
Rule CIC401: Adds were rejected because shared data table was full

Finding: The CICS interval statistics showed that adds to a shared data table were rejected because the shared data table was full.

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

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

Discussion: For VSAM data sets, CICS file control provides applications with *file control commands* that read, update, add, and browse the data sets. There are many features provided by IBM that allow applications to use file control commands to reference VSAM data sets, and allow separate applications to *share* the data. One such feature is the *shared data table* approach.

An application can specify that a CICS VSAM key-sequenced data set (KSDS) file is to use shared data table services. When the file is opened, this specification causes CICS to copy the contents of the file into an *MVS data space*. The records can be accessed in an MVS data space significantly quicker than records read from the VSAM data set or via reads serviced by a Local Shared Resources (LSR) pool.

With shared data table support, the KSDS file is called the *source data set*. The copy of the file in memory is called the *data table*. The process of copying the records from the file to the data table is called *loading the data table*. Whenever a CICS application wishes to reference the VSAM file using normal file control commands, CICS attempts to use the representation of the file in the data table, rather than accessing the source data.

If applications are running in different CICS regions on the MVS image, the region that initially opens a file assigned to a shared data table “owns” the data table. The CICS region that owns the data table is known as a *file-owning region (FOR)*, and is responsible for file access to both the data table and (if necessary) to the VSAM file.

Any other CICS region that uses the file is known as an *application-owning region (AOR)*. The AOR does not need to take any special action to

reference the data table; the linking of the file name used by the AOR to the data table in the FOR is established when the file is defined¹.

When the first file that is defined as a data table is opened in an FOR, the FOR attempts to register itself as a shared data table (SDT) server. This operation is performed automatically and is known as an SDT LOGON. The opening of the file can be caused by the FOR or by the AOR that first accesses the file.

In the FOR, the file is known as a *local file* and, in the AOR, the file is known as a *remote file*. The same region can be both an FOR with some data tables and an AOR for others (that is, a region can own a data table in its own region which it and other regions can access, and the region can also use data tables in other FORs).

CICS supports two types of data table :

- **CICS-maintained data tables.** A CICS-maintained data table is one that CICS keeps in synchronization with its source data set. Any update or delete action on a record in the CICS-maintained data table is automatically applied to the source data set *before* being applied to the data table.
- **User-maintained data tables.** A user-maintained data table (UMT) is one that is not maintained by CICS, but is completely maintained by user code. A UMT is detached from its source data set after the table is loaded from the source data set. After loading of the data table has completed, all file control commands that access the filename are performed only on the data table.

Records are placed into a shared data table in one of three ways: (1) records are placed in the data table by the initial loading of the data table from the source data set, (2) records are subsequently added to the data table from the source data set, and (3) new records are written to the data table after the data table has been loaded.

- **Initial loading of the data table.** During initial loading of the data table, CICS reads the entire VSAM KSDS file and attempts to place all records in the data table. The XDTRD user exit can be used to limit the records that are placed in the data table (using screening criteria appropriate to the applications sharing the data). If all records that pass the screening

¹ A data table is defined by means of the CEDA DEFINE FILE command.

criteria (if any) will not fit into the data table², a “table full” condition applies during the initial loading of the data table.

- **Records subsequently added from the source data set.** Records from a VSAM KSDS file can be added to a data table after the initial loading³ if either (1) records would not fit into the data table because the table became full during initial loading or (2) the user exit had excluded records during initial loading but an excluded record was subsequently required by an application. If the data table had become full during initial loading, some record must be deleted from the data table for another record to be subsequently added from the source data set.
- **New records written to the data table.** New records can be added to a data table via the WRITE file control command. The XDTAD global user exit program can be used to limit the records that are placed in the CFDT as a result of a WRITE request issued to a data table (using screening criteria appropriate to the applications sharing the data). If the data table is at its maximum number of records when a WRITE is attempted, a “table full” condition applies⁴.

Regardless of how records are added to a shared data table, the number of records in the table cannot exceed the number specified by the MAXNUMRECS variable specified for the table. If that limit is reached during loading or while adding to the table, a “table full” condition applies. Encountering a “table full” condition can have several undesirable effects.

- The AOR requesting the record must function ship the request to the FOR, and the record must be retrieved from the source data set⁵. Retrieving the record from the source data set requires more overhead and more elapsed time than retrieval from the data table. Additionally, function shipping requires significantly more overhead than does cross-memory services.

On the FOR, the source data set would be accessed to add the records to the data table. However, since the data table was at its maximum number of records, the records would then be rejected (unless some record had been deleted from the data table while the function-ship

²The MAXNUMRECS parameter of the DEFINE FILE command can be used to limit the number of records that can be placed in the data table. The MAXNUMRECS parameter has a default maximum of NOLIMIT, which means that the entire VSAM KSDS file can be placed in the data table (subject, of course, to screening criteria applied by the XDTRD user exit).

³Records from the source data set cannot be added to a user-maintained data table after the initial loading of the data table. User-maintained data tables are separated from the source data set after loading.

⁴Note that this situation would not normally occur if the record had been updated, since a “read for update” would have been issued to obtain a lock on the record, and the record would simply be re-written to the table.

⁵This description applies only to CICS-maintained data tables. As mentioned earlier, a user-maintained data table is not associated with its source data set after the table is loaded.

process was executed). This sequence causes considerable overhead, and would result in performance degradation.

- If the data table size specified in the file definition is significantly less than the number of records in the source data set, only a small part of the file is loaded. Potentially, many function-shopped requests could be sent by the AOR to the FOR, not only creating unnecessary overhead but also delaying access to the records and causing increased response.

Shared data table statistics are available in MXG file CICFCR. CPEXpert uses the A17DTATF variable to assess the number of records that CICS attempted to add to the table but was unable to do so because the table was full. This count means that the data table already contained the maximum number of records specified in the MAXNUMRECS parameter of the DEFINE FILE command.

CPEXpert produces Rule CIC401 when the A17DTATF value is greater than the **SDTFULL** guidance variable in USOURCE(CICGUIDE). The default value for the **SDTFULL** guidance variable is one, indicating that CPEXpert should produce Rule CIC401 when more than one shared data table full condition was detected. Note that the initial loading of the shared data table normally will produce a count of one for the A17DTATF, so the default value of the SDTFULL variable accounts for this situation.

Many of the above comments apply to a *Coupling Facility Data Table* (CFDT). If CFDT information exists (in MXG CICCFS6D, CICCFS6D, CICCFS8D, and CICCFS9D), CPEXpert will analyze CFDT performance problems. Since a CFDT is a data table (residing in a coupling facility rather than an MVS data space), the data table statistics in CICFCR apply to a CFDT as well as a normal shared data table.

Rule CIC400(series) report data table information regardless of whether the data table resides in an MVS data space or in a coupling facility. Rule CIC420(series) report data table information when the data table resides in a coupling facility.

CPEXpert analyzes the CFDT coupling facility statistics with respect to a CFDT exceeding the MAXNUMRECS value. Rule CIC424 (*MAXNUMRECS was reached for CFDT*) will be produced to report information on a coupling facility basis, if the number of records in a CFDT exceeds the MAXNUMRECS value specified for the CFDT. Additionally, CPEXpert will produce Rule CIC401 to identify the data table in the coupling facility, and will report information about the CFDT just as though it were a data table in an MVS data space. This point is particularly important since the CFDT statistics do not identify a specific CFDT that exceeds the MAXNUMRECS value.

Suggestion: If Rule CIC401 is produced regularly (and particularly if the shared data table full count is high), you consider the following alternatives:

- Increase the number of records allowed in the shared data table. The number of records allowed in the shared data table is controlled by MAXNUMRECS parameter of the DEFINE FILE command. The DEFINE FILE command can be applied at the CEDA panel.

Increasing the number of records in the shared data table would result in a larger table, using more virtual storage. Using more virtual storage could increase CICS paging operations, or increase overall system paging. Consequently, paging rates should be monitored if there is a large increase in the number of records allowed in the shared data table.

- If increasing the number of records allowed in the shared data table is not feasible, determine whether the XDTRD user exit has been used for the VSAM KSDS file. The **EXIT** column of the Rule CIC401 output will show the number of records rejected by the XDTRD or XDTAD user exit.

If the XDTRD user exit has **not** been used during initial loading, consider using the XDTRD user exit and devising appropriate screening criteria to be applied to the VSAM KSDS records as they are read by CICS during initial loading of the table. If the XDTRD user exit **has** been used, consider revising the screening criteria in the user exit to exclude a larger number of records during loading.

- If increasing the number of records allowed in the shared data table is not feasible, determine whether (1) the shared data table had a significant number of WRITE requests and (2) whether the XDTAD user exit has been used. If there was a significant number of WRITE requests and the XDTAD user exit has **not** been used, consider adding the XDTAD user exit and devising appropriate screening criteria to be applied to the VSAM KSDS records as they are written to the data table. If the XDTAD user exit **has** been used, consider revising the screening criteria in the user exit to exclude a larger number of records during loading.

Both the above options should be evaluated with the goal of reducing the number of records that are loaded into the shared data table so some space is available for added records.

- Examine the applications accessing the shared data table to determine whether records can be deleted (using the DELETE command) from the table, or whether they can be deleted earlier. This command can be

used with CICS-maintained shared data tables only⁶ if *the record in the VSAM file should be deleted*.

- You can change the **SDTFULL** guidance variable in USOURCE(CICGUIDE) if you believe that Rule CIC401 is produced too often. If you chose this option, please send a note to Don_Deese@cpexpert.com to explain your reason, so I can better appreciate why a site would knowingly experience table full conditions.

Reference: CICS/TS for OS/390 Release 1.3 *CICS Shared Data Tables Guide*:
Section 6.1 Using the DEFINE FILE command to define data tables
Section 8.4 Interpreting data table statistics

CICS/TS for z/OS Release 2.1 *CICS Shared Data Tables Guide*:
Section 6.1 Using the DEFINE FILE command to define data tables
Section 8.4 Interpreting data table statistics

CICS/TS for z/OS Release 2.2 *CICS Shared Data Tables Guide*:
Section 6.1 Using the DEFINE FILE command to define data tables
Section 8.4 Interpreting data table statistics

⁶With CICS-maintained shared data tables, file update actions against the shared data table will be applied to the file before they are applied to the table.