System z Flash Express
Introduction, Uses, and Benefits

SHARE in San Francisco
February 5th, 2013
Session 13086

Elpida Tzortzatos: elpida@us.ibm.com
Peter Szwed: pkszwed@us.ibm.com
Agenda

• z/OS Customer Value Proposition
• System z Flash Express and z/OS
• Flash Performance Results
• z/OS Flash Roadmap
• Under the Covers – Implementation Highlights
**System z Flash Express IO Adapter**

- Flash Express is a PCIe IO adapter with NAND Flash SSDs (Solid State Drives)
- Flash Express is accessed using the Extended Asynchronous Data Mover
  - Optimized software path for Flash Access based on prior learning with z expanded store
- Flash Express provides continuous availability
  - RAID 10 to cover adapter failure
  - Concurrent Firmware update to cover service
- Flash Express is fully virtualized
  - A single adapter pair can provide Flash to 60 partitions on a CEC
  - Adapter RAS (call home, recovery, etc.) done at system level, not in OS.
  - Transparent migration to new adapter technology
IBM Flash Express – Smarter Availability for Smarter Systems

• Flash Express is an innovative solution designed to help you compete effectively in today’s marketplace
  – Automatically improve availability for key workloads at critical processing times
  – Drive availability and performance for workloads that cannot tolerate paging spikes or inconsistent performance
  – Slash latency for critical application processing such as diagnostics collection

• Extends IBM’s expertise in memory management introducing a new tier of memory using Flash Express

• Provides a secured, resilient and immediately usable solution

• Planned Flash Express and pageable large page exploiters:
  - z/OS V1.13 Language Environment
  - IMS 12 Common Queue Server
  - DB2 10 *
  - Java SDK601 SR4, and Java SDK7 SR3 and by extension:
    - CICS Transaction Server 5.1
    - WAS Liberty Profile v8.5
    - IMS 12
    - DB2

  Traditional WAS 8.0.0x and Traditional WAS 8.5.5 (future) **

**Traditional WAS support is planned for a future date.
*DB2 date to be determined. Support for V10 with APARs is planned.
Flash Express Strengthens Availability

- Innovation to drive availability to exceptional levels
  - Extends IBM's expertise in memory management introducing a new tier of memory using Flash Express
  - Is an *industry unique* application of Flash to improve availability
  - Takes the next step in advanced memory management

- Flash Express can improve availability and reduce latency
  - Improves availability during transition periods and spikes
  - Helps accelerate start of day processing - batch to online
  - Enables faster snapshots of diagnostics (e.g. SVC dump, standalone dump)
  - With pageable large pages can improve performance of DB2 and Java
  - Ideal for applications with random read access and high read/write ratios

- Helps customers deliver vigorous service levels
  - Designed to help provide *exceptional* availability and fast response time
  - Delivered in tandem with pageable large pages for *superior* performance

- Minimal configuration- no special skills needed
  - Usable immediately; no special training required
  - Easy to set up and dynamically configurable
Representative Use Cases - Flash Express

Flash Express can reduce latency delays from paging to bring system availability to new heights and improve overall service levels.

Application related errors will require collection of diagnostics. These diagnostics can be collected faster with Flash Express, reducing paging related delays that can impact your overall availability.

Having your working data resident in Flash can help accelerate start of day processing, and improve service for many industries at the busiest time of their work day - a time when they cannot afford disruptions.

DB2® and Java™ in memory buffer pools work to store and process application data. DB2 and Java can benefit from 1MB pageable large pages with Flash Express, improving overall performance.
Flash Express – What is it?

FLASH Express

► Flash Express is a PCIe IO adapter with NAND Flash SSDs
► Physically comprised of internal storage on Flash SSDs
► Used to deliver a new tier of memory- storage class memory
► Uses PCIe I/O drawer

► Sized to accommodate all LPAR paging
  ▪ Each card pair provides 1.4 TB usable storage (2.8 TB total)
  ▪ Maximum 4 card pairs (4 X1.4=5.6 TB)

► Immediately usable
  ▪ Simplifies capacity planning
  ▪ No intelligent data placement needed
  ▪ Full virtualization across partitions

► Robust design
  ▪ Delivered as a RAID10 mirrored pair
  ▪ Designed for long life
  ▪ Designed for concurrent firmware upgrade

► Secured
  ▪ Flash Express adapter is protected with 128-bit AES encryption.
  ▪ Key Management provided based on a Smart Card
  ▪ Secure Cryptographic Erase meets audit requirements

Flash memory (SCM) blurs the distinction between memory and storage characteristics

All statements regarding IBM’s future directions and intent are subject to change or withdrawal without notice and represent goals and objectives only
Companies competing for the highest quality of service in today’s market must deliver outstanding availability and performance.

Changes in workload processing can impact service levels at critical processing times.

- **Flash Express** is an innovative solution designed to help you improve availability and performance to compete effectively in today’s market.
  - Automatically improves availability for key workloads at critical processing times.
  - Drives availability and performance for workloads that cannot tolerate paging spikes or inconsistent performance.
  - Slashes latency for critical application processing such as start of day processing and also collection of diagnostics (SVC dumps, standalone dumps).
  - Delivered as a new adapter card in the PCIe I/O drawer.

- **Benefits**
  - Improves availability and performance helping companies achieve highest service levels.
  - Delivers a secured, resilient and immediately usable solution.
  - Automatic, requires minimal setup, no special training needed.

All statements regarding IBM’s future directions and intent are subject to change or withdrawal without notice and represent goals and objectives only.
System z Flash and z/OS
Allocating z FLASH

Allocating Flash to a partition

- The initial and maximum amount of Flash Memory available to a particular logical partition is specified at the SE or HMC via a new Flash Memory Allocation panel
- Can dynamically change maximum amount of Flash Memory available to a logical partition
- Additional Flash Memory (up to the maximum allowed) can be configured online to a logical partition dynamically at the SE or HMC
  - For z/OS this can also be done via an operator command
- Can dynamically configure Flash Memory offline to a logical partition at the SE or HMC
  - For z/OS this can also be done via an operator command
- Predefined subchannels, no IOCDS

![Manage Flash Allocation - P87](image)
Full virtualization of physical Flash PCIe cards across partitions, software sees an Abstracted Flash Storage Space...

- Allows each logical partition to be configured with its own SCM address space
- Allocate Flash to partitions by amount, not card size
- Ability to change underlying technology while preserving API

**No Hardware Specifics in Software.**

- Error Isolation, Transparent mirroring, Centralized diagnostics, etc.
- Hardware Logging, FRU Call, Recovery: Independent of software
FLASH for z/OS Paging Value

- Flash Memory is a faster paging device as compared to HDD
  - The value is NOT in replacing memory with Flash but replacing disk with Flash
  - Flash is suitable for workloads that can tolerate paging and will not benefit workloads that cannot afford to page
  - The z/OS design for Flash Memory does not completely remove the virtual storage constraints created by a paging spike in the system. (Some scalability relief is expected due to faster paging I/O with Flash Memory.)
Typical Customer Configurations for FLASH

• Flash card pair memory size is 1.4TB
  – Min: 1 Card Pair
  – Max: 4 Card Pairs

• Typical customer configuration is 6 to 8 LPARs per CEC and 40GB - 80GB for paging configuration dataset size

• Even with 10 LPARs per CEC, each LPAR has 140 GB of Flash Memory available for its paging datasets, more than double the current typical customer configuration.
  – All paging data can easily reside on Flash
  – Data will preferably go to Flash and only go to disk (if any) when Flash is full
  – No intelligent placement of data on internal Flash needed
Flash vs Disk Placement Criteria

**Main Memory**

- Evict Page
  - Check Data Characteristics (i.e. must reside on flash or must reside on disk)
  - If data can reside on either: check space availability
    - Flash full
  - If space available on both check response time statistics
    - Flash is faster

**FLASH**

**Paging Dataset**

HDDs or SSDs
## Flash vs Disk Placement Criteria

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Data Page Placement</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLPA</td>
<td>At IPL/NIP time PLPA pages will be placed both on Flash and disk.</td>
</tr>
<tr>
<td>VIO</td>
<td>VIO data will always be placed on disk (First to VIO accepting datasets with any spillover flowing to non-vio datasets)</td>
</tr>
<tr>
<td>Pageable Large Pages</td>
<td>If contiguous Flash space is available, pageable large page will be written to Flash.</td>
</tr>
<tr>
<td></td>
<td>If Flash is not available in the system configuration pageable large pages will be backed with 4k page frames.</td>
</tr>
<tr>
<td>All other data</td>
<td>If available space exists on both Flash and disk then make a selection based on response time.</td>
</tr>
</tbody>
</table>
z/OS FLASH Use Cases

Paging

• z/OS paging subsystem will work with mix of internal Flash and External Disk
  – Self Tuning based on measured performance
  – Improved Paging Performance, Simplified Configuration

• Begin Paging 1 MB Large Pages only on Flash
  – Exploit Flash’s random IO read rate to get CPU performance by enabling additional use of Large Pages. Currently large pages are not pagable.

• Begin Speculative Page-In of 4K Pages, 1MB Pages only on Flash
  – Exploit Flash’s random IO read rate to get Improved Resilience over Disruptions.
  – Market Open, Workload Failover
Flash Memory Usage and Invocation

- New PAGESCM= keyword in IEASYSxx defines the amount of flash to be reserved for paging
  - Value may be specified in units of M, G, or T
  - NONE indicates do not use flash for paging
  - ALL (default) indicates all flash defined to the partition is available for paging
Flash Memory Usage and Invocation (cont)...

- New messages issued during IPL indicate the status of SCM
  - IAR031I USE OF STORAGE-CLASS MEMORY FOR PAGING IS ENABLED
    - PAGESCM=ALL, ONLINE=00065536M
  - OR
  - IAR032I USE OF STORAGE-CLASS MEMORY FOR PAGING IS NOT ENABLED –
    PAGESCM=None
Flash Memory Usage and Invocation (cont)…

- The D ASM and D M commands are enhanced to display flash-related information/status
  - D ASM lists SCM status along with paging data set status
  - D ASM,SCM displays summary of SCM usage
  - D M=SCM display SCM online/offline and increment information
  - D M=SCM(DETAIL) displays detailed increment-level information

- The CONFIG ONLINE command is enhanced to allow bringing additional SCM online
  - CF SCM(amount),ONLINE
Flash Memory Usage and Invocation (cont)...

P87: Operating System Messages

<table>
<thead>
<tr>
<th>2012200 17.27.02 R71</th>
<th>IEE200I 17.27.02 DISPLAY ASM 143</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE     FULL STAT   DEV DATASET NAME</td>
<td></td>
</tr>
<tr>
<td>PLPA      38%          OK 2002 SYS1.R71.PLPA</td>
<td></td>
</tr>
<tr>
<td>COMMON    6%           OK 2002 SYS1.R71.COMMON</td>
<td></td>
</tr>
<tr>
<td>LOCAL     0%           OK 2003 SYS1.R71.LOCAL</td>
<td></td>
</tr>
<tr>
<td>LOCAL     0%           OK 2021 SYS1.R71.LOCAL1</td>
<td></td>
</tr>
<tr>
<td>LOCAL     0%           OK 2261 SYS1.R71.LOCAL4</td>
<td></td>
</tr>
<tr>
<td>LOCAL     0%           OK 2269 SYS1.R71.LOCAL5</td>
<td></td>
</tr>
<tr>
<td>SCM       0%           OK N/A N/A</td>
<td></td>
</tr>
</tbody>
</table>

PAGEDEL COMMAND IS NOT ACTIVE

<table>
<thead>
<tr>
<th>2012200 17.27.45 R71</th>
<th>IEE2071 17.27.45 DISPLAY ASM 148</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATUS    FULL SIZE   USED IN-ERROR</td>
<td></td>
</tr>
<tr>
<td>IN-USE     0%          33,554,432 13,865 0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2012200 17.28.04 R71</th>
<th>IEE174I 17.28.04 DISPLAY M 150</th>
</tr>
</thead>
<tbody>
<tr>
<td>STORAGE-CLASS MEMORY STATUS</td>
<td></td>
</tr>
<tr>
<td>2848G DEFINED</td>
<td></td>
</tr>
<tr>
<td>ONLINE</td>
<td></td>
</tr>
<tr>
<td>0G-128G</td>
<td></td>
</tr>
<tr>
<td>1872G OFFLINE-AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>0% IN USE</td>
<td></td>
</tr>
<tr>
<td>SCM INCREMENT SIZE IS 16G</td>
<td></td>
</tr>
</tbody>
</table>

Command:

[ ] Priority (select this when responding to priority (red) messages)

Send  Respond  Delete

Close  Help
Flash Memory Usage and Invocation (cont)…

```
2012200 17.30.15 R71

1741 17.30.15 DISPLAY M 163
STORAGE-CLASS MEMORY STATUS - INCREMENT DETAIL
2848G DEFINED
ADDRESS IN USE STATUS
  0G  0%  ONLINE
  16G 0%  ONLINE
  32G 0%  ONLINE
  48G 0%  ONLINE
  64G 0%  ONLINE
  80G 0%  ONLINE
  96G 0%  ONLINE
 112G 0%  ONLINE
ONLINE: 128G OFFLINE-AVAILABLE: 1872G PENDING OFFLINE: 0G
0% IN USE
SCM INCREMENT SIZE IS 16G
```

Command: `d m=scm(detail)`

Priority (select this when responding to priority (red) messages)

Send  Respond  Delete

Close  Help
### Display ASM Command

**`d asm`**

```
IEE200I 17.17.46 DISPLAY ASM 944

<table>
<thead>
<tr>
<th>TYPE</th>
<th>FULL</th>
<th>STAT</th>
<th>DEV</th>
<th>DATASET NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLPA</td>
<td>100%</td>
<td>FULL</td>
<td>02E6</td>
<td>SYS1.PLPA.PAGCOM</td>
</tr>
<tr>
<td>COMMON</td>
<td>61%</td>
<td>OK</td>
<td>02E6</td>
<td>SYS1.COMMON.PAGCOM</td>
</tr>
<tr>
<td>LOCAL</td>
<td>0%</td>
<td>OK</td>
<td>098E</td>
<td>SYS1.LOCAL.PAGEP2</td>
</tr>
<tr>
<td>LOCAL</td>
<td>0%</td>
<td>OK</td>
<td>0987</td>
<td>SYS1.LOCAL.PAGEP3</td>
</tr>
<tr>
<td>LOCAL</td>
<td>0%</td>
<td>OK</td>
<td>098F</td>
<td>SYS1.LOCAL.PAGEP4</td>
</tr>
<tr>
<td>SCM</td>
<td>11%</td>
<td>OK</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>
```

**`d asm,scm`**

```
IEE207I 17.35.02 DISPLAY ASM 947

<table>
<thead>
<tr>
<th>STATUS</th>
<th>FULL</th>
<th>SIZE</th>
<th>USED</th>
<th>IN-ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN-USE</td>
<td>11%</td>
<td>16,777,216</td>
<td>2,096,144</td>
<td>0</td>
</tr>
</tbody>
</table>
```
Flash Related Commands

D M=SCM
IEE174I 17.57.26 DISPLAY M 230
STORAGE-CLASS MEMORY STATUS
80G DEFINED
ONLINE
0G-64G
16G OFFLINE-AVAILABLE
14% IN USE
SCM INCREMENT SIZE IS 16G

D M=SCM(DETAIL)
IEE174I 17.57.30 DISPLAY M 232
STORAGE-CLASS MEMORY STATUS - INCREMENT DETAIL
80G DEFINED
ADDRESS IN USE STATUS
  0G   55% ONLINE
  16G  0% ONLINE
  32G  0% ONLINE
  48G  0% ONLINE
ONLINE: 64G OFFLINE-AVAILABLE: 16G PENDING OFFLINE: 0G
14% IN USE
SCM INCREMENT SIZE IS 16G

CF SCM(16G),ONLINE
IEE195I SCM LOCATIONS 64G TO 80G ONLINE
IEE712I CONFIG PROCESSING COMPLETE
RMF Paging Activity Report

<table>
<thead>
<tr>
<th>VIEW</th>
<th>D10.FLASH.R70.RMF.OUTPUT(SCM0719) - 01.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command</td>
<td>---</td>
</tr>
<tr>
<td>Top of Data</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>001515</td>
<td>1</td>
</tr>
<tr>
<td>PAGING ACTIVITY</td>
<td>-</td>
</tr>
<tr>
<td>1514 Line(s) not Displayed</td>
<td>-</td>
</tr>
</tbody>
</table>

| 001516 | 001517 | 001518 | 001519 |
| z/OS VIR13 | SYSTEM ID R7F | DATE 07/19/2012 | INTERVAL 15.00.00 |
| RPT VERSION VIR13 RMF | TIME 16.00.00 | CYCLE 1.000 SECONDS |

| 001520 | 001521 | 001522 | 001523 | 001524 | 001525 | 001526 |
| MEMORY OBJECTS | COMMON | SHARED | 1 MB | 65 | 0 | 4 |
| 001527 | 001528 | 001529 | 001530 | 001531 | 001532 | 001533 | 001534 | 001535 | 001536 | 001537 | 001538 | 001539 | 001540 |
| 01 MB FRAMES | TOTAL | FIXED | AVAILABLE | IN-USE | TOTAL | AVAILABLE | IN-USE | 016402.1M | 1.295 | 0 | 0 | 17301594 | 267,190 | 3,194,870 | 258,137 | 438,688 |
| MIN | MAX | AVG | 1247484K | MEMORY OBJECTS AND HIGH VIRTUAL STORAGE FRAMES |

152 Line(s) not Displayed
### RMF Paging Activity Report (cont)...

#### Top of Data

<table>
<thead>
<tr>
<th>View</th>
<th>Command</th>
<th>Columns 00001 00124</th>
<th>Scroll -&gt; CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>001104</td>
<td>-PAGE</td>
<td>1545 Line(s) not Displayed</td>
<td></td>
</tr>
<tr>
<td>001546</td>
<td>NUMBER OF SAMPLES = 900</td>
<td></td>
<td></td>
</tr>
<tr>
<td>001547</td>
<td>PAGE DATA SET AND SCM USAGE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Bottom of Data

<table>
<thead>
<tr>
<th>VIEW</th>
<th>Command</th>
<th>Columns 00001 00124</th>
<th>Scroll -&gt; CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>001548</td>
<td>OSCM</td>
<td>134 Line(s) not Displayed</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PAGE</th>
<th>VOLUME</th>
<th>DEV</th>
<th>DEVICE</th>
<th>SPACE</th>
<th>SERIAL</th>
<th>NUM TYPE</th>
<th>ALLLOC</th>
<th>MIN</th>
<th>MAX</th>
<th>AVG</th>
<th>SLOTS</th>
<th>USE</th>
<th>TIME</th>
<th>10 REQ</th>
<th>XFER'D</th>
<th>D</th>
<th>DATA SET NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>001549</td>
<td>-PLPA</td>
<td>R7FPG1</td>
<td>221D</td>
<td>33903</td>
<td>35999</td>
<td>13951</td>
<td>13951</td>
<td>13951</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>SYS1.R7F.PLPA</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001550</td>
<td>OCOMMON</td>
<td>R7FPG1</td>
<td>221D</td>
<td>33903</td>
<td>35999</td>
<td>1945</td>
<td>1945</td>
<td>1945</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>SYS1.R7F.COMMON</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001551</td>
<td>OLOCAL</td>
<td>R7FPG2</td>
<td>222F</td>
<td>33903</td>
<td>59799</td>
<td>17028</td>
<td>17051</td>
<td>17030</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>43</td>
<td>SYS1.R7F.LOCAL</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001552</td>
<td>OLOCAL</td>
<td>R7FPG3</td>
<td>2287</td>
<td>33903</td>
<td>59799</td>
<td>15839</td>
<td>15855</td>
<td>15846</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>82</td>
<td>SYS1.R7F.LOCAL1</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001553</td>
<td>OLOCAL</td>
<td>R7FPG4</td>
<td>228F</td>
<td>33903</td>
<td>59399</td>
<td>16231</td>
<td>16236</td>
<td>16233</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>36</td>
<td>SYS1.R7F.LOCAL2</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001554</td>
<td>OLOCAL</td>
<td>R7FPG5</td>
<td>229F</td>
<td>33903</td>
<td>59399</td>
<td>16255</td>
<td>16264</td>
<td>16257</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>49</td>
<td>SYS1.R7F.LOCAL3</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001555</td>
<td>OLOCAL</td>
<td>R7FPG6</td>
<td>2250</td>
<td>33909</td>
<td>1620K</td>
<td>14622</td>
<td>14530</td>
<td>14526</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>47</td>
<td>SYS1.R7F.LOCAL4</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001556</td>
<td>OLOCAL</td>
<td>R7FPG7</td>
<td>2250</td>
<td>33909</td>
<td>1620K</td>
<td>14622</td>
<td>14530</td>
<td>14526</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>55</td>
<td>SYS1.R7F.LOCAL4</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001557</td>
<td>OLOCAL</td>
<td>R7FPG8</td>
<td>2250</td>
<td>33909</td>
<td>1620K</td>
<td>14622</td>
<td>14530</td>
<td>14526</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>47</td>
<td>SYS1.R7F.LOCAL4</td>
<td>Sys1.R7F.Local</td>
<td></td>
</tr>
<tr>
<td>001558</td>
<td>OSCM</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>8389K</td>
<td>385374</td>
<td>385033</td>
<td>385679</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0</td>
<td>1,234</td>
<td>1,229</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
RMF Monitor II – Page Data Set Activity

<table>
<thead>
<tr>
<th>S</th>
<th>VOLUME</th>
<th>DEV</th>
<th>%SLOTS</th>
<th>PAGE</th>
<th>I/O REQ</th>
<th>AVG PAGES</th>
<th>08:56:43</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>SERIAL</td>
<td>NUM</td>
<td>TYPE</td>
<td>IN USE</td>
<td>TRAN</td>
<td>TIME</td>
<td>RATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>PAGCOM</td>
<td>02E6</td>
<td>33903</td>
<td>100.0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>C</td>
<td>PAGCOM</td>
<td>02E6</td>
<td>33903</td>
<td>60.31</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>L</td>
<td>PAGEP2</td>
<td>098E</td>
<td>33903</td>
<td>0.00</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>L</td>
<td>PAGEP3</td>
<td>0987</td>
<td>33903</td>
<td>0.00</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>L</td>
<td>PAGEP4</td>
<td>098F</td>
<td>33903</td>
<td>0.00</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>L</td>
<td>PAGEO8</td>
<td>0981</td>
<td>33903</td>
<td>0.00</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>L</td>
<td>PAGETD</td>
<td>02E5</td>
<td>33903</td>
<td>0.00</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>L</td>
<td>PAGE99</td>
<td>0484</td>
<td>33903</td>
<td>0.00</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>S</td>
<td>N/A</td>
<td>N/A</td>
<td>4.74</td>
<td>0.000</td>
<td>0.310</td>
<td>166.667</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Command ====> _ Scroll ====> PAGE
F1=HELP F2=SPLIT F3=END F4=RETURN F5=RFIND F6=SORT
F7=UP F8=DOWN F9=SWAP F10=LEFT F11=RIGHT F12=RETRIEVE

Connected to remote server host vmttool1.pok.ibm.com using port 23
uspokl4z-S02-LL-707-Poughkeepsie on uspokl4z
SVC Dump Statistics

• VERBX IEAVTSFS

• Shows total dump capture time, system/task non-dispatch time, page operations required to dump requested address space (real-to-real copies, page-ins, etc)
### SVC Dump Statistics (cont)

<table>
<thead>
<tr>
<th>Description</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dump start</td>
<td>10/09/2012 14:30:29.867495</td>
</tr>
<tr>
<td>Dump end</td>
<td>10/09/2012 14:30:44.224584</td>
</tr>
<tr>
<td><strong>Total dump capture time</strong></td>
<td><strong>00:00:14.357089</strong></td>
</tr>
<tr>
<td>System nondispatchability start</td>
<td>10/09/2012 14:30:29.870030</td>
</tr>
<tr>
<td>System set nondispatchable</td>
<td>10/09/2012 14:30:29.870048</td>
</tr>
<tr>
<td>Time to become nondispatchable</td>
<td>00:00:00.000017</td>
</tr>
</tbody>
</table>
SVC Dump Statistics (cont)

Asid 0071:

- Local storage start: 10/09/2012 14:30:30.424083
- Local storage end: 10/09/2012 14:30:43.011936
- Local storage capture time: 00:00:12.587853
- Tasks reset dispatchable: 10/09/2012 14:30:43.011944
- Tasks were nondispatchable: 00:00:12.587861

- Defers for frame availability: 0
- Pages requiring input I/O: 170196
- Source page copied to target: 16987
- Source frames re-assigned: 566614
- Source AUX slot IDs re-assigned: 15749
Flash Express Performance Results

- All performance information was determined in a controlled environment.
- Actual results may vary.
- Performance information is provided “AS IS” and no warranties or guarantees are expressed or implied by IBM.
Flash Express Performance Test Setup

- z/OS Tests were designed to demonstrate flash performance under paging workloads that are typically encountered in a z/OS enterprise environment
  - SSD performance is not only about the number of IOPS but about steady performance over time and consistent latency
    - **Preconditioned SSDs** with random-write IO engage the device’s wear leveling, error handling, and flash management algorithms
  - Comparison DASD Characteristics used **current device configurations**
    - DS8800 model 2107-951
    - 60 GB cache, cache hit rates of 95-100% were observed during the tests
    - DASD was not shared with any other systems and did not have any I/O traffic other than the paging traffic used for these tests
    - Configured 16 local page datasets spread across 8 LCUs
Flash Express Performance Benefits

Test Results

- **FLASH paging benefits**
  - Improved availability through faster paging at critical times
  - Faster workload transitions (e.g., morning startup)
    - *meaning less time to reach peak transaction rates*
  - Faster SVC dumps (reduced **non-dispatchable** time)
    - *meaning higher availability – more transactions can be run*

- **Pageable Large Page benefit**
  - Java realizes performance benefits from use of large 1MB pageable pages
    - Large pages benefits for JIT Code Cache, 31 bit Java applications
    - No authorization needed to access fixed large pages
    - Approximately 5-8% CPU improvement from PLP
Workload Configuration Block Diagram
Building block – A WAS instance accessing CICS and DB2

Each WAS instance has a WAS Control Region and 3 WAS Servant Regions.
Each WAS Control Region has a 0.5GB heap plus a JIT Code cache.
Each WAS Servant Region has a 2GB heap plus a JIT Code Cache.

Test Configuration
- WAS 7 (3 servants each with two GB heap) + 1 control region (.5 GB Heap)
- CICS V4.2, DB210 on a zEC12
- Storage: DS8800 2107-951 with 60GB cache, very fast device
- Tests simulated morning transition time typical of trading or call center work
- SVC dump measurements were taken for an 18 GB dump.

2-book zEC12
I. Morning Transition
Transition from night batch to OLTP

WAS workload to CICS and DB2 represents OLTP work which is then stopped.
Simulated overnight work consumes real storage pushing other pages out.

~21GB is pushed out to Auxiliary storage.
Morning Transition - Transition from night batch to OLTP

The “Night Work” is then stopped and OLTP work is started (WAS 1 and WAS 2).
Measure the time needed to bring the OLTP work to full speed.

~14GB is paged back in from Auxiliary storage.

Workload Transition
Morning Transition - Results

- During morning transition, workloads using Flash Express reached peak throughput in under 1/4th the time.

- Paging to DASD required about 44 seconds for the workload to reach steady state.

- Paging to Flash required only 10 seconds for the workload to reach steady state.
### Morning Transition - Results Apparent in First 45 Seconds

<table>
<thead>
<tr>
<th>Transaction completion &amp; response time</th>
<th>DASD</th>
<th>Flash</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Transactions within first 45 seconds</td>
<td>251</td>
<td>343</td>
<td>37% increase</td>
</tr>
<tr>
<td>Average response time within first 45 seconds</td>
<td>0.62</td>
<td>0.06</td>
<td>90% reduction</td>
</tr>
</tbody>
</table>

Units in seconds

- Paging to Flash Express during morning transition showed up to a 10 times faster response time and up to a 37% increase in throughput within the first 45 seconds

(1) Test was for the first 45 seconds of morning transition time
II. SVC Dump

SVC dump with pages out

Three of four WAS instances were active. One WAS instance was stopped and most pages were paged out.
Capture an SVC dump of WAS instance 3 and 4, and DB2. Measure the capture time for the SVC dump.
SVC Dump - Results

Flash Express SVC dump elapsed time was up to 25% shorter

<table>
<thead>
<tr>
<th>SVC Dump Metrics</th>
<th>DASD</th>
<th>Flash</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVC Dump size (in bytes):</td>
<td>18GB</td>
<td>18GB</td>
</tr>
<tr>
<td>% of pages from Aux storage:</td>
<td>50%</td>
<td>53%</td>
</tr>
<tr>
<td>DUMP Elapsed time:</td>
<td>189</td>
<td>143</td>
</tr>
<tr>
<td>Max address space non-dispatchable seconds</td>
<td>58.89</td>
<td>13.74</td>
</tr>
<tr>
<td>System non-dispatchable seconds</td>
<td>1.34</td>
<td>0.55</td>
</tr>
</tbody>
</table>

Let’s graph these results….
SVC Dump - Results

- In SVC dump test, steady state performance was achieved up to 4 times faster *

Focus on first 90 seconds.

* Transaction steady state was reached in 14 seconds with Flash Express, vs. 60 seconds DASD
III. WAS and Java

z/OS Java SDK 7:16-Way Performance Shows up to 60% Improvement
64-bit Java Multi-threaded Benchmark on 16-Way

Aggregate 60% improvement from zEC12 and Java7SR3

- zEC12 offers a ~45% improvement over z196 running the Java Multi-Threaded Benchmark
- Java7SR3 offers an additional ~13% improvement (-Xaggressive + Flash Express pageable 1Meg large pages)
WAS benchmark: z/OS Performance for Pageable Large Pages

- The WAS Day Trader benchmarks showed up to an 8% performance improvement using Flash Express.

<table>
<thead>
<tr>
<th>Java 7 SR3</th>
<th>JIT</th>
<th>Java Heap</th>
<th>Multi Threaded</th>
<th>WAS Day Trader 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31 bit</td>
<td>yes</td>
<td>yes</td>
<td></td>
<td>4%</td>
</tr>
<tr>
<td>64 bit</td>
<td>yes</td>
<td></td>
<td>1%</td>
<td></td>
</tr>
<tr>
<td>64 bit</td>
<td></td>
<td>yes</td>
<td>4%</td>
<td></td>
</tr>
</tbody>
</table>

* WAS Day Trader 64-bit Java 7 SR3 with JIT code cache & Java Heap

**DETAILS**
- **64-bit Java heap** (1M fixed large pages (FLPs) or 1M Pageable (PLPs)) versus 4k pages
  - Java heap 1M PLPs improve performance by about
    - 4% for Multi-Threaded workload
    - 5% for WAS Day Trader 2.0
- **64-bit Java 7 SR3 with JIT code cache** 1M PLPs vs without Flash
  - 3% improvement for traditional WAS Day Trader 2.0*
  - 1% improvement for Java Multi-Threaded workload
- **31-bit Java 7 SR3 with JIT code cache and Java heap** 1M PLPs vs without Flash
  - 4% improvement for Java Multi-Threaded workload

* Note: This test used 64-bit Java 7 SR3 with JIT code cache & Java Heap leveraging Flash and pageable large pages. Also, tests used WAS Day Trader app that supports PLP; earlier version of 31-bit Java did not allocate 1M large pages.
Performance Summary for Flash Express

WORKLOAD TRANSITION

- During morning transition, workloads using Flash Express reached peak throughput in under 1/4th the time.
- Paging to Flash Express during morning transition showed up to a 10 times faster response time and up to a 37% increase in throughput within the first 45 seconds.

WAS JAVA PERFORMANCE BENCHMARKS

- The WAS Day Trader benchmarks showed up to an 8% performance improvement using Flash Express.

Improved Availability During Diagnostics

- In SVC dumps, availability was up to 4 times higher for workloads and up to twice as high for systems.
- In SVC dump tests, steady state performance was achieved up to 4 times faster.
- Flash Express SVC dump elapsed time was up to 25% shorter.

* Transaction steady state was reached in 14 seconds with Flash Express, vs. 60 seconds DASD.

DB2

- Up to 28% improvement in DB2™ throughput due to faster CPU and leveraging Flash Express with Pageable Large Pages (PLP).
- Workloads leveraging Flash Express with PLP can see up to a 8% price performance improvement over the z196.

* PLP for DB2 helps DB2 to achieve “additional” up to 3% additional performance on top of zEC12 CPU expected throughput improvements of 25%.
** based on average 5% discount for zEC12 workloads under the AWLC pricing plus up to 3% more performance per MSU with Flash Express.

(1) All tests are comparing the use of Flash Express as compared to using DASD (DS8800)
(2) System non dispatchability and address space non dispatchability time were dramatically reduced enabling work to be processed that would otherwise have been stopped.
z/OS Flash Roadmap
Flash Express Exploitation

*Flash support in z/OS sets the stage for further use*

- **Flash Express announced**
  - 4Q12
    - IBM zEnterprise® EC12 (zEC12) 9/19
    - z/OS support for zEC12

- **1Q13**
  - z/OS V1.13, enabling PTF
  - z/OS RSM Enablement Offering
    - Flash Dynamic Reconfiguration
    - Optional PLPA & Common dataset definition

- **z/OS V1.13 Flash Web Deliverable December 14th 2012**

- **Linux* on System z (native)**

**Planned Flash Express and pageable large page exploiters:**
- DB2 for z/OS
- Java SDK7
- WAS Liberty Profile v8.5
- IMS™ 12
- z/OS V1.13 Language Environment®
- Other (CICS®)

Expect continued middleware exploitation for 1MB pageable large pages
Flash Express Implementation

✓ System Overview
✓ Redundant Physical Structures
✓ Data Protection Mechanisms
✓ Data and Key Encryption
✓ Non-Disruptive Service Techniques
FLASH Express System Overview

- **LPAR 1....n**: SSCH, ORB, AOB, AIDAWs, Data, Data
- **SAP 1....n**: Firmware Management of Adapter, Subchannel, Device control, EADMF control
- **System Bus**: Data Transfer, Control
- **I/O Hub**: PCIe Switch, Storage Class Memory, LPAR1 Increments, RAID Controller, Firmware
- **Flash Express Cards**: PCIe IO Drawer, RAID Controller, Firmware
- **Support Element**: CEC
Redundant Physical Structures

Redundant SE's for configuration and key service
Redundant Physical Structures

- LPAR 1....n
- SSCH ORB
- AOB
- AIDAWs
- Data
- Data

- SAP 1....n
- Firmware Management
- Subchannel
- Device control
- IOP/HSA

- I/O Drawer Connections

- System Bus
- PCIe Switch

- I/O Hub
- Control

- PCIe

- Storage Class Memory
- LPAR1 Increments

- RAID Controller
  - Firmware

Flash Express Cards

- PCIe IO Drawer
Redundant Physical Structures

- Self-Mirrored Adapters w/ dual mirror cables
Data Protection Mechanisms

- Comm links (SAS, PCIe) provide embedded protection & recovery
- CEC-based hardware address protection on communication from adapter
- ECC on internal system memory

RAID10: Protection and performance
- RAID0 = Striping
- RAID1 = Mirroring
- RAID10 = Striped mirrorred data

- CRC and block seq. number stored on SSD
- Additional CRC around block transfer
Data and Key Encryption

✓ On SSD, data is protected with inline encryption (hidden encryptKey)

✓ Access to SSD is via authentication key (authKey) served from SE

During Flash install, in smart card on SE:
- Create authKey (aka PIN)
- Wrap authKey in an encrypted file
- wrapKey stored in smart card
- Wrapped key file stored on SE

SE → CEC-FW authkey service:
- asymmetric protocol – pub/private
- IOP sends public key to SE
- In smart card, Key file unwrapped then encrypted with CEC pubKey
- Encrypted authKey sent to CEC
- CEC 'unwraps' authKey using its privKey

✓ AuthKey used during SSD format and subsequent power cycles
Non-disruptive Service Strategy

✓ Firmware updates
  » Adapter
  » SSD
  » Expanders
  » CEC-FW

✓ Adapter replacement

✓ Cable replacement

✓ Recoveries
  » CEC
  » Adapter
  » SSD
THANK YOU