
Rule WLM105: Subsystem Service Class did not achieve percentile response goal

Finding: CPExpert has detected that a service class did not achieve the percentile response goal that was specified in the service policy in effect. This finding applies to performance goals that specify **percentile response time** as the performance goal. Additionally, this finding applies to service classes that are part of a subsystem (e.g., CICS transactions). This finding is made only if subsystems are installed that support Workload Manager reporting (e.g., CICS/ESA Version 4.1 or later, and IMS/ESA Version 5 or later).

Impact: This finding can have a HIGH IMPACT on performance of your computer system.

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

Discussion: If subsystems are installed that support Workload Manager reporting (e.g., CICS/ESA Version 4.1 or IMS/ESA Version 5), installations can define service classes that describe particular transaction types and specify performance goals for the transactions in the service class. All transactions entering the system that fall into the workload category described by the service class are associated with the service class.

For example, an installation may wish to group all CICS transactions relating to personnel matters into a CICSPERS Service Class. The installation would define classification rules to the Workload Manager so all transactions relating to personnel matters would be placed into the CICSPERS Service Class. The installation would specify a performance goal for the CICSPERS Service Class, and an importance level for the goal.

Notice that the **transactions** comprising the CICSPERS Service Class must actually execute in a CICS region executing CICS at a level of at least CICS/ESA Version 4.1. The CICS region would report transaction performance information to the Workload Manager, and the Workload Manager would attempt to manage system resources to meet the performance goal specified for the CICSPERS Service Class.

The controlling address space must be in its own service class. In our example, suppose that the CICS region is placed into the CICSRRGN Service Class. The CICSRRGN Service Class would be considered a "server" and the CICSPERS Service Class may be one of several "served" transaction service classes controlled by the CICSRRGN Service Class

(other CICS transaction service classes "served" by the CICSRRGN "server" may be related to procurement, administration, miscellaneous, etc.).

The CICSRRGN will have its own performance goals and importance. However, these performance goals and importance are used by the Workload Manager **only at address space start-up** time. After the CICS region has started, its performance goals and importance are ignored by the Workload Manager. The Workload Manager will allocate resources based upon the performance goals and importance of the "served" transaction service classes (in our example, the allocation will be based upon the performance of the CICSRRGN transactions, and other "served" service classes served by the CICSRRGN Service Class).

It is important to appreciate that the Workload Manager **does not** allocate resources to the CICSRRGN Service Class, as CICSRRGN is simply a logical entity that describes transactions and CICSRRGN is not an address space. Rather, the Workload Manager allocates resources to the "server" address space (the CICSRRGN Service Class). Similarly, the Workload Manager does not measure resources consumed by the CICSRRGN Service Class, as CICS does not report this information to the Workload Manager.

One implication of the structure of the "server" and "served" service classes is that the Workload Manager will attempt to meet the performance goals of all "served" transaction service classes that are served by the "server" service class. It does this by allocating resources to the "server" service class. **These additional resources may (or may not) be used to provide service to the transaction service class missing its goal¹.**

Suppose there are multiple "served" transaction service classes associated with a "server" service class. If some "served" transaction service class is failing to achieve its goal, the Workload Manager may allocate additional resources to the "server" service class. These additional resources might allow some "served" service classes to significantly exceed their performance goal and these "served" service classes may not be particularly important.

In our example, suppose that the CICSRRGN Service Class is serving two transaction service classes (the CICSRRGN Service Class we described and a CICSRRGN Service Class). Suppose that CICSRRGN is important but that CICSRRGN Service Class is of lower importance. If the Workload Manager detects that CICSRRGN is not meeting its performance goal, the Workload Manager may allocate more resources to the CICSRRGN Service Class. The CICSRRGN would use the additional resources to provide

¹Please refer to Section 4 for a more complete illustration of the "server" and "served" concepts.

service to both CICSPERS and CICSADMN. Consequently, CICSADMN might significantly exceed its performance goal. Indeed, there is no guarantee that the additional resources would help CICSPERS unless CICSPERS had been properly **defined to CICS** as a higher priority than CICSADMN.

To summarize this discussion, performance goals are associated with "served" transaction service classes while resources are allocated to "server" service classes. Performance (i.e., transaction response time) is recorded at the "served" transaction service class level, while resource use is recorded at the "server" service class level.

Service classes can be defined that have a "percentile" response performance goal. A "percentile" response performance goal means that the performance goal is defined as "x%" of the transactions should complete within "y" time. For example, a typical percentile response goal is that **90% of the transactions should complete within 200 milliseconds**.

This rule (Rule WLM105) deals with performance goals that have been specified as a **percentile response goal** (e.g., "x%" of the transactions should complete within "y" time). Rule WLM104 deals with performance goals for subsystem service classes that have an **average** response goal.

MVS accounts for each transaction executing in the system and determines the transaction's response time². MVS maintains fourteen counters for each service class that has a response goal. The counters represent a response time distribution with respect to the response goal.

For response goals, RMF includes in SMF Type 72 records a count of transactions that completed in varying percentages of the response goal. These transaction counts are recorded by RMF as the "Response Time Distribution Count Table" contained in SMF Type 72(Subtype 3) records³.

The Workload Manager periodically assesses the performance of each service class, comparing the performance achieved by the service class against the performance goals specified for the service class. This assessment is referred to as the "policy adjustment" interval, in that the Workload Manager decides whether to adjust resource policies based on whether service classes are meeting performance goals.

²This response time applies only to the time the transaction was in the system; it does not apply to response time delays experienced in the network.

³Please refer to Exhibit 4-11 in Section 4 for a description of the response time distributions.

For service classes that have a **percentile response time goal**, the Workload Manager determines whether the specified percent of transactions were achieving the response time specified by the response goal for the service class. If more than the specified percent of transactions achieved a response greater than the specified response goal, the system was not meeting performance goals for the service class period. If the importance of the service class is sufficiently high, the Workload Manager may re-allocate system resources in an attempt to meet performance goals.

CPEXpert analyzes the SMF Type 72 records to determine whether service class periods met their performance goals during each RMF measurement interval. For service class periods that have a percentile response performance goal specified, the performance goal is specified as "**x%** of the transactions completing within **y** time." CPEXpert simply sums the transaction count in the first six counters to determine the number of transactions ending within 100% or less of the response goal. This value is divided by the total number of transactions ending to yield the percent of transactions ending within 100% or less of the response goal. If the resulting percentage is less than the performance goal percentage, CPEXpert can conclude that the performance goal was not met.

CPEXpert produces Rule WLM105 when CPEXpert detects that a service class period did not meet its percentile response goal for an entire RMF measurement interval. CPEXpert reports the total transactions that ended during the interval, the number of transactions that met the response goal, the percentage of transactions that met the goal, and the primary and secondary causes of response delay. Additionally, CPEXpert computes the contribution that the primary and secondary causes of delay made to the average transaction response time.

For example, suppose that an installation specified that 90% of the transactions should complete within 100 milliseconds for a service class period serving CICS transactions. CPEXpert might detect that only 80% of the transactions completed within 100 milliseconds, and the performance goal was not achieved. CPEXpert would report the number of ending transactions, the number of transactions that met the 100 millisecond goal, and that only 80% of the transactions met the goal.

CPEXpert would analyze the causes of delay to CICS transactions and report the primary and secondary causes of delay, **if the information is available**. Some subsystems may not provide detailed information about causes of delay⁴. If this case, CPEXpert simply lists "data not available" under the primary and secondary causes of delay column.

⁴Early releases of IMS Version 5 did not correctly report transaction delays.

The subsystem work manager (e.g., CICS) normally reports the causes of delay to the Workload Manager, using the Workload Management Services macros⁵.

CICS reports two separate views of the transactions: the *begin_to_end phase* state and the *execution phase*. IMS reports only *execution phase*.

- **Begin_to_end phase.** The *begin_to_end phase* starts when CICS has classified the transaction⁶. This action normally is done in a CICS TOR region.
- **Execution phase.** The *execution phase* starts when either CICS or IMS has started an application task to process the transaction. For CICS, this normally is done in a CICS AOR region. For IMS, this is the IMS Message Processing Region (MPR).

Some CICS transactions may never enter the *execution phase*, as the transactions will be completely processed in the CICS TOR. Consequently, the number of transactions completing the *execution phase* may be less than the total number of CICS transactions processed by the system.

In our example of CICS transactions, the CICS subsystem work manager would report transaction delays in the following states for the "served" service class:

- **Active state.** The *active state* indicates that there was a program executing on behalf of the work request in the "served" transaction service class, from the perspective of the work manager. In the case of a CICS region, this means that a CICS task has been dispatched by CICS to process the transaction.

However, the *active state* **does not mean that the task is executing** from the perspective of MVS. It simply means that the task has been dispatched by CICS. Other address spaces with a higher system dispatching priority could preempt the task dispatched by CICS and these other address spaces could be using the CPU. The situation in which the CICS application task is denied use of the CPU is unknown to CICS⁷.

⁵Please refer to Section 4 (Chapter 2.2) for a description of the interaction between subsystems and the Workload Manager.

⁶Classifying the transaction into a service class is actually done by the Workload Manager when CICS issues the IWMCLSY macro. Please refer to Section 4 for a more complete discussion of the subsystem work manager (e.g., CICS) interaction with the Workload Manager.

⁷The "denied CPU" state will be reported by the SRM in the CICS RGN service class, since the SRM samples control blocks for the CICS address space.

-
- **Ready state.** The ready state indicates that there was a program ready to execute on behalf of a work request in the "served" service class, but that the work manager has given priority to another work request. In the case of a CICS region, this means that there were more CICS tasks ready to process transactions in the "served" transaction service class than were dispatched by CICS.
 - **Idle state.** The idle state indicates that there were no work requests (e.g., CICS transactions) ready to run in the service class.
 - **Waiting for lock.** The waiting for lock state indicates that some work request (e.g., a CICS task) was waiting for a lock.
 - **Waiting for I/O.** The waiting for I/O state indicates that the work manager was waiting for some I/O request on behalf of the "served" service class. This state could be waiting on an actual I/O operation or waiting on some other function related to the I/O request.
 - **Waiting for conversation.** The waiting for conversation state indicates that the work manager was waiting for a response in a conversation mode.
 - **Waiting for distributed request.** The waiting for distributed request state indicates that some function or data must be routed prior to resumption of the work request.
 - **Waiting for session to be established locally.** The waiting for session to be established locally means a wait for a session to be established on the current MVS image.
 - **Waiting for session to be established in sysplex.** The waiting for session to be established in sysplex means a wait for a session to be established somewhere in the sysplex.
 - **Waiting for session to be established in network.** The waiting for session to be established in network means a wait for a session to be established somewhere in the network.
 - **Waiting for timer.** The waiting for timer means that a work request was waiting for expiration of a timer.
 - **Waiting for another product.** The waiting for another product means that a work request was waiting for another product to provide some service.

-
- **Waiting for a new latch.** The waiting for a new latch means that a work request was waiting for a new latch. A latch is a short-duration lock.
 - **Waiting for SSL thread.** The waiting for SSL thread means that a work request was waiting for a Secure Sockets Layer thread.
 - **Waiting for regular thread.** The waiting for regular thread means that a work request was waiting for a regular thread.
 - **Waiting for work table.** The waiting for work table means that a work request was waiting for a work table registration.
 - **Waiting for unidentified resource.** The waiting for unidentified resource means that the work request was waiting, but that the work manager could not identify the cause of the wait.

The above causes of delay are analyzed by CPEXpert in other rules.

Additionally, CPEXpert could report that the “delay” was because the transaction was switched to a local MVS image, switched to another system in the sysplex, or switched to some system in the network.

C If the transaction was switched to a local MVS image, CPEXpert can perform further analysis on the information for the current system.

C If the transaction was switched to another system in the sysplex, CPEXpert will analyze other systems on which the service class appears. Information will be provided about delays to the service class on these other systems.

C If the transaction was switched to some system in the network, no information is available in the SMF data and no further analysis can be done.

The delays are recorded by RMF from two perspectives: (1) the *begin_to_end phase* of work requests in the service class and (2) the *execution phase* of work requests in the service class. CPEXpert can analyze delays to transactions from both perspectives⁸.

Additionally, some service classes might have *begin_to_end phase* data, but might **not** have *execution phase* data. In this case (and if the basic analysis is based on *execution phase* data), CPEXpert will indicate “NO EXE PHASE DATA” in the PRIMARY,SECONDARY CAUSES OF DELAY,

⁸A CPEXpert guidance variable (the **PHASE** variable) in USOURCE(WLMGUIDE) controls which phase CPEXpert initially analyzes. Please refer to Section 2 for a discussion of how the PHASE guidance variable may be used to direct CPEXpert's analysis and why this guidance may be altered.

and will provide information about the *begin_to_end* phase. Rule WLM116 provides information for this situation.

For SMF Type 72 records related to "server" service class (e.g., a CICS region), RMF records information identifying the service classes served by the server service class. This information is in the "Service Class Served Data Section" of the TYPE 72 records. If CPEXpert discovers that a "served" service class did not achieve its performance goal, CPEXpert identifies the "server" service classes that serve the service class not achieving its performance goal.

The following example illustrates the output from Rule WLM105:

```

RULE WLM105:  SERVICE CLASS DID NOT ACHIEVE PERCENTILE RESPONSE GOAL

Service Class CICADMTX did not achieve its response goal during the
measurement intervals shown below.  The response goal was 75.0 percent
of the transactions completing within 0.090 seconds, with an importance
level of 3.  CICADMTX was defined as a "served" Service Class (e.g.,
IMS or CICS transactions).  The below causes of delay were based upon
BEGIN_TO_END PHASE samples.  CICADMTX was served by CICSRGN.

-----LOACL SYSTEM-----
          TRANS
          %
MEASUREMENT INTERVAL  TOTAL  MEETING  MEETING  PERF  PLEX  PRIMARY,SECONDARY
                       TRANS  GOAL     GOAL    INDX  PI    CAUSES OF DELAY
13:02-13:07,21JUN1994  14,326  9,463    66.1   4.00  4.00  WAIT I/O(65%),READY(22%)
13:07-13:12,21JUN1994  14,307  8,709    60.9   4.00  4.00  WAIT I/O(52%),READY(35%)
13:12-13:17,21JUN1994  14,357  9,216    64.2   4.00  4.00  WAIT I/O(65%),READY(25%)
13:17-13:22,21JUN1994  14,314  8,669    60.6   4.00  4.00  WAIT I/O(40%),READY(51%)
13:22-13:27,21JUN1994  14,287  9,172    64.2   4.00  4.00  WAIT I/O(63%),READY(32%)
13:27-13:30,21JUN1994   8,612  5,639    65.5   4.00  4.00  WAIT I/O(65%),READY(29%)

```

The information associated with Rule WLM102 is shown based on data collected by the *local system*, which is the system being analyzed for performance purposes.

CPEXpert also computes and reports a *sysplex* Performance Index. The WLM maintains both a "sysplex Performance Index" and a "local system Performance Index." Briefly, the WLM first examines the sysplex Performance Index to determine whether a service class period is missing its performance goal and whether action should be taken. After the sysplex Performance Index is examined at a particular Goal Importance level, the WLM then examines the local system Performance Index. Rule WLM140 explains this WLM logic in more detail, and describes the implications of the WLM logic.

Recall that resources are allocated to "server" service classes, and these "server" service have information relating to resources used and relating to possible delays from a system view. After analyzing the information described above related to the "served" service class missing its performance goal, CPExpert analyzes the "server" service class to identify causes of delay from a system view.

In the example of Rule WLM105, CPExpert detected that the CICSADMTX service class did not achieve its performance goal. After analyzing the delays from the perspective of CICS, CPExpert will analyze the delays to the server (CICSRGN), from the perspective of the overall system.

Suggestion: There are no suggestions with this finding. CPExpert will continue analysis and other rules will be produced to provide more information.