

Performance Management from Desktop to Data Center

Jeff Creech and Chris Williams, SAS Institute, Inc.

Abstract

In keeping pace with changing technologies as more systems become complex and diversified, the enterprise faces a revolution in user technologies and systems. The challenges facing the enterprise involve how to best manage changing computer resources as efficiently and effectively as possible. To answer the demand for reliable and cost-effective performance management tools, the SAS® System offers a repertoire of distributed management tools for all hardware platforms. This paper presents how easily SAS/CPE® (Computer Performance Evaluation) tools can readily access local and remote performance management data through a customizable graphical user interface for performance analysts to access, manage, analyze and present system information, and tune system and network performance.

What is distributed performance management?

Throughout the 1980s and early 1990s, users of information technology have chosen a platform that best meets their needs from a price and performance standpoint. There are a number of applications where mainframes remain the price performance leaders, and others where UNIX machines, PCs, and OpenVMS systems are cost effective. Given the influx of heterogeneous hardware environments, applications needs and user requirements, many IS departments are challenged to support all environments to satisfy the diverse population of users while reducing the cost of doing business.

The challenge of delivering information throughout an organization across heterogeneous environments is effected as budgets tighten and demands on computing resources grow, causing several issues to emerge. The most mission-critical resource for an organization is the network. For most companies with a small team of performance analysts responsible for multiple operating systems, the challenge is how to easily manage and utilize performance management data from a single platform. The shift from managing just mainframe systems to encompassing desktop and UNIX server platforms has produced a new paradigm in computing called distributed performance management.

Today, most organizations rely on a variety of computing environments connected via a network pushing cooperative performance management to the forefront of organizational directives. The goal of performance management is mainly economical, as the enterprise determines how to best maximize hardware resources to enhance user productivity, while reducing the cost of doing business. The business of performance management is how to effectively manage the diverse resources required to deliver applications and serve the user base. The goals of distributed performance management are derived from the elements intrinsic to the challenge of effectively managing heterogeneous computing resources driven by a variety of users:

- Maintain mainframe legacy systems as data repositories and online transaction processing (OLTP) systems for robust data structures.

- Manage Open Systems server-based technologies with heterogeneous data sources locally and remotely for decision support and ad hoc applications.
- Employ distributed resource management through a multi-platform, heterogeneous environment to achieve cross-platform interoperability.
- Support user-based processing effectively to positively enhance productivity and performance throughout diverse computing resources.
- Monitor baselines of systems performance in order to set and maintain acceptable service levels for the end-user community.

What steps are involved in managing distributed computing resources?

Effectively managing distributed computing resources involves (1) gaining transparent access to large amounts of system data, (2) using tools to massage data into a manageable format for reporting purposes, and (3) seamless report generator for producing images to use for identifying problem areas in:

- CPU utilization
- response time
- down time
- network collisions
- re-transmission rate
- DASD consumption
- workload characterization

The SAS System has long been the de-facto standard in offering mainframe tools that performance analysts and capacity planners use to answer commonly posed questions like:

- Is the job load spread well throughout the entire 24 hour day?
- Are there I/O or memory constraints preventing effective utilization of my multi-million dollar CPU?
- Are my most utilized volumes spread well across my controllers, and am I making good utilization of my fastest disk strings?

The advent of workstation technologies and client-server strategies challenged the SAS System to answer the needs across the entire enterprise, addressing heterogeneous computing and application requirements, and managing large amounts of diverse data sources for a diverse group of users. Using the SAS System's proven mainframe computer performance technology as a foundation, the SAS System offers a complete spectrum of tools for all facets of performance management of a multi-platform, heterogeneous environment. The SAS/CPE product line, available for MVS, OpenVMS (VAX and Alpha), and Open Systems platforms ranging from Sun's SPARC/Solaris to IBM RS/6000 (AIX) and Hewlett-Packard's HP9000 (HP-UX) assists in data management, analysis and presentation of distributed performance management data.

Distributed Performance Management With the SAS System

Tackling distributed performance management is possible by using SAS System tools that allow performance analysts to study historical computer resource use in hopes of predicting future resource constraints or how to best tune system resources to assist in business justifications for upgrades or new equipment when the system can be tuned no more. Understanding how the SAS System approaches distributed performance management is the first step towards effectively managing an organization's computing resources.

SAS/CPE provides a consistent user interface allowing an analyst familiar with one system to be proficient on another operating system with little re-training. Providing analysts with tools that offer easy movement between operating systems for performance management tasks, allows more time to concentrate on data idiosyncrasies vs. navigating management and presentation tools.

SAS/CPE collects data into a Performance Data Base, or PDB, consisting of SAS Data sets and data views to be portable to other platforms, allowing for centralized storage or reporting. The PDB architecture is defined by a data dictionary that determines (1) which metric or system record is to be read into the PDB, (2) how to filter invalid data points, (3) how data conversions can produce useful units of measure (i.e., convert the counter for the number of disk accesses on a device into a rate per second), (4) how to enforce cross-platform consistency using a standard datetime format and other data cleanup processing.

Once data consistency is achieved, generating customized reports and graphs through an innovative point and click interface offers analysts the ease of creating ad-hoc presentation tools for both interactive and batch environments. The SAS System performance management tools were developed keeping in mind that an analyst's goal is to concentrate on analyzing data for performance tuning vs. developing presentation tools. These performance management tools produce graphics output on the host where the data lives and, using portable SAS catalogs, can be shared or moved to other host systems.

Embracing distributed performance management involves understanding that distributed systems are becoming the norm among computing environments, and that restricting data management to a single system is not conducive to maintaining flexible and innovative hardware configurations. Understanding the need to embrace distributed systems, the SAS System's remote data access strategy offers remote computing services, including data transfer and remote library services. The advantage of accessing performance data with the SAS System's compute services can reduce network traffic by utilizing the hardware, software, and CPU cycles on the remote environment. Remote library services allow transparent access to remote data, guaranteeing the most current and reliable data image available to an application without maintaining a local copy of data.

With the groundwork laid, let's explore how the SAS System and SAS/CPE might be used to address some of the following real-life scenarios.

SAS/CPE and Remote Compute Services

In this example, a manufacturing firm's performance analysts have been using MVS for 15 years and are now responsible for managing a large Sun server running a production Oracle database. They wish to manage the Sun from MVS by submitting report requests from MVS to the Sun server and display the results on the mainframe. This scenario allows analysts to use one central platform to control the reporting of multiple platforms vs. logging into the UNIX system each day to look at reports and/or graphs.

This scenario utilizes the SAS System's remote compute services, which allow processes to remotely compute and then to return computing results to the local system. Integral to remote computing services is the script file containing commands to initiate a SAS session on UNIX, i.e., logs onto the UNIX system and invokes a SAS session. The following steps illustrate how to take advantage of the SAS System's remote computing services to monitor performance management from a central location:

First, in the display manager's program editor window, submit the following code (uppercase=static for SAS, lowercase=site-specific)

```
filename rlink 'MVSID.CONNECT.SCRIPTS(TCPUNIX)';
options comamid=tcpc;
```

On the command line type and submit:

```
signon INTERNETHOSTNAME
```

Next, to execute collection code on UNIX, in the display manager's program editor window, enter the following code:

```
%inc 'COLLECTCODE'
```

'COLLECTCODE' is an example of SAS code which processes a compressed SunNet Manager raw file containing data for host1 through host4:

```
%let retcode=.;
%cpstart(mode=batch,
  access=write,
  pdb=PATHNAME/PDBXXX,
  _rc=retcode;
%put cpstart retcode=&retcode;

%csprocs(pipe,'uncompress -c /rawdatalogfile'
  snmhp snmioc snmltp snmlup snmpgs snmrnc snmrn
  collectr=sunetmgr,
  subset=machine in ('host1','host2','host3','hos
  _rc=retcode);
%put csprocs return code is &retcode;
```

On the command line type and submit:

```
rsubmit
```

Next, to execute reduction code on UNIX, in the display manager's program editor window, enter the following code:

```
%inc 'REDUCECODE'
```

'REDUCECODE' is an example of SAS code which reduces all tables in the detail level to the levels specified in the data dictionary for the PDB:

```
%let retcode=.;
%cpstart(mode=batch,
  access=write,
  pdb=PATHNAME/PDBXXX,
```

```
_rc=retcode;
%put cpstart retcode=&retcode;
%cpreduce();
```

On the command line type and submit:

```
rsubmit
```

Next, to execute reporting code on UNIX, in the display manager's program editor window, enter the following code:

```
%inc 'REPORTCODE'
```

'REPORTCODE' is an example of SAS code which produces a One-Y axis plot of one or more variables over time:

```
%let retcode=.;
%cpstart(mode=batch,
  access=write,
  pdb=PATHNAME/PDBXXX,
  _rc=retcode;
%put cpstart retcode=&retcode;
%cpplot1(datasetname,var1,lab1,var2,lab2);
```

On the command line type and submit:

```
rsubmit
```

Now, the nightly tasks of processing daily raw data, reduction of the PDB, and running the standard nightly reports can be submitted from the MVS system and executed on the UNIX system, and the graphics catalog can be transferred to MVS using SAS/Connect's file transfer capability. To display graphics output during transfer:

First, initialize the MVS graphics device by locally submitting the following goptions statement:

```
goptions device=DEVICENAME;
```

Next, remotely submit statements to transfer the graphics output. Enter the following code locally in the display manager's program editor window:

```
goptions device=grlink display;
proc greplay nofs igout=work.gseg;
  replay _all_;
quit;
```

At the command line type and submit the following:

```
rsubmit
```

These steps not only display the graph locally, but also store the image in work.gseg on MVS for replaying of UNIX image(s) at a later time.

SAS/CPE and Remote Library Services

A major airline has a large investment in MVS and UNIX systems with separate capacity planning management groups. Sometimes, each group needs to share performance data to perform impact studies of client/server applications. One day in particular, the UNIX capacity planning group is aware of a custom application that reconciles data changes between DB/2 on MVS and Sybase on the Sun that starts at 2:00 AM. The group responsible for the data would prefer to reconcile databases every other hour and, prior to changing the system schedule, both MVS and UNIX capacity planning groups want to know the impact of this change.

Using SAS/CONNECT's remote library services, the analysts can conduct an impact study with reports and graphs generated from SAS/CPE for MVS to examine how changing the current application load will affect long and short term performance of the system. As a test plan, the analysts will offload processing from the MVS system, using the UNIX machine as the local platform where the majority of work takes place. The following steps illustrate how to take advantage of the SAS System's remote library services to access remote data and compute tasks locally as a prototype of change to the current system schedule.

First, in the display manager's program editor window on UNIX, submit the following code: (uppercase=static for SAS, lowercase=site-specific)

```
filename rlink 'TCPTSO.SCR';
options comamid=tcpc;
```

On the command line, type and submit:

```
signon MVSHOSTNAME
```

Next, to surface MVS on the local UNIX system, submit the following libname statement in the program editor window:

```
libname mvslib 'USERID.DATA.LOCATE'
server=MVSHOSTNAME;
```

Using SAS/CPE for MVS, the following screen (Figure A) allows analysts to access the MVS PDB and combine this data with the local UNIX PDB:

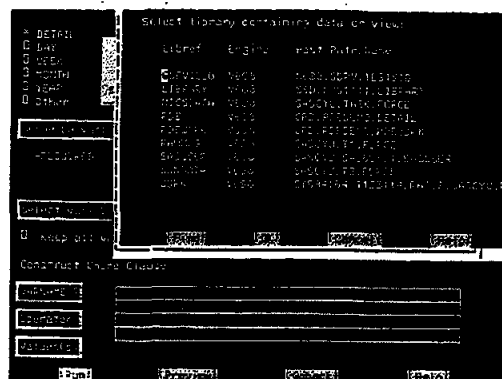


Figure A

Now, both local and remote data is available for local reporting purposes on the UNIX system.

Performance Management and the Executive Information System

A large financial services company has a robust network of PC clients running Novell NetWare and Sun servers. Performance analysts prefer to use the Sun server to manage and analyze performance data collected using SunNet Manager. In addition to SunNet Manager collecting data from their UNIX and PC networks, Public Domain software collects performance data from their Novell servers. Once performance analysts manage and analyze the performance data, the results generated by SAS/CPE are then surfaced to their CIO in an executive information system on Microsoft Windows.

In order to accomplish this scenario, SAS/CPE on the Sun relies on SunNet Manager to collect data from the UNIX machines, the PC, and the networks connecting them. Novell server data is collected from a customer-written generic collector accessible through SAS/CPE on the Sun. Using the generic collector interface in SAS/CPE, any data type can be placed into a row/column format, including the date-time stamp which drives the management of the SAS/CPE PDB. After the Novell server data is processed with SAS/CPE, the reporting interface is available for generating reports and graphs. Using SAS/CONNECT's file transfer services, the reports and graphs stored in a graphics catalog are easily transportable to the PC for use in an executive information based system for the CIO.

The following steps illustrate how to use the generic collector, and additional functionality provided by SAS/CPE:

The first step is to surface performance management data into a SAS dataset or a data view. Next, the data dictionary is used to provide information on: (1) how to process the data after collection, (2) variables contained in the data source, and (3) how to summarize or reduce the variables over time. Upon completion of this step, the data dictionary allows the data described to be accessed identically to other fully-supported SAS/CPE collectors.

Now, using a report and graph template (Figure B) within SAS/CPE, analysts can generate and store images in a graphics catalog for use by their CIO in the executive information system:

```
replay _all_;
quit;
endrsubmit;
```

Using SAS/EIS on the PC in conjunction with functionality provided by SAS/CPE on the Sun, allows analysts to fully exploit the SAS System's ability to surface performance management tasks across systems.

Summary

As systems become complex and diversified, delivering information throughout an organization is a challenge for everyone in today's computer milieu. As systems shift to align mainframe technology with both desktop and UNIX servers, performance analysts struggle with managing a variety of different systems, applications and users. SAS Institute shares the challenges you face and offers performance management solutions for everyone. Use SAS/CPE software as an enterprise-wide paradigm for performance management and look forward to the challenge of being involved in the evolving computer industry.

SAS and SAS/CPE are registered trademarks of SAS Institute, Inc. in the USA and other countries. SunNet Manager is a registered trademark of Sun Microsystems, Inc. ® indicates USA registration.

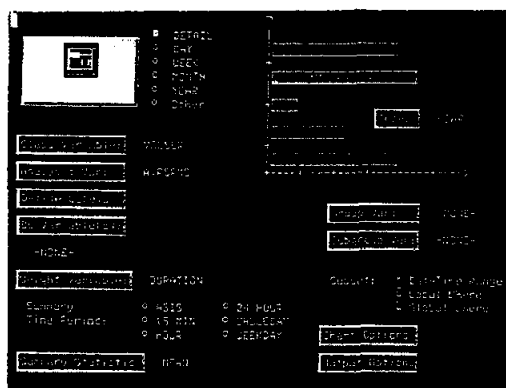


Figure B

Once the template produces an image that is stored in a graphics catalog on UNIX, transferring the catalog to a PC allows graphs to display easily with the following commands: (uppercase=static for SAS, lowercase=site-specific)

```
options remote=unixhost comamid=tcp;
filename rlink 'tcpunix.scr';
simon;
```

```
rsubmit;
/* SAS/CPE graphics code */
endrsubmit;
```

```

goptions device=<windows-device> display;
rsubmit;
goptions device=grlink display;
proc greplay igout=work.gseg nofs;

```