

**Блокировки**

# Блокировки

Блокировка — это механизм, с помощью которого компонент Database Engine синхронизирует одновременный доступ нескольких пользователей к одному фрагменту данных.

Каждая блокировка обладает тремя свойствами:

- гранулярностью (или размером блокировки)
- режимом (или типом блокировки)
- продолжительностью

# Гранулярность блокировок

Ресурс	Описание
RID	Идентификатор строки, используемый для блокировки одной строки в куче
KEY	Блокировка строки в индексе, используемая для защиты диапазонов значений ключа в сериализуемых транзакциях.
PAGE	8-килобайтовая (КБ) страница в базе данных, например страница данных или индекса.
EXTENT	Упорядоченная группа из восьми страниц, например страниц данных или индекса.
HOBT	Куча или сбалансированное дерево. Блокировка, защищающая индекс или кучу страниц данных в таблице, не имеющей кластеризованного индекса.
TABLE	Таблица полностью, включая все данные и индексы.
FILE	Файл базы данных.
APPLICATION	Определяемый приложением ресурс.
METADATA	Блокировки метаданных.
ALLOCATION_UNIT	Единица размещения.
DATABASE	База данных, полностью.

# Режимы блокировки

Режим блокировки	Описание
Совмещаемая блокировка (S)	Используется для операций считывания, которые не меняют и не обновляют данные, такие как инструкция SELECT.
Блокировка обновления (U)	Применяется к тем ресурсам, которые могут быть обновлены. Предотвращает возникновение распространенной формы взаимоблокировки, возникающей тогда, когда несколько сеансов считывают, блокируют и затем, возможно, обновляют ресурс.
Монопольная блокировка (X)	Используется для операций модификации данных, таких как инструкции INSERT, UPDATE или DELETE. Гарантирует, что несколько обновлений не будет выполнено одновременно для одного ресурса.
Блокировка с намерением	Используется для создания иерархии блокировок. Типы намеренной блокировки: с намерением совмещаемого доступа (IS), с намерением монопольного доступа (IX), а также совмещаемая с намерением монопольного доступа (SIX).
Блокировка схемы	Используется во время выполнения операции, зависящей от схемы таблицы. Типы блокировки схем: блокировка изменения схемы (Sch-S) и блокировка стабильности схемы (Sch-M).
Блокировка массового обновления (BU)	Используется, если выполняется массовое копирование данных в таблицу и указана подсказка TABLOCK.
Диапазон ключей	Защищает диапазон строк, считываемый запросом при использовании уровня изоляции сериализуемой транзакции. Запрещает другим транзакциям вставлять строки, что помогает запросам сериализуемой транзакции уточнять, были ли запросы запущены повторно.

# Совместимость блокировок

Совместимость блокировок определяет, могут ли несколько транзакций одновременно получить блокировку одного и того же ресурса.

Если ресурс уже заблокирован другой транзакцией, новая блокировка может быть предоставлена только в том случае, если режим запрошенной блокировки совместим с режимом существующей. В противном случае транзакция, запросившая новую блокировку, ожидает освобождения ресурса, пока не истечет время ожидания существующей блокировки.

# Продолжительность блокировки

Блокировки для выполнения инструкций `SELECT` выполняются только на время чтения ресурса, но не во время запроса.

Блокировки для инструкций `INSERT`, `UPDATE` и `DELETE` сохраняются на все время выполнения запроса. Это помогает гарантировать согласованность данных и позволяет откатывать запросы в случае необходимости.

Когда запрос выполняется в рамках транзакции, продолжительность блокировки определяется тремя факторами:

- типом запроса
- уровнем изоляции транзакции
- наличием или отсутствием подсказок блокировки

# Эскалация блокировок

Эскалация (lock escalation) связана с тем, что по мере увеличения, количества отдельных малых заблокированных объектов накладные расходы, связанные с их поддержкой, начинают значительно сказываться на производительности. Блокировки длятся дольше, что приводит к спорным ситуациям — чем дольше существует блокировка, тем выше вероятность обращения к заблокированному объекту со стороны другой транзакции.

Очевидно, что на некотором этапе потребуется выполнить объединение (увеличение масштаба) блокировок, чем собственно и занимается диспетчер блокировок.



# Взаимоблокировки транзакций





# Как выбирается жертва взаимоблокировки?

По умолчанию в качестве жертвы взаимоблокировки выбирается сеанс, выполняющий ту транзакцию, откат которой потребует меньше всего затрат.

В качестве альтернативы пользователь может указать приоритет сеансов. Если у двух сеансов имеются различные приоритеты, то в качестве жертвы взаимоблокировки будет выбран сеанс с более низким приоритетом. Если у обоих сеансов установлен одинаковый приоритет, то в качестве жертвы взаимоблокировки будет выбран сеанс, откат которого потребует наименьших затрат.

Если сеансы, вовлеченные в цикл взаимоблокировки, имеют один и тот же приоритет и одинаковую стоимость, то жертва взаимоблокировки выбирается случайным образом.