

РЕАЛИЗАЦИЯ СЕРВЕРА БЛОКИРОВОК ДЛЯ СОВМЕСТНОГО ИСПОЛЬЗОВАНИЯ ОБЪЕКТОВ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Герасимчук М.Г.

научный руководитель старший преподаватель Макуха Л.В.

Сибирский Федеральный Университет

Объектно-ориентированное приложение, являющееся инструментом нескольких сотрудников, использует одну базу данных, в связи с этим, данные используемые клиентскими приложениями со временем устаревают, и требуется их обновление. Данную задачу можно решить различными способами, некоторые из них:

1. **Время жизни объекта.** При данном решении задачи данные объектов обновляются через некоторые промежутки времени, объект устарел и при его отображении необходимо создать данный объект из базы данных заново.
2. **Создание при использовании.** При каждом обращении к объекту, объект создаётся заново - полностью перечитывается из базы данных.
3. **Обновление по требованию.** Отслеживать все изменения произведенные пользователями над данными, и обновлять лишь те объекты, которые были изменены.

Недостаток первого способа, в том, что приходится через некоторый промежуток времени запрашивать данные из базы и заново создавать объекты, этот процесс может занимать много времени, особенно это проявляется при большом количестве объектов. Второй способ схож с первым в том, что объекты заново считываются из базы данных (вызывается конструктор объекта), в данном методе необходимо контролировать использование памяти. Третий способ требует наличие в сети дополнительного программного обеспечения – сервера блокировок, который будет контролировать изменение объектов пользователями. Положительная сторона решения задачи первого и второго способа заключается в простоте реализации, недостатком является сложность выбора корректного интервала времени и увеличение времени реакции на действия пользователя при работе с большими объёмами данных. Третий метод сложнее в реализации, однако, изменённый объект сразу становится доступен всем клиентам.

Каждый клиент, работающий совместно с сервером блокировок перед изменением данных объекта, запрашивает информацию у сервера блокировок о состоянии объекта. В зависимости от возвращаемого состояния, клиент может получить доступ к изменению данных, или же нет, в том случае когда данный объект уже изменяется другим клиентом он будет доступен только после того как клиент используемый данный объект завершит работу с ним. В процессе работы, данные изменяются, и клиент, внесший изменения, посылает серверу блокировок сообщение об окончании изменений. Сервер блокировок рассылает сообщения всем подключенным клиентам о том, что данные объекта были изменены, и клиенты которые используют данные, должны заново считать их из базы. Данный процесс схематично показан на рисунке 1.

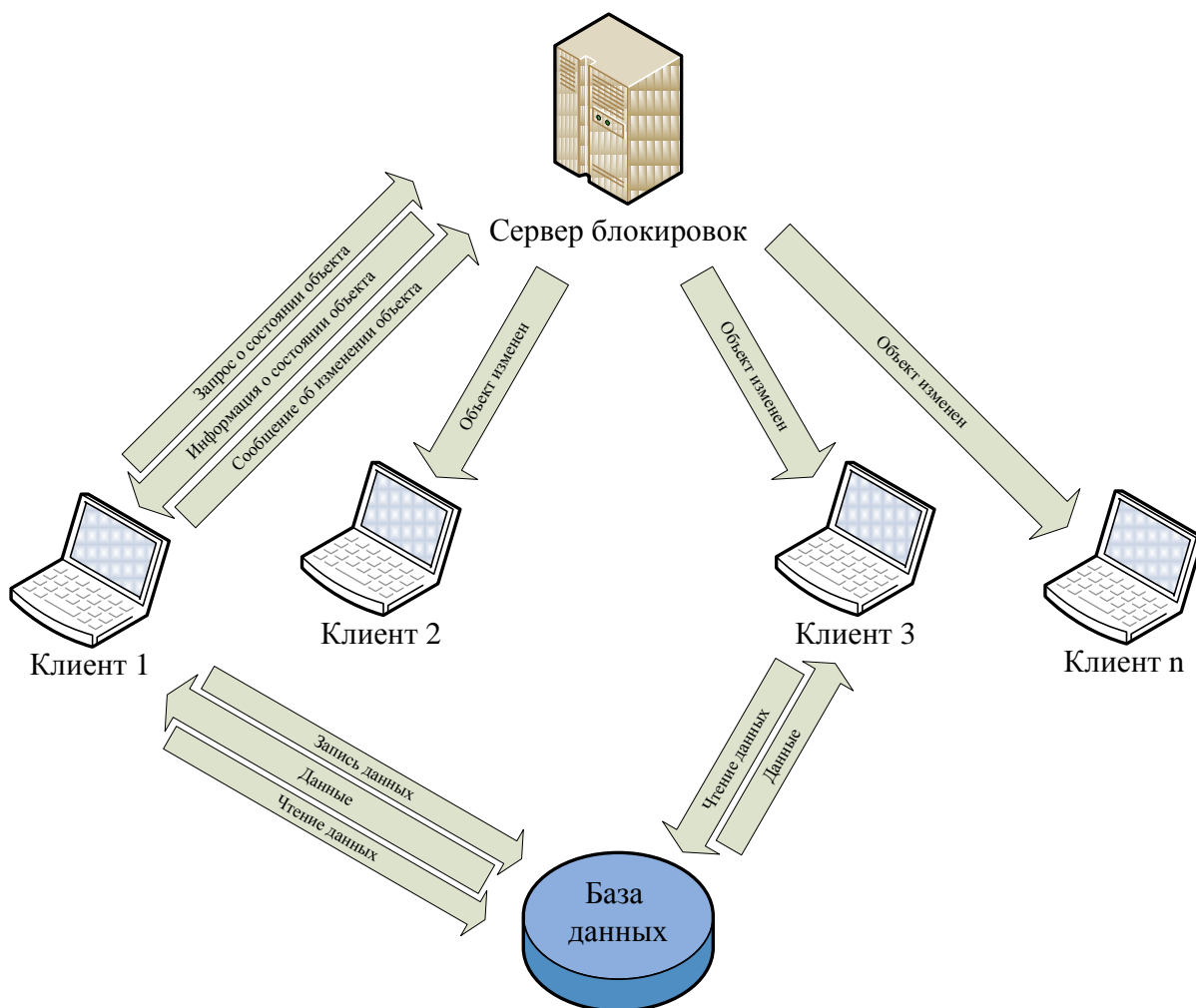


Рисунок 1 – Взаимодействия сервера блокировок с клиентами при изменении блокируемого объекта.

Для уменьшения количества запросов на изменение объектов можно блокировать связанное с ним дерево объектов или отдельную ветвь.

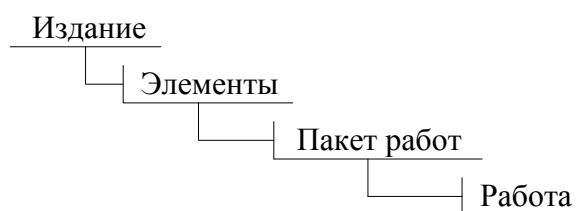


Рисунок 2 – Дерево объекта типа «Издание».

На рисунке 2, изображено дерево объекта «Издание», при изменении любого объекта из данного дерева, также блокируются все связанные с ним объекты - «Элементы», «Пакет работ», «Работа». Преимущество данной блокировки заключается в едином блокировании нескольких объектов одного дерева, например, если пользователю требуется изменить несколько работ принадлежащих одному типу, логично осуществить блокировку целой ветви дерева. За счет такого подхода, время, затраченное на блокировку/разблокировку группы объектов, будет меньше чем при блокировании отдельных объектов. Если рассматривать процесс изменения нескольких объектов, блокировка целой ветви или дерева объектов является лучшей альтернативой, чем блокировка от-

дельных объектов, занимающая большее время, однако после изменения данных клиенту требуется обновить все дерево объектов. В данной статье рассматривается сервер блокировок, осуществляющий блокировки отдельных объектов, блокировать целое дерево не всегда приемлемо, и это добавляет дополнительный функционал клиентского программного обеспечения по идентификации деревьев и проверки принадлежности объекта к дереву.

Для блокировки объекта необходима следующая информация:

- тип объекта – уникальный идентификатор характеризующую группу однотипных объектов;
- идентификатор – уникальный идентификатор характеризующий объект среди однотипных объектов.

Блокировка может иметь следующие состояния:

- объект заблокирован;
- объект разблокирован;
- объект обновляется.

Основная информация, хранящаяся на сервере блокировок:

- список работающих клиентов;
- список заблокированных объектов;
- список ошибок на сервере;
- список неблокируемых объектов;
- идентификаторы приложений, работающих с сервером.

Действия, реализуемые сервером:

- операции добавления и удаления клиентов;
- операции установки и снятия блокировки;
- операции добавления и удаления неблокируемых объектов;
- операции установки и снятия признака изменения объекта;
- операция обновления объекта.