Rule WLM025: The service class period was inactive

Finding: The service class period was inactive for a significant percent of the RMF measurement intervals being analyzed.

Impact: This finding should be viewed as generally having a LOW IMPACT or MEDIUM IMPACT on the performance of the Workload Manager and its ability to manage system resources to meet your performance goals.

Logic flow: This a basic finding. There are no predecessor rules.

Discussion: The Workload Manager generates control blocks to describe each service class period defined in the service policy. These control blocks describe the performance goal, the importance of the goal, the resource group (if any) to which the service class is assigned, etc.

Additionally, the Workload Manager generates control blocks for each service class period to describe the resources used by the service class period, delays to the service class period, etc.

The Workload Manager periodically examines the SRM control blocks describing each address space and acquires samples[1] describing the state of each dispatchable unit of an address space (that is, each TCB or SRB associated with the address space). The Workload Manager accumulates the samples into counters which describe the state of the address space[2]. The samples are summarized by service class period.

The Workload Manager uses the samples in two basic ways: (1) the samples of CPU using and delays are used in computing the execution velocity of service class periods[3], and (2) the samples in various categories are used to guide the actions that the Workload Manager might take as it attempts to reallocate resources to meet specified performance goals.

---

[1]With MVS/ESA SP5.1 Goal Mode, the sampling is done every 250 milliseconds. The sampling interval is recorded in SMF Type 72 records (R723MTVL).

[2]Address spaces can have more than one dispatchable unit (that is, more than one TCB or SRB). If an address space did have more than one dispatchable unit, the Workload Manager would accumulate state samples to describe the state of each dispatchable unit. To simplify the discussion, we shall refer to the sample states as being the "state" of the address space, with readers understanding that multiple dispatchable units could simultaneously be waiting on multiple events, or could be using the processor. Thus, for any single sampling, an address space could be counted in more than one state.

[3]Beginning with OS/390 (Release 3), the Workload Manager calculates execution velocity based on Total Using Samples divided by (Total Using Samples plus Total Delay Samples).

The underlying concept of a sampling process and the analysis of the samples obtained requires that a fairly large number of samples be taken. Basically, sampling normally requires sufficient samples such that the samples can be viewed as representative of the "population" being sampled.

If few address spaces or dispatchable units (TCBs, SRBs, or enclaves) are assigned to a service class, there will be few samples taken. Additionally, if those address spaces or dispatchable units that **are** assigned are primarily idle, few *meaningful* samples will be taken.

Please refer to Section 4 (Chapter 3) for additional information regarding the sampling process, the variables sampled, and how the Workload Manager uses the samples.

CPExpert examines each SMF Type 72 (Goal) record to determine whether sufficient samples are available for analysis. CPExpert simply compares the number of meaningful samples with the **MINSAMP** guidance variable contained in USOURCE(WLMGUIDE). CPExpert retains information for any service class period, by RMF interval, in which the number of meaningful samples is less than the MINSAMP guidance variable. After all RMF records have been analyzed, CPExpert determines whether any service class period was inactive (that is, whether few meaningful samples were available for analysis).

CPExpert produces Rule WLM025 when any service class period was inactive for more than 75% of the RMF intervals, and when this condition was true for all systems in the sysplex being analyzed. CPExpert ignores service class periods with discretionary goals, and ignores system service classes.

The following example illustrates the output from Rule WLM025:

```
RULE WLM025:  THE SERVICE CLASS PERIOD MAY BE UNNECESSARY

  STC_LOW (Period 1): This service class period was inactive for over
  75% of the RMF measurement intervals being analyzed (inactive means
  that less than 100 using and delay samples were collected by the SRM
  for the service class period during the RMF interval). Since the service
  class period was mostly inactive, you should re-evaluate whether this
  service class period is necessary.  Defining unnecessary service class
  periods (1) causes more overhead by the Workload Manager and (2) may
  cause the Workload Manager to be less responsive in managing system
  resources to meet your performance goals.  Please refer to Rule WLM025
  in the WLM Component User Manual for a discussion of the reasons for
  this recommendation.
```

Suggestion: If only a few meaningful samples are taken related to a service class period, the question arises as to why the service class period is in the service definition. There can be a **significant** performance impact of defining a large number of service class periods! Please consider the following:

- The Workload Manager makes its decisions mostly based on samples collected from service class periods executing in the sysplex. If few samples are taken for a particular service class period, the Workload Manager cannot validly make decisions regarding the service class period.

- The Workload Manager will adjust system resource allocation in an attempt to improve performance of only one service class period during a policy adjustment interval[4]. Adjustment to improve performance of only one service class period is done because the Workload Manager must observe the results of the adjustment: whether the adjustment helped performance, hurt performance, or had no effect. If adjustments were made to improve performance of more than one service class period, it would be impossible to determine which adjustment helped or hurt which service class period. Consequently, actions to improve the performance of only one service class period is done during each policy adjustment interval.

  The policy adjustment interval is 10 seconds. If too many service class periods have been defined, the Workload Manager may never be able to adjust system resource allocation to help more than a few service class periods (the most important service class periods with the worst performance). Performance of other service class periods may never be improved, or performance improvement actions may take a long time - simply because of the elapsed time necessary for the Workload Manager to make changes, collect data, analyze the effect of the changes, make additional changes, collect more data, and continue the process.

- Perhaps of equal significance is the overhead associated with analyzing service class periods. The Workload Manager tries to improve performance of the service class period with the worst performance at the highest goal importance. Resources may be taken from the least important service class period with the best performance. The Workload

[4]The Workload Manager will consider adjustments to improve performance of several service classes (starting with the most important service class which has the highest Performance Index). If performance of the first service class analyzed cannot be appreciably improved, the Workload Manager will select the next worst performing service class at the highest importance, etc. After the Workload Manager has "committed" to a policy adjustment for a service class, it will stop analysis and adjust resources for no other service class.

Manager will not simply remove and add resources; rather, the Workload Manager will analyze the net value of the planned action.

The Workload Manager will not add resources unless there is an appreciable net gain to the service class period receiving the resources. Within the same goal importance, the Workload Manager will not remove resources from a service class period unless the net gain to the receiver outweighs the net loss to the service class period the resources are being removed from. The overhead involved with the analysis and decision process increases as the number of service class periods becomes large.

- IBM SRM/WLM developers have indicated that a small number of service class periods is desirable. They have observed that the Workload Manager algorithms typically become increasingly ineffective as the number of service class periods grows. As "rule of thumb" guidance, the developers have stated that most users should define no more than 25-30 service class periods.

From a practical matter, service class periods with a discretionary goal do not cause concern. This is because the Workload Manager groups all service class periods with a discretionary goal (and not assigned to a resource group) into the **$SRMDI00** internal service class, and treats them as a single service class.

CPExpert suggests that you re-evaluate whether the service class period is necessary. Unless unusual circumstances exist, CPExpert suggests that the service class period be removed from the service definition, and the work units reassigned to an existing service class.

You may wish to assign the work units to a report class, if you are concerned about reporting information related to the work units.

Alternatively, you can "turn off" CPExpert's analysis of inactive service class periods (and suppress this rule completely) by specifying **%LET INACTIVE = N;** in USOURCE(WLMGUIDE).

Reference: MVS Planning: Workload Management

MVS/ESA(SP 5):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V1R1):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V1R2):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V1R3):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V2R4):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V2R5):	Chapter 8: Defining Service Classes and Performance Goals

OS/390 (V2R6):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V2R7):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V2R8):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V2R9):	Chapter 8: Defining Service Classes and Performance Goals
OS/390 (V2R10):	Chapter 8: Defining Service Classes and Performance Goals
z/OS (V1R1):	Chapter 8: Defining Service Classes and Performance Goals
z/OS (V1R2):	Chapter 8: Defining Service Classes and Performance Goals
z/OS (V1R3):	Chapter 8: Defining Service Classes and Performance Goals
z/OS (V1R4):	Chapter 8: Defining Service Classes and Performance Goals

Thanks: Thanks to Glenn Bowman (Wakefern Food Corporation, NJ) for suggesting this rule.

