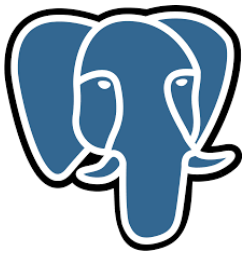
PostgreSQL для разработчиков приложений

Павел Лузанов, Postgres Professional, 17.06.2016

About PostgreSQL



Postgre SQL is a powerful, open source object - relational database system.

www.postgresql.org/about

Серверная часть приложения

Реляционная модель данных

Реляционная модель данных

Зачем?

Реляционные СУБД для реляционной модели

Нормализация данных

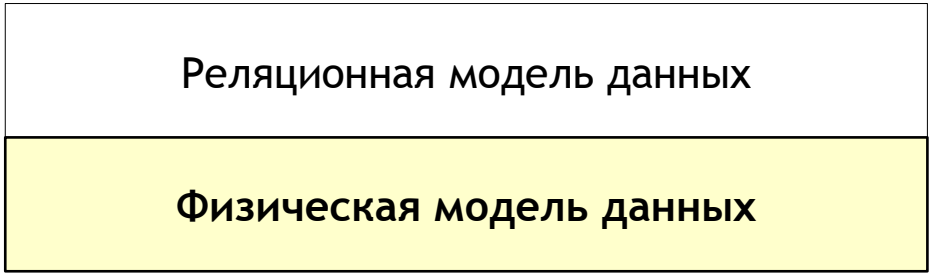
ER-диаграммы

Реляционная модель данных

PostgreSQL

Не зависит от СУБД

Серверная часть приложения



Физическая модель данных

Зачем?

Сама по себе ER-диаграмма не работает

Нормализованная модель не будет работать

Физическая модель данных

Зачем?

- Денормализация
  - добавление избыточности
  - автоматическая поддержка

Физическая модель данных

PostgreSQL

Отношения, атрибуты, связи →

- таблицы, столбцы, ограничения целостности

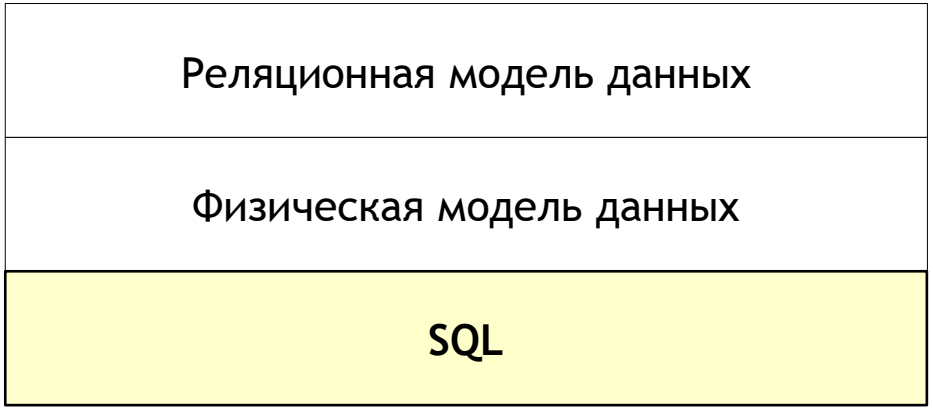
Доступ к данным

- роли, схемы, привилегии

Деномализация данных

- добавление столбцов (восходящая, нисходящая)
- материализованные представления
- индексы (разные типы, для разных задач)

Серверная часть приложения



SQL

Зачем?

Декларативный язык

- оперирует множествами, а не объектами

+ делает это очень эффективно!

SQL

PostgreSQL

Соответствие стандарту ANSI SQL

Богатый функционал

- аналитические функции (оконнные, групповые)
- Common Table Expressions (CTE)
- иерархические запросы
- LATERAL

SQL

PostgreSQL

EXPLAIN: план/факт запросов

Планировщик запросов

- методы доступа к таблицам
- способы соединения таблиц
- статистика

Серверная часть приложения

Реляционная модель данных	
Физическая модель данных	
SQL	**Хранимые процедуры**

Хранимые процедуры

Зачем?

- Производительность
  - меньше команд
  - меньше данных
  - меньше разборов команд
- API для доступа к данным

Хранимые процедуры

PostgreSQL

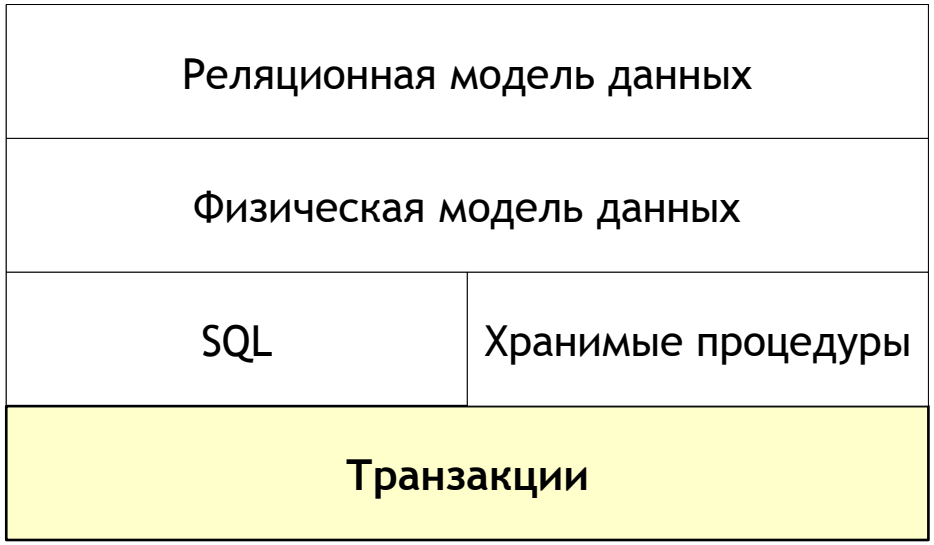
Языки на выбор

Только функции

~~commit;~~

Временные таблицы, глобальные переменные

Серверная часть приложения



Транзакции

Зачем?

ACI D

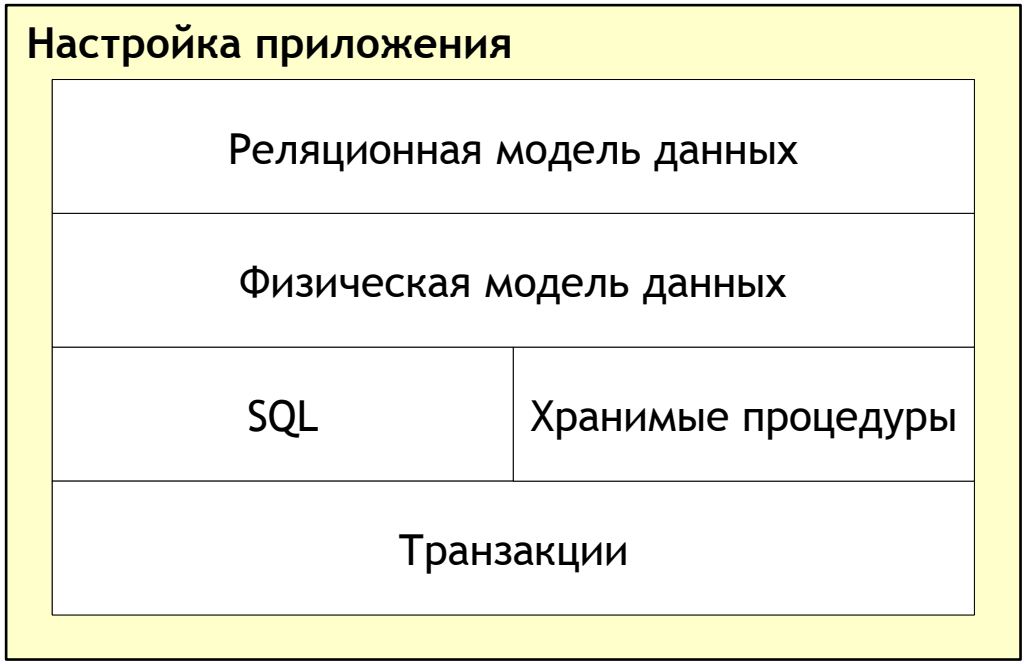
- согласованность - ограничения целостности?
- атомарность
- изоляция

Транзакции

PostgreSQL

- MVCC
  - autocommit=on — ~~согласованность~~
  - изоляция — ~~RU~~, RC, RR, SER
- Блокировки на уровне строки
- DDL

Серверная часть приложения



Серверная часть приложения

Итоги

Разработка приложений для реляционных СУБД

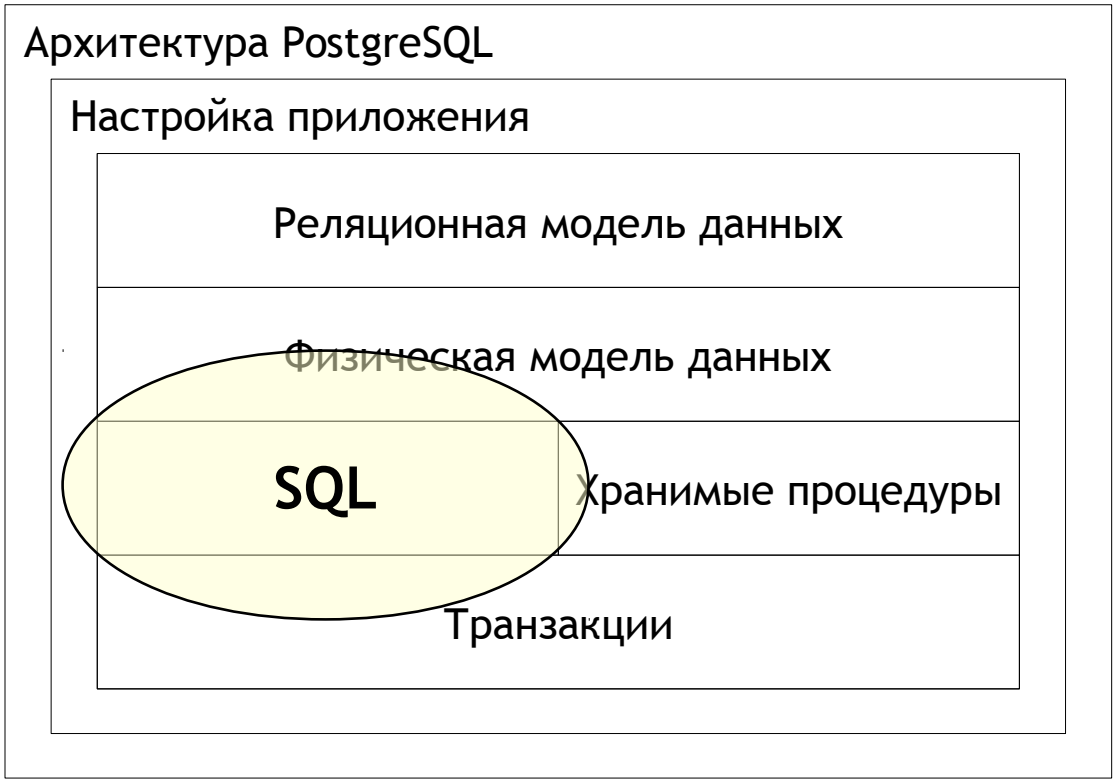
- требует знаний в ряде областей от реляционной модели до управления транзакциями

Разработка эффективных приложений для PostgreSQL

- требует понимания архитектуры PostgreSQL в этих областях

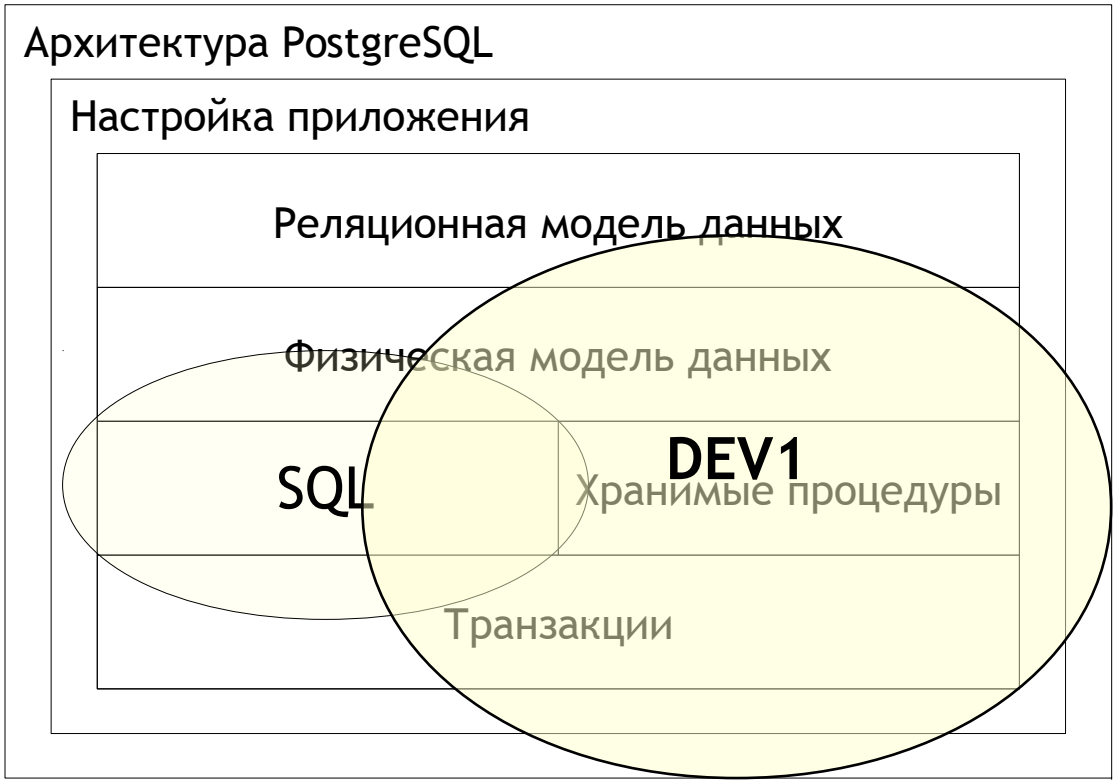
Курсы для разработчиков

Использование SQL



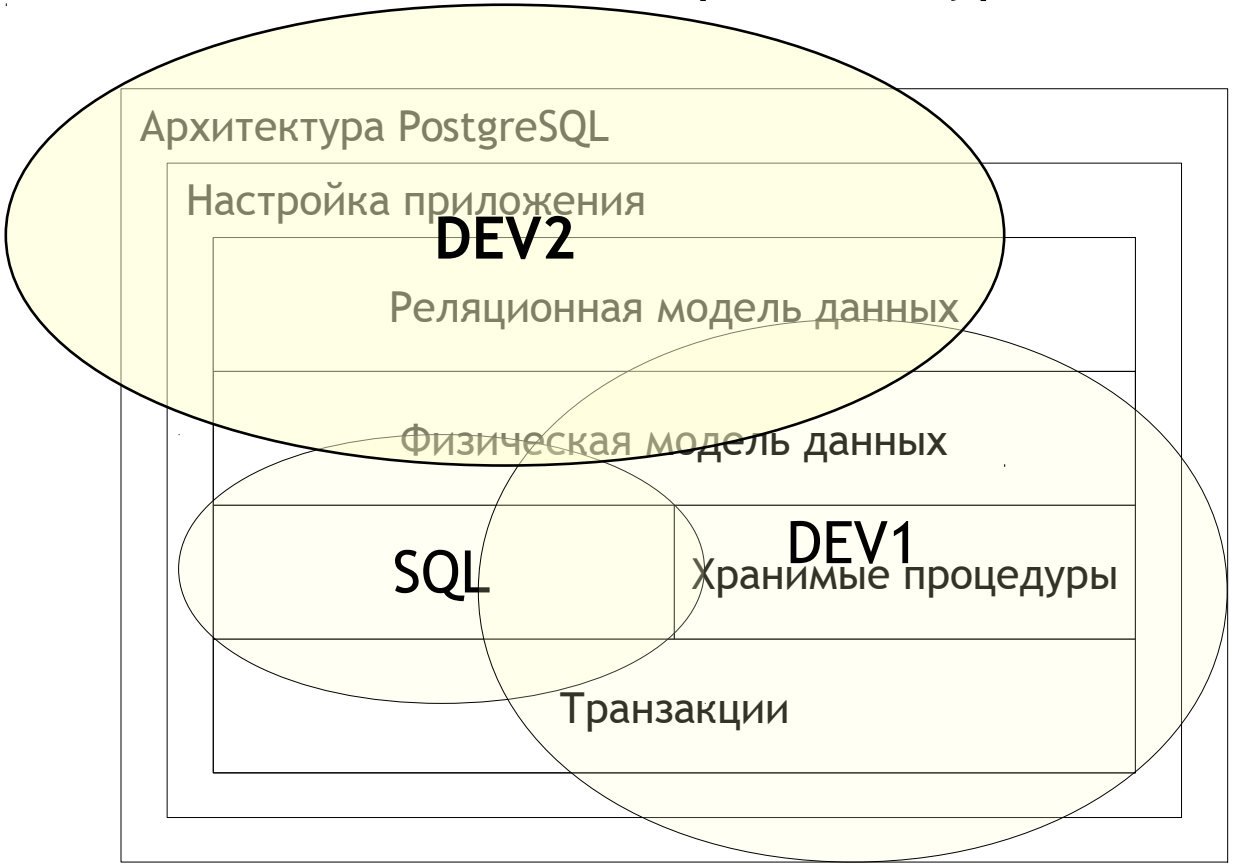
Курсы для разработчиков

DEV1. Базовый курс



Курсы для разработчиков

DEV2. Расширенный курс



Каталог курсов

www.postgrespro.ru/education

DBA1. Базовый курс

DBA2. Расширенный курс

DEV1. Базовый курс

DEV2. Расширенный курс

Использование SQL

Спасибо за внимание!

www.postgrespro.ru/education
edu@postgrespro.ru

Послесловие

PostgreSQL

Так тоже можно:

```
postgres=# create table mydata (
               id    serial,
               value jsonb
           );
CREATE TABLE
```

Зачем?