



ИСТОРИЯ РАЗВИТИЯ БАЗ ДАННЫХ

Берестнева О.Г. Панкрац
Д.А.

ВВЕДЕНИЕ

В истории вычислительной техники можно проследить развитие двух основных областей ее использования. Первая область — применение вычислительной техники для выполнения численных расчетов, которые слишком долго или вообще невозможно производить вручную. Вторая область — это использование средств вычислительной техники в автоматических или автоматизированных информационных системах.



Вторая область использования вычислительной техники возникла несколько позже первой. Это связано с тем, что на заре вычислительной техники возможности компьютеров по хранению информации были очень ограниченными. Поэтому появление съемных магнитных дисков с подвижными головками явилось революцией в истории вычислительной техники. С их появлением началась история систем управления данными во внешней памяти, произошел переход к использованию централизованных систем управления файлами.



НЕДОСТАТКИ ФАЙЛОВЫХ СИСТЕМ

1. Избыточность данных. Файловые системы характеризуются значительной избыточностью, поскольку нередко для решения различных задач управления используются одни и одни и те же данные, размещенные в разных файлах. Из-за дублирования данных в разных файлах память на внешних запоминающих устройствах используется неэкономно, информация одного и одного и того же объекта управления распределяется между многими файлами. При этом довольно тяжело представить общую информационную модель предметной области.



2. Несогласованность данных. Учитывая, что одна и та же информация может размещаться в разных файлах, технологически тяжело проследить за внесением изменений одновременно во все файлы. Из-за этого может возникнуть несогласованность данных, когда одно и то же поле в разных файлах может иметь разные значения.



3. Зависимость структур данных и прикладных программ. При файловой организации логическая и физическая структуры файла должны соответствовать их описанию в прикладной программе. Прикладная программа должна быть модифицирована при любом изменении логической или физической структуры файла. Поскольку изменения в одной программе часто требуют внесения изменений в другие информационно- связанные программы, то иногда проще создать новую программу, чем вносить изменения в старую. Поэтому этот недостаток файловых систем приводит к значительному увеличению стоимости сопровождения программных средств. Иногда стоимость сопровождения программных средств может достигать близко 70 % стоимости их разработки.



Эти недостатки послужили тем толчком, который заставил разработчиков информационных систем предложить новый подход к управлению информацией.

Этот подход был реализован в рамках новых программных систем, названных впоследствии Системами Управления Базами Данных (СУБД), а сами хранилища информации, которые работали под управлением данных систем, назывались базами или банками данных (БД и БнД).



ПОЯВЛЕНИЕ СУБД

История развития СУБД насчитывает более 30 лет. В 1968 году была введена в эксплуатацию первая промышленная СУБД система IMS фирмы IBM. В 1975 году появился первый стандарт ассоциации по языкам систем обработки данных — Conference of Data System Languages (CODASYL), который определил ряд фундаментальных понятий в теории систем баз данных, которые и до сих пор являются основополагающими для сетевой модели данных.

В дальнейшее развитие теории баз данных большой вклад был сделан американским математиком Э. Ф. Коддом, который является создателем реляционной модели данных. В 1981 году Э. Ф. Кодд получил за создание реляционной модели и реляционной алгебры престижную премию Тьюринга Американской ассоциации по вычислительной технике.



ПЕРВЫЙ ЭТАП — БАЗЫ ДАННЫХ НА БОЛЬШИХ ЭВМ

Первый этап развития СУБД связан с организацией баз данных на больших машинах типа IBM 360/370, ЕС-ЭВМ и мини-ЭВМ типа PDP11 (фирмы Digital Equipment Corporation — DEC), разных моделях HP (фирмы Hewlett Packard).

Базы данных хранились во внешней памяти центральной ЭВМ, пользователями этих баз данных были задачи, запускаемые в основном в пакетном режиме.

Интерактивный режим доступа обеспечивался с помощью консольных терминалов, которые не обладали собственными вычислительными ресурсами (процессором, внешней памятью) и служили только устройствами ввода-вывода для центральной ЭВМ.



Программы доступа к БД писались на различных языках и запускались как обычные числовые программы. Мощные операционные системы обеспечивали возможность условно параллельного выполнения всего множества задач. Эти системы можно было отнести к системам распределенного доступа, потому что база данных была централизованной, хранилась на устройствах внешней памяти одной центральной ЭВМ, а доступ к ней поддерживался от многих пользователей-задач.

Появляются первые языки высокого уровня для работы с реляционной моделью данных. Однако отсутствуют стандарты для этих первых языков.



ОСОБЕННОСТИ ПЕРВОГО ЭТАПА

- Все СУБД базируются на мощных мультипрограммных операционных системах (MVS, SVM, RTE, OSRV, RSX, UNIX), поэтому в основном поддерживается работа с централизованной базой данных в режиме распределенного доступа.
- Функции управления распределением ресурсов в основном осуществляются операционной системой (ОС).
- Поддерживаются языки низкого уровня манипулирования данными, ориентированные на навигационные методы доступа к данным.
- Значительная роль отводится администрированию данных.



- Проводятся серьезные работы по обоснованию и формализации реляционной модели данных, и была создана первая система (System R), реализующая идеологию реляционной модели данных.
- Проводятся теоретические работы по оптимизации запросов и управлению распределенным доступом к централизованной БД, было введено понятие транзакции.
- Результаты научных исследований открыто обсуждаются в печати, идет мощный поток общедоступных публикаций, касающихся всех аспектов теории и практики баз данных, и результаты теоретических исследований активно внедряются в коммерческие СУБД.



ВТОРОЙ ЭТАП - ЭПОХА ПЕРСОНАЛЬНЫХ КОМПЬЮТЕРОВ

Появляется множество программ, предназначенных для работы неподготовленных пользователей. Эти программы просты в использовании и интуитивно понятны: это, прежде всего, различные редакторы текстов, электронные таблицы и другие. Каждый пользователь может автоматизировать многие аспекты деятельности. И, конечно, это сказалось и на работе с базами данных. Появились программы, которые назывались системами управления базами данных и позволяли хранить значительные объемы информации, они имели удобный интерфейс для заполнения данных, встроенные средства для генерации различных отчетов. Эти программы позволяли автоматизировать многие учетные функции, которые раньше велись вручную. Компьютеры стали инструментом для ведения документации и собственных учетных функций. Это все сыграло как положительную, так и отрицательную роль в области развития баз данных.



Кажущаяся простота и доступность персональных компьютеров и их программного обеспечения породила множество дилетантов. Эти разработчики, считая себя знатоками, стали проектировать недолговечные базы данных, которые не учитывали многих особенностей объектов реального мира. Много было создано систем-однодневок, которые не отвечали законам развития и взаимосвязи реальных объектов. Однако доступность персональных компьютеров заставила пользователей из многих областей знаний, которые ранее не применяли вычислительную технику в своей деятельности, обратиться к ним. И спрос на развитые удобные программы обработки данных заставлял поставщиков программного обеспечения поставлять все новые системы, которые принято называть настольными (desktop) СУБД. Значительная конкуренция среди поставщиков заставляла совершенствовать эти системы, предлагая новые возможности, улучшая интерфейс и быстродействие систем, снижая их стоимость. Наличие на рынке большого числа СУБД, выполняющих сходные функции, потребовало разработки методов экспорта- импорта данных для этих систем и открытия форматов хранения данных.



ОСОБЕННОСТИ ВТОРОГО ЭТАПА

- ❑ Все СУБД были рассчитаны на создание БД в основном с монопольным доступом. И это понятно. Компьютер персональный, он не был подсоединен к сети, и база данных на нем создавалась для работы одного пользователя. В редких случаях предполагалась последовательная работа нескольких пользователей, например, сначала оператор, который вводил бухгалтерские документы, а потом главбух, который определял проводки, соответствующие первичным документам.
- ❑ Большинство СУБД имели развитый и удобный пользовательский интерфейс, в большинстве существовал интерактивный режим работы с БД, как в рамках описания БД, так и в рамках проектирования запросов. Кроме того, большинство СУБД предлагали развитый и удобный инструментарий для разработки готовых приложений без программирования. Инструментальная среда состояла из готовых элементов приложения в виде шаблонов экранных форм, отчетов, этикеток (Labels), графических конструкторов запросов, которые достаточно просто могли быть собраны в единый комплекс.
- ❑ Во всех настольных СУБД поддерживался только внешний уровень представления реляционной модели, то есть только внешний табличный вид структур данных.



- При наличии высокоуровневых языков манипулирования данными типа реляционной алгебры и SQL в настольных СУБД поддерживались низкоуровневые языки манипулирования данными на уровне отдельных строк таблиц.
- В настольных СУБД отсутствовали средства поддержки ссылочной и структурной целостности базы данных. Эти функции должны были выполнять приложения, однако скудость средств разработки приложений иногда не позволяла это сделать, и в этом случае эти функции должны были выполняться пользователем, требуя от него дополнительного контроля при вводе и изменении информации, хранящейся в БД.
- Наличие монопольного режима работы фактически привело к вырождению функций администрирования БД и в связи с этим — к отсутствию инструментальных средств администрирования БД.
- И, наконец, последняя и в настоящий момент весьма положительная особенность — это сравнительно скромные требования к аппаратному обеспечению со стороны настольных СУБД. Вполне работоспособные приложения, разработанные, например, на Clipper, работали на PC 286.



ТРЕТИЙ ЭТАП - РАСПРЕДЕЛЕННЫЕ БАЗЫ ДАННЫХ

Хорошо известно, что история развивается по спирали, поэтому после процесса «персонализации» начался обратный процесс — интеграция. Умножится количество локальных сетей, все больше информации передастся между компьютерами, остро встает задача согласованности данных, хранящихся и обрабатываемых в разных местах, но логически друг с другом связанных, возникают задачи, связанные с параллельной обработкой транзакций — последовательностей операций над БД, переводящих ее из одного непротиворечивого состояния в другое непротиворечивое состояние. Успешное решение этих задач приводит к появлению распределенных баз данных, сохраняющих все преимущества настольных СУБД и в то же время позволяющих организовать параллельную обработку информации и поддержку целостности БД.



ОСОБЕННОСТИ ТРЕТЬЕГО ЭТАПА

- Практически все современные СУБД обеспечивают поддержку полной реляционной модели, а именно:
 - структурной целостности — допустимыми являются только данные, представленные в виде отношений реляционной модели;
 - языковой целостности, то есть языков манипулирования данными высокого уровня (в основном SQL);
 - ссылочной целостности — контроля за соблюдением ссылочной целостности в течение всего времени функционирования системы, и гарантий невозможности со стороны СУБД нарушить эти ограничения.
- Большинство современных СУБД рассчитаны на многоплатформенную архитектуру, то есть они могут работать на компьютерах с разной архитектурой и под разными операционными системами, при этом для пользователей доступ к данным, управляемым СУБД, на разных платформах практически неразличим.



- Необходимость поддержки многопользовательской работы с базой данных и возможность децентрализованного хранения данных потребовали развития средств администрирования БД с реализацией общей концепции средств защиты данных.
- Потребность в новых реализациях вызвала создание серьезных теоретических трудов по оптимизации реализации распределенных БД и работе с распределенными транзакциями и запросами с внедрением полученных результатов в коммерческие СУБД.
- Для того чтобы не потерять клиентов, которые ранее работали на настольных СУБД, практически все современные СУБД имеют средства подключения клиентских приложений, разработанных с использованием настольных СУБД, и средства экспорта данных из форматов настольных



СУБД ТРЕТЬЕГО ЭТАПА РАЗВИТИЯ

К этому этапу можно отнести разработку ряда стандартов в рамках языков описания и манипулирования данными (SQL89, SQL92, SQL99) и технологий по обмену данными между различными СУБД, к которым можно отнести и протокол ODBC (Open DataBase Connectivity), предложенный фирмой Microsoft.

Так же к этому этапу можно отнести начало работ, связанных с концепцией объектно-ориентированных БД — ООБД. Представителями СУБД, относящимся ко второму этапу, можно считать MS Access 97 и все современные серверы баз данных Oracle 7.3, Oracle 8.4, MS SQL 6.5, MS SQL 7.0, System 10, System 11, Informix, DB2, SQL Base и другие современные серверы баз данных, которых в настоящий момент насчитывается несколько десятков.



ЧЕТВЕРТЫЙ ЭТАП - ПЕРСПЕКТИВЫ РАЗВИТИЯ СИСТЕМ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ

Этот этап характеризуется появлением новой технологии доступа к данным— интранет. Основное отличие этого подхода от технологии клиент-сервер состоит в том, что отпадает необходимость использования специализированного клиентского программного обеспечения. Для работы с удаленной базой данных используется стандартный браузер Internet, например Microsoft Internet Explorer, и для конечного пользователя процесс обращения к данным происходит аналогично использованию Internet.

При этом встроенный в загружаемые пользователем HTML-страницы код, написанный обычно на языках Java, Java-script, Perl и других, отслеживает все действия пользователя и транслирует их в низкоуровневые SQL-запросы к базе данных, выполняя, таким образом, ту работу, которой в технологии клиент-сервер занимается клиентская программа.



Удобство данного подхода привело к тому, что он стал использоваться не только для удаленного доступа к базам данных, но и для пользователей локальной сети предприятия. Простые задачи обработки данных, не связанные со сложными алгоритмами, требующими согласованного изменения данных во многих взаимосвязанных объектах, достаточно просто и эффективно могут быть построены по данной архитектуре.

В этом случае для подключения нового пользователя к возможности использовать данную задачу не требуется установка дополнительного клиентского программного обеспечения. Однако алгоритмически сложные задачи рекомендуется реализовывать в архитектуре «клиент-сервер» с разработкой специального клиентского программного обеспечения.



ЗАКЛЮЧЕНИЕ

У каждого из вышеперечисленных подходов к работе с данными есть свои достоинства и свои недостатки, которые и определяют область применения того или иного метода, и в настоящее время все подходы широко используются.



ВОПРОСЫ

1. Назвать 2 области использования вычислительной техники.
2. Недостатки файловых систем
3. Год запуска и название первой СУБД
4. Охарактеризовать II этап
5. Охарактеризовать IV этап

