



## Лекция 2 Базы и банки данных. История развития ВТ и СУБД.



# ВВЕДЕНИЕ В БД

2



Составитель: доц. Космачева И.М.

# ВВЕДЕНИЕ В БД

3

*Табличная (матричная) структура данных*

Ф.И.О. пациента	Диагноз заболевания	Температура тела, °С	Рост пациента, см	Масса тела пациента, кг
Сидоров П.Е	Холецистит	37,7	171	87,3
Куликов И.С.	Панкреатит	38,4	168	66,7
Семенов В.П.	Пневмония	39,1	180	96,6

Составитель: доц.Космачева И.М.

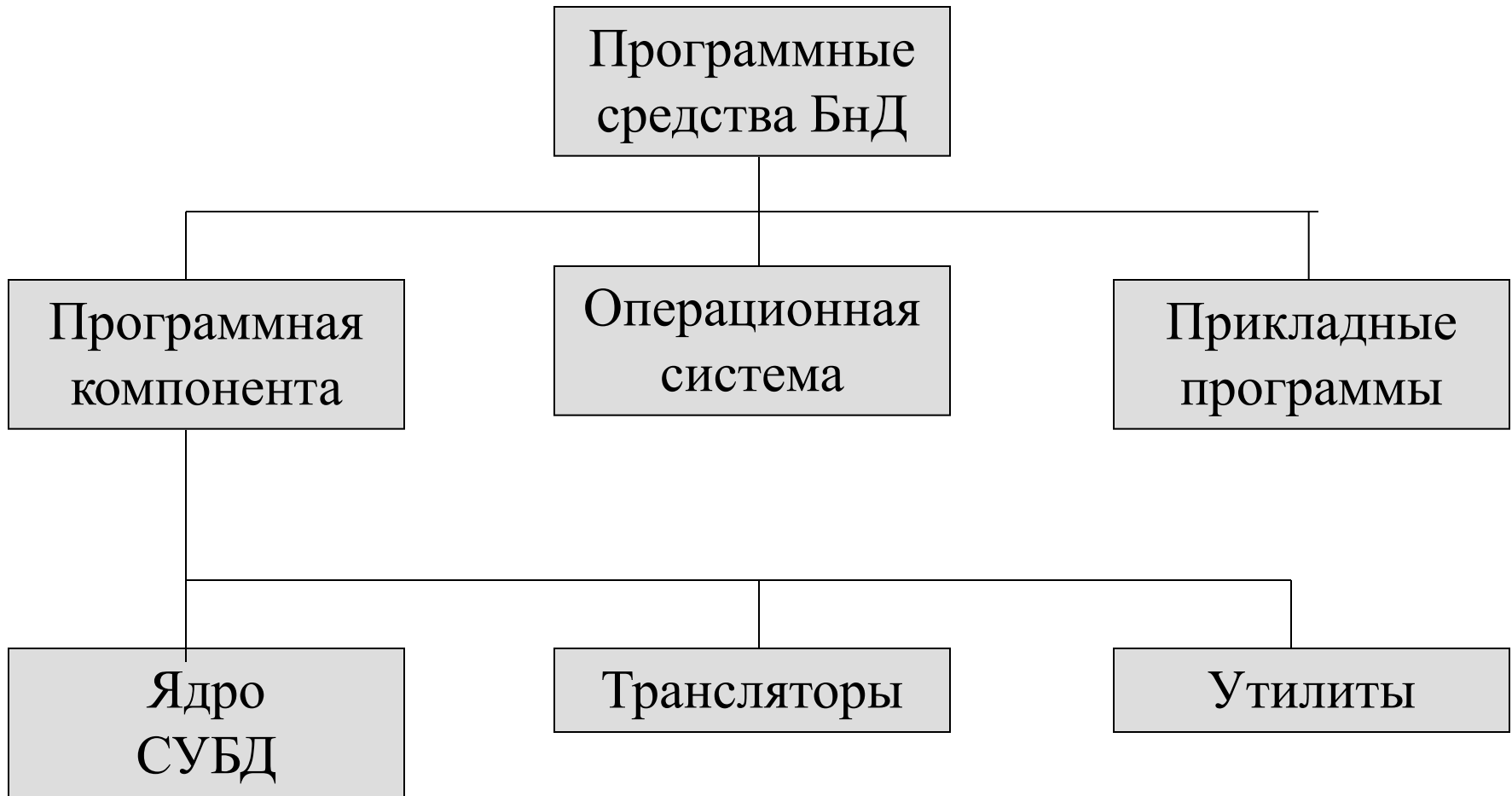
# 1. БАНК ДАННЫХ

**Банк данных** - это система специальным образом организованных баз данных, программных, технических, языковых, организационно-методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования данных.

Современной формой информационных систем являются **банки данных**, имеющие в своем составе:

- вычислительную систему;
- систему управления базами данных (СУБД);
- одну или несколько баз данных (БД);
- набор прикладных программ (приложений БД).

# ПРОГРАММНЫЕ СРЕДСТВА БНД



# ЯЗЫКОВЫЕ

## СРЕДСТВА БНД

- Обеспечивают интерфейс пользователей разных категорий с банком данных.
- В настоящее время используются языковые средства - табличный язык запросов QBE, структурированный язык запросов SQL.
- **Structured Query Language (SQL)**— это непроцедурный язык, используемый для формулировки запросов к данным в большинстве современных СУБД и являющийся индустриальным стандартом (ANSI, ISO).
- **Непроцедурность** языка означает, что на нем можно указать, что нужно сделать с базой данных, но нельзя описать алгоритм этого процесса. Существует много диалектов. Основу языка SQL составляют **операторы**.

# ОПЕРАТОРЫ ОПРЕДЕЛЕНИЯ ДАННЫХ (DATA DEFINITION LANGUAGE DDL):

7

- **CREATE** создает объект БД;
- **ALTER** изменяет объект;
- **DROP** удаляет объект;

# ОПЕРАТОРЫ МАНИПУЛЯЦИИ ДАННЫМИ (DATA MANIPULATION LANGUAGE DML):

8

- **SELECT** считывает данные, удовлетворяющие заданным условиям;
- **INSERT** добавляет новые данные;
- **UPDATE** изменяет существующие данные;
- **DELETE** удаляет данные;



# ОПЕРАТОРЫ ОПРЕДЕЛЕНИЯ ДОСТУПА К ДАННЫМ (DATA CONTROL LANGUAGE DCL):

9

- ❑ **GRANT** предоставляет пользователю (группе) разрешения на определенные операции с объектом;
- ❑ **REVOKE** отзывает ранее выданные разрешения;
- ❑ **SET TRANSACTION** начинает транзакцию и определяет ее поведение.

# ОПЕРАТОРЫ УПРАВЛЕНИЯ ТРАНЗАКЦИЯМИ (TRANSACTION CONTROL LANGUAGE, TCL):

10

- **COMMIT** применяет транзакцию;
- **ROLLBACK** откатывает все изменения, сделанные в контексте текущей транзакции;
- **SAVEPOINT** делит транзакцию на более мелкие участки, применяется для отметки логических точек разрыва в границах транзакции;
- **SET TRANSACTION** начинает транзакцию и определяет ее поведение.

# ОГРАНИЧЕНИЯ ЦЕЛОСТНОСТИ ДАННЫХ

11

База данных Oracle поддерживает следующие виды ограничений:

- **1) PRIMARY KEY**
- **2) UNIQUE**
- **3) FOREIGN KEY**
- **4) CHECK,**
- **5) NOT NULL**

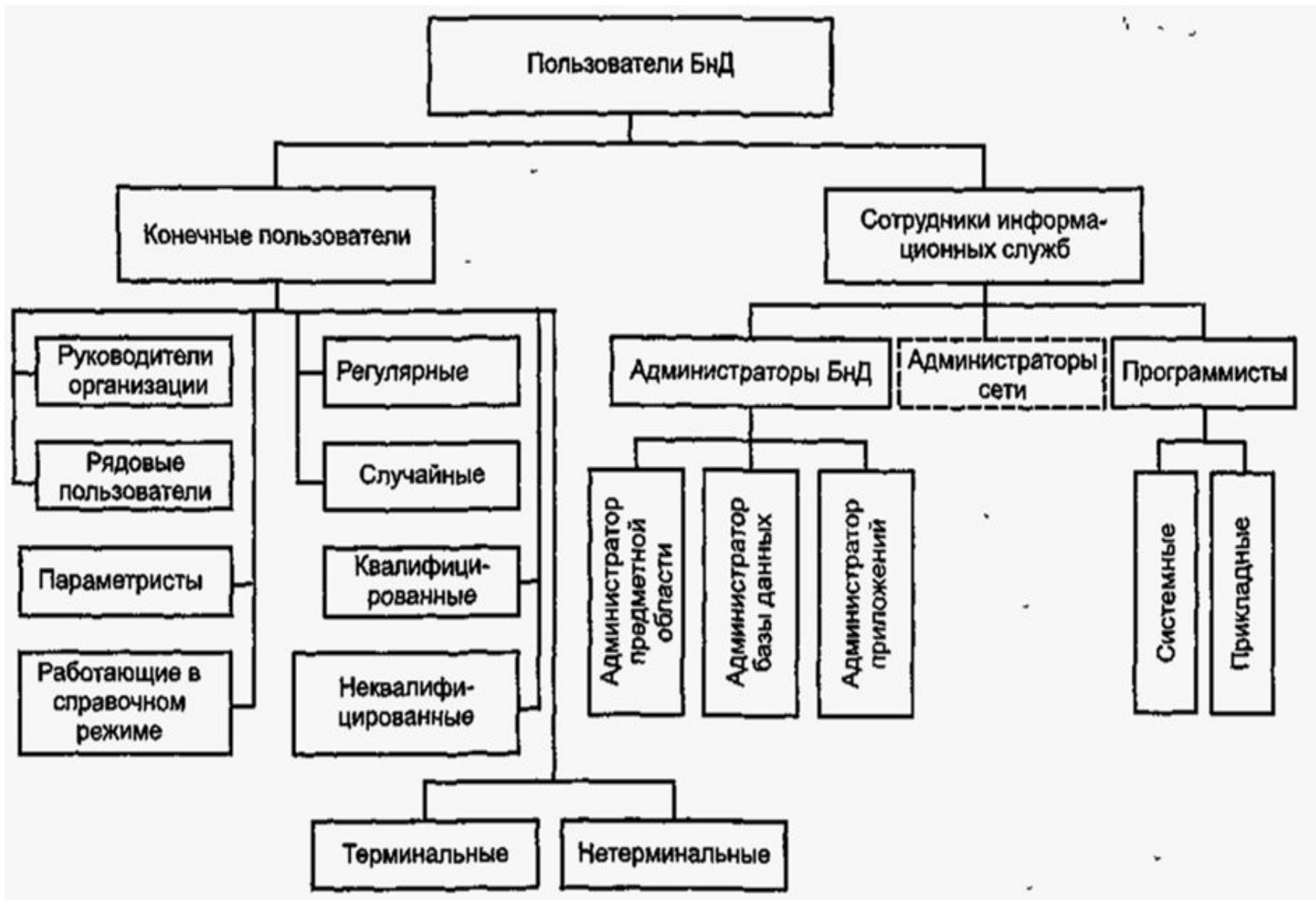
# ЯЗЫК SQL

12

- В начале 1970 г. в одной из исследовательских лабораторий компании IBM была разработана экспериментальная реляционная СУБД System R, для которой затем был создан специальный язык.
- В 1983 году Международная организация по стандартизации (ISO) и Американский национальный институт стандартов (ANSI) приступили к разработке стандарта языка SQL.

# ТЕХНИЧЕСКИЕ СРЕДСТВА БНД

- ❑ Персональные компьютеры
- ❑ Периферийные средства
- ❑ Средства сети (АПД)
- ❑ Специализированные ТС (машины баз данных)



# РАЗВИТИЕ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

15

- Развитие вычислительной техники происходило **в двух основных направлениях.**
- **Первое** направление - применение ВТ для выполнения численных расчетов, которые слишком долго или вообще невозможно производить вручную.
- Это привело к появлению высокоуровневых языков программирования.
- Разработанные подходы к программированию стали применяться не только для математических расчётов, но и для обработки менее формализованных данных: текстовых, графических, затем мультимедийных

# РАЗВИТИЕ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

16

- **Второе** направление - это использование средств ВТ в автоматических или автоматизированных информационных системах.
- Более важным, чем обработка данных, становится **второе направление** развития ВТ – оптимизация, защита хранения данных.
- Эти задачи решают **СУБД**.



# СУБД

17

- **Система управления базами данных (СУБД)** — совокупность программных, технических и языковых средств общего или специального назначения, позволяющая создавать базы данных и манипулировать сведениями из них, обеспечивать их безопасность.
- **Функции СУБД:**
  - управление данными во внешней памяти;
  - управление буферами оперативной памяти;
  - управление транзакциями;
  - журнализация и восстановление БД после сбоев;
  - поддержание языков БД.

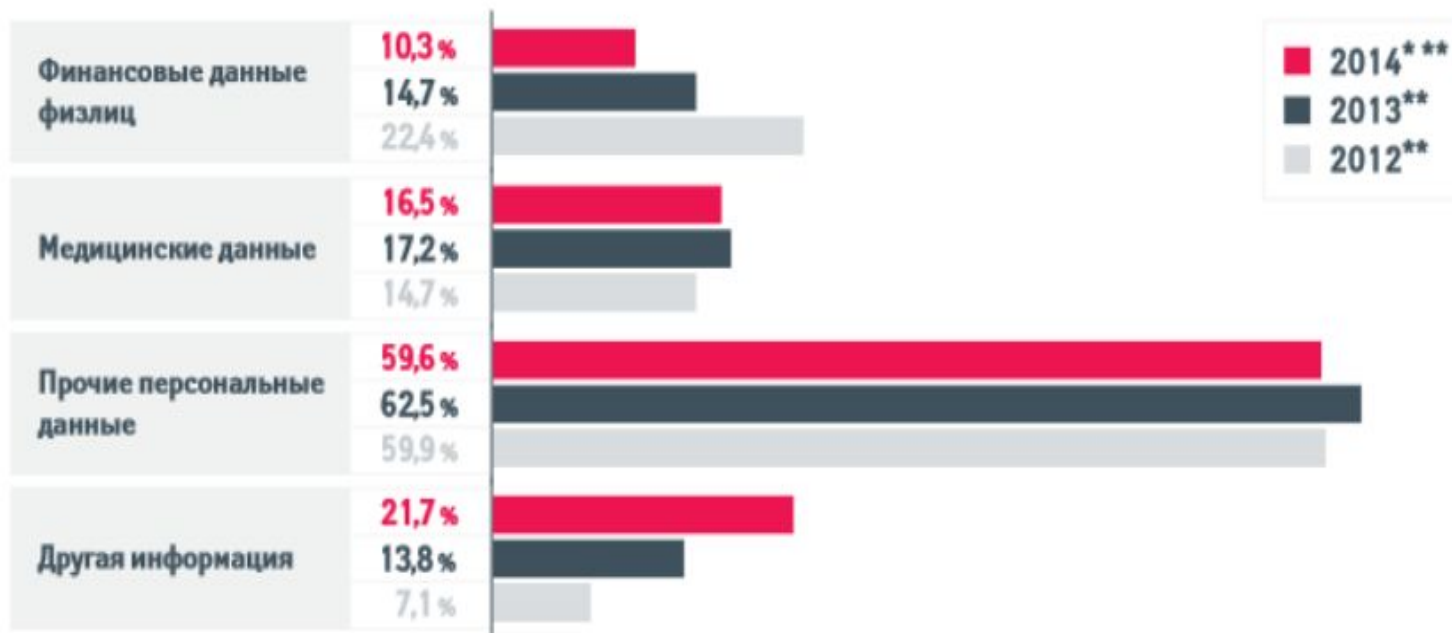
# СУБД

18

- СУБД обеспечивает физическую и логическую независимость прикладной программы от данных.
- В современной СУБД можно выделить внутреннюю часть - **ядро СУБД** (часто его называют Data Base Engine), **компилятор языка БД** (программа, преобразующая текст, написанный на алгоритмическом языке, в программу, состоящую из машинных команд), **подсистему поддержки времени выполнения, набор утилит**.
- Ядро СУБД обладает собственным интерфейсом, не доступным пользователям напрямую и используемым в программах, производимых компилятором SQL (или в подсистеме поддержки выполнения таких программ) и утилитах БД.
- Ядро СУБД является основной резидентной частью СУБД.
- **Ядро СУБД** отвечает за управление данными во внешней памяти, управление буферами оперативной памяти, управление транзакциями и журнализацию.

# СТАТИСТИКА ОТНОСИТЕЛЬНО БД

19



## Какие данные утекают

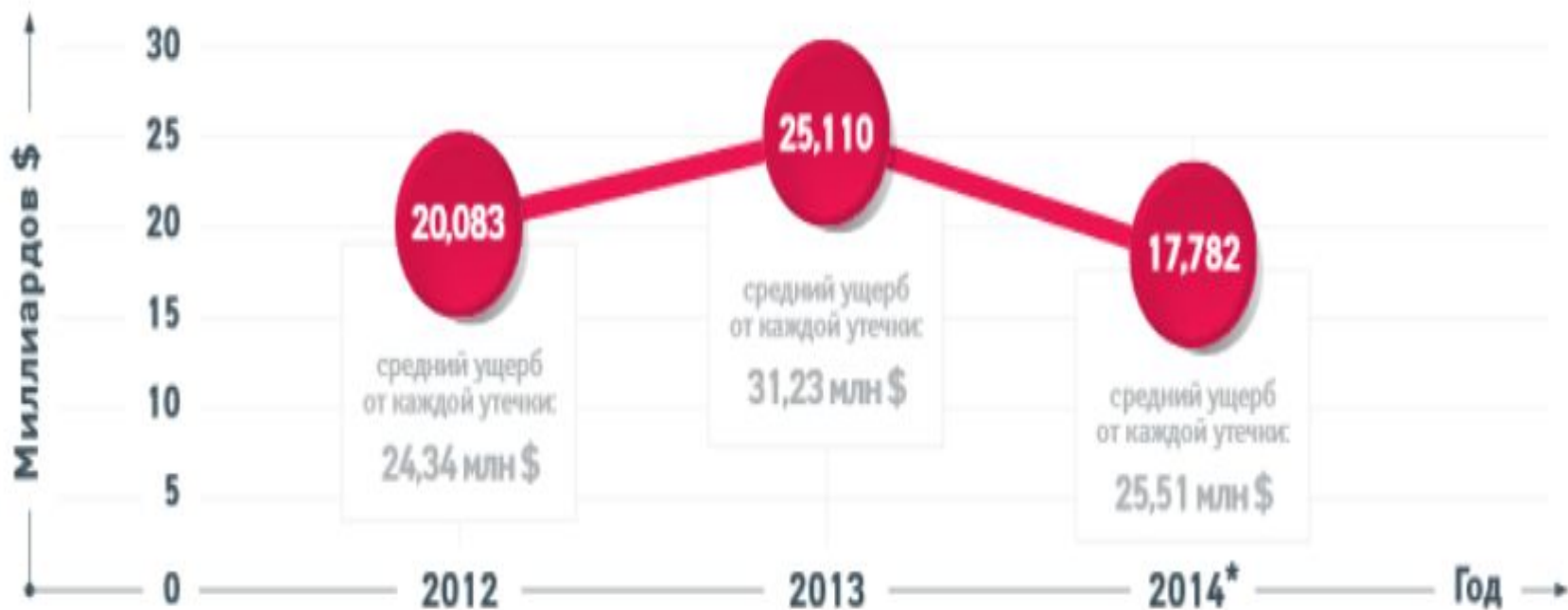
\* – за 11 месяцев 2014 года

\*\* – сумма долей утечек превышает 100%, поскольку в некоторых случаях информация классифицировалась по нескольким категориям одновременно.

Составитель: доц.Космачева И.М.

# СТАТИСТИКА ОТНОСИТЕЛЬНО БД

20



Убытки от утечек информации

\* – за 11 месяцев 2014 года

Составитель: доц.Космачева И.М.

# КРИТЕРИИ, ПО КОТОРЫМ ВЫБИРАЮТ СУБД

21

- производительность,
- безопасность,
- масштабируемость,
- обновляемость,
- уровень техподдержки,
- работа с очень большими данными,
- и цена владения.

# ПОПУЛЯРНОСТЬ СУБД

**DB-Engines** учитывает факторы:

1. Количество упоминаний о продукте в Сети, оцениваемое по результатам поисковых запросов (Google и Bing).
2. Интенсивность технических дискуссий в Сети. Измерялось количество вопросов и число участников обсуждений на сайтах.
3. Количество предлагаемых вакансий с упоминанием определенной СУБД.
4. Число профилей участников профессиональной социальной сети LinkedIn, в которых упоминается продукт, засвеченность в социальных сетях (количество твитов в Twitter с упоминанием конкретной СУБД).

# ПОПУЛЯРНОСТЬ СУБД

23

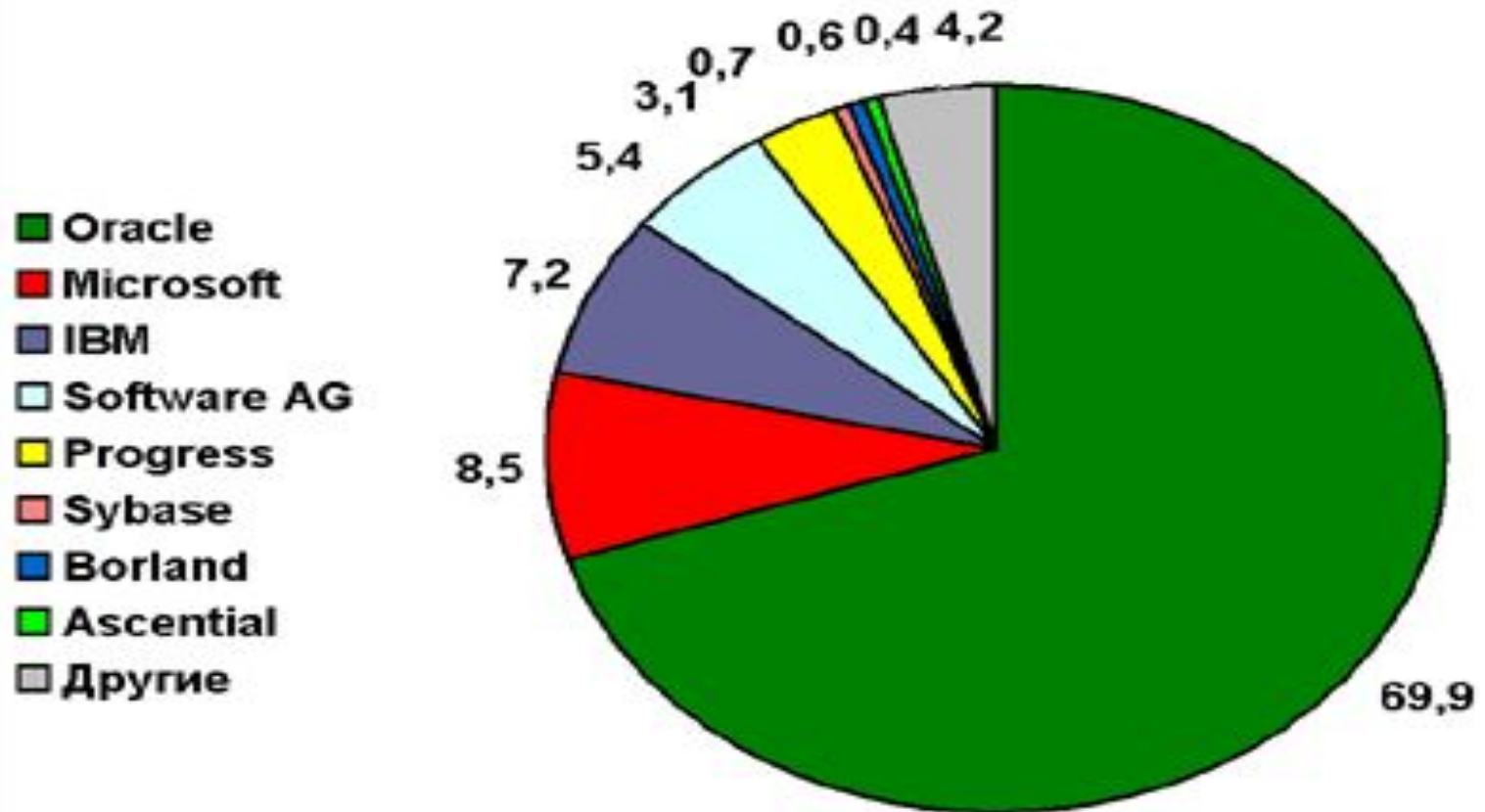
223 systems in ranking, August 2014

Rank	Last Month	DBMS	Database Model	Score	Changes
1.	1.	Oracle	Relational DBMS	1470.86	-14.26
2.	2.	MySQL	Relational DBMS	1281.22	-14.56
3.	3.	Microsoft SQL Server	Relational DBMS	1242.50	-4.10
4.	4.	PostgreSQL	Relational DBMS	249.85	+10.40
5.	5.	MongoDB	Document store	237.36	-1.42
6.	6.	DB2	Relational DBMS	206.42	+4.41
7.	7.	Microsoft Access	Relational DBMS	139.62	-5.00
8.	8.	SQLite	Relational DBMS	88.87	-2.29
9.	9.	Sybase ASE	Relational DBMS	86.17	+2.45
10.	10.	Cassandra	Wide column store	81.90	+0.32

Составитель: доц.Космачева И.М.

# ПОПУЛЯРНОСТЬ СУБД

24



Доли рынка различных разработчиков СУБД в России

Составитель: доц.Космачева И.М.



# СТАТИСТИКА

25

- По данным сайта **DB-Engines**, приводящего рейтинги различных СУБД, нынешнее соотношение популярности **коммерческих и открытых** СУБД составляет примерно **50 на 50**. Более 80% рынка СУБД в течение долгих лет контролируется тремя гигантами – **IBM, Oracle и Microsoft**.
- В последнее время в России появляются такие предложения, например, корейская **TIBERO**, очень похожая на **Oracle**. СУБД **Линтер** является единственной существующей в настоящее время коммерческой российской СУБД. Она разработана и развивается компанией Релэкс, г. Воронеж.
- СУБД **MySQL** бесплатная, стандарт де-факто у российских хост-провайдеров. В открытой СУБД **PostgreSQL** разрабатывается возможность обработки **неструктурированных данных** (текстовые и отсканированные), книги, презентации, картинки, рентгеновские снимки, отпечатки пальцев, музыкальные клипы, диктофонные записи, фотографии, электронные таблицы, фильмы, результаты наблюдений, новости, почта, протоколы чатов, счета и накладные, компьютерные распечатки и т д и т п.)

# ИСТОРИЯ РАЗВИТИЯ СУБД

26

- **1968** году была введена в эксплуатацию первая промышленная СУБД система IMS фирмы IBM.
- Развитие теории БД связано с американским математиком **Э. Ф. Коддом**.
- **Э. Ф. Кодд** - создатель **реляционной** модели данных. В **1981** году получил за создание реляционной модели и реляционной алгебры престижную премию **Тьюринга** Американской ассоциации по **вычислительной технике**.

# ИСТОРИЯ РАЗВИТИЯ СУБД

27

Этапы в развитии направления по обработке данных:

## **Базы данных на больших ЭВМ:**

- ✓ все СУБД базируются на мощных мультипрограммных операционных системах;
- ✓ в основном поддерживается работа с централизованной БД в режиме распределенного доступа;
- ✓ функции управления распределением ресурсов в основном осуществляются ОС;

# ИСТОРИЯ РАЗВИТИЯ СУБД

28

## Базы данных на больших ЭВМ:

- ✓ поддерживаются языки низкого уровня манипулирования данными;
- ✓ значительная роль отводится администрированию данных;
- ✓ проводятся работы по обоснованию и формализации реляционной модели данных, оптимизации запросов и управлению распределенным доступом к централизованной БД, введено понятие транзакции.
- результаты научных исследований открыто обсуждаются в печати, активно внедряются в коммерческие СУБД.
- Введено понятие транзакции. **Транзакция** – законченная единица работы, которая выполняется вся в случае успеха либо не выполняется совсем в случае сбоя, аварии.

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

# ИСТОРИЯ РАЗВИТИЯ СУБД

29

## Эпоха персональных компьютеров:

- ✓ компьютеры стали доступнее, СУБД рассчитаны в основном на монопольный доступ, скромные требования к аппаратному обеспечению настольных СУБД;
- ✓ вырождение функций администрирования БД и отсутствие инструментальных средств администрирования БД;
- ✓ отсутствие средств поддержки ссылочной и структурной целостности БД;
- ✓ создано много систем-однодневок, не отвечающим законам развития и взаимосвязи реальных объектов.
- кажущаяся простота и доступность персональных компьютеров и их программного обеспечения породила множество дилетантов.
- 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.

Составитель: доц.Космачева И.М.

# ИСТОРИЯ РАЗВИТИЯ СУБД

- После процесса "персонализации" начался обратный процесс — интеграция. Умножается количество локальных сетей, встает задача согласованности данных, хранящихся и обрабатываемых в разных местах, но логически друг с другом связанных, возникают задачи, связанные с параллельной обработкой транзакций.
- Появление распределенных баз данных, позволяющих организовать параллельную обработку информации и поддержку целостности БД.
- Большинство СУБД рассчитаны на многоплатформенную архитектуру

# ИСТОРИЯ РАЗВИТИЯ СУБД

31

## Распределенные базы данных:

- ✓ необходимость поддержки многопользовательской работы с БД и децентрализованного хранения данных потребовали развития средств администрирования БД;
- ✓ СУБД имеют средства подключения клиентских приложений, разработанных с использованием настольных СУБД, и средства экспорта данных из форматов настольных СУБД второго этапа развития;
- ✓ разработка ряда стандартов в рамках языков описания и манипулирования данными и технологий по обмену данными между различными СУБД (протокол **ODBC Open DataBase Connectivity** фирмы Microsoft (**открытая** система связи с базами данных)).

# ДАЛЬНЕЙШИЕ ПЕРСПЕКТИВЫ РАЗВИТИЯ

32

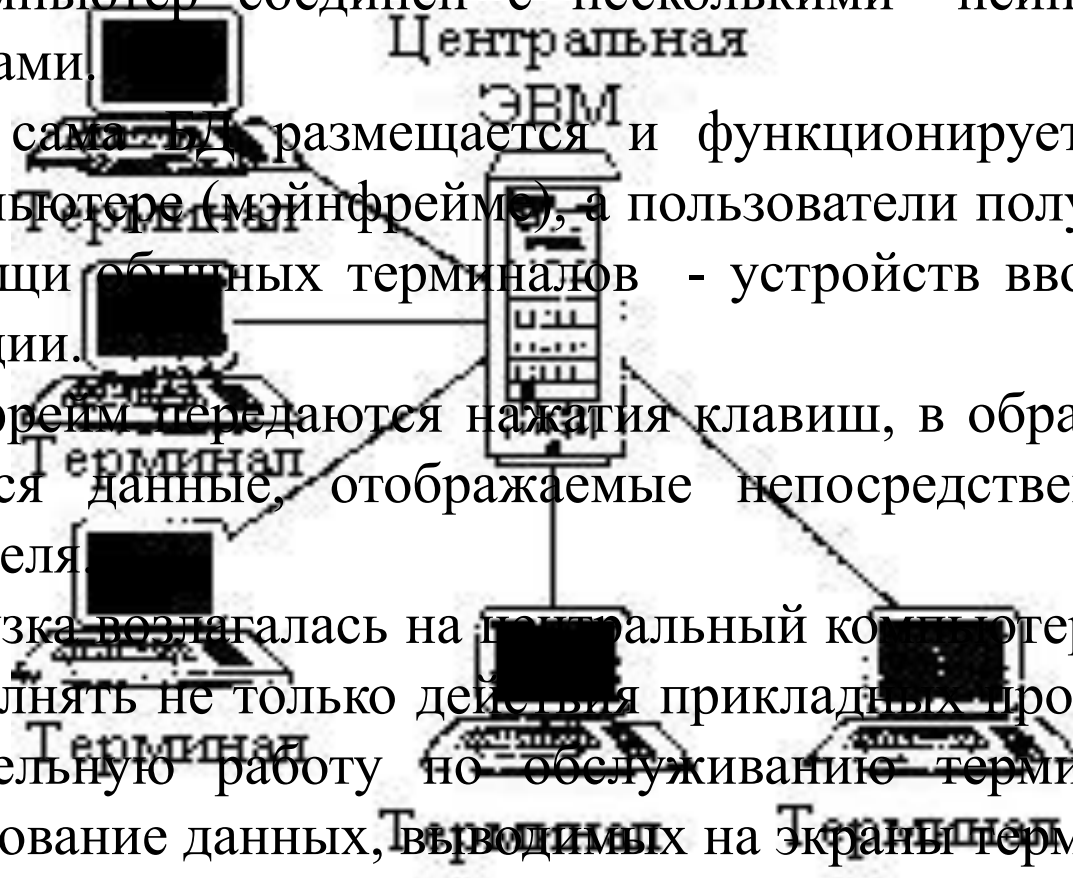
- Появился интернет.
- Отпадает необходимость использования специализированного клиентского программного обеспечения.
- Для работы с удаленной базой данных используется стандартный браузер Интернета, например Microsoft Internet Explorer .



# ТОПОЛОГИЯ АРХИТЕКТУРЫ ТЕЛЕОБРАБОТКИ

33

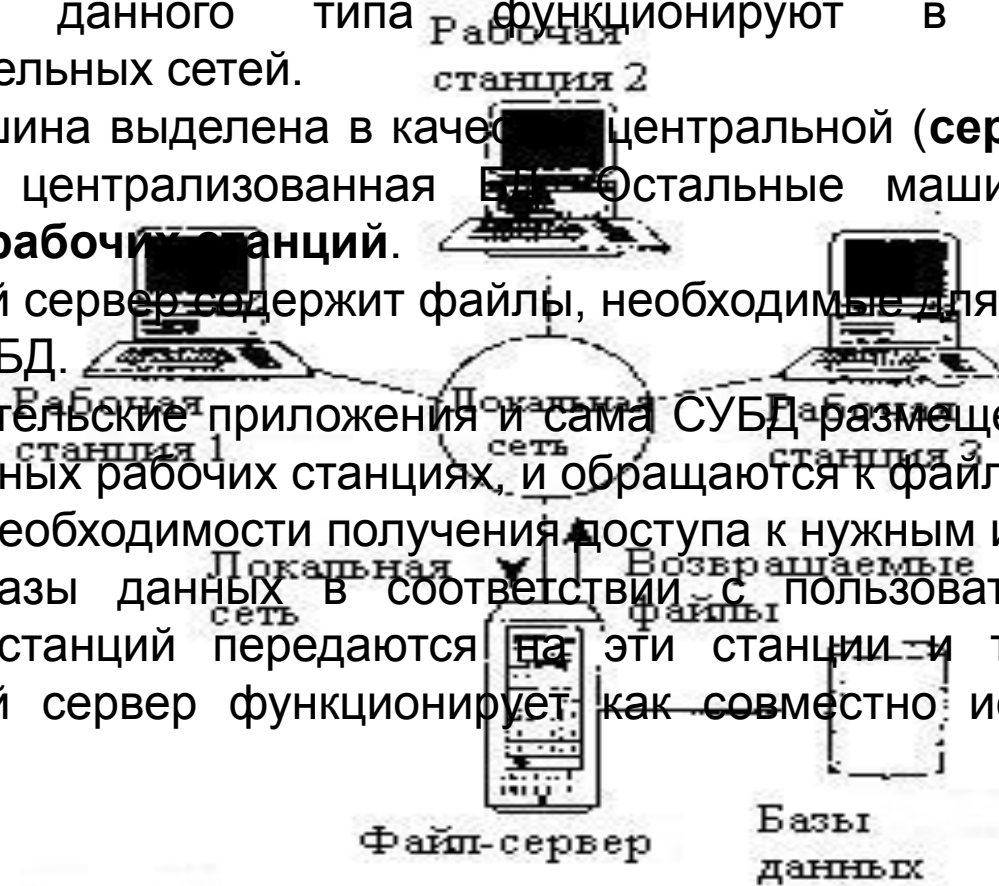
- Один компьютер соединен с несколькими "неинтеллектуальными" терминалами.
- СУБД и сама БД размещается и функционирует на центральном миникомпьютере (мэйнфрейме), а пользователи получают доступ к БД при помощи обычных терминалов - устройств ввода и отображения информации.
- На мэйнфрейме передаются нажатия клавиш, в обратном направлении передаются данные, отображаемые непосредственно на мониторе пользователя.
- Вся нагрузка возлагалась на центральный компьютер, который должен был выполнять не только действия прикладных программ и СУБД, но и значительную работу по обслуживанию терминалов (например, форматирование данных, выводимых на экраны терминалов).



# АРХИТЕКТУРА ФАЙЛОВОГО СЕРВЕРА

34

- Системы данного типа функционируют в рамках локальных вычислительных сетей.
- Одна машина выделена в качестве центральной (**сервер файлов**), на ней хранится централизованная база данных. Остальные машины сети выполняют функции **рабочих станций**.
- Файловый сервер содержит файлы, необходимые для работы приложений и самой СУБД.
- Пользовательские приложения и сама СУБД размещены и функционируют на отдельных рабочих станциях, и обращаются к файловому серверу только по мере необходимости получения доступа к нужным им файлам.
- Файлы базы данных в соответствии с пользовательскими запросами рабочих станций передаются на эти станции и там обрабатываются. Файловый сервер функционирует как совместно используемый жесткий диск.



# АРХИТЕКТУРА ФАЙЛОВОГО СЕРВЕРА

35

## Недостатки:

- Большой объем сетевого трафика.
- Производительность такой системы падает, если требуется интенсивный одновременный доступ к одним и тем же данным.
- На каждой рабочей станции должна находиться полная копия СУБД.
- Управление параллельностью, восстановлением и целостностью усложняется, поскольку доступ к одним и тем же файлам могут осуществлять сразу несколько экземпляров СУБД.

# АРХИТЕКТУРА “КЛИЕНТ/СЕРВЕР”

- Один из компьютеров, имеющий самый большой объем памяти и наиболее высокое быстродействие, становится приоритетным, называемым **сервером**.
- **Сервер** - узловая станция компьютерной сети, предназначенная в основном для хранения данных коллективного пользования и для обработки запросов в ней, поступающих от пользователей других узлов.
- **Клиент** - компьютер, обращающийся к совместно используемым ресурсам, которые предоставляются другим компьютером (сервером)..

# АРХИТЕКТУРА “КЛИЕНТ/СЕРВЕР”

37

- К клиентам не предъявляется столь жестких требований к памяти и быстродействию.
- На них располагаются словари и приложения, служащие своеобразными фильтрами для данных сервера.
- Предполагается существование клиентского процесса, требующего определенных ресурсов, а также серверного процесса, который эти ресурсы предоставляет. Не обязательно, чтобы они находились на одном и том же компьютере.
- Клиент посылает запрос, он обрабатывается сервером, и данные, полученные по запросу, передаются клиенту

# СХЕМА ПОСТРОЕНИЯ СИСТЕМ С АРХИТЕКТУРОЙ “КЛИЕНТ/СЕРВЕР”

38

## Клиент:

- Принимает и проверяет синтаксис введенного пользователем запроса;
- Клиент хранит в компьютере свои приложения, с помощью которых осуществляется запрос данных на сервере.
- Генерирует запрос к базе данных и передает его серверу;
- Отображает полученные данные пользователю.

## Сервер:

- Принимает и обрабатывает запросы к базе данных со стороны клиентов;
- Проверяет полномочия пользователей;
- Гарантирует соблюдение ограничений целостности;
- Выполняет запросы/обновления и возвращает результаты клиенту;
- Поддерживает системный каталог;
- Обеспечивает параллельный доступ к базе данных;
- Обеспечивает управление восстановлением

# ПРЕИМУЩЕСТВА

39

- Обеспечивается более широкий доступ к существующим базам данных.
- Повышается общая производительность системы.
- Стоимость аппаратного обеспечения снижается. Мощный компьютер с большим устройством хранения нужен только серверу.
- Сокращаются коммуникационные расходы. Приложения выполняют часть операций на клиентских компьютерах и посылают через сеть только запросы к базе данных, что позволяет существенно сократить объем пересылаемых по сети данных.

# ПРЕИМУЩЕСТВА

40

- Повышается уровень непротиворечивости данных. Сервер может самостоятельно управлять проверкой целостности данных, поскольку все ограничения определяются и проверяются только в одном месте.
- Данная архитектура может быть использована для организации средств работы с распределенными базами данных, т.е. с набором нескольких баз данных, логически связанных и распределенных в компьютерной сети.



# ТРЕХУРОВНЕВАЯ АРХИТЕКТУРА

41

- Если деловая логика взаимодействия с базой данных (логика, определяющая порядок работы предприятия: какие таблицы и в каком порядке заполнять, что делать при добавлении нового сотрудника и т.д.) изменяется, то приходится заново переписывать клиентские программы (водить новые формы, менять порядок их заполнения и т.д.).
- Если изменения происходят слишком часто, а количество рабочих мест велико, то постоянная переустановка программного обеспечения (которая, кстати, должна осуществляться достаточно быстро) становится серьезной проблемой.
- В таких случаях следует переходить к трехуровневой архитектуре: «тонкий клиент» - сервер приложений – сервер базы данных.

Составитель: доц.Космачева И.М.

# ТРЕХУРОВНЕВАЯ АРХИТЕКТУРА

42

При *трехуровневой архитектуре* в функции клиентской части («тонкий клиент») входит только интерактивное взаимодействие с пользователем, а вся деловая логика вынесена на сервер приложений, который собственно и обеспечивает формирование запросов к базе данных, передаваемых на выполнение серверу базы данных.



Составитель: доц.Космачева И.М.

# ИСТОРИЯ РАЗВИТИЯ СУБД

43

- Файловые системы были первой попыткой компьютеризировать ручные картотеки. Обычно таких папок очень много, они нумеруются и хранятся в одном или нескольких шкафах. В целях безопасности шкафы могут закрываться на замок или находиться в охраняемых помещениях.
- Если понадобится какая-то информация, потребуется просмотреть картотеку от начала до конца.
- Более продуманный подход предусматривает использование в такой системе некоторого алгоритма индексирования, позволяющего ускорить поиск нужных сведений.
- Ручные картотеки позволяют успешно справляться с поставленными задачами, если количество хранимых информационных объектов невелико. Они также вполне подходят для работы с большим количеством объектов, которые нужно только хранить и извлекать.

# ИСТОРИЯ РАЗВИТИЯ СУБД

44

- Однако они совершенно не подходят для тех случаев, когда нужно установить перекрестные связи или выполнить обработку сведений.
- **Файловые системы** - набор прикладных программ, которые выполняют для пользователей некоторые операции, например создание отчетов. Каждая программа хранит свои собственные данные и управляет ими.
- Все перечисленные выше **ограничения файловых систем** являются следствием двух факторов.
  1. Определение данных содержится внутри приложений, а не хранится отдельно и независимо от них.
  2. Помимо приложений не предусмотрено никаких других инструментов доступа к данным и их обработки.

Составитель: доц.Космачева И.М.

# ОГРАНИЧЕНИЯ, ПРИСУЩИЕ ФАЙЛОВЫМ СИСТЕМАМ:

45

- Разделение и изоляция данных
- Дублирование данных
- Зависимость от данных
- Несовместимость файлов
- Фиксированные запросы/быстрое увеличение количества приложений

# ПРЕИМУЩЕСТВА СУБД

46

- Контроль за избыточностью данных
- Непротиворечивость данных
- Больше полезной информации при том же объеме хранимых данных
- Совместное использование данных
- Поддержка целостности данных
- Повышенная безопасность
- Применение стандартов
- Повышение эффективности с ростом масштабов системы

# ПРЕИМУЩЕСТВА СУБД

47

- Возможность нахождения компромисса при противоречивых требованиях
- Повышение доступности данных и их готовности к работе
- Улучшение показателей производительности
- Упрощение сопровождения системы за счет независимости отданных
- Улучшенное управление параллельной работой
- Развитые службы резервного копирования и восстановления

Доставитель: доц.Космачева И.М.

# НЕДОСТАТКИ СУБД

48

- Сложность
- Размер
- Стоимость СУБД
- Дополнительные затраты на аппаратное обеспечение
- Затраты на преобразование
- Производительность
- Более серьезные последствия при выходе системы из строя



# ОСНОВНЫЕ КОМПОНЕНТЫ СИСТЕМЫ ЗАЩИТЫ БАЗ ДАННЫХ

49

- **Разграничение доступа**
- **Защита доступа**
- **Шифрование данных**
- **Аудит доступа к данным**

# ОСНОВНЫЕ КОМПОНЕНТЫ СИСТЕМЫ ЗАЩИТЫ БАЗ ДАННЫХ

50

- каждый пользователь, включая администратора должен иметь **доступ только к необходимой** ему согласно занимаемой должности информации.
- доступ к данным может получить пользователь, прошедший процедуру **идентификации и аутентификации**.
- **шифровать** необходимо как передаваемые в сети данные для защиты от перехвата, так и данные, записываемые на носитель, для защиты от кражи носителя и несанкционированного просмотра/изменения не-средствами системы управления БД (СУБД).
- действия с критичными данными должны **протоколироваться**. Доступ к протоколу не должны иметь пользователи, на которых он ведется. В случае приложений, использующих многозвенную архитектуру, приведенные функции защиты также имеют место, за исключением защиты данных на носителе - эта функция остается за БД. Всеми перечисленными функциями безопасности в той или иной мере оснащены СУБД.

# СИСТЕМНЫЕ ТРЕБОВАНИЯ ДЛЯ УСТАНОВКИ ORACLE 11g

:

51

- **1 GB RAM (512 Мб для ORACLE 10)**
- **1.5 GB своп-памяти (1 GB для ORACLE 10)**
- **400 MB дискового пространства во временной директории Temp**
- **от 1.5 GB до 3.5 GB для ПО Oracle**
- **1.2 GB для базы данных**
- **2.4 GB для области восстановления базы данных**
- **Операционная система – смотреть документацию**

## Oracle Database Enterprise Edition

- **950 \$ NUP**
- **47500 \$ CPU**

# ТИПЫ ПОЛЬЗОВАТЕЛЕЙ

52

- администраторы баз данных;
- администраторы по защите данных;
- разработчики приложений;
- администраторы приложений;
- администраторы сети;
- пользователи базы данных.

# ОБЯЗАННОСТИ АДМИНИСТРАТОРА БД

53

- **установка нового программного обеспечения, конфигурация программного и аппаратного обеспечения (вместе с системным администратором)**
- **обеспечение безопасности** : добавление и удаление пользователей, управление квотами, аудит и разрешение проблем безопасности
- **настройка производительности и мониторинг**
- **резервное копирование и восстановление системы - процедура планового обслуживания, тестирование и настройка.**
- **локализация неисправностей, восстановление системы после сбоя**

**СПАСИБО ЗА ВНИМАНИЕ!**