Rule WLM400: Page-in from auxiliary storage was a major cause of delay

Finding: CPExpert has determined that waiting for page-in from auxiliary storage was a major cause of the service class not achieving its performance goal.

Impact: This finding can have a LOW IMPACT, MEDIUM IMPACT, or HIGH IMPACT on performance of your computer system. The impact of this finding depends upon the percent of time transactions in the service class were waiting for pages from auxiliary storage. A high percent waiting for pages means HIGH IMPACT while a low percent waiting for pages means LOW IMPACT.

Please note that the percentages reported by CPExpert are computed as a function of **the active time of the transactions**, rather than percentages of RMF measurement interval time. The percentages show the impact of page-in delay **on the transactions**, rather than the impact of page-in from an overall system view. This data presentation approach is significant when the service class being delayed is a **server** service class; the page-in delays represent delays to the response times of the served transaction!

Logic flow: The following rules cause this rule to be invoked:

Rule WLM101:	Service Class did not achieve average response goal
Rule WLM102:	Service Class did not achieve percentile response goal
Rule WLM103:	Service Class did not achieve execution velocity goal
Rule WLM104:	Subsystem Service Class did not achieve average response goal
Rule WLM105:	Subsystem Service Class did not achieve percentile response goal
Rule WLM150:	Server Service Class delays
Rule WLM151:	Server Service Class delays

Discussion: The MVS virtual storage environment operates on the principles that:

- The central storage required by any particular address space during execution is a subset of the total central storage required to load the address space. Much of the storage required to load an individual address space is often unused. This storage that **is** regularly used is referred to as the "working set" of the address space. The working set is typically a small part of the overall central storage requirement to initially load an address space. The remaining (typically large) amount of central storage can be used by other address spaces loaded concurrently.

- Idle central storage should be used to prevent unnecessary I/O operations. In fact, central storage generally should be managed to maximize the use of central storage while minimizing I/O operations.

- Central storage can be "allocated" to address spaces based upon the importance of the address space. That is, the "working set" of a low priority address space can be constrained if necessary to allow more important address spaces to have adequate central storage.

- With appropriate external storage and process controls, users of a virtual environment should notice little difference between the performance of the virtual environment and the performance of a non-virtual environment.

Exchanging pages between central storage and auxiliary storage (and between central storage and expanded storage if expanded storage is installed) is the way MVS allows multiple address spaces to concurrently use a finite amount of central storage. When an address space requires a page of storage that has been removed from central storage, a "page fault" occurs. The address space (actually, the TCB or SRB associated with the address space) is unable to continue processing until the page fault is resolved. MVS will locate the page and bring it into central storage.

- The page might actually be in central storage waiting a page-out operation. These pages are "reclaimed" and made available without further effort. No statistics are maintained on the number of reclaimed pages, but this number normally should be small.

- The page may be in expanded storage (for systems with expanded storage). These pages are moved directly from expanded storage; the page-in time is very small (various studies have reported page-in times from expanded storage in the range of 40-75 microseconds). Delays for these page operations do not generally cause a performance problem[1].

- If the page is not in central or expanded storage, the page must be physically brought in from auxiliary storage. It is these page-in operations that Rule WLM400 addresses.

If the page is in expanded storage or in auxiliary storage, a page frame in central storage must be available to hold the page being paged in. The Real Storage Manager normally maintains a number of "available" page frames in central storage to accommodate the page.

The time from the page fault until the required page is available is considered page delay time. During this time, the address space requiring

[1]While delays for page-in operations from expanded storage does not normally cause problems, there are some situations in which the page-in rate from expanded storage can seriously degrade performance. The Workload Manager will monitor and potentially manage service classes or address spaces that experience or cause a high paging rate from expanded storage.

the page normally must wait. During the waiting time, the central storage associated with the address space is wasted for the page delay time. Additionally, other resources allocated to the address space are unusable during the page fault resolution time.

The page delay time may have other, potentially more serious, implications. If the address space is associated with a response-critical application (e.g., a TSO trivial transaction), end-user response will be delayed for the time required to resolve the page fault. If many page faults occur, response may degrade to less than the performance goals for the service class.

The SMF Type 72 records contain information that can be analyzed to determine the amount of delay a service class experienced as a result of page-in operations from auxiliary storage. The page-in delay from auxiliary storage is separately reported in the following delay categories:

- **Private area page-in from auxiliary storage delay**. This delay category means that the address space was experiencing page faults in the private area and the pages were coming from auxiliary storage.

- **Common area page-in from auxiliary storage delay**. This delay category means that the address space was experiencing page faults in the Common area and the pages were coming from auxiliary storage.

- **Cross-memory page-in from auxiliary storage delay**. This delay category means that the address space was experiencing page faults in cross-memory access and the pages were coming from auxiliary storage.

- **VIO page-in from auxiliary storage delay**. This delay category means that the address space was experiencing page faults in VIO and the pages were coming from auxiliary storage.

- **Standard hiperspace page-in from auxiliary storage delay**. This delay category means that the address space was experiencing page faults in standard hiperspace and the pages were coming from auxiliary storage.

- **ESO hiperspace page-in from auxiliary storage delay**. IBM has defined this state to mean that the address space was experiencing page faults in ESO hiperspace and the pages were coming from auxiliary storage. Pages in ESO hiperspace are, by definition, resident only in expanded storage (ESO = Expanded Storage Only), and are never migrated to auxiliary storage. IBM offers the following explanation[2]:

[2]IBM TALKLink RMF FORUM appended at 15:39:18 on 95/05/29 GMT (by YOCOM at KGNVMC)
Subject: Workload Activity Report

"The execution delay for ESO hiperspaces is a calculated value based on the assumption that if an application does a read for an ESO hiperspace page and that page is no longer available (has been cast out), the application will read the data from DASD somewhere. WLM/SRM takes the number of times a read failed in this way and multiplies it by the number of delay samples we expect a read of a page from DASD to represent and report the product as the execution delay samples for ESO hiperspace. This obviously is not a perfect solution, but we needed some way to get an estimate of how much delay is caused to an address space by not having enough expanded for an ESO hiperspace. Such an estimated is needed to properly manage the amount of expanded owned by the address space to the address space's goal."

CPExpert sums the page-in delays for all delay categories. CPExpert produces Rule WLM400 if the total page-in delay from auxiliary storage was a major reason the service class identified in the predecessor rules did not meet its performance goal.

The following example illustrates the output from Rule WLM400:

```
RULE WLM400: PAGE-IN FROM AUXILIARY STORAGE WAS MAJOR PERFORMANCE PROBLEM

   Page-in from auxiliary storage was a primary or secondary reason BATCH
   (Period 1) missed its performance goal.  Auxiliary storage paging caused
   the following delays to BATCH (Period 1), shown by category of page-in:

                           PERCENT
                           PAGE-IN    --PERCENT DELAY BY PAGE-IN CATEGORY--
   MEASUREMENT INTERVAL     DELAY     PVT    COMM    XMEM    VIO    HIPR     ESO
   15:00-15:16,01MAR1994     9.2      0.0     0.0     9.2    0.0     0.0     0.0
```

Suggestion: Page-in delays can be reduced in two basic ways: (1) reduce the time to resolve page faults and (2) reduce the number of page faults.

If the total page-in from auxiliary storage delay is unacceptable, CPExpert recommends that the following actions be considered:

- **Make sure that the paging configuration is optimal**. Review the recommendations in Section 2 of the MVS Initialization and Tuning Guide. CPExpert may produce rules in the WLM050(series) to identify potential problems in the paging configuration. The most common problem has been that installations allocate too few local page data sets.

- **Review performance goals and importance**. The Workload Manager will attempt to manage system resources (CPU and processor storage) to meet the performance goals of important workloads. You should make

sure that the performance goals and importance levels have been properly specified (1) for service classes with more restrictive performance goals or (2) for service classes at higher level or same level goal importance.

- **Reschedule the workload**. Schedule lower priority workloads to a time when they do not compete with critical applications. The Workload Manager will often swap out lower priority workloads to reduce page-in delay for higher priority workloads. However, the Workload Manager may require some elapsed time to identify the problem and take action. Depending upon the dynamics of the workload mix, the Workload Manager may not be as successful as would be manual rescheduling.

- **Ignore the finding**. You may decide that the service class experiencing page-in delays from auxiliary storage is insufficiently important to worry about. The BATCH service class in the example output could be an example of this; you might not worry that batch workload periodically experiences page-in delays and the BATCH service class misses its performance goal.

  You can exclude service classes from analysis[3] by CPExpert if this situation occurs regularly and becomes an annoyance.

- **Acquire additional processor storage**. Page faults occur because the required page is not available in central storage. You may be able to reduce page faults by acquiring additional central storage. Alternatively, you may consider acquiring additional expanded storage, since page fault resolution from expanded storage is extremely fast.

  Acquiring additional processor storage might not reduce page-in delays in some environments. Depending upon the nature of the applications, adding additional central or expanded storage might not have a noticeable effect.

- **Acquire faster paging devices**. If the above options have been exhausted and paging delays are still unacceptable, you should consider acquiring faster paging devices.

[3]Use the EXCLUDE guidance in USOURCE(WLMGUIDE) to exclude service classes from analysis.

