Rule CIC262: Conflicting usage between APPC modegroups may exist

Finding: CPExpert believes that there may be conflicting usage between APPC (LU6.2) modegroups.

Impact: This finding should normally have a MEDIUM IMPACT on the performance of the CICS region. However, the finding could have a HIGH IMPACT on the performance of individual transactions if these transactions are queued for lengthy intervals.

Logic flow: This is a basic finding, based upon an analysis of the CICS statistics. Please refer to Rule CIC260 for a discussion of basic ISC/IRC concepts.

Discussion: Transactions acquire the use of a session in an ISC/IRC environment by using the ALLOCATE command. Conversations can take place between the two CICS regions or systems only after the session has been allocated. Once established, the session normally exists for a long time and can be used by many different transactions. The session normally is terminated by a FREE command.

With LU6.2, the ALLOCATE command may request allocation from a specific session modegroup (with its own characteristics), or may not request any particular session modegroup. If a specific session modegroup is requested, CICS will restrict its allocation attempt to that modegroup. If no particular session modegroup is requested, CICS will attempt to allocate a session from any modegroup, **but selects modegroups in the order in which the modegroups are defined**. This last point may be significant from a performance viewpoint.

A situation can occur in which the modegroup most often requested in a **specific** session allocation request has been defined as the first modegroup in the modegroup definitions. Sessions from this modegroup will be allocated whenever a **generic** (non-specific) session allocation request is received by CICS. If a number of generic session allocation requests were received, all available sessions in the modegroup could be allocated. When a **specific** allocation request for a session from the modegroup was received, CICS would be unable to honor the allocation request and would queue the request (suspending the transaction).

This example refers to the first modegroup only for illustration purposes. CICS selects modegroups in the order in which they are defined, so the discussion would apply to the second or subsequent modegroups if all sessions in the first modegroup were allocated.

©Copyright 1994, Computer Management Sciences, Inc.

CPExpert detects this potential problem when Peak Outstanding Allocates (A20ESTAM) is non-zero for a particular modegroup. This situation indicates that some allocation requests were queued for the modegroup. This, by itself, could indicate that insufficient sessions were defined for the modegroup and would be analyzed by Rule CIC260.

However, CPExpert also noticed that the number of **generic** allocation requests (A14ESTAG) was much larger than the total number of **specific** allocation requests (A20ESTAS) for all modegroups during the interval covered by the CICS statistics. These values **may** indicate that the allocation delays for specific modegroups results from sessions in the modegroup being used to satisfy generic allocation requests. If this is the case, simply reordering the modegroups may solve the problem.

Suggestion: CPExpert suggests that you consider reordering the definition of APPC modegroups.

Modegroups for a connection are represented by TCT mode entries (TCTMEs). The modegroup name is taken from the MODENAME specified in the SESSIONS definition. The order of the TCTMEs is determined by the order in which CICS installs the SESSIONS definition. CICS installs the SESSIONS definitions based on the order of the SESSIONS names stored in the CICS System Definition (CSD) file. The SESSIONS names are stored in ascending alphanumeric sequence. Consequently, you must change the names of the SESSIONS definitions to change the order of the TCTMEs.

You can use the CEDA RENAME command with the AS option to rename the definition with a different SESSIONS name within the CSD group. By managing the order in which the TCTMEs are created, you can ensure that **specific** allocations reference modegroups lower down the TCTME chain. This will avoid conflicts with the **generic** allocates.

Alternatively, you can make all allocates specific to a particular modegroup.

Reference: *CICS/ESA Version 3.1.1 Performance* Guide: pages 76-84.

CICS/ESA Version 3.2.1 Performance Guide: pages 294-301.

CICS/ESA Version 3.3.1 Performance Guide: page 58 and pages 313-320.

CICS/ESA Version 4.1.1 Performance Guide: Section 2.2.23 and Appendix A.1.13.

©Copyright 1994, Computer Management Sciences, Inc.

CICS/TS Release 1.1 Performance Guide: Section 2.2.23 and Appendix 1.1.14.

CICS/TS Release 1.2 Performance Guide: Section 2.2.24 and Appendix 1.1.14.

CICS/TS Release 1.3 Performance Guide: Section 2.2.25 and Appendix 1.1.15.

CICS/TS for z/OS Release 2.1 *Performance Guide*: Chapter 5 (ISC/IRC system and mode entry statistics) and Appendix A (Table 64).

CICS/TS for z/OS Release 2.2 *Performance Guide*: Section 2.2.27 (Interpreting ISC/IRC system and mode entry statistics) and Appendix 1.1.12.