Performance Test Suite Results for SAS® 9.1 Foundation on the IBM® zSeries™ Mainframe

The Power to Know®

A SAS White Paper

Table of Contents

The SAS and IBM Relationship

Since its founding in 1976, SAS has had a technological relationship with IBM. This relationship has expanded to include joint solution development and marketing. Partnering with IBM is a true win-win situation for both SAS and IBM. SAS benefits from IBM's experience in integrating technologies, and IBM aligns itself with a market leader in business intelligence. As a result, joint customers get the most complete set of intelligence and analytical solutions on the market.

Introduction

This paper presents the performance results that were obtained from two, complementary, batch test suites that were run using SAS 8.2 and SAS 9.1. SAS uses batch test suites to monitor the performance of commonly used features of SAS Foundation software during the development cycle and between releases. These features include DATA step processing, some Base SAS procedures, some statistical procedures, and two Data Mining procedures.

One set of test results shows performance at a customer job level, and the other set shows the performance test results for DATA steps and SAS procedures. Customers will find the procedure-oriented results useful in estimating how SAS 9.1 will benefit their existing SAS production jobs that run under OS/390 or z/OS.

Customer Jobs Test Suite

The first test suite is a customer jobs test suite that consists of batch jobs from customer sources. The characteristics of this test suite are described in Table 1.

Job Name	Description	Product/Category
CUSTJ1	Create a data set from input that's contained in job (2,103 observations), manipulate data with DATA step, run PROC GLM	SAS/STAT
CUSTJ2	Use census data from California (CA – 2,037,765 observations), Colorado (CO – 254,039 observations), and Texas (TX – 1,276,240 observations) to create data sets for further processing	Base SAS
CUSTJ3	Sort data and create simple and composite indexes	Base SAS
CUSTJ4	Run PROC FREQ, PROC SUMMARY, and PROC SORT on reporting data sets	Base SAS

continued

Table 1. Customer Jobs Test Suite Characteristics

Job Name	Description	Product/Category
CUSTJ5	Run PROC SORT and PROC UNIVARIATE on reporting data sets	Base SAS
CUSTJ6	Run PROC MDDB to create MDDB on reporting data sets	Base SAS
CUSTJ7	Create a test data set that contains 390,625 observations and run PROC MEANS, PROC UNIVARIATE, and PROC TABULATE on data	univariate stat
CUSTJ8	Create a test data set that contains 6,890,625 observations, and run PROC MEANS, PROC UNIVARIATE, and PROC TABULATE on data	univariate stat
CUSTJ9	Create SQL view	Base SAS
CUSTJ10	Create SQL view	Base SAS
CUSTJ11	Create SQL view	Base SAS
CUSTJ12	Create SQL view	Base SAS

Table 1 (continued). Customer Jobs Test Suite Characteristics

SAS Generated Test Suite

The second test suite is a set of batch jobs that are generated at SAS for testing specific performance characteristics. Usually, this test suite uses artificially generated input data sets that include different combinations of data set sizes and observation widths. The characteristics of this test suite are described in Table 2.

Job Name	Description	Product/Category
GENJ01	Create a data set that has 5 million observations; create 2 indexes during data set creation; append 2.5 million observations; and update indexes at the same time	I/O
GENJ02	Create a copy of the input data set in the same library (24,816,204 observations); create an index of single variable; append 1% (220,000 observations) to copy (25,036,204 observations); DATA step and SQL query for single value contained in index (382,341 observations); DATA step and SQL query for multiple values contained in index (14,989,444 observations); delete data set copy and index	SAS data set query

continued

Table 2. SAS Generated Test Suite Characteristics

Job Name	Description	Product/Category
GENJ03	Create data sets that contain blank character variables and one numeric variable (all data sets have 100,000 observations) SKINNY – 128 Bytes MEDIUM – 1024 Bytes WIDE – 10240 Bytes After data sets are created; do DATA _NULL_ random point= reads based on random number for each data set; no output created	SAS data set I/O
GENJ04	Create data sets that contain blank character variables and one numeric variable (all data sets have 1,700,000 observations and are compressed (COMPRESS=YES)) SKINNY – 128 Bytes MEDIUM – 1024 Bytes WIDE – 10240 Bytes After data sets are created; do DATA _NULL_ random point= reads based on random number for each data set; no output created	SAS data set compression
GENJ05	Create data set that has multiple indexes; create indexes (8 composite and 6 simple) along with data set creation (3,200,000 observations); do multiple DATA _NULL_ steps to read various compound conditions; select 8000, 4000, 3200, and 160 observations	SAS indexed data set performance
GENJ06	Create data set (5,000,000 observations) that has a simple index (values in index should be currently unique, but not enforced); append 1,000,001 observations; update index as added; index values that are no longer unique	SAS indexed data set performance
GENJ07	Create data sets that contain blank character variables and one numeric variable (all data sets have 100,000 observations) SKINNY – 128 Bytes MEDIUM – 1024 Bytes WIDE – 10240 Bytes After data sets are created, do DATA _NULL_ sequential reads based on SET statement; no output created	SAS data set sequential I/O performance
GENJ08	Create data sets that contain blank character variables and one numeric variable (all data sets have 5,000,000 observations and are compressed (COMPRESS=YES)) SKINNY – 128 Bytes MEDIUM – 1024 Bytes WIDE – 10240 Bytes After data sets are created, do DATA _NULL_ sequential reads based on SET statement; no output created	SAS compressed data set sequential I/O performance

continued

Table 2 (continued). SAS Generated Test Suite Characteristics

Job Name	Description	Product/Category
GENJ09	Create data set (677,375 observations) that contains dates; repeat same WHERE statement 240 times; no indexes	SAS data set query
GENJ10	Import transport file; use DATA step to assign classification to variables; run PROC FREQ, PROC DMREG, PROC NEURAL, PROC TRANSPOSE, and PROC SORT on data	SAS/STAT
GENJ11	Create data set from input that's contained in job; run PROC MIXED	SAS/STAT
GENJ12	Create data set from transport data set that contains 18,293 observations; extract 5% random sample (8,993 observations); run PROC REG and PROC LOGISTIC	SAS/STAT
GENJ13	Create data sets from transport files (File A contains 40,895 observations, File B contains 3,328 observations, and File C contains 3,060 observations); run PROC TRANSPOSE on each data set, match-merge data, run PROC PHREG on resulting data set	SAS/STAT
GENJ14	Create input data set from random number seed (2,160 observations), run PROC GLM	SAS/STAT
GENJ15	Create input data set from random number seed (1,000,000 observations); use DATA step to classify variables; run PROC MEANS, PROC TRANSPOSE (on MEANS output), and PROC RANK on original input data set	SAS/STAT

Table 2 (continued). SAS Generated Test Suite Characteristics

Test Environment

The customer jobs and SAS generated test suites were run on an IBM z/800-003 with an IBM 2105-800 ESS. The setup had the following attributes:

- Four 2-GB/sec FICON channel connections
- Parallel Access Volume (PAV) support
- Arrays across Loops (AAL) feature
- 3390-3 logical volumes in Raid5 rank sets on 72-GB physical volumes

While tests were run, other work on the machine was insignificant, which permitted gathering repeatable elapsed time and CPU time results. Even though these conditions (for example, running one job at a time on a lightly loaded system) are not usually found at customer sites, the tests are useful for measuring repeatable elapsed times and determining best CPU times.

Significant SAS 9.1 Performance Improvements

Parallelism Using Threaded Kernel (TK) Architecture

The most extensive architectural change in SAS 9.1 is the introduction of Threaded Kernel (TK) architecture, which supports parallel execution of multiple units of work. TK contributes to improved scalability by enabling SAS tasks to take advantage of multiple CPUs on a processor. Each thread is an independently dispatched unit of work. SAS procedures can take advantage of the TK services in addition to the single-threaded services of the MVA SAS Supervisor.

Figure 1 illustrates how TK architecture works.

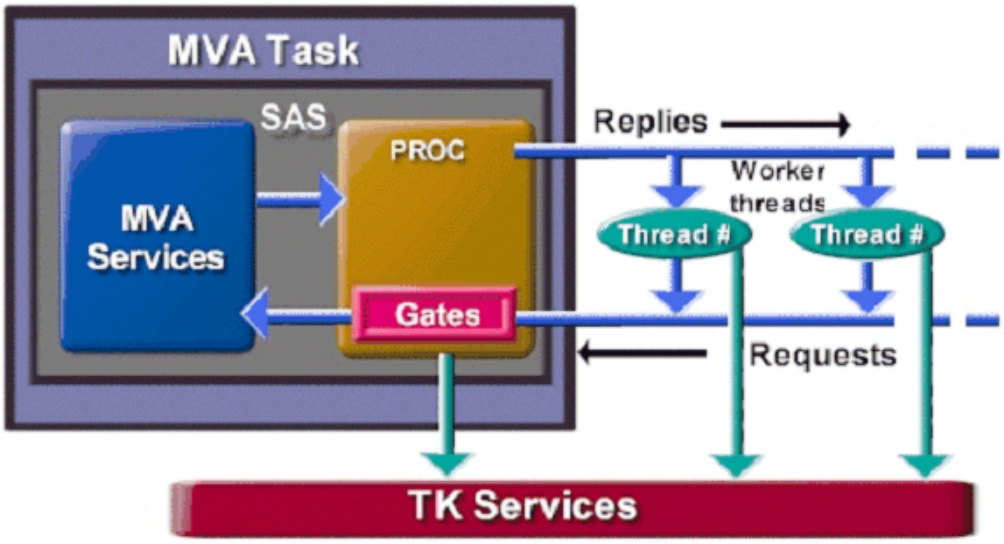

Figure 1. Threaded Kernel (TK) Architecture

How Is SAS TK Architecture Implemented?

SAS TK architecture is implemented for z/OS and OS/390, R2.10, via UNIX System Services (USS). Because USS is used, all SAS tasks (batch jobs, started tasks, and TSO commands) must run under RACF profiles that have an OMVS segment that specifies a user ID (UID) and a group ID (gid). The SAS WORK library can be stored in an HFS or zFS directory under USS.

Note: Threaded procedures require a special UTILLIB to handle multi-threaded utility I/O. The UTILLIB should be implemented in an MVS SAM-E data set or in a USS directory. When the SAS WORK library is in a USS directory, utility files are placed there by default.

SAS procedures that have been thread enabled are listed in Table 3.

Product	Procedures	
Base SAS	MEANS	SQL
	REPORT	SUMMARY
	SORT	TABULATE
SAS/STAT	GLM	LOESS
	REG	ROBUSTREG
SAS Enterprise Miner	DMINE	DMREG

Table 3. Thread Enabled SAS Procedures

Performance Test Suite Results

Customer Jobs Results

Performance can be viewed from the perspective of specific customer tasks or from the perspective of jobs that use a combination of SAS DATA steps and SAS procedures to perform a task. In this study, the performance improvement from the customer's perspective is directly related to the application tasks that are being executed. Table 4 shows the results from customer jobs.

Job Name	Elapsed Time				CPU Time			
	SAS 8.2	SAS 9.1	Change	Ratio	SAS 8.2	SAS 9.1	Change	Ratio
CUSTJ1	1.2	0.95	-0.25	0.79	0.75	0.48	-0.27	0.64
CUSTJ2	105.25	95.09	-10.16	0.90	124.36	111.34	-13.02	0.90
CUSTJ3	25.19	25.72	0.53	1.02	9.53	10.78	1.25	1.13
CUSTJ4	25.11	22	-3.11	0.88	17.53	19.1	1.57	1.09
CUSTJ5	8.9	9.56	0.66	1.07	4.1	4.27	0.17	1.04
CUSTJ6	19.4	20.56	1.16	1.06	10.27	10.58	0.31	1.03
CUSTJ7	5.27	3.85	-1.42	0.73	4.1	5.01	0.91	1.22
CUSTJ8	89.2	62.66	-26.54	0.70	80.15	96.14	15.99	1.20
CUSTJ9	10.58	9.63	-0.95	0.91	3.33	3.44	0.11	1.03
CUSTJ10	10.59	9.68	-0.91	0.91	3.3	3.46	0.16	1.05
CUSTJ11	16.72	16.11	-0.61	0.96	11.47	11.82	0.35	1.03
CUSTJ12	17.19	16.13	-1.06	0.94	12.46	12.41	-0.05	1.00

Table 4. Customer Job-Oriented Test Results

Figure 2 shows the elapsed and CPU time ratios (SAS 9.1.3 : SAS 8.2) for each of the customer jobs. A ratio of less than 1.0 indicates improvement in SAS 9.1.3 compared to SAS 8.2.

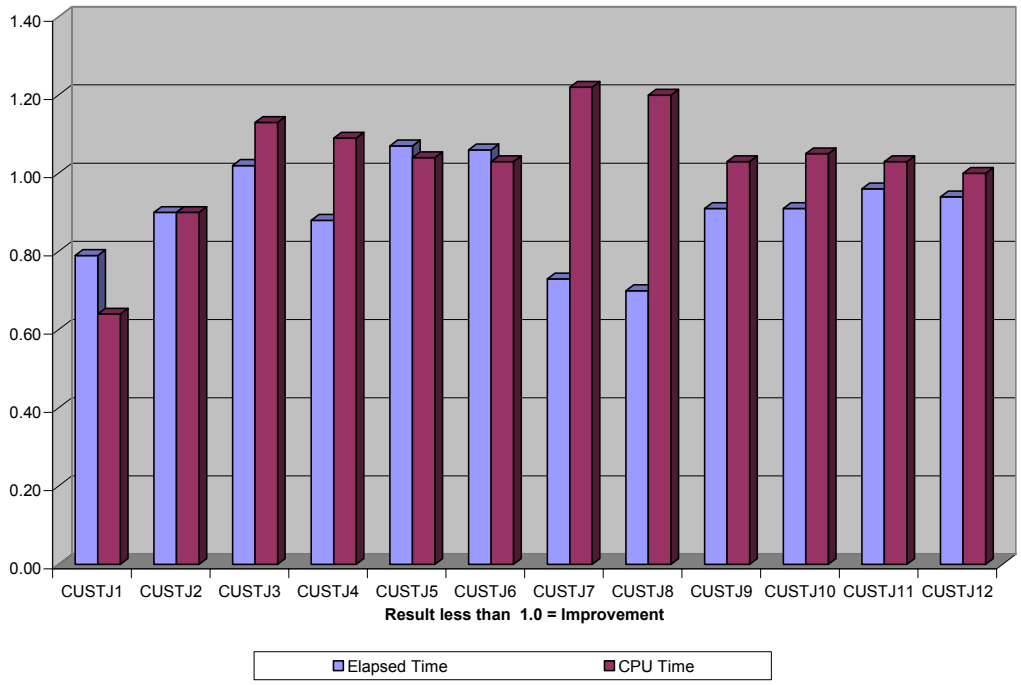

Figure 2. Customer Jobs Performance Test Results Using SAS 9.1.3 and SAS 8.2

DATA Steps and SAS Procedures Test Results

Performance can be viewed from the perspective of each SAS feature (for example, DATA steps and procedures) that's being used in the customer jobs. These results are very useful for SAS development groups, but they also give customers the information they need to estimate the impact of SAS 9.1 performance on their applications.

Table 5 shows the results from customer test streams.

SAS PROC or DATA Step	Elapsed Time				CPU Time			
	SAS 8.2	SAS 9.1	Change	Ratio	SAS 8.2	SAS 9.1	Change	Ratio
APPEND	47.59	38.35	-9.24	0.81	34.55	36.11	1.56	1.05
CIMPORT	51.74	49.49	-2.25	0.96	19.25	17.81	-1.44	0.93
DATA	12.1	7.55	-4.55	0.62	8.15	7.87	-0.28	0.97
DATASETS	16.18	16.23	0.05	1.00	18.56	8.39	-10.17	0.45
DELETE	0.5	0.5	0	1.00	0.3	0.29	-0.01	0.97
DMDB	12.22	10.01	-2.21	0.82	8.13	6.89	-1.24	0.85
DMREG	377.33	311.92	-65.41	0.83	303.75	133.51	-170.24	0.44
FREQ	13.37	11.91	-1.46	0.89	6.73	6.43	-0.3	0.96
GLM	7.07	4.76	-2.31	0.67	5	2.9	-2.1	0.58
LOGISTIC	1080.88	858.85	-222.03	0.79	588.64	381.96	-206.68	0.65
MDDB	53.21	48.85	-4.36	0.92	23.13	22.99	-0.14	0.99
MEANS	47.29	30.84	-16.45	0.65	25.72	29.83	4.11	1.16
MIXED	648.96	432.76	-216.2	0.67	566.08	376.13	-189.95	0.66
NEURAL	2116.78	2331.68	214.9	1.10	1836.9	1219.56	-617.34	0.66
PHREG	819.33	822.4	3.07	1.00	167.27	169.88	2.61	1.02
RANK	461.27	637.22	175.95	1.38	295.26	315.45	20.19	1.07
REG	5.77	3.71	-2.06	0.64	5.07	5.88	0.81	1.16
SORT	16.85	13.02	-3.83	0.77	5.2	4.85	-0.35	0.93
SQL	58.87	59.8	0.93	1.02	33.35	33.43	0.08	1.00
SUMMARY	53.96	45.31	-8.65	0.84	32.15	35.61	3.46	1.11
TABULATE	39.95	17.39	-22.56	0.44	19.39	32.55	13.16	1.68
TRANSPOSE	0.02	0.03	0.01	1.50	0.01	0.01	0	1.00
UNIVARIATE	77.8	75.61	-2.19	0.97	52.75	51.64	-1.11	0.98

Table 5. Comparison of SAS 8.2 and SAS 9.1 on Customer Jobs

DATA Step Improvements

There have been significant DATA step compilation and execution improvements in SAS 9.1, especially for large DATA steps such as DATA steps in MXG software from Merrill Consultants. MXG software's large BUILDPDB DATA step shows a 30% reduction in CPU compilation time and a 12% reduction in CPU execution time.

SAS Procedure Improvements

Many of the vector mathematics routines that are used by SAS statistical procedures were re-written for SAS 9.1 to better exploit the zSeries pipeline architecture. As a result, reductions in CPU time of 25% to 36% have been seen in procedures such as GLM, LOGISTIC, DMREG, NEURAL, and MIXED.

Figures 3, 4, and 5 show the elapsed and CPU time ratios (SAS 9.1.3 : SAS 8.2) for the DATA step and all the PROCs that were used in the tests. A ratio of less than 1.0 indicates improvement in SAS 9.1.3 compared to SAS 8.2.

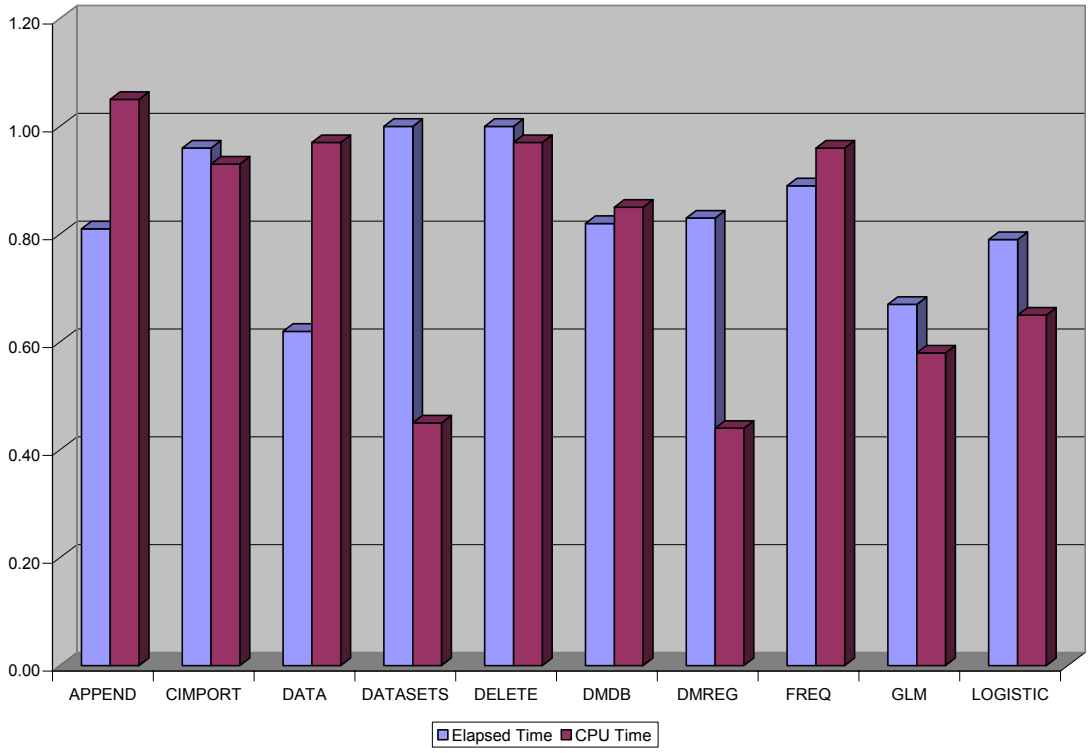

Figure 3. Test Results for Ten SAS Procedures Comparing SAS 8.2 and SAS 9.1.3

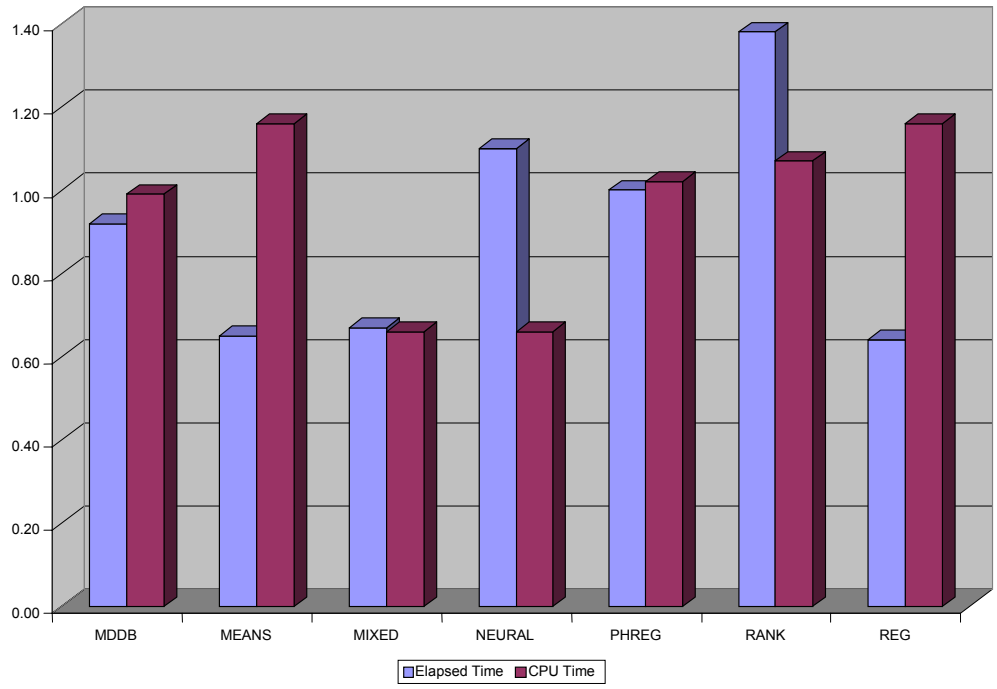

Figure 4. Test Results for Seven SAS Procedures Comparing SAS 8.2 and SAS 9.1.3

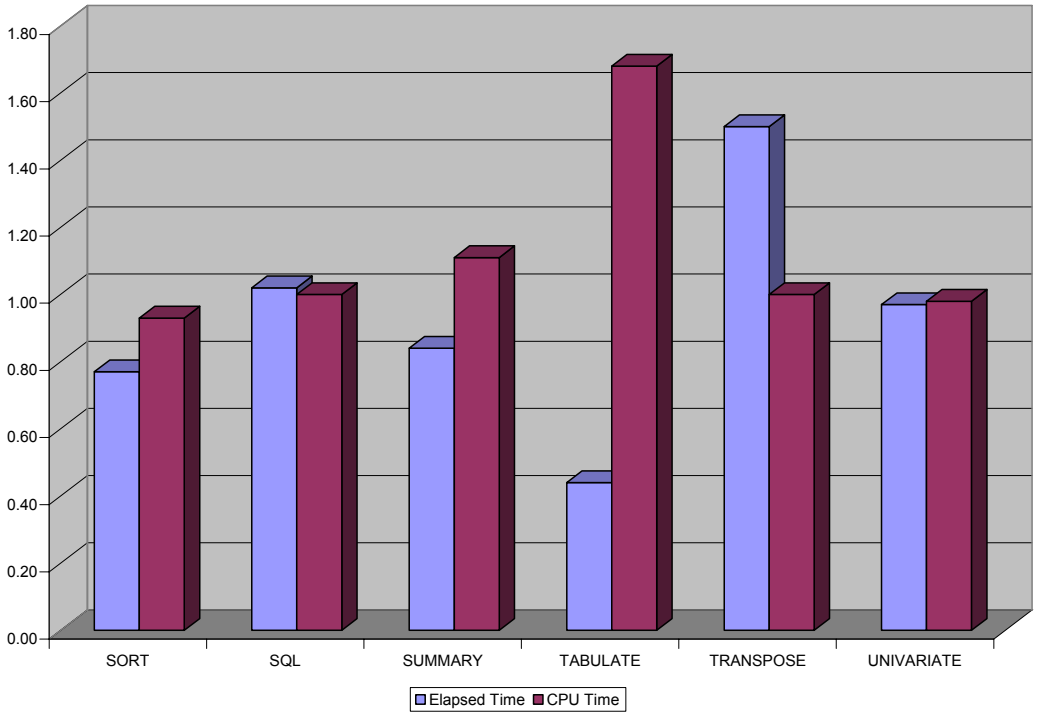

Figure 5. Test Results for Six SAS Procedures Comparing SAS 8.2 and SAS 9.1.3

Graphs Showing DATA Steps and SAS Procedures Improvements

The graphs in this section illustrate the principle "Your Mileage Might Vary". When the data for individual SAS DATA steps is plotted in a box-and-whiskers format, there is a considerable range in the results.

For the DATA step, the results vary based on the size of the DATA step (that is, the amount of code to be compiled) versus the amount of data being processed plus the features used.

For SAS procedures, the results vary based on the features that are used, the volume of data, and the shape of the data (for example, number of distinct class variables).

For each DATA step and each procedure, the box-and-whiskers plots show the following:

- each data point (which is the ratio of SAS 9.1.2 time to SAS 8.2 time; therefore, data points that are less than 1.0 indicate an improvement in SAS 9.1.2 over SAS 8.2)
- the 25th, 50th, and 75th percentiles (bottom, top, and line inside the box, respectively)
- 1.5 times the 25th to 75th percentile range (the whiskers).

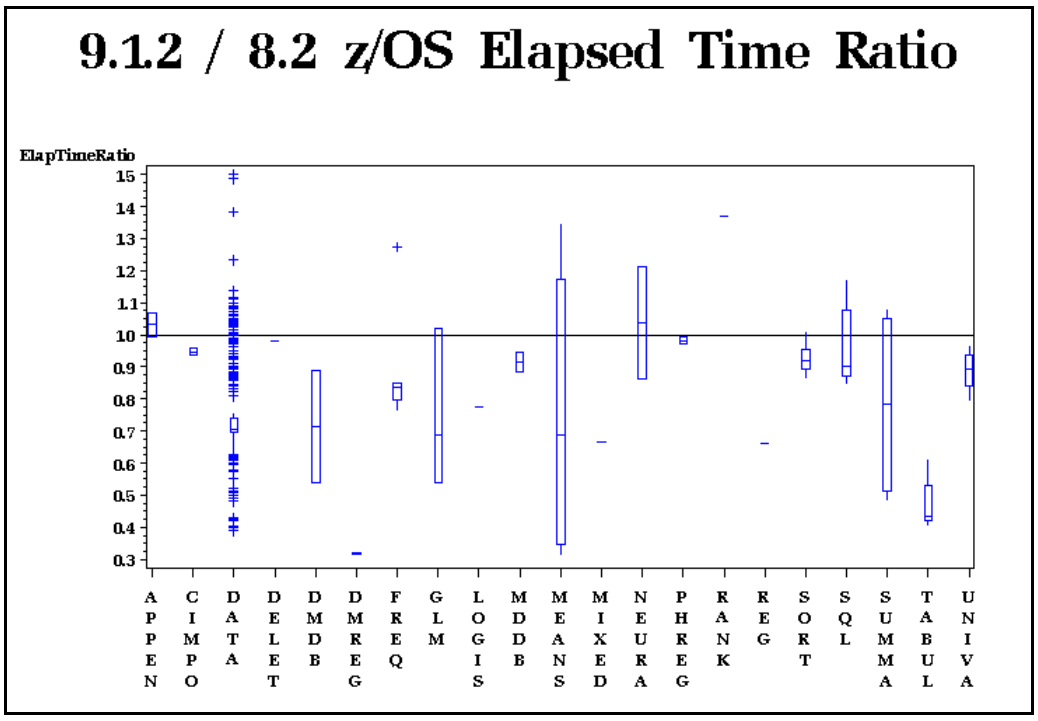

Figure 6. Graph Showing Elapsed Time Ratio

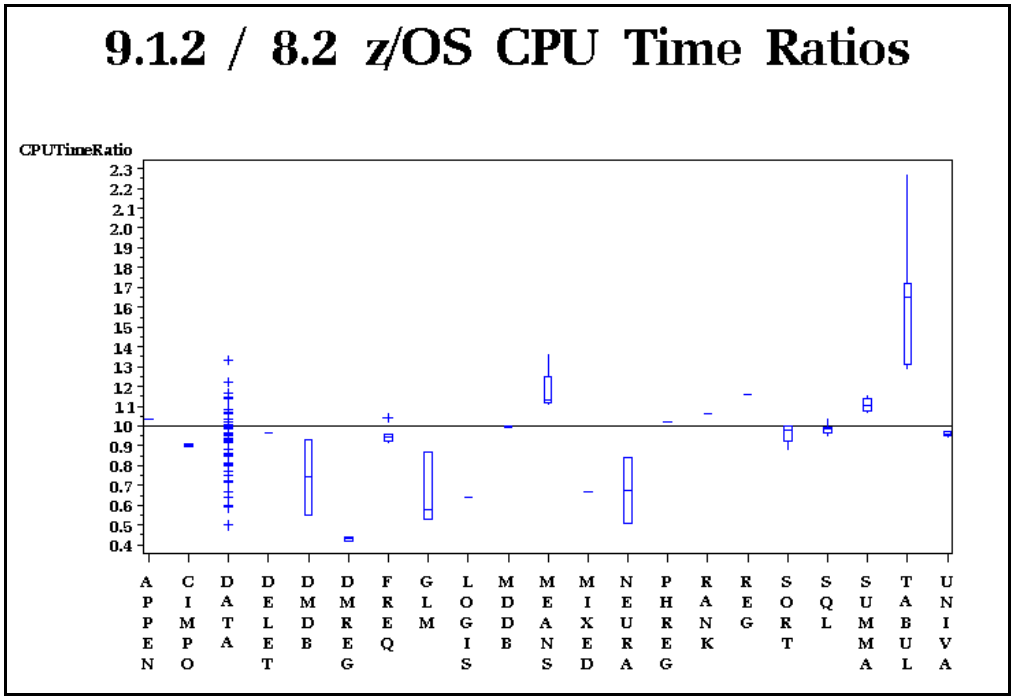

Figure 7. Graph Showing CPU Time Ratios

Summary

SAS Foundation components in SAS 9.1 exhibit both elapsed time and CPU performance improvements over their counterparts in SAS 8.2 in a wide range of test cases. The biggest performance improvements are seen when using the SAS procedures that take advantage of the TK architecture in SAS 9.1 and the statistical procedures that include the improved vector math functions. At the same time, there are significant performance variances that are based on the exact nature of specific DATA and PROC steps and the characteristics of the data.

In addition, customers who have large DATA steps will see significant performance improvements, especially if the number of lines of code in the DATA step is high and the number of observations that are being processed is low.

Contact Information

Dan Squillace, Mainframe Support Manager
SAS Institute Inc.
SAS Campus Drive
Cary, NC 27513
Phone: 919-677-8000
Fax: 919-677-1444
e-mail: Dan.Squillace@sas.com

World Headquarters
and SAS Americas
SAS Campus Drive
Cary, NC 27513 USA
Tel: (919) 677 8000
Fax: (919) 677 4444
U.S. & Canada sales:
(800) 727 0025

SAS International
PO Box 10 53 40
Neuenheimer Landstr. 28-30
D-69043 Heidelberg, Germany
Tel: (49) 6221 4160
Fax: (49) 6221 474850

www.sas.com

 317268US.305