

Министерство образования и науки Украины  
Харьковский национальный университет  
радиоэлектроники

Мазурова О.А., Широкопетлева М.С., Черепанова Ю.Ю.



## Информационные системы

Харьков

2009

## Содержание

Введение.....	3
Теория.....	4
Классификации информационных систем.....	4
Сравнительный анализ СУБД для информационных систем корпоративного вида .....	11
Основные концепции СУБД Oracle.....	14
Выводы.....	19
Перечень ссылок.....	20

## Вводная лекция

Современная информационная система в масштабе корпорации - это комбинация, тесное переплетение различных информационных технологий, предлагаемых сегодня на рынке. Искусство создания таких систем - в сбалансированной интеграции этих технологий и соответствующих программных и аппаратных средств.

Построение корпоративных систем - дело не одной недели и даже не одного года. Это не просто покупка компьютеров и связывание их между собой. Это, прежде всего, осмысление своего бизнеса, понимание путей его развития и неизбежный вывод о том, что успех в настоящем и будущем может быть обеспечен только в случае, если удастся правильно организовать управление информацией. Это возможно сделать через корпоративную информационную систему, которая должна стать не только основой информационного пространства компании, но и гибким инструментом управления бизнесом в сложных, постоянно меняющихся условиях.

В связи с этим вводная лекция посвящена рассмотрению общих вопросов создания информационных систем корпоративного типа - систем, в которых требуется обрабатывать значительные объемы информации и поддерживать большие группы пользователей, охватывая, таким образом, территориально разбросанные предприятия и организации.

В этой лекции рассматриваются существующие классификации информационных систем, проводится сравнительный анализ СУБД, которые могут быть использованы для создания информационных систем корпоративного типа. На основании анализа таких популярных СУБД, как MySQL, Microsoft SQL Server 2000, Oracle и т.п. показаны преимущества и недостатки СУБД Oracle. Рассматриваются основные концепции такого популярного на сегодняшний день СУБД, как Oracle.

## Теория

### Классификации информационных систем

**Информационные системы** - программные системы, которые решают прикладные задачи на основании использования широко известных баз данных (БД). Информационные системы бывают разные по назначению и архитектуре. Мы рассмотрим, какие бывают информационные системы, и выделим, возможно, наиболее сложный вид информационных систем - информационные системы уровня корпораций - системы, которые работают в очень активном режиме обработки информации; которые охватывают по территории города и даже целые регионы. Они требуют больших БД и серьезных СУБД.

Информационные системы бывают нескольких видов. Существуют классификации информационных систем:

- по назначению;
- по архитектуре;
- по уровню организаций, на которых системы будут использоваться;
- по режиму обработки информации;
- по объему поддерживаемых баз данных.

Итак, с учетом реализуемой архитектуры, информационные системы можно разбить на:

- Объектные или бортовые (ЭВМ для стиральных машин, маленькие процессоры и т.п.) ;
- Большие;
- Системы, основанные на технологии клиент-сервер .

С учетом уровня организаций, на которых системы будут использоваться, выделяют следующие информационные системы:

- Enterprise - wide DB (корпорации) - работают в очень активном режиме обработки информации ;

- Department - wide DB (уровня подразделения) - меньшие объёмы информации, несколько серверов сосредоточено в одном месте (здании, городе) ;

-Workgroup DB (уровня рабочей группы) - один сервер для разработки программной документации или программного проекта.

По объему поддерживаемых баз данных информационные системы делятся на:

- Малые ( <10 Мб ) ;
- Средние (10 Мб) ;
- Большие ( » 10 Гб ) ;
- Сверхбольшие ( > 10 Тб ) .

По режиму обработки информации информационные системы делятся на:

- Системы онлайн-обработки транзакций (например, продажа билетов) - On - line Transaction Processing ( OLTP ). Основные транзакции - обработка информации. Изменения в БД происходят не очень часто;

- Decision Support System. Основные операции - выборки. Информация носит статический характер;

- Системы аналитической обработки информации - On - line Analytical Processing ( OLAP ). Используются при сложной математической обработке информации, создаются при других системах как дополнительные сервера. Такие системы используют архивы - Data Mart (склад).

Неразбериха с документами - болезненная проблема для любой компании. Поэтому система автоматизации документооборота, которая позволяет автоматизировать ручные, рутинные операции, автоматически передавать и отслеживать перемещение документов внутри корпорации, контролировать выполнение поручений, связанных с документами и т.д. - одна из важнейших составляющих информационной системы.

Можно выделить два класса подобных систем:

- системы workflow;
- системы groupware.

*Основное назначение* систем обоих классов - автоматизация и поддержка коллективной работы в офисе, однако, есть некоторые различия:

- системы класса groupware ориентированы на автоматизацию работы небольшого коллектива и поддерживают корректное разделение (т.е. совместное использование) информации группой пользователей.

- системы класса workflow ориентированы на автоматизацию работы корпорации и поддерживают разделение работ , т.е. выполнение одной "большой" работы группой исполнителей.

Системы *workflow* и *groupware* не конкурируют между собой, а скорее дополняют друг друга. Выбор одной из них, а также использование их в комбинации определяются задачами,

решаемыми корпорацией. Если вы заботитесь о повышении эффективности работы каждого отдельного сотрудника в коллективе, предпочтение стоит отдать системам класса groupware (например, продуктом Lotus Notes (Lotus Development)). Если же вы более серьезно обеспокоены повышением эффективности работы всей организации в целом, то вам следует остановить свой выбор на системах класса workflow (системы Staffware (Staffware plc) и Action Workflow (Action Technologies)).

С учетом выше рассмотренных классификаций, в данном курсе мы будем ориентироваться на корпоративные информационные системы - системы, которые создаются для информационной поддержки достаточно больших организаций, требующих больших БД и т.п..

Хотя, в принципе, все организации весьма похожи друг на друга. В структуру каждой из них, независимо от рода деятельности, входят многочисленные подразделения, непосредственно осуществляющие тот или иной вид деятельности компании, а также дирекция, бухгалтерия, канцелярия и т.п. Подразделения компании обмениваются между собой информацией, а некоторые из подразделений взаимодействуют с внешними партнерами (см. рис.1).

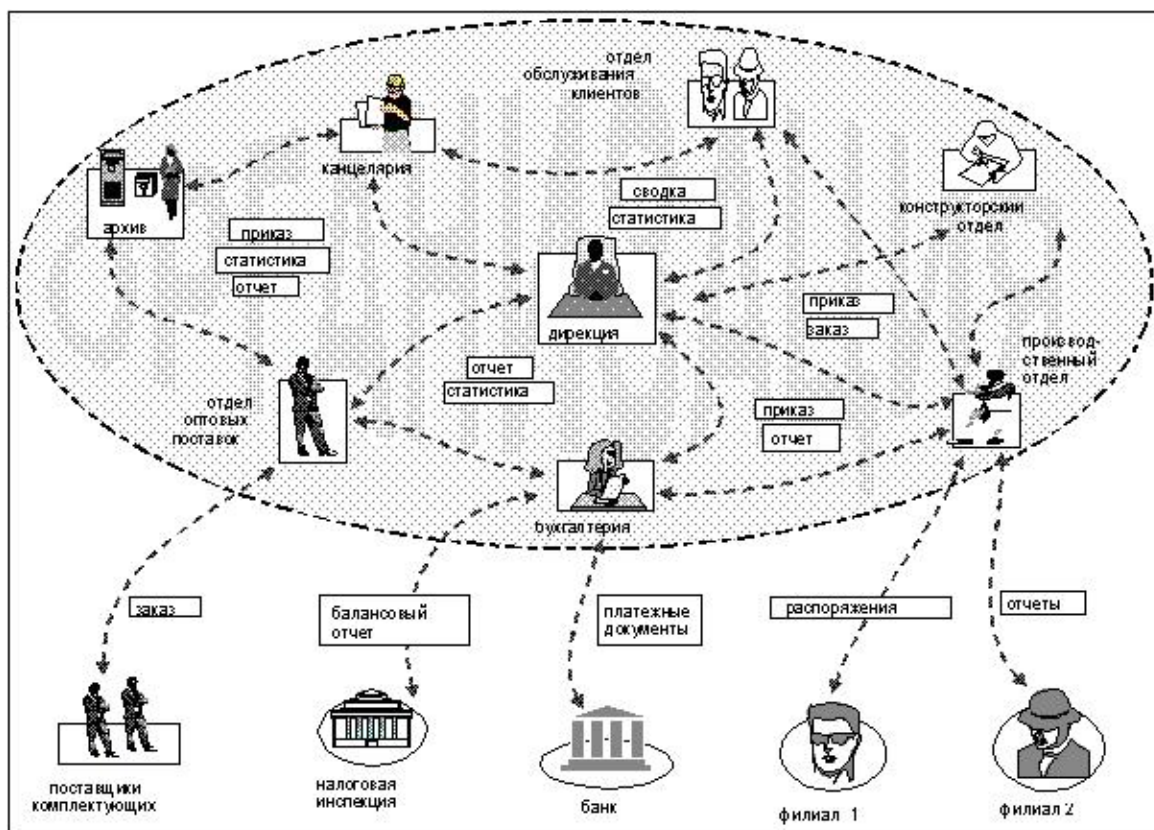


Рисунок 1 - Типовая схема организации («Корпоративные информационные системы: технологии и решения» М.С. Каменова /СУБД, № 3/95)

Таким образом, любая организация - это совокупность взаимодействующих элементов (подразделений), каждый из которых может иметь свою структуру. Элементы связаны между

собой функционально и информационно, обмениваясь документами, факсами, письменными и устными распоряжениями. Кроме того, эти элементы взаимодействуют с внешними системами, причем их взаимодействие также может быть как информационным.

Такой общий взгляд на организацию позволяет сформулировать некоторые общие принципы построения корпоративных информационных систем, т.е. информационных систем в масштабе всей организации.

### **Этапы создания корпоративных систем**

При создании корпоративной информационной системы необходимо выполнить следующие этапы:

- Провести информационное обследование организации.
- По результатам обследования выбрать архитектуру системы и аппаратно-программные средства ее реализации, а именно:
  - систему управления корпоративной базой данных (СУБД корпоративного вида);
  - систему автоматизации деловых операций и документооборота и т.д.;
  - Проектирование базы данных;
  - Программная реализация корпоративной информационной системы.

Информационная система нужна организации для того, чтобы обеспечивать информационно-коммуникационную поддержку ее основной и вспомогательной деятельности. Поэтому прежде, чем вести речь о структуре и функциональном наполнении информационной системы, необходимо разобраться в целях и задачах самой организации, чтобы понять, что же нужно автоматизировать.

Ответы на эти вопросы можно получить только после детального информационного обследования компании. Результатом обследования являются модели деятельности компании и ее информационной инфраструктуры, на базе которых разрабатываются проект корпоративной информационной системы, требования к программно-аппаратным средствам и спецификации на разработку прикладного программного обеспечения.

И прежде всего, при выработке требований к программно-аппаратным средствам необходимо выбрать архитектуру системы. Для корпоративных систем рекомендуется архитектура клиент/сервер.

В качестве аргумента приведем данные о тенденциях среди западных компаний: к середине 1994 года на эту архитектуру перешли 45% организаций в США, 50% в Европе, 29% в Японии.

Интересно также отметить причины, по которым этот переход происходит. Из 300 американских компаний, работающих в среде клиент/сервер, на вопрос "Почему вы перешли на эту архитектуру", 67% ответили: "Архитектура клиент/сервер предоставляет технологию доступа конечного пользователя к информации в масштабах предприятия". Таким образом, архитектура клиент/сервер позволяет создать единое информационное пространство, в котором конечный пользователь имеет своевременный и беспрепятственный (но санкционированный!) доступ к корпоративной информации.

## Выбор СУБД

Выбор системы управления для корпоративной базы данных - один из ключевых моментов в разработке информационной системы. На Украинском рынке присутствуют практически все СУБД, принадлежащие к элитному классу - Oracle, Informix, Sybase, MS SQL Server. Вопрос, какую СУБД использовать, можно решить только по результатам предварительного обследования и получения информационных моделей деятельности компании.

На рынке также представлен достаточно большой набор средств разработки приложений, которые ориентированы как на конкретные СУБД (например, Developer/2000 для Oracle, NewEra для Informix и т.д.), так и могут быть использованы в различных средах. В качестве примера таких универсальных инструментальных средств можно назвать PowerBuilder Enterprise (PowerSoft Corp.), Gupta SQLWindows (Gupta Corp.), Delphi (Borland Int.), Enterprise Developer (Symantec).

## Основы конфигурирования серверов баз данных

Одним из наиболее распространенных классов прикладных систем для серверов, выпускаемых большинством компаний-производителей компьютерной техники, являются системы управления базами данных (СУБД). Серверы СУБД значительно более сложны, чем серверы сетевых файловых систем NFS. Стандартный язык реляционных СУБД (SQL) намного богаче, чем набор операций NFS. Более того, имеется несколько популярных коммерческих реализаций СУБД, доступных на серверах различных компаний, каждая из которых имеет совершенно различные характеристики. Вследствие этого последующий материал будет носить достаточно общий характер.

Дело в том, что почти невозможно корректно ответить на вопрос: "Сколько пользователей данного типа будет поддерживать данная система?". В общем случае скорее можно решить, что определенная конфигурация системы не может выполнить данную задачу, чем решить, что данная конфигурация сможет с ней справиться. Например, достаточно просто определить, что система с одним дисковым накопителем не сможет достичь пропускной способности в 130 обращений в секунду при выполнении операций произвольного доступа к диску, поскольку один диск за одну секунду сможет обработать только 65 таких обращений. Однако система с двумя такими дисками либо сможет, либо не сможет справиться с такой нагрузкой, поскольку может случиться, что в системе имеется какое-либо другое узкое место, вовсе не связанное с дисковой



подсистемой.

Как приложения, ориентированные на использование баз данных, так и сами СУБД сильно различаются по своей организации. Если системы на базе файловых серверов сравнительно просто разделить по типу рабочей нагрузки на два принципиально различных класса (с интенсивной обработкой атрибутов файлов и с интенсивной обработкой самих данных), то провести подобную классификацию среди приложений баз данных и СУБД просто невозможно.

Хотя на сегодня имеется целый ряд различных архитектур баз данных, рынок UNIX-систем, кажется, остановился главным образом на реляционной модели. Абсолютное большинство инсталлированных сегодня систем реляционные, поскольку эта архитектура выбрана такими производителями как Oracle, Sybase, Ingres, Informix, Progress, Empress и DBase. ADABAS компании Software AG - иерархическая система, хотя может обрабатывать стандартный SQL.

Но даже с учетом того, что подавляющее большинство систем работает по одной и той же концептуально общей схеме, между различными продуктами имеются большие архитектурные различия. Возможно наиболее существенным является реализация самой СУБД.

Можно выделить два основных класса систем: системы, построенные по принципу "2N" (или "один-к-одному"), и многопоточные системы (рис. 2). В более старых 2N-реализациях для каждого клиента на сервере используется отдельный процесс, даже если программа-клиент физически выполняется на отдельной системе. Таким образом, для работы каждого клиентского приложения используются два процесса - один на сервере и один на клиентской системе. Многопоточные приложения как раз и разработаны для того, чтобы существенно снизить дополнительные расходы на организацию управления таким большим количеством процессов. Обычно они предполагают наличие одного кластера из нескольких процессов (от одного до пяти), работающих на серверной системе. Эти процессы имеют внутреннюю многопоточную организацию, что обеспечивает обслуживание запросов множества клиентов. Большинство основных поставщиков СУБД в настоящее время используют многопоточную реализацию или двигаются в этом направлении.

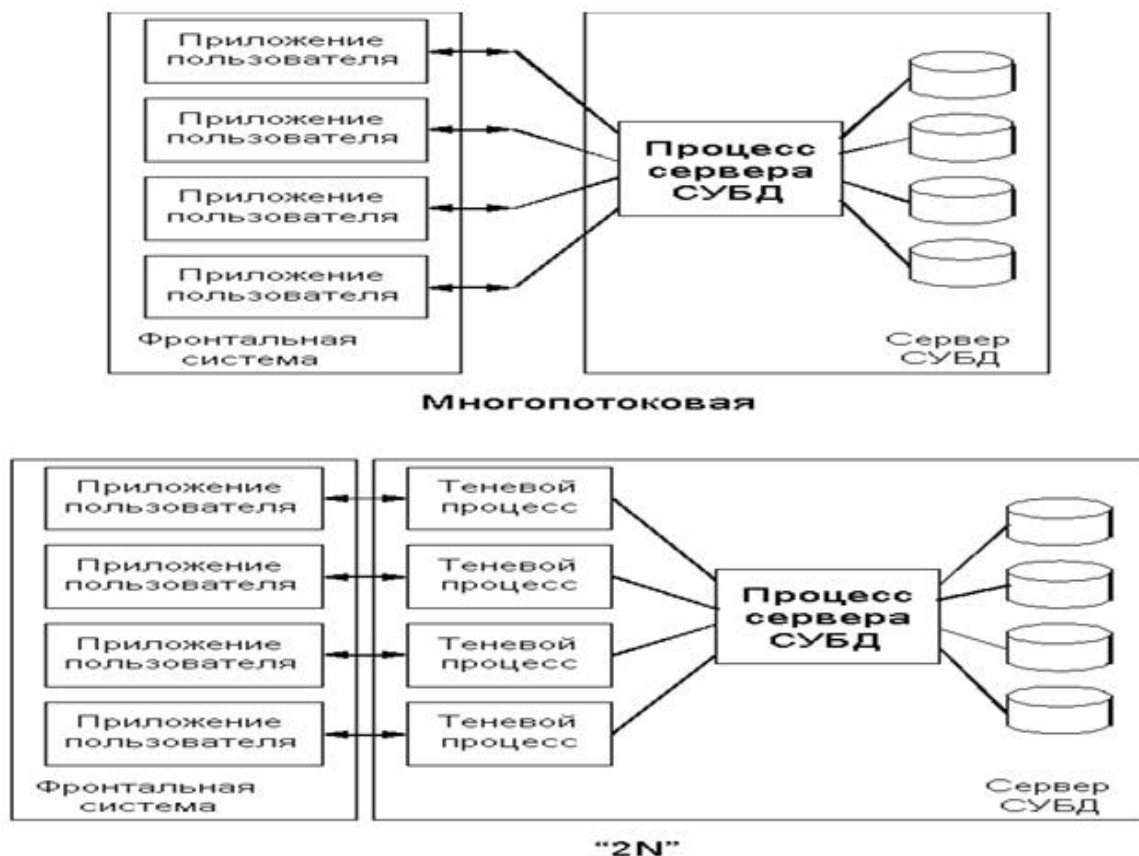


Рисунок 2 - Многопоточковая и 2N системы

Таблица 1 - Архитектура СУБД по поставщикам и версиям

Многопоточковая	2N
Oracle Version 7 (Optional)	Oracle Version 6 и по умолчанию Oracle Version 7 и выше
Sybase v4.9.x, System10	Informix v5.x
Ingres v6.x	Ingres v5.x
Progress version 7	Progress version 6
ADATABASE version 2.1	

В данном курсе мы будем изучать СУБД Oracle. Чтобы ответить на вопрос почему именно эта СУБД, рассмотрим расклад корпоративного рынка СУБД. К 1995 году Oracle занимал около 30% мирового рынка, доля лидерство с DB 2. MS SQL находился на 2 месте, занимая 11% мирового рынка корпоративных СУБД. Но у DB 2 нет MyFame -машины. В данных условиях рынка MS SQL и Oracle являются конкурентами.

К 2001 году компания IBM выдвигается за счет покупки Sybas, при этом Microsoft теряет позиции. В сентябре 2001 года появляется Oracle 9.0 (занимает около 40% мирового рынка).

Произошло изменение рынка за последние 3 года, обусловленное агрессивной маркетинговой политикой фирмы Microsoft, активно продвигающей на рынок MS SQL Server 2005, в котором очень эффективно реализованы идеи корпорации Oracle, такие как триггеры баз данных, технология многомерного анализа данных, установление встроенных статистических функций (реализованы в Oracle 9.0). Многочисленные компании занимаются "переводом" программного обеспечения и имеющихся баз данных на платформу MS SQL Server, но системы, предъявляющие высокие требования к защите и надежности данных все же используют СУБД Oracle.

Таким образом, несмотря на свою дороговизну, СУБД Oracle является одним из лидеров на корпоративном рынке СУБД.

## **Сравнительный анализ СУБД для информационных систем корпоративного вида**

В настоящее время увеличивается спрос на компьютерные системы, разрабатываемые в архитектуре клиент/сервер. Такие системы позволяют предоставлять ресурсы и информацию сервера другим компьютерам, находящимся с ним в сети. Популярность этого подхода обеспечивает большую надежность разрабатываемых систем, а также способствует уменьшению затрат на производство и содержание такого вида систем. Одним из наиболее популярных использованием этого подхода является разработка СУБД в архитектуре клиент/сервер.

### **СУБД mySQL**

MySQL разработал Михаэль Видениус (Michael Widenius). MySQL является относительно небольшой и быстрой реляционной СУБД основанной на традициях Hughes Technologies Mini SQL (mSQL).

Последнюю версию MySQL можно скачать с [www.tcx.se](http://www.tcx.se).

#### **Основные приятные стороны пакета MySQL:**

- Многопоточность. Поддержка нескольких одновременных запросов.
- Оптимизация связей с присоединением многих данных за один проход.
- Записи фиксированной и переменной длины.
- ODBC драйвер в комплекте с исходником.
- Гибкая система привилегий и паролей.

- До 16 ключей в таблице. Каждый ключ может иметь до 15 полей.
- Поддержка ключевых полей и специальных полей в операторе CREATE.
- Поддержка чисел длиной от 1 до 4 байт (ints, float, double, fixed), строк переменной длины и меток времени.
- Интерфейс с языками C и perl.
- Основанная на потоках, быстрая система памяти.
- Утилита проверки и ремонта таблицы ( isamchk).
- Все данные хранятся в формате ISO8859\_1.
  - Все операции работы со строками не обращают внимания на регистр символов в обрабатываемых строках.
- Псевдонимы применимы как к таблицам, так и к отдельным колонкам в таблице.
  - Все поля имеют значение по умолчанию. INSERT можно использовать на любом подмножестве полей.
- Легкость управления таблицей, включая добавление и удаление ключей и полей.

### **Сервер баз данных Microsoft SQL Server**

Основными задачами системы SQL-сервер являются организация одновременного доступа к данным большого количества пользователей, а также манипуляция информацией, хранимой в базе данных. SQL-сервер поддерживает реляционную модель данных.

В системах, организованных в архитектуре клиент/сервер, поддерживается коллективный доступ к данным. Здесь каждый компьютер осуществляет операции, связанные с хранением, доступом и обработкой данных. При этом выполняемые задачи разделяются между сервером и рабочей станцией, причем, исходя из неравенства аппаратных возможностей сервера и рабочих станций, это разделение должно учитываться в создаваемых системах. Так, например, мощность компьютера-сервера отражается на скорости пересылки данных, возможности хранения большего объема информации и т.д.

Рассмотрим кратко основные принципы взаимодействия в системе SQL-сервер:

- SQL-сервер может устанавливаться на платформах Windows NT и Windows 9.x (ввиду более высокой надежности рекомендуется использование Windows NT);
- SQL-сервер выполняет функции создания объектов базы данных (таблиц, индексов и т.д.), осуществляет проверку целостности базы данных и отвечает за безопасность данных в системе;

- доступ пользователя к данным SQL-сервера обычно осуществляется с компьютера рабочей станции. При этом создаются соответствующие приложения, которые позволяют выполнять операции над данными;

- для выполнения ряда задач, связанных в основном с администрированием базы данных системы SQL-сервер, обращение к системе удобно выполнять непосредственно с самого компьютера-сервера;

- в SQL-сервер для манипулирования данными используется язык запросов Transact-SQL, который является переработанной компанией Microsoft версией языка SQL.

Система SQL-сервер сохраняет создаваемые объекты в соответствующих файлах на диске компьютера-сервера. При этом для таких объектов, как база данных, создаются специальные таблицы. В них хранится информация о различных элементах базы данных: индексах, таблицах, пользователях и т.д. Файлы базы данных сохраняются с расширением \*.MDF, а системные файлы с расширением \*.LDF. О возможностях системы SQL-сервер можно судить по некоторым цифрам, представленным в табл. 2.

Таблица 2 - Основные количественные показатели системы SQL-сервер

Показатель	Размер
Максимальный размер базы данных	1048516 терабайт
Максимальное число таблиц, определяемых в одной базе данных	2 миллиарда
Максимальное количество столбцов в одной таблице	1024
Максимальное число столбцов, которые можно определить в одном SQL-запросе	32
Максимальное количество строк	Неограниченно (определяется ресурсами сервера)
Максимальное количество индексов для каждой таблицы	250

Основные операции, связанные с управлением работой SQL-сервера, осуществляются с помощью ряда утилит, входящих в состав системы, основные из которых:

- SQL Server Books Online - приложение, предоставляющее доступ к справочной информации SQL-сервера;

- SQL Server Query1 Analyzer - утилита, позволяющая выполнять команды языка запросов Transact-SQL;

- ISQL - утилита, функционирующая в среде DOS аналогично SQL Server Query Analyzer;

- SQL Server Service Manager - утилита, выполняющая работу, связанную с запуском, остановкой и временной приостановкой работы SQL-сервера;

- SQL Server Enterprise Manager - утилита, предоставляющая пользователю возможности

администрирования SQL-сервера, доступа ко всем его объектам, а также запуска различных утилит и приложений (в том числе вышеописанных).

## Основные концепции СУБД Oracle

Программное обеспечение баз данных - это ключ к решению проблем управления информацией. Вообще говоря, система управления базой данных (СУБД) должна быть способна надежно управлять большими объемами данных в многопользовательской среде, так, чтобы все пользователи могли одновременно обращаться к одним и тем же данным. Все это должно достигаться при обеспечении высокой производительности пользователей базы данных. СУБД также должна быть защищена от несанкционированного доступа, и должна предоставлять эффективные решения для восстановления от сбоев.

Сервер ORACLE обеспечивает эффективные и действенные решения для основных средств баз данных:

- ORACLE поддерживает самые большие базы данных, потенциального размера до сотен гигабайт. Чтобы обеспечить действенный контроль за использованием дорогостоящих дисковых устройств, он предоставляет полный контроль распределения пространства.

- ORACLE поддерживает большое число пользователей, одновременно выполняющих разнообразные приложения, которые оперируют одними и теми же данными. Он минимизирует соперничество за данные и гарантирует согласованность данных.

- ORACLE поддерживает все описанные выше средства, сохраняя высокую степень суммарной производительности системы. Пользователи базы данных не страдают от низкой производительности обработки.

- На некоторых установках ORACLE работает 24 часа в сутки, не имея периодов разгрузки, ограничивающих пропускную способность базы данных. Нормальные системные операции, такие как откат базы данных, а также частичные сбои компьютерной системы, не прерывают работу с базой данных.

- ORACLE может выборочно управлять доступностью данных, как на уровне базы данных, так и на более низких уровнях. Например, администратор может отключить доступ к конкретному приложению (с тем, чтобы можно было осуществить перезагрузку данных этого приложения), не затрагивая других приложений.

- ORACLE удовлетворяет промышленно принятым стандартам по языку доступа к данным, операционным системам, интерфейсам с пользователем и сетевым протоколам. Это "открытая" система, которая защищает инвестиции заказчика. Сервер ORACLE7 был сертифицирован Национальным институтом стандартов и технологий США как 100%-совместимый со стандартом ANSI/ISO SQL89. ORACLE7 полностью удовлетворяет требованиям правительственного стандарта США FIPS127-1 и имеет "маркировщик" для подчеркивания нестандартных применений SQL. Кроме того, ORACLE7 был оценен Правительственным национальным

центром компьютерной безопасности (NCSC) как совместимый с критериями защиты Оранжевой книги; сервер ORACLE7 и Trusted ORACLE7 отвечают соответственно как уровням C2 и B1 Оранжевой книги, так и сравнимым с ними европейским критериям защиты ITSEC.

- Для защиты от несанкционированного доступа к базе данных ORACLE предоставляет защищенные от сбоев средства безопасности, лимитирующие и отслеживающие доступ к данным. Эти средства позволяют легко управлять даже наиболее сложными схемами доступа.

- ORACLE автоматически поддерживает целостность данных, соблюдая "организационные правила", которые диктуют стандарты приемлемости данных. Как следствие, устраняются затраты на кодирование и сопровождение проверок в многочисленных приложениях базы данных.

- Чтобы извлечь максимум преимуществ из данной компьютерной системы или сети, ORACLE позволяет разделять работу между сервером базы данных и прикладными программами клиентов. Вся тяжесть управления совместно используемыми данными может быть сосредоточена в компьютере, выполняющем СУБД, в то время как рабочие станции, на которых работают приложения, могут сконцентрироваться на интерпретации и отображении данных.

- В компьютерных окружениях, соединенных сетями, ORACLE комбинирует данные, физически находящиеся на разных компьютерах, в одну логическую базу данных, к которой имеют доступ все пользователи сети. Распределенные системы обладают такой же степенью прозрачности для пользователей и согласованности данных, что и нераспределенные системы, предоставляя в то же время преимущества управления локальной базой данных.

- Программное обеспечение ORACLE переносимо между различными операционными системами и одинаково во всех системах. Приложения, разрабатываемые для ORACLE, могут быть перенесены в любую операционную систему с минимумом модификаций или вообще без таковых.

- Программное обеспечение ORACLE совместимо с промышленными стандартами, включая большинство стандартных операционных систем. Приложения, разрабатываемые для ORACLE, могут использоваться в любой операционной системе с минимумом модификаций или вообще без таковых. Программное обеспечение ORACLE позволяет различным типам компьютеров и операционных систем совместно использовать информацию посредством сетей.

### **Физическая и логическая структура ORACLE**

База данных ORACLE имеет как физическую, так и логическую структуру. За счет разделения физической и логической структуры базы данных достигается возможность управления физической структурой данных, не затрагивая доступа к логическим структурам данных.

Физическая структура базы данных ORACLE определяется файлами операционной системы, из которых состоит база данных. Каждая база данных ORACLE составляется из файлов трех типов: одного или нескольких файлов данных, двух или более файлов журнала повторения

работы и одного или нескольких управляющих файлов.

Файлы базы данных предоставляют действительную физическую память для информации базы данных.

Логическая структура базы данных ORACLE определяется:

- одним или несколькими табличными пространствами;
- объектами схем базы данных (таблицами, обзорами, индексами, кластерами, последовательностями, хранимыми процедурами).

Логические структуры хранения, включая табличные пространства, сегменты и экстенды, определяют, как используется физическое пространство базы данных. Объекты схем и отношения между ними формируют реляционную структуру базы данных.

Блоки данных, экстенды и сегменты ORACLE предоставляет возможность тонкого контроля за использованием дисковой памяти через структуры логического хранения, включая блоки данных, экстенды и сегменты.

Механизмы ORACLE работают через использование структур памяти и процессов. Все структуры памяти располагаются в основной памяти (иногда называемой виртуальной памятью или памятью произвольного доступа) компьютеров, составляющих систему базы данных.

### **СУБД Oracle имеет собственный язык PL/SQL**

**PL/SQL** - это принадлежащее фирме Oracle процедурное языковое расширение языка SQL. PL/SQL сочетает легкость и гибкость SQL с процедурными возможностями языка структурного программирования, такими как IF...THEN, WHILE и LOOP.

При написании приложения базы данных разработчик должен рассмотреть преимущества использования хранимых подпрограмм PL/SQL:

- Поскольку код PL/SQL может храниться централизованно в базе данных, сетевой трафик между приложениями и базой данных сокращается, что увеличивает производительность как приложений, так и системы.

- Благодаря хранимому коду PL/SQL можно контролировать доступ к данным. При этом методе пользователи PL/SQL могут обращаться к данным лишь так, как это предусмотрено разработчиком приложения (если не предоставлен иной маршрут доступа). - Блоки PL/SQL могут пересылаться приложениями к базе данных, что позволяет выполнять комплексные операции без избыточной нагрузки на сеть.

Даже если PL/SQL не хранится в базе данных, приложения могут пересылать к базе данных не отдельные предложения SQL, а целые блоки, опять-таки сокращая сетевой трафик.



Следующие секции описывают различные программные единицы, которые могут быть определены и централизованно сохранены в базе данных.

**Процедуры и функции** представляют собой совокупности предложений SQL и PL/SQL, сгруппированных в единицу для решения специфической проблемы или выполнения множества взаимосвязанных задач. Процедура создается и сохраняется в базе данных в откомпилированной форме, и может выполняться (вызываться) любым пользователем или приложением. Процедуры и функции похожи друг на друга, с той разницей, что функция всегда возвращает вызывающей программе единственное значение, тогда как процедура не возвращает значения.

**Пакеты** дают метод инкапсулирования и хранения взаимосвязанных процедур, функций, переменных и других конструкторов пакета как единицы в базе данных. Предоставляя администратору базы данных или разработчику приложений организационные преимущества, пакеты в то же время расширяют функциональные возможности (например, глобальные переменные пакета могут объявляться и использоваться любой процедурой в пакете) и увеличивают производительность базы данных (так, все объекты пакета синтаксически разбираются, компилируются и загружаются в память один раз).

**Триггеры базы данных** ORACLE позволяет вам писать процедуры, которые выполняются автоматически в результате обновления, вставки или удаления из таблицы. Такие процедуры называются триггерами базы данных. Триггеры базы данных могут использоваться самыми разнообразными способами для информационного управления вашей базой данных. Например, их можно использовать для автоматизации генерации данных, аудита модификаций данных, введения в действие комплексных ограничений целостности или организации сложных процедур обеспечения защиты.

Многопользовательские системы баз данных, такие как ORACLE, включают средства защиты, которые контролируют обращения к базе данных и использование данных. Например, механизмы защиты выполняют следующее:

- предотвращают несанкционированный доступ к базе данных
- предотвращают несанкционированный доступ к объектам схем
- контролируют использование дисков
- контролируют использование системных ресурсов (таких как время процессора)
- осуществляют аудит действий пользователя

Защита базы данных может быть классифицирована по двум различным категориям: защита системы и защита данных. ЗАЩИТА СИСТЕМЫ включает механизмы, контролирующие доступ к базе данных и ее использование на уровне системы.

Например, защита системы включает:

- действительные комбинации имен пользователей и паролей;
- уполномочен ли пользователь присоединяться к базе данных ;
- объем дисковой памяти, доступный объектам пользователя ;
- лимиты ресурсов для пользователя ;
- активность или неактивность аудита базы данных ;
- какие системные операции разрешено выполнять пользователю .

Защита данных включает механизмы, контролирующие доступ к базе данных и ее использование на уровне объектов.

Например, защита данных включает:

- какие пользователи имеют доступ к конкретному объекту схемы и какие типы действий разрешены каждому пользователю на этом объекте (например, пользователь SCOTT может выдавать для таблицы EMP предложения SELECT и INSERT, но не предложения DELETE) ;
- какие действия подлежат аудиторскому отслеживанию для каждого объекта схемы.

ORACLE управляет защитой базы данных, используя несколько различных средств, среди которых:

- пользователи базы данных и схемы;
- привилегии;
- роли;
- назначения пространства и квоты;
- лимиты на ресурсы;
- аудитинг.

## Выводы по теме "Информационные системы"

Корпоративная информационная система является основой информационного пространства компании, гибким инструментом управления бизнесом в сложных, постоянно меняющихся условиях.

Информационная система корпоративного типа - систем, которая обрабатывает значительные объемы информации и поддерживает большие группы пользователей, охватывая территориально разбросанные предприятия и организации.

Построение корпоративных систем требует, прежде всего, осмысления своего бизнеса, понимания путей его развития и правильной организации управления информацией.

Сравнительный анализ СУБД, которые могут быть использованы для создания информационных систем корпоративного типа, показал, что наряду с такими популярными СУБД, как MySQL, Microsoft SQL Server 7.0 и т.п., СУБД Oracle является наиболее популярной и достаточно мощной СУБД для создания систем такого типа.

Сервер ORACLE обеспечивает эффективные и действенные решения для основных средств баз данных:

- поддерживает самые большие базы данных, потенциального размера до сотен гигабайт;
- поддерживает большое число пользователей, одновременно выполняющих разнообразные приложения;
- сохраняет высокую степень суммарной производительности системы;
- выборочно управляет доступностью данных, как на уровне базы данных, так и на более низких уровнях;
- удовлетворяет промышленно принятым стандартам по языку доступа к данным, операционным системам, интерфейсам с пользователем и сетевым протоколам;
- предоставляет защищенные от сбоев средства безопасности, лимитирующие и отслеживающие доступ к данным;
- автоматически поддерживает целостность данных;
- позволяет разделять работу между сервером базы данных и прикладными программами клиентов;
- программное обеспечение ORACLE переносимо между различными операционными системами и одинаково во всех системах; оно совместимо с промышленными стандартами, включая большинство стандартных операционных систем и т.д.

## Перечень ссылок

### Источники, использованные в материалах

Д.Кренке Теория и практика построения баз данных. СПб. Питер. 2003. -800с.

### Дополнительная литература

Лаура Томсон, Люк Веллинг. Разработка приложений на PHP и MySQL: пер. с англ. СПб. ООО «ДиаСофтЮП». 2003. -672с.

Мамаев Е., Вишневский А. Microsoft SQL Server 7 для профессионалов. СПб. Питер. 2000. -896с.