

Механизм блокировок

Хорошая [статья по блокировкам на ИТС](#), но слегка устаревшая, в 8.3.14 алгоритм пессимистичной блокировки изменен.

[Очень интересная статья](#), исследование блокировок.

[Ускорение работы при высокой нагрузке](#), для погружения.

В простых словах и с примерами [описание](#) управляемой блокировки

Механизмы и типы блокировок.

Критичная тема для понимания работы платформы.

Механизм объектной блокировки - механизм обеспечения целостности объекта (элемент справочника, документ, ...) средствами платформы 1С. В большинстве случаев это связано с интерактивной работой пользователей в формах: редактирование существующих объектов, удаление, создание новых и др.

Механизм транзакционной блокировки - механизм обеспечения целостности и непротиворечивости данных средствами ядра СУБД на уровне транзакций.

Отличия заключается в элементе контроля и выбора точки блокировки. К тому же, у разных СУБД разные уровни блокировок (на уровне записей или на уровне таблиц). Например, элемент справочника состоит из реквизита Наименование и табличной части Характеристики с названием и значением. С точки зрения базы данных, в ней создается основная таблица с ключом, наименованием, и дополнительная таблица Характеристики с индексом, названием, значением и ссылкой на владельца.

Блокировка на уровне объекта подразумевает наличие в БД третьей таблицы, в которой указывается, что Пользователь1 редактирует объект1. При попытке Пользователем2 изменить объект1, в операции будет отказано и до уровня старта изменений в данных дело не дойдет. Но здесь на первый план выходит алгоритм добавления и удаления данных из третьей таблицы.

Блокировка на уровне транзакции подразумевает отсутствие третьей таблицы, и при изменении данных в соответствии с логикой СУБД (либо запись об объекте1 в таблице1 и соответствующих записей в таблице2, либо целиком обеих таблиц), после успешного завершения блокировки снимаются.



- Объектная Оптимистическая. Построена на анализе номера версии объекта, хранящейся в базе данных и номера версии, помещенной в память компьютера в момент считывания данных из информационной базы. Если при записи объекта номера версий на форме и на сервере отличаются, то будет выдано предупреждение о том, что версия объекта изменилась или он был удален. Происходит автоматически при сохранении данных через механизмы формы (не через запрос) или при чтении данных (разные типы блокировок).
- Объектная Пессимистическая неявная (в форме). В тот момент, когда пользователь начинает модификацию объекта (имеется в виду в первый раз нажал на кнопку Сохранить, а не момент открытия или момент изменения данных в поле) в форме, расширение формы устанавливает пессимистическую блокировку. Если после этого другой пользователь попытается выполнить редактирование, ему будет выдано сообщение о невозможности блокировки объекта. Снимается после закрытия формы объекта либо через минуту после сохранения данных. Для отключения пессимистической блокировки в управляемых формах в свойстве основного реквизита надо снять флаг «Сохраняемые данные». Данный флаг определяет будет ли при интерактивном редактировании блокироваться данные основного реквизита, или нет.
- Объектная пессимистическая блокировка явная (в коде)
 - Для управляемых форм. Используется если необходимо просто заблокировать данные на сервере пока выполняются преобразования данных на форме. Для работы с блокировками из управляемой формы без вызова сервера используются методы «ЗаблокироватьДанныеФормыДляРедактирования()» и «РазблокироватьДанныеФормыДляРедактирования()». Как и в случае неявной, контролируются флагом «Сохраняемые данные». Пессимистическая блокировка хранится в сервисе блокировок DataEditLockService кластера 1C. Увидеть блокировку конкретного объекта можно в консоли кластера в разделе Кластеры -> Информационные базы -> Блокировки
 - Без формы Методы «Заблокировать()», «Разблокировать()» и «Заблокирован()», используются для объектов базы данных, существуют только на сервере. Метод «Заблокирован()» не может быть использован для проверки, заблокирован ли вообще объект базы данных, например, другими пользователями. Для этого следует использовать метод «Заблокировать()» в попытке.

```
&НаСервереБезКонтекста
```

```
...
```

```
Попытка
```

```
□тестобъект.Заблокировать();
```

```
□// что-то делаем с объектом
```

```
    тестобъект.Разблокировать();
```

```
Исключение
```

```
□Сообщить("Не удалось заблокировать тест");
```

```
КонецПопытки;
```

- Автоматические. Ответственность за блокировки полностью лежит на СУБД. Это облегчает работу, но создание информационной системы для большого количества пользователей на автоматических блокировках нежелательно (особенно для СУБД PostgreSQL, Oracle BD, т.к. при модификации данных они полностью блокируют таблицу).
- Управляемые. В управляемом режиме есть возможность использовать менеджер транзакционных блокировок данных. В этом режиме система использует гораздо более низкий уровень изоляции транзакций для MS SQL Server и IBM DB2, и блокировку на уровне записей для PostgreSQL. Это позволяет повысить степень параллельности работы пользователей.

Длительность блокировок

Детали установки блокировок

Настройка объектов конфигурации

Режим управления блокировками данных настраивается в свойствах объекта конфигурации, раздел Режим управления блокировками данных. Может быть Управляемый или Автоматический.

Шаблоны кода

Получить версию объекта на клиенте и на сервере.

```
&НаСервереБезКонтекста
```

```
Процедура ПолучитьВерсиюОбъектаНаСервере(ТекОбъект)
```

```
□Сообщить(ТекОбъект.ВерсияДанных);
```

КонецПроцедуры

&НаКлиенте

Процедура ПолучитьВерсиюОбъекта(Команда)

□ПолучитьВерсиюОбъектаНаСервере(Объект.Ссылка);

□Сообщить(Объект.ВерсияДанных);

КонецПроцедуры

Проверка необходимости обновления.

&НаСервереБезКонтекста

Функция ВернутьВерсиюДанныхНаСервере(ТекОбъект)

□ Возврат ТекОбъект.ВерсияДанных;

КонецФункции

&НаКлиенте

Процедура ОболочкаОбработчика()

□ВерсияНаСервере = ВернутьВерсиюДанныхНаСервере(Объект.Ссылка);

□Если ВерсияНаСервере <> Объект.ВерсияДанных Тогда

□□ЭтаФорма.Прочитать();

 ПроверкаЭлементовНаВидимость();

□КонецЕсли;

КонецПроцедуры

Проверить, заблокирован ли объект

// Попытка установки блокировки

Объект = Номенклатура.ПолучитьОбъект();

Попытка

□Объект.Заблокировать();

Исключение

□// Данные объекта уже заблокированы.□

КонецПопытки;

Проверка предположений.

Проверка факта запрета блокировки другим объектом.

&НаСервереБезКонтекста

Процедура ПроверкаПериодаБлокировкиНаСервере()

□тест = Справочники.ДляТранзакций.НайтиПоНаименованию("первый");

□тест2 = Справочники.ДляТранзакций.НайтиПоНаименованию("первый");

□тестобъект = тест.ПолучитьОбъект();

□тестобъект2 = тест2.ПолучитьОбъект();

□Попытка

□□тестобъект.Заблокировать();

□□Сообщить("Объект тест заблокирован");

□Исключение

□□Сообщить("Не удалось заблокировать тест");

□КонецПопытки;

□Попытка

□□тестобъект2.Заблокировать();

□□Сообщить("Объект тест2 заблокирован");

□Исключение

□□Сообщить("Не удалось заблокировать тест2");

□КонецПопытки;

КонецПроцедуры

Объект тест2 заблокировать не удалось.

Блокировка держится до конца транзакции

Revision #19

Created 11 February 2025 15:10:53 by Admin

Updated 13 February 2025 15:28:28 by Admin