
Rule WLM010: Velocity goals have values which are too similar

Finding: CPExpert noticed that two or more service classes have been defined with an execution velocity goal and the Goal Importance was the same for both service classes, but that the actual values for the execution velocity goals were numerically similar. The Workload Manager cannot usually differentiate among execution velocities which are numerically similar.

Impact: This finding should be viewed as generally having a MEDIUM IMPACT on the performance of the service classes involved.

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

Discussion: Installations may specify an *execution velocity goal* for a service class period. An execution velocity is a measure of how fast work should run when the work is ready to run, without being delayed waiting for access to a CPU or delayed waiting for access to processor storage¹. The purpose of specifying an execution velocity goal is to allow installations to specify how important it is to have work processed, when the work has no time-related measure (that is, a response requirement is not associated with the work).

The execution velocity is computed based on samples collected at periodic sampling intervals² by the System Resources Manager (SRM). The SRM sampling code interrogates address space control blocks (TCBs, SRBs, OUCBs, and OUXBs) to determine the state of each address space assigned to a service class. Sampling counts associated with the service class are updated based upon the state³ of the address spaces.

Please refer to Rule WLM104 for additional discussion of an execution velocity goal.

The sampling process takes samples every 250 milliseconds, resulting in 4 samples per second. The Workload Manager analyzes the results of the samples during the policy adjustment interval, which occurs every 10

¹Processor storage is composed of *central storage* and *expanded storage*. The third category of storage is *auxiliary storage*.

²With MVS/ESA SP5.1, the sampling interval is 250 milliseconds. The state of each TCB or SRB associated with an address space is sampled every 250 milliseconds, beginning from address space initiation.

³Note that an address space can be in multiple states (for example, a CICS region might be using multiple processors concurrently, while some CICS tasks were also waiting on some function). Thus, the sample counts can total more than 100% of the sample intervals for any address space.

seconds. Consequently, the Workload Manager makes its policy adjustment decisions related to execution velocity, based on only 40 samples (4 samples per second * 10 seconds = 40 samples).

The margin of error based on 40 samples is sufficiently large that the process cannot differentiate between execution velocities which are numerically similar. If the execution velocity goals are numerically similar, the sampling error may be greater than the difference among the execution velocity goals.

IBM suggests that at least 10 should separate the goals (e.g., 10, 20, 30, etc.) for the Workload Manager to manage the system resources to meet the different execution velocity goals⁴.

It is unfortunate that the *RMF Workload Activity Report* produced in Compatibility Mode lists execution velocity achieved as "xx.x" (for example, 61.3 execution velocity). Such a "precise" listing of the computed execution velocity misleads users into believing that the Workload Manager can manage system resources to achieve such detailed goals. Consequently, many early users of MVS/ESA SP5 (Goal Mode) have defined execution velocity goals with numerically similar values.

CPEXpert produces Rule WLM010 when execution velocity goals are specified with less than 10 separating the goals, and the goals have the same Goal Importance.

The following example illustrates the output from Rule WLM010:

Suggestion: CPEXpert suggests that you review the execution velocity goals specified for the service classes listed with Rule WLM010. You should revise the goals so that at least 10 separates each goal, or you should recognize that the Workload Manager cannot manage system resources to differentiate among the specified goals.

⁴This advice applies only to velocity goals with the same Goal Importance. During Policy Adjustment analysis, the Workload Manager examines service class periods in order of the Goal Importance. The service class with the highest Goal Importance is examined first. If the service classes have a different Goal Importance, the Goal Importance will provide the discrimination between the service classes and the execution velocity goals can be numerically similar without worry. Please see Section 4 (Chapter 3.5) for a discussion of the Policy Adjustment process.

RULE WLM010: VELOCITY GOALS HAVE VALUES WHICH ARE TOO SIMILAR.

CPEXPERT noticed that two or more service classes at the same Goal Importance have been defined with an execution velocity goal, but that the actual values for the execution velocity goals were numerically similar. The Workload Manager cannot usually differentiate among execution velocities which are numerically similar. At least 10 should separate the goals (e.g., 10, 20, 30, etc.) for the Workload Manager to manage the system resources to meet the different goals. This finding applies to the following service classes:

SERVICE CLASS	EXECUTION VELOCITY	SERVICE CLASS	EXECUTION VELOCITY
APPCFEED	50	ASCH	55
APPCFEED	50	ST_TOOLS	57
ASCH	55	MONITORS	50
ASCH	55	ST_TOOLS	57
ASCH	55	TPNSFEED	50
MONITORS	50	ST_TOOLS	57
ST_TOOLS	57	TPNSFEED	50

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

"Migrating to the MVS Workload Manager", Peter Enrico (IBM Corporation Workload Manager developer), 1995 SHARE Winter Meeting

"MVS Workload Manager Velocity Goals: What you don't know can hurt you", John Arwe, IBM Corporation, CMG'96 Proceedings.