Configuring z/OS to Ensure Successful DASD Swap
Using the CRITICALPAGING Function

Potential Problem Scenario

During a DASD Swap, using Basic HyperSwap, GDPS HyperSwap Manager or other swap technologies, a system may require access to a page that is currently paged out. In order to resolve the page fault, I/O to a paging DASD device is required. If the page device is among the devices that are frozen/suspended during the DASD Swap, the page fault will not be immediately resolved. The page fault will be resolved when I/O to the device is resumed. If the page fault is not resolved in a timely fashion the DASD Swap may fail due to a timeout. Depending on the specific page that took the page fault, and on the impacted code path, a system may be removed from the sysplex during a DASD Swap.

This may result in a system being removed during a DASD Swap. Depending on the SFM policy, SFM may remove the system. Alternatively, in a GDPS HyperSwap Manager environment, GDPS may remove a system if it fails to respond to a phase of the HyperSwap.

While it may be most likely that a page fault occurs in XCFAS in this scenario, it is possible a similar result may occur for any address space critical to DASD Swap.

CRITICALPAGING

IBM strongly recommends enabling the CRITICALPAGING function to significantly reduce the possibility that address spaces critical to DASD Swap completion will require page fault resolution. The set of declared critical system address spaces for DASD Swap are: RASP (RSM), GRS, CONSOLE, XCFAS. Also, the address spaces associated with Basic HyperSwap in base (HSIB), Basic HyperSwap API (HSIBAPI), and GDPS HyperSwap Communication Task (often jobname GEOXCFST) are declared critical. The page fault avoidance is accomplished by RSM “hardening” the pages associated with declared critical address spaces. This is accomplished by making the pages in these address spaces least likely to be evicted from real memory and paged out to an auxiliary DASD device by z/OS’s page replacement algorithms. This includes storage for the critical address spaces, dataspaces and scroll hiperspaces associated with critical address spaces, 31-bit common and PLPA storage.

By simply enabling the CRITICALPAGING function the IBM address spaces associated with Basic HyperSwap and GDPS HyperSwap Manager will automatically receive the benefit of CriticalPaging. For other swap technologies, you may need to create program properties table (PPT) entries for each of address spaces which need to be “hardened.”

The RSM, IOS and XCF components each provide a piece of the new CRITICALPAGING function which is included in z/OS 1.12. PTFs are available zOS 1.10 and 1.11 in Table 1.
Table 1. z/OS releases, APARs for each component and associated PTFs which must be active on a system in order to use the CRITICALPAGING support.

The PTFs for all components are required to be on a system to enable the function. The PTFs and the function can be enabled one system at a time.

Enabling CRITICALPAGING

Explicit system programmer actions are required to enable the CRITICALPAGING function.

A system programmer should do a real storage assessment for each system prior to enabling CRITICALPAGING. The goal of the assessment is to ensure no undue burden is placed upon other system address spaces or applications as a result of activating CRITICALPAGING. First, review the RMF STORF reports to determine the peak auxiliary slot count for each critical address space. Figure 1 has an example for XCFAS.
Next, review the RMF VSTOR report to determine the size of PLPA and CSA.

<table>
<thead>
<tr>
<th>NUMBER OF SAMPLE</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATIC STORAGE MAP</td>
<td></td>
</tr>
<tr>
<td>AREA</td>
<td>ADDRESS</td>
</tr>
<tr>
<td>EPVT</td>
<td>21200000</td>
</tr>
<tr>
<td>ECSA</td>
<td>E51E000</td>
</tr>
<tr>
<td>EMLPA</td>
<td>0</td>
</tr>
<tr>
<td>EFLPA</td>
<td>E51B000</td>
</tr>
<tr>
<td>EPLPA</td>
<td>A4FC000</td>
</tr>
<tr>
<td>ESLPA</td>
<td>E520000</td>
</tr>
<tr>
<td>ENUC</td>
<td>10000000</td>
</tr>
<tr>
<td>---- 16 MEG BOUNDARY ----</td>
<td></td>
</tr>
<tr>
<td>NUCLEUS</td>
<td>FD4000</td>
</tr>
<tr>
<td>SQA</td>
<td>E53000</td>
</tr>
<tr>
<td>PLPA</td>
<td>C940000</td>
</tr>
<tr>
<td>FLPKA</td>
<td>C93000</td>
</tr>
<tr>
<td>MLPA</td>
<td>0</td>
</tr>
<tr>
<td>CSA</td>
<td>9000000</td>
</tr>
<tr>
<td>PRIVATE</td>
<td>2000</td>
</tr>
<tr>
<td>PSA</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 2. RMF VSTOR sample output, highlighted values for CSA and PLPA.

A very simple guideline is to increase the memory on the system by the sum of the peak auxiliary counts for all of the declared critical address spaces multiplied by 4K, plus the amounts for PLPA+EPLPA and CSA+ECSA. Additional consideration can be given if the system “never” pages according to RMF reports. If a system “never” pages then it may be acceptable to add little or no storage to the system and receive the full benefit of CRITICALPAGING support. Similarly, additional consideration should be given to a system which currently pages heavily. If the system pages heavily then it may very well be the case that pages are paged out from DASD Swap critical address spaces regardless of their specification in SCHDExx or being a system default specified address space. This is done in an attempt to avoid further impact to the system by a critical central storage shortage. If pages are stolen from critical DASD Swap address spaces the system becomes exposed to the problem scenario.

The CRITICALPAGING function is NOT active by default. Overt action by the system programmer must be taken to enable the CRITICALPAGING function. To activate the function update COUPLExx to contain the following CRITICALPAGING statement on each system in the relevant sysplex.

FUNCTIONS ENABLE (CRITICALPAGING)

Note: It is not possible to enable the CRITICALPAGING function using the SETXCF command, an IPL is required. When the system is IPLed MSGIXC373I will be issued indicating CRITICALPAGING is enabled.
The output of D XCF,COUPLE will also indicate CRITICALPAGING is enabled, MSGIXC357I.

Enabling the CRITICALPAGING function will ensure the system defined address spaces critical for Basic HyperSwap or GDPS HyperSwap or Freeze functions are hardened. Thus, enabling CRITICALPAGING will harden RASP (RSM), GRS, CONSOLE, XCFAS. Also, the address spaces associated with Basic HyperSwap in base (HSIB), Basic HyperSwap API (HSIBAPI), and GDPS HyperSwap Communication Task (often GEOXCFST jobname). No further action is required to harden these spaces.

For other swap technologies you may need to create PPT entries. Additional programs can be declared in a user defined SCHEDxx parmlib member. An example of a user PPT entry:

```
/**************************************************************************/
/*                                                                     */
/* ADD NEW PPT ENTRY FOR PROGRAM NAME, jobA                           */
/*     TURN ON CRITICALPAGING                                         */
/*                                                                     */
/**************************************************************************/
PPT PGMNAME(jobA)          /* PROGRAM NAME FOR jobA            */
CANCEL                   /* PROGRAM IS CANCELLABLE           */
KEY(0)                   /* PROGRAM IS TO HAVE THE PROTECTION */
NOSWAP                   /* PROGRAM IS NON-SWAPPABLE          */
PRIV                     /* PROGRAM IS PRIVILEGED AND IS MANAGED */
DSI                      /* PROGRAM REQUIRES DATASET INTEGRITY */
PASS                     /* PROGRAM CANNOT BYPASS SECURITY   */
SYST                     /* PROGRAM IS A SYSTEM TASK AND IS NOT */
AFF(NONE)                /* PROGRAM HAS NO PROCESSOR AFFINITY */
```
NOPREF /* PROGRAM DOES NOT NEED TO HAVE ALL PRIVATE AREA SHORT-TERM FIXED PAGES ASSIGNED TO PREFERRED STORAGE
OTHER OPTIONS ARE:
* SPREF - MUST HAVE ALL PRIVATE AREA SHORT-TERM FIXED PAGES ASSIGNED TO PREFERRED STORAGE
* LPREF - MUST HAVE ALL PRIVATE AREA LONG-TERM FIXED PAGES ASSIGNED TO PREFERRED STORAGE */

HONORIEFUSIREGION /* PROGRAM SHOULD HONOR REGION AND MEMLIMIT REQUESTS FROM EXIT IEFUSI */

CRITICALPAGING /* PROGRAM IS CRITICAL TO DASD SWAP OPERATION AND SHOULD NOT HAVE ITS STORAGE PAGED OUT (IF POSSIBLE) */

It is possible to dynamically update the address spaces for which CRITICALPAGING is declared. Modify the SCHEDxx and then SET SCH=xx. MSGIEF731I will be issued indicating CRITICALPAGING has been activated or ignored for the program.

IEF731I SCHEDxx LINE num: PPT STMT [FOR PGMNAME name] [IGNORED. | ACCEPTED.] REASON = kwrc.

kw: 22
CRITICALPAGING

kw: 23
NOCRITICALPAGING

rc: 48
The keyword cannot be specified with this program name. For example, you can not specify the CRITICALPAGING / NOCRITICALPAGING keyword with any of the program names associated with address spaces deemed critical by IBM for Basic HyperSwap or GDPS HyperSwap or Freeze functions.

rc: 52
The CRITICALPAGING keyword was specified with a program whose associated storage is not all paged in. To ensure that a program's associated storage is all paged in after dynamically changing its properties, the program should be ended and restarted, if possible.
Disabling CRITICALPAGING

It is possible to disable CRITICALPAGING on a per-system basis using the SETXCF command:

SETXCF FUNCTIONS,DISABLE=(CRITICALPAGING)

The FUNCTIONS ENABLE(CRITICALPAGING) statement must be removed from COUPLExx to avoid re-enabling CRITICALPAGING during IPL.

It is NOT recommended to disable the CRITICALPAGING function in a DASD Swap environment.

Conditions Under Which Critical Storage Will be Paged Out

If the function is disabled, traditional paging algorithms will apply to the previously declared critical address spaces. MSGIXC373I will be issued.

IXC373I XCF / XES OPTIONAL FUNCTIONS DISABLED:
CRITICALPAGING

If the set of CRITICALPAGING address spaces changes on a system where CRITICALPAGING is enabled via the SET SCH=xx, then all pages currently paged in will be hardened by RSM. Any pages associated with the address space already paged out will NOT be proactively paged in by RSM. If the paged out pages are referenced, they will be hardened by RSM when they are successfully paged in.

After the CRITICALPAGING is enabled the STORF report may indicate RSM has auxiliary slots. This does not necessarily mean that critical pages were stolen from RSM. RSM’s component trace dataspace can be paged out. Also, RSM has slot ownership of other areas such as 31-bit shared pages that can still be paged out.

If the system is experiencing a real frame shortage, the system might still page out storage associated with the critical address spaces, to avoid a severe system impacting issue or system outage.

Conclusion

When used correctly, enablement of the CRITICALPAGING function will significantly reduce the likelihood of experiencing DASD Swap hang or failure or system outages due to inaccessible paged-out storage during a DASD Swap.