The New IBM zEnterprise EC12
It’s All About Workload Optimization
If you haven’t seen CA Mainframe Chorus, you haven’t seen the graphical, intuitive workspace that management system experts call

“the most impressive mainframe innovation in decades.” -Jon Toigo, drunkendata.com

Mainframe. The mainstay of the enterprise.
If you haven’t seen CA Mainframe Chorus, you haven’t seen the graphical, intuitive workspace that management system experts call Mainframe. The mainstay of the enterprise.

–Jon Toigo, drunkendata.com
## Articles

<table>
<thead>
<tr>
<th>Page</th>
<th>Title</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Accessing Big Data in a Multi-Platform Environment</td>
<td>Wayne Kernochan</td>
</tr>
<tr>
<td>10</td>
<td>The New IBM zEnterprise EC12: It's All About Workload Optimization</td>
<td>Joe Clabby</td>
</tr>
<tr>
<td>16</td>
<td>Performance Troubleshooting Using the RMF Device Activity Report</td>
<td>Steve Guendert, Ph.D.</td>
</tr>
<tr>
<td>20</td>
<td>z/OSMF Simplifies System Management</td>
<td>Anuja Deedwaniya</td>
</tr>
<tr>
<td>28</td>
<td>Smart Mainframe Backup Strategies &amp; Recovery Processes</td>
<td>Rick Weaver</td>
</tr>
<tr>
<td>33</td>
<td>CICS Explorer for z/VSE Eases CICS Transaction Server Monitoring Challenges</td>
<td>August Madlener</td>
</tr>
<tr>
<td>38</td>
<td>CICS 101: CICS Resources and Tables</td>
<td>Phyllis Donofrio</td>
</tr>
<tr>
<td>40</td>
<td>Planning Your Upgrade to DB2 10 for z/OS</td>
<td>Willie Favero</td>
</tr>
<tr>
<td>46</td>
<td>Speeding Up Your Mainframe: Easier Said Than Done</td>
<td>Marcel Den Hartog</td>
</tr>
<tr>
<td>51</td>
<td>Rational Developer for z/OS: A New Paradigm for Mainframe Development</td>
<td>Prasad Ganti</td>
</tr>
<tr>
<td>54</td>
<td>System z Security Equals Vigilance</td>
<td>Karl Schmitz and Peter Spera</td>
</tr>
<tr>
<td>57</td>
<td>10 More Ways to Improve RACF Performance</td>
<td>Robert S. Hansel and Robert L. Whittle</td>
</tr>
<tr>
<td>60</td>
<td>AFP Generation From a Web Interface</td>
<td>Ambadas Choudhari and Balaji Krishna Murthy</td>
</tr>
</tbody>
</table>

## Columns

<table>
<thead>
<tr>
<th>Page</th>
<th>Title</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Publisher's Page</td>
<td>Bob Thomas</td>
</tr>
<tr>
<td>14</td>
<td>Vendor Watch: System z—The Platform That Just Keeps Evolving</td>
<td>Mark Lillycrop</td>
</tr>
<tr>
<td>27</td>
<td>IT Management: Are You Getting the Most Out of Your Mainframe Investment?</td>
<td>Marcel Den Hartog</td>
</tr>
<tr>
<td>32</td>
<td>Pete Clark on z/VSE: z/VSE Wish List</td>
<td>Pete Clark</td>
</tr>
<tr>
<td>37</td>
<td>Linux on System z: Upcoming Installer and Boot Changes for RHEL</td>
<td>David Boyes, Ph.D.</td>
</tr>
<tr>
<td>44</td>
<td>Technical Insights: Use Backup and Recovery Tools to Support Day-to-Day Operations</td>
<td>Rick Weaver</td>
</tr>
<tr>
<td>50</td>
<td>Data Perspectives: What Can You Do to Avoid DB2 Locking Problems?</td>
<td>Craig S. Mullins</td>
</tr>
<tr>
<td>56</td>
<td>Mainframe Security: Security for Tape Data Sets</td>
<td>Stu Henderson</td>
</tr>
<tr>
<td>64</td>
<td>IT Sense: Friction and Lubricity</td>
<td>Jon William Toigo</td>
</tr>
</tbody>
</table>

October/November 2012 • Volume 10/Number 5 • www.EnterpriseSystemsMedia.com
If you’re not getting more from less when it comes to delivering superb DB2 performance to business applications, maybe it’s time to relieve the pressure with the Responsive Systems Buffer Pool Tool® for DB2.

Our Buffer Pool Tool® for DB2 is a proven solution currently being used by many of the largest financial institutions in the world that suffered from the same DB2 performance problems. These Fortune 500 companies needed to speed the processing of critical business transactions accessing DB2 on the mainframe, increase throughput, improve user productivity, and delay processor upgrades as long as possible.

To date, we’ve given DB2 relief to hundreds of IT managers!

LET US GIVE YOU SOME DB2 RELIEF
Call us today to see how we can help you the way we have helped so many others. By taking a few minutes now, you could enjoy a pay-off in happy business users, improved throughput, and lower costs for years to come!

Call us at 732-972-1261 today to learn more, or email us at info@responsivesystems.com
zEnterprise EC12: The Most Powerful Mainframe Ever

Our cover story for this issue of Enterprise Tech Journal features the new zEnterprise EC12 (zEC12). Author Joe Clabby says that although the zEC12 features the fastest commercial processor in the industry, the real breakthrough is its increased workload optimization. He goes on to say, “By improving processing speed and capacity, IBM’s new zEC12 has become a fast, powerful, highly integrated, well-balanced computing environment designed to serve a wide variety of workloads, including transactional, batch, interactive, scientific, industry, and analytics applications.”

Embedded in IBM’s announcement of the zEnterprise EC12 were these facts, which I found particularly interesting:

- The 12 in zEC12 stands for the 12th generation (not sure what the 196 stood for in the z196). EC stands for Enterprise Class.
- The zEC12 is the result of an investment by IBM of more than $1 billion in research and development.
- The zEC12 was developed primarily in Poughkeepsie, NY, as well as in 17 other IBM labs around the world.
- The zEC12 has 25 percent more performance per core and 50 percent greater total system capacity than the z196.
- The zEC12 is the first high-end mainframe to be able to run without a raised data center floor.
- The zEC12 is the first general purpose server to use the same transactional memory technology used in IBM’s Blue Gene/Q-based “Sequoia,” the fastest supercomputer in the world.
- The zEC12 is the first IBM mainframe to include internal solid state technology with Flash Express, a new memory technology that improves the performance of data-intensive applications.
- The zEC12 supports heterogeneous platform requirements with the new zEnterprise BladeCenter Extension (zBX) Model 003 to run workloads running on AIX, POWER7, Linux on IBM System x, and Microsoft Windows on IBM System x servers.

The announcement of the zEC12 continues the IBM mainframe’s incredible legacy and should put to rest any discussion of its ultimate demise, along with the fact the mainframe and its associated software, storage, and services account for 40 percent of IBM’s profits.

I hope you enjoy this issue of Enterprise Tech Journal. ETJ
Your mainframe has a little spending problem.

Your mainframe wants more MIPS, more CPU and more real estate – in fact, anything it can get. It’s time to lower its credit limit. BMC mainframe solutions reduce costs and optimize resources, forcing your mainframe to actually earn its keep.

Learn more at bmc.com/mainframe
Accessing Big Data in a Multi-Platform Environment

For many organizations, the question of how to handle Big Data has become much more urgent over the last year. Big Data is of unprecedented size, is typically more unstructured, and is more likely to reside initially on public cloud server farms. It isn’t just a “hot topic”; it’s integral to the effective use of new analytics capabilities that are a high corporate priority.

These organizations typically don’t have monolithic platforms. Their data handling occurs on multiple platforms ranging from mainframes to many PC server grids and farms. Big Data solutions must layer on top of data handling solutions that integrate these platforms to varying degrees.

IBM DB2 10 offers a useful test case for handling Big Data access in a multi-platform environment. IBM DB2 runs in an integrated fashion on all major platform types, from mainframes to Linux and Windows small-server networks. IBM DB2 has recently undergone a major upgrade in version 10 and research shows the new version achieves impressive improvements in performance and scalability across major use cases—including Big Data. So how should we approach maximizing IBM DB2 10 for multi-platform Big Data usage?

**Index, buffer, and workload management.** This includes “jump scan,” which is better management of indexes in buffers that reduces the need for as many indexes—a capability several early adopters cited as a major performance booster. Another new feature is “smart” pre-fetching of indexes and data, reducing the need for index reorganizations. Also of particular interest is the ability in workload management to set limits on CPU and other resources that DB2 can consume. In the real world, this often allows more effective load balancing of resource consumption between DB2 apps and other apps in other virtual machines, leading to improved performance for both.

Big Data tends to be “index light” (much of the original Web data is stored as semi-flat files), so tuning these features specifically for Big Data probably won’t add much. However, tuning workload management in general pre-operation should deliver positive Big Data performance improvements on platforms that aren’t dedicated to a data warehouse or to a specific operational data handling task.

**Continuous data ingest.** This ability to “push the envelope” by extending the ability of the data warehouse to accept updated data in near-real-time has two major effects: Data is typically more timely and the need for data warehouse downtime to “mass download” huge amounts of updates is reduced.

Before deployment, data warehousing IT should “tune” IBM DB2 10 Big Data continuous data ingest like a thermostat, running experiments to find the optimal balance between performance on current tasks and near-
TRANSFORM
IT+BUSINESS+YOURSELF

Transform your mainframe environment with the solutions and training only EMC can deliver.
Visit us at SHARE San Francisco, February 3-8, 2013.
real-time delivery of actionable Big Data into the data warehouse. A key tuning consideration is the Extract, Transform, Load (ETL) engine because Big Data is less “pre-cleansed” than previous data warehousing data.

**Adaptive compression.** Most users understand the storage cost savings from compression; fewer understand the significant performance benefits. These benefits come not just because you can load more data into main memory at once, but also because things such as online backup and online reorg are much faster, reducing their performance overhead on regular transaction processing. (By the way, in IBM DB2, the data need not be decompressed during processing in main memory, speeding up things further.)

As the name suggests, adaptive compression is an extension of previous IBM DB2 compression techniques to compress tables, pages, and archive logs in a way that changes over time as the characteristics of these items change. IBM estimates that, on top of previous techniques, adaptive compression can compress uncompressed data down to perhaps 15 percent of its original size, or an additional 40 to 60 percent compared to the last IBM DB2 version.

Before implementing IBM DB2 10, IT may want to consider “training” adaptive compression on likely Big Data transaction mixes. This should give IT a good feel for the right ratio of main memory and disk storage to optimize Big Data processing. Initial indications are that most organizations considering Solid-State Disk (SSD) as part of the storage tier will find 8:1 to 9:1 ratios for both memory/SSD and SSD/hard disk to be optimal.

---

**Querying improvements.** IBM DB2 10 has performance improvements especially useful for specific types of Business Intelligence (BI) workloads, including star-schema, join/sort, aggregation, and hash join queries.

Big Data is index light and “schema light,” so IT professionals should focus pre-operation on tuning and optimizing join operations, which are more common in handling unstructured Big Data.

**Multi-temperature data management.** This colorfully named feature simply reflects into the database administration realm the increasing cost-effectiveness of Storage Area Networks (SANs). Within IBM DB2 10, you can distinguish between, say, SSDs, Fibre Channel (FC) disk, Serial Advanced Technology Attachment (SATA) disk, and archival tape, and achieve close to the performance of SSDs/FC disks at close to the cost of SATA disk/tape, while decreasing backup times by more than 60 percent.

This requires that users be able to distinguish between data that’s new and may be updated and data that’s old and is rarely or never changed. Hence, it’s especially useful for users that have already begun the journey into storage tiering. According to IBM, industries such as insurance and retail can cost-effectively keep customer policy/claim/sales data in “hot” storage for fast reaction with historical data on customers/sales in “cooler” storage for reporting and deeper analysis.

Many organizations will have similar storage-level tiering solutions. For these, however, the database is a bit of a “black box.” IT should aim during initial testing to coordinate storage and database tiering solutions, especially those involving flat-file storage (a typical storage mechanism of social media Big Data).

**Additional Tweaks**

Multi-platform users may also want to consider implementing solutions that integrate their platforms more completely, allowing better load-balancing of Big Data transactions. In the IBM case, both zEnterprise (mainframe integration with Linux/Windows blade servers) and PureFlex (integration of Windows/Linux server farms and blade/high-end UNIX/Linux servers) should deliver additional performance benefits.

**Final Thoughts**

These suggestions don’t capture the likelihood that if you upgrade to IBM DB2 10, a major amount of Big Data performance improvements will happen out of the box. One beta customer testified that average operational database app performance improvements in initial testing were 50 percent with the key Enterprise Resource Planning (ERP) app improving by 90 percent. These improvements were compared to the last previous 9.x version of IBM DB2.

For some, tuning IBM DB2 10 for Big Data may be “icing on the cake.” But it’s tasty icing. ETJ

---

Wayne Kernochan has covered the computer industry as an analyst for the last 20 years. He’s president of the analyst firm Infrastructure Associates, which focuses on infrastructure software, IT agility, data usefulness, TCO/ROI studies, competitive analysis, and mainframes.

Email: wkernochan@aol.com
Less complexity. Less setup. MORE INNOVATION.

Compuware Mainframe Solutions are intuitive, powerful and delivered through a modernized environment to help you minimize distractions while you develop, test, debug and tune your applications. So you can unleash your innovative potential.

This is the future of your mainframe.

Learn more at: compuware.com/mainframe
Approximately every two years, IBM introduces a new generation of its mainframe architecture—the System z—and August 2012 marked two years since IBM introduced its mainframe z196 and zEnterprise hybrid architectures. The new mainframe has been dubbed the zEnterprise EC12 (zEC12), and it’s clearly focused on running workloads faster than ever before while extending IBM mainframe strength in reliability, availability, and security.

From a strategic perspective, it’s clear the zEC12 is being positioned as a highly scalable, enterprise-strength, variable workload processing machine. The mainframe’s strength in the past has been the execution of large transaction and batch workloads, but with continued investments in Linux and Java on System z, the mainframe is now well-positioned to execute the most modern Java workloads. Meanwhile, IBM has also introduced new reliability, availability, and security improvements—extending the mainframe lead over other architectures in availability and security. In fact, the mainframe is the only commercially available enterprise server to have ever achieved the prestigious Evaluation Assurance Level (EAL) Level 5+ security rating. In addition to positioning the mainframe as an enterprise-strength workload processor, IBM also announced improvements to its zEnterprise hybrid computing configuration; this configuration positions the mainframe as a central governance node that controls and optimizes workloads on attached POWER- and x86-based blades.

From a technology perspective, the zEC12 features the fastest commercial processor in the industry—the z processor running at 5.5 GHz. To speed the feed of data to this processor, IBM has expanded Level 2 cache by 33 percent, and has doubled the cache in Levels 3 and 4 from 24MB to 48MB, and from 192MB to 384MB. Both actions are designed to enable the zEC12 to process work more quickly; the first place processors look for data is in cache memory, so by increasing cache, IBM is increasing the rate at which data flows to the processor. To further improve processor performance, IBM has honed its approach to processing with second-generation, out-of-order design; this provides a way to keep the microprocessor busy by executing jobs when they become available, even if they’re out of order. Further, IBM has added multi-level branch prediction, which provides a means to speed processing using a predictive method.

But improving processor speed and increasing cache represent only two of the steps IBM has taken to increase the speed at which workloads can be executed. IBM has also added new instructions to the z processor, and has built new facilities that speed workload execution. Using these new instructions, applications can make calls to the
processors that expedite the speed at which an application is processed. Facilities, or groups of instructions, further accelerate application processing speed; for instance, by using a new run-time instrumentation facility to process Java workloads, the zEC12 can increase Java workload performance by up to 45 percent on the mainframe, and new DB2 facilities enable DB2 database workload speed to be increased by 30 percent (see Figure 1). Plus, IBM has added a new transactional execution facility that improves parallelism and scalability.

IBM’s workload tuning focus isn’t solely focused on Java and Linux. With the new zEC12, IBM makes it possible to run analytics and Online Transaction Processing (OLTP) side-by-side on System z. This means that a zEC12 and IBM DB2 Analytics Accelerator (IDAA) can share the same data between operational applications and analytics applications. This enables data warehouses, analytics, and OLTP to all run as the same workload in a seamless, real-time environment instead of on siloed, separate databases. This can be seen as a major advantage because it provides a single entry point into a workload-optimized system that combines optimal tuning for both OLTP and data warehousing. It reduces the need to separate these two workloads into independently managed operational environments. Some IBM customers even claim that by using this solution, they’ve been able to develop new insights that were previously unattainable, given that data was held in data silos.

A Closer Look at the zEC12

When evaluating a new mainframe, many IT buyers look first at the processor, then at memory, internal throughput (the speed at which data moves within the mainframe chassis), and I/O (Input/Output) bandwidth (the speed at which data flows to and from external devices such as storage).

While processing speed and cache memory have increased, the amount of main memory the zEC12 can exploit has remained the same as the predecessor z196 at 3TB of Redundant Array of Independent Memory (RAIM). According to Kelly Ryan, director and business line executive for System z Platforms, memory wasn’t expanded in the EC12. “We looked at our customers’ workloads and memory requirements, and given how efficiently System z uses memory, we chose to focus on workload execution speed and tuning rather than adding more memory.” She also noted that the preceding generation z196 “doubled memory size from its predecessor while also improving the reliability of memory with RAIM.”
“We did, however, add a new tier of memory that improves the availability of some application workloads” added Greg Lotko, vice president and business line executive for System z. “With our new Flash Express Solid State Drive (SSD) product, we can now use SSDs as a form of memory to help our systems transition more quickly between modes of operation. For instance, a financial institution may run a batch mode at night, and then need to transition to an interactive mode when the doors open for business in the morning. Flash Express makes our system resources available more quickly, so that the System z can tolerate paging spikes or inconsistent performance during a transition period.”

As for internal throughput—a measure of how much work a System z can process internally as measured in MIPS—the maximum System z MIPS rating in a single footprint has changed from 52286 to 78426—meaning the zEC12 can now do 50 percent more work than its predecessor. It should also be noted that the cost for MIPS has decreased for traditional workloads, while the cost for MIPS on IBM’s specialty engines (Integrated Facility for Linux [IFL], System z Integrated Information Processor [zIIP], and System z Application Assist Processor [zAAP]) has dropped by 20 percent. This is seen by several IT analysts as an attempt by IBM to expand its System z application portfolio by capturing next-generation Java/Linux-based workloads on System z.

From a system I/O bandwidth perspective, I/O bandwidth remains the same at 288 GBps—still the fastest internal throughput rate in the industry.

**Summary**

By focusing on workload performance and reliability, availability and security, IBM is positioning the System z to compete head-to-head with other systems architectures—such as Oracle’s UltraSPARC, Hewlett-Packard’s Itanium, and a host of x86-based servers—while differentiating the System z in the areas of availability, security, manageability, and scale. This strategy appears to be working for IBM, as the company now claims it has seen:

- Seven straight quarters of a positive two-year Cumulative Aggregative Growth Rate (CAGR)
- More than 140 new mainframe accounts since third quarter 2010 when the zEnterprise was launched
- More than 66 percent of the top-100 System z clients run Linux on System z, and more than 33 percent of all System z customers comprise more than 20 percent of all System z MIPS installed on IFLs (System z microprocessors with modified microcode that enable thousands of Linux instances to be launched and managed within a System z environment). IFL growth is important to the mainframe’s future because it enables today’s modern, Java-based applications to run on the mainframe and the mainframe to be positioned as a very large, very efficient Linux consolidation server.

In addition to tuning the zEC12 to execute workloads faster, IBM reasserted its commitment to its zEnterprise systems environment—an environment that makes the mainframe the central governance node for attached blades. To wit, IBM also announced it has made improvements to its zEnterprise BladeCenter Extension (zBX) blade chassis environment (the new zBX Model 003).

By improving processing speed and capacity, IBM’s new zEnterprise EC12 has become a fast, powerful, highly integrated, well-balanced computing environment designed to serve a wide variety of workloads, including transactional, batch, interactive, scientific, industry, and analytics applications. The new EC12 showcases all the major System z differentiators: powerful processing, advanced reliability, availability and security, simplified manageability, efficiency at scale—and most important—the ability to optimally execute a wide range of workloads across traditional System z and hybrid computing environments.

Joe Clabby is noted for his research/analysis and public speaking abilities; he has written numerous specialized analytical reports on computer technology vendors and has spoken around the world on evolving computing trends. He has been in the computing industry for more than 30 years in sales, product marketing, and research and analysis. He is president of Clabby Analytics, and was formerly vice president of Systems and Storage at Summit Strategies, as well as group vice president of Platforms and Services at Boston-based Aberdeen Group.

Email: jclabby@aol.com
If excessive software upgrade charges and maintenance fees are ruining your bottom line, take advantage of these MacKinney Systems’ printing products. **NO upgrade charges, ever!**

**JES Queue for Printers (JQP)**, Annual license - $6,500 (1-100 printers)
Eliminates BUS/TAG and SNA printers by utilizing your existing TCP/IP network. Prints to VTAM or TCP/IP printers and replaces LRS/VPS and similar TCP/IP or VTAM printing software.

**VTAM Virtual Printer (VVP)**, Annual license - $2,995
Capture the 3270-based data stream sent to a VTAM printer from any VTAM application (CICS, IMS, etc.) Route output created by your online applications to the system spool. Accept reports from other operating systems (UNIX, LINUS, Windows, etc.) via the LPD server. Replaces LRS/DRS.

**JES Report Broker (JRB)**, Annual license - $6,500
Automatically archive JES reports and view them quickly and securely with a web browser. Distribute reports via Email, FTP and the Web.

"We enjoy our JQP and JRB products a lot. We use the JRB product extensively. Users review their reports online or get copies of them via email. They also print parts of the reports at their local printers. Our productions jobs are set up to send emails at the end of the job with the max CC of the job. This helps everyone track our daily business processes and react accordingly. Users like the web piece of the online JRB report feature. JQP lets us print directly to printers throughout the building from the mainframe and print IP to our Solimar and IBM printers. Thanks and keep up the good work.”

Johny Mercer, Systems Programmer Sr., Idaho Department of Labor

See for yourself how MacKinney printing solutions work in your environment. Download a free trial today from www.mackinney.com, or call 417 882 8012 for a free analysis.
Looking back at the changes—in terms of personal communication, information sharing, online commerce and transaction management—that have taken place in the way we do business over the last five decades, it seems inconceivable that one IT platform could have evolved to support the most critical business applications throughout the whole period. Yet the mainframe, and the System/360 architecture that still sits at its heart, has evolved continually to ensure the world’s banks, retailers, manufacturing companies, and utilities have sufficient compute power and capacity available to support and exploit the most exacting customer requirements. Such is the flexibility of the architecture that—despite an astronomic growth in data sharing and storage requirements and a completely new “cloud-based” paradigm that was unknown just a few years ago—System z has continued to keep pace and retain its position as the data server of choice for both new and established applications.

Recently, IBM announced the zEnterprise EC12 (zEC12), its latest mainframe processor family, details of which are covered elsewhere in this issue. As with every top-end announcement, some of the statistics are pretty mind-blowing, such as the 120 microprocessors running at 5.5GHz, capable of executing more than 75,000 MIPS. The fact that the zEC12 delivers 25 percent more performance and 50 percent more capacity with the same energy footprint (and more or less the same price point) as its predecessor is also a phenomenal feat.

But the industry has moved beyond trading feeds and speeds, and the more interesting announcements concern the technologies that point the way forward for the System z both as a native system and a hybrid host for heterogeneous applications. These include numerous tools to improve data security, such as the Crypto Express4S co-processor, which brings the system up-to-date with the most exacting international standards, and transaction memory technology, which has been borrowed from the Sequoia supercomputer to improve the way the mainframe handles concurrent applications in memory.

The new zAware feature is another interesting development. Described as “taking analytics and applying it inward,” it’s essentially a highly intuitive monitoring tool for tracking, analyzing, and correcting anomalies in system behavior, a first step (says IBM) toward policy-driven automatic recovery. The mainframe has always led the way in automated operations and recovery, and this tool promises to take that leadership one step further.

Clearly, cloud computing—in all its internal and external forms—raises many issues for IT service managers in areas such as capacity and performance management and security. With the System z’s hybrid capabilities (about which more will be announced later this year), and its indisputable ability to handle the most complex workloads, there can be few cloud-related challenges that are beyond the capabilities of the mainframe.

The main market for the zEC12 will, of course, be among IBM’s largest and most mature users—and the signs are that take-up of the initial shipments will be fairly brisk. As with previous generations, we can expect to see a BC12 announcement in a few months’ time, aimed at the general-purpose market where competition with other platforms is more intense. Much of the growth in the mainframe market—particularly in Asia—is taking place at the Business Class (BC) level, but profit margins here are much slimmer for IBM, and the company will be aiming to make the most of lucrative Enterprise Class (EC) sales before rolling out its smaller models. Nevertheless, if the System z really is to capture its rightful share of new workloads, rather than relying on traditional MIPS, it’s in the BC space that this will happen.

It’s also worth sparing a thought for the independent software vendors in the System z space. Those that have specialized in providing tools for the established market have struggled in recent times, as IBM itself and the larger software companies have boosted their own revenues by knocking out or absorbing the smaller players. But as the System z moves forward into the complexities of cloud computing, supporting UNIX and Windows applications as well as z/OS, VM and Linux, there will be many opportunities for companies with the right knowledge and skills to deliver a new generation of third-party tools.

Mark Lillycrop is CEO of Arcati Research, a U.K.-based analyst company focusing on enterprise data center management issues. He is publisher of the Mainframe Market Information Service and the Mainframe Yearbook.

Email: mainframe@arcati.com; Website: www.arcati.com
We reduced our annual costs, got the same functionality as NetView®, plus an IP monitor we’d wanted for years

SYSTEM PROGRAMMER
MID-WESTERN INSURANCE COMPANY

If you’re using NetView® or NetMaster® for z/OS to control your network you’re probably spending twice as much as you need to.

Using a bulky, expensive solution to monitor your mainframe isn’t just hard on your technicians: it’s hard on your budget. Why pay for what you don’t need? VNAC (VitalSigns™ for Network Automation and Control) is a lean mainframe monitor that gets the job done at a fraction of the cost.

With VNAC you can slash your network monitoring budget in half and get an IP monitor with diagnostic tools free.

Save 50% or more on annual costs. Shops of 300 to 30,000 MIPS report average annual cost savings of 50% when they migrate from NetView or NetMaster to VNAC.

Save big on costly MIPS. Our users confirm that switching to VNAC reduces CPU utilization dramatically, helping them save on operating costs and delaying costly upgrades.

Get an IP monitor at no cost. VNAC comes with a free VitalSigns IP network monitor, giving you the total coverage you need.

Plus: • Flexible choices for implementation and deployment. • User-friendly interface. • Simple panel-driven 2 hour installation and fully supported migration.

In today’s tough economy, no enterprise can afford to keep using software that’s inordinately expensive to run. If you’re a NetView or NetMaster user, it’s time to take a closer look at what VNAC can do for your monitoring budget.

Call us today for your free 60 day trial or personalized webcast
Phone: 763.571.9000 | Web: www.sdsusa.com
The article “The Performance Information Gap Paradox” (June/July Enterprise Tech Journal available at http://entsys.me/cj18t) discussed the challenges of leveraging recent System z I/O technology enhancements without a solid grasp of how to optimize them and, more broadly, our diminished understanding of performance management in general. This article will continue that discussion and assume there was an application Service Level Agreement (SLA)/Service Level Objective (SLO) for transaction response time that wasn’t being met.

Was this caused by something in the I/O environment? This article will go through one of the key RMF reports used in mainframe I/O performance management/troubleshooting: SMF 74-1, the RMF Direct Access Device Activity report. This report contains response time information and information on the various components of response time. It can be used to further narrow down what may be the root cause of the problem and provide a good idea of what other RMF reports we should check. Figure 1 summarizes some of these reports and which components they’re used with.

Figure 2 shows an example of a Direct Access Device Activity report. This report shows little to no activity, but our purpose here is to discuss the various fields, their meaning, and what they measure.
Luminex Channel Gateway X (CGX) is the leading disk-based virtual tape storage solution for mainframe data centers of all sizes. Ease the transition to virtual tape with Luminex Tape Migration services, then experience how CGX’s speed, simplicity and reliability improves all aspects of mainframe tape operations.

With no minimum or maximum capacity restrictions, connectivity options from ESCON through 8 Gb FICON and the ability to use existing and next-generation storage systems, CGX is the Mainframe Virtual Tape solution that relieves the pain caused by physical tape in every mainframe data center.

Get help now. You deserve it.

WARNING: CGX will cause euphoria, enhanced mainframe tape performance, and a desire to perform frequent disaster recovery testing. Users are encouraged to call a colleague and discuss their symptoms.

www.luminex.com/cgx
The purpose of this report is to give information on I/O device use for all online devices you requested. This can be done by device number, device class, or volume serial number (VOLSER). The report helps with analysis of device performance, identification of bottlenecks caused by particular devices, and in overcoming inefficient resource use.

As CPU speeds increase, the I/O response time becomes the determining factor in the average transaction response time. I/O response time consists of two components: I/O service time and I/O wait time. We can use the following formula to illustrate this:

\[ I/O \text{ Tr} = I/O \text{ Ts} + I/O \text{ Tw} \]

The average I/O response time is one of the fields contained in Figure 2. We can get better response time by decreasing either I/O wait time (T_w) or I/O service time (T_s). Figure 3 further defines the components of I/O response time.

Consider some of the fields that compose the “meat” of the report and a brief explanation of each:

- **Storage group** identifies the storage group to which the device belongs. This is reported only when the specified volumes are members of a storage group.
- **DEVNUM** is the device number (hexadecimal) used to identify a specific physical I/O device.
- **DEVICE TYPE** is the defined type of device.
- **VOLUME SERIAL** is the volume serial number of the volume mounted on the device at the end of the reporting interval.
- **PAV** is the number of parallel access volumes (base and alias) available at the end of the reporting interval. If this number changed during the reporting interval, it will be followed by an asterisk (*). If the device is a HyperPAV base device, the number is followed by an H. In Figure 2, all these devices are HyperPAV base devices. In the case of HyperPAVs, the value is the average number of HyperPAV volumes (base and alias) in the interval defined by the equation in Figure 4.
- **LCU** is the hexadecimal number that identifies the Logical Control Unit (LCU) to which the device belongs. If this field is blank, it may be because this is a non-dedicated device in a z/VM guest system environment.
- **DEVICE ACTIVITY RATE** is the rate at which start subchannel (SSCH) instructions to the device successfully completed, defined by the equation in Figure 5.
- **AVG RESP TIME** is the average number of milliseconds the device required to complete an I/O request. It’s a reflection of the total hardware service time and front-end software queuing time involved for the average I/O request to the device (IOSQ).
- **AVG IOSQ TIME** is the average number of milliseconds an I/O request must wait on an IOS queue before a SSCH instruction may be issued. IOSQ is drastically reduced by implementing dynamic or Hyper PAVs.
- **AVG CMR DELAY** is the average number of milliseconds that a successfully initiated start or resume function needs until the device indicates acceptance of the first command. It’s a measure of how long the channel waits for the controller at the start of an I/O operation; for example, it indicates how busy the controller is.
- **AVG DB DELAY** is the average number of milliseconds of delay experienced by I/O requests to this device because the device was busy. Device busy can mean one of several things: The volume may be in use or reserved by another system, a head of string busy condition has occurred, or some combination of these events has occurred.
- **AVG PEND TIME**. PEND refers to the number of milliseconds an I/O request is queued in the channel. PEND is a reflection of the time between the acceptance of the SSCH function at the subchannel (known as SSCH-function pending) and acceptance of the first command associated with the SSCH at the device. It’s inclusive of the time waiting for an available channel path and control unit, as well as any delays associated with shared DASD contention.
- **AVG DISC TIME**. Disconnect time is a reflection of the time when a device was in use, but wasn't transferring data. DISC is the average number of milliseconds the device was disconnected (not transferring data).
while processing an SSCH instruction. With the FICON protocol, there's really no disconnect time concept like there was in ESCON. However, an equivalent figure is accounted for by the Channel Measurement Facility.
Reasons for disconnect time include:

- Synchronous remote copy (such as Peer to Peer Remote Copy [PPRC])
- Read cache miss
- Sequential write hits faster than the controller can accept in cache
- Control Unit (CU) busy
- Multiple allegiance or PAV write extent conflicts
- Reconnect miss (ESCON).

AVG CONN TIME is the average number of milliseconds the device was connected to a channel path and actually transferring data between the device and central storage. It's a measure of data transfer time as well as the search time and the time needed to maintain channel path, control unit, and device connection.

% DEV UTIL is the percentage of time during the interval when the device was in use. It includes both time when the device was involved in I/O operations and when it was reserved but not involved in an I/O operation.

Using the Report

If application/transaction response time is the problem, and an SLA/SLO isn't being met, this report will help you identify the specific device(s) involved in your performance problem. Once you've identified these specific devices with the abnormally high average response times, it's time to further drill down and determine what specific component(s) of average response time (IOSQ, PEND, CONN, DISC) are abnormally high.

Once you've identified these components of response time, it becomes possible to start further analysis to determine what's causing that component of response time to be abnormally high, and what other records/reports you should be looking at to determine the root cause of the performance problem. In other words, you can start to determine what pieces of hardware are being "problem children." Then you can determine what adjustments/tuning you need to perform to solve the problem. Let's briefly discuss some examples:

- High IOSQ time is typically solved by implementing dynamic PAVs or HyperPAV's.
- High disconnect time probably warrants further investigation using the Cache Activity report (SMF 74-5) or the ESS Link Statistics report (SMF 74-8).
- High connect time and high pending time probably warrant further investigation using the I/O Queuing Activity report (SMF 78), Channel Path Activity report (SMF 73) and/or the FICON Director Activity report (SMF 74-7).

You can start to determine what pieces of hardware are being “problem children.” Then you can determine what tuning you need to perform.

Conclusion

The Device Activity report is the foundational tool for I/O-related performance problem root cause analysis/determination and subsequent troubleshooting. It’s essential to know what information is contained in this report, how the information is obtained and calculated, and how it will point you further along in your analysis. This knowledge is highly beneficial to any personnel in your mainframe and mainframe storage organizations.

Future articles will discuss analyzing, investigating, and troubleshooting high connect time and high disconnect time using the appropriate records and reports. ETJ

Dr. Steve Guendert is a principal engineer and solutions architect for a networking vendor. A senior member of the Institute of Electrical and Electronics Engineers (IEEE) and the Association for Computing Machinery (ACM), he also serves on the board of directors for the Computer Measurement Group (CMG). Email: sguendert@ieee.org; Twitter: @DrSteveGuendert
Z/OSMF SIMPLIFIES System Management

By Anuja Deedwaniya

The IBM z/OS Management Facility (z/OSMF) is quickly gaining acceptance globally across all industries by companies of all sizes; now in its third release, it delivers on IBM’s mainframe simplification strategy, easing system management tasks for new and experienced systems programmers. The latest version of z/OSMF, which comes with z/OS V1R13, offers significant enhancements and functions (see Figure 1).

Software Deployment

The Software Deployment function lets you clone any System Modification Program/Extended (SMP/E)-installed software (including additional non-SMP/E files or data sets) whether it’s an IBM product or not. You can clone the software within the system or sysplex with shared DASD (also called local deployment) or across the sysplexes in an enterprise (also referred to as remote deployment). This capability reduces the complexity of software cloning.

With the z/OSMF Software Deployment task, IBM has provided a codified solution based on best practices that will bring rigor and discipline to the practice of rolling out software on z/OS while also making it easier. It will ensure that all artifacts of the software instance are copied instead of just selected parts; this can reduce problems in the future.

You can adopt this new function gradually, starting with mainframe products that lack appropriate rigor, then move on to service levels, the whole system, and the operating system. This will help your site avoid having to continuously manage the software deployment process.

The Software Deployment function provides a checklist approach with embedded wizards to guide you through the process, starting with selecting or defining the software
Automated Capacity Management for z/OS Batch

**Manage Your Capacity**

ThruPut Manager AE automates z/OS batch management to make optimal use of existing capacity.

**Do More with Less**

With the efficiencies of ThruPut Manager AE you can do more with less and reduce MIPS or defer CPU upgrades, all with commensurate software savings.

**Save Monthly Software Costs**

ThruPut Manager AE monitors capacity utilization (the 4-hour rolling average) to further save software costs when sub-capacity pricing is used.

It automatically constrains low importance batch as a peak approaches and resumes normal service as it passes. This results in significant monthly software savings with no impact to online service or mission-critical batch.

**The Leader in End-to-End Batch Automation**

ThruPut Manager AE is a radical leap forward in datacenter automation technology. It simplifies the datacenter environment and enhances batch service to users, while delivering year-on-year datacenter savings.

Simplicity • Service • Savings
to be deployed (see Figure 2). The software to be deployed, called a software instance, may contain one or more products based on what you want to manage and deploy as a single entity.

The deployment process allows for identifying missing required PTF service on instances that will share resources with the deployed software in the sysplex and also across sysplexes. Without the z/OSMF Software Deployment function, there’s no method to identify missing required PTF service across different sysplexes in a single action. You must manually generate and compare the software service reports on each sysplex to determine what’s missing. Furthermore, by ensuring all required PTFs are installed, future problems, including a possible system outage, can be avoided. z/OSMF Software Deployment can identify fixes and functions on a prior software level that will be regressed after a deployment operation. It can also identify required HOLD actions that must be reviewed prior to a deployment operation.

**Generated reports.** The reports generated at the conclusion of the checklist summarize all the planned updates on the target system before any actual updates are made to the target system. The reports can be saved for a later audit or used for problem determination if needed.

**Granular authorization.** A key software management enhancement in z/OSMF 1.13 with APAR PM40764 is Granular Authorization, which lets security administrators control users’ access to certain functions or objects by defining specific System Authorization Facility (SAF) profiles. For example, a DB2 administrator can be authorized to add and modify software instances and deployments for DB2 software only, and a z/OS administrator may be allowed to only work with z/OS and not DB2. The default is that all authorized z/OSMF users will have access to all software instances and deployments.
Web Access to ISPF
The new release of z/OSMF delivers a browser-based interface to the traditional ISPF under the z/OS classic interface category. It’s similar to the ISPF that z/OS systems programmers know and love; the changes provide access to ISPF from a browser anytime, from any place, with no 3270 emulator required. Also, there’s some color instead of the “green screen,” and fields can be coded to be mouse-clickable. A unique new feature is the way the split-screen function has been implemented. z/OSMF ISPF allows horizontal and vertical splits, so you can have up to four panes that can be individually sized and have a maximum of eight ISPF tabs open simultaneously (see Figure 3). It also supports up to 10 concurrent login sessions by the same userid from different browser sessions and a Time Sharing Option (TSO) session. Of course, this requires enabling ISPF profile sharing for the enterprise (see the z/OSMF Configuration Guide for details).

An overwhelming value of enabling ISPF for the Web is that it allows for easy exploitation of the new z/OSMF application linking function. This lets you launch an ISPF application from any other Web application, which can be another z/OSMF task or another IBM or non-IBM application.

Application Linking
The Application Linking capability supports linking and launching, with or without context, from a z/OSMF task or an external Web application (IBM or non-IBM) to another z/OSMF task or external Web application. IBM has provided Representational State Transfer Application Program Interfaces (RESTful APIs) to enable application linking and a Graphical User Interface (GUI) in z/OSMF under the administration category. This lets you define events as well as handlers for the events. z/OSMF has exploited this function in the current z/OSMF 1.13 release in its Incident Log task under the Problem Determination category.

The Incident Log
With a single mouse click, the Incident Log provides a consolidated view of all the system abnormal ending (abend) problems across the sysplex, for IBM or non-IBM products and components that produce such diagnostic data (see Figure 4). The system will also capture and manage log snapshots as first failure data capture at time of error and logically attach this to the incident. Users don’t need to manually search and extract log data for further diagnostics.

The Incident Log task lets you send diagnostic data using a wizard that reduces the overall time required from about 30 minutes per problem to about 30 seconds, and requires no deep system skills. Furthermore, you can attach and send additional diagnostic data that the service team may require. You can send data easily for any product on your z/OS system.

Some organizations have started using the Incident Log as their main interface to File Transfer Protocol (FTP) data since the wizard will create all the required Job Control Language (JCL) as well as compress/terse the data as needed.
Multiple users can log into the same z/OSMF instance and look at the same information simultaneously. So, when a problem does occur, it's ideal for triage, especially for any problems that span components or products. Without these new features, users must look at logs, find the dumps, and apply Interactive Program Control System (IPCS) skills to access the dumps. There's also no easy way to get one view of all the problems across the sysplex with great filtering and sorting options.

The Incident Log task provides details about an incident at a glance, such as the product name, component id, problem symptom string, date/time of error, system and release, etc. With a single click, you can launch directly into ISPF browse within the log snapshot data set. After sending the diagnostic data, you can view the status of FTP jobs by launching directly into the System Display and Search Facility (SDSF) or an equivalent product, after a one-time setup of the handler via the application linking manager. This minimizes the number of steps you must perform each time and provides an overall seamless experience.

**Workload Manager**

Workload Manager offers an easy-to-use interface to manage workload management policies for the sysplex that supports creating, editing, printing, installing, and activating policies. Best practice checks are built in. Recent enhancements include granular authorization for the different actions that can be performed with a policy (i.e., view, edit, update, and install).

With this feature, a broader group of people, including managers, can view and even understand policies and how workload is being managed, while only a subset can actually
update, install, or activate it. Workload Manager is easy to navigate and offers table-driven views and built-in best practices; it can significantly reduce the overall time needed to work with policies (i.e., minutes vs. hours).

The process to optimize the service definition based on best practices using the existing ISPF application used to be cumbersome and time-consuming. In contrast, when using the z/OSMF function, all you have to do is check the best-practice hints the GUI displays for policy elements, and modify the policy elements as needed. With the robust filtering and sorting capabilities, this takes minutes, or at most, a few hours, when done initially. Customized views can be saved for each user. The print feature allows for filtered previews and report-like outputs; the export/import functions support data sets as well as workstation files. You also can access a history of every action performed on the service definitions.

As the team transitions to using the z/OSMF WLM function, there may be a time when the ISPF and z/OSMF application for Workload Manager are both being used in parallel, and that's fine. You just need to decide whether the master service definitions should reside in the host sequential data sets, or the z/OSMF repository. Then the appropriate exports/imports can occur with proper naming conventions. The application will manage all the serialization and prevent any mistakes or overlays.

The software deployment function provides a checklist approach with embedded wizards to guide you through the process.
Resource Monitoring

Customers who use IBM’s Resource Monitoring Facility (RMF), especially the RMF Monitor III function, can configure and use the resource monitoring tasks in z/OSMF and instantly see the impact of the Workload Manager policy in effect. From one view in the System Status task, they can view the overall performance index and see whether the important workloads are meeting their goals. While some metrics and dashboards have been pre-defined for z/OS, it’s easy to define additional dashboards and even mix Linux, AIX, and z/OS metrics in one view. When additional dashboards are defined, these can only be viewed and accessed by the user who has defined it. If the user wants to share the dashboard among the team, they can open it in a new browser tab and then save the URL for the dashboard as a link in the z/OSMF navigation, giving it the right authorization for the role and users who can access it. With z/OSMF designed for role-based authorization that extends even to individual links, this is a useful function.

Network Configuration

z/OSMF has always provided network configuration functions. The Configuration Assistant is a GUI application for setting TCP/IP policy-based networking functions for some of the technologies for z/OS Communication Server policy agents such as IPSEC, AT-TLS, etc. Today, handling common network configuration process tasks can take many hours or even days for initial setup. With Configuration Assistant for Communication Server in z/OSMF, everything is simpler. You need only go to the IP security perspective, add a connectivity rule for an IP filter, and use application setup tasks to assist with the configuration and setup of the required applications. The Configuration Assistant will help you deploy the configuration files to your z/OS system in about 30 minutes.

z/OS Jobs

In addition to the RESTful APIs for application linking, z/OSMF is also hosting new RESTful APIs for managing jobs on z/OS. This feature also supports the batch modernization initiative for the platform. Instead of relying on FTP transfers, this new function uses a Web 2.0 RESTful interface to submit jobs, get job output or status, or cancel jobs.

Security

z/OSMF supports and builds upon the z/OS security with role-based, application-level authorization support. Besides supporting granular authorization in individual tasks, z/OSMF also added support for defining custom roles in this latest release via the SAF mode support.

Conclusion

z/OSMF has a broad set of functions to help the z/OS systems programmer be more productive, ranging from problem data management to configuration and software management. It’s designed to reduce the learning curve for newer systems programmers and help make life easier for experienced systems programmers. The Web-based GUI interface makes it more user-friendly and there’s also excellent online help for every function and feature. For more information, visit the z/OSMF Website at http://www-03.ibm.com/systems/z/os/zos/zosmf/.

Anuja Deedwania is a senior technical staff member in the z/OS Development lab and a software architect for the z/OS Simplification and System Management initiative at IBM in Poughkeepsie, NY. She holds a bachelor’s degree in Electrical Engineering and a master’s degree in Computer Science. She has more than 29 years of experience at IBM in various areas of mainframe design and development and is responsible for the design and delivery of the IBM z/OS Management Facility. Email: anujad@us.ibm.com
Recently, I engaged in several discussions on various mainframe-related LinkedIn groups about the need to try and run with up-to-date software to help the company cut costs. The comments I received in these discussions were both interesting and bothersome.

First, it’s interesting to see that people agree with what I see as anecdotal evidence from my own experience. Most respondents state that much of the new functionality and enhancements in the latest releases of the software solutions they use will help them run their mainframes more efficiently. But at the same time, the two reasons for delaying these upgrades are bothersome:

- Perceived product quality. Not many of those commenting felt they wanted to be on the front line when it came to using the latest and greatest. Many had previously had bad experiences with quality issues that ultimately were expensive to solve.
- Many stated they simply didn’t have the time to install, test, and conduct a quality assurance inspection on those new releases. They lacked the human resources to do this properly and decided they would only migrate if they were forced to do so by their vendor.

The first reason—quality—is understandable, especially on the mainframe. We’re expected to run a 24x7 operation where downtime simply isn’t acceptable. Mainframers have an “if it isn’t broke, don’t fix it” mentality, but in my honest opinion, we’ve gone too far in this regard because of the second reason raised—lack of resources.

This is a real problem, one I’ve blogged about many times. Companies (and their managers) have relied on the fact the mainframe is simply always “on” for far too long. Outsourcing, staff reductions, and lower budgets for a platform that’s more complex than ever will have an effect sooner or later. Large system failures are more common than they were five to 10 years ago, and analysts are the first to state this is understandable because of the fact the technology is “ancient,” “more than 20 years old,” “has existed for a very long time,” etc. What they fail to see is that this isn’t a technology issue; it’s a management issue. The mainframe today is as reliable as it has been for decades and is based on technology that’s as modern as any other technology in IT.

If a 20,000 MIPS shop spends 50 to 60 percent of its general processing power running DB2, an out-of-the-box savings of 5 percent (on the low side) can be achieved by simply upgrading to DB2 10; it will save them 500 MIPS. In times where every MIPS counts, this means a lot. These savings alone (or delayed investments in a MIPS upgrade, if you will) will buy you the tools you need to automate more of the migration process, ensuring you will need fewer people to perform the upgrade. That’s time and money well spent, if you ask me.

The world has changed and some of the outages I spoke about have made people aware that something needs to happen. We must use this momentum to make our point: A mainframe needs TLC, and this can be achieved by running—and using—the most recent software possible, and making sure it’s operated by the right people who have the right tools to get the job done.

Marcel den Hartog is senior advisor, Mainframe, EMEA for CA Technologies. He’s a frequent speaker at both internal and external events such as Guide Share Europe (GSE) and IBM’s Technical University, where he talks about CA Technologies’ mainframe strategy, vision, and trends. Before joining CA Technologies in 1986, he worked as a programmer/systems analyst on VSE and MVS systems, starting with CICS DL1/IMS and later DB2. Email: Marcel.Hartog@ca.com

Are You Getting the Most Out of Your Mainframe Investment?

Mainframers have an “if it isn’t broke, don’t fix it” mentality, but in my honest opinion, we’ve gone too far in this regard...
Recovery processes are like “the elephant in the room.” Like ignoring the elephant, it’s often just too difficult to openly discuss how unplanned outages have, or could in the future, wreak havoc on business if the right backup and recovery procedures aren’t established and followed.

An unplanned outage can damage customer satisfaction, result in lost business, and hurt a company’s reputation. Here are some examples:

- An outage recently caused the Automated Teller Machine (ATM) network of a major bank to shut down for hours, and customers couldn’t withdraw money.
- A Web services company had a power outage that caused its company’s Websites to go down, resulting in a significant loss of business for the site’s customers.
- Another large company had an outage so severe it wiped out important files. The mirrored files weren’t backed up, so the company couldn’t recover key information. As a result, the employees had to manually re-enter about 18,000 jobs into a scheduling package based on documentation, institutional memory, and whatever they could find from prior runs to rebuild the schedule.

Fortunately, if you address recovery head-on and develop intelligent backup strategies and recovery processes, you can reduce or eliminate many unplanned outages. You can also
Balancing Improved tape conversion, migration and stacking with...

**FATSCOPY**

“FATSCOPY met all of our requirements moving from our IBM 3494 ATL to a new EMC VTS.”

“Migrated 15TB’s of tape data over a 6 week period of time and did not cause any downtime.”

“We migrated more than 1400 tapes (9840’s) to a VTS in less than a day.”

“FATSCOPY allowed us to quickly consolidate 200 MAGSTAR tapes to three 3592 tapes.”

Erasing Tapes > Mapping a Tape > Copying a Tape > Recovering Tapes with I/O Errors or Overwritten Tapes

**FATSCOPY**, a z/OS high-performance, easy-to-use, tape copy, stacking, migration and conversion utility that can automate the process of copying and recataloging tape-based data sets from one tape media to another, including Virtual Tape.

**FATSCOPY**’s flexible selection criteria allows you to select by data set name prefix, volume prefix (or ranges) and by catalog device type.

**ADDITIONAL FEATURES INCLUDE:**

- Checkpoint Restart
- Auditing Reports
- Simulation Reports
- Tape ERASURE
- Tape CERTIFICATION
- Tape RECOVERY


COMPETITIVELY PRICED, FLEXIBLE LONG & SHORT-TERM RENTALS AVAILABLE. For a No-Obligation FREE Trial, call: 973-890-7300, e-mail: sales@fdrinnovation.com, or visit: www.fdr.com/fatscopy
mitigate some of the risks associated with outages by leveraging automation.

Evaluate Planned and Unplanned Downtime

Before updating the recovery processes for your organization, it’s important to understand the impact of unplanned—as well as planned—downtime. Planned downtime includes various activities such as database maintenance (performing copies and reorgs) and schema changes. Planned downtime typically doesn’t result in a recovery event unless the reason for the downtime requires a back out (e.g., a structural change is abandoned). Recovery involves restoring the last image copy tape, then applying log data to it to get up to a valid recovery point.

In a well-run organization, the back-out event is planned but rarely executed. If you’re doing a system upgrade, you can prepare in case the system upgrade fails. Then you can perform a back out to where the upgrade started and mitigate the impact of the outage.

Unplanned downtime, by its nature, is a surprise and can be caused by hardware failures, software errors, user errors, or poor maintenance. Many failures occur after a hardware or software upgrade. The recovery process can depend on the cause of the outage. That’s why you need to perform detection and analysis to determine the cause and scope of the failure. If recovery is manual, the operator may not be familiar with the recovery options. A program failure could result in one record being impacted or multiple databases being corrupted.

How much downtime, whether planned or unplanned, is acceptable? Most IT organizations have a recovery Service Level Agreement (SLA) or Recovery Time Objective (RTO) measured in hours for unplanned outages—two to six hours is common, depending on the application and its impact on the business. How long your systems are down depends on what caused the outage and what’s required to fix it. During a major outage, one enterprise company had to keep support staff on the phone all night for two nights; certain applications were down the whole time. The company also discovered months later they had data integrity problems in their databases because of the way they performed the recovery.

Get Smart

Smart backup strategies can mean the difference between a slow or rapid recovery. They’re based on developing recovery SLAs. An RTO must be established; it can be different for certain databases or applications, depending on business needs.

Making copies consumes many CPU cycles. Some unplanned outages occur when copies are made offline. Many IT organizations still schedule copies based on how they were set up 20 years ago. A smart copy strategy would answer the question, “What is the recovery objective for this particular application or this particular database within this application?” Some databases are more important than others. If the RTO for a particular database is only 15 minutes, then look at the size of the object and the amount of log traffic that gets put on that object, and incorporate that information into your backup strategy. You may have to run backups several times a day. It’s important to reset your thinking on what a backup strategy requires.

Consider Performing Incremental Image Copies

It’s also smart to do the minimal backup required while maintaining recovery capabilities. This entails performing incremental image copying, in which you copy only the blocks or pages that have changed instead of the entire database. This approach can increase recovery speed. However, an incremental image copy reads the database less efficiently than a full copy. An incremental approach requires analysis to decide whether you need to copy a page or a block. Generally, if more than 10 percent of your database is changing, it’s faster and less expensive to take a full copy.

You may also decide to run full copies weekly and make incremental copies at other times. If enough data has changed, you can always revert to making full copies more frequently. Some tools can perform the function automatically and dynamically adjust backups based on business needs at various times to meet seasonal demands.

You can make copies while the database is online, which eliminates outages. As long as log data is applied to the online copy to a consistent recovery point, data integrity is protected. Doing online copies may make it more attractive to copy multiple times a day.

Make Copies to Disk or Tape

Many companies keep at least the most current copy on disk. A copy can be made to tape for archival or Disaster Recovery (DR) without impacting the disk copy. This action can reduce recovery time significantly by eliminating tape mount time and allowing for more parallel recoveries.

Smart recovery processes include ensuring that plenty of log data is available on disk. For DB2, this means having large “active” logs. For IMS, it means having large Online Log Data Sets (OLDS). These files should be sized so that any recovery event can retrieve all the log data from disk that has been created since the most recent image copy.

Disk-based copies and logs enable faster recovery, as well as parallel recoveries. If you’re doing an application recovery based on a certain point in time, you’re probably recovering dozens of databases and hundreds of indexes, and they’re all recovering to the same time period. Being able to perform recovery in parallel can reduce the overall outage for the recovery. This capability ensures that plenty of log data is available on the disk.

Following Best Practices

DR is a special-case recovery event. DR plans typically are tested with some frequency and processes are documented to reduce the recovery time. Local recoveries are typically caused by application program failures; however, local recovery is rarely practiced. Sometimes, an application program failure requires a recovery to a prior point in time. Smart recovery processes can identify all the application objects impacted by the application program failure and recover only those databases that require this
Smart backup strategies can mean the difference between a slow or rapid recovery.

The tool should let you exploit processor cache or intelligent storage devices.

You can improve recovery with technology in various other ways. Recoveries can be made more efficient by using tools that sort log data and merge it with copy input. This process can allow for back-out processing to a specified time. This period can be used to build coordinated recoveries between DB2, IMS, and VSAM applications, which accelerate the speed of recovery.

You can also make recoveries more efficient by identifying objects that have been changed since the specified recovery point and recover only those objects. Recovery can be eliminated if the application program failure impacts only certain records. If the updates were logged, the log records can be used to restore the data to its original state without requiring a recovery.

A flexible recovery process enables searching for or specifying a recovery point and ensuring data consistency after a recovery. The recovery solution could determine that only a subset of the database objects actually requires recovery, eliminating the downtime for unnecessary recoveries. Some applications have components running in several database management systems, such as IMS and DB2. A point-in-time recovery for one side of the application may require a point-in-time recovery for the other, too. A flexible recovery process allows for consistent coordinated recovery to any point in time for all the DB2, IMS, and VSAM components of an application.

A user may inadvertently update more data than intended. Only a subset of the data is affected, so a full recovery isn't required or even desired. A flexible recovery process would allow for a back out of only the affected data, leaving the rest of the application available for update.

Summary

Backups should be scheduled with the RTO in mind. The cost and impact of backups can be reduced or eliminated with the right tools. Innovative recovery techniques and automation reduce the impact of unplanned outages while ensuring data consistency after recovery.

DR is a special-case recovery. It's more likely, however, that an unplanned outage will affect the production database but not result in a declaration of disaster. Local recovery, therefore, should be a top priority for IT organizations.

Automation and smart processes can help reduce the impact of unplanned outages. The technology used should help you recover only what you need. Conduct a business impact analysis to identify strategic applications and databases. Focus on those databases and applications, determine what the recovery SLA should be, then develop the backup strategy to support it.

How Can Technology Help?

Technology can help reduce or eliminate unplanned downtime outages. For example, you can leverage technology to offload copies to System z Integrated Information Processors (zIIPs) to reduce the CPU cost of backups and make the image copying process more efficient.
W
ouldn’t it be great if we had a consolidated listing of all 
z/VSE products, similar to the list IBM makes available 
of open and closed APARs for CICS Transaction Server 
1.1.1 for z/VSE? Having this information for z/VSE and other 
products would be a tremendous asset. I personally think 
this is an excellent idea, has merit, and should be considered. 
I would carry it one step further to say the list should include 
all z/VSE products, both IBM and aftermarket.

Yes, I’m aware you can search the APAR databases of 
products and compile and maintain a list, and that you can 
repeat the search when needed and update the contents. 
Some of us already do this, updating the contents on a 
regular basis.

However, it appears this sort of list would be more 
accurate, effective, and efficient if it was compiled one or more 
times daily using an automated process rather than by hand.

While there might be some objections, I’m not sure they 
would override the value of having this type of list available. 
Additionally, it would replace some of the existing 
documentation that’s maintained by hand. A perfect 
example is the current PTF list in the installation buckets.

The only objection I might hear is, “But everyone will 
know how many problems we have.” Yes, but they can 
readily find out that information by doing some searches. 
With the quality of the products today being so good, I 
really don’t expect anyone to know or care. Having accurate, 
fast access to this type of information is way more important 
than a concern about the number of problems.

Not to get on my soapbox, but yes, we all like products 
with few problems, but we also like products from vendors 
that support their products and quickly repair issues found. 
It’s the combination of buggy products and no fixes (or 
unknown fixes) that cause users to abandon products.

What do you z/VSE users think? Email me at 
pclark@cprsystems.com or start a discussion thread 
on VSE-L.

News From Boeblingen and Hursley

IBM recommends that users of z/VSE 4.3 and 5.1 review, 
order, and install the PTF associated with APAR numbers 
z/VSE 4.3 DY47403 and z/VSE 5.1 DY47407 as soon as 
possible or before going into production. This will help you 
avoid significant VSAM performance issues.

The APAR is now marked Hiper, but wasn’t originally. 
Because it’s now labeled Hiper, the PTF will be included on 
the next service-level refresh. Additionally, the z/VSE 
Service and Development team is aware of the impact of the 
issue and is committed to getting the information and the 
PTF to z/VSE users.

It’s the combination of buggy 
products and no fixes 
(or unknown fixes) that cause 
users to abandon products.

The original APAR was opened to fix SVC 50 being 
called for a VSAM “illogical condition.” The first issue was 
VSAM not always being called correctly for logical record 
access and DL/I CI-level access in CICS.

The second issue is poor performance on compressed 
files, which could affect CICS and batch. Many users 
encountering this will perceive this as a z/VSE, CICS, and/or 
VSAM poor performance issue.

This APAR has been the source of some significant 
discussions concerning impact, territory, and responsibility. 
I would like to remind everyone that we’re all on the same 
team, trying to accomplish the same task as efficiently and 
effectively as possible. Thanks to everyone who was involved 
and worked toward the common goal of resolving the issue. 
Also, my apologies to anyone whose turf or toes were 
violated or infringed upon; that was never the intent.

Thanks for reading this column; see you all in the next 
issue.

Pete Clark works for CPR Systems and has spent more than 45 years working with 
z/VSE in education, operations, programming, and technical support positions. He’s looking forward to celebrating the 50th anniversary of z/VSE. It’s his privilege 
to be associated with the best operating system, best users, and best support 
people in the computer industry. Email: pclark@cprsystems.com
CICS Explorer was introduced in November 2008 for z/OS. Now this feature is available for CICS Transaction Server (TS) on z/VSE. This enhancement provides services that let you display CICS status information on a workstation in a compact, complete way.

Even experienced CICS TS users can find monitoring CICS activity and resources challenging; most of the information comes from CICS statistics or via CICS inquire commands. These interfaces require some knowledge to operate. CICS Explorer makes it easier to collect this information without requiring deep knowledge of the CICS commands and utilities used if retrieving the information the traditional way using CICS transactions. ➤
CICS Explorer consists of two parts, a workstation-based client and a host-based server. The client is based on Eclipse Rich Client Platform (RCP) technology, which is a cross-platform framework for building and deploying PC desktop applications. With the new client code and the server code on z/VSE 5.1.1, CICS Explorer now works with CICS TS on z/VSE.

The new server code is part of CICS TS for VSE/ESA. It lets the client retrieve useful data from a single CICS TS partition. The CICS Explorer client can connect to any CICS TS partition on z/VSE, or any CICS or CICSPLEX on z/OS.

One advantage is that the client code is the same for z/VSE and z/OS, so it can be used by large shops with both operating systems and multiple CICS partitions or regions as well as by small z/VSE shops with only one CICS partition. CICS Explorer client can be configured for individual needs.

### CICS Explorer Capabilities on z/VSE

The CICS Explorer displays resource information in views. You can determine the size and position of the views and information to be shown. Views can be grouped to a perspective, which is a layout of one or more views in the workbench. You can decide how you want to lay out the views in the CICS Explorer and save the layout as a new perspective. You can switch between different perspectives as needed.

The client will retrieve information from the z/VSE CICS TS at first selection or when you press the refresh button. For any failing request, a message appears in the error log window of the client.

An example of such a view is an overview of all files defined to CICS TS. Detailed information about each file (such as status of the file, allowed operations on it, and

<table>
<thead>
<tr>
<th>Operation request</th>
<th>External Resource Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>View</td>
<td>CICSExploreResourceName</td>
</tr>
<tr>
<td>Document Templates</td>
<td>CICSDocumentTemplate</td>
</tr>
<tr>
<td>Files</td>
<td>CICSLocalFile</td>
</tr>
<tr>
<td></td>
<td>CICSRemoteFile</td>
</tr>
<tr>
<td>Interval Control</td>
<td>CICSIntervalControl</td>
</tr>
<tr>
<td></td>
<td>CICSMROConnection</td>
</tr>
<tr>
<td>Queues - TD Queues</td>
<td>CICSExtrapartitionTDQueue</td>
</tr>
<tr>
<td></td>
<td>CICSSDirectTDQueue</td>
</tr>
<tr>
<td></td>
<td>CICSTntrapartitionTDQueue</td>
</tr>
<tr>
<td></td>
<td>CICSSRemoteTDQueue</td>
</tr>
<tr>
<td>Queues - TS Queues</td>
<td>CICSTSQueue</td>
</tr>
<tr>
<td>Programs</td>
<td>CICSProgram</td>
</tr>
<tr>
<td>Regions</td>
<td>CICSRegion</td>
</tr>
<tr>
<td>Tasks</td>
<td>CICSTask</td>
</tr>
<tr>
<td>TCP/IP Services</td>
<td>CICSTCPIPService</td>
</tr>
<tr>
<td>Terminals</td>
<td>CICSTerminal</td>
</tr>
<tr>
<td>Transactions</td>
<td>CICSLocalTransaction</td>
</tr>
<tr>
<td></td>
<td>CICSSRemoteTransaction</td>
</tr>
<tr>
<td>Transaction Classes</td>
<td>CICSTransactionClass</td>
</tr>
</tbody>
</table>

**Figure 1: Supported Views**

An example of a CICS Explorer view is shown in Figure 1.

**Figure 2: Sample for a Program View With Detailed Information on Program CEEDCOD**

A sample view for a program is shown in Figure 2.
Here's how this works: The CICS Explorer Client connects to CICS TS on z/VSE through the CICS Web interface using the HTTP 1.0 protocol. The CICS Explorer server code receives the client request and converts it into EXEC CICS INQUIRE commands or Control Information Queries. Results from these commands and queries are converted back to HTTP format and returned to the client. Results appear in views. The client will recognize that the connected server is z/VSE and adjust the views accordingly.

For z/VSE, we have a one-to-one client-server connection to the so-called Single Server Management Interface in CICS TS (see Figure 3). For z/VSE, CICS Explorer Client talks to each CICS partition individually; there's no support for CICSplex to accommodate grouping together several CICses. Each client request is answered directly for this CICS.

Not all views the CICS Explorer client offers are supported for z/VSE because the related resource isn't available for z/VSE. Examples include pipelines or Java-related resources. A meaningful message is issued if the view isn't supported.

The CICS Explorer connecting to z/VSE displays CICS resources or data only. This first edition of CICS Explorer doesn't support update capabilities such as enabling or disabling programs. Update capabilities are listed under Definitions in the client display functions under Operations.

Figure 1 shows supported operations. The external resource name is used in the query requests; it directly relates to the selected operation in the client.

Views with many instances, such as programs or terminals, require a reasonable amount of storage in the CICS TS partition. For example, with the delivered standard programs, you need 450MB of Extended Dynamic Storage Area (EDSA) storage to display programs.

Figure 2 shows a sample for a program view with detailed information on program CEEDCOD. The overview of programs will list all programs defined and show information such as status, use count, and language the program is written in; double-clicking shows detailed information in a separate window. This works similar for other views. On the lower middle window, the error log appears. You also can show the host connections with associated credentials.

Compared to the traditional approach, retrieving this information for program CEEDCOD using transaction CECI is more complex. You need to know the program name. To get a full list of all programs available to CICS, you can retrieve the list of defined programs through CEDA transaction, but this requires some knowledge of the CICS Control Dataset (CSD) and its organization.

The details for program CEEDCOD shown in the right part of Figure 2 were retrieved in the traditional way using CECI INQUIRE PROG(CEEDCOD); the output is displayed on several pages similar to Figure 4.

So, in the traditional way, all the information on all programs and program details can be gained with only two or more separate requests resulting in two or more separate output displays not linked to each other. The user needs to remember the program name he wants details on and enter it in a second request. This is much easier using the new CICS Explorer.

Besides the more static information in the programs view, the regions view will show more dynamic information such as maximum tasks allowed and number of times the maximum number of tasks was reached. This information, and the short on storage indication, provide valuable performance indicators. For the information shown in the regions view, there's no equivalent CEMT request since CEMT INQUIRE SYS doesn't show all values included in the regions view.

Another helpful view is the task view; it shows what's happening in the CICS partition. TS and TD queue views will help determine usage of temporary storage and transient data queues.

CICS Explorer Client: Getting Started

To activate the CICS Explorer, you must define a connection and credentials to the CICS Explorer Client. This requires installing z/VSE 5.1.1 (or z/VSE 5.1.0 plus selected PTFs). The connection type is defined as well as the IP address and the port that was set up on VSE for the CICS Explorer. The credentials provide the CICS Explorer with the required user and password information to complete the connection.

You may have multiple connections from one client to various CICs TS partitions on different z/VSE systems. You can easily swap between different CICses with the manage connections button on the lower right side of the CICS Explorer Client (see Figure 2). The window is marked here with the name of the currently active connection, iuiinst. Press the green button to disconnect from CICS. For z/VSE, it's important you select CICS Management Interface as type of connection; otherwise, the client can't connect to z/VSE.

Setting up the CICS Explorer Server


When you launch the CICS Explorer, by default it opens in the same language as the operating system of your workstation, provided the language is one CICS Explorer supports. Currently supported languages are English (en), Japanese (ja), and Simplified Chinese (zh). You can launch the CICS Explorer in any of the supported languages independent of the operating system language. There are several ways you can launch the CICS Explorer in one of the...
The CICS Explorer displays resource information in views. You can determine the size and position of the views and information to be shown.

supported national languages. For details, see the CICS Explorer help information.

**CICS Explorer Help**

You can use the CICS Explorer online help to browse, search, and print system documentation. It includes a user guide that provides a basic tutorial. It introduces the CICS Explorer and outlines its base functions. A text search facility for finding information you need by keyword is also available. Of course, you can also find information about the CICS Explorer on the VSE homepage.

**Security**

All HTTP requests the client issues use basic authentication. The special Web analyzer program, DFHWBADY, validates any client request. Basic authentication requires that CICS TS is started with SEC=YES in the System Initialization Table (SIT). Besides this basic authentication, the connection between the client and CICS can be secured with standard Secure Sockets Layer (SSL) encryption.

**Technical Details**

On the server, two new functional parts are shipped as one Program Temporary Fix (PTF) and are part of CICS TS for z/VSE:

- A client interface handles the incoming and outgoing requests on the server side.
- The system interface interfering with CICS TS resolves the requests.

CICS Explorer has access to CICS operations only; there's no access to the CSD. If a request is received for a specific resource referenced by an external resource name (e.g., CICSLocalFile), the client interface passes the request to the system interface that retrieves the information; for example, by using the EXEC CICS INQUIRE FILE command. The system interface provides the information to the client interface, which converts the information to HTTP format. The response going back to the client consists of a sort of header record indicating how many files were found, if any, or about any errors that occurred. If there were no errors and at least one file was found, the header record is followed by a body with details about each file known to CICS TS. The client will then receive the information and display it in the appropriate view. Details can be shown with a double-click at the file record of interest.

**Debugging**

There are a couple trace facilities in the client. On the server side, CICS TS debug capabilities such as AUXTRACE can be useful. Service personnel can also request partition dumps and transaction dumps. There are also a couple of debugging transactions available, but these are to be used only if instructed by service personnel.

**Conclusion**

CICS Explorer is a useful tool to retrieve all kinds of information about CICS TS on z/VSE. It needs some amount of partition storage, but its impact on CPU usage is affordable since it's directly dependent on the number of requests the client issued. The client and server setup are easily established and will help in administrating CICS TS, even if changing and updating must occur in the traditional manner. CICS Explorer isn't intended for application programmers but is a great resource for administrators and operators. ETJ

August Madlener is a senior programmer with the IBM z/VSE Development and Change team. He has worked more than 30 years on different components of VSE. He's currently assigned to the development team for CICS Explorer Server code running on CICS Transaction Server for z/VSE.

Email: madlener@de.ibm.com
Upcoming Installer and Boot Changes for RHEL

In previous columns, I’ve referred to several changes coming up through the Fedora “experimental test” environment into Red Hat Enterprise Linux (RHEL). One of these affects the initial RAMdisk construction method used to help in the boot process, incorporating a new tool called Dracut. RHEL 6.3 and its derivatives incorporate Dracut, and it changes the operation of a RHEL-based system start-up in several ways.

To review a bit, at boot time Linux loads the kernel, a parameter line, and an initial RAMdisk (“initrd”) file containing a memory image of a filesystem containing enough commands and configuration files to allow the kernel to find the root file system. This process can be time-consuming, and on non-System z platforms tends to be the limiting factor in how quickly a system can be booted or rebooted; this is due to the process of testing the environment and loading appropriate device drivers and configurations (System z has so few device types that this usually isn’t a problem for us). It’s possible to customize this initial image, and most of the distributors do a fair amount of customization to get things going with various device types and configurations for individual virtual and physical machines.

Unlike the existing process, Dracut is an attempt to have a completely generic initial RAMdisk with little hard-coded into the initial RAM filesystem. The initial RAM filesystem loaded from the initrd file has basically one purpose: locating and getting the root filesystem mounted so we can transition to the real root filesystem and continue start-up. Instead of scripts hard-coded to test for certain devices and do various things, Dracut depends on the userspace device functions (udev) to create device nodes at device discovery time and configure them when found, searching for the real root filesystem device node. When we have the root filesystem’s device node, the kernel mounts it and carries on using the information on the real root filesystem. This event-driven approach helps limit the amount of time required in the initial RAM filesystem code, loading and executing only what’s needed in the machine configuration instead of testing every possible device even if there aren’t any devices that match the script. This approach makes boots of less than 5 seconds possible and helps simplify use of non-Extended Count Key Data (ECKD) disks as boot volumes on System z.

Most of the initial RAMdisk generation functionality in Dracut is provided by several generator modules that are sourced by the main Dracut script to install specific functionality into the initial RAMdisk. The modules live in the “modules” subdirectory and use functionality provided by Dracut functions to do their work. A presentation by Harald Hoyer of Red Hat at www.harald-hoyer.de/personal/files/dracut-fosdem-2010.pdf describes Dracut in detail.

So, how does this apply to us in the System z world? In general, it means a couple of things: fewer runs of zipl (in that less information is hardcoded into the initial RAMdisk), and it’s now realistic to think much harder about IPLing Linux from a Named Shared System (NSS) instead of disk. By using the dynamic configuration capabilities of Dracut, the need to run zipl to capture changes to disk configurations and other start-up parameters is dramatically reduced. Most of the information stored in the initial RAM disk image in the current environment can be determined by Dracut; perhaps in future releases, it will no longer be necessary to remember to run zipl after disk changes. Our testing showed a few bugs yet, but it’s definitely better than what we had before.

Also, the idea of IPLing from an NSS is particularly tempting, as it allows use of Internet Small Computer System Interface (iSCSI)-based disk as root filesystems; converged storage and data networking have provided a significant push in data center and enterprise network design, and IBM’s current method of managing Fibre Channel Protocol (FCP) disk storage on z/VM still isn’t well-integrated into the z/VM environment (i.e., the major directory managers still don’t support it well). iSCSI-based storage allows a number of optimizations, removing some of the limits on the number of paths to a storage device, and in a Single System Image (SSI) cluster environment, the restrictions on device addresses and placement no longer apply. Most storage vendors support iSCSI, and use of iSCSI storage is rapidly outpacing traditional FCP storage on distributed hosts. A little experimentation shows it’s possible to easily create a shared IPLable Linux NSS that has no ECKD or FCP disk and can be migrated easily between SSI hosts in a z/VM 6.2 environment.

All these positives aside, creating and managing a system based on Dracut is different. The presentation mentioned previously describes many of the differences; it’s worth reading to understand what’s coming next. 

Dr. David Boyes is CTO and president of Sine Nomine Associates. He has participated in operating systems and networking research for more than 20 years, working on design and deployment of systems and voice/data networks worldwide. He has designed scenarios and economic models for systems deployment on many platforms, and is currently involved in design and worldwide deployment of scalable system infrastructure for an extensive set of global customers.

Email: info@sinenomine.net

ETJ
In the early years of CICS, most of the resources any region used were produced by coding a table. These tables contained resources the region required to run system functions and application programs. Some resources were programs, some transactions, some journals for recovery, etc. The tables were coded so the source could be assembled and made into an executable load module. The load module produced wasn’t a program such as one created from COBOL source; it was a specially constructed module with a specific name CICS could recognize when required. Since only CICS used these, the module name had the standard prefix (DFH) followed by the resource category (such as PPT for programs) and a suffix that was unique so customers could create multiple copies. For example, there could be a DFHPPTFI for the CICS region that ran financial applications and a DFHPPTWH for the CICS region that ran warehouse applications. The main table was the System Initialization Table (SIT), which was used at CICS start-up and contained most of the system parameters for the region. Figure 1 shows the format for a typical SIT. The suffix would be used to load the parameters at start-up. A complete explanation of the SIT and its contents appears in the first article in this series. IBM provides some basic table examples in a library called DFHSAMP.

**Why Tables Had to Go**

While table support lasted many years, it became one of the biggest limitations to online availability and 24x7 processing requirements. The contents of tables are only loaded at CICS start-up, so any changes require a recycle of the region. Many installations now only recycle their regions after weeks or months. Something had to change. The Terminal Control Table (TCT) became unmanageable since networks were growing and reassembling this table, and recycling the regions wasn’t feasible.

IBM responded by introducing Resource Definition Online (RDO). Each new release of CICS delivered additional resources that could be defined and maintained via this facility. The first delivery was for programs since they were the most prolific resource to most customers. Defining programs every time a new application was introduced was time-consuming and cumbersome. Moreover, most test or development regions had frequent changes so reassembling these resources and recycling the regions was a maintenance burden. IBM has done an amazing job of delivering RDO and removing tables from CICS support.

Resources defined via RDO are stored in a CICS file called the CICS System Definition (DFHCSD). Each region can have its own DFHCSD, but regions usually share the file since resources can be defined in the file but not used by all regions. It makes the maintenance process more efficient and flexible. When CICS initializes, it reads whatever tables are still specified to the region and then reads from the DFHCSD for groups that were identified in the DFHSIT as necessary for this region. One region can specify whatever groups it needs; other regions can use others. The groups that are loaded are specified via the DFHSIT parameter, GRPLIST. In most cases, customers use more than one list to customize the resources any region uses. For example, the following GRPLIST parameter could be specified:

```
GRPLIST=(DFHLIST, REGXLIST)
```

This would cause CICS to first load the groups in list DFHLIST, which are the defaults for all CICS regions. It would then load all the groups in REGXLIST and the resources contained within. There can be a concatenation of four lists in this parameter so customers can use lists common to all regions and unique to a specific environment. Once the region is up and running, CICS doesn’t re-read the DFHCSD unless told to do so by specific commands. These commands can “install” new or changed resources, which are loaded out of the DFHCSD for this
region. For a complete explanation of the CEDA, CEDB, and CEDC transactions, refer to the Resource Definition Guide in the CICS library.

DFHRPL and Maintenance Issues

One of the other limitations to continuous processing was the content of load libraries in the DFHRPL (the libraries from where the application programs are loaded). The concatenation of this resource is critical. When CICS receives a request to load a program, it starts at the top of the concatenation and continues down the chain until it finds a module that matches the name. If there are multiple modules in the concatenation with the same name, CICS never uses the second, it always uses “first hit.” So, if an application programmer re-creates a new load module in a lower library, it will never get used.

In addition, the concatenation of DFHRPL is loaded at CICS start-up and can't be changed. No new libraries can be added; no libraries can be removed. Changes to the list can only be done when the region is recycled. In recent releases, however, IBM has delivered the definition and maintenance of DFHRPL via RDO. While the coding of the DFHRPL libraries can be put into CICS at start-up, it can also be supplemented via an RDO group for that region. This group can contain multiple libraries and be modified while CICS is up. This is accomplished via the RDO LIBRARY definition, which can deliver more flexibility to the DFHRPL concatenation. If you haven't looked into using this resource, especially in a development environment that requires many changes to the application libraries, refer to the Resource Definition Guide of the release you're running for a complete explanation of this facility.

Automatic Installation of Resources or Auto-Install

As previously covered, most resources in CICS were originally defined via tables that were moved to RDO for more flexibility. The first implementation in that progression were terminal resources. Even if terminals could be defined via RDO, there were still thousands of devices that would have to be defined that way. This was a huge limitation to availability. Even back in the '90s, the resource that required the most maintenance were terminals. There may be hundreds of programs and transactions, but they didn't come close to the thousands of terminals and the way networks grew regularly.

CICS Virtual Storage (VS) V1.R7 provided the ability to auto-install any device that requested access to CICS. Since the device (at that time) had to be defined in Virtual Telecommunications Access Method (VTAM) and therefore passed the request to CICS, it passed the terminal id, terminal type, and other characteristics to CICS at the time of connection. CICS was then able to read the control block and extract the necessary information to build the terminal definition within CICS. This was done (and still is) via multiple “models” stored in CICS that attempted to match the characteristics VTAM delivered with the characteristics stored in the model definitions to produce a proper terminal control block (TCTTE). No unique definition for each terminal was required in CICS, and the terminal definition was built and connectivity was produced for each device.

The result was phenomenal. Terminal definitions were built as the device requested connectivity, and the control blocks and associated storage were deleted and freed when the terminal signed off. This was an availability advantage and saved tons of storage because the definition didn't remain in the region since it may not be needed again. This greatly improved network support. Customers flocked to CICS VS V1.R7 just to take advantage of this major breakthrough.

The next enhancement to auto-install came shortly after terminals with the delivery of programs. This was probably the other most widely accepted resource to be auto-installed. So many programs in CICS have the same characteristics. They're probably COBOL and follow the Application Programming Interface (API) standards for command-level CICS. These programs must be compiled using the CICS translator and then have the “stub” for all CICS programs inserted into the load module. When CICS is called to execute any program, the Loader Domain finds the module in DFHRPL and detects the characteristics before passing control. The characteristics would be the language, link-edit attributes (24-bit or 31-bit), size, etc. Since the Loader Domain can detect these characteristics, there really doesn't need to be a program definition since CICS can determine most of them itself. The removal of program definitions was a huge leap and made CICS maintenance much easier.

Summary

CICS continues to evolve and become more efficient. The basic core principles remain that made CICS the most popular transaction processing software in the industry, and new features in every release help customers with daily support.

Phyllis Donofrio is president and owner of her own consulting services company, which has been providing CICS and network expertise for more than 25 years. She is the author of three books, many technical articles, and was a regular speaker at past SHARE and IBM CICS and Messaging Technical Conferences. She’s not on Facebook, and doesn’t Tweet, but you can contact her via email.

Email: PDonofrio@aol.com
Has DB2 10 for z/OS been on your mind lately? Are you thinking it’s time to start putting together a DB2 10 migration plan? If yes, that’s great. However, if upgrading isn’t yet on your to-do list, why not? DB2 10 became Generally Available (GA) on Oct. 22, 2010. It’s definitely not new anymore. In fact, the number of licenses being shipped is higher than previous versions. DB2 10 is stable and customer reports have been extremely positive.

Maybe the age and the future of available service for the DB2 version you’re running should be a migration catalyst? Both DB2 Version 8 and DB2 9 have been around awhile. DB2 9 went GA on March 16, 2007, and DB2 V8 dates back to March 26, 2004. It saw its End of Service (EoS) on April 30, 2012; DB2 9 is scheduled for EoS on June 27, 2014.

Skip-Level Migration

If you haven’t yet started, it’s time to formulate a DB2 10 upgrade plan. Remember, planning and migrating are two different tasks. Even if implementing DB2 10 immediately isn’t right for you, there are still numerous tasks you can complete in preparation for that effort—things that will help position your DB2 system for a successful upgrade to DB2 10 when the time is right.

Most of the steps required to move to DB2 10 are pretty much the same whether you’re on DB2 V8 or DB2 9. If you’re on V8, you must decide if your next DB2 migration will be to DB2 9 or DB2 10. If there are new functions in DB2 10 that you’re interested in, can you afford to wait to complete two full migrations before you can use those functions? Skip-level migration, moving from V8 directly to DB2 10, can get you where you want to be sooner than later.

The opposite is also true, however. If there’s functionality you need now in DB2 9, can you wait through an extended migration, one that will likely take longer than a single version migration, to get that new functionality?

That brings us to another question: Are you doing a skip-level migration for the right reasons? You don’t skip migrating to DB2 9 to get to DB2 10 because you think it will be faster than going to each version individually. You aren’t going to save yourself much time that way.

It’s nice that skip-level migration is a tested feature. It was part of the DB2 10 beta. You aren’t the first to try it. A good percentage of migrations to DB2 10 have come from V8. You need only understand the impact of a skip-level migration. You’re getting all the new stuff and getting rid of some of the old stuff in a process that previously would have taken you two complete migrations. Consider, too, the learning curve. There’s definitely a lot to absorb when you skip over an entire version of DB2.

What things can be completed regardless of the migration strategy? What can be performed now so it won’t interfere with what must be completed when a DB2 10 migration occurs?

Upgrade Prerequisites

Any DB2 upgrade has prerequisites. Usually, the prerequisites can just be completed, no matter when you
Dynamic STEPLIBs

and
dynamic ISPF

• Specify task and STEPLIB concatenations dynamically within ISPF
• Eliminate multiple LOGON procedures and repetitive LOGON / LOGOFF sequences
• Switch among multiple DB2 subsystems in a single ISPF session
• Reduce LOGON CPU and elapsed times by as much as 90%
• Run any ISPF application dynamically (without the need to pre-allocate dialog libraries)
• Remove all those datasets from your LOGON procs and eliminate merged libraries with hundreds or even thousands of members

TASKLIB — don't run TSO/ISPF without it!

Relational Architects International

For a free trial call 800 776-0771 www.relarc.com

ISPF is a registered trademark of IBM Corp.
If you haven’t started already, it’s time to formulate a DB2 10 upgrade plan.

plan to start the actual upgrade. First, determine the minimum requirements necessary for an upgrade to DB2 10 for z/OS. Some of the requirements, such as operating system level and machine type, aren’t usually things you can just “take care of” over a weekend. You should build a pre-plan covering what must be completed before you can implement a migration plan.

The absolute bare minimum requirement for a move to DB2 10 for z/OS is a processor that supports z/Architecture running in 64-bit mode; for example, the z890, z990, z9, z10, z196, z114, or newer System z processors. That processor must have at least one Logical Partition (LPAR) running with z/OS 1.10 or above and must have adequate real storage to support the combined requirements of DB2 10, z/OS, and Data Facility Storage Management System (DFSMoS), along with access methods, telecommunications, batch, and other customer applications. Caution: DB2 10 will likely require an increase in real storage over whatever you were using for DB2 V8 or DB2 9.

There may be other hardware and software requirements you should be aware of, depending on what you plan to use in DB2 10. Refer to the most recent DB2 10 Program Directory (GI10-8892) for a complete list of installation requirements and additional recommendations and considerations, including migration/fallback Authorized Program Analysis Reports (APARs). Especially helpful are APAR II14474 for a DB2 V8 to DB2 10 migration and APAR II14477 for a DB2 9 to DB2 10 migration. Note: You can use your favorite Internet search program to locate any APAR by number. In almost all cases, specifying just the APAR number is sufficient.

Although z/OS 1.10 is the minimum operating system level required for DB2 10, it may not be the level you actually start at. For example, DB2 10 can take advantage of improved open, close, allocation, and de-allocation functions in z/OS 1.12. Consider, too, that z/OS 1.10 went out of service on Sept. 30, 2011, z/OS 1.11 will be going out of service on Sept. 30, 2012, and z/OS 1.12 has been GA since Sept. 24, 2010.

If migrating from a previous version of DB2, there are also minimum levels that can be used as a starting point. DB2 10 has two starting points from which a migration can be launched. DB2 10 supports a migration from a DB2 9 that’s running in New Function Mode (NFM). DB2 10 also supports a skip-level migration from a DB2 V8 running in NFM.

DB2 10 uses a newer Internal Resource Lock Manager (IRLM), IRLM 2.3, than previous DB2s. DB2 V8 and DB2 9 both currently use IRLM 2.2 so an IRLM upgrade is required. You should consider resizing the IRLM. One of the more significant changes in DB2 10 is the restructure of the DB2 catalog. Part of the restructuring is the use of row-level locking for many catalog tables. This new use of row-level locking by the catalog could result in an increase in the storage size of the DB2 10 IRLM. Also, be aware that the IRLM is installed into the same System Modification Program (SMP) zone of the IRLM used by IMS.

When you migrate to DB2 10, some functions are removed. You should verify that these functions aren’t being used or take corrective action anytime before starting the DB2 10 migration. One feature receiving more than its share of press finally is the removal of private protocol. Any packages that access a remote location must be bound, specifying the option DBPROTOCOL (DRDA). If it’s still bound with private protocol and attempts to reach a remote location, the application will fail with message DSNT225I.

You can and should also eliminate the ability to bind Database Request Modules (DBRMs) directly into a plan. In DB2 10, all DBRMs are bound into packages and those packages are bound into plans. Plans in DB2 10 will always use packages. If a plan runs in DB2 10 that contains DBRMs bound directly into it, DB2 will automatically create packages from the DBRMs and rebind the plan using those packages. Handling this change before migration would reduce the impact of this change on your migration.

Another change that could be managed before a migration involves AQUIRE (ALLOCATE) and MEMBER options on the BIND and REBIND PLAN commands. DB2 will issue a warning if AQUIRE (ALLOCATE) is specified and uses AQUIRE (USE). DB2 will also issue a warning message if the MEMBER option is specified. However, for MEMBER, the specified DBRM is bound into a package and then binds that package into a plan.

You should also determine how to gather information about the performance of your SQL in the current DB2 V8 or DB2 9 environment. Devise a strategy describing how EXPLAIN detail and DB2 accounting and statistics information will be retained for use should an access path regression occur once DB2 10 Conversion Mode (CM) is up and running.

DB2 10 changed some rules for using EXPLAIN. EXPLAIN tables must use Unicode encoding and can’t be in a format older than DB2 V8. Migrating the EXPLAIN tables to a DB2 V8 or DB2 9 format and Unicode conversion can precede any DB2 10 migration. Migration and conversion of the EXPLAIN tables is detailed in APAR PK85068.

You should also be aware of the removal of the AIV Extender, Text Extender, and Net Search Extender, along with Net.Data, effective with DB2 9.
If planning a skip-level migration, you should reformat the Boot Strap Data Set (BSDS) and be aware of the removal of DB2 managed stored procedures and the deprecation of simple table space.

As of DB2 9, Partitioned Data Set Extended (PDSE) is required for certain target and distribution libraries in addition to a few SMP/E libraries.

There are four DB2 target and distribution libraries that must use PDSE in DB2 9 and DB2 10. This change affects you if you’re performing a skip-level upgrade from DB2 V8 to DB2 10. The change is already part of a DB2 V8 to DB2 9 migration. The four data sets are ADSNLOAD, ADSNLOD2, SDSNLOAD, and SDSNLOD2. Everything that must be completed is documented in the DB2 10 Installation and Migration Guide (GC19-2974).

With every new version, new DSNZPARM keywords are introduced, defaults and maximums are changed, and some keywords are eliminated. You will want to note the DSNZPARM keywords that are eliminated. Determine if whatever action the ZPARM controls is being eliminated by the next version of DB2 or if it’s becoming part of the product. If something you use is going away, or something you choose not to use is becoming part of the product, determine how that will affect you and plan appropriately.

When upgrading from V8 to DB2 9, some subsystem parameters—MAX_OPT_ELAP, MORE_UNION_DISTRIBUTION, RELCURRH, STORPROC, SUPPRESS_TS_CONV_WARNING, and TABLES_JOINED_THRESHOLD—will be removed. If migrating to DB2 10 from DB2 9, EDMBFIT, LOGAPSTG, MAX_UTIL_PARTS, OPTIXOPREF, PARTKEYU, and SIMISSKY are removed. If a skip-level migration is performed, all the ZPARMs listed previously will be removed. To learn more, refer to “A First Look: DB2 10 DSNZPARM Changes” in the June/July 2011 z/Journal.

Summary

If you’ve been avoiding using DFSMS with DB2 for z/OS, you can’t do so anymore. The DB2 10 catalog and directory must be DFSMS-managed. The SMS environment the DB2 catalog and directory uses must be established before you can begin a migration (or installation) to DB2 10. This is one more migration task you can complete before your actual DB2 10 migration begins. For more information, see “The DB2 Catalog Gets a Makeover” in the December/January 2011 z/Journal.

Regardless of your migration timetable, there are steps you can take to prepare. PM04968 supplies the DB2 10 DSNTIJPM routine to DB2 V8 and DB2 9 as job DSNTIJPA. DSNTILPA can be used to check out DB2 9 and DB2 V8 to ensure they’re suitable for use with DB2 10 for z/OS. The APARs cover all the details about DSNTIJPM. Running this one routine can have a significant effect on the success of your DB2 10 migration.

Education is important, too. Build your DB2 10 and DB2 migration skills as soon as possible. Start by downloading the DB2 10 product documentation from the Web. For example, www.ibm.com/support/docview.wss?uid=swg27019288 points to the DB2 10 product library in PDF format (BookManager is no longer supported in DB2 10) and provides links to the Information Center on the Internet. The Information Center contains DB2 product documentation for all active versions in one place. It’s easy to search and easy to read and is an excellent place to get DB2 for z/OS information.

There’s also a free, one-day DB2 Migration Workshop available from IBM. It’s a terrific way to get ready for a DB2 10 migration. Contact your IBM DB2 specialist for information.

If you want to verify that your DB2 is ready to migrate to DB2 10, information is abundantly available. A little planning and pre-migration cleanup can make for a smooth DB2 10 migration.
Use Backup and Recovery Tools to Support Day-to-Day Operations

Your company’s investment in backup and recovery tools can be leveraged in a variety of ways that you may not have considered. These resources aren’t just for managing disasters or significant unplanned outages, which happen infrequently. The following examples describe how to maximize your investment in these tools by also using them to support your day-to-day operations:

**Replicate DB2 data.** Recovery tools can be used to replicate data from one system to another. You can use these backup and recovery tools in a variety of ways to support DB2 data replication.

Assume that you copy all the databases in the application and then port those copies to some other site or subsystem, such as a quality assurance system, where you upload the databases for regression testing the applications. This is a common process, and there are actions that you can take to make it less expensive to extract that information, apply the data, and perform more automation to complete the process. These actions will help you reduce the cost of supporting the application group and their testing efforts because they provide faster, cheaper copy and recovery techniques that can reduce or eliminate CPU consumption and elapsed time impacts.

**Transform structures.** You may not have considered using a backup and recovery tool to perform certain kinds of structural changes, but these resources have engine components that can support that capability. For example, a health insurance company uses recovery tools to manage structural changes across the DB2 subsystems and bring new systems into production quickly. The company leverages a transformation facility that ensures the data and associated structures maintain their integrity during structural changes.

**Migrate changes.** Another replication technique is to take data from one place—such as a protection DB2 application—and import it into another application, such as an Oracle database. This technique lets you extract changes to the DB2 data from the DB2 log and then format the data. This process allows you to apply the data to the Oracle database. As a result, you can migrate changes from one operating system or database to another and exploit your investment in backup and recovery products to support operations.

**Support audit reporting.** Another innovative approach is to use backup and recovery products to support audit reporting requirements, such as those that apply to the Health Insurance Portability and Accountability Act (HIPAA), Sarbanes-Oxley (Sarbox), and other regulations. For example, these resources can help identify who has been changing data. You also need to determine if power users are making changes they aren’t authorized to make or if there are other change-related actions that could violate company policies. These tools enable you to extract data from the logs and provide audit reports to know what data has changed and what the data looked like before and after the change.

**Perform “practice” recoveries.** Having to perform local recovery is rare, and typically the expertise to do this isn’t always available onsite because, fortunately, disasters occur infrequently. If your tools allow you to practice recovery processes, you can use them to test the scenarios before an outage occurs so you can plan preventive actions. Without testing the processes in advance, you could actually have a recovery spiral—where problems become worse before everything gets recorded. For example, if you’re doing recovery of an application database to a point in time and forget to take along a related database or index, there’s an inconsistency within and between databases that must be addressed.

The solutions you use to test recovery should provide an estimate in hours, minutes, and seconds related to the time required to perform a complete disaster recovery. This planning capability offers the opportunity to periodically practice recovery without actually impacting availability of live applications. You need the right tools to make this happen.

**Return on Investment**

As the pressure increases to do more with less to help your company remain competitive, maximizing the value you extract from every investment will be increasingly important. While your organization may have invested in backup and recovery tools to prevent against a total disaster, keep in mind that by also using these resources to support day-to-day operations, you can gain an even greater return on that investment. ETJ

---

Rick Weaver, DB2 software consultant for BMC Software, has more than 35 years of experience in systems and database administration for IMS and DB2 database systems and has been involved in developing large, complex, mission-critical applications in a variety of business areas. He has been at BMC since 1994. His current area of focus is DB2 recovery and coordinated recovery.

Email: Rick_Weaver@bmc.com
IBM Edge 2013

June 10 – 14
Mandalay Bay | Las Vegas, NV

The premier storage event is bigger and even better with an expanded agenda that includes:

- Storage
- PureSystems
- Cloud technologies
- x86 servers
- Networking
- Big data and analytics
- Virtualization
- High performance computing
- System X

Registration is now open. Get the details at: ibm.com/edge
Speeding Up Your Mainframe:

Easier Said Than Done

By Marcel den Hartog

In recent months, many conversations with customers and fellow mainframe professionals have centered on performance management on the mainframe or, to be more precise, the dying art of performance management. Almost all these conversations have ended with the conclusion that something needs to be done to address this problem.
Before we do this, however, we should all agree that there are really two types of performance management people:

- Those who understand systems performance and deal mainly with the operating system, networking, storage, etc.
- Those more specialized in application performance management who focus on application code, databases, middleware such as WebSphere MQ, and Transaction Processing (TP) monitors.

The two types differ significantly and require different skills, even though they both eventually come together for the user when they engage with a well-performing application. Though the process of solving a performance problem doesn't differ greatly between the two groups, because they're on different teams, communication problems arise that can also impact performance.

The following outlines the specific steps professionals should undertake to get a better handle on performance management.

**Reactive vs. Proactive Approaches**

The term “performance management” implies the existence of a process, and that some level of control is being exerted over that process. Too many companies deal reactively vs. proactively with issues—essentially solving performance problems vs. actually managing performance so issues don't occur. Usually, performance problems are identified when users report them or some mainframe processes take much longer than they should and begin to wreak havoc. Because companies fail to truly manage the problem, different groups go off in different directions to try and solve one or more issues on their own. We all know this, and most of us will agree this isn't the right thing to do. So, how can we shift from a reactive to proactive mode?

**Start With Your Service Desk**

Because so much these days is connected to cost, the best starting point in proactively addressing performance management is your service desk. This is the only place to identify which performance issues were reported in the past six months and which departments (both from the business and IT) were involved, how many users were affected, and how long it took to solve the problems. This action alone should provide a good indication of the impact and the associated costs of performance issues. Once you have data, you can rightly assume that, with proper management, 50 percent of these issues can be prevented. As with everything, preparation is key, and since you're going to spend money, make sure your management understands the value that stands to be derived from the investment.

**Use Your Toolbox**

Only numbers tell the tale, and you should use the tools in your toolbox to come up with those numbers. It's essential to have a tool that measures the response the end user experiences. Without it, you're flying blind. Additionally, this type of tool should be able to help you identify the largest resource consumers on the mainframe.

Determine what other performance management tools you have and how they integrate with each other.

Once you have your numbers, create a baseline. This is crucial because it will help you identify the components that behave differently from what you've defined as normal or acceptable. A baseline should focus, first and foremost, on the business transactions and the agreed-upon Service Level Agreements (SLAs). For this reason, it's important that your tool can actually monitor and measure at that high level. But your baseline needs to drill down to the single operation that might have impacted performance.

**Assign Ownership**

Each week, create a list of the top-10 transactions or processes that take longer than you've defined in your baseline and include those that show a significant change in behavior over the past x days. The latter information often provides an indication that something out of the ordinary is occurring and could eventually result in a bigger problem. If you're lucky, some of these processes will overlap. For example, a poorly performing transaction can be tied to a performance problem with an IMS or DB2 database or a sudden peak in network traffic. Now the management part of performance management kicks in. You need an owner who can ensure every item on the list is entered into the service desk, so all activities can be tracked and, more important, management reporting is automated. Unless someone owns an action on the top-10 list, nothing will be resolved.

**Use Your Tools to Find the Problem**

After the initial investigation, it should be clear whether the problem is an application or systems issue. Although your company may have the products or tools to find every detail of every dark corner of z/OS, including your databases, network and storage necessary to identify the problem, experience has shown that this is only part of the solution. With the myriad products at your company that manage different aspects of the mainframe, it's important for your mainframers to take the time to learn these tools.

This is also the time when different teams need to communicate their findings. Even in a team with mainframers, communication between those who manage the system and those who manage applications can be challenging. Imagine what this means when we include teams that manage platforms other than the mainframe. This is one reason why using the service desk as a repository is critical. People can always go back to determine who else is working on an issue, what has already been done, and how others solved a particular problem. Knowledge management is only as good as the people entering it; so this is another discipline a designated owner should manage.

**Solve the Problem and Provide Feedback**

Once you've found the cause(s) of a particular performance problem, you need to do some looping to determine what should happen next. Can you turn up a buffer setting and does that require a restart? What's the risk of changing the application? How quickly can this occur and
what is the impact of getting this into production in time before it causes a real problem? Do you need to recompile or re-test many more applications, or does it require an upgrade of a vital piece of software? Can you make the change yourself, or is someone else managing that piece of infrastructure for you? If it’s the latter, how much extra will it cost and when can they do it? Certainly, a Configuration Management Database (CMDB) would help here to assess the impact of a change. It may also help you re-order your priorities so you can focus on the things you can really solve.

**Update Your Practices**

It’s straightforward, yes, and probably not the first time you’ve read what’s described here. But as times have changed, so has the process. In the past five years, your mainframe has grown into what it is today: an extremely complex, constantly changing organism where thousands of components interact with thousands of other components. Even though we manage this organism with fewer people than ever, we still try to manage it the same way we did 10 years ago. That’s where the problem lies: things simply can’t be done the way they’ve been done in the past. The environment has undergone too many changes. The balance, or proper relationships, between processes, people, and products needs to be revisited. Let’s look at where improvements can be made:

**Processes:**
- Measure constantly, not only when you’re told things are wrong.
- Know the cost of failure. Understand the cost and impact of a slow-performing application and you can sell the project to solve it.
- Manage risk and expectations. Once you find a solution, implementing it may sometimes be difficult. Plan for the short-, mid-, and long-term and set different goals for each.

**People:**
- Assign an owner. Without one, your tuning project will fail.
- Accept that you’re dealing with fewer experts and fewer people doing more, different things. They need the right (integrated) tools that help them cope with those dynamics.
- Understand that people who are chasing problems are under stress and will come up with sub-optimal resolutions. Working proactively will help find real (long-term) fixes instead of workarounds.

**Tools:**
- It’s likely that most of your tools are more than 20 years old. Revisit them and try to rationalize. More and optimized integration and better support for new technology, such as System z Integrated Information Processor (zIIP) specialty processors, will save you time and money in the short- and long-term.
- Realize that your mainframe doesn’t operate in a vacuum, so neither should your tools. Cross-platform support is key to solving the most complex performance problems.

**Integration Makes It Work**

We’ve touched on “integration” several times, and for good reason. It’s the one element that brings it all together and yields:

**Reduced complexity.** Only with well-integrated solutions can you effectively manage mainframe performance today and in the future. The mainframe has become too complex to be managed merely with standalone tools.

**Improved productivity.** Integration also helps your reduced staff be more productive. Interfaces become more familiar throughout the process, and drill-down processes are supported without having to retype values in several different interfaces. Additionally, integration supports more automation. At a time when mainframe management has slowly come to include younger staff, automation will help them become productive, too.

**Better support of processes.** Integration is essential when we want to examine the complex processes needed to undertake proper performance management. This begins with an initial analysis (what are the problems and what’s their impact?) and continues on to risk analysis, once we’ve determined what must change.

By integrating your tools and processes, performance management goes from being a daunting and resource-intensive challenge to an achievable, realistic goal.

**Conclusion**

Mainframe performance challenges are common. Usually, a company’s technical staff is capable of solving the problem, but often lacks the time and resources (money) to do so effectively. Consider, for example, the case in which a company’s IT staff was only allowed to work on a performance problem if they could re-create it in their test environment. The organization wouldn’t allow performance management solutions to run in production due to high MIPS usage. Needless to say, the problem was solved by simply presenting the company with a project plan that showed the cost of doing nothing, the estimated cost of poorly performing applications, and documented evidence that their toolset was obsolete.

Better performance management isn’t impossible to achieve and can be realized easily. All that’s needed is a bit of common sense, the ability to look closely at where improvements can be made, and the willingness to re-evaluate your current toolbox.

Marcel den Hartog is senior advisor, Mainframe, EMEA for CA Technologies. He's a frequent speaker at both internal and external events such as Guide Share Europe (GSE) and IBM's Technical University, where he talks about CA Technologies’ mainframe strategy, vision, and trends. Before joining CA Technologies in 1986, he worked as a programmer/systems analyst on VSE and MV3 systems, starting with CICS DL1/IMS and later DB2.

Email: zMarcel@ca.com
CALLING ALL ENTERPRISE IT PROFESSIONALS

Join SHARE for the best in enterprise IT content, networking and insight into new and upcoming products and services.

SHARE in San Francisco gives you a chance to dive deep into IT best practices and technical content designed to keep you on the forefront of the IT industry. SHARE in San Francisco offers technical content addressing key industry topics including:

- Mobile in the Enterprise
- Big Data — Big Analytics — Big Needs
- Cloud in the Enterprise
- Virtualization in the Enterprise

Plus everything you want to know about System z, z/VM, z/OS and zEnterprise!

New to SHARE in San Francisco – Mobility Spotlight

Featuring BYOD and MDM as part of a Mobility Spotlight, SHARE continues to bring you the most current content on IT hot topics that are relevant to you. Join us for this unique two-day spotlight as we focus on how advancements in mobile technologies impact the mainframe and the enterprise.

Invite Your Manager

In conjunction with SHARE in San Francisco, SHARE ExecuForum, an exclusive, two-day forum designed for enterprise IT leaders, provides attendees with access to industry thought leaders while encouraging meaningful networking amongst peers. SHARE ExecuForum will address the issues that are top-of-mind to today’s IT business leaders including managing mobile, big data/big analytics, the IT skills/generational gap, leadership topics and much more! To find out more, visit www.SHARE.org/2013ExecuForum.

Learn more and register for SHARE in San Francisco today at www.SHARE.org/SanFrancisco.
What Can You Do to Avoid DB2 Locking Problems?

Locking is a complex issue and can be at the root of many performance problems. But if you follow the guidance offered here, you can reduce the frequency of locking issues in your shop. ETJ

To minimize lock timeouts, design your application programs with locking in mind from the start. Limit the number of rows accessed by coding predicates to filter unwanted rows. Doing so reduces the number of locks on pages containing rows that are accessed but not required. Also, design update programs so the update is issued as close to the COMMIT point as possible. Doing so reduces the time that locks are held during a unit of work, which also reduces timeouts (and deadlocks).

Speaking of COMMITs, it's important to design all your batch programs with a COMMIT strategy. A COMMIT externalizes the modifications that occurred in the program since the beginning of the program or the last COMMIT. A COMMIT makes sure all modifications have been physically applied to the database, thereby ensuring data integrity and recoverability. Failing to code COMMITs in a data modification program is what I like to call “Bachelor Programming Syndrome”—in other words, fear of committing. This can cause lock timeouts and lock escalation.

There are techniques available to DBAs to minimize lock timeouts. When an object is being accessed concurrently by multiple processes, consider using the MAXROWS option of the CREATE TABLESPACE statement to cause fewer rows to be stored on a single page. The fewer rows per page, the less intrusive page locking will be because fewer rows will be impacted by a page lock.

Deadlocks also can cause problems. A deadlock occurs when two separate processes compete for resources held by one another. For example, a deadlock transpires when PGMA has a lock on PAGE1 and wants to lock PAGE2 but PGMB (at the same time) has a lock on PAGE2 and wants a lock on PAGE1. One of the programs must be terminated to allow processing to continue. One technique to minimize deadlocks is to code your programs so that tables are accessed in the same order. By designing all application programs to access tables in the same order, you reduce the likelihood of deadlocks.

Locking is a complex issue and can be at the root of many performance problems. But if you follow the guidance offered here, you can reduce the frequency of locking issues in your shop. ETJ


DB2 lock contention issues can be frustrating problems to investigate and debug. Before blaming DB2 (as is the usual response), try to answer the following questions to help identify the cause of the locking problems:

- Has the application ever run without problems?
- Have the lock timeouts or deadlocks started recently?
- What level of DB2 are you running?
- Does the problem only occur at certain times?
- What has changed on the system (e.g., number of users, number of applications, amount of data, Program Temporary Fixes [PTFs] to DB2 or any other relevant software, etc.)?
- What has changed in the application (e.g., isolation level, concurrent executions, volume of data, etc.)?

When one application program tries to read data that’s in the process of being changed by another, the DBMS must control access until the modification is complete to ensure data integrity. Most DBMS products—DB2 included—use a locking mechanism for all data items being changed. Therefore, when one task is updating data on a page, another task can’t access data (read or update) on that same page until the data modification is complete and committed.

When multiple users can access and update the same data at the same time, a locking mechanism capable of differentiating between stable data and uncertain data is required. Stable data has been successfully committed and isn’t involved in an update in a current unit of work. Uncertain data is currently involved in an operation that could modify its contents.

This is a simplified discussion of locking and isn’t intended to explain all the nuances of locking in DB2 for z/OS. Instead, let’s look at techniques for remediating locking problems.

Lock timeouts are perhaps the most vexing issue encountered by DB2 professionals. The longer a lock is held, the greater the potential impact to other applications. When an application requests a lock that’s already held by another process, and the lock can’t be shared, that application is suspended. A suspended process temporarily stops running until the lock can be acquired. When an application has been suspended for a pre-determined period of time, it will be terminated. When a process is terminated because it exceeds this period of time, it’s said to timeout. In other words, a timeout is caused by the unavailability of a given resource.
RATIONAL DEVELOPER for z/OS: A New Paradigm for Mainframe Development

BY PRASAD GANTI

Rational Developer for z/OS (RDz) is an Integrated Development Environment (IDE) for mainframe developers. A Graphical User Interface (GUI)-based tool that runs on the PC, it helps mainframe developers connect to the mainframe and most of the analysis and development work they normally do in the ISPF/TSO environment through a 3270 emulator.
Using RDz delivers significant performance gains over a 3270 emulator. Stemming from the multiple and concurrent views with context-sensitive editor windows, TSO/UNIX shell windows, DB2 windows, debug windows and job status windows, RDz provides multiple views into the mainframe. For example, one view can display a context-sensitive editor to edit a COBOL program; another view can show a context-sensitive editor to edit Job Control Language (JCL); a third view can be used to debug a program; a fourth view can be used to run a TSO command; and a fifth view can be used to run a UNIX shell command. Since a significant amount of work shifts to the PC, it can lead to lower MIPS usage on the mainframe.

But this developer’s dream comes at a price. There’s a learning curve and little to no support for third-party, non-IBM tools often used on the mainframe. However, IBM provides extensive support via free training, seminars, and a strong development community; these helped facilitate RDz adoption at our site. Our organization is a major mainframe development center supporting a global financial institution. We use all the major mainframe technologies such as batch COBOL, JCL, VSAM, DB2, MQseries, WebSphere Application Server, CICS, etc. We were looking at RDz to improve developer productivity and reduce MIPS on the mainframe.

Here’s the story of the adoption of the tool in our organization.

Plan

We conducted a pilot study to learn more about RDz, document its advantages and disadvantages, and train a small group of power users who would later become internal experts on the tool. We found clear advantages and disadvantages.

Advantages:

• RDz provides a rich GUI with multiple, concurrent views into source files, DB2 tables, a 3270 terminal emulator, TSO commands, UNIX commands, etc.
• RDz enables multiple, concurrent connections to different Logical Partitions (LPARs).
• RDz provides smart, context-sensitive editors you can use to edit different types of source code. The Context Assist feature assists with language-specific verbs (i.e., COMPUTE verb in COBOL). Also, regular ISPF editor commands still work in these GUI-based editors.
• RDz lets you edit and view multiple programs simultaneously.
• RDz provides integrated graphical tools for viewing and manipulating database entities and data. It displays pictures of relationships between different DB2 entities. This information can be exported into design documents. It supports connectivity to other databases such as Oracle, SQL Server, etc. DB2 commands run on the System z Integrated Information Processor (zIIP) and System z Application Assist Processor (zAAP), reducing the cost of MIPS. You can also run queries and retrieve data in tabular format. Java development can be concurrent with mainframe development; there’s no need for a separate IDE for Java development.
• You can run TSO and UNIX command line commands directly in their shell windows.
• Local and remote syntax checking helps develop code that will compile cleanly in fewer attempts than normal.
• You can edit Basic Mapping Support (BMS) maps for CICS using a visual or text editor.
• Sources can be copied from the mainframe to the PC.

![Figure 1: RDz Running Multiple Views](image-url)
allowing work to be done in offline mode without any connection to the mainframe.

- RDz is built using the Eclipse toolkit, an open source toolkit on which several other tools are built. The look and feel are familiar to anyone who has used other Eclipse-based tools such as Rational Team Concert (RTC), another tool we’re now piloting.

- Some functions, such as editing and local syntax checking, now occur on the PC, as opposed to the mainframe, reducing mainframe MIPS and yielding overall savings.

- The provision of multiple concurrent views, context-sensitive editors, easier analysis of data flow, local and remote syntax checks, etc. enhance developer productivity.

Figure 1 shows RDz and the multiple views the developer is accessing simultaneously. In the middle of the screen is the editor. One source is being edited, while other tabs show other sources that are open for viewing/editing. At the right is a tree showing the mainframe connection and the list of files that can be retrieved. In the lower half of the screen is a UNIX shell with command and result windows.

Disadvantages:

- Third-party tools such as Compuware’s File-AID, Serena’s ChangeMan, IBM’s InfoMan, CA’s SAR, etc. aren’t supported directly in RDz. This isn’t IBM’s shortcoming as such. But it’s a problem nevertheless because, like other mainframe shops, we do use a myriad of such tools. Since RDz is an Eclipse-based tool, a community of third-party developers is emerging to provide plug-ins that can be installed into RDz. Specifically, some vendors have a plug-in that can be installed into RDz much like an Internet browser plug-in. We’re using Compuware’s File-AID plug-in in RDz, which is offered free. Some products such as Serena’s ChangeMan have a plug-in that’s available for an extra licensing fee. Some tools (e.g., InfoMan and SAR) have Web access. But some vendors don’t yet have a solution.

- There’s a learning curve involved and a mindset change for mainframe developers who have been using a 3270 emulator for several years. We anticipated this slower adoption of the tool in our organization.

Implementation Challenges

Management decided to roll out the tools to the hundreds of developers across our organization. We trained each developer for a full day on RDz basics, and some underwent a second day of training on advanced topics such as DB2 access, BMS maps, etc. Some refresher sessions were conducted periodically to supplement the training.

Software installation consists of a daemon on the mainframe and a PC-based thick client. The PC-based piece can be implemented on each developer’s PC or on a centralized virtual server, which is accessible to each developer. The virtual server solution is preferable if the infrastructure in an organization is geared to support it. Administration and upgrades become easier. Currently, we have RDz installed on individual PCs. Ultimately, we plan to make it available on virtual servers.

We’re using the floating license server; this eliminates the overhead for administering individual licenses.

Offshore contractors accessing the mainframe through RDz need access to certain ports. Firewall changes need to be made to open up these ports. This applies to us and any site that outsources development work. Secure connectivity must be established between the outsourcing site and host site.

A companion tool to RDz is the IBM Debug Tool, used for debugging COBOL, PL/1, and Assembler programs. Using the tool in CICS regions is challenging, as only one debug tool can exist in a given CICS region at any time. If the IBM Debug Tool is to be installed, we suggest you uninstall the existing debug tool first.

Each mainframe connection and activity a developer makes is logged. We wrote a utility to read the log and produce statistics on tool usage. We use these statistics to monitor tool usage and reporting to senior management.

Conclusion

Several months have elapsed since our mass rollout of RDz. Some developers have adopted the tool more than others. While acknowledging the drawbacks, we’re also encouraging its use. Measurement of RDz usage and continuous management support have eased adoption. We’re still looking for some third-party tools to be used from the PC, either through RDz or a Web link.

We’ve learned that replacing an existing tool with a new one poses some challenges. Chief among them is ensuring that all the functions performed by the existing tool can also be found in the new tool or that workarounds are available. Also, dealing with people’s resistance takes lots of patience. We expect complete adoption of RDz, ultimately, and phasing away the 3270 terminal emulators. ETJ

Prasad Ganti is senior manager for Application Development at Citigroup in New York. He’s responsible for application development on mainframes involving DB2, CICS, MQ, and WebSphere Application Server. He also coordinates the implementation of RDz for his shop.

Email: srinivas.p.ganti@citi.com
When thinking of System z, availability, scalability, reliability, and security come to mind. At the core of these strengths is system integrity. As the 40th anniversary of System z approaches, it's fitting to examine some of the exciting enhancements and security improvements made over the years.

As System z has evolved, IBM's commitment to system integrity has also advanced in two major ways to meet customer needs.

First, processes have been adjusted to account for new technologies and evolving internal guidelines. We constantly see new types of threats, both internal and external to the enterprise. Internal threats are growing at customer sites as potentially disgruntled employees join the ranks of malicious employees. Across the world, terrorist threats continue to mount in the IT area and beyond, where many are enduring a difficult economy. In response, IBM has continued to adjust its processes to the times, allowing customers to be vigilant in how they maintain their systems with regard to system integrity and security. Changes in System z’s architecture and system integrity processes have addressed many threats. This process has evolved into a rigorous combination of automation, testing, and manual observation.

Second, IBM continuously listens to customers about their needs; the company has implemented many improvements to the customer notification process. The process started years ago as letters were sent via mail to individuals at customer locations. Customers now receive an email when a new security or system integrity patch is available. They can then use the IBM Security Portal to obtain the information necessary to keep their systems up-to-date. Customers can also now get Common Vulnerability Scoring System (CVSS) data for each new fix.

Without system integrity, security could be defeated, scalability would be tenuous, availability could be temporary, and reliability could be a house of cards. Ensuring the security and system integrity of an enterprise requires a watchful eye across all operating system environments. This includes being vigilant about applying relevant security and system integrity Authorized Program Analysis Reports (APARs) for z/OS and z/VM. Maintenance of all operating systems deployed on System z should be part of an enterprise security policy. The z/VSE environment is no different. More information regarding the latest process and availability of security and system integrity for System z and general maintenance information is available at www.ibm.com/systems/z/advantages/security/integrity.html.

A Heritage of System Integrity

In the early 70s, the need for computer security was recognized and increasingly became a required quality of service. Access to data and the ability to process it correctly are paramount. A new focus on privacy laws as they pertained to computers and other electronic processing equipment introduced the requirement for individual privacy.

System integrity has continued to evolve over the last 40 years. The birth of system integrity came in 1973 with the announcement of MVS. Former IBMer, W.S. McPhee, articulated in a whitepaper on system integrity in 1974 what he called “a major step in the direction of increased operating system security capability.” He introduced a definition of system integrity that remains fundamentally unchanged to this day. The only minor shift in the definition was to change from general password protection to a more overarching definition to contain the broader resource access control available in RACF, which would emerge later.

As time marched on, the platform continued to evolve. The 80s introduced cross-memory services in ways that didn’t compromise address space boundaries. In the 90s, IBM introduced parallel sysplex, which was built in a way that ensured system integrity wasn’t undermined. The move to 31-bit addressing in the 80s and then 64-bit addressing in the 2000s brought additional challenges. The 2000s ushered in System z, continuing the legacy of system integrity that began with MVS in the 70s, MVS/XA in the 80s, and OS/390 in the 90s. It now continues with z/OS.

As the System z platform evolved, new facilities were added and were appropriately restricted, using techniques such as limiting access to authorized programs, using System Authorization Facility (SAF), checking for restricting access to authorized users, and other technologies to guard the mediation of system resources. This ensures that fundamental principles, such as System z address space isolation, aren’t broken.

Besides the rigorous internal processes in place to discover and investigate potential security concerns internally, it’s possible that an external entity might uncover
a system integrity vulnerability. Regardless of how easy or hard it is to exploit a system integrity vulnerability, IBM will accept APARs and fix vulnerabilities. Once accepted, further evaluation will occur, but it’s unlikely IBM will ever leave a known system integrity exposure uncorrected.

IBM makes security and system integrity a priority; it reaffirmed the system integrity statement for z/OS in the announcement for release 9 on Aug. 7, 2007, and provides a constant reference and reminder for z/OS (http://www-03.ibm.com/systems/z/advantages/security/integrity_statement_zos.html) and z/VM (www.vm.ibm.com/security/zvminteg.html) in the System z security section of the IBM Website.

z/OS System Integrity

System integrity ensures the operating system’s security controls can’t be bypassed. In the context of z/OS, system integrity is defined as the inability of any program not authorized by a mechanism under the installation’s control to circumvent or disable store or fetch protection, access a resource protected by the z/OS Security Server (RACF), or obtain control in an authorized state; that is, in supervisor state, with a protection key of less than eight (8), or Authorized Program Facility (APF)-authorized. One example of a vulnerability on z/OS might be a Supervisor Call (SVC) routine that allows an unauthorized program to enter an authorized state. In the event an IBM system integrity problem is reported, IBM will always take action to resolve it.

System Integrity for z/VM

System integrity from a z/VM perspective is comprised of three attributes more fitting to the functionality of a hypervisor Control Program (CP):

- The ability of the hypervisor CP to operate without interference or harm, intentional or not, from the guest virtual machines
- The inability of a virtual machine to circumvent system security features and access controls
- The ability of the hypervisor to protect virtual machines from each other.

Given the powerful features and functions z/VM provides, it’s critical to ensure that system integrity is maintained and a guest can’t inadvertently or maliciously tamper with the z/VM environment beyond their authority. This is why, for example, it’s critical that the diagnose instructions must ensure a non-authorized guest can’t perform operations in the CP that are beyond the capabilities of the class for that guest.

System Integrity for z/VSE

The z/VSE environment doesn’t have the same formal system integrity statement as z/OS and z/VM; however, z/VSE provides a similar rigorous investigation and analysis of potential security vulnerabilities. When appropriate security patches are produced, customers with a need to know are informed. More details on the z/VSE process appear at www-03.ibm.com/systems/z/advantages/security/integrity_zvse.html and at www-03.ibm.com/systems/z/os/zvse/support/index.html. As with all the System z operating environments, IBM strongly recommends that z/VSE customers validate the currency of their security and system integrity services regularly.

System Security Maintenance

There are often two sides to security patch management. There’s the individual applying the patch and the team or individual who determines the criticality of the patch to the current enterprise environment. Sometimes, they’re the same individual; sometimes, they’re different teams working together to provide a safe, current computing environment.

IBM provides the necessary security and system integrity data to customers with valid licenses for z/OS or z/VM. These customers identify who in their organization has a need to access this critical data. The terms and conditions for gaining access to the System z security portal can be found on the aforementioned Website. Limiting the scope of this data minimizes the attack surface of your System z investment, reducing risk. The goal is to ensure the right security data is in the hands of those responsible for the system and give them every opportunity to succeed in a timely manner.

The current security portal includes data for both z/OS and z/VM. The data for z/OS is provided as enhanced HOLDDATA for APARs that have the Sec/Int flag turned on. System Modification Program/Extended (SMP/E) reads this HOLDDATA and produces a report, alerting the security team of all security or system integrity maintenance that needs to be applied. An important recent addition to this process is CVSS data for z/OS and z/VM APARs. This new data includes base and temporal scores and the vector that produced those scores. The vector lets customers extend the CVSS data to calculate the environmental score for their specific environment, facilitating assessment of the risk to their enterprise. For z/VM, the APAR, Program Temporary Fix (PTF) information, and CVSS data are all included in one file for analysis from both the patch management and security teams. The CVSS data for z/OS is provided in a separate file to be used in conjunction with the HOLDDATA file.

Summary

A more detailed explanation of the current System z security process is available at www-03.ibm.com/systems/z/advantages/security/integrity.html, including specifics pertaining to z/OS, z/VM and z/VSE, and a Frequently Asked Questions (FAQ) document. You can access the IBM System z security portal by registering using the information at www.ibm.com/systems/z/advantages/security/integrity_sub.html. Once you register, you can subscribe to the portal to receive notifications whenever a new z/OS or z/VM security or system integrity PTF is available. E11

Karl Schmitz is a senior software engineer with IBM. He leads the IBM z/OS System Integrity Competency Center.
Email: kdsch@us.ibm.com

Peter Spera is a senior software engineer with IBM. He’s focused on security for Linux on the System z and is also involved with other areas such as system integrity and vulnerability reporting for System z.
Email: spera@us.ibm.com
We continue our discussion of data set security by addressing security for tape data sets. Tape security is different from disk security in three respects: tape labels, tape management software, and the DEVSUPxx member of parmlib.

Tape labels include both external (paste-on) labels and two types of internal labels (records written on the tape itself). Each tape cartridge is identified by a unique, six-character volume serial number (volser). This number is printed on the external label on the side of each cartridge in both eyeball-readable and bar code form.

When your tape librarian receives a box of tapes from the manufacturer, the manufacturer has already written the first record on each cartridge. This is the internal volume label, which includes the same volser as on the paste-on label. When your program needs a tape data set, the system issues a mount message to the operator for that tape’s volser. When the operator (or a robot) mounts the tape, the system reads the volume label and checks that the volser number is correct. This protects against operators mounting the wrong tape.

The second internal label, the header label, provides security. It contains the dsname of the data set and is written on the tape just before the data set itself. When you read a tape data set, the system checks that both the volser in the volume label and the dsname in the header label are correct. This sort of checking is one reason the mainframe is considered more secure than distributed platforms.

You need to address two concerns regarding the security you get from label checking. The first is the ability to bypass the label processing, called BLP. Programmers make this happen by coding LABEL=(2,BLP) on the DD card for the data set. Any user who can do this can bypass security for all tape data sets.

To illustrate this, suppose the master payroll file, named PROD.PAYROLL.MASTER.DATA, is on a tape whose volser is 123456. I have no business reading this data set, but I really want to know other people’s salaries. Suppose my userid is STU, and I code LABEL=(2,BLP),DSNAME=STU.PAYROLL.MASTER.DATA, VOL=SER=123456 on my DD card. The system has the operator mount the tape, but doesn’t verify the dsname is correct. When the security software is called, it sees that the high-level qualifier of the dsname is my userid, so it permits me to access the data. In this way, the ability to BLP lets me access any tape data set I want.

You use your security software to control who can BLP. If you don’t use the security software, then the ability to control BLP depends on settings in JES that are implicitly insecure. Of course, each security package has its own method to do this. RACF uses a FACILITY class rule named ICHBLP. (Note that RACF only uses this rule if the TAPEVOL class is active. Many RACF shops activate TAPEVOL without defining any rules in it.)

CA ACF2 controls the ability to BLP by use of a bit flag in the LID (logonid or userid) record named TAPE-BLP. CA Top Secret controls BLP by permitting to VOLUME rules with ACCESS(BLP).

The second concern regarding tape labels is the fact the header label has room for only 17 characters of the dsname, the rightmost 17 characters. Dsnames can have a maximum length of 44 characters. This introduces a security exposure unless you provide a way to control the full 44 characters of the dsname.

The exposure works like this. Suppose a data set named PROD.PAYROLL.MASTER.DATA is on the tape whose volser is 123456. My userid is STU and I want to read it. I code a DD card with VOL=SER=123456,DSNAME=STU.PAYROLL.MASTER.DATA. The check of the dsname in the header label passes, since I make sure the rightmost 17 characters match. The security software gives me access based on the full 44-character dsname, whose high-level qualifier is my userid.

The solution almost everyone uses is to carry the full 44-character dsname someplace else: in the tape management software, which we will address in the next column.
A previous article presented 10 ways to ensure RACF quickly and efficiently processes the thousands of logon and access authorization requests it receives each minute. This article, which is an update to our 2006 article, addresses 10 additional techniques for further improving performance, including the use of new features that have since been added to RACF. However, before we look at the new tips, let’s revisit the original list (available at http://entsys.me/4pbeq):

1. Implement the Global Access Table (GAT)
2. Maximize in-storage resident data blocks
3. RACLIST classes whenever feasible
4. RACGLIST the RACLISTed classes
5. Increase Enqueue Residency (ERV)
6. Use Virtual Lookaside Facility (VLF) caching
7. Make efficient use of groups
8. Replace OPERATIONS with storage administration authorities
9. Implement Sysplex coupling facility caching

Now let’s consider 10 new techniques:

1. Increase the sets of generic profiles stored in each address space. Started tasks, batch jobs, and Time Sharing Option (TSO) users frequently access multiple data sets with the same High Level Qualifier (HLQ) and multiple resources associated with the same general resource class. These data sets and resources are often protected by generic profiles. Generic profiles are RACF database entries that usually contain masking characters (e.g., *) and typically protect several data sets or resources with similar names. An example is PAY.MASTER.**, which would protect data sets whose names begin with PAY.MASTER.

Upon receiving the first request for access to a data set in a new HLQ or resource in a new class, RACF retrieves and stores a list of all the related generic profiles in the user’s address space. This list is known as a Generic Anchor Table Entry (GATE). RACF uses these GATEs to identify the specific profile protecting each data set or resource the user attempts to access. After identifying the profile, RACF then retrieves and stores a copy of it in the user’s address space for access authorization checking. RACF reuses these in-memory copies of profiles for subsequent access checking rather than repeatedly fetching the same profiles. Over time, a user address space will accumulate multiple GATEs and copies of profiles for rapid reuse.

In releases prior to z/OS 1.12, RACF keeps four GATEs in memory for each address space. Once all are in use, a request for a new HLQ or resource class causes RACF to discard the least recently referenced GATE, along with all its accumulated profiles, and replace it with a new GATE. An address space randomly accessing many different HLQs and resource classes may experience GATE thrashing, where lists and copies of profiles are constantly fetched, dropped, and fetched again. When the number of profiles involved is large, such address spaces may experience a noticeable degradation in performance.

In z/OS 1.12, RACF introduced a new option called GENERICANCHOR, which supports retention of more than the default four GATEs. Each address space may now have up to 99 GATEs. The number of GATEs can be specified either for the entire system, for individual jobs, or
a combination. The RACF operator command SET GENERICANCHOR is used to establish and modify these specifications. If you have a performance-sensitive address space that conceivably might be experiencing GATE thrashing, consider using this feature to increase its number of GATEs to match the number of HLOQs and classes it might reference during normal processing.

If you RAclist a general resource class as discussed in the previous article, RACF won’t need to build GATEs for it because all the profiles are already stored in memory.

2. Periodically reorganize the RACF database. RACF retrieves data from its database in 4K blocks. Each block may contain one or more profiles or profile segments. Over time, RACF administrative actions can cause the following negative effects on performance:

• A profile can be expanded by commands such as PERMIT and CONNECT, causing it to spill over into an additional block.

• Profile and segment deletions can empty all but a small percentage of a block, wasting database and buffer space.

• Each newly added profile segment may be stored in a different block from its corresponding profile, thereby requiring additional I/O to fetch the segment. This most commonly affects logons involving a user profile and its TSO, CICS, or OMVS segment.

To address these issues, reorganize all RACF databases at least annually using the IRRUT400 utility. IRRUT400 realigns index and profile blocks into consecutive order, stores profile segments adjacent to their corresponding profiles, reclaim space for subsequent growth, and eliminates any index errors.

3. Use GRS rather than hardware RESERVEs for sharing a RACF database. When two or more systems share a database in non-Sysplex, data-sharing mode (no coupling facility being used), RACF uses exclusive hardware RESERVEs to serialize the database for most updates. The system holding an exclusive RESERVE locks out all the other systems until it has processed all its update requests. This lock is on the entire DASD volume and affects all data sets on the volume.

Global Resource Serialization (GRS) can convert RESERVEs to global ENQs. Each system is given exclusive control for one update request at a time and the lock is only for the RACF database, not the entire DASD volume. Use of GRS avoids the contention and monopolization issues associated with exclusive RESERVEs.

4. Split the database into multiple data sets. A RACF database is usually allocated as a single pair of data sets, one for the primary and one for its corresponding backup. Such a RACF database has a single set of in-storage resident data block buffers. For large, highly active databases, these buffers may not be sufficient to adequately handle the workload during peak periods.

A z/OS RACF database can be split into as many as 99 primary/backup data set pairs, each with its own subset of profiles. The RACF range table ICHRNRNG is used to specify how profiles are distributed across the various data sets. Each individual data set is given its own set of buffers.

The major drawback to splitting a RACF database is that a change to the ICHRNRNG table requires a Sysplexwide IPL for implementation. Changes are needed occasionally to adjust how the profiles are distributed, especially if adoption of a new profile naming convention has created a profile placement imbalance. Scheduling such IPLs is becoming increasingly more difficult.

5. Isolate the RACF databases. Even if you implement many of the features mentioned in this pair of articles, your RACF database is still likely to incur significant amounts of I/O, especially during peak logon and system usage periods. Placing other high use data sets on the same DASD volume as the RACF database can delay the retrieval of profiles, potentially impacting the performance of all systems sharing the database. If possible, don't put any other data sets on a volume that contains a RACF database. If other data sets are placed on a volume with a RACF database, ensure they won't cause any hardware RESERVEs. This advice applies to both the primary and backup data sets, and, for a split database, to all component data sets.

6. Log judiciously. One essential function of security is monitoring system events to detect suspicious activity such as unwarranted use of high-powered authorities or attempts to access sensitive production files. Monitoring in a RACF environment creates System Management Facility (SMF) records. Excessive logging can potentially overwhelm SMF buffers. In addition, large amounts of DASD and tape resources can be required for record collection and archiving. While it's reasonable to log violations, access to sensitive data, and updates to system files, logging every data set access probably isn't. Take care when monitoring users who might access many z/OS UNIX files and directories as this can generate numerous SMF records.

This recommendation shouldn't be used as a justification for turning off the SETROPTS option OPERAUDIT, which monitors use of the powerful OPERATIONS authority. Installations that rely heavily on OPERATIONS authority should instead seek to replace its use with the storage administration authorities recommended in our prior article. The SMF data collected by OPERAUDIT is instrumental to identifying and eliminating OPERATIONS use.

7. Reduce updates to last-access date. Every time a user logs onto the system, RACF updates the "last-access” date and time in the user's profile, resulting in a write operation to the RACF database. RACF needs this information to enforce password change frequencies and perform automatic revokes due to inactivity. However, RACF only needs to know the most recent date that a user logged on. The most recent logon time is immaterial to these functions, and repeatedly updating this field throughout the day is of little value.
In z/OS 1.11, RACF introduced an option to limit updating of the last-access date to just the first logon of the day. This only occurs when an application (e.g., CICS) passes its APPLID to RACF, along with a USERID and password during the logon process and only when an APPL class profile protects that APPLID. When the APPLDATA field of the corresponding APPL class profile has the value “RACF-INITSTATS(DAILY)”, RACF won't update the user profile when the user's last-access date has already been set to today’s date.

By default, TSO doesn’t pass an APPLID to RACF during log-on processing. Beginning with z/OS 1.10, it’s now possible to instruct TSO to pass an APPLID and leverage this new performance feature. Instructions for activating this feature can be found in the z/OS TSO/E Customization manual.

8. Code NOYOURACC on RLIST commands. When a user issues an RLIST command to list the profile protecting a general resource, RLIST reports the user’s own level of access to the resource. This seemingly innocent behavior can impact performance when the resource is protected by profiles in a member/grouping class pair. Because Universal Access (UACC) settings (i.e., default access) and access permissions in multiple profiles affect the result, RACF must retrieve and RAclist all the profiles in the class pair, then perform an authorization check to determine the user’s access. All this activity occurs in the user's address space. For member/grouping class pairs with thousands of profiles, a single RLIST may require significant I/O to the RACF database to fetch the profiles and large amounts of CPU time to process them. Usually, all this processing is wasted because the person executing the RLIST isn’t interested in his or her own access.

RACF administrators with SPECIAL authority can bypass this processing by specifying the NOYOURACC parameter with the RLIST command. It can be abbreviated as NOY.

Use of RLIST parameters ALL and RESGROUP to list a member class resource will also result in all grouping profiles being retrieved and inspected to identify every profile covering the resource.

9. Avoid I/O-intensive commands and utilities during peak system activity. Certain commands and utilities can cause substantial amounts of I/O to the RACF database. Others lock the database and block all other use until they're finished. Commands LISTUSER * and LISTGRP *, for example, list every user and group, respectively, and each may need to read every user and group profile to gather the required information. The IRRUT200 database backup utility temporarily locks the database while it makes a copy. The IRRUT400 reorganization utility and the IRRDBU00 database unload utility also lock the database, although problems can be avoided simply by using a copy of the database for input rather than the live database. Executing SETROPTS REFRESH commands can generate a large number of I/O requests, especially when either a GENERIC REFRESH for data sets or a RAclist REFRESH for a general resource class with many profiles is issued. Except for emergencies, delay execution of these commands and utilities until off-peak periods.

Note that CONNECT and REMOVE commands, which add and remove users from groups, each generate three updates to the database. Executing large batches of these commands during peak log-on periods can delay users trying to gain system entry.

10. Keep the database free of unnecessary profiles. Profiles take up space in the database as well as space in the database index, GATEs, and unload copies of the database, and they also consume CPU cycles when processed by commands and utilities. Profiles for non-existent users and resources and for unnecessary groups waste space and processing time. Their mere existence can confuse, mislead, and complicate research and analysis of system and security issues. Periodically review the validity of all profiles and remove those that are obsolete.

Conclusion

We often view RACF as the ultimate bureaucrat because everything must wait until RACF passes judgment. We encourage every installation to take action to ensure that RACF responds to each request as expeditiously as possible. This will entail a collaborative effort involving RACF administration, capacity planning, storage administration, and systems programming. We hope the suggestions in this and our prior article help you maximize RACF’s responsiveness.

Robert S. Hansel is lead RACF consultant and president of RSH Consulting, Inc., a RACF professional services firm he founded in 1992. He has worked with RACF since 1986 as a manager, administrator, technician, analyst, auditor, and instructor. He supports several RACF users groups throughout the U.S. Email: R.Hansel@rshconsulting.com; Website: www.rshconsulting.com

Robert L. Whittle is a senior RACF consultant at RSH Consulting, Inc. He has been a RACF systems programmer and consultant since 1992. Email: R.Whittle@rshconsulting.com; Website: www.rshconsulting.com
AFP Generation

FROM A WEB INTERFACE
Advanced Function Printing (AFP) is an IBM proprietary protocol used for printing bulk documents such as bank statements. We have an application that takes as input user text messages, typically marketing messages in specific fonts and sizes, and generates the corresponding AFP overlays. These overlays are merged with the main customer documents, also in AFP form, and printed. This application used to be written in Visual Basic, and a third-party tool was used in the application for generating AFP overlays from the user input text.

Some time ago, we took on the task of converting this application to WebSphere with a Web interface. We could find a third-party text editor and spellchecker to integrate with the Java front-end, but we couldn’t find a Java Application Program Interface (API) for converting the given text to AFP either in rich text or XHTML format. We searched extensively and even approached IBM, but they weren’t interested. This was surprising because AFP is an IBM format and they are a big promoter of Java and Web technologies.

To solve our dilemma, we decided to build a proprietary converter. What follows is a description of how we converted the XHTML string from the front-end, Web-based editor to an AFP overlay, both of which are eventually stored in a DB2 database. The new application runs in WebSphere on z/OS and uses DB2 for storing data.

**The Process**

Figure 1 shows the overall application flow. At a high level, there’s a Web-based editor that lets a user create a message. The message is stored in a DB2 table in XHTML format. Insertion of the message into the DB2 table causes a trigger to execute. This trigger converts the XHTML message into AFP overlays.

The conversion from XHTML to AFP overlays occurs in the following steps in a COBOL-stored procedure invoked by a DB2 trigger:

- Parse the XHTML to separate various text segments and their respective presentation properties. We used COBOL.
XML API for this; the extracted text data and corresponding presentation properties are separated and stored in a temporary table. The presentation properties are stored using the standard IBM font naming conventions.

- The word wrapper algorithm takes the text and its corresponding properties to format it into fixed dimensions of the message box on the physical document to be printed.
- The output of the word wrapper algorithm is in the form of Overlay Generation Language (OGL) commands. OGL is an IBM compiler that takes in formatting commands and generates an AFP overlay.

An Example

Figure 2 shows an example of how a given message undergoes the necessary conversions:

- The user creates a message in the Web-based text editor (see Figure 3).
- The editor stores the message, in the form of XHTML, in a DB2 table. Figure 4 shows the XHTML generated for the previous message.
- This XHTML string is parsed and word wrapped to create OGL code (see Figure 5). The characters aren't visible in the OGL code because the font selected uses EBCDIC code page T1001004.
- This code is run through the OGL compiler to generate the AFP overlay, which is stored in the DB2 table. The stored AFP object can be viewed through an AFP viewer plug-in for the Web browser. The result (see Figure 6) matches the message the user entered into the Web-based text editor.

Summary

We solved our problem with a homegrown solution. If your site uses AFP for printing bulk documents and you need to handle user text messages as inputs and generate AFP overlays, you may want to follow our example. If you do so, let us know how it works for your organization.

Ambadas Choudhari is a certified technology architect with Infosys Ltd. He has more than eight years of IT experience in the financial services domain. He has architected and designed solutions based on a range of technologies, including AFP, CICS Business Transaction Services (BTS), Web technologies, and Extract, Transform, Load (ETL) tools. Email: ambadas_choudhari@infosys.com

Balaji Krishnamurthy has more than five years of design and development experience with mainframe technologies. Through his work as a consultant to securities and banking firms, he has become familiar with the intricacies of Overlay Generation in AFP, font presentation architectures on various platforms, and XML processing on mainframes. He's currently working as a software designer for RBS Technology Services. Email: balinkrish2000@gmail.com
### AD INDEX

<table>
<thead>
<tr>
<th>COMPANY</th>
<th>WEBSITE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC Software</td>
<td><a href="http://www.bmc.com">www.bmc.com</a></td>
<td>5</td>
</tr>
<tr>
<td>CA Technologies</td>
<td><a href="http://www.ca.com">www.ca.com</a></td>
<td>IFC, 1</td>
</tr>
<tr>
<td>Canam Software</td>
<td><a href="http://www.canamsoftware.com">www.canamsoftware.com</a></td>
<td>26</td>
</tr>
<tr>
<td>Compuware</td>
<td><a href="http://www.compuware.com">www.compuware.com</a></td>
<td>9</td>
</tr>
<tr>
<td>Dino-Software</td>
<td><a href="http://www.dino-software.com">www.dino-software.com</a></td>
<td>23, 25, IBC</td>
</tr>
<tr>
<td>DTS Software</td>
<td><a href="http://www.dtssoftware.com">www.dtssoftware.com</a></td>
<td>43</td>
</tr>
<tr>
<td>EMC Corp.</td>
<td><a href="http://www.emc.com">www.emc.com</a></td>
<td>7</td>
</tr>
<tr>
<td>IBM</td>
<td><a href="http://www.ibm.com">www.ibm.com</a></td>
<td>45</td>
</tr>
<tr>
<td>Innovation Data Processing</td>
<td><a href="http://www.innovationdp.fdr.com">www.innovationdp.fdr.com</a></td>
<td>29, BC</td>
</tr>
<tr>
<td>Luminex</td>
<td><a href="http://www.luminex.com">www.luminex.com</a></td>
<td>17</td>
</tr>
<tr>
<td>MacKinney Systems</td>
<td><a href="http://www.mackinney.com">www.mackinney.com</a></td>
<td>13</td>
</tr>
<tr>
<td>MVS Solutions, Inc.</td>
<td><a href="http://www.mvssol.com">www.mvssol.com</a></td>
<td>21</td>
</tr>
<tr>
<td>Relational Architects Global</td>
<td><a href="http://www.relarc.com">www.relarc.com</a></td>
<td>41</td>
</tr>
<tr>
<td>Responsive Systems</td>
<td><a href="http://www.responsiveinternalex.com">www.responsiveinternalex.com</a></td>
<td>3</td>
</tr>
<tr>
<td>SHARE</td>
<td><a href="http://www.share.org">www.share.org</a></td>
<td>49</td>
</tr>
<tr>
<td>Software Diversified Services</td>
<td><a href="http://www.sdssusa.com">www.sdssusa.com</a></td>
<td>15</td>
</tr>
</tbody>
</table>
Friction and Lubricity

Most of us learned about friction in high school science class: For example, “dry friction” is the resistance to relative lateral motion of two solid surfaces in contact; “fluid friction” is between layers of viscous fluids; “skin friction” is a component of drag, as in an airplane body moving through air or a boat hull through water; and “internal friction” is a force resisting motion between elements in a solid material undergoing deformation.

We learned that the predictable consequences of friction were twofold: kinetic energy converted to heat, and also caused the wear or deformation of the substances that were experiencing friction over time. Finally, you may have learned that friction wasn’t a “fundamental force,” but a byproduct of fundamental forces, such as electromagnetism.

That, in turn, made the calculation of friction, by necessity, a function of empirical testing and theorizing rather than the product of a set calculation or algorithm.

As technologists, that was about all we needed to know to understand why moving parts in our infrastructure wear out. Disk drives fail, as do power supply fans and tape cartridges. We know that heat is transferred into our data center as a function of inefficient power conversion and kinetic energy transference—both friction-related. Energy costs being what they are, we’re all dealing with the costs of friction and doing our best to contain them.

Friction also has meanings, perhaps metaphorical, that align with specific non-scientific endeavors. Carl von Clausewitz wrote about it in his military treatise, Vom Kriege (On War), using the term to describe the impact of misfortune, mistakes, and misinformation—the so-called “fog of war”—on the accomplishment of even seemingly simple goals; such friction increases difficulty. Economists, business school professors, and financial sector reporters often use the term friction to describe the impact of transaction costs on a financial market. Oliver Williamson, who was awarded the 2009 Nobel Prize in Economics, is the latest to popularize the financial usage, noting that friction may result from both internal and external transaction costs. The distinction may be useful even for technology planners to understand.

External transaction costs are essentially the costs of engaging in transactions that are imposed by external institutions in the marketplace. Everything from forces of nature, to government regulations, to negative press, to graft and “breakage” are said to impose transaction costs as goods are exchanged between companies A and B. By contrast, internal transaction costs may be represented by the need to build or buy a highly specialized piece of equipment to support a special important or urgent business process—equipment that can’t be repurposed to other activities to improve their return on investment, extend their useful life, or otherwise realize some sort of economy of scale.

Some would say that buying a specialized or proprietary technology device—say, a proprietary storage array or a proprietary server hypervisor software kit—is introducing friction into your environment. Once the decision has been made to use a specific technology or product, changing to a different technology or product usually has a prohibitively high transaction cost. At the same time, staying with the selected technology in a monopolistic relationship with the vendor is also fraught with high internal transaction costs that may include expensive warranty and maintenance agreements, replacement part costs, user/administrator training expense, supplier-prescribed upgrade paths, etc. These internal transaction costs have a way of increasing over time. This phenomenon is called a technology lock-in, and the friction it embodies generally creates its own kind of heat and wear. We see the consequences as declining efficiency, increasing cost of ownership, angry budget-makers, addiction to headache medicine, alcohol or worse, and the like. It’s all scientific, metaphorically speaking.

According to Williamson and others, the logical outcome of high internal transaction costs—or of the friction created when internal transaction costs appear to be greater than external transaction costs—is business downsizing, often via outsourcing. Confronting business operations with intractable internal transaction costs, business planners look for options and there’s always an outsourcer who is ready to do the function at a lower cost (whether or not their claims are valid). Limited rationality and opportunistic behavior, says Williamson, often determine behavior and decision-making.

Conversely, when external transaction costs are greater than internal transaction costs, the logical outcome is business growth as processes are performed internally rather than externally sourced. The idea is to do things less expensively or with greater certainty of outcome or with greater reliability in terms of quality.

We would do well to keep this insight about financial friction front of mind as we consider the options for delivering IT services to our firms. Your comments are welcome. ETJ

Jon William Toigo is a 30-year veteran of IT and the author of 17 books. He is also CEO and managing principal partner of Toigo Partners International, an analysis and consulting firm serving the technology consumer in seven countries.

Email: jtoigo@toigopartners.com; Website: www.it-sense.org
HSM ADMINISAURUS (HSA)


HSA is an HSM health monitoring utility that reports on current and historical mission-critical data and provides diagnostic auditing of problem areas so corrections can be made quickly and easily, saving time and money.

HSA utilizes a wide-range of report formats that provide all the details needed to monitor your DFSMShsm environment including specific information related to the CDS, log, and internal data records for a fast and comprehensive view of your environment.

The audit process quickly and accurately pinpoints errors to prevent loss of services and downtime due to storage shortages, recall/migration failures, and out-of-date backups of essential data.

For more information call 703-768-2610 or email DINO at sales@dino-software.com.
Around the Clock Non-Disruptive Data Migration

- Simple to install and use
- Will significantly reduce migration time
- Allows you to react quickly to performance issues without impacting your applications
- Vendor neutral product supporting all DASD hardware (HDS, IBM & EMC)
- Used by more than 1,400 installations Worldwide

“We swapped 600 production volumes in less than 4 hours, running up to 40 volumes at a time!”

LEARN MORE
Scan the QR code to register and access additional information.

COMPETITIVELY PRICED AND FLEXIBLE RENTAL TERMS! CALL: 973-890-7300,
E-MAIL: SALES@FDRINNOVATION.COM OR VISIT WWW.FDR.COM FOR YOUR FREE NO-OBLIGATION TRIAL!