
Rule WLM071: Detected Wait swaps occur too often

Finding: CPExpert has determined that an excessive number of Detected Wait swaps occurred during the measurement interval.

Impact: This finding can have a LOW IMPACT or MEDIUM IMPACT on performance of your computer system. The level of impact depends upon how many Detected Wait swaps occurred and upon your management objectives.

Logic flow: The following rule causes this rule to be invoked:
WLM450: Swap-in delay helped cause service class to miss goal

Discussion: Detected Wait swaps occur because the SRM detects that a resident transaction has not been dispatchable for two seconds of real time or eight SRM seconds, without issuing the WAIT, LONG=YES macro. (Eight SRM seconds represent about 320 milliseconds on an IBM-3090/600S single image or 865 milliseconds on an IBM-3084Q. Consequently, the eight SRM seconds will cause a detected wait on most of today's systems.) Detected Wait swaps usually are caused by cross memory services, applications that treats the terminal as SYSIN or SYSPRINT, teleprocessing applications (e.g., test CICS regions) that are not marked non-swappable, etc.

Additionally, STIMER wait values of less than the 0.5 seconds required to trigger a Long Wait swap may trigger a Detected Wait swap if the wait time is more than 8 SRM seconds (this situation is unlikely, but could occur on IBM-3090/180 and above).

The basic purpose of Detected Wait swaps is to detect address spaces which are using central storage, but which are not dispatchable. If the address spaces are not dispatchable, perhaps their central storage should be made available for other purposes.

From this definition of Detected Wait swaps, these swaps generally should be a fairly small percentage of the overall swaps, unless TSO is used as a gateway to some other application. However, Detected Wait swaps commonly account for almost 5% of the total swaps, and sometimes account for over 30% of the total swaps. A large percent of Detected Wait swaps should be investigated, since these applications tie up real storage before the SRM detects that they have not been dispatchable.

CPEXpert produces Rule WLM071 if swap-in delay was a significant factor in causing a service class to miss its performance goal, if Detected Wait swaps accounted for a significant percent of the total swaps, and if the number of Detected Wait swaps exceeded the **SWAPDW** guidance variable in USOURCE(WLMGUIDE).

Suggestion: CPEXpert recommends that you examine the specific applications causing the Detected Wait swaps. These applications are not dispatchable for an extended period of time, but do not advise the SRM so that they can be swapped out when they become non-dispatchable. They occupy storage for the period between when they become non-dispatchable and the SRM detects this status. This central storage could be added to the Available Frame Queue and made available to executing address spaces.

Alternatively, you may wish to review whether the address spaces should be swappable. Under some conditions, it may be better to make the address spaces non-swappable and reduce the Detected Wait swaps.

You can determine the workload experiencing the Detected Wait swaps by examining the RMF Monitor II "Address Space State Data Report". This report shows the reason for the last swap out for the jobs.

- You can select class, status, and domain. You should use the command **ASD(A,A)** to request a report of all address spaces that are currently executing, non-swappable, or swapped out and eligible for swap-in.
- Review the reason for last swap out. This information will be displayed in the "R LS" column. The reason codes are shown in the RMF; search for reason "DW". These identify address spaces experiencing Detected Wait swaps.

Detected Wait swaps seem to be application dependent, with little guidance available in the literature about how to prevent Detected Wait swaps.

We would appreciate your feed-back regarding the causes and cures of Detected Wait swaps in different applications! We can make this information generally available to users of CPEXpert.