
Rule DB2-303: Locks were escalated to exclusive mode

Finding: Locks were escalated to exclusive mode.

Impact: This finding can have a LOW IMPACT, MEDIUM IMPACT, or HIGH IMPACT on the performance of the DB2 subsystem. The level of impact depends on the number of locks that were escalated to exclusive mode.

Discussion: Lock escalation is the act of releasing a large number of page or row locks, held by an application process on a single table or table space, to acquire a table or table space lock, or a set of partition locks, of lock Mode S or lock Mode X¹.

Lock escalation balances concurrency² with performance by using page or row locks while a process accesses relatively few pages or rows, then changing the lock to table space, table, or partition locks when the process accesses many pages or rows. Thus, DB2 allows applications to concurrently access information so long as the performance cost of locking is not prohibitive.

The page or row locking uses *significant* processing time. Once the performance cost of locking is unacceptable, DB2 changes (or escalates) the locks to lock the entire table space, table, or partition (thus holding a single lock) and releases all the individual page or row locks. This lock escalation can reduce concurrency because other applications *may* not be able to access the table space, table, or partition (depending on the locking required). However, the lock escalation improves performance because the overhead of managing many locks is exchanged for the overhead of managing only a single lock.

When it occurs, lock escalation varies by table space, depending on the values of LOCKSIZE and LOCKMAX, as described below.

- The *LOCKSIZE* clause on the CREATE TABLESPACE and ALTER TABLESPACE statements specifies the size of a lock held on a table or table space by any application process that accesses it. The default specification of LOCKSIZE=ANY allows DB2 to choose the size of the lock. Other options can be specified in unique situations to explicitly restrict the size of the lock. These options are

¹Please refer to Rule DB2-301 for a discussion of different lock modes.

²Concurrency is the ability of more than one application process to access the same data at essentially the same time.

LOCKSIZE=TABLESPACE, LOCKSIZE=TABLE, LOCKSIZE=PAGE, and LOCKSIZE=ROW. However, the default LOCKSIZE=ANY should be used unless the situation is unique.

- *LOCKMAX* is an option, available with DB2 Version 4, that is specified on the CREATE TABLESPACE and ALTER TABLESPACE statements. The LOCKMAX option defines the maximum number of page or row locks that an application process can hold simultaneously in the table space. If a program requests more than that number, locks are escalated. The page or row locks are released, and the intent lock on the table space or segmented table is promoted to S or X.

Prior to DB2 Version 4, the maximum number of page or row locks that an application process can hold simultaneously in the table space is applied to **all** table spaces, via the LOCKS PER TABLE(SPACE) field of panel DSNTIPJ. This specification on panel DSNTIPJ is still made with DB2 Version 4, but the specification can be overridden for individual table spaces using the LOCKMAX clause.

DB2 Version 4 extended the lock escalation process by:

- Applying the count of locks to row locks in the same way as to page locks.
- Lock escalation applies not only to table spaces defined with LOCKSIZE ANY, but also to those defined with LOCKSIZE PAGE or ROW.
- A different lock escalation cutoff point can be selected for a specific table space by specifying LOCKMAX for that table space. Lock escalation of a table space can be disabled by setting LOCKMAX to 0.

While it is possible to improve system performance by using table locks or table space locks, lock escalations are generally undesirable and are caused by processes that use a large number of page or row locks.

CPEXpert compares the QTXALEX variable in DB2STATS (the number of locks escalated to exclusive mode) with the **QTXALEX** guidance variable in USOURCE(DB2GUIDE). CPEXpert produces Rule DB2-303 when the number of locks escalated to exclusive mode exceeds the value specified by the QTXALEX guidance variable.

The default value for the QTXALEX guidance variable is 0, indicating that CPEXpert should produce Rule DB2-303 when any locks were escalated to exclusive mode.

The following example illustrates the output from Rule DB2-303:

RULE DB2-303: LOCKS WERE ESCALATED TO EXCLUSIVE MODE

Lock escalation releases a large number of page or row locks, held by an application process on a single table or table space, to acquire a table or table space lock, or a set of partition locks, of mode S or X. Escalation can cause unpredictable response times. Lock escalation should happen only when an application process updates or references (if repeatable read is used) more pages or rows than it normally does. This situation occurred during the intervals shown below:

MEASUREMENT INTERVAL	NUMBER OF LOCK REQUESTS	NUMBER OF LOCKS ESCALATED TO EXCLUSIVE
17:14-17:44, 16SEP1999	121,040	6

Suggestion: If Rule DB2-303 is produced frequently, you should consider the following alternatives:

- CPEXpert will produce Rule DB2-305 if more than 25% of the lock escalations caused timeouts or deadlocks. You can assess the importance of this rule (DB2-302) based on whether timeouts or deadlocks were reported by Rule DB2-305.
- You might alter the table to increase LOCKMAX and thus decrease the number of escalations.
- Alternatively, you might revise the application to ensure that it commits more frequently.
- Alternatively, you might let the process that is causing the lock escalations begin by locking the entire table space, using the statement LOCK TABLE. This would prevent concurrency, but it is a reasonable solution for some end-of-month or end-of-year situations when a process updates more pages than it normally does.
- Alternatively, you can alter CPEXpert's analysis by modifying the **QTXALEX** guidance variable in USOURCE(DB2GUIDE).

Reference: DB2 for OS/390 Version 3: Administration Guide
Section 2.4.1.23 (IRLM Panel 2: DSNTIPJ)
Section 5.7.4.5.3 (Lock Escalation)

DB2 for OS/390 Version 4: Administration Guide
Section 5.7.4.5.3 (Lock Escalation)
Section 5.7.6.3.3 (LOCKMAX Clause)

DB2 for OS/390 Version 5: Administration Guide

Section 5.7.4.5.3 (Lock Escalation)
Section 5.7.5.3.3 (LOCKMAX Clause)

DB2 for OS/390 Version 6: Administration Guide
Section 5.7.4.5.3 (Lock Escalation)
Section 5.7.5.3.3 (LOCKMAX Clause)

DB2 UDB for OS/390 and z/OS, Version 7: Administration Guide
Section 5.7.4.5.3 (Lock Escalation)
Section 5.7.5.3.3 (LOCKMAX Clause)

DB2 UDB for z/OS Version 8: Administration Guide
Chapter 30. Improving concurrency
Aspects of Transaction Locks
Lock Escalation

DB2 UDB for z/OS Version 9: Performance Monitoring and Tuning Guide
Chapter 8. Improving concurrency
Aspects of Transaction Locks
Lock Escalation

DB2 10 for z/OS: Managing Performance
Chapter 27. Programming for concurrency
Options for tuning locks

DB2 11 for z/OS: Managing Performance
Chapter 17. Concurrency and locks
Lock escalation