Rule WLM363: Non-paging DASD wait time was a major cause of DASD delay

Finding: CPExpert has determined that non-paging DASD wait time was a major cause of delay in DASD response for the I/O operations of the service class.

Impact: This finding may have a MEDIUM IMPACT or HIGH IMPACT on the performance of the service class. The finding applies only with OS/390 Release 3 and subsequent versions.

Logic flow: The following rule causes this rule to be invoked:
Rule WLM361: Non-paging DASD I/O activity caused significant delays

Discussion: Non-paging DASD I/O time is the time from the issuance of the SSCH instruction until the device is selected by the control unit. This time is caused by queuing for the path (wait for channel, wait for control unit or wait for head-of-string), and can be caused by other systems sharing the device (wait for device).

CPExpert examines the non-paging DASD I/O wait time contained in SMF Type 72 records (field R723CIWT). CPExpert produces Rule WLM363 if the non-paging DASD I/O wait time accounted for a significant percent of the non-paging DASD I/O for the service class missing its performance goal.

If the service class missing its performance goal is a transaction service class (for example, composed of CICS 4.1 transactions), CPExpert will identify the server service class (for example, the CICS region). CPExpert will then analyze the DASD I/O times for the server.

The following example illustrates the output from Rule WLM363:

```
RULE WLM363: NON-PAGING DASD WAIT TIME WAS A MAJOR CAUSE OF DASD DELAYS

   TSO: A major part of the DASD I/O delay to the service class
   is attributed to non-paging DASD wait (DASD PEND time and control unit
   queue time).  Please refer to the WLM Component User Manual for advice
   on how to minimize DASD wait time.
```

Suggestion: Large device PEND times usually involve the following situations:

- **Shared devices**. If the device is shared with another system, PEND time may indicate contention with the other system. Large PEND times in shared-device environments usually involve situations very similar to those described under IOSQ time:

  - **Multiple data sets active on the volume**. This situation is the most common and easiest to solve. The data sets can be redistributed among different volumes, to eliminate the queuing at the channel level (reflected as PEND time) for the single volume.

    If some of the data sets are not required to be shared, then the Data Base Administrator has complete flexibility to move these data sets (subject, of course, to the performance implications of the target devices). These data sets should be moved to a non-shared device.

    If the data sets are required to be shared, then they must be relocated to shared devices.

  - **Multiple applications or users using the same data set on the volume**. Depending upon the data set characteristics, duplicate copies of the data set may be placed on different volumes. This would solve the PEND problems cause by contending systems. If this option is feasible, the data sets could be placed on non-shared devices, likely resulting in even more performance improvement.

  - **Multiple application systems may be using the volume experiencing high PEND times**. In this case, perhaps application redesign or scheduling can solve the problem.

  Additionally, large PEND times for shared devices could be caused by RESERVE from the other system. The applications issuing the RESERVE should be examined to determine whether the RESERVE is required. If the RESERVE **is** required, the above situations should reviewed to determine whether improvements can be achieved.

- **Non-shared devices**. Large PEND times for devices that are not shared may mean that there are insufficient paths available to the device. Too much I/O may be directed to many devices on the path, control unit, or head-of-string. The data sets can be redistributed among different volumes on different paths, control units, or heads-of-string. This will reduce the hardware-level queuing. Alternatively, the entire volume may be moved to a different (less busy) head-of-string or path.

  If redistributing the data sets or moving the volume is not feasible, then the device should have more paths. Depending upon the existing

configuration, this may involve re-configuring existing channel paths, or acquiring additional hardware.

- **Devices attached to cached controllers**. Large PEND times for devices attached to cached controllers may imply a high percent of read miss operations, or non-volatile storage (NVS) writes for IBM-3990-3 devices. Fairchild [1] lists four ways in which staging in caching controllers can cause hidden device busy (with the device busy potentially reflected in high PEND time):

  - The normal (random) caching algorithm stages all records to the end of the track after a requested record is read.

  - The normal (random) caching algorithm stages all records from the beginning of the track to the requested record if a front-end miss occurs.

  - Most writes to extended function IBM-3990 (Model 3) go into NVS with a subsequent destaging required.

  - The sequential caching algorithm stages all records to the end of the track after the requested record is read, and stages in all of the next track. IBM-3990 (Model 3) controllers stages in all of the next three tracks.

- **Dual Copy Initialize**. Large PEND times for IBM-3390 devices may be caused by dual copy initialize. In this case, the dual copy initialize should be turned off.

[1] Fairchild, Bill, "The Anatomy of an I/O Request", *Conference Proceedings*, CMG'90, the Computer Measurement Group, Chicago, IL.

