Rule CIC104: Maximum Task (MXT) specification may be too large

Finding: CPExpert has detected that the value specified for the MXT operand in the System Initialization Table (SIT) may be too large.

Impact: This finding has a MEDIUM IMPACT on the performance of the CICS region. This finding applies only to CICS/ESA Version 3.1.1 and later versions.

Logic flow: This is a basic finding, based upon an analysis of the daily CICS statistics.

Discussion: The MXT operand in the System Initialization Table (SIT) limits the total number of concurrent tasks in the CICS region. Please refer to Rule CIC101 for additional information about the MXT operand.

Some installations have specified MXT=999, with the intention of allowing an unlimited number of tasks in storage. That is, the number of tasks will be limited only by the demands of CICS users. Additionally, specifying MXT=999 is recommended by IBM's CICS MOR Tuning and Performance Guide.

With CICS/TS Release 5.1, the MXT system initialization parameter has a default value of 500, a minimum setting of 10, and a maximum setting of 2000. For many installations, the default value of 500 is significantly higher than should be used.

Specifying a large value for MXT had little adverse effect before CICS/ESA Version 3.1.1. However, for CICS/ESA Version 3.1.1 and later versions, this specification can have an adverse effect on available storage and on CICS performance.

With CICS/ESA Version 3.1.1 and later versions, CICS uses the MXT parameter to determine the kernel storage allocation. This allocation is calculated in 4-K, 5K, and 8K byte units. (The number of units is derived based upon the value of the MXT operand.) The CICS Performance Guide explains that the kernel stack storage allocations are made as follows:

- Below the 16-megabyte line:

    - One 5K-byte segment for each task, up to the MXT limit, plus four, **and**

  - One 4K-byte segment for every 10 tasks (based on MXT, plus four, rounded down).

- Above the 16-megabyte line:

  - One 8K-byte segment for each task, up to the MXT limit, plus four, **and**

  - One 4K-byte segment for every 10 tasks (based on MXT, plus four, rounded down).

Thus, if you specify MXT=100 in the SIT, CICS requires the following storage:

- Below the line: (104 * 5K) + (10 * 4K) = 560K bytes

- Above the line: (104 * 8K) + (10 * 4K) = 872K bytes

Specifying MXT=100 might be a reasonable specification in many environments and the above calculation shows the storage required to support these tasks. However, if you specify MXT=999, the amount of storage is increased significantly!

- Below the line: (1003 * 5K) + (100 * 4K) = 5,415K bytes

- Above the line: (1003 * 8K) + (100 * 4K) = 8,424K bytes

Few installations would require such a high specification for MXT, and most installations would begin to experience CICS stress and short-on-storage conditions because so much storage was required for the kernel stacks. The space allocated for kernel stacks is **permanently** allocated - so MXT=999 cannot be specified just for initialization with the intent of lowering the value during operation. The storage will be allocated based upon the value CICS is provided during initialization, and remains permanently allocated.

If the maximum number of tasks specified on the MXT operand is too high for the available storage, CICS will not be able to obtain the amount of MVS storage it needs for kernel stack segments. In this case, CICS progressively reduces the MXT value until it is able to obtain the MVS storage that it needs. CICS reduces the MXT value using the following algorithm:

$$MXT = \frac{MXT - 32}{2} + 32$$

Having a large value for a Terminal Owning Region (TOR) is not usually detrimental unless the region is also an AOR region. TOR regions use their space simply for queuing, and thus generally do not experience short-on-storage conditions.

However, significant overhead in IBM's Workload Manager (WLM) can be caused by setting an unnecessary large value for MXT if the CICS region has a response goal. This is because the WLM will generate Performance Blocks for the MXT value. The WLM scans these Performance Blocks to obtain statistsics necessary for managing allocation of system resources. Excess Performance Blocks cause unnecessary overhead. This overhead can be significant when large number of performance blocks are generated.

CPExpert computes the percent of active tasks as a function of the value of the MXT keyword. When the percent of active tasks is less than the **PCTMXTLO** guidance variable, CPExpert concludes that the MXT value may be too high. Rule CIC104 is produced to report this conclusion.

If the condition exists for **any** CICS statistics interval in the data being analyzed for a region, CPExpert lists the percent of active tasks as a function of all intervals, and places '***' beside those intervals in which the conclusion was reached. This listing is done to give a sense of perspective of (1) how often and (2) when the condition was detected.

Initially, CPExpert arbitrarily produced Rule CIC104 if the number of active tasks was less than 75% of the MXT value. The **PCTMXTLO** value was introduced to allow users to specify guidance to CPExpert.

Suggestion: CPExpert suggests that you review the MXT value in the SIT and determine why it is specified so much higher than required by normal CICS operation. Unless there are unusual circumstances, you may wish to specify the MXT operand with a value more representative of the tasks actually processed by the region.

The MXT ceiling normally should not be reached unless management wishes to restrict access to the CICS region.

The value for the MXT operand normally should be sufficiently high that tasks are not restricted by MXT. Of course, if storage is a constraint, you might have to use the MXT operand to restrict access to CICS by tasks. However, be very careful not to allow the MXT value to be reached in an MRO environment (unless you have a very unusual environment).

Alternatively, you can adjust CPExpert's analysis by using the PCTMXTLO guidance to specify different guidance to CPExpert.

This rule probably should be ignored if the region is a TOR-only region and you are executing in Compatibility Mode. There usually is little harm done by allocating more kernel stacks than required in a TOR region for Compatibility Mode.

The rule **does** apply for a TOR if you are executing in Goal Mode, since unnecessary overhead would be generated by MVS in scanning Performance Blocks created for each possible task.

Reference: CICS/OS/VS Version 1.7 - Not applicable.

CICS/MVS Version 2.1.2 - Not applicable.

CICS/ESA Version 3.1.1 Performance Guide: page 53, page 295, and page 307.

CICS/ESA Version 3.2.1 Performance Guide: page 191, page 203, and page 271.

CICS/ESA Version 3.3.1 Performance Guide: page 201, page 221, and page 291.

CICS/ESA Version 4.1.1 Performance Guide: Section 4.7.3, Appendix A.1.4, and Appendix C.7

CICS/TS Release 1.1 Performance Guide: Section 4.7.3, Appendix 1.1.28, and Appendix 4.8.

CICS/TS Release 1.2 Performance Guide: Section 4.7.4, Appendix 1.1.4. and Appendix 1.1.29.

CICS/TS Release 1.3 Performance Guide: Section 4.11.3, Appendix 1.1.5. and Appendix 1.1.32.

CICS/TS for z/OS Release 2.1 *Performance Guide*: Chapter 23 (MXT) and Appendix A (Table 131).

CICS/TS for z/OS Release 2.2 *Performance Guide*: Section 4.10.3 Setting the maximum task specification (MXT)

CICS/TS for z/OS Release 2.3 *Performance Guide*: Section 4.10.3 Setting the maximum task specification (MXT)

CICS/TS for z/OS Release 3.1 *Performance Guide*: Section 4.10.3 Setting the maximum task specification (MXT)

CICS/TS for z/OS Release 3.2 *Performance Guide* (dfht3c00.pdf):
Chapter 19. Virtual and real storage: performance considerations
Setting the maximum task specification (MXT)

CICS/TS for z/OS Release 4.1 *Performance Guide* (SC34-7033):
Chapter 19. Virtual and real storage: performance considerations
Setting the maximum task specification (MXT)

CICS/TS for z/OS Release 4.2 *Performance Guide* (SC34-7177):
Chapter 4. CICS Transaction Manager: performance and tuning
Setting the maximum task specification (MXT)

CICS/TS for z/OS Release 5.1 Performance Guide (SC34-2864) |
Chapter 6. CICS Transaction Manager: performance and tuning |
Setting the maximum task specification (MXT) |

CICS MRO Tuning and Performance Guide: page 37.

IBMLINK, Document Q579023.

IBMLINK, Document Q465409.