

Технологии базы данных

Введение

Литература

- Дейт К. Введение в системы баз данных. 7-е издание.— М.: Издательский дом «Вильямс», 2000 (и более поздние издания, с 2024г в 2х томах)
- Дейт К. SQL и реляционная теория. – М.: Символ-Плюс, 2010. (и более поздние издания)
- Моргунов Е.П. PostgreSQL. Основы языка SQL. — СПб.: БХВ-Петербург, 2018 (есть электронная версия)
- Новиков Б.А., Горшкова Е.А., Графеева Н.Г. Основы технологий баз данных. — М.: ДМК Пресс, 2020 (есть электронная версия)
- Моргунов Е. П. PostgreSQL. Профессиональный SQL — М.: ДМК Пресс, 2025(есть электронная версия)

Источники в интернете

- postgrespro.ru
- postgresql.org
- edu.mmcs.sfedu.ru
- ~~www.sql.ru~~
- Форум по разным темам resql.ru

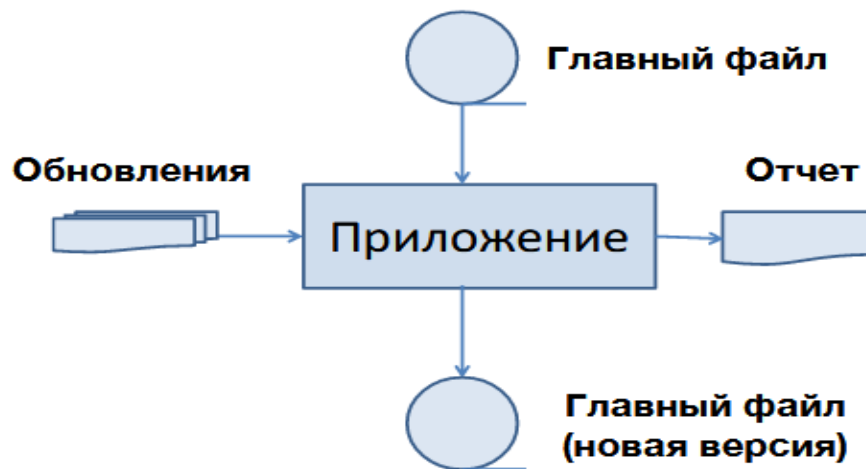
Информационные системы

- Информационная система – программный комплекс, функции которого состоят в:
 - поддержке надежного долговременного хранения информации,
 - выполнении специфических для данного приложения преобразований информации и/или вычислений,
 - предоставлении пользователям удобного и легко осваиваемого интерфейса.

Примеры ИС

- банковские системы
- системы резервирования билетов
- системы автоматизации управления
- системы учета, биллинговые системы
- системы поддержки принятия решений
- прочие

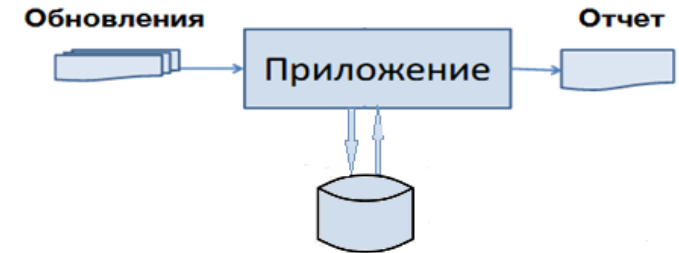
Файловые системы (последовательный доступ)



- Проблемы
 - Большое время отклика
 - Полная перезапись данных
 - Хранение последовательных копий для восстановления

Файловые системы (прямой доступ)

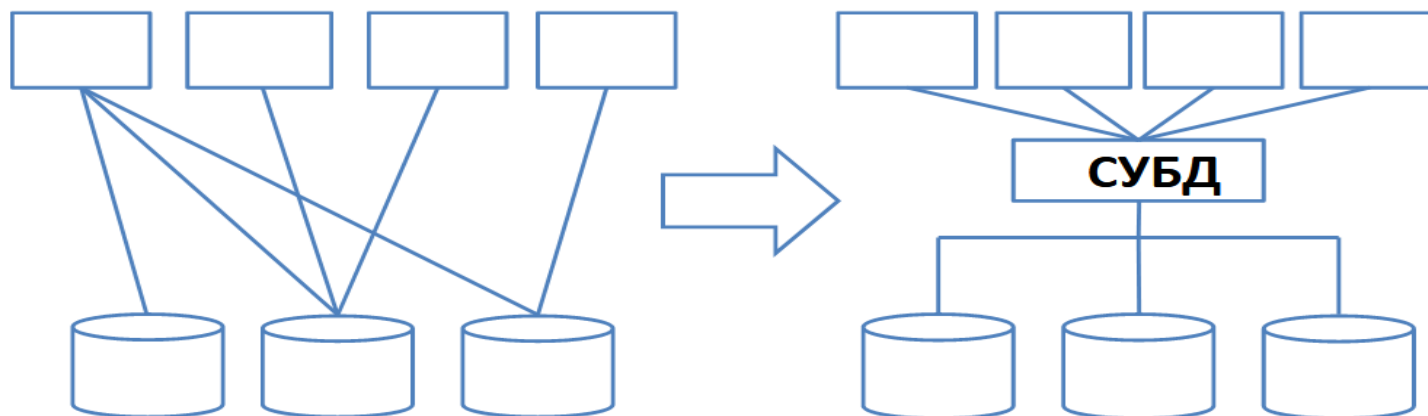
- Но проблемы остаются
 - О структуре информации знает только приложение
 - Нет средств согласования связанной по смыслу информации, возможно хранящейся в нескольких файлах
 - Сложность реализации алгоритмов обработки информации в терминах операций чтения/записи
 - Ограничение на параллельную обработку в многопользовательской среде
 - Нет средств корректного восстановления после сбоев



Что нужно ИС ?

- совместное хранение информации и метаданных, представляющих семантику информации
- наличие средств, обеспечивающих управление данными
- поддержание согласованности данных
- наличие языка, обеспечивающего легкий доступ к информации
- авторизация доступа к информации
- обеспечение эффективной одновременной работы многих пользователей
- возможность возврата после сбоя к некоторому предыдущему согласованному состоянию
- возможность журнализации работы пользователей

Централизация управления данными



Терминология

Комплекс программ, обеспечивающий потребности информационных систем, называется

Системой управления базами данных (СУБД)

В настоящее время используется термин «сервер баз данных»

База данных (БД) – это набор хранимых данных и их метаданных (описаний), используемых информационной системой/системами.

В настоящее время БД может хранить также алгоритмы обработки данных

Основные функции СУБД (CODASYL 1971 г.)

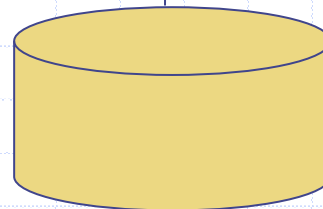
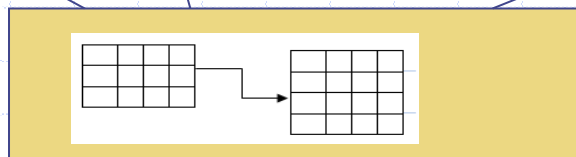
- Обеспечение физической независимости данных
- Поддержка концептуальной модели предметной области (средства для работы с метаданными)
- Обеспечение целостности информации
- Поддержка языка доступа к информации (например, SQL)
- Авторизация прав пользователей
- Журнализация изменений состояния БД
- Поддержка транзакций и обеспечение механизма фиксации и отката транзакций, управление асинхронным выполнением транзакций в многопользовательском режиме
- Обеспечение восстановления целостного состояния после сбоев

Модель языка описания данных ANSI/SPARC (стандарт 1975г.)

Внешний (прикладной)



Концептуальный



Внутренний

Концептуальный уровень

В центре архитектуры ANSI/SPARC лежит концептуальный уровень

Концептуальный уровень описывает данные и их взаимосвязи с наиболее общей точки зрения, — концепции архитекторов базы, используя реляционную или другую модель

Физический (внутренний) уровень

Внутренний уровень позволяет скрыть подробности физического хранения данных (носители, файлы, табличные пространства, ...) от концептуального уровня.

Отделение внутреннего уровня от концептуального обеспечивает так называемую **физическую независимость** данных.

Внешний уровень

На внешнем уровне описываются различные подмножества элементов концептуального уровня для представлений данных различным пользовательским программам

Каждое приложение получает в свое распоряжение часть представлений о данных, но полностью концептуальный уровень скрыт

Отделение внешнего уровня от концептуального обеспечивает **логическую независимость** данных

Типы СУБД по модели данных

- Системы инвертированных списков
- Иерархические
- Сетевые
- Реляционные
- Объектно-ориентированные и объектно-реляционные
- Многомерные
- Логические или экспертные
- Слабоструктурированные
- . . .

Типы СУБД по способам взаимодействия с приложением

- файл – серверные
 - Примеры: Microsoft Access, Paradox, dBase, FoxPro
- клиент – серверные и серверы приложений
 - Примеры: Oracle, PostgreSQL, MS SQL Server, Firebird, Interbase, IBM DB2, Informix, Sybase Adaptive Server Enterprise, MySQL, Caché
- встраиваемые
 - Примеры: OpenEdge, SQLite, BerkeleyDB, Firebird Embedded, Microsoft SQL Server Compact

PostgreSQL

- ◆ PostgreSQL — это бесплатная объектно-реляционная система управления базами данных с открытым исходным кодом





PostgreSQL история

- ◆ PostgreSQL началась в 1986 году как проект POSTGRES в Калифорнийском университете в Беркли под руководством профессора Майкла Стоунбрейкера
- ◆ 1986–1993 (POSTGRES): Разработка оригинальной системы с поддержкой объектно-реляционных концепций. Команда выпустила несколько версий (от первой в 1989 до версии 4.2 в 1993 году), после чего проект закрылся
- ◆ 1994 (Postgres95): Студенты Эндрю Ю и Джолли Чен добавили в кодовую базу поддержку языка SQL вместо старого языка PostQueL. Код сократили на 25% и перевели на ANSI C, что повысило скорость работы на 30–50%

PostgreSQL история



- ◆ 1996 (Рождение PostgreSQL): Проект переименовали в PostgreSQL, чтобы подчеркнуть поддержку SQL. Вышла версия 6.0, заложившая основу современного открытого глобального сообщества разработчиков
- ◆ 2005 (Версия 8.0): Появилась нативная поддержка Windows и встроенная репликация, сделавшая систему популярной в корпоративном секторе
- ◆ Настоящее время: Система регулярно обновляется международным сообществом (Global Development Group), поддерживая передовые стандарты SQL, JSON и масштабируемость



PostgreSQL. Главные особенности

- ◆ **Надежность:** Полная поддержка стандартов ACID (гарантирует сохранность данных при сбоях и транзакционную целостность данных). Это делает её стандартом для финансовых систем и критически важных приложений
- ◆ **Расширяемость:** Возможность создавать собственные функции, типы и индексы. Можно писать хранимые процедуры и функции не только на стандартном PL/pgSQL, но и на Python, JavaScript, C и Java
- ◆ **Объектно-реляционная модель:** Позволяет использовать наследование таблиц



PostgreSQL. Главные особенности

- ◆ Гибкость: Поддержка сложных типов данных, включая JSON, массивы и пользовательские структуры
- ◆ Масштабируемость: Подходит для проектов любого размера — от малых приложений до систем с высокой нагрузкой
- ◆ Многоверсионность (MVCC): Реализует параллельный доступ к данным — читающие процессы не блокируют пишущие, и наоборот, что сильно повышает скорость при высокой нагрузке



PostgreSQL. Области применения

- ◆ Веб-разработка и мобильные приложения
- ◆ Корпоративные системы учета (например, решения на базе 1С)
- ◆ Аналитика больших объемов данных и геоинформационные системы

PostgreSQL в России

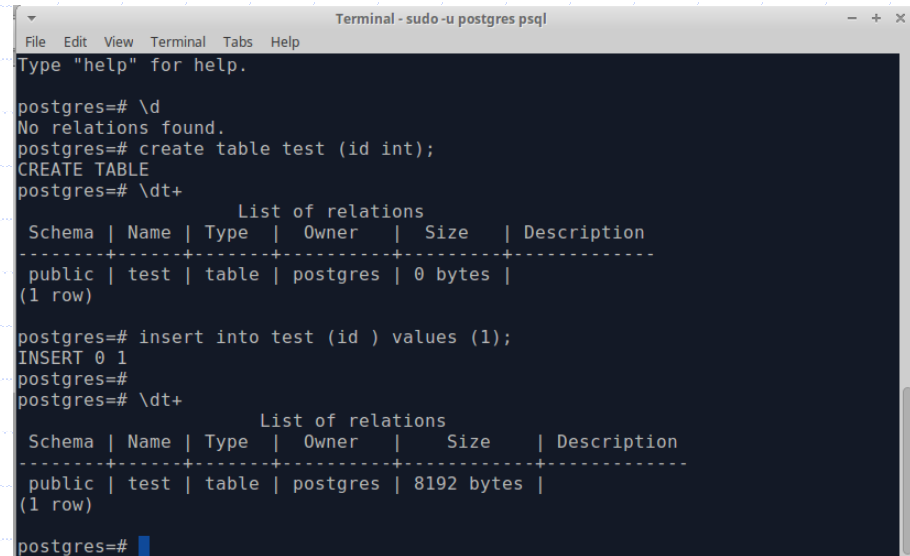


- ◆ 2015 - Основание компании, объединение энтузиастов PostgreSQL, выпуск первой российской коммерческой СУБД
- ◆ Postgres Professional — экосистема из более чем десятка продуктов для эффективной работы с данными
- ◆ СУБД Postgres Pro, обеспечивает надежную работу более чем 3 000 крупнейших компаний
- ◆ Занимает первое место на российском рынке СУБД России и продолжает развивать свободно распространяемую СУБД PostgreSQL



Инструменты для разработки

- ◆ psql: Стандартный интерактивный терминальный клиент, поставляемый вместе с PostgreSQL



```
Terminal - sudo -u postgres psql
File Edit View Terminal Tabs Help
Type "help" for help.

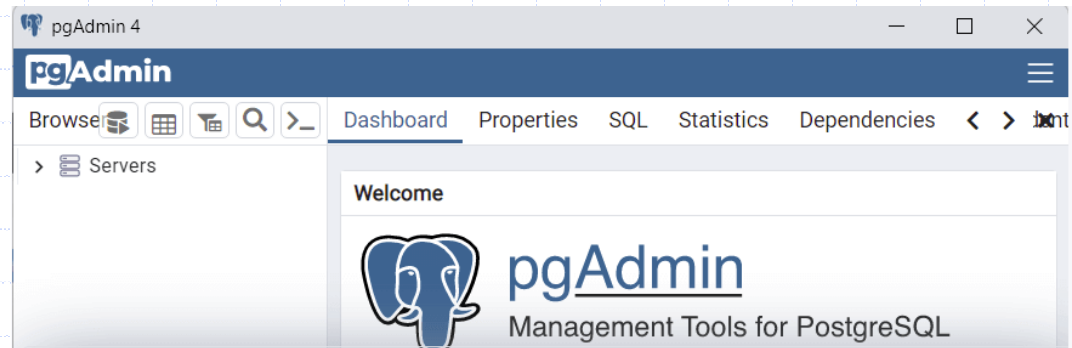
postgres=# \d
No relations found.
postgres=# create table test (id int);
CREATE TABLE
postgres=# \dt+
          List of relations
Schema | Name | Type | Owner  | Size  | Description
-----+-----+-----+-----+-----+-----
public | test | table | postgres | 0 bytes | 
(1 row)

postgres=# insert into test (id ) values (1);
INSERT 0 1
postgres=#
postgres=# \dt+
          List of relations
Schema | Name | Type | Owner  | Size  | Description
-----+-----+-----+-----+-----+-----
public | test | table | postgres | 8192 bytes | 
(1 row)

postgres=#
```

Инструменты для разработки

- ◆ pgAdmin: Официальный и бесплатный инструмент с открытым исходным кодом, созданный специально для PostgreSQL



Инструменты для разработки

- ◆ DBeaver поддерживает доступ к разным СУБД через драйверы JDBC

DBeaver

Универсальный менеджер баз данных

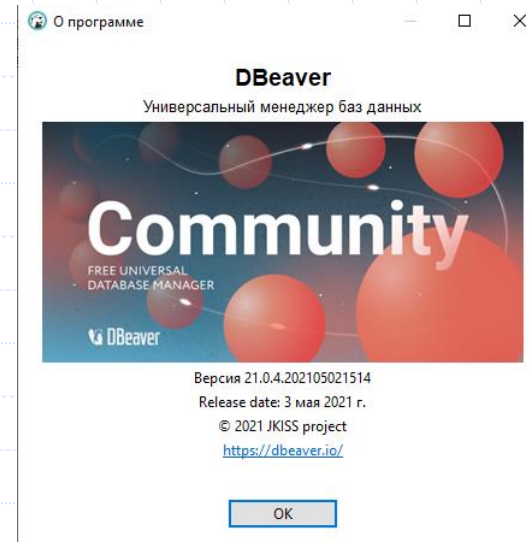


Версия 5.1.4

30 июля 2018 г.

(C) 2018 JKISS project

<https://dbeaver.io/>



Инструменты для разработки

- DataGrip поддерживает доступ к разным СУБД через драйверы JDBC

