

# Использование PostgreSQL в веб-приложениях

**Николай Самохвалов**  
**Иван Золотухин**  
компания Postgresmen



# План действий

11 <sup>00</sup>	Первый кофе-брейк
12 <sup>15</sup>	Второй кофе-брейк
13 <sup>30</sup>	<b>Обед</b>
15 <sup>30</sup>	Третий кофе-брейк
16 <sup>45</sup>	Четвёртый кофе-брейк
18 <sup>00</sup>	Завершение программы

# План работ

- Вводные темы
- Установка. Инструментарий
- Устройство PostgreSQL
- Оптимизация
- Программирование
- Полнотекстовый поиск и многое другое

# Что такое PostgreSQL?

**PostgreSQL** – свободно распространяемая объектно-реляционная система управления базами данных (ORDBMS), наиболее развитая из открытых СУБД в мире и являющаяся реальной альтернативой коммерческим базам данных.

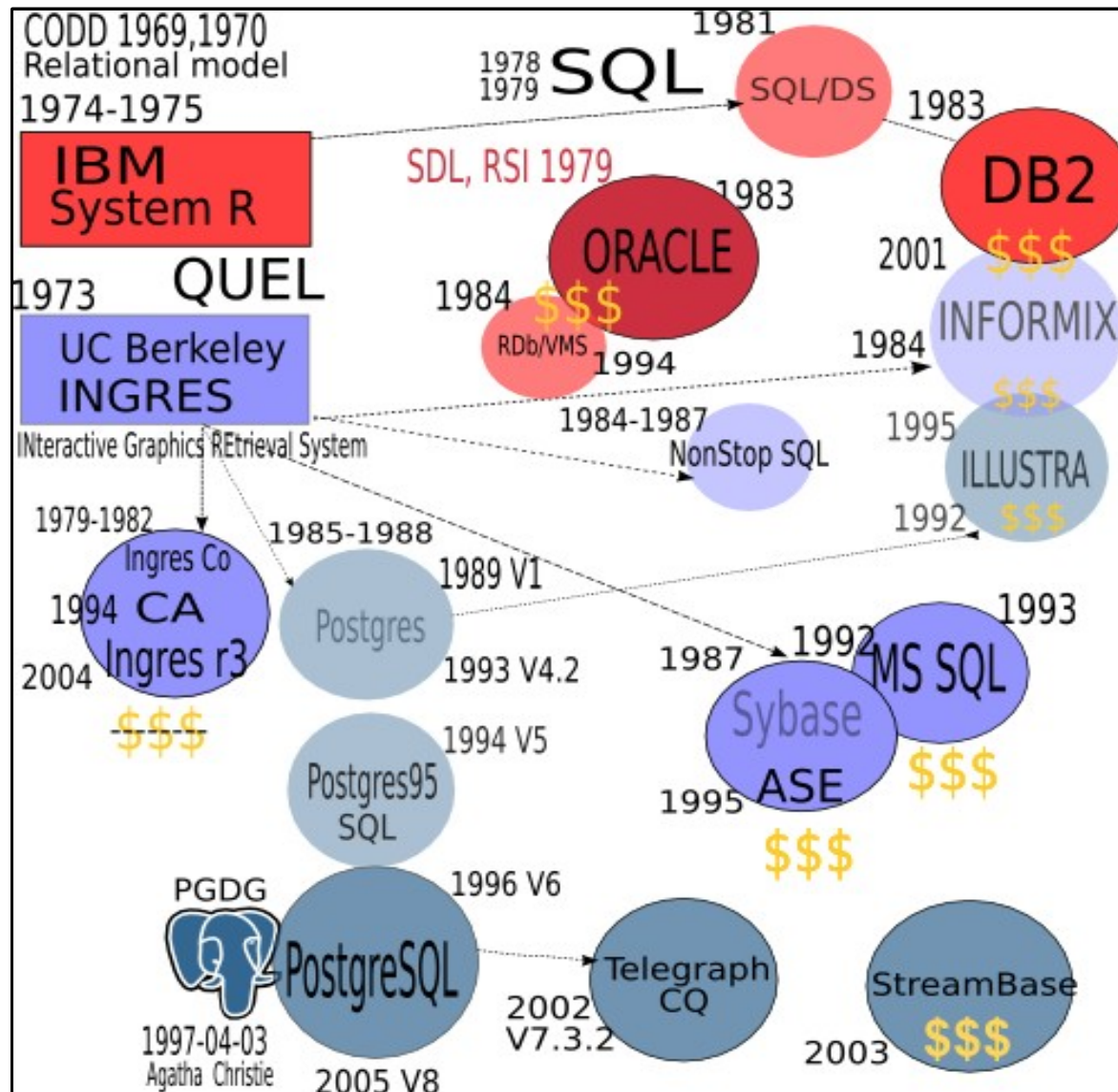
# Что такое PostgreSQL?

PostgreSQL произносится как **post-gress-Q-L**

В разговоре часто употребляется **postgres**  
(**пост-гресс**).

Также, употребляется сокращение  
**pgsql** (**пэ-жэ-эс-ку-эл**).

# Что такое PostgreSQL?



# Функциональность

Название	ASE	DB2	FireBird	InterBase	MS SQL	MySQL	Oracle	PostgreSQL
Лицензия	\$\$\$	\$\$\$	IPL <sup>2</sup>	\$\$\$	\$\$\$	GPL/\$\$\$	\$\$\$	BSD
ACID	Yes	Yes	Yes	Yes	Yes	Depends <sup>1</sup>	Yes	Yes
Referential integrity	Yes	Yes	Yes	Yes	Yes	Depends <sup>1</sup>	Yes	Yes
Transaction	Yes	Yes	Yes	Yes	Yes	Depends <sup>1</sup>	Yes	Yes
Unicode	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Schema	Yes	Yes	Yes	Yes	No <sup>5</sup>	No	Yes	Yes
Temporary table	No	Yes	No	Yes	Yes	Yes	Yes	Yes
View	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes
Materialized view	No	Yes	No	No	No	No	Yes	No <sup>3</sup>
Expression index	No	No	No	No	No	No	Yes	Yes
Partial index	No	No	No	No	No	No	Yes	Yes
Inverted index	No	No	No	No	No	Yes	Yes	Yes
Bitmap index	No	Yes	No	No	No	No	Yes	No <sup>6</sup>
Domain	No	No	Yes	Yes	No	No	Yes	Yes
Cursor	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes
User Defined Functions	Yes	Yes	Yes	Yes	Yes	No <sup>4</sup>	Yes	Yes
Trigger	Yes	Yes	Yes	Yes	Yes	No <sup>4</sup>	Yes	Yes
Stored procedure	Yes	Yes	Yes	Yes	Yes	No <sup>4</sup>	Yes	Yes
Tablespace	Yes	Yes	No	?	No <sup>5</sup>	No <sup>1</sup>	Yes	Yes
Название	ASE	DB2	FireBird	InterBase	MS SQL	MySQL	Oracle	PostgreSQL

# Некоторые ограничения

Название	Значение
Максимальный размер БД	Unlimited
Максимальный размер таблицы	32 TB
Максимальная длина записи	1.6 TB
Максимальная длина атрибута	1 GB
Максимальное количество записей в таблице	Unlimited
Максимальное количество атрибутов в таблице	250 - 1600 в зависимости от типа атрибута
Максимальное количество индексов на таблицу	Unlimited

# PostgreSQL сегодня

Текущая версия – 8.2 (8.2.3)

Релиз версии 8.3 ожидается в июне 2007

# PostgreSQL сегодня

- Надежность

ACID, MVCC, WAL, PITR, Slony

- Безопасность данных

root, SSL, pg\_hba.conf, ROLE

- Производительность

B-tree, hash, R-tree, GiST, Gin; geqo; partitioning; Slony, pgpool

- Расширяемость

pg\_catalog, наследование, GiST, Gin, contribs

# PostgreSQL сегодня

- ISO/ANSI SQL (SQL:200x)  
схемы, представления, триггеры, rules, 2PC...
- Типы данных  
varlena, массивы, GIS, композитные, GiST
- Интерфейсы  
C, C++, C#, python, perl, ruby, php, Lisp и т.д.
- Процедурные языки  
PL/pgSQL, pl/Tcl, PL/Perl и pl/Python; PHP, Java, Ruby, R, shell

# PostgreSQL сегодня

- Простота

# Кто использует

- Sony Entertainment (EnterpriseDB)
- Skype
- Cisco
- Fujitsu
- NTT
- Apple
- SUN Microsystems (Solaris, 24x7 support)

# Кто использует

- SourceForge
- LAMP: Linux/Apache/ Middleware(Perl,PHP,Python,Ruby)/ PostgreSQL
- New Zealand's Electoral Enrolment Centre
- .ORG, .INFO domain registry
- Многотерабайтные архивы астрономических данных

# Кто использует: Россия

- Рамблер
- 1С:Предприятие (наряду с MS SQL)
- Многие крупные проекты Рунета
- Beeline

# Интерфейсы абстракции (PHP)

✓ Good Thing

Примеры:

- PDO
- PEAR::DB
- ADOdb
- Creole
- DbSimple

# ПО на PHP с поддержкой PostgreSQL

- Blog: Serendipity, <http://s9y.org/>
- Wiki: Mediawiki, <http://www.mediawiki.org/>
- CMS: Drupal, <http://drupal.org/>
- CMF: CakePHP, <http://www.cakephp.org/>
- Bug-tracker: Mantis, <http://www.mantisbt.org/>
- Forum: FUDforum, <http://fudforum.org/>

# Установка в связке с PHP

- `cd postgresql-8.2.3`  
`./configure`  
`make`  
`make install`
- `cd php-5.2.1`  
`./configure --with-pgsql [--with-apxs]`  
`make`  
`make install`

# Начальная настройка

Всего 2 файла:

- `postgresql.conf` (основные настройки)
  - `listen_addresses` (сетевые интерфейсы)
  - `shared_buffers` (общая память)
  - `work_mem` (память для сортировок)
- `pg_hba.conf` (настройки доступа)
  - `host all all 127.0.0.1/32 md5`

# Стартовые скрипты

- **Linux:** `contrib/start-scripts/linux`
- **FreeBSD:** `contrib/start-scripts/freebsd`
- **Windows:** обычная служба
- **Любая другая ОС:** не изобретайте, все придумано до вас. Google it!

# Терминал psql

- отличный терминал
- многострочные команды и история (Сергей Копосов) \n
- autocompletion Tab
- встроенная помощь \h
  
- \i, \o, psql -c, echo | psql
- psql < file, etc

# GUI средства: pgAdmin III

The screenshot displays the pgAdmin III interface. The top menu bar includes File, Edit, View, Tools, and Help. Below the menu is a toolbar with various icons for file operations and database management. The main window is divided into three sections:

- Object browser:** A tree view showing the database structure. The selected database is 'aspid\_sr' under 'localhost (localhost:5432)'. The tree includes Servers (3), Databases (3), Casts (9), Languages (1) (plpgsql), Schemas (1) (public), Replication (0), postgres, and wiki.
- Properties window:** A table showing the properties of the selected database 'aspid\_sr'. The table has two columns: Property and Value.

Property	Value
Name	aspid_sr
OID	16787
Owner	postgres
ACL	
Tablespace	pg_default
Encoding	UTF8
Allow connections?	Yes
Connected?	Yes
System database?	No
Comment	

- SQL pane:** A text area containing SQL commands. The first two lines are comments: '-- Database: aspid\_sr' and '-- DROP DATABASE aspid\_sr;'. The third line is the command: 'CREATE DATABASE aspid\_sr WITH OWNER = postgres'.

At the bottom of the window, a status bar shows 'Restoring previous environment... Done.' and a timer indicating '1,02 secs'.

# GUI средства: phpPgAdmin

phpPgAdmin - Mozilla Firefox

File Edit View Go Bookmarks Tools Help delicio.us

http://phpPgAdmin.kattare.com/phpPgAdmin4/

Getting Started Latest Headlines Новости

Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source Options

Gmail - Inbox PostgreSQL: Frequ... 0000186: Настро... 404 Not Found CVS -- Система У... PHP: PostgreSQL F... phpPgAdmin

phpPgAdmin

PostgreSQL 7.3.17 running on localhost:5473 – You are logged in as user "phpPgAdmin", 28th Mar, 2007 11:38AM SQL | Find | Logout

phpPgAdmin: PostgreSQL 7.3.17? marek?

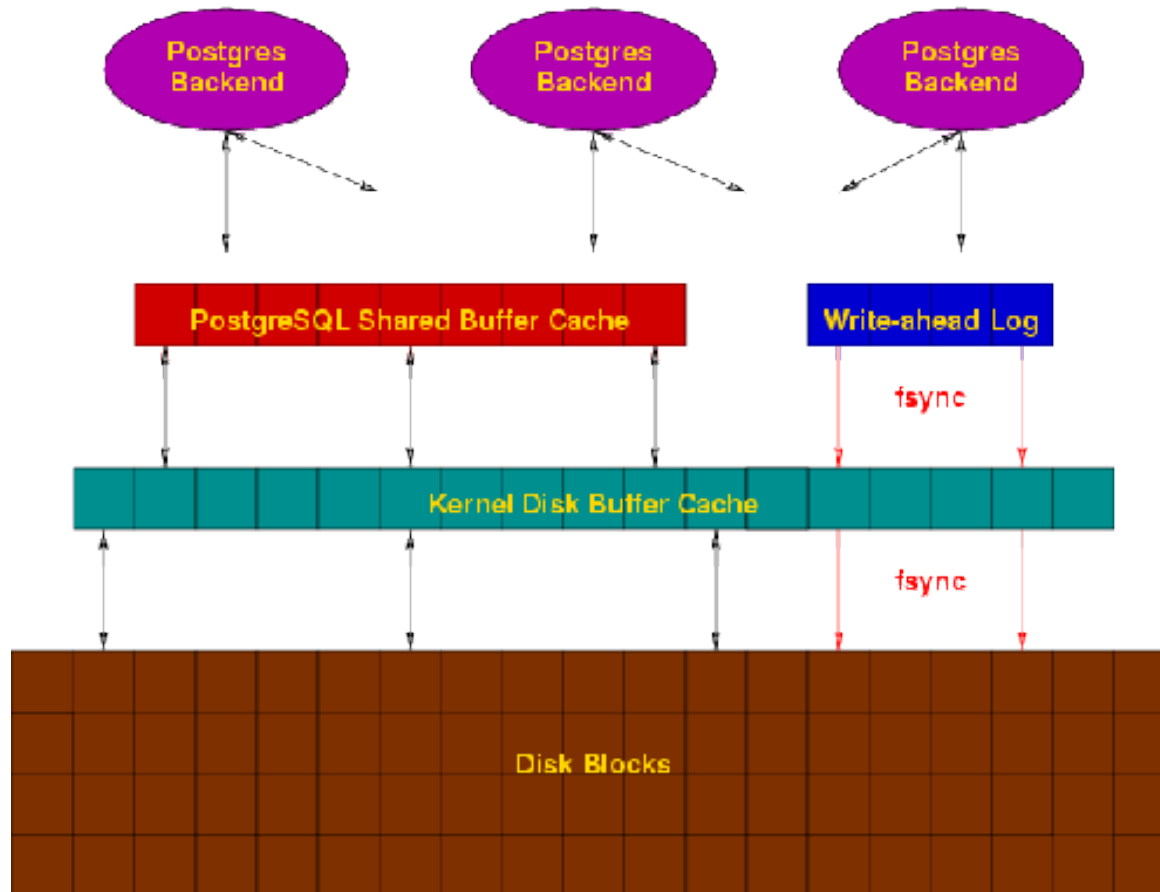
Schemas? SQL? Find Variables? Processes? Locks? Admin Privileges? Export

Type	User/Group	CREATE	TEMPORARY
	public	No	No
	user phpPgAdmin	Yes	Yes
	user test	Yes	Yes

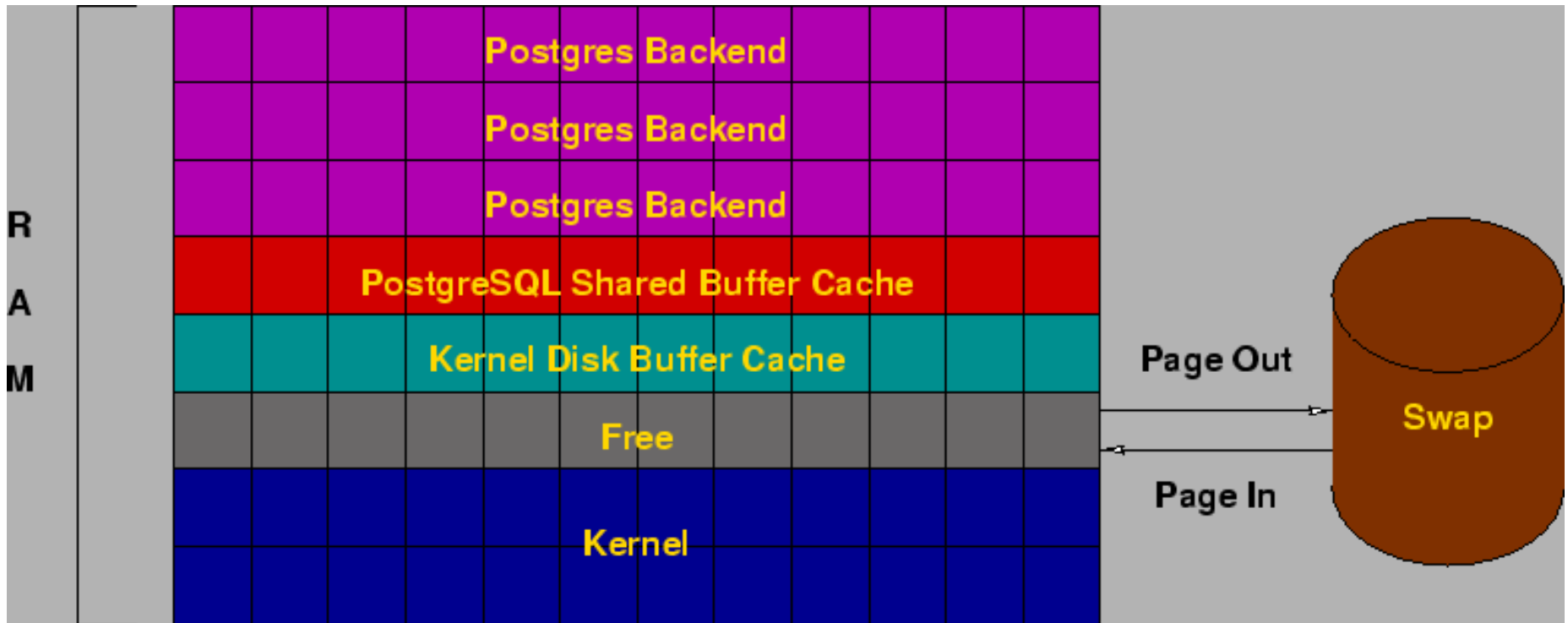
Grant | Revoke | Show all databases

Done 0.673s

# Устройство PostgreSQL



# Устройство PostgreSQL



# Устройство PostgreSQL

- ACID-совместимая база данных
  - atomicity (атомарность)
  - consistency (непротиворечивость)
  - isolation (изоляция)
  - durability (надежность)

# Устройство PostgreSQL

- MVCC: Multiversion Concurrency Control
  - xid – transaction id
  - каждая запись имеет xid\_start и xid\_end
  - каждая транзакция видит версию базы в момент xid\_start
  - записи не удаляются, а просто помечаются xid\_end

# Устройство PostgreSQL

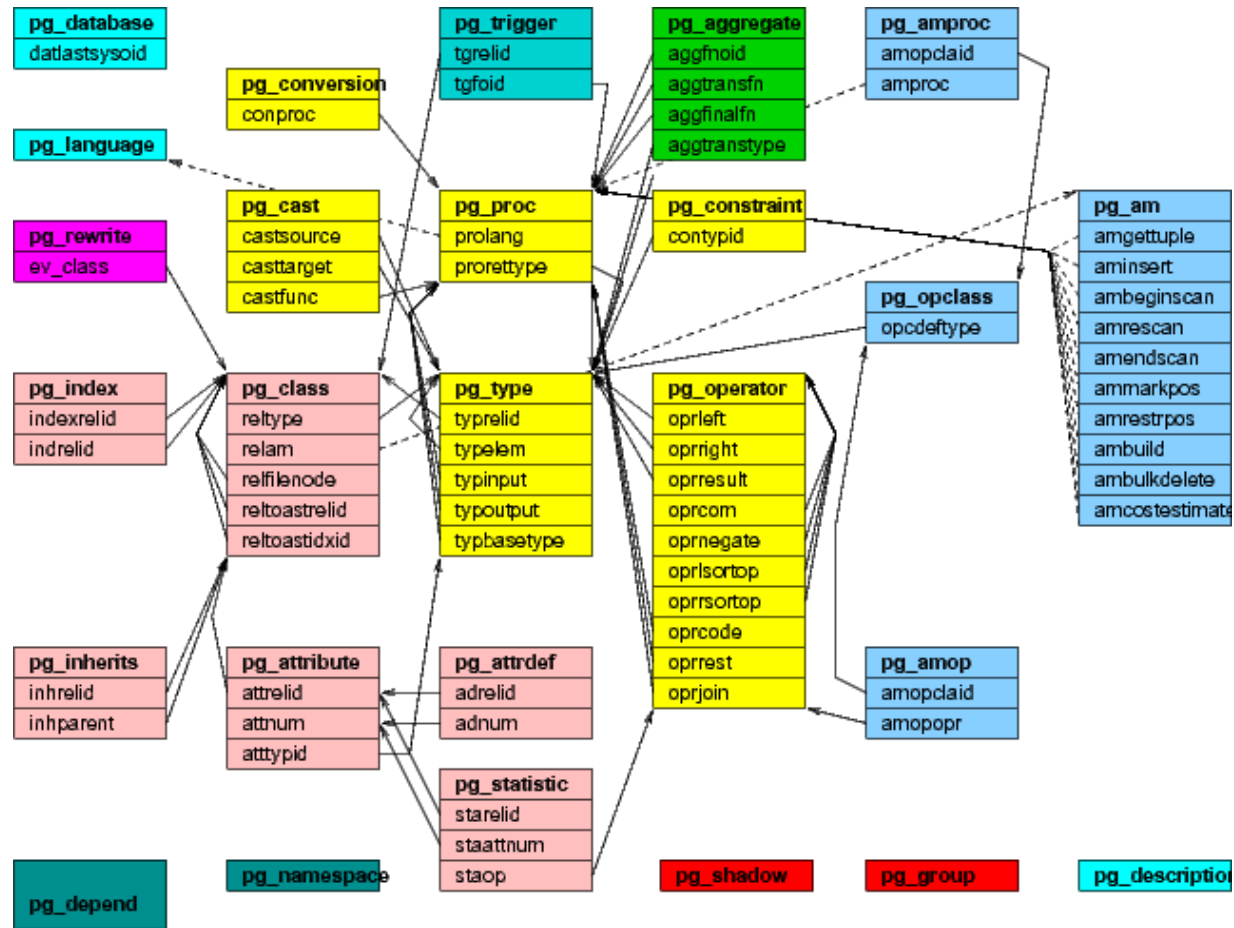
- MVCC – накапливаются старые версии данных
- требует пылесоса – VACUUM
- VACUUM: re-use мертвых данных
- VACUUM FULL: физическое удаление мертвых данных и дефрагментация базы
- autovacuum

# Устройство PostgreSQL

WAL – Write Ahead Log

- механизм протоколирования всех транзакций
- позволяет восстановить систему после возможных сбоев
- **все изменения данных записываются на диск только после их гарантированного журналирования в WAL**
- PITR – Point In Time Recovery

# Устройство PostgreSQL



# postgresql.conf

- `shared_buffers`  
~ ¼ RAM, старайтесь положить всю базу в память!
- `work_mem`  
свободная RAM / макс. число соединений
- `effective_cache_size`  
средний размер дискового кеша
- `checkpoint_timeout`  
несколько минут
- `checkpoint_segments`  
N \* 16MB

# Производительность PostgreSQL

## Hardware

- Диски > RAM > CPU
- Чем больше дисков, тем лучше
- Отделяйте pg\_xlog от данных
- RAID 1+0 / 0+1 > RAID 5
- Не ставьте на сервер другие приложения

# Индексы в PostgreSQL

- B-tree
- Hash
- R-tree
- GiST (обобщенное поисковое дерево)
- GIN (обратный индекс)

# Обработка запроса в PostgreSQL

1. Parser (синтаксический анализатор)
2. Planner (выбор оптимального пути)
3. Executor (непосредственное выполнение)

SQL – декларативный язык. СУБД решает, как именно будет выполняться запрос.

# Обработка запроса в PostgreSQL

1. Parser (синтаксический анализатор)
2. Planner (выбор оптимального пути)
3. Executor (непосредственное выполнение)

SQL – декларативный язык. СУБД решает, как именно будет выполняться запрос.

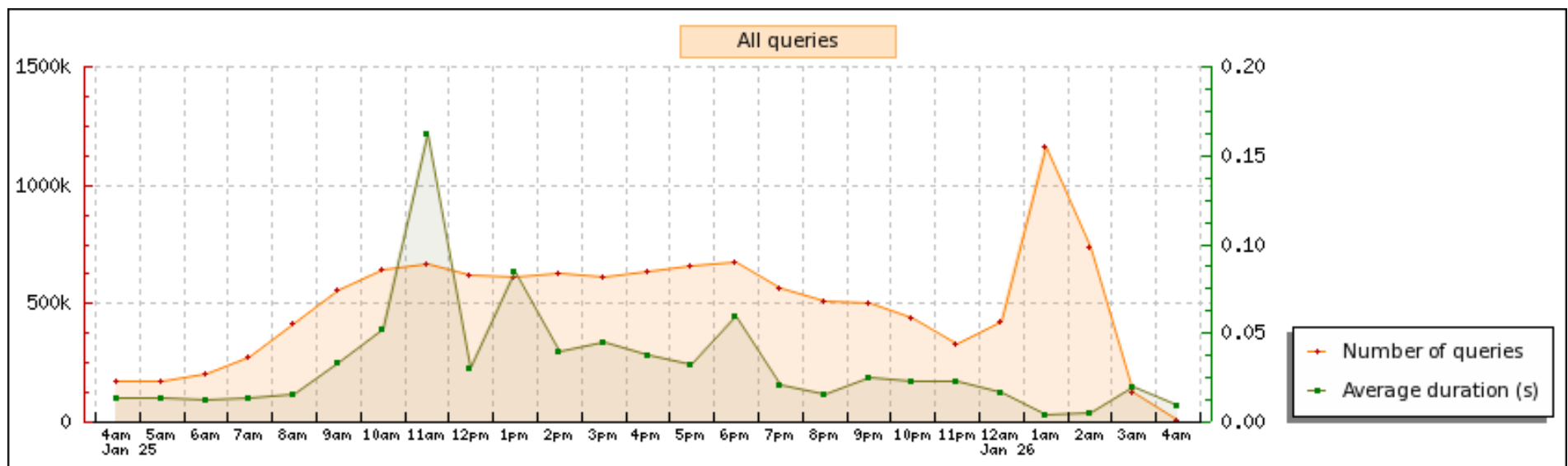
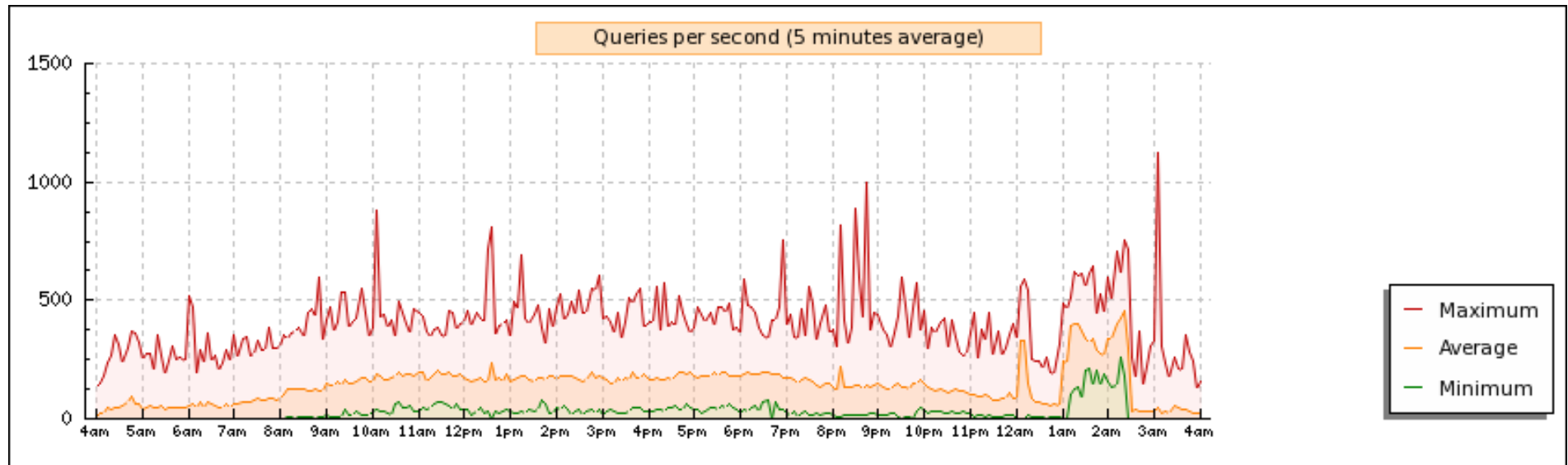
# EXPLAIN ANALYZE

- План запроса – дерево
- Узлы – действия
  - соединения (join)
  - сортировка
  - просмотр таблицы
- Выполнение происходит от листьев к корню
- Оценка количества строк и стоимости

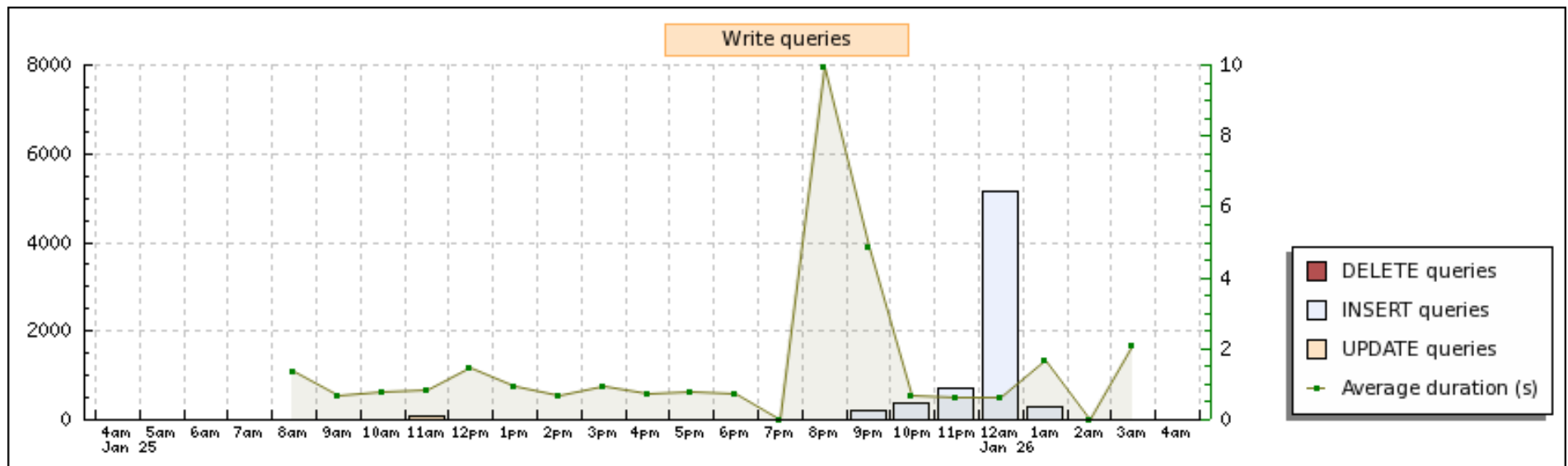
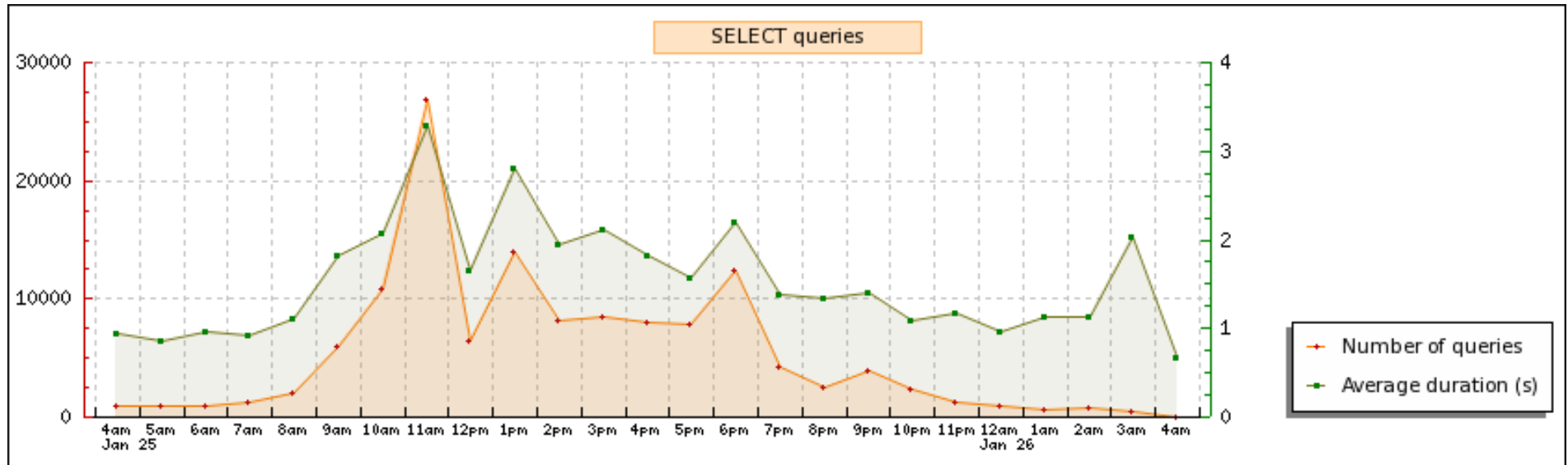
# EXPLAIN ANALYZE

- Способы просмотра таблицы
  - Seq Scan
  - Index Scan
- Способы подготовки данных
  - Sort
  - Hash
- Способы соединения (join)
  - Nested Loop
  - Merge Join
  - Hash Join

# pgFouine: анализ логов



# pgFouine: анализ логов



# Наследование

```
CREATE TABLE obj (  
    obj_id INTEGER,  
    obj_created TIMESTAMPTZ DEFAULT CURRENT_TIMESTAMP  
);
```

```
CREATE TABLE person(  
    obj_id INTEGER PRIMARY KEY,  
    person_name VARCHAR(32) NOT NULL,  
    CHECK(obj_id >= 1000000000 AND obj_id < 2000000000)  
) INHERITS(obj);
```

```
CREATE TABLE car(  
    obj_id INTEGER PRIMARY KEY,  
    obj_type_id INT2 NOT NULL DEFAULT 2,  
    car_model VARCHAR(16) NOT NULL,  
    CHECK(obj_id >= 2000000000 AND obj_id < 3000000000)  
) INHERITS(obj);
```

# Функции / хранимые процедуры

- **SQL**
- **PL/pgSQL**
- **PL/Perl**
- PL/Tcl
- PL/Python
- PL/R
- PL/Java
- plPHP
- plRuby
- ...и даже больше!

# Функции: SQL

```
CREATE OR REPLACE FUNCTION hello(text)
RETURNS text
AS $BODY$
    SELECT 'Hello ' || $1 || '!';
$BODY$ LANGUAGE sql IMMUTABLE;
```

# Функции: PL/pgSQL

```
CREATE OR REPLACE FUNCTION hello(text)
RETURNS text
AS $BODY$
BEGIN
    RETURN 'Hello ' || $1 || '!';
END;
$BODY$ LANGUAGE plpgsql IMMUTABLE;
```

# Триггеры

```
CREATE OR REPLACE FUNCTION ts2_page_title()
RETURNS "trigger"
AS $BODY$
BEGIN
    IF TG_OP = 'INSERT' THEN
        NEW.titlevector =
            to_tsvector('default',NEW.page_title);
    ELSIF NEW.page_title <> OLD.page_title THEN
        NEW.titlevector :=
            to_tsvector('default',NEW.page_title);
    END IF;
RETURN NEW;
END;
$BODY$ LANGUAGE plpgsql VOLATILE;

CREATE TRIGGER ts2_page_title BEFORE INSERT OR UPDATE
ON page FOR EACH ROW EXECUTE PROCEDURE ts2_page_title();
```

# Rules и представления

```
CREATE VIEW myview AS SELECT * FROM mytab;
```

```
CREATE TABLE myview(...);
```

```
CREATE RULE "_RETURN" AS ON SELECT TO myview  
DO INSTEAD  
    SELECT * FROM mytab;
```

# GiST – Generalized Search Tree

- R-tree
- tsearch2
- intarray
- pg\_trgm
- ltree
- hstore
- pg\_sphere

# **The End**

Спасибо за внимание!