Лекция. **Технологии распределенного хранения данных** (продолжение)

1. Модель представления распределенных данных
2. Проектирование распределенных баз данных
3. Управление распределенными данными

1. Модель представления распределенных данных

С концепцией баз данных тесно связана известная идея многоуровневого представления данных, предложенная исследовательской группой в области баз данных *ANSI*/*SPARC*. Структурная основа этого представления включает в себя три уровня, каждому из которых ставится в соответствие модель данных.

Очевидно, что такие сложные образования как распределенные базы данных могут потребовать большего числа уровней представления данных для реализации принципа независимости описания структуры баз от данных. Действительно, рассмотренная трехуровневая архитектура в случае распределенной базы данных расширена до пяти уровней (рис. 1).

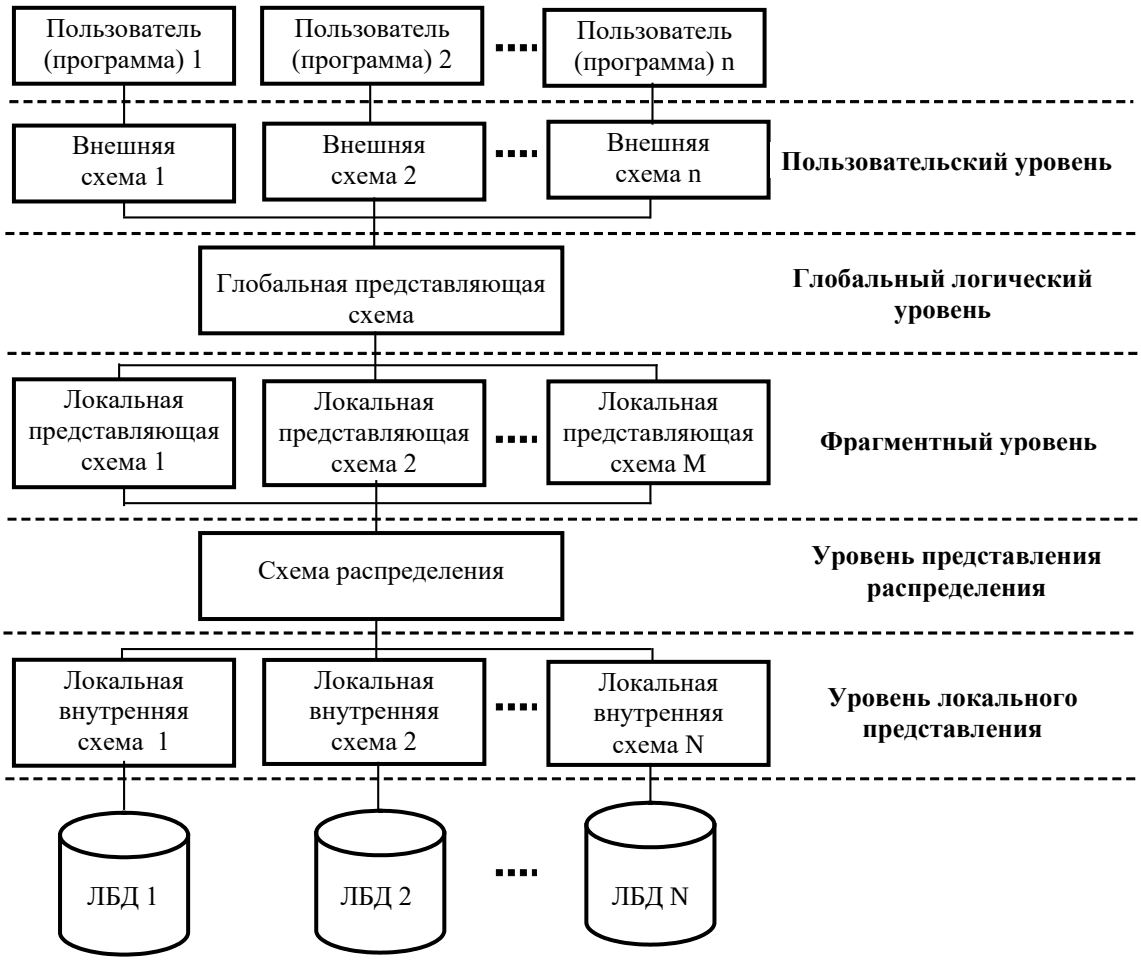

Рис. 1. Уровни представления данных в РБД

• *Пользовательский уровень представления данных*. Служит для описания части базы данных, доступной одному конкретному пользователю или группе пользователей, рассматриваемых как один. Эта часть многоуровневого представления является, по сути, внешней моделью в концепции ANSI. Каждый

пользователь может иметь отличное от других пользователей представление, соответствующее его требованиям и требованиям разграничения доступа.

- *Глобальный логический уровень представления данных.* Этот уровень подобен концептуальному уровню представления концепции *ANSI*. Используется для описания логической структуры всей распределенной базы данных, то есть в представлении администратора РБД. При описании РБД этому уровню ставится в соответствие глобальная представляющая схема.
- Существование третьего и четвертого уровней объясняется распределенной природой базы данных.
- *Фрагментный уровень представления данных*. Используя этот уровень, администратор РБД определяет несвязанные подмножества базы данных, то есть логические фрагменты, и описывает их средствами СУБД в виде локальных представляющих схем.
- *Уровень представления распределения данных*. На данном уровне определяется географическое расположение экземпляров каждого логического фрагмента. Уровень допускает существование нескольких физических фрагментов, соответствующих логическому фрагменту. Этому уровню соответствует схема распределения данных.
- *Уровень локального представления данных*. Соответствует описанию той части базы данных, которая существует в конкретном узле. Несомненно, что эта локальная база может рассматриваться как с точки зрения логической, так и физической структуры. Однако локальное представление считается описанием логической структуры, при этом физическая структура является скрытой от администратора РБД. Локальная БД, как база в полном понимании этого слова, имеет несколько уровней представления, но в данном рассмотрении эти уровни не принимают участия.

2. Проектирование распределенных баз данных

Реализация концепции РБД не может не поставить перед разработчиками распределенных информационных систем ряда проблем. Первоочередные проблемы, связанные с распределением данных в сети, подлежат разрешению на стадии проектирования РБД. Поэтому названная стадия жизненного цикла РБД имеет ряд особенностей по сравнению с проектированием локальных баз данных.

В распределенной информационной системе логически целостная база данных может быть фрагментирована и широко распределена по сети с целью улучшения производительности и надежности информационной системы. Фрагментация и распределение данных без централизованного планирования часто являются причиной несогласованного использования данных. Рассматриваемая последовательность этапов проектирования распределенной базы данных учитывает этот факт.

- *Этап анализа предметной области*. Его реализация предполагает изучение и описание предметной области, а также анализ пользовательских потребностей в информации.

- *Этап концептуального проектирования*. По результатам предыдущего этапа разрабатывается инфологическая модель (информационная структура) предметной области.
- *Этап логического проектирования* (проектирования реализации). Во время его проведения осуществляется выбор системы управления распределенной базой данных и наложение ограничений выбранной системы на информационную структуру. Результатом логического проектирования выступает глобальная структура базы данных.
- *Этап расчленения базы данных*. Рассматриваемый этап связан с делением глобальной базы данных на логические фрагменты. При решении задачи расчленения учитывают требования к обработке данных, характеристики выбранной системы, а также характеристики технических и программных средств в узлах сети. Результатами являются совокупность логических фрагментов и размер каждого из них.
- *Этап размещения базы данных*. На этом этапе решается задача выбора узлов сети для размещения в них хранимых фрагментов, соответствующих логическим фрагментам базы данных. Определенные ограничения накладывают требования по обработке данных и разграничению доступа, особенности сети передачи данных (ее топология, пропускная способность каналов), а также характеристики аппаратуры и программного обеспечения узлов вычислительной сети. Решение представляет собой перечень узлов, с каждым из которых связан список фрагментов базы данных.
- *Этап проектирования локальных баз данных*. Является заключительным в рассматриваемой последовательности. На нем осуществляется проектирование физических структур локальных баз данных, образованных в результате выполнения всех процедур предшествующих шагов.

Главное отличие перечисленной совокупности этапов от последовательности проектирования централизованных баз данных состоит в наличии этапов расчленения и размещения БД. Поэтому названные этапы заслуживают более подробного рассмотрения.

При *расчленении* исходная глобальная база разделяется на множество логических фрагментов, называемых также разделами. Естественно, что должно выполняться требование о сохранении информации, то есть разделы должны содержать все сведения, имеющиеся в исходной глобальной базе. Дополнительно на процесс формирования разделов накладываются ограничения по их допустимому размеру, времени реакции на запрос и надежности обращения. Вследствие этого в раздел рекомендуется объединять такие используемые часто совместные записи, чтобы он улучшал характеристики времени ответа на запрос. Также следует стремиться к получению требуемого уровня надежности, используя по возможности меньшую кратность дублирования, то есть степень локализации ссылок должна быть высокой при минимальном числе копий хранимых фрагментов. Допустимый размер каждого раздела, как неделимой совокупности данных, определяется фиксированным объемом памяти в каждом узле сети. И в

общем случае ограничения на класс допустимых расчленений накладывает емкость внешних запоминающих устройств в узлах.

Задача *размещения* распределенной базы данных решается сравнительно просто для двух стратегий: централизации и дублирования. Вполне очевидно, что отпадает необходимость в выполнении процедуры расчленения базы данных. Если принята первая стратегия, то перед разработчиком стоит один вопрос: в каком узле следует разместить базу данных? Реализация второй стратегии для каждого узла сети требует решения вопроса: размещать или не размещать полную копию базы? Ответы на поставленные вопросы зачастую предопределены и зависят от структуры сети, объема памяти в ее узлах, перечня альтернатив и здравого смысла.

Значительно сложнее представляется задача размещения при использовании стратегии расчленения и особенно сложной при смешанной стратегии. В случае реализации стратегии расчленения необходимо: во-первых, расчленить базу на логические фрагменты и, во-вторых, разместить каждый фрагмент в конкретном узле с учетом ограничений на размещение. Задача является итеративной и возможно, что расчленение базы данных потребуется проводить неоднократно. Если же используется смешанная стратегия, решение становится более сложным: каждый логический фрагмент может быть размещен в любом числе узлов. Количество перестановок фрагментов растет очень быстро, и это является одной из причин того, что ограничиваются нахождением не оптимального, а рационального размещения.

Решение задачи оптимального размещения фрагментов по узлам сети методами линейного программирования возможно при довольно жестких допущениях о характере потока запросов к базе, предопределенном числе и неизменности этих запросов, заранее известном числе хранимых фрагментов. Количество переменных и ограничений прогрессирует с увеличением числа узлов сети и поэтому получение решения возможно лишь для задач малой размерности. Решение также усложняется при предъявлении менее жестких требований к характеру запросов пользователей. Подходы к решению используют в своей основе метод динамического программирования. Если же типовые запросы первоначально неизвестны, то используются статистические методы для определения этих запросов, а результаты их определения служат входными данными для решения задачи размещения.

В целом процесс разработки распределенных баз данных отличается высокой трудоемкостью и значительными материальными затратами. Задача выбора наилучшего соответствия между характеристиками РБД и методами распределения данных в сети требует всестороннего анализа, так как принятые проектные решения оказывают непосредственное влияние на реализацию и последующее функционирование информационной системы.

3. Управление распределенными данными

Концепция баз данных предполагает независимое от задач пользователей накопление и ведение лишенных избыточности взаимосвязанных данных. Все функции по управлению данными и взаимодействию пользователей или прикладных программ с базами возлагаются на системы управления базами

данных, а пользователю предоставляются специальные языковые средства для описания, модификации и выборки данных. В случае распределения данных одной из проблем, подлежащих разрешению, является интеграция баз данных, созданных в рамках различных локальных СУБД, и разработка систем управления распределенными базами данных.

В общем случае система управления распределенными базами данных (СУРБД) трактуется как система, обеспечивающая возможность работы с несколькими локальными базами данных и реализующая принцип логической интеграции данных, физически распределенных между взаимосвязанными вычислительными ресурсами. В функции СУРБД входят: управление доступом к данным, анализ и распределение транзакций, сопряжение различных программных средств (операционных систем, локальных СУБД), а также управление трансляцией запросов.

Реализация функции *управления доступом* к данным обеспечивает согласованность операций, связанных с проверкой формальной правильности запросов, управлением одновременным обращением к данным многих пользователей, защитой данных. Что касается *анализа* и *распределения транзакций*, то соответствующие модули СУРБД проводят анализ поступивших от пользователей запросов, их декомпозицию на подзапросы и выбор дистанционного или локального метода доступа для каждого из этих подзапросов. Выполнение функции *сопряжения* обеспечивает порождение макрокоманд для согласования взаимодействия различных операционных систем и организации методов доступа к локальным СУБД, автоматический вызов трансляторов. Наконец, функция *трансляции* реализует проверку корректности текстов пользовательских запросов, преобразование этих текстов в форму внутреннего языка сети и языков манипулирования данными локальных СУБД.

Многообразие уже созданных к настоящему времени СУРБД предполагает их группирование с целью всесторонней характеристики.

Классификация СУРБД. В основу этой классификации могут быть положены разнообразные признаки. Представляется целесообразным указание признаков классификации с последующей краткой характеристикой выделяемых группировок.

Используемые принципы управления. Введение в рассмотрение этого признака предполагает наличие СУРБД с централизованным и децентрализованным управлением.

Централизованное управление распределенной базой данных предполагает расположение СУРБД в одном из узлов сети. Локальные СУБД подчинены этому узлу. Достоинством централизованных СУРБД несомненно является относительная простота программного обеспечения и реализации. К недостаткам следует отнести необходимость прохождения всех запросов через главный узел и отсутствие связи между локальными СУБД при выходе из строя СУРБД.

Децентрализация управления требует многократного копирования модулей СУРБД и размещения их в узлах сети. Каждая копия имеет возможность управления доступом к любой локальной базе данных. Повышение надежности таких систем и снижение транспортных расходов по сравнению с

централизованными СУРБД достигаются за счет существенного усложнения программного обеспечения. Действительно, каждая копия СУРБД в произвольный момент времени должна иметь информацию о процессах, протекающих во всех локальных базах данных, и учитывать ее при синхронизации запросов к РБД.

Принципы проектирования систем. Деление всего множества СУРБД по данному признаку позволяет обнаружить системы, базирующиеся на стандартных, промышленно эксплуатируемых типовых СУБД, и системы, в основе которых лежат уникальные разработки.

Использование стандартных СУБД имеет то неоспоримое достоинство, что позволяет объединять в рамках распределенного банка данных существующие базы при относительно небольших затратах. Но сложности разработки интерфейсов между уровнем локального представления и верхними уровнями не гарантируют высоких эксплуатационных характеристик таких систем. По существу, СУРБД рассматриваемого класса являются надстройками над существующими СУБД и обеспечивают интеграцию локальных логических представлений в рамках логической структуры распределенной базой данных. Программный аппарат выполняется в виде драйверов.

Гораздо большей эффективностью в конкретных приложениях обладают СУРБД, основанные на оригинальных разработках. Это достигается значительным удлинением периода их разработки, который начинается практически с нуля, а адаптация существующих систем к нуждам пользователей имеет свои трудности.