS were migrating their applications, and turned our attention to our own “needs”.

8.1 PC-SAS Software

SAS/Base, SAS/FSP and SAS/Graph we ran on the mainframe, and were the minimum requirement on PC-SAS, but we also bought the following:

SAS/STAT:	Because we desperately want to be statisticians. PROC REG is useful for trend analysis, and simple correlation analysis. One day we might even use PROC CLUSTER.
SAS/CONNECT:	Allows the whole team to access all these wonderful add-ons, the power of the NT server, to process server data at the server, all from their desktops!
SAS/ACCESS ODBC:	To provide direct access to SQL Server databases (like BEST/1 Visualizer).
SAS/OR:	Operations Research, for critical path analysis, reporting on job schedules.
SAS/SHARE:	If we can get near real time updates from the mainframe (say every hour) then we'll need SHARE to allow concurrent updates and reads by users.

We licensed all the above for the NT Server, but also SAS/BASE and SAS/CONNECT for 3 workstations for the Capacity Planning team.

8.2 NT Server Hardware

We have decided to isolate our MXG processing from the BEST/1 Visualizer databases, mostly because this is a St George database management team requirement (that no applications run on database servers). This will also provides more computing power for MXG, and allow us more control over the SAS/MXG applications server. Our SAS/MXG server looks like:

- Dual 866 MHZ PIII processors (not XEON)
- ServerWorks mainboard, 64-bit PCI bus
- 512 MB PC133 memory
- 2x 18GB SCSI disks, 7200RPM, RAID-1 (mirrored) configuration
- 2x 18GB SCSI disks, 7200RPM, RAID-0 (striped) configuration

We locate the MXG databases and MXG work libraries on the striped disk, and use the mirrored

disk for system, application, and user space.

8.3 BUILDPDB Runtimes

Using the standard BUILDPDB job, running against the peak hour of the day (1.3M CICSTRAN observations) the following times are achieved:

Runtime = 13 minutes, 14 seconds
CPU time = 3 minutes, 2 seconds

Using SAS options COMPRESS=YES returns the following times:

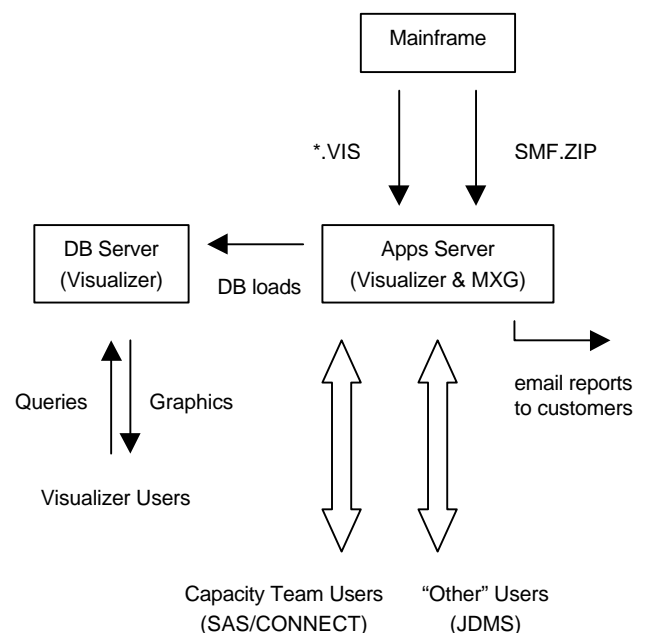
Runtime = 8 minutes, 31 seconds
CPU time = 3 minutes, 49 seconds

So, for the cost of 26% more CPU time, we can reduce runtime by 36% → we increase the %CPU requirement from 23% to 45%.

This is a significant result since for some applications the costs of compression can considerably outweigh the benefits.

8.4 New Performance Architecture

A high level view of the new architecture is:



8.5 PC-SAS Features in Use

The extra functionality available in PC-SAS has allowed us to exploit a number of exciting new areas.

8.5.1 JDMS

A free (yes, FREE) offering from SAS Institute is a lightweight JAVA-based front end for remote computing called JDMS - Java Display Manager

System. JDMS does not require a local copy of PC-SAS, and does not require SAS/CONNECT to be licensed at the server. Whilst not the most attractive front-end, it is functional and cheap, and offers the standard Programming, Log, and Output windows to the user.

We now have several JDMS users - all people outside the Capacity Planning function who couldn't really do without SAS after all!

8.5.2 GROUPWARE INTERFACE

This is SAS Institutes term for interfacing to email systems. We use it to distribute management and exception reports automatically overnight. It required us to set up mail client on the server, but is a highly effective and personal way to communicate with our internal customers. Of course, we also place all reports on a web server for broader availability and archiving.

8.6 TBE (To Be Exploited)

There are still many exciting features of PC-SAS and our new architecture that we have yet to exploit. The only barrier is time!

- Real time (or near real time) updates to the MXG PDBs by using SAS/SHARE to allow background updates.
- Exception reporting, with reports emailed automatically to recipients.
- Event notification, direct to team by email.
- Other SAS/ODS features.
- Effective download capability courtesy of Gigabit link between mainframe and LAN.

9. CONCLUSION

If the reader suspects that we are SAS and MXG bigots, then that is because we are! And even more than before.

The cost of the PC platform has always been attractive, but only recently has PC performance and capacity become capable of processing a mainframe's quantity of data in realistic timeframes. In St George's case we have acceptable run times, for prices guaranteed to excite any manager. The cost of purchasing and implementing the PC-based solution was well within the alternative upgrade fee. Annual maintenance for the new solution is a fraction of what would be required for the increased mainframe!

MXG is more accessible than before, both because we manage the databases more carefully, and because we have a richer output environment.

We have far greater flexibility with output - although we licensed SAS/GRAPH on the mainframe, we seldom used it easily because we had no directly attached graphic output device. With PC-SAS we are spoilt for choice and had to argue over which output formats to use!

Although at the time St George did not enjoy being forced to migrate, with the benefit of hindsight we would like to thank SAS for providing us the opportunity to evaluate what we were getting from our "legacy" system, and for opening our eyes to the world of PC-SAS.

APPENDIX – BIBLIOGRAPHY

A1 Running MXG on NT

95INT019 How to process SMF/RMF data in a Windows and Windows NT environment with performance comparisons to mainframe processing
Author: Turner, George W & Johnson, Steven A

95CEU023 Executing MXG and other SAS applications under OS/2, Windows and UNIX
Author: Merrill, HW Barry

A2 Processing large amounts of SMF data

98INT5401 40,153,273,652
Author: Hopf, Chuck, Merrill, HW Barry

A3 Basic Reading

Merrill's Expanded Guide (MXG) to Computer Performance Evaluation Using the SAS System, Merrill Consultants, 1984

MXG NEWSLETTER TWENTY-FIVE

A4 z/Series Implementation

Is Z 4 U?
Published: CMGA2001
Author: David Vasey.