Rule CIC333: High percent waits on shared temporary storage buffer pool lock

Finding: The CICS Shared Temporary Storage Queue Server statistics showed that there was a high percent of requests to the shared temporary storage buffer pool that waited on a buffer pool lock.

Impact: This finding has a LOW IMPACT or MEDIUM IMPACT on the performance of the CICS region.

Logic flow: This is a basic finding, based on an analysis of the data. The finding applies only with CICS/Transaction Server for OS/390 or for z/OS.

Discussion: Shared temporary storage queues are stored in named pools in an MVS coupling facility. A shared TS pool consists of an XES list structure, which is accessed through a cross-memory queue server region.

There are two situations in which shared temporary storage requests can wait on the shared TS pool:

- **The queue pool is locked for exclusive use**. Message **DFHXQ0407** (CF structure strname is not available for shared use) issued if a request for a TS pool cannot be satisfied because the queue pool is locked for exclusive use by some other job (such as a queue pool unload or reload job).

  The server is terminated in this situation. Consequently, this situation is unlikely to be the cause of frequent waits on a shared temporary storage buffer pool lock.

- **SUSPEND on resource type TSPOOL**. Resource type TSPOOL indicates that the maximum number of concurrent requests (10) for a temporary storage pool in the coupling facility has been reached. The task resumes when one of the requests completes.

Shared temporary storage queue server buffer pool statistics available in MXG file CICXQ2. CPExpert uses data in CICXQ2 to calculate the percent of requests to the shared temporary storage buffer pool that waited on a buffer pool lock, using the following algorithm:

$$\textit{Percent TS requests that waited on a buffer pool lock} = \frac{\textit{S2BFPWTS}}{\textit{Total TS requests}}$$

where S2BFPWTS = Maximum number of queue index buffer pool buffers used
TS requests = S2BFGETS + S2BFPUTS + S2BFFRES

S2BFGETS = Requests to get a buffer
S2BFPUTS = Requests to put back buffer with valid contents
S2BFFRES = Requests to put back a buffer as empty

CPExpert produces Rule CIC333 when the percent of requests to the shared temporary storage buffer pool that waited on a buffer pool lock is greater than the value specified by the **TSPCTWBP** guidance variable in USOURCE(CICGUIDE). The default value for the **TSPCTWBP** is 0.1, indicating that CPExpert should produce Rule CIC333 whenever more than one tenth of the requests waited on a buffer pool lock.

Suggestion: If this finding is produced often, you should consider the following alternatives:

- Verify that tasks that are performing operations on the same temporary storage queue are intended to do so. Possibly, the ID of the queue is unintentionally not unique.

- If possible, create more temporary storage queues to reduce the contention between tasks.

- If a task has made an unusually large number of shared temporary storage requests, it could be looping. You should verify that the tasks are not looping.

- Change the TSPCTWBP guidance variable in USOURCE(CICGUIDE) so Rule CIC333 is produced only when you wish to be aware of a larger number of situations when requests wait on a buffer pool lock.

- You can “turn off” the rule using the process described in Section 3 of this User Manual. However, this alternative is **not** recommended! You should always be aware of situations when requests wait on a buffer pool lock.

Reference: *CICS/TS for OS/390 Release 1.1*
CICS Problem Determination Guide: Section 2.3.12.6 (Resource type TSPOOL)

CICS/TS for OS/390 Release 1.2
CICS Problem Determination Guide: Section 2.3.12.6 (Resource type TSPOOL)

CICS/TS for OS/390 Release 1.3
CICS Problem Determination Guide: Section 2.3.12.6 (Resource type TSPOOL)

CICS/TS for z/OS Release 2.1

CICS Problem Determination Guide: Chapter 6 (Dealing with waits: Resource type TSPOOL)

CICS/TS for z/OS Release 2.2

CICS Problem Determination Guide: Chapter 6 (Dealing with waits: Resource type TSPOOL)

