

---

## Rule WLM652: False Lock Contention was high for the indicated structure

---

**Finding:** The false lock contention for the indicated structure was higher than guidance provided by IBM.

**Impact:** This finding can have a LOW IMPACT, MEDIUM IMPACT, or IMPACT on the signalling performance of the sysplex. The level of impact depends on the amount of false lock contention. However, when analyzing the impact of this finding, you should keep in mind that (1) false lock contention requires overhead and (2) false lock contention is unnecessary as it normally can be eliminated.

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

**Discussion:** *Locking* is the mechanism used to reserve all or part of a database so that other programs will not be able to update the data until the user placing the lock has finished processing the data. By locking the data, users can be sure that the information they are processing is current. Without locking, users might lose updates or access invalid or incomplete data. Locking is necessary, of course, only if one or more of the users of the data will be performing updates. If no updating of the data is performed, locking is unnecessary; the data may be concurrently accessed by any number of user without worry that the data is incomplete or invalid.

*Lock contention* occurs when one user wishes to access data and some other user has placed a lock on the data. The user wishing to access the data usually is suspended until the data is available (that is, until the lock is released). Techniques such as separating data, choosing locking parameters, and monitoring for contention can be used to provide a balance between concurrency of access, isolation and integrity of data, and efficient use of system resources..

The coupling facility lock structure contains information used to determine cross-system contention on a particular resource. IRLM assigns (or "hashes") locked resources to an entry value in the lock structure in the coupling facility. IRLM uses the lock table to determine whether a resource is locked. If the lock structure defined on the coupling facility is too small, the hashing algorithm can select the same lock table entry for two different locks. This situation is termed *false lock contention*. The user wishing to access the locked data is suspended until it is determined that there is no real lock contention on the resource.

---

SMF Type 74 (Subtype 4 - Coupling Facility Activity) records contain information describing the requests for data, the number of requests that were delayed because of lock contention, and the number of requests that encountered false lock contention. CPExpert analyzes this information to determine whether an excessive percentage of requests encountered false lock contention.

IBM documents have been inconsistent with respect to guidance for false lock contention.

- IBM's *Setting up a Sysplex* document contained Section 6.3.1: Lock Contention prior to OS/390 Version 2 Release 6. This section specified an objective that no more than 0.1% of **total** requests should experience false lock contention. This section was removed completely with OS/390 Version 2 Release 6 with the comment in the CHANGES section that the document “has been updated with more recent information about tuning coupling facilities. No guidance about excessive false lock contention is contained in the *Setting up a Sysplex* document after V2R5.
- With DB2 Version 4, Section 6.3.2.3: Avoid False Contention, IBM stated “If possible, try to keep false contention to no more than 50 percent of total global lock contention. (However, if total global lock contention is a very low value, it might not be as necessary to reduce false contention.)”
- With DB2 Version 5, IBM removed that statement from Section 6.3.2, and provided no specific guidance regarding false lock contention. Rather, IBM calculated false contention as “the number of false contentions divided by the total number of requests that went to XES (excluding asynchronous requests).” That particular calculation in the example given by IBM resulted in 0.4, which IBM concluded “false contention is 0.4 percent, a very good number.”
- *DFSMSdfp Storage Administration Reference* (SC26-7331) for OS/390 V2R10 contains the statement: “A good goal is to have total (real and false) global lock contention of less than one percent. The false contention component of the total global lock contention should be less than one-half of one percent, and ideally, should be substantially less than this.” Additionally, the discussion on defining a lock structure states “These lock size estimates include the memory requirements for both the lock table and the record-lock memory. Use these estimates as rough initial values to help you attain a locking structure with a desired false contention target of approximately one-half of 1% or less. “

- OS/390: RMF Performance Management Guide (SC28-1951) still contains the “no more than 0.1% of **total** requests” statement, but this document likely has not been not regularly updated.

CPEXpert divides R744SFCN (the number of times any request encountered false lock contention) by R744STRC (the total number of lock-related requests) for lock structures, to yield the percent of requests that experienced false lock contention. CPEXpert produces Rule WLM652 when this percent is more than the value specified for the **FALSECNT** guidance variable.

The default value for the **FALSECNT** guidance variable is 0.5%, indicating that CPEXpert should produce Rule WLM652 when more than one-half of one percent of the lock-related requests encountered false lock contention.

CPEXpert additionally checks that the overall lock contention was at least as high as 25% of the value specified in the **LOCKCONT** guidance variable. This test is made to avoid spurious production of Rule WLM652.

The following example illustrates the output from Rule WLM652:

```

RULE WLM652: FALSE LOCK CONTENTION WAS HIGH

DB2DBP2_LOCK1: The number of locks with false contention should be less
than 0.5% of the total requests. The false lock contention exceeded the
guideline for this structure. False lock contention occurs when the
hashing algorithm hashes to the same lock table entry (hash value) for
two different locks. False lock contention can be reduced by increasing
the size of the lock structure or selecting a better value for the
MAXUSRS parameter in IRLMPROC. Refer to Rule WLM652 in the WLM Component
User Manual for additional suggestions.

MEASUREMENT INTERVAL      TOTAL LOCK      FALSE LOCK      PERCENT FALSE
15:15-15:30,02OCT1996    REQUESTS        CONTENTION      LOCK CONTENTION
                          12,676          2,654           21

```

**Suggestion:** False lock contention often can be reduced by increasing the size of the lock structure or selecting a better value for the MAXUSRS parameter in IRLMPROC. Please note that if you do increase the size of the lock structure, you should increase by a power of 2 to avoid wasting storage.

Additionally, CPEXpert suggests that you consider the following alternatives, depending on the type of lock structure experiencing the contention:

- If the structure involved is DB2, you should refer to the "Avoiding False Lock Contention" section in the DB2 Data Sharing: Planning and Administration document for your version of DB2.

- 
- If you are using VSAM Record Level Sharing (RLS), you should refer to "Avoiding False Lock Contention" in the DFSMSdfp Storage Administration Reference.
  - You can adjust CPEXpert's analysis by altering the value specified for the **PCTFALSE** guidance variable in USOURCE(WLMGUIDE).

**Reference:** OS/390: Setting Up a Sysplex (GC28-1779) for OS/390 prior to V2R6  
Section 6.3.1: Lock Contention

OS/390: DB2 for MVS/ESA Version 4 Data Sharing: Planning and Administration (SC26-3269)  
Section 6.3.2.3: Avoid False Lock Contention

OS/390: DB2 for MVS/ESA Version 5 Data Sharing: Planning and Administration (SC26-8961)  
Section 7.4.2.3: Avoid False Lock Contention

OS/390: DB2 for MVS/ESA Version 6 Data Sharing: Planning and Administration (SC26-9007)  
Section 7.5.2.3: Avoid False Lock Contention

OS/390 and z/OS: DB2 for MVS/ESA Version 7 Data Sharing: Planning and Administration (SC26-9935) |  
Section 6.5.2.2: Avoid False Lock Contention |

DFSMSdfp Storage Administration Reference for OS/390 (SC26-7331)  
Section 14.1.8.2: Avoiding False Contention

OS/390: RMF Performance Management Guide (SC28-1951)  
Section 6.4.4.2: Don't Make Additional Work

z/OS: RMF Performance Management Guide (SC33-7992) |  
Section 6.2.4.2: Don't Make Additional Work |