РАЗРАБОТКА ОБЪЕКТНОЙ МОДЕЛИ РАСПРЕДЕЛЕННОЙ БАЗЫ ДАННЫХ С УЧЕТОМ ОСОБЕННОСТЕЙ СУБД MICROSOFT SQL SERVER

Лаздынь С.В., Телятников А.О., Остроухова Я.И.
Донецкий национальный технический университет, г. Донецк
кафедра автоматизированных систем управления
E-mail: slazd@ukr.net, Alexander.Telyatnikov@gmail.com

Abstract

Lazdyn S.V., Telyatnikov A.O., Ostroukhova Y.I. Development of the distributed database object model taking into account features of the DBMS MS SQL Server. *A new object model of distributed database (DDB) taking into account the features of the DBMS Microsoft SQL Server is developed, that provides the increase of exactness and authenticity of modeling. As the tool of statistical data collection about the DDB work the SQL Profiler utility is chosen. Experimental researches on the real DDB was confirmed the model adequacy and exactness.*

Постановка проблемы. При создании компьютерных информационных систем (КИС) для крупных предприятий используются распределенные базы данных (РБД), в которых данные размещены по множеству узлов при помощи фрагментации и репликации. Развитию и распространению КИС с РБД также способствует то, что большинство современных систем управления базами данных (СУБД) содержат средства для создания и поддержки РБД.

РБД представляет собой сложную динамическую систему, в которой выполняется множество запросов к распределенным данным, производятся обновления множества копий, размещенных на разных узлах компьютерной сети. Производительность РБД зависит не только от параметров технических средств (серверов, каналов связи), но и от того, насколько рационально распределены данные в системе. Поэтому задачи моделирования РБД с целью обеспечения высокой эффективности их работы возникают как при проектировании новых КИС, так и при модернизации существующих систем.

Анализ последних исследований в области моделирования РБД. Вопросам моделирования РБД посвящен ряд научных работ и публикаций. Весомый вклад в развитие этого направления внесли Г.Г. Цегелик, А.Г. Мамиконов, В.В. Кульба и другие ученые [1-3]. К недостаткам разработанных моделей РБД можно отнести то, что используемый в них аналитический подход не позволяет в полной мере учесть все особенности построения и функционирования РБД, а именно: не учитывает фрагментацию и репликацию данных, не отражает динамические процессы, происходящие в системе.

Представляет интерес использование объектно-ориентированного подхода для моделирования РБД, предложенного в работе [4]. Разработанная в ней объектная модель РБД представляет собой систему взаимодействующих объектных моделей ее типовых компонентов таких как узел, канал связи, запрос, таблица данных. Такой принцип построения модели позволяет устранить основные недостатки предыдущих разработок, а также обеспечивает возможность моделировать КИС с РБД, имеющие различную конфигурацию, как с точки зрения технических средств, так и размещения данных по узлам. Основным недостатком указанной модели является то, что она разработана для общего случая, не учитывает особенностей и возможностей конкретной СУБД, которая используется в моделируемой РБД.

Выбор СУБД и формулировка цели работы. В настоящее время практически все коммерческие СУБД предлагают те или иные инструменты для создания распределенных систем. К таким СУБД относятся IBM DB2, Oracle, Microsoft SQL Server, Ingress и др. Среди них самой распространенной является Microsoft (MS) SQL Server по причине широкого

использования во всем мире платформы Windows, на которой эта СУБД базируется. По сравнению с указанными выше СУБД MS SQL Server имеет ряд преимуществ, таких как невысокая стоимость, простота администрирования. Вместе с тем эта СУБД обладает всеми необходимыми функциональными возможностями для создания и поддержания РБД. Для сравнения и выбора СУБД используется TPC-анализ производительности, который показывает отношение количества транзакций, обрабатываемых за некий промежуток времени к стоимости всей системы. По этому показателю MS SQL Server сейчас является мировым лидером. На основании вышесказанного в качестве объекта для моделирования выбрана РБД, работающая под управлением СУБД MS SQL Server.

Поэтому целью данной работы является совершенствование разработанной в [4] объектной модели РБД с учетом особенностей ее функционирования в СУБД Microsoft SQL Server, что обеспечит повышение точности и достоверности моделирования процессов, протекающих в РБД.

Особенности СУБД MS SQL Server при работе с распределенными базами данных. Рассмотрим специфику реализации в данной СУБД двух основных процессов, связанных с РБД: выполнение распределенных запросов и распространение обновлений (репликация данных) [5].

При выполнении распределенных запросов производится обращение к распределенным источникам данных, которое в MS SQL Server реализуется с помощью технологии OLE DB. Обращение к конкретному источнику данных происходит через специального поставщика (OLE DB Provider). В составе MS SQL Server существует набор поставщиков, предоставляющих возможность обращаться к большинству современных источников данных, таких как Oracle, DB2, ADSI, ODBS, Exchange. Для идентификации данных на удаленном источнике в MS SQL Server используются связанные серверы.

Связанный сервер представляет собой запись в системной таблице sysservers базы данных master, в которой указывается для каждого источника данных: используемый поставщик OLE DB, имя источника данных, его размещение, строка для установления соединения, каталог (имя базы данных), имя компьютера сети, сопоставление и некоторая другая информация.

В процессе выполнения распределенного запроса весь запрос разбивается на подзапросы, каждый из которых передается на свой связанный сервер. Связанный сервер выполняет подзапрос, обращающийся к его локальным данным, и результаты выполнения подзапроса передает серверу, инициировавшему запрос.

Репликация (распространение обновлений) представляет собой совокупность механизмов MS SQL Server, обеспечивающих отображение изменения данных, сделанных на одном сервере, на другие серверы. Подсистема репликации MS SQL Server основывается на технологии Replication Distribution Interface, которая позволяет включать в репликацию данных не только серверы SQL Server 2000, SQL Server 7.0 или SQL Server 6.x, но и другие системы хранения и обработки данных, например, Microsoft Access, Oracle, dBase, Paradox. С помощью этой подсистемы администратор может создать распределенные гетерогенные системы масштаба предприятия.

Подсистема репликации MS SQL Server использует понятие издатель (publisher), дистрибьютор (distributor) и подписчик (subscriber). Издателем называется сервер, который предоставляет (публикует) расположенную на нем информацию другим серверам. Подписчиком является сервер, который принимает данные от издателя. Этот сервер подписывается на одну или более публикаций и периодически копирует опубликованные данные. Один подписчик может получать данные от множества издателей. Промежуточным звеном между издателем и подписчиком является дистрибьютор, который собирает всю информацию, получаемую подписчиком от издателей. В качестве дистрибьютора может

быть сконфигурирован как отдельный сервер, так и сервер, являющийся издателем или подписчиком.

Подсистема репликации MS SQL Server реализована в виде специализированных агентов, выполняемых на сервере как самостоятельный процесс. Эти агенты подключаются к серверам-участникам репликации и выполняют создание копий данных и тиражирование их между другими серверами. В подсистеме репликации SQL Server существует пять агентов, каждый из которых выполняет определенную роль:

- агент моментальных снимков (Snapshot Agent);
- агент чтения журнала (Log Reader Agent);
- агент чтения очереди (Queue Reader Agent);
- агент распределения (Distribution Agent);
- агент сведения (Merge Agent).

Агенты репликации анализируют изменения, сделанные на подписчиках и издателях, подготавливают пакеты данных, копируют их подписчикам, решают конфликты изменений. В зависимости от используемого типа репликации и функции сервера, набор агентов на конкретном сервере может сильно различаться.

В Microsoft SQL Server применяется три следующих базовых типа репликации:

- репликация моментальных снимков (Snapshot Replication);
- репликация транзакций (Transactional Replication);
- репликация сведением (Merge Replication).

Анализ возможностей различных типов репликаций показал, что наилучшим из них является третий тип. Главным его достоинством, по сравнению с двумя другими, является независимость от узла-издателя. Даже если соединение с узлом-издателем отсутствует, все остальные участники репликации смогут сразу же получить модифицированные другим подписчиком данные. Репликация сведением хорошо работает при установленном соединении или без него. При установленном соединении модифицированные данные в течение очень маленького промежутка времени отображаются на остальных узлах.

Таким образом, Microsoft SQL Server имеет современные механизмы и технологии для работы с распределенными базами данных.

Разработка объектной модели РБД с учетом особенностей ее функционирования в Microsoft SQL Server. Распределенная база данных как объект моделирования представляет собой сложную динамическую систему, для исследования которой использован объектно-ориентированный подход.

В результате объектно-ориентированного анализа были выделены следующие типовые компоненты распределенной базы данных: узел, канал передачи данных, запрос, подзапрос, таблица, набор данных. Для моделирования выделенных типовых компонентов РБД разработаны соответствующие классы объектов. При построении моделей типовых компонентов, их взаимосвязей и поведения применялся унифицированный язык моделирования UML (Unified Modeling Language) [6]. Рассмотрим построение объектных моделей типовых компонентов.

Узел распределенной базы данных является центральным компонентом исследуемой системы и предназначен для хранения и обработки данных. Узел РБД представляет собой систему, состоящую из нескольких аппаратных и программных элементов. Наиболее важным программным компонентом узла является система управления базами данных, которая представляет собой программное обеспечение, управляющее доступом к базе данных. К аппаратному обеспечению узла относятся подсистема основной памяти, дисковая подсистема.

Для моделирования работы узла распределенной базы данных разработан соответствующий класс объектов “Узел РБД”. Основными свойствами узла РБД являются

наименование узла, код узла, код дистрибьютора, состояние и очередь обработки, методами – обработка подзапросов, постановка подзапроса в очередь обработки, освобождение узла.

В MS SQL Server каждый узел может быть сконфигурирован как издатель, подписчик или дистрибьютор. Дистрибьютор является промежуточным звеном между издателем и подписчиком и может быть сконфигурирован на узле-издателе, узле-подписчике или отдельном узле. Для каждого издателя устанавливается только один дистрибьютор. Каждое инициируемое обновление в первую очередь передается дистрибьютору, который распространяет его всем остальным узлам, содержащим фрагменты с обновляемыми данными. Применение дистрибьютора позволяет гибко управлять загрузкой серверов, а изменением расположения дистрибьютора можно добиться повышения производительности работы распределенной базы данных. Поэтому, чтобы учитывать влияние выбора узла-дистрибьютора на производительность работы распределенной базы данных в класс “Узел РБД” используется свойство "Код дистрибьютора".

Если в момент поступления подзапроса на узел РБД для обработки данный узел занят обработкой другого подзапроса, то новый подзапрос может быть поставлен в очередь ожидания обработки, и находиться там до момента освобождения узла. Тот факт, что занят узел РБД обработкой подзапроса или нет, отражается в свойстве "Состояние".

Свойство "Очередь обработки" представляет собой динамический массив объектов класса "Подзапрос", в который помещаются объекты подзапросов, поступивших на узел.

Класс "Узел РБД" обладает методом "Обработка запроса". Также имеются методы для работы с очередью ожидания обработки: "Постановка в очередь обработки" и "Освобождение узла".

Основной метод, выполняемый узлом – "Обработка запросов". Время выполнения обработки l-го запроса на j-м узле определяется следующим выражением :

$$T_{lj} = K_l \cdot t_{lj},$$

где K_l - количество возвращаемых записей в результате выполнения l-го запроса, обновления, $l \in [1, N_h]$; N_h - количество запросов (обновлений) в системе; t_{lj} - время обработки одной записи, возвращаемой в результате выполнения l-го запроса на j-м узле, $j \in [1, m]$; m - количество узлов в системе.

Метод "Постановка в очередь событий" осуществляет добавление объекта класса "Подзапрос" в массив очередь обработки. Метод "Освобождение узла" осуществляет проверку наличия запросов в очереди обработки и, если есть хотя бы один такой запрос, для него вызывается метод обработки запросов.

Канал связи представляет собой средство передачи данных между узлами РБД. Для моделирования работы каналов связи в процессе выполнения запросов и распространения обновлений создан класс объектов “Канал передачи данных”. Свойствами класса являются пропускная способность, состояние, трафик, очередь передачи, а методами: передача данных, постановка данных в очередь передачи, освобождение канала.

В зависимости от того, осуществляется ли передача данных по каналу в текущий момент времени, он может находиться в двух состояниях: свободен или занят. Для этого в классе "Канал передачи данных" необходимо свойство "Состояние". Та часть пропускной способности канала, которая используется для передачи данных в текущий момент времени отражена в свойстве "Трафик".

Если необходимо передавать информацию, а канал связи полностью занят, то эти данные попадают в очередь передачи и находятся там до освобождения канала. Поэтому модель канала содержит свойство "Очередь передачи" и методы "Постановка данных в очередь передачи" и "Освобождение канала".

Операция передачи подзапросов, ответов на подзапросы и обновлений реализована в модели канала методом "Передача данных".

Время выполнения передачи l-го запроса по k-му каналу определяется выражением:

$$Tn_{lk} = \frac{V_l}{B_k},$$

где V_l - объем данных l-го запроса, Мбайт; B_k - пропускная способность k-го канала передачи данных, Мбайт/с, $k \in [1, N_c]$; N_c - количество каналов передачи данных в системе.

На узлах РБД через определенный интервал времени инициируются запросы. Каждый запрос в свою очередь инициирует множество подзапросов, обращенных к удаленным источникам данных. Для моделирования выполнения запросов создан класс объектов "Запрос". Основными свойствами этого класса являются инициируемые подзапросы (представляет собой вектор указателей на объекты класса "Набор данных") и узел выполнения запроса, а методами – запуск запроса, инициация подзапроса, завершение запроса.

Выполнение подзапросов происходит следующим образом: после инициации подзапросы поступают на узлы, содержащие запрашиваемые данные, где происходит их обработка. Сформированный ответ передается на узел, где он был инициирован по каналам передачи данных. Для моделирования подзапросов разработан класс объектов “Подзапрос”. Основными его свойствами являются набор данных, к которому происходит обращение в подзапросе, объем подзапроса, среднее время обработки одной записи, количество возвращаемых записей и объем ответа. Для данного класса разработаны следующие методы: запуск подзапроса, постановка в очередь передачи, передача, постанова в очередь обработки, обработка, постановка ответа в очередь передачи, передача ответа, завершение подзапроса.

Для определения длительности обработки подзапроса на выполняемом узле используются свойства "Время обработки единицы информации" и "Объем ответа".

Методы "Запуск подзапроса", "Постановка в очередь передачи", "Передача", "Постановка в очередь обработки", "Обработка", "Постановка ответа в очередь передачи", "Передача ответа" и "Завершение подзапроса" предназначены для фиксации соответствующих моментов наступления особых состояний подзапроса в процессе выполнения.

Для моделирования характеристик таблиц РБД разработан класс “Таблица РБД”. Основными его свойствами являются: код таблицы, код родительской таблицы, код издателя. Таблица может представлять собой иерархическую структуру, когда в состав таблицы входят фрагменты, которые тоже являются таблицами. Для моделирования этой особенности класс "Таблица РБД" содержит свойство "Код родительской таблицы".

Для моделирования характеристик наборов данных разработан класс "Набор данных". Основными свойствами класса являются код узла хранения таблицы и код таблицы, а методом – проверка ограничения на наличие в РБД хотя бы одной копии таблицы.

Общая объектная модель распределенной базы данных, функционирующей под управлением MS SQL Server, построена как система взаимодействующих объектов ее типовых компонентов.

Схема взаимосвязей объектов модели РБД приведена на рис. 1 в виде диаграммы UML. На диаграмме изображены классы всех объектов модели с указанием их свойств и методов. Между объектами присутствуют описания отдельных связей, которые соединяют модели типовых компонентов РБД в единую систему. Для связей определены роли (описание взаимодействия каждого из участников связи) и множественность, которая указывает на то, сколько экземпляров одного класса может быть связано с одним экземпляром другого класса.

Данная диаграмма отражает предметную область функционирования распределенной базы данных. На диаграмме видно, что подзапросы являются составной частью запросов, которые выполняются на узлах распределенной базы данных. Подзапросы обращаются к наборам данных распределенной базы данных, которые в свою очередь предоставляют

данные для подзапросов. Подзапросы, обновления и ответы на подзапросы передаются по каналам передачи между узлами.

Моделирование выполнения распределенных запросов и распространения обновлений в РБД. В процессе функционирования распределенной базы данных реализуются два основных процесса: выполнение распределенных запросов и распространение обновлений. Приведем описание этих процессов с помощью UML диаграмм [6].

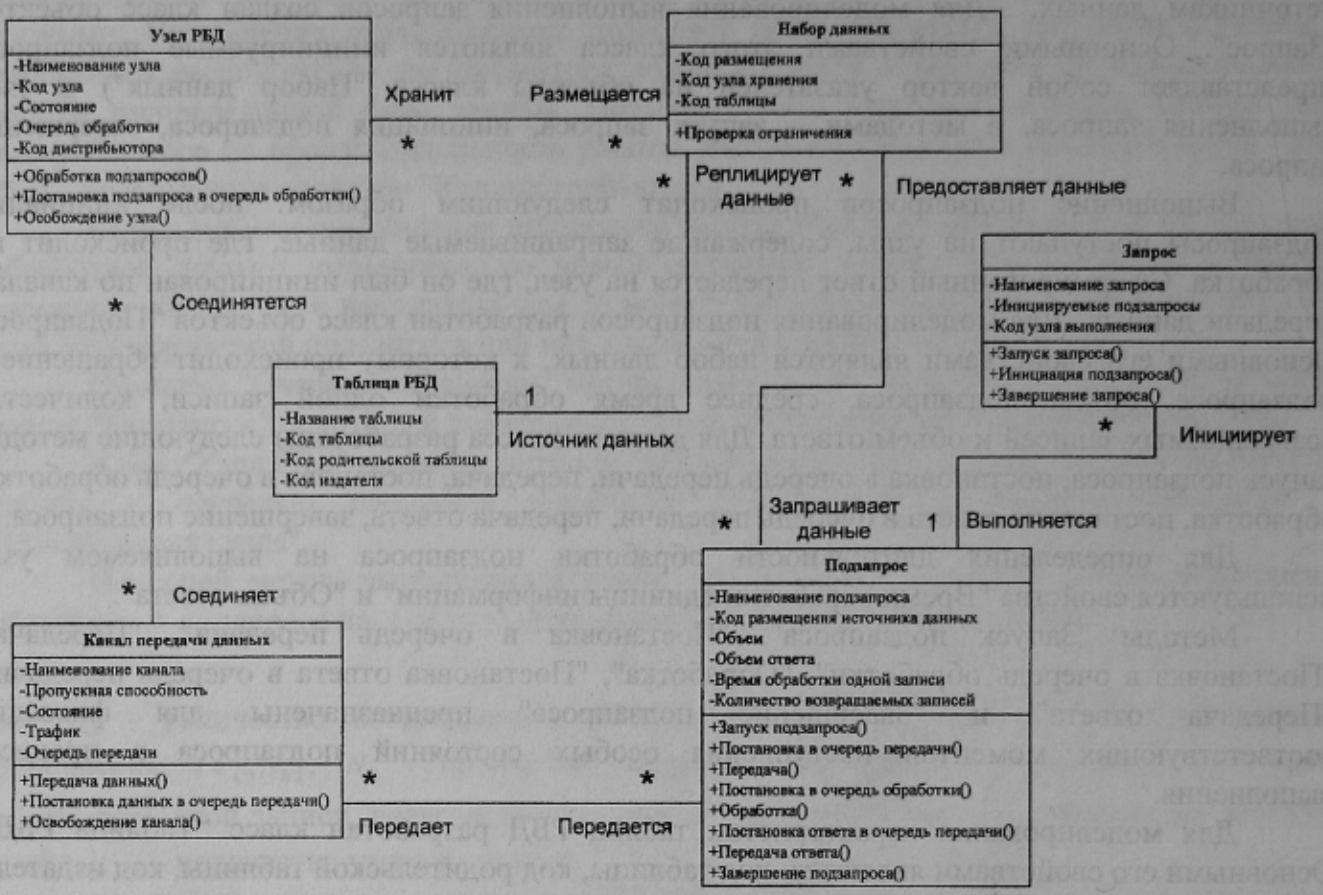

Рисунок 1 - Схема взаимосвязей объектов модели РБД

Упорядоченное во времени взаимодействие объектов при выполнении запроса представлено в виде диаграмм UML на рис. 2 и 3. На рис. 2 представлено взаимодействие объектов классов "Запрос" и "Подзапрос", упорядоченное взаимодействие объекта класса "Подзапрос" с другими объектами предоставлено на рисунке 3. Рассмотрим упорядоченное во времени взаимодействие объектов при выполнении распределенных запросов.

Вначале на узле распределенной базы данных инициируется запрос. Запрос разбивается на подзапросы, каждый из которых содержит обращение к одному набору данных, расположенному на удаленном узле. Затем каждый подзапрос становится в очередь передачи данных соответствующего канала. Если канал свободен, то начинается передача, иначе подзапрос ожидает освобождения канала. После завершения передачи каждый подзапрос становится в очередь ожидания обработки. Если узел свободен, то подзапрос начинает обрабатываться, в противном случае он ожидает освобождения узла. При обработке подзапроса происходит формирование ответа. Далее происходит передача ответа на узел, инициировавший подзапрос аналогично процедуре передачи самого подзапроса. Когда ответы на все подзапросы переданы, завершается выполнение запроса.

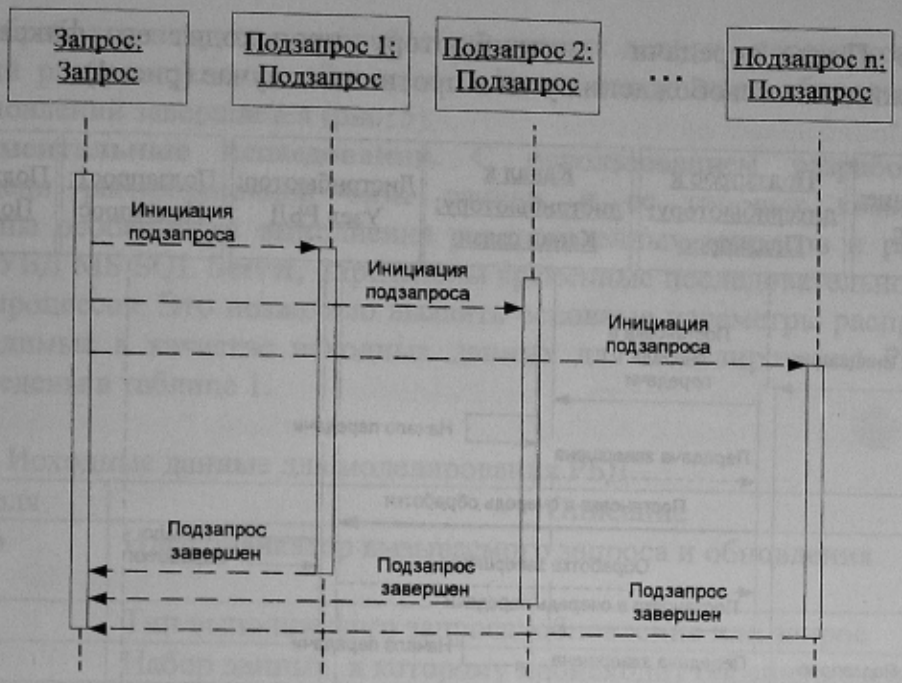

Рисунок 2 - Взаимодействие объектов классов "Запрос" и "Подзапрос"

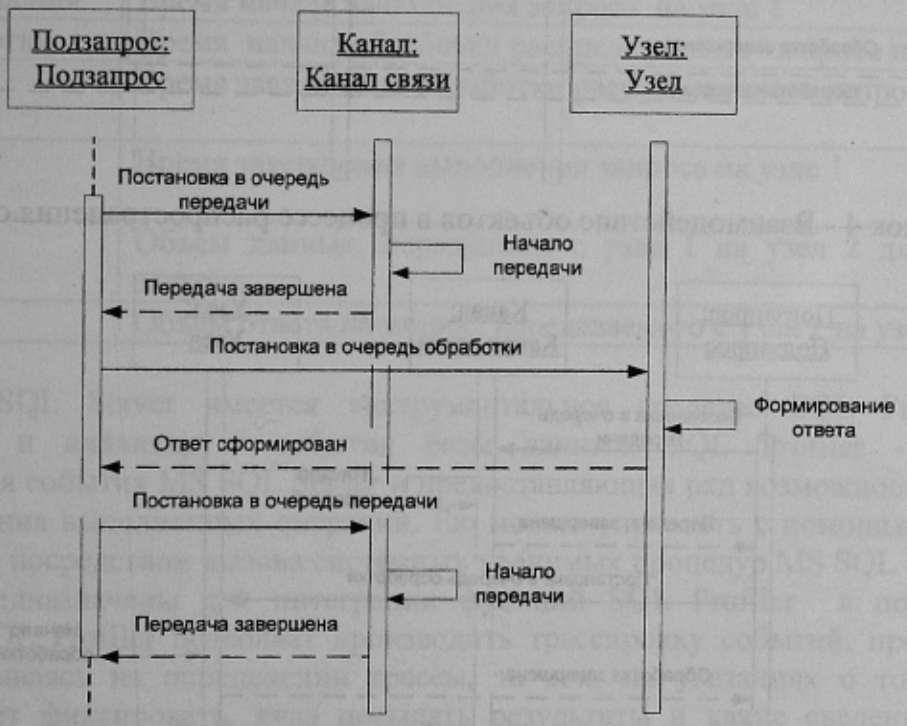

Рисунок 3 - Взаимодействие объектов классов "Подзапрос", "Канал передачи данных", "Узел" в процессе выполнения распределенных запросов

Рассмотрим теперь процесс распространения обновления. Этот процесс изображен на рис. 4 и 5 в виде UML диаграмм с учетом временной составляющей.

Данный процесс происходит следующим образом. На узле инициируется обновление. Далее обновление инициирует подзапрос к дистрибьютору, который становится в очередь передачи канала, соединяющего узел, инициировавший обновление, с узлом-дистрибьютором для набора данных, в котором было произведено изменение. Если канал занят, то подзапрос становится в очередь передачи данного канала, иначе передается на узел

- дистрибьютор. После передачи дистрибьютору, происходит его фиксация, если узел свободен, или ожидание освобождения узла в противном случае (рис. 4).

Рисунок 4 - Взаимодействие объектов в процессе распространения обновлений

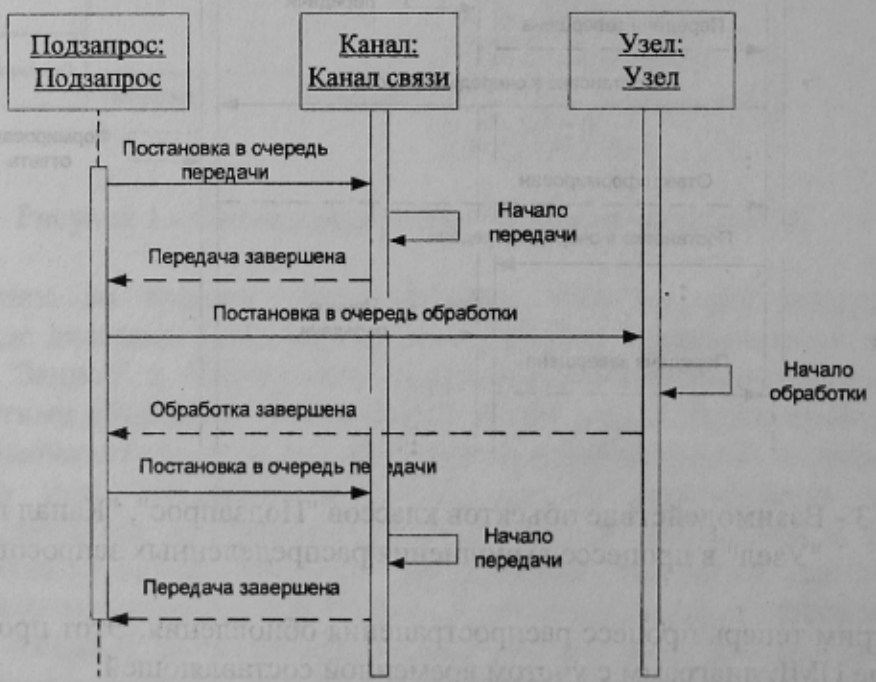

Рисунок 5 - Взаимодействие объектов классов "Подзапрос", "Канал передачи данных", "Узел" в процессе распространения обновлений

На узле дистрибьюторе формируется множество подзапросов, которые с дистрибьютора по каналам передачи данных передаются всем узлам, содержащим копии с обновляемыми данными, аналогичным способом. После того, как подзапрос на обновление

был принят и обработан на узле с копией обновляемых данных, передается ответ на узел, инициировавший распространение обновления. После завершения обновления всех копий выполнение обновлений завершается (рис. 5).

Экспериментальные исследования. С использованием разработанной общей объектной модели распределенной базы данных и ее типовых компонентов были проанализированы особенности выполнения распределенных запросов и распространения обновлений в СУБД MS SQL Server, определены временные последовательности каждого из происходящих процессов. Это позволило выявить основные параметры распределенных баз данных, необходимые в качестве исходных данных для моделирования РБД. Указанные параметры приведены в таблице 1.

Таблица 1. Исходные данные для моделирования РБД

Название поля	Описание
Идентификатор запроса	Идентификатор вызываемого запроса и обновления
Тип запроса	Тип выполняемого запроса: обновление или запрос
Набор данных	Набор данных, к которому происходит обращение в подзапросе
Узел 1	Имя узла, на котором инициирован запрос
Узел 2	Имя узла, к которому обращен запрос
Начало выполнения	Время начала выполнения запроса на узле 1
Начало обработки	Время начала обработки распределенного запроса на узле 2
Завершение обработки	Время завершения обработки распределенного запроса на узле 2
Завершение выполнения	Время завершения выполнения запроса на узле 1
Объем 1	Объем данных, переданных с узла 1 на узел 2 для выполнения запроса
Объем 2	Объем ответа на запрос, передаваемого с узла 2 на узел 1

В MS SQL Server имеется инструментальное средство SQL Profiler, которое протоколирует и анализирует события базы данных. SQL Profiler - это утилита, регистрирующая события MS SQL Server и предоставляющая ряд возможностей для анализа и воспроизведения выполненных операций. Ею можно управлять с помощью графического интерфейса или посредством вызова системных хранимых процедур MS SQL Server, которые специально предназначены для интеграции функций SQL Profiler в пользовательские приложения. SQL Profiler позволяет производить трассировку событий, происходящих на сервере, основываясь на определении трассы, то есть на указаниях о том, какие типы событий следует фиксировать, куда посылать результаты и какие сведения необходимо включать. Поэтому, в качестве инструмента сбора необходимых для моделирования РБД исходных данных была выбрана утилита SQL Profiler.

Для проведения экспериментальных исследований разработанной модели РБД в качестве объекта была выбрана компьютерная информационная система компании “Конти” (г. Донецк, Украина). Данное предприятие является крупным производителем кондитерской продукции на Украине. В состав компании “Конти” входят 4 фабрики: – три в Украине (г.г. Донецк, Константиновка, Горловка) и одна в России (г. Курск). Компания имеет распределенную систему сбыта, состоящую из пяти филиалов (складов продукции), из них четыре в Украине – в г.г. Донецк, Киев, Львов, Николаев, и один филиал в России – г. Воронеж, а также несколько региональных представительств. Информационная система компании “Конти” имеет распределенную архитектуру, построенную на РБД, в состав которой входит 10 узлов: центральный узел (корпоративный сервер), по одному узлу на

каждой фабрике и в каждом филиале. В данной распределенной информационной системе используется СУБД Microsoft SQL Server.

На основе реальных параметров компьютерной информационной системы компании "Конти", с помощью разработанной модели РБД были проведены вычислительные эксперименты и выполнена статистическая обработка полученных результатов. Для проверки адекватности модели РБД проводилось сравнение длительностей выполнения запросов и распространения обновлений, полученных с ее помощью, с реальными показателями, полученными с использованием утилиты SQL Profiler. Расхождение результатов моделирования с реальными данными составило менее 10%.

Таким образом, результаты экспериментальных исследований показали, что разработанная объектная модель РБД правильно отражает основные процессы, протекающие в распределенных информационных системах и обеспечивает достаточную степень точности при моделировании реальных РБД.

Выводы.

1. Разработана новая объектная модель РБД, которая учитывает особенности СУБД Microsoft SQL Server при выполнении распределенных запросов и распространении обновлений, что обеспечивает повышение точности и достоверности моделирования.

2. Анализ процессов, протекающих в РБД позволил определить перечень параметров РБД, которые являются исходными данными для модели. В качестве инструментального средства сбора статистических данных о работе РБД выбрана утилита SQL Profiler.

3. В качестве объекта для моделирования выбрана распределенная компьютерная информационная система компании "Конти" (г. Донецк), в которой используется СУБД Microsoft SQL Server. Экспериментальные исследования на реальных данных подтвердили работоспособность и адекватность модели.

4. Разработанная объектная модель РБД может использоваться как для проведения анализа функционирования распределенных баз данных с целью выявления "узких мест" в системе, так и для оптимизации распределения данных по узлам РБД совместно с одним из методов оптимального поиска, например генетическими алгоритмами.

Литература

1. Цегелик Г.Г. Системы распределенных баз данных. – Львов: Свит, 1990. – 168 с.
2. Мамиконов А.Г., Кульба В.В., Косяченко С.А., Ужастов И.А. Оптимизация структур распределенных баз данных в АСУ. – М.: Наука, 1990.– 240 с.
3. Галкин В.Е. Методы оптимальной организации распределенной информационной системы. // Автоматизация и современные технологии. – 2004. – № 4. – С. 13 – 17.
4. Телятников А.О. Разработка объектной модели распределенной базы данных // Наукові праці Донецького національного технічного університета. Випуск 74. – Донецьк: ДонНТУ, 2004. – С. 192–200.
5. Шпеник М., Следж О. Руководство администратора баз данных Microsoft SQL Server 2000. – М.: Издательский дом "Вильямс", 2004 – 928 с.
6. Буч Г., Рамбо Дж., Джекобсон А. Язык UML. Руководство пользователя.: Пер. с англ. – М.: ДМК, 2000. – 432 с.