Simplify
Thanks, Jim. **Simplify with six words about z/OS.**

How many of us would prefer to retreat to a place of Zen simplicity, listen to a babbling brook, and hear the birds sing? In today’s world of constant news headlines, media sound bites, and Internet instant-messaging status, six simple words can carry great significance.

Micro-blogging on services such as Twitter has recently surged in popularity.

We are all too busy to read the wealth of information presented to us each day. A few simple and powerful words can make a big difference.

Throughout history, often pithy, succinct phrases are the most easily remembered. For example, recall Julius Caesar’s famous line:

“I came, I saw, I conquered.”

Literature also has memorable combinations of merely six words:

“Gather ye rosebuds while ye may.” – Robert Herrick

“Brevity is the soul of wit.” – William Shakespeare

“Her voice is full of money.” - F. Scott Fitzgerald

Technology often spurs the use of shorthand. Consider:

“Watson! Come here! I need you!” spoken by Alexander Graham Bell when testing the first telephone.


How about our own z/OS? One description is: “reliable, secure, available, scalable, networked, versatile.”

We decided to ask some of our own z/OS experts, including architects, designers, developers, and testers, to provide their own six-word summaries of what z/OS means to them. In this issue, we provide some of their quotations about our favorite mainframe operating system.

And that’s just a start. In this issue, we’ve gathered together articles that will help simplify your programming tasks with z/OS—everything from what IBM is doing to help out the z/OS novice to enhancements to CICS Explorer, database management with Optim, and an introduction to a new member of our z/OS family z/OSMF.

Here’s another six word phrase: “The best information about z/OS simplification.” Simply check out the table of contents. And be sure to browse the pages of this issue to read what our astute experts have to say about z/OS in their own six words.

IBM makes z/OS simple for you—that sums up in six words what we’re committed to doing as we help make your job easier and your work more productive!

The Editors

newsletr@us.ibm.com
Contents:  Featured articles

4  Simplifying System z
KELLY PUSHONG

6  An introduction to z/OSMF
ANUJA DEEDWANIYA, RENATA RAND MCFADDEN, AND SHAYNE GRANT

22  Simplified key management with TKLM for z/OS
STEVEN HART, ROAN DAWKINS, AND JASON KATONICA

27  Using Optim integrated data management tools
CURT COTNER

29  IBM CICS Explorer
PAUL JOHNSON

49  IBM z/OS Communications Server Configuration Assistant
JEFF CATES, JORGE PADILLA, AND MARK WRIGHT

9  Now I get z/OSMF!
JOHN CANALE AND ANUJA DEEDWANIYA

10  An introduction to the z/OSMF Incident Log
BOB ABRAMS, NINA GORADIA, AND ANUJA DEEDWANIYA

13  Setting up operlog and logrec for z/OSMF Incident Log
BOB ABRAMS AND TATIANA OLIVER

16  Common Event Adapter
BOB ABRAMS AND SUSAN Z. DEMKOWICZ

19  Using system logger for z/OSMF
GISELA CHENG AND JIM HERTZIG

25  Get ready to migrate to z/OS V1R11!
MARNAL WALLE

32  SMF log stream usability enhancements
ANTHONY SOFIA

33  Fail-safe your sysplex root for high availability in z/OS V1R11!
AHILAN RAJADEVA

34  What’s your problem?
KATHY PFEIFFER AND WILLIE FAVORO

37  Looking at z through Eclipse-colored glasses
LARRY ENGLAND AND STEVEN MA

40  Easy as HCM
FRIEDRICH BEICHTER

41  Use HCM instead of HCD
REGINE REICHERT AND MARTINA WEIDLER

43  Ask Mr. Catalog
STEVEN BRANCH

44  Well, LookAt that! You can search APARs too!!

45  zFavorites

46  BCPii: Control your HMC and support element directly from z/OS apps
STEPHEN WARREN

48  May we help?

51  HOW TO  slim down your SMF type 30 record data
STEVEN JONES AND BONNIE ORDOÑEZ

52  New improvements for allocation
BRIAN KEULING, GIRIJA VARANASI, AND STEVEN JONES

55  We’re not RAS-ting on our laurels
GISELA CHENG AND AL J. NOLL

58  z10 Brings You 1 MB real memory page size
ELPIDA TZOZTATOS, CHARLES MARI, AND PAULA SPENS

60  System z energy use and efficiency estimate
OGECHI O. ANUNOBI AND PAUL S. HAGEN

62  Using Basic HyperSwap to prevent system and application outage
WILLIAM J. ROONEY, GREGORY E. McBRIDE, AND TARIQ HANIF

64  Max out your LPAR throughput with HiperDispatch
E. OZAN (OZ) BARAN AND M. ALEX CARABALLO

67  More mainframe memories: Printers
JOHN EELS

69  Wake up your system with Java SDK6
NORM AARONSON

71  Better than 31-bit performance using 64-bit SDK
KISHOR PATIL, LEVON STEPANIAN, AND THERESA TAI

73  z/OS Capacity Provisioning update
MICHAEL GROETZNER AND RALF THELEN

76  Using ICSF and RACF to restrict access to cryptographic keys and resources
JESSICA BONNER AND BOB PETTI

79  X509 digital certificate technology
WAI CHOI

81  Our contributors

83  Corrections and clarifications

Back cover: Simply irresistible
There’s something about Alice (and Zach!)

**Simplifying System z**

**BY KELLY PUSHONG**

Yes, there is something about Alice… and we’re not talking about Alice from Lewis Carroll’s famous novel … although some people might argue that a new z/OS® system programmer feels as bewildered as Alice did in Wonderland!

I’d like to introduce you to Alice, a persona developed at IBM® to represent today’s junior z/OS system programmer with one year of experience.

The IBM user experience and development teams are using the Alice and Zach personas to design solutions that are aimed at helping make z/OS system programmers who are new to the mainframe more readily productive. This optimizes system programmer tasks so that a mixed skilled workforce can maximize productivity and quality. This work is part of a larger cross-company effort to make the IBM System z® mainframe easier to use for a greater number of Information Technology professionals.

Personae gratae!

So, what exactly is a persona? A persona is a fictitious character created to represent a group of specific users of a solution or product that share common demographics and traits. For each persona, a profile is created that includes professional information such as job roles, responsibilities, skills, behavior patterns, and even some personal details to make the persona a more realistic character.

Why use personas? Personas help user experience and development teams by providing a common understanding of the target audience. Personas provide a reference point that can be used iteratively to stay focused on the needs of a specific user throughout the entire design and development process. An important aspect of creating a persona is validation and input from customers. In developing the Alice and Zach personas, IBM collaborated with over one hundred z/OS customers from installations of all sizes.

And let’s not forget her mentor, Zach, a persona that represents a senior z/OS system programmer with 25 years of experience.

There’s something about Alice and Zach Why focus on Alice and Zach? Studies show that the IT industry as a whole is on the verge of a skills shortage. The mainframe shortage is expected to be more acute because of the time it takes new people to gain proficiency on the platform. IBM studies and research confirm that even for systems with low complexity, it takes two years for someone with no mainframe experience to become proficient at system programming tasks and a minimum of five years for that same person to attain expert knowledge. The increase in the number of universities that are teaching mainframe skills is a direct response for the need to better prepare students for mainframe jobs, but we at IBM also need to help simplify the day-to-day tasks performed by a z/OS system programmer.
Easing the pain

Today’s z/OS operating system is the result of 45 years of innovation and evolution, and there are plenty of opportunities for simplification. Using the Alice and Zach personas has helped IBM identify and prioritize the following areas that will provide the most benefit to our customers:

- **Problem management and analysis** has been identified by customers as a high priority focus area. The z/OS Management Facility (z/OSMF) Incident Log will help simplify the task of gathering the right documentation for the support center. The Incident Log provides a consolidated view of the problem data collected for an incident with the ability to send the package of collected data to IBM support or another vendor of your choice. Prevention of problems before they affect system availability has always been a strong characteristic of z/OS. Predictive Failure Analysis® is an example of a recent function that will help prevent errors before they occur.

- **z/OS installation, migration, and maintenance** are other areas of focus. Two examples of solutions that are aimed at addressing this area are IBM Health Checker for z/OS migration checks, which help simplify the process of determining whether a migration action is required or not and SMP/E V3R5 FIXCAT HOLDDATA, which simplifies the task of determining which PTFs you are missing from a given PSP bucket.

• The third system programmer task requiring simplification is configuration. IBM is addressing this area by providing solutions that incorporate best practices and error checking. z/OS Management Facility Configuration Assistant, WLM Service Definition Editor, and IBM Health Checker for z/OS health checks are three examples of solutions in this area. Going forward, IBM’s objective is to create an integrated set of configuration tools that eliminate as many steps and manual input as possible and simplify the navigation between tools.

For great articles that tackle mainframe simplification in this issue, check out Read more at the end of this article.

Simplify, simplify, simplify!

A key part of the strategy to simplify and modernize the z/OS system programmer user experience is the role- and task-oriented, browser-based user interface that z/OSMF provides. z/OSMF makes its debut in September 2009 with the Incident Log and Configuration Assistant capabilities.

IBM plans to extend z/OSMF with additional capabilities in the focus areas outlined above and to leverage this technology to achieve integration between tools to improve task productivity.

Into the future

This is just the beginning of our journey to simplify the day-to-day lives of Alice and Zach, our z/OS system programmer personas. Expect to see more solutions focused on z/OS problem management and analysis, installation, migration, maintenance, and configuration delivered in the z/OSMF in the future! As I noted at the start of this article, the focus on the z/OS system programmer is part of a larger cross-company effort to make the IBM System z mainframe easier to use.

In this issue, you’ll find information on simplification efforts aimed at application developers, CICS® architects, developers and systems programmers, database developers, and security administrators!

Read more


For more information on z/OSMF, see “An introduction to z/OSMF” on page 6, “Now I get z/OSMF!” on page 9, “What’s in your (incident) log?” on page 10, and “Night at the operlog” on page 13.

For security administration, see “Keys to the kingdom” on page 22.

For database developments, see “Simply manage the data” on page 27.

For CICS simplification, see “Simply CICS” on page 29.

For problem determination and management, see “What’s your problem?” on page 34.

For information about Communications Configuration Assistant and z/OSMF, see “A simple helping of the good stuff” on page 49.
Are you a z/OS system programmer, new to z/OS and trying to learn quickly how to use the green screen interface? Or are you a programmer with many years of experience but have many systems to manage and not as much time to do it? Or maybe with all the systems you need to take care of, all the time is consumed in day-to-day activities like configuration, migration, problem analysis and management, (you get the picture), leaving little time to work on developing and installing new function? How would you like to work with a web-based GUI interface instead to simplify things?

Help is on the way! IBM z/OS Management Facility (z/OSMF) V1R11, a new zero-priced product for z/OS customers, begins easing some of your pain points and helps improve productivity. z/OSMF is designed to simplify, optimize, and modernize your system programmer experience starting with problem data management and TCP/IP policy-based configuration.

**Introducing z/OSMF!**

z/OSMF V1R11 is introduced on z/OS V1R10 or above to help reduce the number of manual steps required to perform a task and, where possible, to provide end-to-end guidance, preferably automated, and always with the end user task in mind. The interface is intuitive with embedded active user assistance in the user interface (UI) (for example, message and panel help, wizards that guide users through tasks, and interactive troubleshooting aids).

z/OSMF is a Web 2.0-based solution that delivers function in a task-oriented manner and incorporates a browser interface that communicates with the z/OS system. Hence, you can access and manage your z/OS system from anywhere you have a browser available. Everything you need is installed on z/OS, so there is no separate and complicated client installation required, and you still have all the z/OS Qualities of Service (QoS).

The z/OSMF application run time is a special version of WebSphere® Application Server called the IBM WebSphere Application Server OEM Edition for z/OS v7.0, which is included with the product. z/OSMF exploits functions provided by z/OS system components such as Common Information Model (CIM) server, Common Event Adapter, Communication Server and System REXX™ (SYSREXX™), as shown in Figure 1.
Setup
You can set up z/OSMF and the runtime WebSphere Application Server OEM through easy to use shell scripts. To set up WebSphere Application Server OEM, the system programmer who performs setup invokes a configuration shell script, which prompts for and saves configuration settings, as well as generates required jobs to be run. Additional scripts create the instance of WebSphere Application Server OEM to complete the setup. Next, the system programmer invokes the z/OSMF configuration shell scripts that set up the z/OSMF data file system and prompt and save the input values to be used for the rest of the setup process. The scripts generate all the RACF® commands into a REXX exec to be verified and run by your security administrator.

Note: While the scripts generate RACF commands by default, if your installation uses a different security product, you can create a script with the necessary System Authorization Facility (SAF) commands for your system.

To complete the setup, the scripts prime the data repository with the identity of the z/OSMF administrator and, lastly, verify the system setup. Additionally, they generate messages and log statements to indicate the setup procedure and any failures or errors that they have encountered.

Accessing the interface
When the basic setup is complete and WebSphere Application Server OEM is started, you can access z/OSMF through a browser session with the following URL shown in Figure 1 (bottom left): https://hostname:port/zosmf

For hostname:port, be sure to substitute the host name and port number for your installation.

The z/OSMF GUI consists of a login section on the top left, a navigation tree below, and a work area on the right. The navigation tree provides the list of tasks that the user can perform. In the work area, tasks for user interaction are launched and displayed. The Welcome task is launched when you load the page or authenticate with z/OSMF. You can now log in with the previously created z/OSMF administrator identity.

Functions and tasks
As Figure 2 shows, after you log in to z/OSMF, the Welcome task changes to contain more information to help you get started and to learn more about z/OSMF. z/OSMF V1R11 has four basic categories in the navigation tree:

- Configuration
- Problem Determination
- Links
- z/OSMF Administration

Authorizing users for the interface
The initial z/OSMF setup creates and authorizes the z/OSMF administrator ID so that it has access to all the provided tasks. The administrator decides which additional users should have access to some or all of z/OSMF tasks. Using sample scripts, the administrator defines the user to both z/OS resources and the z/OSMF GUI.

The new user must be defined to z/OSMF through the Users task under the z/OSMF Administration category in the navigation tree. The definition includes the z/OS user ID for the given user and the Role to which the user is assigned.

z/OSMF uses the concept of Roles to group users to manage their access to tasks. There are two primary Roles in z/OSMF V1R11:

- z/OSMF Administrator
- z/OSMF User
Typically the z/OSMF User Role should be used to provide access to most if not all non-administrative tasks within z/OSMF. The z/OSMF Administrator Role should be reserved for those users who are customizing z/OSMF. Role definitions can be modified at any time by accessing the Roles task under the z/OSMF Administration category in the navigation tree. The Roles task allows the user to select which tasks the given role should have access to. As a convenience, the current list of users that are part of that role is shown to provide a quick reference. More specifically, each user’s navigation tree only contains those tasks for which the assigned Role gives the user authority to access.

In order to log in as a z/OSMF user, you need to enter the SAF ID and password required for your system. Actions taken from that browser session after you have successfully authenticated are performed under the user identity that was used to log in to z/OSMF. In order to simplify the user experience, the z/OSMF GUI is customized for each user based on the user role membership and the role definition. More specifically, each user’s navigation tree only contains those tasks for which the administrator has authorized the user to access.

Help is always available
Context-sensitive helps are accessible at all times to help users with using z/OSMF. Within each task and panel, a hyperlink on the upper right hand side can be used to launch a new window with help information. Additionally when messages are displayed, each message provides a hyperlink to the helps for the respective message. These context-sensitive helps eliminate the need for the user to hunt down the proper manual and section or interrupt the given task.

z/OSMF begins easing some of your pain points and helps improve productivity.

z/OSMF: Simply the best
This was just an introduction to z/OSMF V1R11. z/OSMF V1R11 starts you on a journey toward simplification and modernization of z/OS management tasks for system programmers. For additional topics on z/OSMF in this newsletter, be sure to read the following:

• “Now I get z/OSMF!” on page 9.
• “What’s in your (incident) log?” on page 10.
• “A night at the operlog” on page 13.

For information about z/OSMF, see IBM z/OS Management Facility User’s Guide, SA38-0652.

- Kershaw Mehta,
  System z Platform Strategy Manager
Now I get z/OSMF!

z/OS and z/OSMF, a natural partnership—at a glance

BY JOHN CANALE AND ANUJA DEEDWANIA

OK. We’ve introduced you to IBM z/OSMF V1R11 announced with z/OS V1R11, the modern, easy to use system programmer interface for managing your z/OS system, starting with problem data management and TCP/IP policy-based configuration. But did you know that z/OS V1R10 also supports z/OSMF to help get you started right away, so you don’t have to wait until you complete your migration to z/OS V1R11? Well you might wonder how do I get z/OSMF? You can order z/OSMF as a CBPDO, or if you are ready to install z/OS V1R11, you can place a ServerPac order that includes z/OSMF. Understand that at the time of this publication the Configuration Assistant function of z/OSMF is available only on z/OS V1R11 or later releases.

It’s really that simple!
z/OSMF is designed to be installed using the scripts provided. This shortens the overall cycle getting the product operational and is simpler and less error prone. Even if you have a CBPDO order or are setting up another instance of z/OSMF, end-end configuration and set up has been simplified.

Special attention is paid to serviceability of this stack with an installation verification program (IVP) designed to validate the z/OS server side set up and a client environment checker to ensure and help with the correct set up of your browser. While z/OSMF V1R11 supports the use of Microsoft® Internet Explorer 6 and 7 as well as Mozilla Firefox 2 and 3, we have seen the best performance with Firefox 3.

Multiple system programmers from your company can log in to the z/OSMF system at the same time. It doesn’t matter where they are physically located as long as they have a workstation with Windows® XP operating system and Web browser access to their work site. If there was a system problem that generated multiple SVC dumps and triage was necessary to review the incidents, and then divide and conquer, your programmers can all log in to the system and work together. For more information, see the article “What’s in your (incident) log? An introduction to the z/OSMF Incident Log” on page 10.

Figure 1 shows how today’s access to your enterprise will be simplified by z/OSMF for the future. Keep an eye on z/OSMF. Over time, we want z/OSMF to mature and play more closely with other IBM and non-IBM products. To set the ball rolling now, you can add links to other products that you use right into your z/OSMF navigation panel. You can also save Internet and intranet links that your system programmers commonly use all in one place, instead of everyone having to save their own browser favorites. Everything you need to access is in one place and all users of z/OSMF have the same centrally managed access.
What’s in your (incident) log?

An introduction to the z/OSMF Incident Log

BY BOB ABRAMS, NINA GORADIA, AND ANUJA DEEDWANIYA

Let’s say you’re investigating a problem and you don’t know which dumps were taken lately or which ones have to do with system abends? In which component and subsystem did the abends occur? What other system activity was happening at the same time?

As part of your investigation, you open a problem with the service organization for the suspected product, IBM, or independent software vendor (ISV), and they ask you to send in your diagnostic data. Now it’s time to resolve the problem. First, you need to locate all that documentation, which most likely exists in a number of places (dumps, logs); you must name the data in such a way that the service professional is able to work with it, then figure out the firewall information; and finally it’s time to terse the dump before transmitting it.

In z/OS V1R11, the IBM z/OS Management Facility (z/OSMF) is a new offering for z/OS customers intended to simplify some aspects of the system programmer’s job. For more about z/OSMF, check out “An introduction to z/OSMF” on page 6 in this issue. To help speed problem determination it includes a new Incident Log to help with problem data management by collecting diagnostic documentation related to an incident and making it available for analysis.

An SVC dump related problem is an incident. In z/OS V1R11, z/OSMF begins to address the issues related to gathering diagnostic data for problems detected by the system (through dumped system abends) and for diagnostic data captured by the operations or system programming staff. As a result, incidents go in two general classes of SVC dumps:

- Incidents that are related to system abends
- Incidents that are “user defined” or requested by the installation using the DUMP or SLIP command

At the time a SVC dump is written to a data set, the system gathers and preserves additional diagnostic data. This includes snapshots of the operations log (operlog), log record (logrec) detail, and a logrec summary. As a result, the system maintains the diagnostic data, typically necessary when diagnosing a problem, associated with each incident. The processing is described in the article “New Kid on the Block: the Common Event Adapter” on page 16.

Hello, Incident Log

z/OSMF for V1R11 offers a browser-based user interface to simplify viewing and management of the system detected incidents and their associated diagnostic data. When using this interface, you experience fewer errors while obtaining, aggregating, and sending the collection of diagnostic data to IBM or an ISV and are able to perform these tasks more easily. Let’s look at some of the key functions available in the Incident Log task in z/OSMF V1R11. Figure 1 provides an example of an Incident Log panel on z/OSMF.

Figure 1. Incident Log panel
Extracts the details
The summary list, as shown in Figure 1, gives you a consolidated view of all incidents occurring on all systems in the sysplex that communicate through the same sysplex dump directory. You can drill down to view the details of any given incident that you want to work with.

Correlates incidents
The Incident Log also provides the ability to correlate the incident with an external problem number, like a problem management record (PMR) or Electronic Technical Response (ETR), and any internal problem tracking ID that your installation uses for problem management purposes.

Makes DAE easier
The task of informing dump analysis and elimination (DAE) to take the next dump for the symptom string of a selected incident is much easier because you do not have to interact with the IPCS DAE panel to figure out the associations yourself.

Push-button terse and send
A key benefit of the user interface is the ability to send diagnostic data to a specified network target through the "push of a button" using a wizard. The Incident Log locates, compresses, and sends, using FTP, all the diagnostic data associated with the incident. You can send this data to the IBM service or to an ISV that supports other software on your z/OS system.

A log with many views
Figure 1 is a view of the z/OSMF Incident Log summary that lists all incidents in the sysplex that meets filter criteria. By default, the Incident Log displays all incidents occurring in the prior three-day period. Other views include filters for system name, affected component, or product name, problem management tracking ID, or service problem number. You can use the filters individually or define a complex filter with one or more attributes. The Incident Log functions include:
- Display list of incidents including filter, sort, and configure tables
- Set properties to associate problem number and tracking identifier
- Display properties and view the list of diagnostic data and logs
- Send diagnostic data using FTP to define FTP destinations and profiles (firewall)
- Manage FTP jobs status to view, cancel jobs, delete status
- Allow next dump for specific incident symptom string.
- Delete incident along with associated diagnostic data

Figure 2 is a view of the Diagnostic Details, which shows you additional properties for an incident. The Incident Log, as shown in Figure 1, has summary information about the incident. The Diagnostic Details has additional information, like problem symptoms detected by the system when the SVC dump was taken. Dump title, system/sysplex name, abend code and reason code identifying the detected error, load module, and CSECT detecting the problem, and symptom string built by the system to represent the problem. The Diagnostic Details Data tab shows how the diagnostic data items are associated with the incident.
Figure 3 is a view of the Send Wizard that walks you through the process of sending the diagnostic data using FTP to a destination server.

The Incident Log allows you to predefine (again through its web pages) your favorite FTP destinations. It also allows you to define FTP profiles (firewall or proxy information). These destinations and profiles are then made available to all users of your z/OSMF Incident Log.

The wizard guides you through the necessary steps to successfully prepare and send through FTP the diagnostic data to any destination. You start by selecting which destination you want to send to and the appropriate firewall/proxy profile to use, if any. You can also edit the JCL job card information for the FTP jobs. After you click the Finish button, the Incident Log submits one job per diagnostic data item.

The Incident Log also provides the ability for you to check the high-level status of the FTP Jobs that were submitted against an incident, to see if they are still in progress, completed successfully, or failed.

Help, please
z/OSMF has context-sensitive help information integrated with each panel, which you can select through a “Help” link. Clicking on the link at any point brings up context sensitive help to guide you through the display and actions available for that panel. Every message the Incident Log application displays also contains help information, which makes all of the information available immediately. Just click on the message identifier and the interface presents help. The help in z/OSMF is also searchable—it’s like having the user’s guide at your fingertips.

Scripts
The z/OSMF Incident Log is dependent on a number of components, including IBM WebSphere Application Server OEM Edition for z/OS Version 7.0 (also referred to as WAS OEM 7.0) and a number of z/OS components. To help make configuration a breeze, z/OSMF delivers shell scripts and REXX utilities to configure the z/OSMF libraries and security definitions across all components in the software stack. You can specify naming conventions or allow the tool to take the defaults. The scripts do the rest of the configuration. Scripts are also available to drive the z/OS system definitions necessary for managing the Incident Log’s use of components like the Common Information Model (CIM) server, Common Event Adapter (CEA), system logger, system REXX, and others.

IVP
After you configure the environment, you can run the installation verification program (IVP) available for the Incident Log. The IVP checks whether all dependent components are set up properly to get the most benefit from the Incident Log. The IVP is also handy for troubleshooting the set up if necessary.

Try it!
This version of the Incident Log simplifies the task of gathering, tracking and sending diagnostic data for dump-related incidents. The IBM z/OS Management Facility User’s Guide, SA38-0652, contains consolidated information related to installing, configuring, using, and troubleshooting the product end-to-end. It’s the one-stop guide for explaining the operation of the z/OSMF Incident Log and can help answer your questions about the complete solution.
Night at the operlog

Setting up operlog and logrec for z/OSMF Incident Log

BY BOB ABRAMS AND TATIANA OLIVER

A number of years ago, did you look at configuring the operation log (operlog) and log record (logrec) log streams for your sysplex and decide not to pursue it? If so, it’s time to reconsider.

Operlog and logrec are important z/OS diagnostic logs that provide a recording of system activity. The operlog and logrec log streams capture message and error log information from all systems in the sysplex, and writes that information to log streams managed by the system logger component of z/OS. The log streams are written to coupling facility structures and are ultimately backed up to system managed storage (SMS)-DASD data sets. The operlog and logrec log streams have been the strategic method for capturing sysplex-scope log data for many years. Today, in the z/OS Management Facility’s (z/OSMF) Incident Log, we leverage the log streams to automate the gathering of diagnostic data associated with an SVC dump.

In support of the z/OSMF Incident Log, z/OS captures “diagnostic log snapshots” when an SVC dump is taken if the installation is using operlog and logrec. These “diagnostic log snapshots” become part of the data associated with each “incident” in the z/OS incident Log. Furthermore, they become part of the “package” of diagnostic data that is sent to IBM Service professionals using the Incident Log’s “Send Diagnostic Data” action. To find out more about the z/OSMF Incident Log, check out “What’s in your (Incident) Log?” on page 10.

Connecting the dots
Setting up the operlog and logrec log streams is covered across a number of z/OS books and coordinated though a table. See References for more information.

This article describes each of the component areas and “connects the dots” to help clarify how to configure the system to write the message log to the operlog log stream and the error log to the logrec log stream.

You can configure each member of a sysplex to put its messages and commands in the operlog using a few configuration steps, and then activate using an operator command. The messages are represented in the log stream by data blocks containing the text and control information for each message (in control block format). The log stream containing the operlog data is hosted in a coupling facility (CF) structure to facilitate actively writing to a common log. The coupling facility resource manager (CFRM) maintains the physical shared structures where the log streams reside. The system logger provides the mechanism used to retrieve and review the associated data spanning long periods. After the data writes to log streams, the tools accessing operlog and logrec continue to work, as depicted in Figure 1.

Figure 1. Accessing operlog and logrec on multiple systems in the sysplex
The log stream LS_DATACLASS value must specify an SMS data class that assigns VSAM SHAREOPTIONS (3,3) to the log stream DASD data sets. This is the name of the SMS data class to use when allocating the DASD offload data sets for the log stream. The LS_SIZE specifies the size, in 4K blocks, of the log stream DASD data sets for the log stream.

Similarly, to eliminate the problem of having to manage multiple logrec data sets, you can choose to define one CF-structure logrec log stream to collect information about software and hardware errors occurring across all systems in the sysplex. Using the log stream eliminates worry about the logrec data set filling up, scheduling a daily offload, or concatenating multiple history data sets.

Figure 2 shows the logical relationship of the different players involved in maintaining sysplex-scope log streams for operlog and logrec. Policy information that resides in couple data sets defines the log streams. Only one active LOGR policy is allowed in the sysplex, and it is recorded only on the primary couple data set. The LOGR couple data set includes the rules for corresponding log stream definitions, information about the CF list structure, and the status information about the log streams. However for logrec, you need to define the LOGR subsystem, which allows EREP to read the log stream output as if it originated in a data set. Activate the LOGR subsystem through use of the IEFSSNxx parmlib member, specifying:

3. Define your security parameters, allowing READ access for everyone to these log streams, and WRITE access to those who need to perform maintenance (such as delete operlog records) and then you’re ready to activate the log streams. For additional details, see References.

Ready, set, activate!

Now that you’ve run the jobs to set up the CF structure and LOGR profiles, activate the logrec log stream by using the SETLOGRC LOGSTREAM command. You can display the current state of logrec using the D LOGREC command, which shows the current “medium” (log stream or data set). You can run EREP (IECEREP1) against the log stream to obtain reports, create history data sets, and clear records from the log stream. However, to allow EREP to access the log stream, you need to modify the JCL to reference the LOGR subsystem that you set up earlier. Here is the sample JCL:

Beyond that, most of the steps to define the sysplex scope operlog and logrec streams are essentially the same.

1. Define a log stream structure in the CFRM policy using the IXCMIAPU utility.
2. Prepare the log stream definitions, also using IXCMIAPU, resulting in the “LOGR policy”. The log stream names are as follows:
   - Operlog uses SYSPLEX.OPERLOG.
   - Logrec uses SYSPLEX.LOGREC.ALLRECS.

Figure 2. Relationship of the couple data sets and the CF-resident log streams

You can also use the SETSSI command to establish the LOGR subsystem dynamically.

You can also manage the operlog log stream using IEAMDBLG. IEAMDBLG formats operlog logrecs to syslog format, writes them to a target file, and deletes the old records from the log stream. This is very useful for creating archive versions, managing archives, handling data retention, and the like. Tail management includes formatting the records to syslog format, writing them to a target file, and deleting the old records from the log stream. To activate operlog, use the VARY OPERLOG,HARDCPY command (no “O” in HARDCPY).
SDSF accesses the operlog log stream, formats the logged messages, and provides filtering capabilities in the same way the single system SYSLOG spool data set does. Use the SDSF LOG O command to view operlog.

**Helpful commands**

Table 1 contains some of the most useful commands for providing information on the status of the hardcopy log and logrec recording, and troubleshooting information for log streams and structures.

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>D C,HC</td>
<td>Status of the hardcopy log</td>
</tr>
<tr>
<td>VARY OPERLOG,HARDCPY</td>
<td>Manually activate operlog if not activated automatically at IPL</td>
</tr>
<tr>
<td>VARY OPERLOG,HARDCPY,OFF</td>
<td>Deactivate operlog for the issuing system</td>
</tr>
<tr>
<td>D LOGGER,C,LSN=SYSPLEX.OPERLOG,DETAIL</td>
<td>Display operlog log stream connections</td>
</tr>
<tr>
<td>D LOGGER,C,LSN=SYSPLEX.LOGREC.ALLRECS,DETAIL</td>
<td>Display LOGREC log stream connections</td>
</tr>
<tr>
<td>D LOGGER,L</td>
<td>System logger log status</td>
</tr>
<tr>
<td>D LOGGER,ST</td>
<td>Display system logger status</td>
</tr>
<tr>
<td>D LOGGER,STR</td>
<td>Display information on system logger structures</td>
</tr>
<tr>
<td>SETLOGRC LOGSTREAM</td>
<td>Switch recording from the logrec DASD data set to the log stream</td>
</tr>
<tr>
<td>SETLOGRC DATASET</td>
<td>Switch recording of logrec records back to DASD</td>
</tr>
<tr>
<td>D LOGREC</td>
<td>Display logrec status</td>
</tr>
</tbody>
</table>

**Advantages of operlog and logrec**

The advantage of using operlog and logrec is that you have a sysplex-wide view of events and errors. This view reduces the time and effort you spend diagnosing and solving multi-system problems. You no longer have to merge separate reports to get a complete picture of any hardware or software problems in the sysplex. It is easier to see exactly what problems occurred on all the systems in the order they happened.

Furthermore, directing these logs to log streams allows the system to capture diagnostic log snapshots when an SVC dump occurs.

**References**

For more information, see the following z/OS publications:

- In *z/OS MVS Setting Up a Sysplex*, SG24-6898, see Chapter 9 for tables that describe the tasks related to setting up the operlog and system logger
- In IBM Redbooks® Systems Programmer’s Guide to z/OS System Logger, SA22-7625, see Chapter 6 for a description of operlog and logrec log streams; section 6.2 lists the steps for setting up logrec log stream; section 6.3 lists the steps for setting up operlog
- *z/OS MVS System Commands*, SA22-7627.

**Acknowledgment**

We’d like to thank John McDowell for his contributions to this article.

---

Table 1. Summary of log commands
Perhaps while migrating to a recent release of z/OS you discovered a new component that issues a message during the IPL announcing it is running in “minimum mode.” While scratching your head and wondering what this new “CEA” entity is, you discover that you must set up some security definitions to help CEA get out of “minimum mode.” Common Event Adapter (CEA) is a new z/OS component. This article looks at the role CEA plays in the system to give you an appreciation of its importance.

Meet CEA
The CEA component provides a set of internal z/OS services related to new simplification functions. Current exploiters are:

- The Hardware Management Console (HMC) and support element in turn, uses the BCP internal interface (BCPii) component in z/OS V1R10
- The z/OS Management Facility (z/OSMF) Incident Log that runs with z/OS V1R11

The new CEA address space initializes during the system IPL, sets up some data areas, and then goes dormant until its services are necessary. There’s no need to worry about CEA taking large amounts of CPU or storage resources away from the system when it’s not being exploited.

CEA provides two types of system functions:

- An internal event subscription and delivery mechanism to obtain asynchronous event information about the resources being managed on z/OS. The notification of these events is delivered to the requesting z/OS UNIX® process using AF-UNIX sockets that, in turn, uses a set of internal z/OS interfaces.
- A required one-time security setup to use z/OS UNIX sockets, you must perform a one-time security setup that adds CEA as an exploiter of the z/OS UNIX environment. If you skip this setting, CEA is in “minimum mode.” In minimum mode, CEA exploiters like BCPii do not function. After you configure the security setting, you no longer have to worry about it, and CEA considers itself in “full mode.” This removes that annoying “minimum mode” message during IPL. After you perform this one-time setup, you can easily move CEA to “full mode” in the same IPL by stopping CEA and restarting it. (This also allows the security product to recognize the new user ID).

You’ll see future z/OS simplification functions leverage the CEA event delivery in later z/OS releases.

Instrumentation for the z/OSMF Incident Log
The CEA component provides the base function for setting and obtaining system incident data used by the z/OSMF Incident Log. z/OSMF uses the z/OS Common Information Model (CIM) server to exchange data about system resources, like z/OS SVC dumps. CIM “providers” (special routines called on behalf of user interface requests for information) invoke CEA system routines to obtain data about dump-related incidents (for example, problem identification, SVC dump title, and symptom string). These functions are not dependent upon z/OS UNIX services, and are therefore available even when CEA is operating in “minimum mode.” Some of these services employ System REXX execs to invoke underlying utilities that dynamically allocate, compress system data sets and the like. So be sure that System REXX (the AXR address space) is active.

CEA does the “heavy lifting” for the z/OSMF Incident Log
The CEA component provides a number of z/OS system functions in support of the z/OSMF Incident Log, as shown in Figure 1.

When the system writes an SVC dump to a dump data set, CEA automatically captures diagnostic log snapshots associated with the dumped problem. CEA captures a number of diagnostic files around the time the dump occurs as follows:

- 30 minutes (default) of message data obtained from the operation log (operlog) log stream
- 1 hour of log record (logrec) detail obtained from the logrec log stream, and a 24-hour logrec summary report.

The operlog and logrec detail snapshots are captured in DASD log streams. This data is available to send to IBM Level 2 and other destinations you choose, such as those for independent software vendor (ISV) software support.

- The system maintains data about all z/OS incidents in the sysplex dump directory, which should be a shared VSAM data set. CEA saves all diagnostic information that is of concern to the incidents in this directory so that it...
can be accessed independently of the dump data sets. CEA uses Interactive Problem Control System (IPCS) services to access the sysplex dump directory and retrieve the inventory of dump related incidents. Both processes serialize on the sysplex dump directory data set. It is therefore critical that no user attempt to access the sysplex dump directory in an IPCS session as doing so will prevent the system from updating the dump information.

- When requested by the Incident Log user, the CIM server interrogates CEA about the incidents in the sysplex, and returns the incident information as well as information about the diagnostic log snapshots.

Setting the configuration

CEA is dependent on you to set up the following system resources:

- Set up system logger to create the operlog and logrec log streams. CEA creates its diagnostic operlog and logrec snapshots only when the WTO hardcopy log is being written to operlog and logrec is written to the logrec log stream. The operlog and logrec must be written to log streams for CEA to create its diagnostic operlog and logrec snapshots.
- Set up the sysplex dump directory as a shared VSAM data set by using BLSCDDIR CLIST. For the z/OSMF Incident Log to reflect all incidents in the sysplex, it is important to share the sysplex dump directory data set across all systems in the sysplex.
- Configure dynamic dump allocation to allow the system to allocate SVC dump data sets when necessary, using the data set name pattern that you specify on the DUMPDS command. Doing this gives the dumps unique names so CEA can retrieve them from the dump data set name recorded in the sysplex dump directory. If you do not use this system function, or you have a process that renames the dump data sets when they are created (such as when moved to a different volume), see IBM z/OS Management Facility User’s Guide, SA38-0652 for further instructions.
- Review the settings in the CEAPRMxx parmlib member for capturing diagnostic logs and creating the log snapshots. See Figure 2. Remember to modify the STORAGE, BRANCH and COUNTRYCODE parameters, and remove the “/” and “*/” around each.

Figure 1. z/OSMF Incident Log functions
/* DEFAULT CEAPRM00 PARMLIB MEMBER */
/* SNAPSHOT(N) */
HLQ(CEA)
/**************************************************************************
/* THE STORAGE PARAMETER MUST BE SPECIFIED TO INDICATE WHERE */
/* DIAGNOSTIC INFORMATION IS TO BE STORED. CHOOSE ONE OF THE */
/* FOLLOWING: */
/* STORAGE(STORCLAS(SMS_STORAGE_CLASS)) */
/* or */
/* STORAGE(VOLSER(VOL001,VOL002)) */
/**************************************************************************
/* COUNTRYCODE AND BRANCH OPTIONS WHICH ARE COMMENTED OUT BELOW */
/* SHOULD BE FILLED IN BY THE CUSTOMER. */
/* BRANCH(999) */
/* COUNTRYCODE(000) */
/**************************************************************************
DUMPCAPTURETIME
(
  SLIP(OPERLOG(00:30:00) LOGREC(01:00:00)
       LOGRECSUMMARY(24:00:00))
  DUMP(OPERLOG(00:30:00) LOGREC(01:00:00)
       LOGRECSUMMARY(24:00:00))
  ABEND(OPERLOG(00:30:00) LOGREC(01:00:00)
       LOGRECSUMMARY(24:00:00))
)
In z/OS V1R11, the IBM z/OS Management Facility (z/OSMF) recognizes when an incident occurs that results in taking an MVS™ dump. z/OSMF preserves the diagnostic data so you can solve the problem or, after opening a PMR, send the incident to IBM for resolution. The incident includes the MVS dump, snapshots of the operation logs (operlog), the log record (logrec), and the EREP event history report.

Let’s unravel the mystery behind the preservation of diagnostic data using the system logger facility and look at some tips for configuring and to set up system logger for z/OSMF. We’ll also point out the key “players” in case you want to optimize storage resources.

System logger background and basics
z/OSMF uses the z/OS Common Event Adapter (CEA) to gather diagnostic data for operlog and logrec, which are exploiters of z/OS system logger. Figure 1 displays a typical system logger setup:

Figure 1. System logger setup

For diagnostic snapshots, the most important system logger parameters are:

- **LSR**: The LSR parameter is a system logger value that defines the maximum number of log streams in the system logger policy, which is stored in the LOGR coupling data set (CDS). This number needs to be large enough to allow for two snapshot log streams in the z/OSMF Incident Log for each dump recorded plus two additional for the model log streams, which are used as templates to define the storage attributes for the snapshots. The LSR parameter is the only global system logger value you need to consider changing for diagnostic snapshots.

- **STG_SIZE** and **LS_SIZE**: The STG_SIZE and LS_SIZE parameters control the size of the data sets that are used for writing and storing the diagnostic snapshots. The snapshots use DASD only log streams that require the use of staging data sets and offload data sets. System logger first writes the snapshot of data to the staging data set, and then, when the staging data set reaches a threshold, it offloads it to offload data sets.

Fine-tuning
The important parameters for fine-tuning the LOGR policy are:

- **HIGHOFFLOAD = 80**
- **LOWOFFLOAD = 0**

By specifying the HIGHOFFLOAD and LOWOFFLOAD values, you can further tune the system logger offload processing. These values start or stop the offload processing before the staging data set is empty or nearly full.

CEA performs the snapshot processing during post dump processing. The design of this support requires that capturing of the snapshot be complete in a small, fixed amount of time. If the system logger is not set up properly, the snapshots can be incomplete or might not be created. Problems during snapshot processing result in CEA0600I messages from the system.

Setting up z/OSMF
It’s important that you customize the CEASNPLG job in SAMPLIB. The CEASNPLG job defines the snapshot log stream models that system logger uses for storage allocation. If you don’t define the model log streams, the default is to use the
allocation values of the source log streams, which in many cases causes undesirable results. For example, if STG_SIZE or a suitable data class is not specified for the source log stream, the STG_SIZE can default to two tracks for the staging data set, which fills up quickly and causes system logger to perform many offloads to make room for the snapshots. If LS_SIZE is very large, the offloaded snapshot log stream can be a waste of space.

Customizing z/OSMF
For customization, you can easily determine the values to specify for the CEASNPLG job. The STG_SIZE and LS_SIZE specify the allocation size for the staging data set and offload data sets. These parameters override the DATACLAS allocation. All other attributes of the DATACLAS and STORCLAS still apply and the sample job uses the LIKE keyword to pick up any missing values from the source log stream. If LS_DATAACLASS has a suitable allocation value, you can use it without specifying the LS_SIZE value.

A good way to determine appropriate values for STG_SIZE and LS_SIZE is to look at the current operlog or logrec EREP detail edit report and see how many records (lines) occur typically in the time window DUMPCAPTURETIME from CEAPRMxx specifies.

Specify the STG_SIZE and LS_SIZE values as a number of 4 K blocks. If the system has 100,000 operlog records in a 30 minute period, a typical snapshot requires 100,000 80 byte records (100,000 x 80 bytes) or approximately 2000 4K blocks. Use STG_SIZE(4000) and LS_SIZE(2000) to make sure the entire snapshot fits in the staging data set. Do the same when evaluating logrec or use the same values from the operlog snapshot.

The following example shows a job that creates the diagnostic log stream model for the logrec snapshot:

```
//SNMODEL JOB (034D000,TS),IBMUSER,CLASS=C,MSGCLASS=T,
//             NOTIFY=IBMUSER
//STEP1 EXEC PGM=IXCMIAPU
//SYSPRINT DD   SYSOUT=*  
//SYSIN   DD   *
DATA TYPE(LOGR)
DEFINE LOGSTREAM
   NAME(CEA.MODEL.LOGREC)
   MODEL(YES)
   DASDONLY(YES)
   STG_SIZE(4000)
   LS_DATAACLASS(VSAMLS)
   LIKE(SYSPLEX.LOGREC.ALLRECS)
/*
```

To ensure that the staging data sets are the proper size, you can monitor the SMF 88 records, which system logger provides.

Taking snapshots of operlog and logrec
When an incident occurs and after the system creates a dump, problem data management in CEA preserves the diagnostic data by taking a snapshot of the operlog and logrec as shown in Figure 2.

An example
The data that CEA captures depends on the parameters in the CEAPRMxx parmlib member. Let’s look at the following snippet of CEAPRMxx:

```
DUMPCAPTURETIME
(
   SLIP (OPERLOG(00:30:00)  LOGREC(00:59:00)
          LOGRECSUMMARY(24:00:00))
   DUMP (OPERLOG(00:30:00)  LOGREC(00:59:00)
          LOGRECSUMMARY(24:00:00))
   ABEND (OPERLOG(00:30:00)  LOGREC(00:59:00)
            LOGRECSUMMARY(24:00:00))
)
```

Figure 2. Snapshot of operlog and logrec
In this example, the snapshot captures 30 minutes of operlog for a SLIP dump, console dump, and ABEND dump. For example, if the incident (dump) occurs at 1:30 pm, the snapshot of the operlog starts at 1:00 pm and ends after the incident occurs. The snapshot captures about one hour of logrec for each type of the three dumps in the same manner.

CEA uses system logger services to create the snapshot of operlog and logrec. These services include the following tasks:

- Defining the log stream, using the IXGINVNT macro
- Connecting to log stream, using the IXGCONN macro
- Browsing and reading log data, using the IXGBRWSE macro
- Writing data to the log stream, using the IXGWRITE macro
- Disconnecting from the log stream, using the IXGCONN macro

As part of snapshot processing, system logger connects to the operlog and logrec as the SOURCE log streams. It browses and reads from the log streams for the log blocks defined in the time interval defined in CEAPRMxx. For example, for 30 minutes of operlog log stream, CEA calculates the start time and end time of log block (30 minutes before the incident dump to the time the incident occurred) as the range of log records to copy. System logger creates the snapshot log streams for operlog and logrec and writes the data to the snapshot log streams.

During the process of snapshot creation, if system logger encounters a problem, message CEA0600I is issued with the return code and reason code for the particular system logger service that had the problem. Here is an example of CEA0600I:

CEA0600I The z/OS Diagnostic Snapshot option failed.
SNAPSHOT TYPE: OPERLOG
SOURCE NAME: SYSPLEX.OPERLOG
TARGET NAME: SKIP.OY1.C3AB4A63.DEACF781

SYSTEM LOGGER SERVICES USED IN THE SNAPSHOT OPERATION:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>NAME</th>
<th>PARAMETER</th>
<th>RETURN CODE</th>
<th>REASON CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOURCE:</td>
<td>IXGBRWSE</td>
<td>READCURSOR</td>
<td>0</td>
<td>00000000</td>
</tr>
<tr>
<td>TARGET:</td>
<td>IXGWRITE</td>
<td></td>
<td>0</td>
<td>00000000</td>
</tr>
<tr>
<td>DIAG1</td>
<td>DIAG2</td>
<td>DIAG3</td>
<td>DIAG4</td>
<td></td>
</tr>
<tr>
<td>SOURCE:</td>
<td>00000000</td>
<td>00000000</td>
<td>00000000</td>
<td>00000000</td>
</tr>
<tr>
<td>TARGET:</td>
<td>00000000</td>
<td>00000000</td>
<td>00000000</td>
<td>00000000</td>
</tr>
</tbody>
</table>

CEADIAG1 = 00000004 CEADIAG2 = 00000F21

CHECK FOR CONDITION: LOOK UP REASON CODE 0865 for API IXGWRITE FOR THE MEANING AND SUGGESTED ACTION.

In this example, system logger was able to create the snapshot for operlog but ran out of time (the small fixed window) during the write to the target, thus this snapshot might be incomplete. When it first wrote to the target, service IXGWRITE received reason code 865 (the staging data set was full and had to wait for offloading to finish before it could write to the target). This condition indicates that the size of the staging data set is not adequate so you should adjust the STG_SIZE.

**z/OSMF in action**

After installing and properly setting up z/OSMF, all incidents are captured and the diagnostic data made available to z/OSMF Incident Log. Authorized users can run z/OSMF to manage the incident.

After they send the problem package to IBM, they can invoke z/OSMF to delete the dump and snapshots of log streams. The snapshot log streams are deleted from the CDS, staging data set, or offload data sets, so system logger can reuse these resources.

### References

For more information of z/OS system logger, see the following publications:

- IBM Redbooks *Systems Programmer’s Guide to: z/OS System Logger*, SG24-6898-01 available at the following Web site:
  [ibm.com/redbooks](http://ibm.com/redbooks)
- “New kid on the block: Common Event Adapter” on page 16.
- *z/OS MVS Setting Up a Sysplex*, SA22-7625-17
Do you need data encryption?
The dramatic increase in legal requirements and privacy regulations to protect sensitive data over the past 15 years says the answer is, most likely, yes. You have probably already realized this, and taken steps to make data encryption an integral part of your computer infrastructure whether you’re in the Healthcare industry trying to comply with Health Insurance Portability and Accountability Act (HIPAA), or in the retail industry trying to pass the Payment Card Industry Data Security Standard (PCI DSS) audit. But what about managing all those keys that go along with setting up your robust, new encryption solution? How do you manage them across the enterprise?

**Keys to the kingdom**

Trying to accomplish key management across an entire enterprise on various platforms and devices using an assortment of encryption solutions is a sure way to give even the most savvy security professional a migraine. Relief is on the way with IBM Tivoli® Key Lifecycle Manager for z/OS, a new product that provides centralized and simplified key management for storage devices like IBM 3592, Linear Tape-Open (LTO), and DS8000™.

**TKLM overview**

Tivoli Key Lifecycle Manager (TKLM) is IBM’s latest storage device encryption solution. It allows enterprises to create, manage, back up, and distribute their cryptographic key material from a single control point. TKLM stems from the existing Enterprise Key Manager (EKM) solution. Unlike EKM, which only provided a key server, TKLM provides key management, security policy capabilities, and a web-based user-interface for ease of use. It leverages the existing security strengths of the z/OS platform by using Integrated Cryptographic Services Facility (ICSF), System Authorization Facility (SAF), and Java™ based key stores to store all the key material.

The main function of TKLM is to distribute key material to devices when they request it without the need for human interaction. When a storage device needs a key to encrypt its data, it sends a request to TKLM for a new key. The device can explicitly request the specific key that it wants, or it can allow TKLM to choose the key. If the device does not specify a key in its request, TKLM assigns a key based on defaults that were set up either for that device type or for that specific device.

For instance, you can map a device to a key or a set of keys. If this mapping exists, then TKLM uses this information to decide which key to distribute. If no such mapping exists, then TKLM distributes the key that was specified as the default key for that device type. All this communication occurs over a secure channel so as not to compromise the key material.

When the device obtains the key, it is free to encrypt to its heart’s content, never worrying about how it will retrieve all this data that it is encrypting. This is because TKLM keeps the decrypting key tucked away in a secure place (remember the key store) so the device can then return to TKLM when it needs to decrypt the information and request the decrypting key. This is one of those cases in which the device requests a specific key. Figure 1 describes this process in more detail.

---

**Figure 1. An overview of TKLM and tape encryption**

Diagram showing the process of a device requesting an encryption/decrypting key from TKLM, if it doesn't already have it.
Currently, TKLM supports device types for 3592, LTO, and DS8000 Turbo devices. 3592 devices allow the use of a default certificate. With LTO devices, you can specify a default key group. A key group, in this sense, is an internal TKLM construct that gives TKLM the flexibility to choose from any key within that group. TKLM does not support the notion of a default key group or certificate for DS8000 devices. Each DS8000 device that requests a certificate must have a device-certificate mapping specified in TKLM.

TKLM also helps in the management of the key material by sending notifications when keys are close to their expiration date, and by allowing an installation to automatically have its in-use keys "roll over" to a new set of keys at a specified date.

System Services Runtime Environment for z/OS

TKLM runs in the IBM System Services Runtime Environment for z/OS (SSRE). SSRE provides a native Web services runtime environment for select system-level applications that run on z/OS and is provided at no charge to licensed z/OS customers. The pre-configured nature of SSRE allows for a simple installation process. See Figure 2.

As Figure 2 shows, SSRE uses a SAF back end (in our examples, RACF) as its security mechanism. The RACF commands necessary for SSRE to function are provided with the product, but additional changes will need to be added when TKLM is installed into the SSRE environment. During this installation, a Java Database Connectivity (JDBC) connection is created between SSRE and the DB2 instance that is going to host the TKLM data. Because of these external dependencies, the TKLM administrator must coordinate the installation with both the security administrator and the database administrator.

When TKLM is successfully installed, you can access it through a web browser by going to SSRE’s Integrated Solutions Console (ISC), which is the same tooling mechanism that is used to administer the SSRE environment. This administration console Web page contains a section for TKLM, and all configuration changes from this point are done either through the Web page or the wsadmin command line interface supplied with SSRE. You can use the WAS Information Center as an additional resource for any post-installation configurations for SSRE that are related to TKLM.

Encrypting Storage Devices

TKLM supports several encryption-capable devices. LTO Ultrium tape drives, the 3592 family of tape drives, and the DS8000 Turbo devices all include data encryption capabilities within the drives.

LTO Ultrium tape drives

The Linear Tape-Open (LTO) tape drives are ideal for the needs of a wide range of small systems. If an LTO tape drive makes a request, TKLM fetches an existing symmetric key from a key store and securely transfers this key (using asymmetric cryptography) to the LTO drive. The LTO drive then uses this symmetric key to encrypt or decrypt the data on the device.

3592 tape drives

The 3592 tape drive family, with the IBM System Storage™ TS1130 Tape Drive as the latest in the line, offers a robust and high-performance solution. The TS1130 tape drive is able to utilize existing IBM 3592 data cartridges at higher levels of capacity and performance. These tape drives have an advantage in that RSA key pairs are used to wrap the symmetric keys and store them on the tape. These encrypted tapes can be shared with business partners, who can “re-wrap” the symmetric key using a public key to access the material.

DS8000 Turbo devices

The IBM System Storage DS8000 Turbo series is designed to support the most demanding business applications with its exceptional performance and superior data throughput. The DS8000 Turbo can help simplify the storage environment by consolidating multiple storage systems onto a single DS8000 system. For complete information, see the following Web site:

ibm.com/systems/storage/disk/ds8000/

TKLM z/OS key stores

TKLM exploits the Java Cryptography Extension (JCE) to use ICSF, SAF, and file-based key stores as key repositories on z/OS. In TKLM, only one master key store can be selected at configuration time, so you need to do careful planning to avoid the wasteful rework of changing your master key store at a later time. The available master key store selections for
TKLM on z/OS are:
- JCEKS
- JCECCAKS
- JCECCARACFKS
- JCERACFKS

Table 1 summarizes these different key store types.

You can only use the SAF-based key stores, JCECCARACFKS and JCERACFKS with devices that use asymmetric key material, such as the 3592 tape drives and DS8000 drives. If you are planning to use TKLM to manage a device that requires symmetric keys, such as an LTO tape drive, do not select JCECCARACFKS or JCERACFKS as your master key store type.

Acknowledgment:
Special thanks to Ashwin Venkatraman for his contributions to this article.

Resources
- TKLM Information Center:

<table>
<thead>
<tr>
<th>Key store type</th>
<th>Software, hardware, SAF</th>
<th>Key generation</th>
<th>Where keys are stored</th>
</tr>
</thead>
<tbody>
<tr>
<td>JCEKS</td>
<td>Software</td>
<td>Performed by IBMJCE provider software algorithms.</td>
<td>Password-protected file-based key store.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JCECCAKS</td>
<td>Hardware</td>
<td>Leverages ICSF and Cryptographic Coprocessors running on z/OS. Key material is generated by ICSF/hardware.</td>
<td>If TKLM is configured with ICSF protection turned on, symmetric key material is stored in the ICSF Cryptographic Key Data Set (CKDS), and asymmetric key material is stored in the ICSF Public Key Data Set (PKDS), both being protected by the ICSF master key. If ICSF protection is not turned on, clear data is stored in the password protected file-based key store.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JCECCARACFKS</td>
<td>SAF-based keyring using ICSF to store asymmetric key material</td>
<td>Leverages ICSF and Cryptographic Coprocessors running on z/OS. Key material is generated by ICSF/hardware.</td>
<td>Uses ICSF to store asymmetric key material. If TKLM is configured with ICSF protection on, asymmetric key material will be stored in the ICSF PKDS, protected by the ICSF master key and the SAF keyring. If ICSF protection is not turned on, clear asymmetric key material will be generated by hardware and then stored in the SAF keyring.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JCERACFKS</td>
<td>SAF-based keyring not using ICSF to store asymmetric key material</td>
<td>Key material is generated by the SAF backend.</td>
<td>All asymmetric key material is stored in the SAF keyring.</td>
</tr>
</tbody>
</table>

Table 1. Key store types
Get ready to migrate to z/OS V1R11!

BY MARNA WALLE

The latest z/OS release is available at the end of September 2009—and if you want to get it up and running, you'll need to know some of the migration actions that can affect you. To get ready, take a look at these new changes that are coming in z/OS V1R11. Knowing about them can help you get to the latest and greatest z/OS faster. (Of course, all migration actions can be found in the important z/OS Migration book, so make sure to read that book when you prepare to move!)

**Action: Ensure that zFS sysplex_admin_level=2 is available on all systems**

z/OS Distributed File Service zSeries File System (zFS) introduces some very important coexistence considerations with z/OS V1R11. Before introducing z/OS V1R11 into the sysplex,

1. Ensure that you’ve installed the correct zFS coexistence PTFs
2. Moved zFS to sysplex_admin_level of 2

Failure to do both these steps will result in zFS not correctly initializing on z/OS V1R11.

**What you can do now:**

1. First, install OA25026 through rolling IPLs to your z/OS V1R9 or V1R10 systems. When you IPL z/OS V1R9 or V1R10 with OA25026, your zFS sysplex_admin_level will automatically be 1.
2. Next, through another set of rolling IPLs or by restarting zFS, set the zFS sysplex_admin_level to 2 in your IODFSPRM file (the default is level 1). When all the systems are at zFS sysplex_admin_level 2, you are now ready for z/OS V1R11 zFS. You can use IBM Health Checker for z/OS check IBMZFS,ZOSMIGV1R11_ZFS_INTERFACELEVEL to verify that other systems are at level 2.

**Action: Accept the CustomPac Installation Dialog default file system of zFS**

Because zFS is the recommended file system type, your z/OS V1R11 ServerPac will switch the default file system type from HFS to zFS. If you want all your z/OS V1R11 file systems to be zFS for the installation, there is nothing else for you to do. (This default change affects file system data sets that you’ve saved and merged forward into your ServerPac configuration, as well as new file system data sets not found in your merged configuration.)

**What you can do now:**

1. Decide if you want to accept this new default of zFS for all file system in your z/OS ServerPac or if you’d like to use HFS file systems.
2. If you want to use HFS, during your ServerPac installation, be sure to issue the CHANGE DSNTYPE ZFS HFS command in the Modify System Layout portion.

**Action: Migrate from LDAP Server to IBM TDS**

z/OS V1R10 was the last release to ship the Integrated Security Services LDAP Server. If you are still using this old LDAP Server, you must move to the IBM Tivoli Directory Server (TDS).

**What you can do now:**

1. Move to the IBM TDS right now—on your z/OS V1R9 or V1R10 system. IBM TDS has been provided in z/OS since V1R8. To determine if you are still using the old LDAP Server, try the IBM Health Checker for z/OS check LDAP_USE_TDS.

2. If necessary, update your CONSOLxx parmlib member, and remove your IEAVMXIT message flood automation processing

**IBM Health Checker for z/OS check CNZ_OBSOLETE_MSGFLD_AUTOMATION** can help you verify that you’re not using the obsolete version of message flood automation anymore.

**Action: Remove message flood automation exits**

As of OA25602 (and incorporated into z/OS V1R11), message flood automation has been enhanced to eliminate its use of message exit IEAVMXIT and command exit CNZZCMXT. If you are using message flood automation, you must remove whatever you’ve added to these exits.

**What you can do now:**

1. Update your MPFLSTxx parmlib member
2. If necessary, update your CONSOLxx parmlib member, and remove your IEAVMXIT message flood automation processing

**IBM Health Checker for z/OS check CNZ_OBSOLETE_MSGFLD_AUTOMATION** can help you verify that you’re not using the obsolete version of message flood automation anymore.

**Action: Determine if you will be using IBM TDS WLM support**

As of z/OS V1R11, IBM TDS supports using WLM to set performance goals within the LDAP server. The default is to use this IBM TDS WLM support, but using it requires some setup.

**What you can do now:**

1. Decide if you will use this support, or disable it. Naturally, using this WLM support is recommended, so plan to do some security customization and update your WLM policy. (If a default service class for LDAP isn’t configured in WLM, all LDAP server operations will run under the discretionary goal.)
2. If you want to disable the WLM support, specify the srvStartUpError option as “ignore” in the LDAP configuration file.

**Action: Determine if you will be using IBM TDS WLM support**

As of z/OS V1R11, IBM TDS supports using WLM to set performance goals within the LDAP server. The default is to use this IBM TDS WLM support, but using it requires some setup.

**What you can do now:**

1. Decide if you will use this support, or disable it. Naturally, using this WLM support is recommended, so plan to do some security customization and update your WLM policy. (If a default service class for LDAP isn’t configured in WLM, all LDAP server operations will run under the discretionary goal.)
2. If you want to disable the WLM support, specify the srvStartUpError option as “ignore” in the LDAP configuration file.

**Action: Remove message flood automation exits**

As of OA25602 (and incorporated into z/OS V1R11), message flood automation has been enhanced to eliminate its use of message exit IEAVMXIT and command exit CNZZCMXT. If you are using message flood automation, you must remove whatever you’ve added to these exits.

**What you can do now:**

1. Update your MPFLSTxx parmlib member
2. If necessary, update your CONSOLxx parmlib member, and remove your IEAVMXIT message flood automation processing

**IBM Health Checker for z/OS check CNZ_OBSOLETE_MSGFLD_AUTOMATION** can help you verify that you’re not using the obsolete version of message flood automation anymore.
**Action: Run the syslog daemon in the background when starting from the UNIX shell**

In z/OS V1R11, if you run the syslog daemon from the UNIX shell (or shell script), it will not automatically run in the background. If you start syslogd in this way, you must add a trailing ampersand (&). Failure to do this will result in not having control returned to your shell session until syslogd ends.

**What you can do now:**
No changes are required if you start syslogd from a procedure. However, if you start syslogd from /etc/rc or from the UNIX shell, remember you must make updates to include a trailing ampersand on the command line.

**Action: Use ALL, DATE, or VERSIONS with the (H)BDELETE command**

As of z/OS V1R11, if you want to delete all backup versions of a data set, you must specify ALL on the DFSMSshm (H)BDELETE command. ALL is a new keyword and is mutually exclusive with DATE and VERSIONS.

**What you can do now:**
1. Decide if you want to delete all the backup versions of a data set.
2. If so, specify ALL when you are on z/OS V1R11. ALL indicates that active and retained copies (except for retired versions if they exist) will be deleted. VERSIONS only deletes active backup copies. DATE deletes active or retained backup versions.

**Action: Update JES2 macros and exit routines**

A new JES2 checkpoint level (z11) is introduced in z/OS V1R11 JES2, but there is no requirement to move from your current level (z2) to z11 unless you want additional capabilities. However, the addition of this new checkpoint level does introduce changes you must accommodate when migrating to z/OS V1R11 JES2.

A variable extension is added to the job output element (JOE) data area for z/OS V1R11 enhanced transaction SYSOUT selection for the SYSOUT API and extended status subsystem interfaces (SSIs). This extension is in the BERT data area in the JES2 checkpoint data set. A service, $DOGJOE, is provided to manage this data area in a release independent manner. $DOGJOE must be used when the JOE data areas are being modified, and should be used to examine fields in the JOE data area.

**What you can do now:**
Understand the updates that you must make to your exits and the interfaces to certain macros. You can find these updates in z/OS Migration, GA22-7499-15, and z/OS JES2 V1R11 publications. Remember, these updates are required regardless of what JES2 checkpoint level you are using.

**Action: Define a new SAF profile that controls access to the SDSF SYSLOG**

z/OS V1R11 SDSF provides a logical SYSLOG that replaces the HASPINDEX-based SYSLOG. This logical SYSLOG requires the following new SAF profile in the JESSPOOL class:

```
+MASTER+.SYSLOG.SYSTEM.sysid
```

Users must now have READ access to this resource to access the SYSLOG. Even if you use ISFPARMS for SDSF security, this profile is still required.

**What you can do now:**
If you do not activate the JESSPOOL class, there is no migration action to take.
If you do activate the JESSPOOL class, do the following:
1. Before defining this new profile and to help temporarily with this migration action, consider setting the Security.Syslog.UseSAFRecvr custom property in ISFPARMx to TRUE. By using this property, users with authority to the LOG command can use the z/OS V1R11 SDSF SYSLOG panel without the need for the profile.
2. Create the SAF profile and permit users to it.

**Last but not least...**

Two important recent enhancements can help you migrate to z/OS V1R11:

1. **SMP/E V3.5 FIXCAT**: Use FIXCAT “IBM.Coexistence.z/OS.V1R11” to see which z/OS V1R11 coexistence service is missing from your z/OS V1R9 or V1R10 system. Taking a couple of minutes to run the REPORT MISSINGFIX command after you have performed a RECEIVE on the latest Enhanced HOLDDATA is simple to do and can easily save you from many problems later about missing coexistence and fallback PTFs for the system.

2. **IBM Health Checker for z/OS checks**: Don’t forget that as of z/OS V1R10 (and continuing in z/OS V1R11), checks have been added to help you detect and verify migration actions. Use the FIXCAT “IBM.Function.HealthChecker” to see which checks are available on your z/OS V1R9 or V1R10 system that you haven’t yet installed. You’ll need to ACTIVATE many of the checks when you’re ready to plan for your migration.
I’m a database guy, so I spend a lot of my time talking to customers who power their business-critical applications using DB2 for z/OS specifically because of its performance, reliability, and availability. But I also hear undercurrents of concern for the future—will we be able to hire people who can manage the environment? How can we get college hires skilled in writing applications without spending a lot of money on training them ourselves? How can we reduce the time and money we spend finding issues in complex application environments using Java and .NET?

These concerns are legitimate, and that’s one of the reasons IBM offers an integrated data management portfolio to help solve these problems. Previously, these integrated data management tools were marketed under the Data Studio and Optim™ family names. Now, both portfolios are marketed under one name: Optim.

Integrated data management

As you know, IBM has been in the data management tools business for a long time. What’s different now, with integrated data management, is that we’re creating an integrated, modular environment to manage enterprise application data, and optimize data-driven applications, through all phases of the data lifecycle, from requirements (design phase) to retirement (archiving and deletion). The environment is modular, which means you can pick and choose tools that focus on particular aspects of the data lifecycle that are most important to you. For this article, I’ll focus on the aspects of development productivity and management (problem isolation and capacity planning).

Development productivity

The integrated tools are built upon open source Eclipse or use Web interfaces, both of which college hires are familiar with. You’ll want to get your developers using Optim Development Studio, which gives them all the tools they need to be immediately productive in developing stored procedures, XML queries, and high-performance Java applications (Figure 1).

With Optim Development Studio, you can easily create Java data access code, native SQL procedures, and Java stored procedures, all of which are eligible for specialty processor cost savings. You can develop and debug Java and native SQL procedures using a sophisticated debugger. Integration with Rational® Developer for System z extends this environment with COBOL, and PL/I stored procedure development and...
debugging in a single, integrated environment. Optim Development Studio also lets you service-enable any of these stored procedures with a simple drag and drop operation.

The most amazing thing about these offerings—the developer is almost unaware that the application is going to run on z/OS, so people using these tools don't need special z/OS training.

Problem isolation and capacity planning in complex environments

The other issue with application environments such as .NET and Java is that the environment is just more complex. It’s also a lot harder to isolate problems because so many database operations run using dynamic SQL under generic identifiers. You can’t even tell which application issued the query. This makes it difficult to take advantage of Workload Manager to help you prioritize workloads appropriately.

And administrators who are used to the abundant metrics and diagnostics with such tools as RMF™ and Tivoli OMEGAMON® provide to them for CICS/COBOL applications will suddenly find themselves in the dark with Java and .NET applications. So forget about letting inexperienced administrators handle these environments. There just isn’t enough information available to speed up problem isolation.

By using Optim pureQuery Runtime,.NET and Java applications can now use static SQL execution to avoid costly statement prepares. Database administrators can group SQL statements into packages with unique identifiers to help pinpoint problems faster (Figure 2). These package identifiers can also have WLM service classes assigned to them and will be used in reporting, such as RMF reporting classes and DB2 accounting records, to help with capacity planning. Plus you get the performance advantages of pre-bound queries, which can reduce CPU overhead and lock in the access plan for consistent performance.

With the Optim integrated data management tools, skills are easily learned and transferable because of shared artifacts used among the tools. Your developers will use the same Optim tools for both z/OS and UNIX® or Windows database applications, so your developers will have portable skills across all the systems in your enterprise. In addition, the Optim tools can help improve your ability to meet service level agreements by improving problem isolation, performance, capacity planning, and workload analysis.

Sharing more information:

- Visit the Optim product web site:
  [ibm.com/software/data/optim](http://ibm.com/software/data/optim)

Also, check out the following pages on developerWorks®:

- For Optim zone:

- For Community Space:

See these pertinent articles:


- “More pureQuery Performance: Now with .NET applications accessing DB2 for z/OS.” March, 2009, by Simon Harris, Todd Munk, and Rajesh Sambandhan:
  [www.dbmag.intelligententerprise.com/story/showArticle.jhtml?articleID=216401198](http://www.dbmag.intelligententerprise.com/story/showArticle.jhtml?articleID=216401198)

Acknowledgement

I would like to thank Kathy Zeidenstein for her contributions to this article.
In today's environment, system programmers are continually asked to do more with less. Too often IT is seen as a "cost" rather than a "benefit" to the business with the system programmer often navigating treacherous waters.

Maintaining complex IT infrastructures to provide high availability that businesses demand is labor-intensive, requiring both product specific tooling and home grown (and maintained) code. One moment you're on ISPF, a browser the next, a PC based client the next (along with that piece of paper inscribed with various notes along the way!) It ends up looking more like a living history of user interface trends, rather than a cohesive way to do your job.

Although tools can help, each tool takes time to install and evaluate, and even after that you're faced with a huge learning curve before you can gain real value from it. System programmers simply do not have time to "try out" new tools that often, and unless the tools can give immediate value, they are almost impossible to justify to management in terms of the additional expense. Finally, training new employees is also time intensive as a hundred different ways exist to do essentially the same thing.

Introducing IBM CICS Explorer

For CICS-centric management, there is a light at the end of the tunnel.

Introduced in November 2008, the IBM CICS Explorer is an Eclipse-based rich client platform (RCP) that can interface with the system management functions of IBM CICS Transaction Server, along with plug-ins for IBM CICS Configuration Manager, IBM CICS Interdependency Manager, and IBM CICS Performance Analyzer.

Development of this interface follows Agile principles, which in customer terms means that you don’t have to wait two years for enhancements. Indeed, by the time this article goes to press, multiple deliveries will have been made with more to come. With the latest version of CICS Explorer, released with the General Availability of CICS Transaction Server 4.1, standard software support is now also provided.

**IBM CICS Explorer**

CICS Transaction Server 4.1 introduces the CICS Management Client Interface (CMCI), a programming interface that provides remote access to CICS system management function. This interface can connect either directly to CICS management function or to the system management facilities of CICSPlex® System Manager (SM), a component of CICS Transaction Server.

**End user interface is easy**

The latest version of CICS Explorer provides a consistent CICS interface for both CICSPlex SM users and non-CICSPlex SM users alike. Everyone can benefit.

Access to definitional resource repositories like CICSPlex SM data repository (DREP) and CICS system definition (CSD) files as well as operational resources like dynamic workload management and system events is provided, along with the latest CICS capabilities introduced in CICS Transaction Server 4.1.

Setup time is minimal for both the CICSPlex SM and non CICSPlex SM users. Configuration of server and CICS Explorer connectivity can be achieved in minutes for those currently using CICSPlex SM or for those requiring pure CICS connectivity. For users new to using CICSPlex SM, access takes only a few more steps.

**Installing CICS Explorer**

Installation of the latest CICS Explorer can also be achieved in a matter of minutes. You simply download the client and extract it into a subdirectory. You are then able to exploit existing configuration data.

Connection to the CICS CMCI and tools is achieved through a simple dialog within CICS Explorer. After establishing connectivity in the initial setup phase, CICS Explorer automatically signs on to the appropriate interface to access the associated information as requested by the user. Compare this to the past, when you were required to constantly sign on to the appropriate product.
Navigating CICS Explorer
Navigational context in CICS Explorer is achieved through selecting the appropriate CICSplice, Scope, or CICS system. These context changes are communicated to all associated windows, ensuring that you are always looking at consistent sets of information.

Views are launched through familiar administration and operational menu selection, or by reference from within a given view (for example, programs used by transactions). You can also take actions against resources such as disabling or discarding a runtime resource.

View content is customizable along with filtering capability; you can manipulate the layout of fields within views or the views themselves through drag-and-drop techniques. Full help facilities are integrated, providing hover help and integration into the CICS TS Information Center.

For resource definition, rich editors are available that guide the user on how to input data, reducing entry errors and simplifying resource creation and modification.

Figure 1 shows an example of CICS Explorer connected to CICSPlex SM. It shows two CICSPlexes, along with the CICS transactions in each of the CICSPlexes. You can see an example of a rich editor to the right.

IBM CICS tools integration
Of course, the manageability of CICS can be enhanced by using IBM CICS tools. You can also take advantage of IBM CICS Explorer by exploiting the software development kit (SDK). For example, Independent Software Vendors (ISVs) can use the toolkit to integrate their products into the CICS environment.

Here are some of those tools and plug-ins.

CICS tools
CICS Configuration Manager (CM) provides access to multiple CSDs and DREP contexts. Resource editing facilities, complete with site specific validation verification, are available, including audit trails for backtracking, finding out "who changed what, where, and when". This tool also provides comprehensive reporting capabilities for resource definition analysis. Change management enablement allows modification definitions to be automatically transformed across change boundaries. You can generate online and offline reports to enable users to operate either interactively online, or offline browsing reports over a cup of coffee.

CICS Interdependency Analyzer (IA) automatically collects and maintains resource relationships within DB2 in order to understand application affinity and perform application analysis. This tool also captures information on CICS application usage of DB2, IMS™, and WebSphere MQ.

CICS Performance Analyzer provides access to all your CICS SMF data and associated data in CICS Transaction Gateway, IBM WebSphere MQ, DB2 and the z/OS system logger. It enables you to analyze your systems from an historical perspective. This tool also provides extensive ready-to-use reports and extracts to reduce the setup time and to quickly make you productive in analyzing the performance of your CICS systems as well as comprehensive tailoring facilities with which to build your own reports.

Plug-ins
Each tool comes with an associated plug-in to CICS Explorer. Information associated with each tool is accessible within Eclipse, and the plug-ins are easy to install.

The CICS Configuration Manager plug-in extends CICS Explorer to provide access to multiple CSDs or CICSPlex SM contexts. This plug-in also provides access to the resource audit logs for CICS CM and extensive reporting.

The CICS Interdependency Analyzer plug-in integrates the currently existing CICS IA Eclipse client, which provides access to all the information collected by CICS IA on application affinities and resource usage.

The CICS Performance Analyzer plug-in provides access to performance summary data that is extracted from SMF to a CSV
file, from an historical database to a CSV file, or accessed directly from data loaded into DB2. The plug-in allows the data to be displayed in both a tabular and interactive graphical form.

Pervasive integration with plug-ins

\[ 1 + 1 = 3? \]

1 + 1 = 2  (Basic algebra)
1 + 1 = 1  (Boolean algebra)
but 1 + 1 = 3?

Integrating user interfaces on the glass and contextual navigation within the set of views is not uncommon in some user interface technologies available today. What CICS Explorer offers is even more pervasive integration.

Navigation to the information and actions WITHIN another tools perspective, from within the user’s current perspective of choice, allows the user to continue to more easily perform the task at hand without leaving that perspective.

For example, after adding the CICS IA plug-in, the views within the CICS SM perspective that display files now provide access to “used by program” and “used by transaction” relationships.

In this way 1 + 1 really can make 3! Or more! Figure 2 shows the additional actions enabled as a result of plug-in installations.

Extending the Explorer experience

CICS Explorer allows you to extend its capabilities:

- Help provides the ability to add customer annotations.
- For most substantial extension, CICS Explorer provides a complete SDK of visual widgets and Java classes that “instantiate” the CICS object model. The model classes are not dependant on the Eclipse framework, and you can use them outside the Eclipse environment.
- Navigation to the information and actions WITHIN another tools perspective, from within the user’s current perspective of choice, allows the user to continue to more easily perform the task at hand without leaving that perspective.

The IBM Rational Developer for System z (RDz) product exploits the SDK to enhance the experience of the CICS development community. RDz makes use of consistent representation of data through rich editors, as well as access to additional information from the CICS Explorer tools plug-ins that can provide more information to developers targeting CICS subsystems.

As the demands on system programmers increase, various tasks traditionally performed by them, such as deploying applications into unit test systems, in today’s IT world are being delegated out to development staff. It’s good to know that this delegation can occur in a controlled manner, without exposing the systems to undue risk.

In conclusion

CICS Explorer is the home of CICS-centric system management. Within its Eclipse based framework, it can integrate with other products, and provide ease of use for end users and vendors alike, which makes it a highly productive interface for the CICS system management needs of today and tomorrow.

For screenshots, downloadable demos, and the latest information on IBM CICS Explorer, the SDK and CICS tools plug-ins, see:

ibm.com/cics/explorer

For more information on IBM CICS Tools, see:

ibm.com/software/htp/cics/tools

For the latest information on IBM CICS Transaction Server, see:

ibm.com/cics

Figure 2. CICS Explorer and plug-in capabilities
Made to order a better log stream recorder

SMF log stream usability enhancements

BY ANTHONY SOFIA

Beginning in z/OS V1R9, the System Management Facility (SMF) is able to support using log streams to store SMF data. A new utility for dumping SMF data from log streams, IFASMFDL was born. IFASMFDL had big shoes to fill as its older sibling, IFASMFDP, had been successfully running for many years. The log stream recording environment was quite different from the data set recording environment and initially IFASMFDL had some growing pains. IFASMFDL needed some new functions to help allow for an easier migration to SMF log stream recording.

Playing copycat

The first thing IFASMFDL found necessary was the ability to delete SMF data from the log stream. With data set recording, as one data set fills up, IFASMFDP is used to DUMP that SMF data set somewhere (such as a GDG concatenation) and then to CLEAR the SMF data set to make it available to SMF again. You can also combine the two operations using the ALL option.

Log stream recording does not follow this model, as the log stream is a contiguous stream of data, so IFASMFDL adapted the CLEAR and ALL concepts into two new options on the LSNAME keyword:

- `LSNAME(IFASMF.EXAMPLE.LS1,OPTIONS(ARCHIVE))`
- `LSNAME(IFASMF.EXAMPLE.LS1,OPTIONS(DELETE))`

The ARCHIVE option dumps and deletes the SMF data from the log stream that matches specified filters. The DELETE option omits the dump step and simply deletes the specified range of SMF data from the log stream. With the ability to delete the data from the log stream, it is no longer necessary to rely on the RETPD parameter for the SMF log streams. This can help those with high volumes of SMF data to minimize the amount of storage needed for logger offload data sets.

Learning a new trick

When IFASMFDL was introduced, the only option for specifying a date range was the traditional DATE keyword. This was adequate for data set processing; however, when using log stream recording, hard-coding a DATE keyword is not always convenient. It is often the case with log stream recording where the previous day or week of data is what needs to be dumped for processing. This means that JCL needs to be updated every time a dump is to occur. To simplify this, IFASMFDL has added a new keyword, RELATIVEDATE, which allows for a calculation of dynamic date range at IFASMFDL execution time.

For example,

- `RELATIVEDATE(BYDAY,1,1)` only selects yesterday’s SMF records from the log stream. The BYDAY keyword can also have the value of BYWEEK or BYMONTH to use different units in the calculation. The two numbers specified control how many days, weeks or months to go back from the current date and how many days, weeks or months to gather, respectively.

In another example if the format is

- `RELATIVEDATE(BYMONTH,3,3)`

During the month of July, SMF gathers records from the second quarter of the current year. However, if the same example was for October, SMF gathers data from the third quarter.

There is no longer a need to manually update JCL for regularly scheduled dumping. This more closely recreates the situation with data set recording where a single job could be used every time a dump needed to occur.

Bringing everything together

With the combination of the ARCHIVE and DELETE options and the RELATIVEDATE keyword, simpler migration paths to using SMF log stream recording now exist. If the goal is to keep all SMF data in the log stream and then use the new RELATIVEDATE keyword to build automated jobs to extract data required for business processing. On the other hand, use ARCHIVE and RELATIVEDATE in combination to move SMF data out of the log stream to data sets at automated intervals.

Availability and more information

These new functions for IFASMFDL are available in z/OS V1R11 and as PTFs for z/OS V1R9 and z/OS V1R10 using APAR OA27037. For more information about new functions for IFASMFDL, see z/OS MVS System Management Facilities (SMF), SA22-7630.
In shared file system configurations, the sysplex root file system is the sysplex-wide root that provides access to all directories from all systems in a shared file system configuration (known as a sysplex). Only one sysplex root file system is allowed in a shared file system configuration. When the sysplex root file system becomes unavailable and unrecoverable because of I/O or DASD failure in the shared file system configuration, it can lead to a multisystem outage that might require a sysplex IPL to recover. This is a classic case of a single point of failure in the shared file system configuration.

**Alternate sysplex root file system**

Help is here! In z/OS V1R11, we have provided a method to define and set up an alternate sysplex root file system in the shared file system configuration as a “hot standby” for the active current sysplex root file system. In the event of sysplex root file system failure, the system automatically activates the alternate sysplex root file system as the new sysplex root file system. The replacement of the sysplex root file system is transparent to applications and users on any system in the shared file system configuration.

**Requirements**

The following information summarizes requirements for setting up an alternate sysplex root file system:

- A shared file system configuration is required; however, the shared file system configuration can be a single system.
- All systems in the shared file system environment must be at z/OS V1R11 or later releases.
- The alternate sysplex root must have the same mount points and symbolic links as the current sysplex root.
- The file system type for the alternate sysplex root and the current sysplex root must be either hierarchical file system (HFS) or zSeries® file system (zFS); however, you do not have to specify identical file system types for both the alternate and the current sysplex root.
- The alternate sysplex root physical file system (PFS) must be active on all systems in the shared file system configuration.
- If the SECLABEL class is active and the MLFSOBJ option is active, the multilevel security label for the alternate sysplex root must be identical to the assumed multilevel security label of the current sysplex root.
- The sysplex root or any directories in the sysplex root file system must not be exported by the distributed file system (DFS™) or server message block (SMB) server.
- The real path name for the mount points in the current sysplex root must not exceed 64 characters in length.

Getting to the root: setting up the alternate

Follow these steps to establish an alternate sysplex root in a shared file system environment as follows:

1. Allocate a new file system to be used as the alternate sysplex root file system. Consider the following rules:
   - The user identifier (UID), group identifier (GID) and the permission bits of the root directory in the alternate sysplex root file system must match the root directory in the current sysplex root file system.
   - If the SECLABEL class is active and the MLFSOBJ option is active, the multilevel security label for the alternate sysplex root must be identical to the assumed multilevel security label of the current sysplex root.
2. On the alternate sysplex root file system, set up the mount points and the symbolic links. The mount points and the symbolic links must be same as the ones on the current sysplex root file system. Use `pax` shell command or `copytree` utility to populate the alternate sysplex root using the current sysplex root as a source.
3. Specify ALTROOT in the BPXPRMxx parmlib member with the mount point in the current sysplex root file system root directory.

**Restriction:** The ALTROOT mount point must not exceed 64 characters in length.

The following is an example:

```
ALTROOT FILESYSTEM ('NEW.SYSPLEX.ROOT.ZFS')
MOUNTPOINT('/sysalt')
PARM ('FSFULL')
```
In z/OS V1R11, we have provided a method to define and set up an alternate sysplex root file system in the shared file system configuration as a “hot standby” for the active current sysplex root file system.

4. Make sure that all systems in the shared file system environment have direct access to the new file system and can locally mount it.

5. Process the ALTROOT statement by using the SET OMVS command or by initializing the OMVS with the updated BPXPRMxx parmlib member.

Considerations for creating the alternate sysplex root file system
When you create the alternate sysplex root, consider the following:
• If possible, place alternate sysplex root file systems on different controllers from the current sysplex root file systems.
• Use the SETOMVS SYNTAXCHECK operator command to validate the ALTROOT syntax.
• When changes are made to the current sysplex root after the alternate sysplex root has been successfully established, you must make the same changes to alternate sysplex root as well.
• The alternate sysplex root is mounted as read-only and designated as AUTOMOVE.
• When the alternate sysplex root becomes the current sysplex root, it is mounted in read-only mode and designated as AUTOMOVE regardless of the current sysplex root settings.

In addition, you can manually replace the corrupted sysplex root file system or a failing sysplex root file system (that is, one that has not completely failed yet) using the F OMVS,NEWROOT MVS operator command with the new FORCE option. It is best to mount the sysplex root file system in read-only mode.

The root of the matter
To avoid the single point of failure in your shared file system configuration, plan to set up an alternate sysplex root. For more information, see z/OS UNIX System Services Planning, GA22-7800-16, “Chapter 5 Managing the z/OS UNIX file system.”

What’s your problem?

Problems are easy. We all have problems. That’s life. Solutions are harder to come by. Say your daughter has just dropped out of a prestigious university to pursue a career as a puppeteer, and you want to convince her that jobs for puppeteers are hard to come by. Yet you don’t want to crush her dreams. Maybe, you think, she could be the next Jim Henson or..., you know, um, some other puppeteer who was really successful like Jim Henson. What do you do?

How should we know? This is a z/OS publication, not some radio call-in show with some know-it-all host. We don’t have the answers to life’s problems. We do, however, have the answers to your z/OS problems, and we’ve collected them all in a book—z/OS Problem Management, G325-2564-04. This book can help you solve almost any problem you encounter on z/OS. In addition to containing straightforward diagnosis material, the publication now contains information about using Predictive Failure Analysis (PFA). Using PFA, you may be able to prevent problems.

Bookmark this library page and the solution is always one click away:

ibm.com/systems/z/os/zos/bkserv/r10pdf/#zs
Oh, the places you’ll go
Mainframe communities go social

BY KATHY PFEIFFER AND WILLIE FAVERO

Are you looking for ways to connect with members of the mainframe community but not sure where to start? Mainframe user groups and social networking sites provide excellent opportunities for mainframe professionals, new hires, students, and faculty members to network, collaborate, and stay connected with each other.

Check out these popular mainframe user groups, communities, and social networking sites you can explore.

User groups and communities
User groups and communities are an excellent vehicle for exchanging experiences, ideas, and technical expertise. Such groups and communities can include both organized events and online support.

CBT Tape
Internet-based CBT tape is a collection of freeware for the IBM mainframe operating system environment. The software on this site is contributed by others and is openly shared. The entire collection of free software can be downloaded at: www.cbttape.org

Enterprise Computing Community (ECC)
In 2008, Marist College was awarded a two-year grant from the National Science Foundation to build an academic and industry community to revitalize undergraduate education in enterprise computing. The ECC is a growing, continuing worldwide community of academic and industry partners. (Students are welcome to join, too.) Members engage both in face-to-face events and participate through online collaboration to discuss enterprise computing content organization and dissemination, as well as participate in outreach and engagement opportunities (including K-12). Membership is free. You can sign up today at the following Web site: ecc.marist.edu

IBM Destination z
IBM Destination z is a fast-growing and vibrant community for IBM System z collaboration and innovation that helps you make the most of your mainframe. Destination z provides an online meeting place for customers, systems integrators, IBM Business Partners, software vendors, and academic institutions to connect with each other and with mainframe experts to gain quick access to industry information and solutions. Through IBM Destination z, you can learn about the latest technologies and best practices, get information to drive business decisions, obtain skills, read case studies that reveal financial and business benefits to running on the mainframe, and connect with schools from around the world to take advantage of their mainframe program. In addition, IBM Destination z provides a blog and a discussion forum (see links below) and sponsors an Enterprise Computing scholarship program for students who are interested in pursuing or furthering their education in enterprise computing. Get involved today at: ibm.com/systems/z/destinationz/

NaSPA (Network and Systems Professionals Association)
NaSPA is the networking resource for Network and Systems Professionals. NaSPA encourages the sharing of technical information and the development of ethical standards among industry professionals. See the following Web site: www.naspa.com

SHARE and GSE (GUIDE SHARE Europe)
SHARE, Inc. is an independent, volunteer-run association renowned for its semi-annual technical conferences, which deliver hundreds of sessions from industry experts, including hands-on instruction labs, and provide valuable professional networking opportunities. With over 2,000 member companies, representing over 20,000 IT professionals worldwide, SHARE is one of the largest enterprise computing user groups in the world. Members include top international corporations, universities and colleges, federal government organizations, and industry-leading consultants. (And you get a complimentary copy of this newsletter when you participate!) Locate the next SHARE conference event at the following Web site: www.share.org

Like SHARE, GSE is an independent, volunteer-run association of companies, organizations and individuals who are involved in Information and Communication Technology (ICT) solutions based on IBM architectures. Formed by the merger of GUIDE and SHARE Europe, GSE provides an effective...
forum supporting the needs of today’s IT professional. For more information on GSE and scheduled European and regional events, visit the following Web site:

www.gse.org

zNextGen is the gateway for mainframe newcomers and emerging System z professionals. This growing community of professionals and students embarking on careers in enterprise technologies seeks to expedite its members’ professional development in a crucial segment of the IT workforce. This user-driven community is sponsored by IBM and SHARE. There is no cost to become a member. As a member, you will be able to network with established IT professionals, attend periodic meetings with zNextGen members, network with members at a SHARE conference, and attend mainframe beginner sessions recommended by the zNextGen project during that conference. You can find more information about zNextGen and how to become a member at:

www.znextgen.org

Social networking sites
The way in which people communicate online is evolving and at the forefront of this change is social networking. Social networking sites provide people with the ability to connect with groups and individuals online, using Web 2.0 tools. People tend to view these sites as places where friends go to chat, share music and pictures, and hang out. The business and IT communities, however, also use such sites to promote the sharing of enterprise computing information and to discuss ideas across boundaries. Social networking provides some impressive benefits. Sites can be used quickly and efficiently to build up an extensive network of contacts and allow individuals to post useful information to a targeted audience (in our case, the mainframe community), provide tips and insight, or share links for other Web sites, articles, blogs, or forums.

Discussion forums
Be sure to check out these forums for even more good stuff:

- IBM-MAIN (IBM Mainframe Discussion Group) – A list server for information sharing among users of IBM Mainframe computers. The IBM-MAIN list server is hosted by the University of Alabama: planetmvs.com/ibm-main/
- IBM Destination z:
  ibm.com/developerworks/forums/forum.jspa?forumID=1128

Blogs
Here are some blogs of interest:

- IBM Destination z:
  ibm.com/developerworks/blogs/page/destinationz
- IBM DeveloperWorks – Enterprise class innovation: System z perspectives:
  ibm.com/developerworks/blogs/page/systemzblogger
- Mainframe blog
  A collection of opinions from engineers, developers, programmers, users, and influencers who share a passion for the past and future of mainframe computing:
  www.mainframe.typepad.com/

Linkedin
These Web sites offer professional networking:

- System z Advocates:
  www.linkedin.com/groups?gid=155723
- zNextGen:
  www.linkedin.com/groups?gid=719667

Facebook
We’ve got a presence on these Facebook sites as well:

- IBM Mainframe professionals:
  www.facebook.com/group.php?gid=15888095404
- IBM System z on Campus (Keyword: systemz mainframe):
  www.facebook.com/group.php?gid=26433265296
- zNextGen:
  www.facebook.com/group.php?gid=19631089217

Twitter
A Web-based and short message service (SMS) that keeps people connected through brief micro-blogging exchanges (with entries limited to 140 characters or less):

- IBM System z Twitter:
  twitter.com/IBM_System_z

No job too big for z
- Larry England,
  Rational Application Tooling Technical Solution Architect
Looking at z through Eclipse-colored glasses

BY LARRY ENGLAND AND STEVEN MA

Looking around at modern development needs for cross-platform applications, cross-language applications, application integration, and data sharing, you can easily see that the beloved green-screen applications of yesteryear are less suited to meet these needs than a modern integrated development environment (IDE) like Eclipse. Young folks entering the workforce tend to look at System z through Eclipse-colored glasses and recognize the benefits that a sophisticated IDE like Eclipse can provide. IBM recognizes these benefits too. This is why IBM offers IBM Rational Developer for System z—an IDE built on the Eclipse platform.

Look back
In the early years of the S/360™, application development was accomplished using punched cards creating batch jobs. In those days, programs were written to a punch card of 80 columns and fed to the mainframe for processing: a process that could take many hours. TSO (Time Sharing Option) was introduced to allow user-written programs to be manipulated through a 3270 terminal, affectionately known to most people as the “green screen.”

Personal computers came along in the early 1980s and contributed to a change in the programming experience. Things got a little easier for mainframe programming: 3270 emulators emerged as an alternative to physical terminals, and punch cards were phased out by advances in digital storage devices. The user interface to the mainframe had also evolved, expanding from the command-line TSO to the menu-driven Interactive System Productivity Facility (ISPF) panels, which made interaction with the mainframe easier. Application development on the mainframe, however, remained largely unchanged for decades, and a manual and labor-intensive process.

Look around
We are now in the age of Web 2.0. Over the last ten to fifteen years, the evolution of the Internet (enabled largely by the popularity of personal and laptop computers) has led to an explosive growth in hardware technology, software technology, and programming languages. Nowadays, business applications written in multiple programming languages and running on a multitude of platforms are the norm rather than the exception. Business applications have gotten more sophisticated, and the need for these applications to communicate and interoperate has never been greater.

Java has emerged as a very popular programming language in the last ten years. Most computer programming students “speak” Java. A large number of these students also know Eclipse, which is the most widely used IDE for Java application development. However, Eclipse isn’t just a Java development platform. It is “an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle,” according to the Eclipse Web site:

www.eclipse.org

In short, the Eclipse platform can be extended into an IDE for anything you want to develop, including System z applications.

Since its introduction, the Eclipse platform has inspired a variety of offerings for System z application development, such as Rational Developer for System z. Through these offerings, System z developers are reaping the benefits of a common IDE for cross-language and cross-platform development.

Look at the advantages of Eclipse
Sophisticated, cross-platform, and cross-language applications are the norm these days, and the need to integrate applications and share data to meet evolving business objectives is paramount. Satisfying this need using green-screen tools is not easy, because many were designed to optimize development for a limited set of languages and platforms.

On the other hand, modern IDEs like Eclipse are designed to improve programmer productivity by providing a consistent user interface, and a set of development tools that work in concert, regardless of programming languages and deployment platforms.

And with a new generation of tools comes a new generation of developers. To them, an IDE is the “de facto” development environment for writing and testing code. This is because they are trained and skilled in using these tools.
Convincing new developers to use green-screen tools is a hard sell and requires retraining. What’s more, you would lose out on their expertise with IDEs.

Having a common IDE for all of your development (regardless of what languages are used and what platforms are targeted) makes good business sense. While you don’t necessarily need your Java and PL/I developers to be best buddies, it would be nice if they used the same tool, shared the same on-the-job training, and could efficiently work with each other on a business application. Eclipse makes this possible! It effectively breaks down the language and platform barriers between development groups by giving them a common development environment.

While the green screen will continue to have a special place in our hearts and the history of System z, it is clear that the future of application development and management tools is Eclipse.

Look at IBM Rational Developer for System z
IBM Rational Developer for System z (shown in Figure 1) is an IDE built on the open-source Eclipse platform. In a nutshell, it supports cross-language and cross-platform development and makes it easy to modernize and transform your mainframe applications to service-oriented architecture.

One of the design principles of Rational Developer for System z is to make it seamless for desktop users to learn. In addition, we designed it to be consistent with Eclipse—from the look and feel of the user interface to the interaction behavior. The result is a tool that developers familiar with Eclipse can easily learn and master. Rational Developer for System z also honors host requirements such as host code pages, data set characteristics, and language-specific settings. This makes it both easy and safe to use for development on (and off) System z. Rational Developer for System z lets users easily view and work with their data sets on z/OS using the built-in Remote Systems Explorer. What used to take lots of typing (for example, copying a data set member from one system to another) can now be accomplished using a simple drag-and-drop operation.

Figure 1. IBM Rational Developer for System z
Editing of code is made easy with features such as syntax highlighting, real-time error checking for COBOL, and content assist. More advanced features like COBOL code refactoring, code-snippet reuse, and the ability to visually compare and merge code between two programs provide mainframe application developers with a set of modernized editing capabilities that have long been enjoyed by their counterparts on the distributed side.

Rational Developer for System z also extends the Eclipse debug framework to support debugging of mainframe applications. As shown in Figure 2, you can debug a COBOL application running on System z using the Eclipse Debug perspective.

This seamless user experience across platforms and languages exemplifies the power and advantages of Eclipse-based solutions and makes it simple, easy and attractive for new (and old) developers to create applications on System z—one of the crucial factors in making any platform sustainable.

Rational Developer for System z recognizes the history, the breadth, and the depth of the platform and that it cannot be everything to everyone. In 2008, it published a set of programming APIs (currently still a technology preview) to support integration with third-party Eclipse-based tools. Using the APIs, programmers can write their own plug-in to adapt Rational Developer for System z to their home-grown tools and processes, thus eliminating the need to change their current processes.

**Look forward**

As the great Greek philosopher Heraclitus said, “The only constant is change.” (Or as we like to say, “shift happens.”) System z is an integral part of the ever-changing information technology ecosystem. As business challenges evolve, one must adapt. If your company has not looked into adopting an open-source Eclipse-based solution, maybe now is the time to take a closer look.

**Acknowledgment**

Special thanks goes to Susan Mazzara for reviewing this article.

---

*Figure 2. Using IBM Rational Developer for System z to Debug COBOL applications*
Easy as HCM

Use our free demo

BY FRIEDRICH BEICHTER

You are a z/OS system programmer responsible for the I/O configuration of your System z hardware and z/OS systems. You are working with Hardware Configuration Definition (HCD) using the ISPF panels to define the processor configuration with the logical channel subsystems, logical partitions, control units and devices. Wouldn’t it be nice if you had a graphical display of your I/O equipment together with the paths that can be used to address it?

Have you heard of the Hardware Configuration Manager (HCM)? It’s a Windows-based application that provides a graphical user interface to HCD. HCM shows you the physical I/O configuration diagram, and provides:

• A broad set of locate, filter and zoom capabilities
• Extensive graphical and textual reporting
• Easy-to-use definition tasks for setting up the I/O Definition File (IODF)

Unfortunately, because HCM is an optional priced feature of z/OS, it may take some effort to convince your management to invest in it.

To help you do this, the HCD/HCM development team provides a demo version of HCM. See Figure 1. The demo version is based on the HCM functions of z/OS V1R10, and can be downloaded from the z/OS HCD and HCM Web site at: ibm.com/servers/eserver/zseries/zos/hcm/

You can connect the HCM demo version to your z/OS host system and load your actual IODF to visualize your specific hardware configuration. Evaluating how the tool can serve your installation’s needs is easy, because you can use it to navigate your own I/O configuration, and can generate reports of familiar data. You can also add physical objects like cabinets and cables to the shown paths.

See all the great functions available with HCM and give them a try. You are only restricted from using the IODF update functions and the functions that interact with other systems management products. For example, you are restricted from using Tivoli System Automation to manage ESCON® or FICON® directors, and the Resource Management Facility (RMF) to display current performance data for your I/O.

The z/OS HCD and HCM Web site contains a tutorial which provides step-by-step guidance on navigating the HCM diagram. Working through the tutorial, applying it to your specific IODF, will help you understand the diagramming and reporting capabilities of HCM. Evaluating how HCM can benefit you couldn’t be easier!
Nobody likes to spend time flip-flopping between two ways of accomplishing a task, and if you are a hardware configuration management (HCM) user, you are no exception. Working with configuration packages or IOCP input data sets, you do not want to have to switch between hardware configuration definition (HCD) and HCM to perform your work.

Now you don’t have to. Two useful functions available in HCM with z/OS V1R10 allow you to perform the following tasks through HCM dialogs:

- Working with configuration packages
- Creating and migrating IOCP input data sets

With z/OS V1R10, HCM fully supports the configuration package concept, offering dialogs to do the following:

- Create, delete, and edit configuration packages within a work IODF
- Add or delete objects to and from a particular configuration package
- Transmit completely defined configuration packages from a production IODF to desired target systems, either in attended or unattended mode.

With these new functions HCM participates in the benefit of having a single data repository, while distributing only subsets of configuration data to remote activating systems, always ensuring that processor tokens are kept in sync.

Figure 1 shows the configuration package dialog that you can start from an opened HCM configuration. It allows you to create, delete, modify, and edit configuration packages:

Figure 1. HCM Edit menu for configuration packages
To be able to transmit a fully defined configuration package to a specific target system, you follow a sequence of windows in this HCM dialog, which asks for transmission and target system related data.

Creating and migrating IOCP input data sets using HCM
Sometimes it is necessary to build an IOCP input data set from an IODF. Starting with z/OS V1R10 HCM you can do this without leaving HCM and starting HCD. Figure 2 shows the new HCM dialog that allows you to do this:

This dialog allows you to create a data set containing IOCP or I/O configuration statements from an IODF. The resulting data set can be stored on the host or downloaded to the workstation automatically. If the statements are downloaded to the workstation, you can easily edit the resulting file with any available workstation tool. Or, you can send it to another user through e-mail attachment or copied to CD or memory stick if it is needed for a standalone IOCP.

When you have finished your changes in the file, the statements can be re-migrated into the IODF using HCM as well. Use the HCM Import I/O Configuration Statements dialog to perform the re-migration.

Note that it is not possible to create the IOCP or I/O configuration data set or perform the re-migration in standalone mode because access to the IODF is needed.

Information
There you have it—one stop shopping for your HCM and HCD needs. For more information about these HCM and HCD functions, see:

- z/OS V1R10 Hardware Configuration Manager User’s Guide, SC33-7989-09
- z/OS V1R10 Hardware Configuration Definition User’s Guide, SC33-7988-07

With these new functions HCM participates in the benefit of having a single data repository, while distributing only subsets of configuration data to remote activating systems.
Often I get questions from colleagues and clients about ICF catalogs whose answers I think are of benefit to all ICF catalog users. These articles will list some of the more common questions and provide answers that, I hope, you will find useful. I have worked with ICF Catalog for over 25 years as a developer, designer and architect. If you have a question, please send it to: branchs@us.ibm.com

1. Should I use extended catalog sharing?

If your installation is sharing catalogs, you should definitely consider extended catalog sharing (ECS). The default sharing mechanism for catalogs is very expensive because it requires a RESERVE and a read to the VVDS on the volume on which the catalog resides. This RESERVE and read are necessary to get the catalog self-describing VVR and happens for every access to that catalog. The RESERVE locks out all other access to the volume, and the read is a real I/O—no buffering allowed. ECS stores the necessary sharing information in the coupling facility and eliminates the RESERVE and VVR reads therefore providing substantial catalog performance enhancements.

ECS is not difficult to implement. You must have a coupling facility and include the ECS structure in the coupling facility resource manager (CFRM) policy. See the topic on “Using enhanced catalog sharing mode” in z/OS DFSMS Managing Catalogs, SC26-7409 for more information.

2. How many aliases can a user catalog have?

The number of aliases that a user catalog can have is a factor of the maximum catalog record size, 32,400 bytes. The number of aliases in a user catalog is dependent on the length of the alias name, so there’s no way to define an exact number. If you have very short alias names, the number is around 3500. If you are using multi-level aliases, the names are longer leaving less room for the number of aliases.

All alias names are in the user catalog connector record. This is so that we can delete all of the alias records when we delete the user catalog connector record. The user catalog connector record contains:

- Name cell that is 52 bytes in length
- Owner cell that is 20 bytes in length
- Volume cell that is 28 byte in length
- Association cell (variable in length and contains the alias names)

Therefore, the number of aliases is:

\[
\frac{(32400 - 52 - 20 - 28 - 3)}{\text{alias name length} + 1}
\]

or

\[
\frac{32297}{\text{alias name length} + 1}
\]

If we assume an average length for alias names of eight characters then the number of aliases possible is 3588 (32297/9).

3. Why does IBM recommend discontinuing the use of IMBED, REPLICATE and KEYRANGE?

IMBED imbeds the VSAM sequence set in the data component control area (CA). The sequence set is the lowest level of the index. Each entry in a sequence set points to a data control interval (CI) within a data CA. The record keys within a data CI are in collating sequence, but the data CIs within the data CA are not necessarily in collating sequence. The name “sequence set” came about because its function is to determine the collating sequence of the data CIs within the data CA. By imbedding the sequence set in the first track of the data component CA, it requires only a switch to the correct head to read the data CI after the data CI is determined from the sequence set.

REPLICATE makes copies of the sequence set around the track. This gives VSAM the ability to read the next sequence set record without waiting for the track to rotate back to the first record. However, there was a downside...
to REPLICATE when updating a sequence set record, it had to be rewritten for all the copies.

KEYRANGE is an option that allows distribution of keys among volumes. If you knew an application had a set of keys that would contend for arm movement, you could put those keys on different devices and possibly reduce arm movement on a given device.

VSAM striping is an alternative to key range that spreads the VSAM CIs over different physical control units (PCU).

With cached Redundant Array of Independent Disks (RAID) devices, these options no longer make sense because they were designed for the old DASD. Volumes on RAID devices do not have the same mechanical aspects. There is no predicting the arm movement of a RAID when emulating a 3380 or 3990 device. Therefore, you cannot define any new IRK data sets. IBM is looking into ways to convert backed-up IRK data sets to non-IRK data sets to enable recovery of back-ups.

When this conversion is available, IRK functionality will be removed from VSAM.

4. Can a catalog be greater than 4 GB in size?

Catalogs are limited to 4 gigabytes (GB) in size. With the extended address volume (EAV), you can place more and more data sets on the volume. Thus, the number of data sets cataloged is expected to grow creating a need for larger and larger catalogs. IBM is aware that many customers would like to see this limit extended.

5. We are thinking of increasing the number of user catalogs per volume to two user catalogs. Is there a reason, from the IBM perspective, not to do this?

Other than putting all your eggs in one basket, there is no reason at all that you cannot have two (or more) user catalogs on the same volume. With the reliability of RAID devices, this is typically not a concern.

When sharing, catalog management issues a RESERVE on the volume for each user catalog access. Having many RESERVES on the same volume might cause some contention depending on the activity of the catalogs involved. However, when you use extended catalog sharing (ECS), the RESERVE isn’t issued, which means you could use ECS to get around the RESERVE contention.

More Mr. Catalog

Be sure to check the z/OS Hot Topics Web site in October for another installment of Ask Mr. Catalog:

ibm.com/systems/z/os/zos/bkserv/hot_topics.html

If you aren’t already on our subscriber list, send an e-mail with the word subscribe in the subject line to:

newsletr@us.ibm.com

Well, LookAt that!

You can search APARs too!!

Did you know that in addition to performing a search on messages in LookAt, you can also perform a search on an APAR to a message? For example, if you look up message IEE459I, you get back the message details for that message. Let’s say that when you look up the message, you realize some of the details might not be as up-to-date as they should be. If that is the case, you can do a search on “apar for IEE459I”. If an APAR happens to be associated with that message, the APAR information also appears. As a reminder, messages can be searched from either the LookAt Mobile Edition or from the LookAt homepage at the following Web site:

ibm.com/systems/z/os/zos/bkserv/lookat/
It’s the zFavorites for System z credit card CD! You’re gonna love this! It has all sorts of helpful Web links, like those for:

- Hardcopy
- Operating systems
- Software
- Language and tools
- ISV development and applications
- Product documentation
- Marketing information
- Education
- Support
- Links to FREE downloads
- IBM Redbooks sampler
- WebSphere Application Server
- XML

To use the CD insert it in any standard CD device, and it should start automatically. If it does not, click Start > Run, and then type x:\index.htm (where x is your CD drive letter) and press Enter.

Additional copies of zFavorites CD (GK3T-4331-15) are separately orderable.
The application doesn’t fall far from the tree

BCPii: Control your HMC and support element directly from z/OS apps

BY STEPHEN WARREN

Someone recently brought to my attention that the acronym BCPii (which stands for Base Control Program internal interface) looks very similar to Wii (Nintendo Corporation’s immensely popular gaming console). Well, I don’t know if system programmers will be whipped up into the same frenzy and craziness to install BCPii the second they get their hands on it, just as the world was captivated a couple of years ago when Wii was introduced, but you will definitely want to check out this new z/OS BCP component and see what it can do for you and your installation.

Have you ever wanted to control your central processor complexes (CPCs) and logical partitions (LPARs or CPC images) and perform other Hardware Management Console (HMC) and support element (SE) functions through z/OS APIs, but you didn’t want the overhead of a suite of products that need to be installed? Or were you concerned about the security implications of having your HMC network exposed to the outside world? Enter BCPii!

BCPii provides an easy-to-use high-level programmable interface that any z/OS authorized application running in any address space can use to perform HMC-like functions. This eliminates potential security concerns by communicating directly to the HMC/SE through the operating system and not requiring an HMC network connection to any intranet or internet network.

The APIs

BCPii provides a set of APIs that allow an application to:

- Obtain the System z topology of the current interconnected CPCs, LPARs, and capacity records
- Query various CPC, LPAR, and capacity record information
- Issue commands against both the CPC and image to perform hardware and software-related functions
- Listen for various hardware and software events which can take place on various CPC and images

Imagine that you have an application that needs to keep track of the list of CPCs available in your installation and how many CBU test activations are remaining on each CPC, the list of LPARs on those CPCs, which operating systems are running on all those images, as well as the OS levels and if z/OS, which sysplex they are a part of. Finally, the “Current Defined Capacity” for each LPAR is desired. All of this and much more are easily accomplished through the creation of a small application using BCPii services.

The BCPii APIs supports applications using either the C or Assembler programming language. These applications must be both APF- and SAF-authorized, as well as having the proper authority granted at the HMC/SE to allow an application to access these functions. The z/OS UNIX System Services (z/OS UNIX) environment also has additional BCPii support built-in to facilitate the listening of events from this environment.

The z/OS BCPii APIs available include:

- HWILIST (BCPii List)
- HWICONN (BCPii Connect)
- HWIDISC (BCPii Disconnect)
- HWIQUERY (BCPii Query)
- HWICMD (BCPii Command)

- HWIEVENT (BCPii Event (for non-USS callers))
- HwiBeginEventDelivery, HwiEndEventDelivery, HwiManageEvents, HwiGetEvent (for USS callers)

HWILIST

This API allows an application to retrieve HMC and BCPii configuration-related information. Using this service, it is possible to obtain the list of:

- All CPCs interconnected with the local CPC
- The LPARs contained on an individual CPC
- The capacity records contained on an individual CPC
- The events already registered on a particular BCPii connection.

The values returned from this service can be used on subsequent HWICONN (Connect) calls.

HWICONN

This establishes a logical connection between the application and a CPC, LPAR on a particular CPC, or a capacity record on a particular CPC. The connect token returned from this service is used a handle for all subsequent communication to this connected object.

HWIDISC

This API releases a logical connection no longer needed.
**HWIQUERY**
This API retrieves information about objects managed by the HMC/SE related to CPCs, LPARs, or capacity records. Examples of data which can be obtained from a query request include:

**CPC information:**
- General information (such as name, serial number, machine type, etc.)
- Status information (operating status, etc.)
- Capacity information (capacity backup and capacity-on-demand information)

**Image information:**
- General LPAR information (LPAR name, OS information)
- Capacity information (defined capacity, processor weights)

**Capacity records information:**
- General capacity record information (name, activation or expiration dates, activation days)
- Status information (records status)
- Capacity information (the actual capacity record.)

**HWICMD**
This API directs hardware and software commands to CPCs and images. Examples of CPC commands include activation and deactivation of a CPC, a CBU, or COD request. Examples of image commands include: sysreset, load, start or stop of CPs, addition or removal of temporary capacity, and operating system commands.

**HWIEVENT**
This API for non-z/OS UNIX callers registers and unregisters an application and its connection for notification of hardware and software events occurring on the connected CPC or image. BCPii registers the user with event notification facility (ENF) for an event so that an ENF exit is driven only when that event occurs on the desired CPC or image name. Examples of events for which users can register, include command completions, status changes, capacity changes, disabled waits, and BCPii status changes.

**HwiBeginEventDelivery, HwiManageEvents, HwiGetEvent and HwiEndEventDelivery**
This API for z/OS UNIX callers facilitate callers in getting events delivered directly to their application using the underlying common event adapter (CEA).

**Example**

How would a typical application look? Putting together the APIs into an application is easy. The code shows each CPC name and image returned for the CPC list specified in the initial call:

```plaintext
Call HWILIST (ListCPCs)

Call HWICONN (CPC name (input), CPCConnectToken(output))
Call HWIQUERY (CPCConnectToken(input), QueryParms(HWI_CBUTESTAR))
Call HWICONN (CPC name (input), CPCConnectToken(output))
Call HWIQUERY (CPCConnectToken(input), QueryParms(HWI_OSNAME, HWI_OSTYPE, HWI_OSLEVEL, HWI_SYSPLEX, HWI_DEFCAP))
Call HWIDISC (ImageConnectToken)
Call HWIDISC (CPCConnectToken)
```

**How does all of this work?**
If the BCPii is properly configured, a new address space called HWIBCPII starts automatically at IPL. This address space manages all application connections; builds and receives all internal communication requests to the SE; provides an infrastructure for storage required by callers and for the transport communicating with the SE; provides diagnostic capabilities to help with BCPii problem determination; and provides security authentication of requests.

The address space processes the API request, invokes an underlying System z API, if necessary, which then invokes the BCPii transport processing, which communicates with the support element. You can stop the address space if necessary by issuing a STOP HWIBCPII command (for maintenance updates, for example), and then restarting using the HWISTART procedure.

**What do I need to do to get this all working?**
BCPii needs to be configured properly in order for the BCPii APIs to be available to a z/OS application.

**FACILITY class profiles will need to be set up by the security administrator to allow/restrict access to particular resources on the HMC network.**
- **z/OS UNIX** BCPii application requires a couple of additional steps to grant a z/OS UNIX application the authority to listen to ENF68 events. This includes enabling the CEA address space as well as activating additional security profiles required by the CEA.

**Current exploiter**
Capacity provisioning manager (CPM), introduced in z/OS V1R9, allows an installation to activate or remove temporary capacity based on either a given schedule (called scheduled-based provisioning), or based on workload performance (workload-based provisioning). The communications CPM used in this release required the HMC network to be connected to the internet or intranet network.
In z/OS V1R10, with APAR OA24945 installed, CPM extends its capabilities to use BCPii to communicate with the HMC/SE to make these capacity updates possible, allowing the HMC network to be completely isolated from the Internet or intranet network. For more information, see "Just right! z/OS Capacity Provisioning update" on page 73.

Getting started
BCPii is part of z/OS V1R11 and can be enabled on z/OS V1R10 through APAR OA25426, PTF UA47493.

We recommend reading up on the particulars of BCPii before enabling BCPii in your installation. BCPii provides documentation for installation, APIs, messages, codes, commands, diagnostics, and other information in these z/OS publications listed below:

- z/OS MVS Programming: Callable Services for High-Level Languages, SA22-7613-05. This is the best place to start and contains all of the installation documentation, including what levels of hardware microcode are required on System z9® and System z10™.
- z/OS MVS System Commands, SA22-7627-21.
- z/OS Initialization and Tuning Reference, SA22-7592-19.
- z/OS MVS Diagnosis: Tools and Service Aids, GA22-7589-15.

Ensuring the security of your dynamic infrastructure

In a dynamic infrastructure, System z embraces cloud computing and Service-Oriented Architecture (SOA) bringing state-of-the-art virtualization, energy efficiency, centralized management, and security.

System z allows you to mitigate new risks with a complete set of security protections. Are you benefiting from these protections? Do you:

- Enforce data confidentiality with tape encryption, the Encryption Facility or ICSF?
- Improve security, efficiency and compliance with zSecure?
- Facilitate single-sign-on and security integration with Tivoli Directory Server?
- Manage certificates and keys centrally with PKI Services?
- Centralize enterprise-wide security services on System z?
- Enforce tight security for System z SOA middleware or Web Services?
- Use security features for compliance to standards such as PCI DSS?

If you have any questions about these best practices (or need assistance in design, implementation, operations, or training) contact IBM Systems Lab Services and Training at:

stglst@us.ibm.com

For more information on z/OS V1R11, see:

ibm.com/systems/z/os/zos/
ibm.com/systems/services/labservices
A simple helping of the good stuff

IBM z/OS Communications Server Configuration Assistant

BY JEFF CATES, JORGE PADILLA, AND MARK WRIGHT

If column 72, fixed block LRECL 80, and punch cards have been part of your life, you will recall that long before there were personal computers and laptops, there was the mainframe. In college, our first program was typed on punch cards. The professor made us use punch cards so that we could see how programmers “used” to write programs. And there are those stories about dropping boxes of punch cards and never being able to get them back in the original order. So much for the simple good old days!

Today, after a transition from 3270s, computers are mostly viewed using graphic user interfaces (GUIs). IBM z/OS Communications Server has been part of this evolution by providing the Configuration Assistant, born with z/OS V1R7 and continuing to grow up.

Are you considering using the following z/OS Communication Server technologies?
• Application Transparent Transport Layer Security (AT-TLS)
• Defense Manager
• IP security
• Intrusion Detection Services (IDS)
• Network Security Services (NSS)
• Qualities of Service (QoS)
• Policy based routing (PBR)

The Configuration Assistant will greatly improve your productivity to get these technologies up and running.

No more syntax
The Configuration Assistant lets you focus on your tasks rather than on syntax. For example, if you want to encrypt your FTP server data, you can configure that with a few button clicks. Alternatively there is the z/OS Communication Server IP Configuration Reference, SC31-8776, with over 1700 pages for your consumption.

Using this publication you create files and type in configuration parameters. Let the Configuration Assistant guide you. It watches for parameters that conflict and for common configuration mistakes. Typos and syntax errors are a thing of the past. It provides online help so that you don’t need to study the entire IP Configuration Reference.

Up and running
In z/OS V1R11, the Configuration Assistant helps with more than correct configuration. To use one of the aforementioned technologies, several software components must be coordinated to provide all the function. The Configuration Assistant helps with these setup activities as well. Let’s take AT-TLS as an example.

The primary activity for the administrator and the Configuration Assistant is to create AT-TLS policy. But before that policy can be used, Policy Agent must be properly configured and active on the host system. AT-TLS generates trace records to SYSLOG, which requires that the SYSLOG daemon be configured and active as well. Each TCP/IP stack that receives AT-TLS policy direction needs an adjustment to its configuration profile. And all of these components must be given Security Access Facility (SAF)—or, in our case, RACF—permission before they can be active on the host system.

To help simplify, the Configuration Assistant provides you with a tracking list that includes all of these setup tasks. Each task comes with detailed instructions, sample materials (such as Policy Agent configuration files or RACF jobs that reflect the current configuration), and status information to track task completion including dates.

Freedom from fiddling with flat files
As we noted above, the primary task of the administrator is to create the policy for these communication technologies. The goal of the Configuration Assistant is to allow the administrator to manage that policy data without having to worry about statement names, syntax, or references. We recommend that all configuration activities be performed with the Configuration Assistant. But it is also important that policy data be kept synchronized with the data at the Configuration Assistant.

Policy data import is a Configuration Assistant function that provides this data synchronization capability and can be used in the following cases:
• The administrator has existing policy files and did not use the Configuration Assistant to produce those policy files. Start the Configuration Assistant, and use policy data import to bring in one or more complete policies for management by the Configuration Assistant.
• The administrator uses the Configuration Assistant to manage policy files. In an emergency, some change was made to a policy file, and it is now out of synch with the configuration managed by Configuration Assistant. Use policy data import to read the modified policy file and synchronize it with the Configuration Assistant.

Choose your platform
The Configuration Assistant is available as a download from the z/OS Communication Server Web site: ibm.com/software/network/commserver/zos/support/

After downloading, you install it on your workstation, which is simple enough. It would be even simpler to use through your Web browser.

The Configuration Assistant is not the only application designed to modernize the mainframe. IBM has created a new application that runs on WebSphere Application Server and contains components to handle different aspects of z/OS management. The name of this new product is z/OS Management Facility (z/OSMF).

Starting in V1R11, the Configuration Assistant will be available as part of the new z/OSMF infrastructure. This means that the Configuration Assistant will run in the z/OS system and can be accessed by the system administrators through a Web browser. With the Configuration Assistant running natively on the z/OS system, you can simplify the configuration tasks even more. For example, now you can save all the generated materials directly to the system without the need of using FTP to transfer the files.

If you have been using the Configuration Assistant on your workstation, moving to z/OSMF is very easy. Transfer your existing work into the new platform and begin accessing your data from your Web browser. Check out “An introduction to z/OSMF” on page 6.

Keeping it current
The Configuration Assistant is designed to meet industry Request for Comments (RFC) standards. As Communication Server is updated to meet new standards, the Configuration Assistant provides help in configuring its new functions. Both IPSec and AT-TLS have new functions in z/OS V1R10 and z/OS V1R11. As you migrate to the latest Configuration Assistant, if a new function is a good idea for all users to implement, you can obtain it without changing anything. In other cases, if the new function is not for most users, you can continue using the Configuration Assistant without noticing the new stuff. It is there and available to you, but perhaps it’s on an “Advanced” tab.

Can we make it easier?
Our users provide us with excellent feedback, so please keep it coming. One user challenged us to see just how easy we could make it to configure AT-TLS. In z/OS V1R11, you can configure a TCP/IP stack to use AT-TLS for a specific application in just a few mouse clicks. Another user showed us how if the Configuration Assistant could provide address groups, they could dramatically reduce the number rules configured for their IPSec policy. This is now available in z/OS V1R10 and later releases.

The evolution continues…
Nobody has to worry about dropping boxes of punch cards anymore. But learning about mysterious controls statements and how to put them together is often not very easy either. We intend to keep pushing on that E-Z button and perhaps one day all this syntax will go the way of punch cards.

IBM mainframe:
Often imitated,
ever duplicated

- Kathy Pfeiffer,
IBM Academic Initiative, System z Program Manager
SMF data is as valuable as gold to most installations. SMF data plays a vital role in dealing with a host of system management needs, including monitoring and tuning system capacity and performance, collecting audit data, data recovery (such as for catalogs), and system usage billing.

The possible uses for collected SMF data are extensive, but the individual records are rarely evaluated. This is because, when it comes to SMF data, the sum of the parts plays a larger role than each of the individual records. In fact, much of the data to be found in individual SMF records may be nothing but a bunch of zeroes! In some cases, perhaps way too many zeroes, such as when SMF generates null SMF type 30 EXCP data for every volume allocated—even SMS candidate volumes that weren’t used by the job. These candidate volumes are defined to SMS Storage Groups to give programs the flexibility to expand to additional volumes after initial allocation. The EXCP data for an unused candidate volume is, of course, zeroes because no I/O was performed on the volume. The resulting SMF type 30 record for a job with lots of unused SMS candidate volumes contains lots of zeroes.

**Trimming the fat**

All those extra zeroes don’t harm anything, but they do take up space and CPU cycles. An installation that has implemented the DFSMS™ Dynamic Volume Count function can see a significant swelling of its type 30 records, depending on the size of their candidate volume lists. Reducing this extra data can save space in SMF data archives and reduce the time and system resources needed for dumping, reading and processing the SMF data. These are all very good things for the health of your system! Elimination of this null data also reduces the time needed for DDCONS processing, because fewer record areas means there is less data to scan and consolidate.

**The workout**

In z/OS V1R10, you can choose whether you want smaller, tighter type 30 records, or if you want the extra null EXCP sections to be generated for unused SMS candidate volumes. The SMFPRMxx parmlib member contains a new parameter, EMPTYEXCPSEC, with two options:

- **SUPPRESS**—coded as EMPTYEXCPSEC(SUPPRESS)—will tell SMF to suppress those null type 30 EXCP sections for devices that are candidate volumes.
- **NOSUPPRESS**—coded as EMPTYEXCPSEC(NOSUPPRESS)—will continue the existing behavior of recording data for unused volumes.

Why would you want the extra empty EXCP sections? There’s information in data, even data that’s all zeroes. One could be using the size of the EXCP sections to further tune their system. For example, a storage administrator could analyze the records to see which data sets could have their storage class altered to reduce the dynamic volume count. (See z/OS DFSMS Storage Administration Reference, SC26-7402, for volume count and dynamic volume count planning.)

**A slimmer, trimmer record-type 30!**

To illustrate the impact of the new EMPTYEXCPSEC(SUPPRESS) SMFPRMxx parmlib option, let’s assume the average job uses 20 data sets, 10 of which have 10 unused candidate volumes. This would mean 3,000 bytes less information in the type 30 records for 100 volumes that are not being used. For one job, that is probably insignificant, but multiply that by the thousands of jobs run by heavy users of SMS, and it starts to add up.

The SMF data you collect is important to you, as is the time you invest in analyzing the data. Collecting only the data that matters helps reduce storage costs and processing times, which helps keep your z/OS systems cost-effective. Installing z/OS V1R10 and using SMF’s EMPTYEXCPSEC is an action you can take now for a leaner, trimmer SMF.
Simplification comes in many forms. Having the system “do more” for you is the easiest form, but having new commands and facilities that are available to help you to avoid unnecessarily complicated procedures is another. Even improving elapsed time for what should be simple operations can help simplify, if it lets you move on to the next item on your ever-growing to-do list. That’s why the MVS allocation development team made these improvements in z/OS V1R10 and V1R11.

**Tools to use now**

In z/OS V1R10, we’ve added new functions to help simplify the management and installation of the system. By the time you read this article, tape load balancing, VARY UNAVAILABLE and a new migration action abatement item will be available for immediate use.

**Tape load balancing**

When automatic tape libraries were introduced, the devices were all similar, containing a limited number of devices. Balancing requests across the libraries simply involved randomizing across the set of libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes. Customers increased their tape library workload to take advantage of the tape libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes.

Customers increased their tape library workload to take advantage of the tape libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes. Customers increased their tape library workload to take advantage of the tape libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes.

When automatic tape libraries were introduced, the devices were all similar, containing a limited number of devices. Balancing requests across the libraries simply involved randomizing across the set of libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes. Customers increased their tape library workload to take advantage of the tape libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes. Customers increased their tape library workload to take advantage of the tape libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes. Customers increased their tape library workload to take advantage of the tape libraries available. When newer libraries were introduced, they contained more tape drives and supported more volumes.

Some customers started divvying up their workload by changing the storage classes assigned to the libraries, just to ensure that the workload was more balanced. With the Tape Load Balancing function, included in z/OS V1R11 and available in z/OS V1R10 with APAR OA26414, you can change how the system allocates scratch workload to the libraries. Using the ALLOCxx parmlib member parameter, SYSTEM TAPELIB_PREF(BYDEVICES), MVS balances the workload across the libraries according to the number of devices the library has; the more devices, the more scratch requests it receives.

In this way, you can better balance the workload across dissimilar libraries. The default parameter, SYSTEM TAPELIB_PREF(EQUAL), allows you to continue to use the current behavior if you are happy with the current behavior or just want to keep the status quo.

**VARY UNAVAILABLE**

Another feature in z/OS V1R10 is the VARY UNAVAILABLE MVS operator command. Prior to V1R10, system programmers who wanted offline tape devices to stay offline could not easily hide them from device allocation. When an eligible device is offline, the recovery allocation part of allocation can bring the device online to satisfy the request.

In V1R10, we’ve added the VARY UNAVAILABLE command to mark an offline tape device unavailable for use by allocation.

This can also assist with operations. Recovery allocation lists all the eligible offline devices in message IEF877E so that the operator can select the device to bring online. When the list contains unusable devices, it gets confusing. With the VARY UNAVAILABLE command, the system excludes all those offline devices that need to remain offline, making the list more relevant, simplifying the response.

**V1R10 migration**

Another addition to z/OS V1R10 comes with APAR OA27917. The z/OS V1R10 Migration Guide contains an action under the heading “Do not specify volume information on allocations intended to uncatalog a data set.” In z/OS V1R10, allocation tightened up the requirements for using DISP=(xxx,UNCATLG) to prevent the system from removing the wrong data set from the catalog. But this left the installation having to manually check their JCL to ensure they wouldn’t be caught unexpectedly.

APAR OA27917 introduces “a happy medium,” in the form of a new ALLOCxx parmlib keyword, SYSTEM VERIFY_UNCAT, with the options FAIL, TRACK and MSGTRACK. FAIL continues the new behavior of simply failing the job. TRACK allows the job to complete normally while recording relevant information.
in the console tracking facility for later use. MSGTRACK also allows the job to complete normally while issuing a message to the job log and recording to the console tracking facility. In this way, the installation can migrate to the new behavior on its own schedule, rather than being obligated to do it when installing the release.

Tools to plan for
z/OS V1R11 has even more simplification to look forward to:

- A new JCL keyword for tape library allocations
- A new wrinkle on IEFBR14
- New operator commands to display and update ALLOCxx dynamically

Please do not ignore my unit name!
Allocations to SMS-managed tape devices (that is, tape library requests) usually ignore the unit name specified on the JCL UNIT parameter because the devices are chosen from the list of eligible devices selected by SMS. However, the ability to use one or a few selected devices from that list is of great value to many customers, especially those who want to install and test new features on tape library devices or add GTF traces for debugging. As more and more devices are added to tape libraries to improve availability, there is increasing demand to be able to direct the allocation to a small subset of the library devices.

In V1R11, we added a JCL keyword SMSHONOR (with equivalent dynamic allocation text unit X’0076’) as the fourth subparameter on the UNIT keyword. SMSHONOR causes allocation to honor the unit name and only use those devices common to both the unit name specified and the tape library devices selected by SMS.

Like waiting for the wrong bus
z/OS V1R11 also contains a long-awaited function, in the form of “Delete without DFSMShsm recall”. Most commands and utilities that can delete data sets are smart enough to realize that the data set is migrated and can delete it without recalling it first. Batch allocation, with its predilection for “gathering” all resources before starting a job step, always recalls data sets, even if the program to be run is the IEFBR14 program and the data set is being deleted with DISP=(OLD,DELETE). Deleting data sets using IEFBR14 is very common, but watching the data set be recalled to DASD before deleting it is like waiting for the wrong bus.

With a new option specified in the ALLOCxx parmlib member, SYSTEM IEFBR14_DELMIGDS, set to NORECALL, MVS allocation skips the recall for data sets being deleted with IEFBR14, allowing them to be deleted without delay. The default setting, SYSTEM IEFBR14_DELMIGDS(LEGACY), will allow installations to choose to recall data sets before deleting them, again allowing them to hold off enabling new features until they are ready.

Hold off on that IPL
Over the years, MVS device allocation has been updated to allow many defaults and options to be set at IPL using the ALLOCxx parmlib member. These options include properties such as what behavior the system should take for recovery allocation (when no devices are available to satisfy an allocation request), and default space amounts (used when parameters such as number of tracks are not specified).

In the recovery allocation example, the options would include WAIT or WTOR, indicating that the system wait for devices to become unallocated, or issue a WTOR for the operator to respond with an offline device to be used. The only problem with the above function is that it is not dynamic: one must re-IPL to change the settings.

Allocation now supports new commands to set and display these options. The SETALLOC command allows for updates to allocation options without having to re-IPL. Any way to avoid an IPL is a benefit, as these are expensive. Since there are more buttons and tuning knobs than there were in years past, it becomes increasingly important to be able to update these settings without having to IPL.

The syntax is borrowed from the ALLOCxx parmlib member syntax, with the exception of using commas and equal signs instead of spaces and parentheses. (Note that SETALLOC does not support the 2DGT_EXPDT keyword.) It allows the setting of multiple options within the same keyword, like UNIT OR CATLG_ERROR. For example, one can set the default PRIMARY, SECONDARY and DIRECTORY blocks on the SPACE keyword within one command:

```
SETALLOC SPACE, PRIMARY=20, SECONDARY=10, DIRECTORY=5
IEF010I SETALLOC COMMAND SUCCESSFUL
PRIMARY SET TO 20.
SECONDARY SET TO 10.
DIRECTORY SET TO 5.
```

Improving elapsed time for tape operations can help you move on to the next item on your ever growing to-do list.
Any values set with the SETALLOC command will take effect on the next allocation request (or in the case of TIOT SIZE, the next job step started).

The DISPLAY ALLOC,OPTIONS command displays the options that are currently in effect. You can then determine if these options are suitable for your current allocation needs. You can then change these options using a SETALLOC command. Here is an example of the display output:

```
D ALLOC,OPTIONS

IEF0031 15.37.28 ALLOC OPTIONS 886
SPACE
  PRIMARY: 100
  SECONDARY: 30
  DIRECTORY: 5
  MEASURE: AVEBLK
  BLKLENGTH: 8192
  ROUND: ROUND
  PRIM_ORG: CONTIG
  RLSE: RLSE
UNIT
  NAME: SYSALLDA
  UNITAFF: 3390
  REDIRECTED_TAPE: TAPE
TIOT
  SIZE: 32 (MAX DDS: 1635)
SDSN_WAIT
  WAITALLOC: NO
VOLUME_ENQ
  POLICY: WTOR
VOLUME_MNT
  POLICY: WTOR
SPEC_WAIT
  POLICY: CANCEL
ALLC_OFFLN
  POLICY: WAITNOH
  MAXNWAIT: 5
  POLICYYNW: WTOR
CATLG_ERR
  FAILJOB: YES
  ERRORMSG: YES
2DGT_EXPDT
  POLICY: WARN
VERIFY_VOL
  POLICY: YES
SYSTEM
  IEFBR14_DELMIGDS: NORECALL
  TAPELIB_PREF: BYDEVICES
  REMIND_INTV: 60
```

With these easy to use tools in hand, we hope you’ll agree that some system programmer tasks, such as collecting diagnostic data, deleting migrated data sets and dealing with parmlib settings, are getting easier and simpler, reducing your overhead for running your systems.

-CICS, IMS, and DB2, oh my!

-Larry England,
Application Tooling Technical Solution Architect
We’re not RAS-ting on our laurels

BY GISELA CHENG AND AL J. NOLL

Do you want to intervene and unmount an automount-managed file system when it fails to unmount? Do you want notification when there’s recovery of an unowned file system? Do you want the D OMVS,WAITERS command to include the date and time that the activity started? Do you know who the last one to issue the automount shell command was? Do you want more information in the D OMVS,PFS command?

Well, you can have it all in z/OS V1R11! All this function is part of the new RAS enhancements for physical file systems (PFS) that improves usability and enhances problem determination for z/OS UNIX System Services:

Intervene the unmount
You can manually unmount an automount-managed file system after an unmount failure. Prior to z/OS V1R11, an issue with the physical file system meant there was no way to intervene when an unmount attempt failed and automount continuously unmounted a file system. In z/OS V1R11, when this behavior occurs, a new BPXF250I message issues to both console and system log (syslog) advising of the event and suggesting a course of action as follows:

```
SY1 BPXF250I AUTOMOUNT FACILITY
CANNOT UNMOUNT FILE SYSTEM POSIX.HFS01
RETURN CODE = 0000007A,
REASON CODE = 5B210102
```

In the shell, you’ll see POSIX.HFS01 is pending unmount when the df command is issued, as follows:

```
# df
Mounted on    Filesystem            Avail/Total    Files      Status
/rel_11       (*AMD/rel_11)         0/8            0          Available
/rel_11/test  (POSIX.HFS01)         Not Available  Pending unmount reset or force
/SY1/etc      (ZOS111.SY1.ETC.HFS)  1592/4320      4294967050 Available
/SY1          (ZOS111.SY1.HFS)      1056/1440      4294967275 Available
```

You can manually perform an unmount of the file system using the TSO command with the FORCE option or from the ISHELL command.

Unowned recovery notification
Before z/OS V1R11, there was no indication of recovery or active for an unowned file system. Beginning with z/OS V1R11, a new BPXF251I message issues in both console and syslog of the owning system to notify you of the recovery of unowned file system when it becomes active.

```
SY1 BPXF251I FILE SYSTEM ZOS111.SY1.HFS
HAS BEEN RECOVERED AND IS NOW ACTIVE
SY1 BPXF251I FILE SYSTEM ZOS111.SY1.ETC.HFS
HAS BEEN RECOVERED AND IS NOW ACTIVE
```
In z/OS V1R11, when an issue with the physical file system occurs, the BPXF250I message issues to both the console and syslog suggesting a course of action.

Name the date and time

Before z/OS V1R11, the age value was relative to the time the D OMVS, WAITERS command was issued, and it was difficult to correlate between systems in sysplex environment. Beginning with z/OS V1R11, the response to D OMVS, WAITERS command includes the date and time the activity began in order to assist the correlation between the systems:

```
BPXO063I 19.22.48 DISPLAY OMVS 475
OMVS     000E ACTIVE OMVS=(27)
MOUNT LATCH ACTIVITY:
USER   ASID     TCB       REASON                      AGE
------------------------------------------------------------------
HOLDER:
TC0      0023  008E6E88     PFS Recycling                  00.03.14
          TIME: 2009/04/17 19.19.33
          IS DOING: Running
WAITER(S):
OMVS     000E  008DC528     FileSys Mount                  00.00.02
          TIME: 2009/04/17 19.22.45
FILE SYSTEM LATCH ACTIVITY: NONE
OUTSTANDING CROSS SYSTEM MESSAGES:
SENT SYSPLEX MESSAGES:
USER   ASID   TCB  FCODE  MEMBER   REQID     MSG TYPE     AGE
------------------------------------------------------------------
MEGA     0028 008AF6F8  1024 SY1      01000096 Mount        00.00.04
          TIME: 2009/04/17 19.22.43
RECEIVED SYSPLEX MESSAGES:
FROM   FROM           FROM
ON TCB ASID   TCB  FCODE  MEMBER   REQID     MSG TYPE     AGE
------------------------------------------------------------------
008DC528 0028 008AF6F8  1024 SY1      01000096 Mount        00.00.04
          TIME: 2009/04/17 19.22.45
          IS DOING: Mount Latch Wait-Latch 2
```

Who, when, where, how?

Before z/OS V1R11, there wasn’t a report for the result of an automount run. Beginning in z/OS V1R11, when automount finishes running, message BPXF260I in syslog gives you the following results:

- Who (user ID initiate automount)
- When (date/time)
- Where (which system)
- How (what automount policy used)

You can track the automount changes by checking the syslog.

```
BPXF260I AUTOMOUNT POLICY WAS CHANGED AT 2009/04/01 16:41:43
BY USER ELVIS ON SYSTEM SY2 WITH POLICY
```

How’s your Physical File System?

Before z/OS V1R11, the response to the D OMVS,P command, message BPXO046I, did not include the operation status of the PFS. Beginning with z/OS V1R11, message BPXO068I replaces BPXO046I. It provides the following detailed information about the status of file system:

- Active and when it was started (indicated in A under the ST column)
- Inactive (shown in I)
- Stopped (shown in S)
- Unavailable (shown in U)
A new interface for PFS reports the status. AUTOMOUNT is the first exploiter to report the status of an automount run so you can find out if the automount policy was changed. The following example shows message BPXO068I including automount status information (issue the D OMVS,P command for similar results):

```
SY1  BPXO068I 14.20.49 DISPLAY OMVS 457
OMVS 000E ACTIVE OMVS=(1T,HR)

PFS CONFIGURATION INFORMATION

PFS TYPE ENTRY ASNAME DESC ST START/EXIT TIME
TCP EZBPFINI SOCKETS A 2009/04/13 13.58.56
UDS BPXTUNIT SOCKETS A 2009/04/13 13.46.11
AUTOMNT BPXAMTD LOCAL A 2009/04/13 13.46.10
-PS GFSCINIT REMOTE U 2009/04/13 13.55.54
TFSX BPXTFS COLONY1 LOCAL A 2009/04/13 13.45.43
TFS BPXTFS LOCAL A 2009/04/13 13.45.43
ZFS IOEFSCM ZFS LOCAL A 2009/04/13 13.43.20
HFS GFUAINIT LOCAL A 2009/04/13 13.43.19

PFS TYPE DESCRIPTION MAXSOCK OPNSOCK HIGHUSED
TCP AF_INET 10000 4 4
AF_INET6 32000 1 1
UDS AF_UNIX 10000 1 2

PFS TYPE PARAMETER INFORMATION
-PS AttrCaching(Y)
HFS CURRENT VALUES: FIXED(0) VIRTUAL(57)

PFS TYPE STATUS INFORMATION

AUTOMNT TIME=2009/04/13 14:19:29 SYSTEM=SY1 USER=MEGA
```

The next example shows message BPXO068I without automount status information in the console (because automount status hasn’t been run):

```
SY1  BPXO068I 14.20.49 DISPLAY OMVS 554
OMVS 000E ACTIVE OMVS=(1T,HR)

PFS CONFIGURATION INFORMATION

PFS TYPE ENTRY ASNAME DESC ST START/EXIT TIME
TCP EZBPFINI SOCKETS A 2009/04/13 13.58.56
UDS BPXTUNIT SOCKETS A 2009/04/13 13.46.11
AUTOMNT BPXAMTD LOCAL A 2009/04/13 13.46.10
-PS GFSCINIT REMOTE U 2009/04/13 13.55.54
TFSX BPXTFS COLONY1 LOCAL A 2009/04/13 13.45.43
TFS BPXTFS LOCAL A 2009/04/13 13.45.43
ZFS IOEFSCM ZFS LOCAL A 2009/04/13 13.43.20
HFS GFUAINIT LOCAL A 2009/04/13 13.43.19

PFS TYPE DESCRIPTION MAXSOCK OPNSOCK HIGHUSED
TCP AF_INET 10000 4 4
AF_INET6 32000 1 1
UDS AF_UNIX 10000 1 2

PFS TYPE PARAMETER INFORMATION
-PS AttrCaching(Y)
HFS CURRENT VALUES: FIXED(0) VIRTUAL(57)
```

What happens in a mixed sysplex environment?

In a sysplex environment with mixed releases of z/OS, you get different responses for similar scenarios. For example, say that you have a sysplex with SY1 and SY2 where SY1 is running with z/OS V1R11 and SY2 is running z/OS V1R10 or lower:

**AUTOMOUNT unmount failed:**
- **SY1:** message BPXF250I displays to both console and syslog, and the file system has to be manually unmounted.
- **SY2:** AUTOMOUNT continues to attempt to unmount the file system.

**Unowned file system recovery:**
- **SY1:** message BPXF251I issues to both console and syslog.
- **SY2:** no message is issued.

**The D OMVS,WAITERS command:**
- **SY1:** message BPXF261I includes additional date and time information.
- **SY2:** message BPXF261I does not include date and time information.

**AUTOMOUNT command and the D OMVS,PFS command:**
- **Scenario 1:** AUTOMOUNT command runs in SY1; issues command DOOMS,PFS on both systems:
  - **SY1:** message BPXF260I in syslog only and BPXO068I in both console and syslog
  - **SY2:** message BPXO046I in both console and syslog
- **Scenario 2:** AUTOMOUNT command runs in SY2; issues command DOOMS,PFS followed on both systems:
  - **SY1:** message BPXF261I in syslog only (as shown in the following example) and BPXO068I in both console and syslog
  - **SY2:** message BPXO046I in both console and syslog

To get these RAS enhancements, there are no prerequisites for installation. Just climb the RAS ladder by upgrading from z/OS V1R10 to z/OS V1R11. Find out more in z/OS MVS System Messages, Vol 3, SA22-7633-17.
As the amount of central storage on a single system image increases, managing that storage becomes more complex and requires additional system resources. The availability of 64-bit addressing in both physical and virtual memory has resulted in a dramatic increase in application memory sizes. Applications that were once limited to a virtual memory size of 2 GB are now able to span 16 EB. Database management systems can have larger than ever buffer pools with typical working set sizes of several gigabytes.

The performance of such memory-intensive applications is dependent on the performance of the underlying memory management infrastructure. Time spent converting virtual addresses into physical addresses slows down an application often in a way that is not evident to standard performance tools. In many cases, there is an opportunity to increase the performance of the underlying memory management infrastructure, resulting in higher application performance.

The infrastructure and the need for 1 MB pages

The translation look-aside buffer (TLB) contains virtual-to-real translations used for quick lookups when a virtual address is referred to. This hardware cache array contains a fixed number of slots, each representing one 4 KB virtual page. When a virtual address is referenced, but does not contain an entry in the TLB, the dynamic address translation (DAT) structures (region tables, segment tables, and page tables) used in the virtual-to-real conversion must be traversed (and possibly built) for this reference.

As application memory sizes increase, this "TLB miss" occurs more frequently because TLB sizes continue to remain relatively small due to low access time requirements and hardware space limitations. As a result, TLB coverage today represents a much smaller fraction of an application working set size. A larger number of TLB misses equates to an increase in time spent converting virtual addresses into physical addresses. Applications can suffer a significant performance penalty resulting from an increased number of TLB misses, as well as an increased cost of each TLB miss.

The time taken to translate virtual addresses for applications with large working sets can often be minimized (which can thereby improve performance) by increasing the TLB entry coverage. Large page support (1 MB page backed by 256 contiguous 4 KB real storage frames) provides an increased TLB coverage without proportionally enlarging the TLB size because each entry in the TLB represents a wider range of addresses. This provides applications with better TLB coverage and therefore better performance by decreasing the number of TLB misses that an application incurs. Figure 1 shows the comparison between handling 4 KB DAT structures and large page DAT structures.

![Figure 1. z/OS 4 KB and large page DAT structures](image-url)
When running on a z10™ processor, large page performance improvements result from an increased hit rate in the TLB cache. Having one TLB entry for each MB page also means that the TLB2 cache will have more room to house other TLB entries. The TLB1 cache would still have one TLB entry per 4 KB. In addition, this support provides a foundation for the major consumers of central storage to avoid the substantial overhead of managing pages at the 4 KB level by using 1 MB pages instead. Applications and middleware would need to change in order to take advantage and exploit 1 MB pages.

**Are you configured for large pages?**

Table 1 shows a summary of how System z10 hardware and z/OS releases handle large page support requests with or without eDAT architecture support:

<table>
<thead>
<tr>
<th>z10 Hardware with eDAT support</th>
<th>z/OS version and release</th>
<th>Large page support requested</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>More than 4 GB of real memory • AND z/OS V1R9 with APARs OA20902, OA25485 • OR z/OS V1R10 with APAR OA25485</td>
<td>Request will be accepted</td>
</tr>
<tr>
<td>Yes</td>
<td>Releases earlier than z/OS V1R9 OR z/OS V1R9 without APAR OA20902</td>
<td>Request will be rejected</td>
</tr>
<tr>
<td>No</td>
<td>More than 4 GB of Real Memory • AND z/OS V1R9 with APARs OA20902, OA25485 • OR z/OS V1R10 with APAR OA25485</td>
<td>• Request will be rejected when pageframesize=1MEG • Request will be backed with 4 KB pages when pageframesize=MAX</td>
</tr>
<tr>
<td>No</td>
<td>Releases earlier than z/OS V1R9 OR z/OS V1R9 without APAR OA20902</td>
<td>Request will be rejected</td>
</tr>
</tbody>
</table>

**Using large pages**

The new CVTEDAT bit indicates that the enhanced DAT architecture for large page performance improvements result from taking advantage and exploiting large pages. Key factors to consider when granting access to large pages are the amount of large frame area (LFAREA) currently in use is monitored by SRM. System issues appropriate messages to alert the operator when usage exceeds certain thresholds.

Enhancements to RMF in z/OS V1R9 provide information for large pages in the SMF71 record and in the RMF reports RMF Postprocessor Paging Activity and RMF Monitor III Resource and Storage.

**Who should use large pages?**

Large page support is a special-purpose performance improvement feature not recommended for general use. Its usage provides performance value to a select set of applications that are primarily long running memory access intensive applications. Some applications can be severely degraded by the use of large pages. Short-lived processes with small working sets size are not good candidates for large pages. Consider memory usage and a workload’s page translation overhead when trying to either estimate the potential benefit or understand measured performance differences using pages larger than 4 KB.

Because large page support can be used by both authorized programs and unauthorized programs (with read authority to facility class IARRSM. LRPAPAGES), and are non-pageable and consume real storage, carefully consider which applications are granted access to large pages. Key factors to consider when granting access to large pages are memory usage, workload page translation overhead, and available large frame area.

**References**

- z/OS Initialization and Tuning Reference, SA22-7592
- z/OS Initialization and Tuning Guide, SA22-7591

---

**Table 1**: Summary of how System z10 hardware and z/OS releases handle large page support requests with or without eDAT architecture support.
In today’s world, environmental issues continue to gain serious momentum. Many data centers have maxed out their power from the grid, and constant fluctuations of energy costs are rampant. Let’s not forget the belt-tightening forced on many companies as a result of the current financial crisis. Not surprisingly, IT executives are scrutinizing energy costs because these costs are becoming an important consideration in green data center server solutions. In some cases, energy costs may be a limiting factor in IT decisions!

As shown in Figure 1, mainframe customers need not lose heart. The mainframe boasts increased performance per unit of energy consumption for newer generations. IBM continues to innovate on this platform.

Techline’s System z energy use and efficiency estimate offering
In the current IT landscape, you should know the energy impact of your next server upgrade or acquisition before you submit the next proposal for approval. With a new offering, the IBM Techline Capacity Planning and Sizing Team can help estimate the energy impact of System z servers. This offering is available at no additional cost.

The Mainframe Energy Comparison Tool provides estimates of energy savings and efficiency gains for upgrades of n-1 and older servers (9672, z800, z890, z900, z990, z9® BC, z9 EC) to target servers (z9 BC, z9 EC, z10 BC, z10 EC™).

Estimates are provided for:
- Power and cooling total watts
- Power and cooling costs
- Power and cooling costs per MIP
- Energy efficiency (performance per kilowatt (kW)).

You can use this tool for energy comparison of consolidation scenarios in which two or three existing servers are consolidated onto a single z9 or z10 server. Or, you can use the tool to compare an existing server to up to three different target servers being considered for the upgrade. At a minimum, the inputs required for the analysis are the text configuration files for the servers under consideration.

How it works
Based on the configuration files provided, the tool computes the power consumption for the existing servers and the target servers. The power for cooling the data center allocated for one or more servers is estimated. The total watts required for power and cooling is then computed. A total monthly and annual cost of power and cooling is also computed using the installation’s cost per kW-hour.

Other important metrics evaluated when comparing energy consumption for different mainframe generations are the energy efficiency (performance per kW) and the power and cooling cost per MIP. The result is a high quality report, like Figure 2, (including tables and charts) showing the details of the analysis.
Go Green! Contact Techline for an estimate
To get a Techline analysis, contact your IBM account team and fill-out the request form in the WP101396 techdoc available on the following web site:

ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101396

Follow the instructions for sending your data to Techline for an analysis. A specialist will run the analysis and provide a report for your review.
Swap disk failure for continuous availability

Using Basic HyperSwap to prevent system and application outage

BY WILLIAM J. ROONEY, GREGORY E. MCBRIDE, AND TARIG HANIF

In April 2008, IBM announced a new and exciting function called Basic HyperSwap. Basic HyperSwap is designed to improve z/OS availability by allowing z/OS to recover from the failure of one or more volumes (entire storage controllers even) without causing system or application outage.

What is Basic HyperSwap?

Basic HyperSwap is a low-cost, single-site, continuous-availability disk solution that allows the configuration of disk-replication services using an intuitive graphical user interface (GUI). It is designed to provide seamless swapping between primary and secondary disk volumes in the event of planned and unplanned outages such as hardware maintenance, testing, or device failure.

Basic HyperSwap helps eliminate disk failures as a source of application outages by letting you specify a set of storage volumes to synchronously mirror. For example, in the event of a permanent I/O error, Basic HyperSwap automatically switches I/O requests to the secondary copy, thereby masking the failure from the application and avoiding a restart of the application (or system) after the failure. You can also initiate a planned failover to a secondary disk before performing hardware maintenance on primary storage controllers (or simply to periodically test the function).

Basic HyperSwap is enabled through a no-charge product called Tivoli Storage Productivity Center (TPC) for Replication Basic Edition (which was formerly called Productivity Center (TPC) for Replication a no-charge product called Tivoli Storage). TPC for Replication provides device discovery, the GUI, and the configuration repository for Basic HyperSwap.

TPC for Replication Basic Edition is also part of a family of disk replication products, which allow customers to easily exploit more sophisticated storage replication solutions, such as the recently announced HyperSwap Global Mirror, while continuing to use the same set of products they have already installed and learned how to use.

If you are already a z/OS V1R9 or later customer, it is relatively inexpensive to start using Basic HyperSwap. Because it uses synchronous mirroring, you do not have to replicate all of your volumes. Instead, you can start small, say with just critical system volumes such as the system residence (SYSRES), spool, catalog, paging, and the zFS root. As the value of Basic HyperSwap becomes more apparent, you can add other volumes, such as key application databases, to the replication session.

Requirements for Basic HyperSwap

In order to begin using Basic HyperSwap, you will need:

- z/OS V1R9 (with PTF for APAR OA20658), or later
- TPC for Replication Basic Edition (a no charge product), or one of the priced features of TPC for Replication
- DB2 V8.1 or later (if you want to use DB2 as the repository). Alternatively, you can use an embedded database included in the TPC for Replication product.
- An IBM storage controller with the Advanced Copy feature Metro Mirror (Such as IBM ESS Models 800, DS6000™ and DS8000)
- On DS8000, you also need the RM Ethernet Adapter Feature

Installing Basic HyperSwap

The TPC for Replication server is installed using SMP/E instead of an ad hoc script or process. This might appear to be more complex than using launchpad.bat or setup.exe, but it provides the control and documentation that z/OS system programmers have come to expect.

Post installation tasks and tips

TPC for Replication is a WebSphere application, and includes a copy of IBM WebSphere Application Server OEM Edition for z/OS. The IWNINSTL job sets up the post SMP/E install applications, updates the properties and configuration files, installs the application, and updates permissions. It also points the properties files to the correct locations on your machine. To customize and run the IWNINSTL job, refer to the TPC for Replication Installation and Configuration Guide, SC32-0102. In particular, pay careful attention to filling in the WebSphere Application Server-related parameters.

After successfully running the post-install job, refer to the manual’s post-installation tasks for z/OS to set up a JDBC provider, and a data source for TPC for Replication that will provide a connection to the database. After configuring the data source, test the connection to the database. If the connection does not work, verify that you have set the paths for the DB2 .jar files correctly in the WebSphere Application Server environment variables. If the connection is successful, that means WebSphere Application Server is able to communicate to your database.
Here are a few post-installation hints and tips.

- If you are using the WebSphere administration console to configure the JDBC provider and data source, and certain command buttons described in the TPC for Replication Installation and Configuration Guide, SC32-0102, do not appear (such as “New” or “Delete”), this is probably because the user ID that is logged on to the WebSphere administration console does not have sufficient authority. The user ID must have administrator authority to perform such configurations.

- Make sure that IBM WebSphere Application Server OEM Edition for z/OS is able to use the ports as defined in the WASOEMdflt.cfg file. To do this, make sure that the configuration for the PORT section of the TCP/IP profile reserves sixteen consecutive ports for WebSphere Application Server. The port range should start with the same value that the variable _SRE_PORT_BASE_ is set to in the IBM WebSphere Application Server OEM Edition for z/OS configuration file. The default value is 32200.

- The default configuration after installing IBM WebSphere Application Server OEM Edition for z/OS may not be optimal for TPC for Replication. You will likely get better performance and lower CPU utilization by tuning.

For WebSphere tuning information, see IBM Redbooks Performance Monitoring and Best Practices for WebSphere on z/OS, SG24-7269, available from:

www.redbooks.ibm.com

Configuring Basic HyperSwap

You can configure a TPC for Replication Basic HyperSwap session through the TPC for Replication GUI or Command Line Interface (CLI). The following steps use the GUI. There are five steps required to configure and start a session:

1. Add storage systems to the TPC for Replication configuration.

Use the Add Storage system wizard that is available from the Storage Systems panel to add the source and target storage systems to the TPC for Replication configuration.

2. Create Peer to Peer Remote Copy (PPRC) paths (optional).

We recommend that you create two PPRC paths between each source and target storage system. Go to the ESS/DS Paths panel from the left-hand navigation panel, and select Manage Paths. Select a source and target storage system along with the logical storage subsystem and port numbers for the PPRC paths. If you skip this step, TPC for Replication creates a single path between each of the source and target logical subsystem pairs when the Basic HyperSwap session is started.

3. Create a TPC for Replication Basic HyperSwap session.

From the TPC for Replication health overview panel, select Sessions. Select the Create Session button to start the create session wizard, and then select the Basic HyperSwap session. Proceed through the wizard filling in the session name, description (optional), and assign location labels to the source and target sites. The newly created session now appears in the session overview.

4. Add copy sets (volume pairs) to the TPC for Replication Basic HyperSwap session.

Select the new session. From the Select Action drop down, choose Add Copy Sets to start the add copy set wizard. The wizard allows the matching of all the volumes between like-configured source and target storage systems, like-configured logical subsystems, or on an individual pair basis.

5. Start the Basic HyperSwap session.

From the session overview or the session details panel, select the Start option from the Select Action drop down and confirm that you want to start the TPC for Replication session.

Managing Basic HyperSwap

The default when starting the Basic HyperSwap session is for the session to come up with HyperSwap enabled. As soon as the source and target volume reach the prepared state, the configuration is loaded into z/OS and all of the volumes have protection against source volume failures.

You can disable and enable the HyperSwap function using the View/Modify Properties option on the Select Action drop down. z/OS operator commands are also available to enable or disable HyperSwap.

You can add or remove copy sets from the TPC for Replication session at any time. When you add a new copy set to the session, the source volume is protected after the new copy set reaches the prepared state.

Want more features?

There are two priced features of TPC for Replication available:

- Tivoli Storage Productivity Center for Replication for System z adds the Metro Mirror, Global Mirror, and FlashCopy® session types.

- Tivoli Storage Productivity Center for Replication for System z Three Site BC adds the Metro Global Mirror and HyperSwap Global Mirror session types.

If you want to upgrade to one of these, it is as simple as purchasing a license for the new feature and installing it.
Weight! Weight! Do tell me!

Max out your LPAR throughput with HiperDispatch

BY E. OZAN (OZ) BARAN 
AND M. ALEX CARABALLO

In z/OS Integration Test, we run with HiperDispatch enabled on z/OS logical partitions (LPARs) on our System z10 Enterprise Class Server. An LPAR gets a percentage of the physical processors for which it has logical processors defined based on its LPAR weight. For brevity, we call this the “share” of the central processor complex (CPC) for the LPAR. When we received HiperDispatch, we observed some processor bottlenecks and other issues related to latent demand on the largest one of these LPARs. Because a high processor demand compared to CPC share for the LPAR can cause such issues, we decided to take a closer look at our LPAR weight settings to see what they were telling us. The changes we were able to make resulted in an increase of our throughput and a drop of our LPAR management time.

In this article, we want to share with you our experiences, the methodology you can use to perform a quick simple check of your configuration, and how to optimize your LPAR weight settings in order to maximize HiperDispatch benefits.

Please remember that even though this article is focused on general purpose processors, HiperDispatch manages the IBM System z Application Assist Processor (zAAP) and System z Integrated Information Processor (zIIP) as well and that the principles that apply to general-purpose processors also apply to zAAPs and zIIPs.

Relationship between HiperDispatch and LPAR Weights

First let’s look at how HiperDispatch and LPAR weights relate to each other. The goal of HiperDispatch is to redispach a unit of work to the same subset of physical processors to optimize the usage of the hardware processor caches. With HiperDispatch, LPAR weights are used to determine the LPAR share and how many of an LPAR’s assigned logical CPs (LCP) can be “PARKED.” A “PARKED” processor cannot be dispatched by z/OS and does not attempt to run work.

Tell me how this works

Let’s look at an example to understand. Assume that we have a CPC with 40 processors, 4 z/OS LPARs, all within the same sysplex, and that the LPAR weight and LCP assignments are as described in Table 1.

<table>
<thead>
<tr>
<th>LPAR name</th>
<th>Initial weight</th>
<th>LCPs assigned</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYS1</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>SYS2</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>SYS3</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>SYS4</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>TOTAL</td>
<td>400</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 1. Relationship between LPAR weights and assigned LCPs for four systems in a sysplex
First, we calculate what is called an LPAR fair share to help us figure out the number of processors an LPAR will be guaranteed at any point in time as follows:

\[
\text{LPAR fair share} = \frac{\text{LPAR weight}}{\text{Sum of LPAR weights}}
\]

Next, we can calculate the number of processors guaranteed to the LPAR:

\[
\text{Number of processors guaranteed} = \text{LPAR fair share} \times \text{Total number of processors on CPC}
\]

\[
\text{Number of processors Guaranteed} = N \times M
\]

Where,

\(N\) = Number of processors that will be completely assigned to this LPAR. These are considered high-share processors.

\(M\) = The fraction of a processor that will be assigned to this LPAR. These are considered medium-share processors.

Let’s take SYS1 in Table 1 as an example:

SYS1 fair share = \(\frac{100}{400} = 0.25\)

SYS1 number of processors guaranteed = \(0.25 \times 40 = 10.0\)

\(N = 10\)

\(M = 0\)

Because SYS1 is assigned 20 LCPs, 10 of these LCPs now can be dispatched on these 10 processors \((N+M)\). The remaining 10 LCPs are considered low-share processors and are “PARKED.” They will be “UN-PARKED” when SYS1 needs additional capacity and only if physical capacity, in the CPC, is available. Also, because SYS1 M is less than 50% and it has low-share processors available to it, PR/SM™ converts one of the 10 high-share processors to a medium-share processor so that the following is assigned to SYS1:

- Nine processors as high-share processors
- One processor as a medium-share processor
- Ten processors as low-share processors

Checking the LPAR weights

Ensure that the initial and minimum weights of your LPARs are not the same.

Assuming that you are exploiting Intelligent Resource Director (IRD) LPAR weight management, if you set the initial and minimum LPAR weights the same, WLM cannot decrease the weight of one LPAR in order to increase that of another LPAR. This minimizes the effectiveness of HiperDispatch.

If WLM could make the dynamic weight changes, the LPAR fair share and the number of guaranteed processors for each LPAR would be recalculated and result in a higher degree of affinity between each LPAR and the processors assigned to it.

Ensure that your LPAR weights are aligned with the processor needs of your LPARs.

First calculate the number of processors each LPAR needs as follows:

\[
\text{Number of processors needed to satisfy LPAR demand} = \text{Number of LCPs assigned} \times \text{Average LCP utilization percentage}
\]

Let’s go back to our example in Table 1. We listed the number of LCPs assigned to each LPAR in the table. We looked up the average LCP utilization percentage for each LPAR in the PARTITION DATA REPORT section of the RMF CPU ACTIVITY REPORT. See Table 2.

<table>
<thead>
<tr>
<th>LPAR name</th>
<th>High-share processors assigned (per earlier calculations)</th>
<th>LCPs assigned</th>
<th>Average LCP utilization percentage</th>
<th>Number of processors needed to satisfy LPAR demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYS1</td>
<td>9</td>
<td>20</td>
<td>90%</td>
<td>18</td>
</tr>
<tr>
<td>SYS2</td>
<td>9</td>
<td>20</td>
<td>10%</td>
<td>2</td>
</tr>
<tr>
<td>SYS3</td>
<td>9</td>
<td>20</td>
<td>10%</td>
<td>2</td>
</tr>
<tr>
<td>SYS4</td>
<td>9</td>
<td>20</td>
<td>10%</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 2. Average LCP utilization and processors needed to satisfy LPAR demand
Clearly most LPARs do NOT need the nine high share processors assigned to them (for example, SYS2 needs two but is assigned nine). On the other hand, SYS1 needs double what it is assigned. Because SYS1 is assigned only nine processors but is using 18 processors worth of capacity, it is getting the remaining nine processors worth of capacity by running on low-share processors.

The low-share processors of SYS1 run in place of the high-share or medium-share processors of another LPAR that do not have any work to run. As soon as the high-share or medium-share processor of the LPAR wants to run, the low-share processor of SYS1 stops running and must run on another processor or wait to run again. This is because low-share processors are guaranteed far less access to processor than high-share or medium-share processors. HiperDispatch provides the most value when all LPARs on the CPC are able to dispatch all work on their high and medium share processors.

Solution and results
Going back to the z/OS Integration Test, in order to tune our environment, we followed these guidelines:

1. We studied LPAR utilizations over time and assigned LPAR weights so that each LPAR would be initialized with a number of CPs assigned to it as high-share CPs that closely matched its workload.

2. We assigned each LPAR distinct minimum, maximum, and initial weights to allow IRD LPAR weight management to dynamically manage the LPAR weights and adjust the number of high and medium processors on each LPAR as needed. This helps each LPAR to run on its high-share and medium-share processors as much as possible to get the most value out of HiperDispatch.

These changes resulted in an increase of our throughput and a drop in our LPAR management time.

Lessons learned: do tell me
Our goal was to maximize throughput and minimize LPAR management time.

Without HiperDispatch, it was important to line up the LPAR weight definition with the number of LCPs assigned to an LPAR in order to reach both these goals.

With HiperDispatch, it was more important to set the LPAR weight settings based on the desired apportionment of capacity to the partitions. This can help allow z/OS and PR/SM to better optimize cache. We didn’t need to worry much about defining exactly the right amount of LCPs to achieve both these goals. If you are in doubt, you can assign more LCPs and let HiperDispatch park the ones that are unnecessary.

Also, remember to follow the recommendations from the “Planning Considerations for HiperDispatch Mode” IBM white paper by Kathy Walsh and Steve Grabarits at the following Web site:

ibm.com/support/techdocs/atsmastr.nsf/WeblIndex/WP101229

Go simple, go green, go z/OS

- Dave Hans,
z/OS User Experience and Design manager
The good ol’ days
They were large, noisy beasts quite unlike anything seen in a modern data center. The 1403 was representative. With its two-speed, hydraulically-driven paper carriage controlled by a punched paper tape, tractor pin feeds for perforated fan-fold paper, high-speed motor-driven print train or chain, and 132 separate print hammers (one per print position), it could print hundreds of lines per minute, or up to 13 pages a minute, depending on the print train and the form. A print train comprised slugs with characters on their faces, running in a track.

Normally, characters were duplicated several times on the train to speed printing; trains with many special characters printed more slowly because there were fewer slugs with each character on the train. Spun around by a large electric motor at high speed, these slugs formed the outer layer of a slug-ink ribbon-paper sandwich. When the right character was going by the right column on the paper, a print hammer would hit the paper from behind, driving it into the inked ribbon and both of them into the character on the passing slug. With the cover open, a running 1403 was incredibly—even painfully—loud.

The operator would change the reversing printer ribbon when the characters started to lighten or when the ribbon jammed. The ribbon could jam when the pattern of print across the form was uniformly heavy on one side and light on the other. The constant pounding on one side (usually, the left) would cause that side of the ribbon to stretch more than the other, and sooner or later it would begin to run off the spools, tripping a lever and stopping the printer. Inside every ribbon box was a pair of plastic gloves to use when you changed the ribbon.

With no ribbon installed, a 1403 would happily print blank pages. More than one new operator was fooled into thinking that super-secret reports were being printed in invisible ink. Such an operator would jam the printer when the pattern of print across the form was uniformly heavy on one side and light on the other. The constant pounding on one side (usually, the left) would cause that side of the ribbon to stretch more than the other, and sooner or later it would begin to run off the spools, tripping a lever and stopping the printer. Inside every ribbon box was a pair of plastic gloves to use when you changed the ribbon.

With no ribbon installed, a 1403 would happily print blank pages. More than one new operator was fooled into thinking that super-secret reports were being printed in invisible ink. Such an operator would jam the printer when the pattern of print across the form was uniformly heavy on one side and light on the other. The constant pounding on one side (usually, the left) would cause that side of the ribbon to stretch more than the other, and sooner or later it would begin to run off the spools, tripping a lever and stopping the printer. Inside every ribbon box was a pair of plastic gloves to use when you changed the ribbon.

Woe betide the operator who left a cup of coffee atop a 1403 Model N1—when the printer ran out of paper, its motor-driven cover would rise to notify the operator that it was hungry again. This notification was especially effective if the operator was checking to see how full the output stacker was and the coffee was hot (ouch!). However, with its full-length covers, the Model N1 was noticeably quieter than earlier models when its covers were closed.
2821 control units
Both the 2540 and the 1403 shared a common control unit, the 2821. Because it used core storage to keep track of the characters on the 1403’s print train in something called a Universal Character Set (UCS) buffer, the 2821 could usually be powered down and back up without the UCS buffer having to be reloaded. The 2821 was based on Standard Modular System (SMS) logic, which consisted of discrete components—transistors, resistors, capacitors, and inductors—soldered to phenolic cards. The cards were plugged into large boards, and card-to-card connections on and between boards were made with wire-wrap technology. SMS was the last technology that could actually be repaired in the field at need, although official policy was to replace a failed card.

Problems
The most common problem with 1403s was a print-hammer failure. A print hammer could be changed quickly by an experienced customer engineer (CE). When properly installed and “flight timed,” the print hammer would strike the paper at exactly the right time to keep the characters in that column properly centered between adjacent rows. Visually, all 132 flight hammers resembled a piano keyboard, and a bad or maladjusted hammer often resembled a key that would not return completely to its upright position next to the others on the keyboard. If the hammers were all aligned but some were not printing centered characters, they had to be flight timed, which required running a test program called OLTEP (On-Line Test Executive Program) and making adjustments in the timing circuitry in the 2821 control unit.

The revolution: 3800 laser printers and beyond!
In 1975, we announced the 3800, and data centers were never the same. The laser-based 3800 printed 20,000 lines a minute; that was over 150 pages per minute with most forms. It was over ten times the speed of the 1403. Running flat out, the 3800 could empty a fresh box of paper in under 10 minutes. It did not take many 3800s to keep print pool operators hopping during peak printing times. Roll-feed paper followed quickly. The rolls of paper, about five feet in diameter, weighed… a lot. Special equipment was used to move these big rolls into the storage rooms, and from there to the print pools (and the raised floor tiles along the way were replaced with stronger ones to bear the weight). The rolls solved a big part of the problem—feeding the printers. The other half of the problem, emptying the printers and sorting the output, remained.

Burn! Tear! Shred!
Another new feature of the 3800 came along to help with that, the Burster, Trimmer, and Stacker, or BTS. Known by some as the “burner, tears, and shredder,” the BTS had some early teething problems that were quickly addressed. It would separate the sheets and stack those for each job on a short conveyor, with each job’s output overlapping the prior one. This way, operators did not have to search for the header pages and separate the output before putting it in the bins from which people would pick it up. Between the 3800 itself and the BTS, threading a 3800 from scratch took a while. Splicing tape was used to join the tail end of one roll (or box) to the start of the next to avoid having to re-thread the entire machine when the paper supply ran dry.
Wake up your system with Java SDK6

BY NORM AARONSON

What's happened in the land of Java on z/OS since the last Hot Topics article on the subject? (See "Wake up your system! New enhancements and products for Java on z/OS" in z/OS Hot Topics Newsletter Issue 17, August 2007, GA22-7501-13.)

In short, there's been the introduction of two new SDK6 products, less emphasis placed on the SDK1.4.2 products, more enhancements in JZOS and security, improved performance, a cookbook, and a growing importance of 64-bit.

The “new” SDK6 products
Though available since the end of 2007, the SDK6 products are becoming more and more popular. If they aren't already, these products should be the ones you choose:

- IBM 64-Bit SDK for z/OS, Java Technology Edition, Version 6 (5655-R32)

These SDK6-level products:

- Contain the latest JZOS batch technology and z/OS Java security,
- Are particularly useful on zAAP processors, and
- Are imbedded in WebSphere Application Server for z/OS V7.0.

What's more, applications previously constrained by 31-bit addressing limitations can break that barrier with the 64-bit SDK6-level product. That product also contains performance improvements for heaps in the 2-32 gigabyte (GB) range.

What's happening with SDK1.4 products?
IBM SDK for z/OS, Java 2 Technology Edition, Version 1.4 (5655-156) has been withdrawn from marketing as of September, 2008. Although this 31-bit product from 2002 will be in service until September, 2011, no enhancements have been added for years. z/OS V1R11 is the final release of z/OS that will support this product.

The 64-bit SDK1.4.2 product is already out of service.

It is now time for customers to move to the later z/OS Java products to take advantage of their better technology and enhancements.
It is now time for customers to take advantage of better technology and enhancements in the newer z/OS Java products.

JZOS! Updated for the z/OS Java products
z/OS Java continues to be a mainstay of WebSphere Application Server for z/OS V6.1 and V7.0. We’ve also recently added more customer-driven improvements for applications that do not run under WebSphere.

The IBM JZOS Batch Toolkit for z/OS SDK is the set of tools that addresses many of the functional and environmental shortcomings in Java batch capabilities on z/OS. Introduced into the z/OS Java product in 2006, it includes a native launcher for running Java applications directly as batch jobs or started tasks. It also includes a set of Java methods that make access to traditional z/OS data and key system services directly available from Java applications.

We added new enhancements to the toolkit at the end of 2008. These enhancements make it even easier to run Java batch jobs on z/OS (particularly for traditional z/OS programmers).

Cooking with Java?
In the last year, the z/OS Java team has been updating a JZOS cookbook on the IBM alphaWorks® Web site. This cookbook highlights a Java application, packaged in an Eclipse project, using JZOS.

It’s so great to migrate
At this point in time, the SDK5 products are the most predominately used. They are based on the same underlying technology that the SDK6 products use, and take advantage of IBM System z Application Assist Processor (zAAP) processing and the JZOS batch enhancements. They are valuable, and we have no current plans to withdraw service for them. However, don’t expect enhancements to the SDK5 products. If you want enhancements, this would be a great time to migrate. Migration to a new SDK is easy, because z/OS Java SDKs are totally independent of each other. You can order and concurrently run one, two, or all in your installation, and each can be serviced separately.

There are, however, some considerations for migrating your Java applications. Although most SDK1.4.2 Java applications can run on SDK5 and most SDK5 applications can run on SDK6, some applications cannot. See the following Web sites for compatibility considerations:

[Java Compatibility](java.sun.com/j2se/1.5.0/compatibility.html)
[Java Compatibility](java.sun.com/javase/6/webnotes/compatibility.html)

Want more information?
For an overview and details of the IBM z/OS Batch Toolkit for z/OS SDK, refer to the Web site:
[IBM z/OS Batch Toolkit for z/OS](ibm.com/servers/eserver/zseries/software/java/jzos/overview.html)

For more information on the JZOS cookbook, see the Web site:
[IBM z/OS Batch Toolkit for z/OS](ibm.com/servers/eserver/zseries/software/java/jzos/overview.html)

For everything else you ever wanted to know about Java for z/OS, see the Web site:
[IBM z/OS Batch Toolkit for z/OS](ibm.com/servers/eserver/zseries/software/java/jzos/overview.html)
The incredible shrinking 64-bit object!

Better than 31-bit performance using 64-bit SDK

BY KISHOR PATIL, LEVON STEPANIAN, AND THERESA TAI

The 31-bit software development kits (SDKs) have traditionally provided better application performance and smaller memory footprint compared to 64-bit SDKs. As Java applications grow in complexity and scale, 31-bit applications are pushing the limits of Java and native heaps.

While not memory-constrained, 64-bit virtual addressing inflates average object size by 40%. Additionally, it reduces data locality (because the distance between related objects in memory grows), increases data cache and translation look-aside buffer (TLB) miss rates, and results in fewer objects occupying the same-sized heap. The latter condition increases garbage collector activity, but you can address it by using a larger heap.

64-bit SDK V6 (z/OS and Linux® on System z) introduces two new features, which allow customers to reduce or eliminate the performance gap between 31-bit Java and 64-bit Java:

- Large heap support with compressed references
- IBM System z10 large page exploitation

Large heap support for 64-bit Java with compressed references

Compressed references reduce 64-bit objects to 31-bit object sizes. The reduction in object size is achieved by strategically placing objects in specific virtual address ranges and exploiting the effect that double-word alignment of objects has on the lower bits of object addresses. Object reference fields are decompressed before being de-referenced and compressed prior to being updated using address shift operations. This eliminates the problems described earlier, but adds a little overhead.

Three different compression/decompression schemes are available varying in path length and heap size. The least-expensive shift-by-0 scheme can attain up to 2 gigabytes (GB) of heap (if the heap-top address is below 2^{32}). The moderately-expensive shift-by-1 can attain up to 6 GB (heap-top address below 2^{33}). Finally, the expensive shift-by-2 or shift-by-3 scheme can attain up to 30 GB of heap (heap-top address below 2^{35}).

Large page support in IBM System z10 can be transparently used by Java-based middleware and applications to obtain significant performance gains by reducing the overhead of page table look-up for real-to-virtual address translation.

Performance results

The DayTrader* 1.2 benchmark measurements were performed on IBM z10 with two dedicated LPARs, 12-way z/OS V1R9 LPAR hosting WebSphere Application Server, Version 7.0 with two servants and an 8-way z/OS V1R9 LPAR hosting DB2. The Java Multi-threaded benchmark measurements were performed on IBM z10 with dedicated 16-way z/OS V1R9 LPAR.

* Copyright © 2009 The Apache Software Foundation, Licensed under the Apache License, Version 2.0. Apache and the Apache feather logo are trademarks of The Apache Software Foundation.

Originally developed by IBM as the Trade Performance Benchmark Sample, DayTrader was donated to the Apache Geronimo community in 2005.
The charts in Figure 1 show 64-bit Java performance compared to the baseline 31-bit Java Performance. The leftmost bar shows the performance gap with the default 64-bit SDK. The subsequent bars towards the right show the performance improvements achieved by enabling the compressed references feature in isolation, the large pages feature in isolation, and finally, both of the features together.

As seen from the benchmark results, compressed references and large pages independently reduce the 31/64-bit performance gap significantly. But when the technologies are combined, 64-bit SDK V6 can match or even out perform 31-bit SDK V6.

**Conclusion: better performance**

64-bit Java compressed references technology and System z10 large page support enable Java-based enterprise applications to overcome the virtual memory constraints of 31-bit, without compromising the advantages of 31-bit such as better throughput and smaller memory footprint!

For more information on these features, prerequisite APARs, usage, and detailed performance analysis, we encourage you to read the article, “Match 31-bit WebSphere Application Server performance with new features in 64-bit Java on System z,” on IBM PartnerWorld® at: ibm.com/partnerworld/wps/whitepaper/systemz/java_websphere/performance

**A note about performance**

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

**Acknowledgements**

The authors acknowledge the IBM Java performance team for the performance measurements in Figure 1.
n today’s large systems computing environment z/OS Capacity Provisioning helps you manage capacity for general purpose processors (CP) and specialty processors (IBM System z Integrated Information processor or zIIP and IBM System z Application Assist processor or zAAP) for System z10 servers that are running one or more z/OS systems. z/OS Capacity Provisioning simplifies the monitoring of critical workloads, and its automation features can help to activate additional resources faster than manual operation. Based on IBM On/Off Capacity on Demand (On/Off CoD) for a temporary capacity increase, you can activate or deactivate using a policy you define.

Capacity Provisioning gives you the flexibility to activate temporary capacity in either of two ways:

- Manually on a given schedule
- Automatically on workload performance requirements

When choosing the workload model, you define a policy to activate capacity within specified periods. This is when your important work might miss its goals because of a capacity bottleneck. Performance information about the workloads, such as their performance index (PI) and delays, help decide when activation level changes are appropriate.

Gather PI
To gather PI information, the Capacity Provisioning Manager, the runtime component on z/OS, observes the workload on your critical z/OS systems. z/OS Workload Manager (WLM) provides the performance information and Resource Measurement Facility (RMF) and Common Information Model (CIM) servers running on each observed system retrieve the information. This function is already available with z/OS V1R9 with APAR OA20824 and part of z/OS V1R10 and V1R11.

Figure 1. Capacity Provisioning configuration
z/OS V1R10 with PTF UA47421 adds the following improvements:

- Logical configuration with logical processor management
- Communication service to the hardware that uses the new z/OS Basic Control Program internal interface (BCPii) instead of TCP/IP connection

The Basic Control Program internal interface (BCPii) is a new z/OS component that programs use to communicate directly to the hardware without the need of any network connection. You can define very granular access rights within your z/OS security manager. To find out all about the BCPii, check out “BCPii: Control your HMC and support element directly from z/OS apps” on page 46.

When using BCPii to communicate to the hardware, the Capacity Provisioning Manager must be able to access these CPCs from every host system it runs on. Therefore the set-up may need to be done on multiple systems. For Capacity Provisioning, access to CEA and BCPii is needed. To access the hardware, security profiles for CPCs and temporary capacity record are needed.

Tips when setting up the service element

- You must specify the SNMP community name (the one you previously specified in the APPLDATA of your HWI. TARGET security profile) in the SE of each CPC that the Capacity Provisioning Manager accesses.
- The community name must have WRITE access.
- To limit the network access to the local host, specify an address of 127.0.0.1 and a network mask of 255.255.255.255.
- As you did for SNMP, you must enable the SNMP APIs and make capacity change API requests allowable.

Tips when setting up Capacity Provisioning for BCPii

The changes to Capacity Provisioning are really simple. For Capacity Provisioning BCPii communication is “just another protocol”. To switch the communication from SNMP to BCPii, add the following configuration key to the hardware access PARM member of the Provisioning Manager parameter data set:

```
Topology.Protocol = INTERNAL
```

After the definitions are complete, you no longer need a TCP/IP connection from the Capacity Provisioning Manager runtime system to the HMC or SE.

Logical processor management

If you already use the VARY CPU management in WLM or vertical CPU management (also called HiperDispatch mode), you might wonder why you want another component to manage your logical processors. Logical processor management in Capacity Provisioning not only adjusts the number of logical processors in conjunction with changes to the number of physical processors but also helps in situations where you have performance bottlenecks because of a low number of logical processors. It can simplify detection of such situations and allow for additional conditions when you can add physical capacity.

Here are the three situations in which you can benefit from logical processor management:

1. All the shared logical processors map to physical processors. The number of logical processors on your system is lower than the number of physical processors. When you still have processors offline and the logical processor utilization is above 95%, you must set a logical processor online.

2. All the shared logical processors map to physical processors. The number of logical processors of your system is equal to the number of physical processors. When processors are still reserved and the system detects the need for additional capacity, it activates new capacity. Afterwards, you must set a logical processor online.

3. When a temporary physical processor needs to be removed by Capacity Provisioning and this would lead to more logical processors than shared physical processors you must set a processor logical offline.

You can decide in the Capacity Provisioning Control Center (see Figure 2) which system will notify you about the logical processor change.

The runtime system is the system where the Capacity Provisioning Manager is running. Messages on the runtime system are as follows:

```
CPO3910I CONFIG ONLINE for processors at system R76 in sysplex SVPLEX7 requested. CP/zAAP/zIIP: 1/0/0
```

When this CPO3910I message appears, the workload does not get a benefit from the physical resource until you set a requested logical processor online in Figure 2, that is a logical general purpose processor.

The managed system is the system where the logical processor is configured online or offline.

Messages on the managed system are as follows:

```
CEZ02000I: Requesting CONFIG ONLINE for CPU <cpu-id>
```

In the CEZ02000I message example, the cpu-id represents a general purpose processor.
Leaving it to WLM
When you are running WLM managed processor handling, Capacity Provisioning does not issue a message but leaves it to z/OS WLM to manage the logical online processors.

Benefits of logical processor management
Capacity Provisioning can inform you through messaging about processor changes. You must define the maximum numbers of processors (CP or specialty) for your system. In addition, you must define which system receives the messages. This can be either the system running Capacity Provisioning Manager (runtime system) or the system that sets logical processor online/offline (managed system).

Define these settings using the Capacity Provisioning Control Center (CPCC) as part of the logical processor scope in the Capacity Provisioning Policy as shown in Figure 2.

In addition, ensure that your LPAR configurations contain enough reserve processors. This helps when Capacity Provisioning adds a physical processor and a reserved processor gets set from reserved to an offline state and thereby can be set online when a request is made. If you have not planned for such situations, you can add reserved processors to a running LPAR using the service element.

To find out more about all the information contained in this article, see z/OS MVS Capacity Provisioning User’s Guide, SC33-8299.
Cryptography is the science of hiding information, whether you are a mother spelling “b-e-d-t-i-m-e” to the father of your child, or an ATM user withdrawing a couple twenties from your local bank. The intent is for the sender and receiver to communicate with each other using a special code to disguise their message, confident that the information they share is not easily understood by whoever might be eavesdropping.

Software applications that exchange data can use special numeric keys to scramble (encrypt) the data using advanced mathematical algorithms. These algorithms make it nearly impossible for an eavesdropper to unscramble (decrypt) the communication data without knowing the encrypting key. Limiting access to the key material is one of the most critical aspects of any cryptographic environment.

In addition to the hardware protection provided by cryptographic coprocessors, the Integrated Cryptographic Services Facility (ICSF), a software element of z/OS, lets you restrict access to key material and cryptographic services. This is done using Resource Access Control Facility (RACF) profiles in various System Authorization Facility (SAF) classes.

Restricting access to ICSF services
You can use the CSFSERV class to restrict access to the ICSF programming interface—the callable services themselves. By creating profiles within the CSFSERV class, you can selectively enable specific users or applications to call particular ICSF services to perform cryptographic operations.

In order to restrict use of a particular service, do the following:

1. Use the RDEFINE command to create a profile for the service:
   
   ```
   RDEFINE CSFSERV CSFKRR UACC(NONE)
   ```

   The above command creates a profile with the name CSFKRR in class CSFSERV with a universal access authority (UACC) of NONE. The CSFKRR profile protects the key record read service, so this will prevent unauthorized users from reading keys from the key store data sets used by ICSF.

2. Use the PERMIT command to give users access to the service. For example, to authorize a user to read keys, you enter the PERMIT command:
   
   ```
   PERMIT CSFKRR CLASS(CSFSERV) ID(MYUSER) ACCESS(READ)
   ```

3. Activate the CSFSERV class (if it is not already active) and place it in common storage.
   
   ```
   SETROPTS CLASSACT(CSFSERV) RACLIST(CSFSERV)
   ```

Restricting access to keys
You can protect ICSF keys by their labels using the CSFKEYS class. The name of the profile matches the name of the key label. For example, if a key has the label MYDESKEY, you could enter the following RDEFINE command to create a profile to protect the key. The SETROPTS command refreshes the CSFKEYS class in common storage.

```plaintext
RDEFINE CSFKEYS MYDESKEY UACC(NONE)
SETROPTS RACLIST(CSFKEYS) REFRESH
```
In WARN mode, a calling application will not fail if access is denied. An SMF record, however, will be generated. You can use these audit records to determine the usage profiles for tokens and set up the necessary security policies before enabling FAIL mode.

The following commands set up a default policy for tokens that do not have a match in the key store:

```
RDEFINE CSFKEYS CSF-CKDS-DEFAULT UACC(NONE)
RDEFINE CSFKEYS CSF-PKDS-DEFAULT UACC(NONE)
```

To restrict the creation of duplicate tokens (the situation in which a token that is already associated with one key label is stored again under a different key label), create the following RACF profiles:

```
RDEFINE XFACILIT CSF.CKDS.TOKEN.NODUPLICATES
RDEFINE XFACILIT CSF.PKDS.TOKEN.NODUPLICATES
```

Be aware that the key store policy control for token checking must be enabled in order to enable any of the controls described in the rest of this article.

**Enabling granular access to keys**

The granular key label access controls allow you to assign different levels of key access for each user/application depending on the intended function. Specifically, you can allow key data to be read, but restrict key data from being updated or created. See Table 1.

```
RDEFINE XFACILIT CSF.CKDS.TOKEN.CHECK.LABEL.WARN
RDEFINE XFACILIT CSF.PKDS.TOKEN.CHECK.LABEL.WARN
```

```
RDEFINE XFACILIT CSF.CKDS.TOKEN.CHECK.LABEL.FAIL
RDEFINE XFACILIT CSF.PKDS.TOKEN.CHECK.LABEL.FAIL
```

In WARN mode, a calling application will not fail if access is denied. An SMF record, however, will be generated. You can use these audit records to determine the usage profiles for tokens and set up the necessary security policies before enabling FAIL mode.

The following commands set up a default policy for tokens that do not have a match in the key store:

```
RDEFINE CSFKEYS CSF-CKDS-DEFAULT UACC(NONE)
RDEFINE CSFKEYS CSF-PKDS-DEFAULT UACC(NONE)
```

To restrict the creation of duplicate tokens (the situation in which a token that is already associated with one key label is stored again under a different key label), create the following RACF profiles:

```
RDEFINE XFACILIT CSF.CKDS.TOKEN.NODUPLICATES
RDEFINE XFACILIT CSF.PKDS.TOKEN.NODUPLICATES
```

Be aware that the key store policy control for token checking must be enabled in order to enable any of the controls described in the rest of this article.

**Enabling granular access to keys**

The granular key label access controls allow you to assign different levels of key access for each user/application depending on the intended function. Specifically, you can allow key data to be read, but restrict key data from being updated or created. See Table 1.

```
RDEFINE XFACILIT CSF.CSFKEYS.AUTHORITY.LEVELS.WARN
```

```
RDEFINE XFACILIT CSF.CSFKEYS.AUTHORITY.LEVELS.FAIL
```

Placing restrictions on key export

The symmetric key label export controls enable you to independently control who can export a data key using the Symmetric Key Export callable service (CSNDSYX). This enables you to allow access to a data key for other purposes, while preventing it from being exported. Two profiles exist in the X FACILIT class to enable these controls. Each profile enables the control for the corresponding key type (AES or DES).

To enable the controls for both AES and DES keys, enter:

```
RDEFINE XFACILIT CSF.XCSFKEY.ENABLE.AES
RDEFINE XFACILIT CSF.XCSFKEY.ENABLE.DES
```

This performs a SAF check (based on the key label) in the XCSFKEY class and requires UPDATE authority. If the XCSFKEY class is not defined to RACF or is not activated, the SAF check will be against the CSFKEYS class and will require only READ authority. If there is no profile covering the resource in the XCSFKEY class, access is not be granted.

<table>
<thead>
<tr>
<th>Required access level</th>
<th>Without granular access enabled</th>
<th>With granular access enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read a key</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>Create a new key</td>
<td>READ</td>
<td>UPDATE</td>
</tr>
<tr>
<td>Write to an existing key</td>
<td>READ</td>
<td>CONTROL</td>
</tr>
</tbody>
</table>

**Table 1. Increased access requirements when granular access is enabled**

There are granular key label access controls for both WARN and FAIL mode.

To enable the control for WARN mode, enter:

```
RDEFINE XFACILIT CSF.CSFKEYS.AUTHORITY.LEVELS.WARN
```

To enable the control for FAIL mode, enter:

```
RDEFINE XFACILIT CSF.CSFKEYS.AUTHORITY.LEVELS.FAIL
```
Controlling how asymmetric keys are used and how symmetric keys are exported

By applying the PTF for APAR OA28855 to ICSF FMID HCR7751 (Cryptographic Support for z/OS V1R8-V1R10 and z/OS.e V1R8 Web deliverable), you can gain more control over key export. This is accomplished using the PKA Key Management Extensions control and the ICSF segment of profiles in the CSFKEYS and XCSFKEY classes. The format of the ICSF segment is shown in Figure 1.

```
[ ICSF(
  [ ASYMUSAGE( SECUREEXPORT | NOSECUREEXPORT )
    HANDSHAKE | NOHANDSHAKE ) ]
  [ SYMEXPORTABLE( BYANY | BYLIST | BYNONE ) ]
  [ SYMEXPORTCERTS( cert-label1 ... )
    ADDSYMEXPORTCERTS (cert-label1 ... )
    DELSYMEXPORTCERTS (cert-label1 ... )
    NOSYMEXPORTCERTS ]
  [ SYMEXPORTKEYS( key-label1 ... )
    ADDSYMEXPORTKEYS (key-label1 ... )
    DELSYMEXPORTKEYS (key-label1 ... )
    NOSYMEXPORTKEYS ]
) ]
NOICSF
```

ICSF will check this segment if the CSF. PKAEXTNS.ENABLE profile exists in the XFACILIT class.

The ASYMUSAGE field is used to restrict an RSA key from being used with certain services.

- The HANDSHAKE services are PKA Encrypt, PKA Decrypt, Digital Signature Generate, and Digital Signature Verify.
- The SECUREEXPORT services are Symmetric Key Generate, Symmetric Key Export, and Symmetric Key Import.

For example, if the ICSF segment specifies ASYMUSAGE(NOHANDSHAKE), the key cannot be used in any of the PKA encryption or digital signature services, but can be used by default for the secure export services.

The SYMEXPORTABLE field determines whether a particular AES or DES data key can be exported with an RSA key.

- BYANY means it can be exported by any RSA key.
- BYNONE means it cannot be exported at all.
- BYLIST means the RSA export key must be in one of the SYMEXPORTCERTS or SYMEXPORTKEYS lists. The SYMEXPORTKEYS field contains a list of key labels identifying RSA keys stored in the PKDS. The SYMEXPORTCERTS field contains a list of certificate labels identifying certificates stored in either a SAF key ring or an ICSF PKCS #11 Token. (In the context of PKCS #11, a token is a representation of a cryptographic device, such as a smart card reader. See the “Protecting PKCS #11 Resources” sidebar on this page.)

After you have set up the ICSF segment specifications, you can enable the PKA Key Management Extensions control for WARN of FAIL mode.

To enable the control for WARN mode, enter:

```
RDEFINE XFACILIT CSF.PKAEXTNS.ENABLE.WARNONLY
```

To enable the control for FAIL mode, enter:

```
RDEFINE XFACILIT CSF.PKAEXTNS.ENABLE
```

For more information

As you might know, RACDCERT is the RACF command for creating digital certificates on z/OS. It's been around for more than a decade, but there is still some confusion surrounding the command. Want some "bits" of information about RACDCERT? Need some "tips" on how to use it? Here are some "tipbits" that might help you understand the basics.

Profiling the certificate profile

We know that RACF is all about profiles. The implementation of certificates in RACF also follows this profile path. A certificate is represented by a profile. The certificate itself is stored in a field in the profile. When a certificate is installed or created in RACF, you need first to create a profile. The certificate profile name takes the following form:

\texttt{RACDCERT "tipbits"}

Building the certificate profile

The maximum length of any RACF profile name is 246 characters, and the certificate profile name is no exception. A long serial number or distinguished name might cause the profile name to exceed the limit allowed. Normally 246 characters are long enough to hold the serial number and the distinguished name. But we've recently seen customers buying certificates from a well-known Certificate Authority (CA) but failing to install them in RACF. The reason? The CA's certificate has a very long distinguished name. The company put a very long certificate practice statement, which should be the content of a certificate extension, into multiple components in the distinguished name. This produces a certificate profile name that exceeds the 246-character limit.

Tipbit: Before you buy certificates to install with RACF, be sure to check the length of the distinguished name of the issuer. When you create a signing certificate, use a proper distinguished name. IBM plans to support certificates with longer distinguished names on z/OS V1R10 and V1R11 in the first quarter of 2010.

Controlling the certificate content

When you are creating a certificate through RACDCERT GENCERT, consider two kinds of supplied keywords. One kind of keyword is used for the certificate content like the following keywords:

- \texttt{SUBJECTSDN} (subject distinguished name)
- \texttt{NOTBEFORE/NOTAFTER} (validity period)
- \texttt{ALTNAME} (alternate name extension).

The other kind of keyword is used to indicate the attributes of the key pair like the following keywords:

- \texttt{SIZE} (key size of the public/private key)
- \texttt{PCICC/ICSF} (where the key pair is generated and stored).
All these keywords give you some control over how the certificate and key are generated. However, RACF needs additional information to generate certificates. In order not to complicate the RACDCERT command, RACF uses some hard-coded values, like the signing algorithm that is used to sign the certificate.

Signing algorithm
To sign the certificate two cryptographic steps are required. The first step is to hash all the certificate information. The second step is to encrypt the hashed result with the private key of the CA. There are many choices of hashing and encryption algorithms. Currently RACDCERT uses SHA1, which is thought to be more secure than MD5. But the hashing algorithms used to generate certificates are often changed over time to follow the National Institute of Standards and Technology (NIST) recommendations.

Certificate ownership
Once in a while, concerns arise from auditors about the ownership of RACF certificates. As we discussed in the previous paragraph, RACDCERT uses a RACF profile to represent a certificate. Because the certificate profile is different from other types of RACF profiles, however, auditors sometimes become concerned about the ownership of RACF certificates.

Tipbit: Unlike the pre-loaded certificates in your browser, which you are using without even being aware of their existence, pre-loaded certificates are installed in RACF with the NOTRUST status. To use these certificates, you have to alter the status to TRUST. In other words, as long as you don’t change the status to TRUST, you can treat these pre-loaded certificates as if they are not there.

Pre-loaded certificates
Another concern is about pre-loaded well-known CA certificates in RACF. Some of them might expire during the life of the release, and no automatic update occurs to remove such certificates.

More information
For information about the latest certificate and security enhancements for RACF and z/OS V1R11, see the following Web site, and search on “Security”:

ibm.com/common/ssi/cgi-bin/ssialias
?subtype=ca&infotype=an&appname=iSource&supplier=897&letternum=ENU S209-029

You can find more about the RACDCERT command syntax in the RACF Command Language Reference, SA22-7687. For implementation examples, see the RACF Security Administrator’s Guide, SA22-7683.

Tipbit: The owner field in a certificate profile does not contain the owner ID of the certificate. It contains the ID from which the RACDCERT GENCERT command used to generate the certificate is issued. But consider this scenario: The person who generated certificates for others in an installation might have left the company. Nevertheless, the certificates created by that person are still valid although the deleted ID is still in the owner field.

Most everything from A to z/OS

- Susan Z. Demkowicz,
  System z Software Developer
Our Contributors

Norm Aaronson, an IBM Senior Engineer, is the Release and Project Manager for Java on zSeries. His 33 plus years with IBM includes projects in VTAM®, Cornell Supercomputing, VM, RS/6000® SP™, and IEEE Standards.

Bob Abrams is a Senior Technical Staff Member in z/OS Development at IBM Poughkeepsie. He has been with IBM for over 30 years. Bob’s focus is on reliability, availability and serviceability (RAS) and problem determination for the z/OS platform.

Ogechi O. Anunobi is an IBM IT Specialist with Techline Global Center of Excellence in Dallas, Texas. She is a member of the System z Capacity Planning and Linux Server Consolidation Sizing team. She holds a Master of Science degree in Management Science and Engineering from Stanford University and a BS in Electrical/Electronic Engineering from University of Port Harcourt.

E. Ozan (Oz) Baran is a Software Engineer for the z/OS Integration Test team at IBM Poughkeepsie. He is currently working on Parallel Sysplex® Performance and a number of BCP components. He holds a Computer Science (BS) degree from Clarkson University and an Information Technology (MS) degree from Rensselaer Polytechnic Institute (RPI).

Friedrich Beichter is the design and development team leader for HCD/HCM. He joined IBM at the Boeblingen Lab in 1977 and since has worked on z/OS, OS/390®, z/VM® and VSE.

Jessica Bonner is a tester in the z/OS Security Area. She has extensive experience with both ICSF and TKE. She has worked closely with IBM customers on a number of occasions and is a presenter at the SHARE conferences.

Stephen M. Branch is an IBM Senior Software Engineer at Silicon Valley Lab whose 25 year career has dealt with all aspects of DFSMS. He is a member of the ICF Catalog Department. Stephen holds a Master of Science Degree from Louisiana State University.

John Canale is a second line manager at IBM Poughkeepsie where he manages the z/OS Enabling Technologies Development Organization. He has been with IBM for 27 years and has held various technical and management positions in z/OS Development.

M. Alex Caraballo is a Software Engineer for the z/OS Integration Test team at IBM Poughkeepsie. She spent 18 years working on z/OS Storage (DASD and tape), z/OS I/O, systems performance and z/OS base components. She holds a Computer Science (BS) degree from Universidad Central of Venezuela.

Jeff Cates is a Senior Software Engineer in z/OS Communications Server development at IBM Raleigh. In his 28 years with IBM he has helped develop a variety of software products.

Gisela Cheng is an Advisory Software Engineer for z/OS Enabling Technologies at IBM Poughkeepsie, New York. She has worked in various OS system development teams, as well as application development and software integration area. Gisela holds a Master of Science degree from Union College.

Wai Choi, CISSP, works in RACF/PKI design and development. Since joining IBM in 1997, she has been working on digital certificate test, development and design. Wai is a frequent speaker in SHARE, Vanguard, and zExpo conferences. She also actively participates in the RACF-L forum answering certificate related questions.

Curt Cotner is an IBM Fellow and Vice President and chief technology officer for the DB2 family and Informix® IDS database servers, with oversight responsibility for all the client software offerings used with DB2 and IDS including client runtime APIs and the application development and administration tools offerings. Before his current assignment, Curt was the chief architect for the z/OS DB2 development team.

Dawn Davis is a Staff Software Engineer in the z/OS security development organization in Poughkeepsie, New York. In his 7 years at IBM, he has worked in development organizations ranging from z/OS Core Components to Enterprise Workload Manager™ (EWLM) and IBM Systems Director. He is now in z/OS Development on the TKLM for z/OS Development team.

Anuja Deedwania is a Senior Technical Staff Member in z/OS Development at IBM Poughkeepsie. She has been with IBM for 26 years and leads the design and delivery of z/OS Simplification and Systems Management initiative.

Susan Z. Demkowicz is an Advisory Software Engineer for z/OS Enabling Technologies at IBM Poughkeepsie, New York. Her expertise in projects on behalf of the Simplification Initiative includes Common Event Adapter and Jobs Instrumentation. Susan holds a Master of Science degree from Boston University.

John Eells is a former MVS and OS/390 system programmer, more recently a ServerPac designer, now a Senior Software Engineer in the IBM Systems & Technology Group. John’s current assignments include z/OS Technical Marketing.

Larry England is a Senior Technical Staff Member at IBM’s Silicon Valley Lab in San Jose, California. He has architectural responsibilities for application development tools on System z. He holds a Masters Degree in Computer Science from Oregon State University.

Willie Favero is a Senior Certified IT Software Specialist with IBM Silicon Valley Lab, a zChampion, and part of the Academic Initiative for System z.

Nina Goradia is a Senior Software Engineer in z/OS Development at IBM Rochester. She is the development team lead for the Incident Log. Her focus has been User Interface development and z/OS Simplification.

Shayne Grant is a Software Engineer in z/OS Development at IBM. He has been with IBM for 11 years and is a team lead working on the z/OS Simplification and Systems Management initiative.

Michael Groetzner is an Advisory Specialist for z/OS Programming at IBM Germany. His expertise includes z/OS Capacity Provisioning. Michael holds a degree in Computer Science from the University of Cooperative Education in Stuttgart, Germany.

Paul S. Hagen is an IBM Certified Consulting Information Technology Specialist for the Washington Systems Center in Gaithersburg, Maryland. He is a member of the System z Hardware Advanced Technical Support team. Paul holds a BS degree in Electrical Engineering from the University of Cincinnati.
Tariq Hanif is an Advisory Software Engineer in the z/OS System Verification Test organization in Poughkeepsie, New York. His areas of focus are disaster recovery, GDPS®, and Parallel Sysplex. Tariq has been with IBM for five years.

Steven Hart, CISSP, is a Staff Software Engineer for z/OS Cryptographic Services in Poughkeepsie, New York. In his 6 years with IBM, he has worked on Trusted Key Entry (TKE), Encryption Facility, and TKLM for z/OS. Currently he is the TKLM for z/OS Service Team Lead and works on the Integrated Cryptographic Services Facility (ICSF) development team.

James Hertzig is an Advisory Software Engineer for z/OS Enabling Technologies at IBM Poughkeepsie, New York. James has a diverse z/OS background in development and service. He holds an MS in Computer Science from Marist College.

Paul Johnson is CICS System Management and Tools Architect at IBM Hursley. Paul has had lead roles in CICS system management, CICSplex System Manager, CICS Performance Manager, and CICS Configuration Manager. Twenty-five years ago Paul joined IBM after doing post-doctoral research in high energy physics and is a member of the CICS planning and strategy group, responsible for system management.

Steven Jones has been with IBM for 20 years, working in variety of BCP components, including GRS and XCF. He now leads the development teams for SMF, allocation, and scheduler.

Jason Katonica is an Advisory Software Engineer for z/OS security services in Poughkeepsie, New York. In his 10 years with IBM he has worked in all phases of software development including design, development, function test, and service on software including TKLM, System SSL, and various Java security components including IBMJCE, IBMJCECCA, and IBMJSSE.

Brian Keuling has been a software engineer on the Allocation Development team since joining IBM in 2007.

Steffen Koenig is a Software Developer SAP on System z Development at IBM Boeblingen Lab. His expertise includes Object Oriented Analyses and Design, Development in C++ and Java, J2EE Development, Web 2.0 Technologies, SAP on System z and IT Infrastructure Optimization and Automation. Steffen has been with IBM for 6 years.

Steven Ma is a User Experience Professional for Rational at IBM Silicon Valley Lab. He holds an MASc in Mechanical and Industrial Engineering from the University of Toronto. Steven is the usability lead for Rational Developer for z.

Charles Mari is a software engineer in the IBM Systems and Technology Group currently working in the Real Storage Memory Management area of z/OS Core Technologies.

Gregory E. McBride is a Senior Technical Staff Member in the Tivoli Storage Productivity Center for Replication organization. He is an expert in storage replication and is currently the lead architect for the TSPC-R. Greg has been with IBM for 14 years.

Renata Rand McFadden is a Senior Software Engineer in z/OS Development at IBM Poughkeepsie. She has been with IBM for 12 years. Renata is the Scrum Master for the z/OS Simplification and Systems Management initiative.

Barbara Neumann, an Advisory Software Engineer and a Managing Editor of z/OS Hot Topics, is the lead information developer for z/OS Language Environment®. She holds an MBA from the University of Rochester.

Al J. Noll is a software engineer for z/OS UNIX System Services development and test at IBM Poughkeepsie. Graduating from Northeastern University in 1984, Al has held various positions in security, information development, tools development and software function test.

Tatiana Oliver is a Staff Software Engineer for z/OS Enabling Technologies at IBM Poughkeepsie, New York. She has been with IBM for almost 10 years, working in Function Test on various z/OS components. Tatiana holds a Bachelor of Science degree from Rutgers State University.

Bonnie Ordoñez is a software programmer, doing service and development for the SMF and Scheduler components since joining IBM in 1998. Before that, she was a system programmer for Miami Dade County.

Jorge Padilla graduated from the University of Puerto Rico in 2004 and joined the IBM z/OS Communications Server group. He has worked in the Configuration Assistant for the past four years.

Kishor Patil (patil@ca.ibm.com) is a software developer in the TR-JIT Compiler team from the IBM Canada Lab.

Bob Petti is currently a test team leader in the z/OS Security area. He has worked on numerous security and middleware components, including ICSF, LDAP, SSL, and zFS.

Kathy Pfeiffer is a Senior Software Engineer at IBM Poughkeepsie. She is the Academic Initiative System z program manager.

Kelly Pushong is an IBM Certified Executive Project Manager in z/OS Strategy at IBM Poughkeepsie. Kelly joined IBM in 1988 and has experience in z/OS development, design, project management and strategy. She is the strategy manager responsible for z/OS Simplification and Systems Management initiative.

Ahlian Rajadeva works in z/OS UNIX Kernel and File System Development at IBM Poughkeepsie, New York.

Regine Reichert was a member of the z/OS HCD/HCM development team. She is now a member of z/OS Performance Management development team and has been with IBM since 1991 at the Boeblingen Lab.

William J. Rooney is a Senior Technical Staff Member for the System z Operating Systems Development organization at Poughkeepsie, New York. His expertise includes I/O configuration, I/O quality of service, and z/OS storage replication. Bill has been with IBM for 31 years.

Anthony Sofia is a Software Developer for z/OS System Management Facilities (SMF) at the Poughkeepsie, New York site and has been at IBM for 5 years.

Paula Spens is a software engineer in the IBM Systems and Technology Group currently working in the Real Storage Memory Management area of z/OS Core Technologies.
Levon Stepanian (levons@ca.ibm.com) is a software developer in the TR-JIT Compiler team from the IBM Canada Lab.

Theresa Tai is a senior software engineer at IBM System z Systems and Technology Group in Poughkeepsie, New York. Theresa’s focus areas are WebSphere Application server, Portal Server, IBM JVM and IBM specialty engines for new workloads on System z.

Ralf Thelen is a Test Leader and Quality Focal Point for Workload Management and Capacity Provisioning at IBM Boeblingen, Germany. His expertise includes Workload Manager, Capacity Provisioning algorithm verification and test automation. Ralf holds a computer science diploma from university Bonn and has been with IBM for 19 years.

Elpida Tzortzatos is a Senior Technical Staff Member in the z/OS Core Technologies department at IBM, where she leads the Memory Management design and development teams. She was instrumental in the successful delivery of several key enhancements in the memory management area such as 64-bit real and virtual support and large page support.

Girija Varanasi is a software programmer on the Allocation Development team at IBM Poughkeepsie. Since joining IBM in 2001, she has worked in msys for Setup, and the supervisor and allocation components.

Marna Walle is an IBM Senior Technical Staff Member in the z/OS Build and Installation department at IBM Poughkeepsie. Her current responsibilities include z/OS release migration and new installation technologies of the z/OS platform. Marna is a frequent speaker on z/OS installation topics at SHARE and other customer conferences.

Stephen Warren is an Advisory Software Designer and Developer for the BCPii component of z/OS in IBM Poughkeepsie. Steve has been with IBM for 20 years.

Martina Weidler is member of the z/OS HCD/HCM development team for almost 15 years. Before that, with a longer period for paid leave for working parents, she was engaged in VSE development.

Mark Wright is a Senior Software Engineer in z/OS Communications Server development at IBM Raleigh. He has 27 years at IBM and has worked on the Configuration Assistant for the past five years.

Corrections and clarifications

Nostra culpa! We’ve revised two of our articles from last issue March 2009 (GA22-7501-16). You can find the revised articles on our latest z/OS Hot Topics Web site at:

ibm.com/systems/z/os/zos/bkserv/hot_topics.html

• “New bells and whistles for STP” by Paul Hagen and George Ng
• “Stop spinning your storage wheels” by Jack A. Altman, Leslie F. Sutton, and Peter G. Sutton
**SIMPLY IRRESISTIBLE**

We want your opinions, ideas, and feedback! Customer partners participate in scenario walkthroughs for z/OS, help validate functional requirements, review, and provide feedback on early z/OS design prototypes, or provide input into IBM education. If you are new to z/OS or work in a small and medium-sized IT shop, we are particularly interested in talking to you.

Participation in the zSimplification Customer Partners Program is voluntary, and is scheduled at your convenience. Sessions are held using conference calls and web conferences.

If you want to help us simplify z/OS, send your name, company, and contact information to:

lbostian@us.ibm.com