Postgres вчера, сегодня, завтра

Олег Бартунов,
Александр Коротков
Postgres Professional

26 апреля 2016, Санкт-Петербург

Что такое PostgreSQL

PostgreSQL - это свободно распространяемая объектно-реляционная СУБД (ORDBMS)

Расширяемая — типы данных, операторы, функции, индексы

Поддержка ANSI SQL (1992, 1999, 2003, 2011), NoSQL (key-value, JSON, JSONB)

Разрабатывается независимым мировым сообществом с существенным российским участием

Произношение: post-gress-Q-L, post-gres, пост-грес, pgsql (пэ-жэ-эс-ку-эль)

Web: http://www.postgresql.org, лицензия: BSD, MIT - like

Российский вендор - компания Postgres Professional

Важнейшие свойства PostgreSQL

Надежность и устойчивость PostgreSQL

Надежность PostgreSQL является известным и доказанным фактом на примере многих проектов, в которых PostgreSQL работает без единого сбоя и при больших нагрузках на протяжении нескольких лет.

Кроссплатформенность

PostgreSQL поддерживает все виды Unix, включая Linux, FreeBSD, Solaris, HPUX, Mac OS X, а также MS Windows.

Конкурентная работа при большой нагрузке

PostgreSQL использует многоверсионность (MVCC) для обеспечения надежной и быстрой работы в конкурентных условиях под большой нагрузкой.

Масштабируемость

PostgreSQL отлично использует современную архитектуру многоядерных процессоров - его производительность растет линейно до 64-х ядер. Кластерные решения на базе Postgres XL обеспечивают горизонтальную масштабируемость.

Расширяемость

Расширяемость PostgreSQL позволяет добавлять новую функциональность, в том числе и новые типы данных, без остановки сервера и своими силами.

Доступность

PostgreSQL распространяется под лицензией BSD, которая не накладывает никаких ограничений на коммерческое использование и не требует лицензионных выплат. Вы можете даже продавать PostgreSQL под своим именем !

Независимость

PostgreSQL не принадлежит ни одной компании, он развивается международным сообществом, в том числе и российскими разработчиками. Независимость PostgreSQL означает независимость вашего бизнеса от вендора и сохранность инвестиций.

Превосходная поддержка

Сообщество PostgreSQL предоставляет квалифицированную и быструю помощь. Коммерческие компании предлагают свои услуги по всему миру.

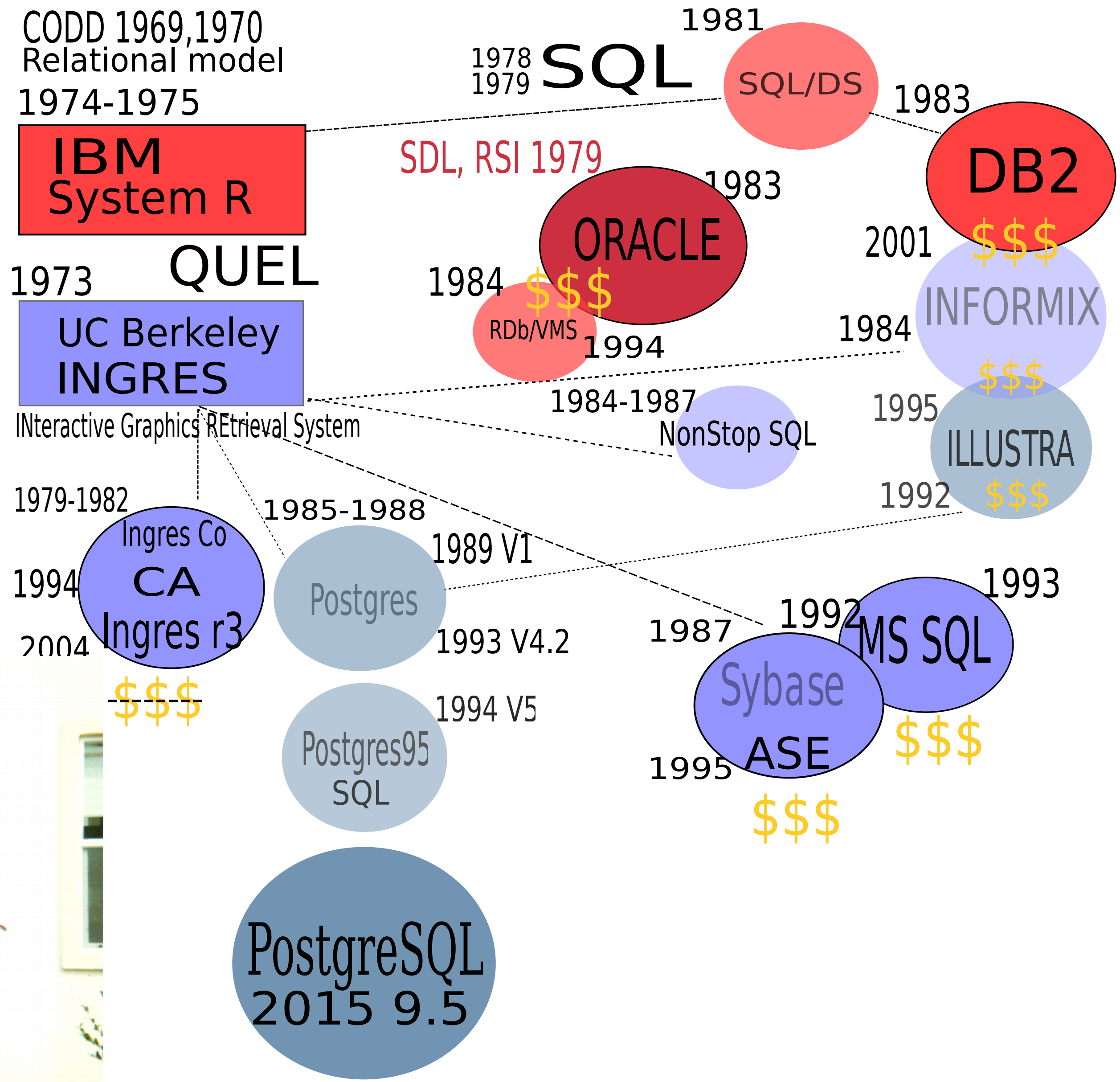
PROFESSIONAL Postgres

Пользователи PostgreSQL

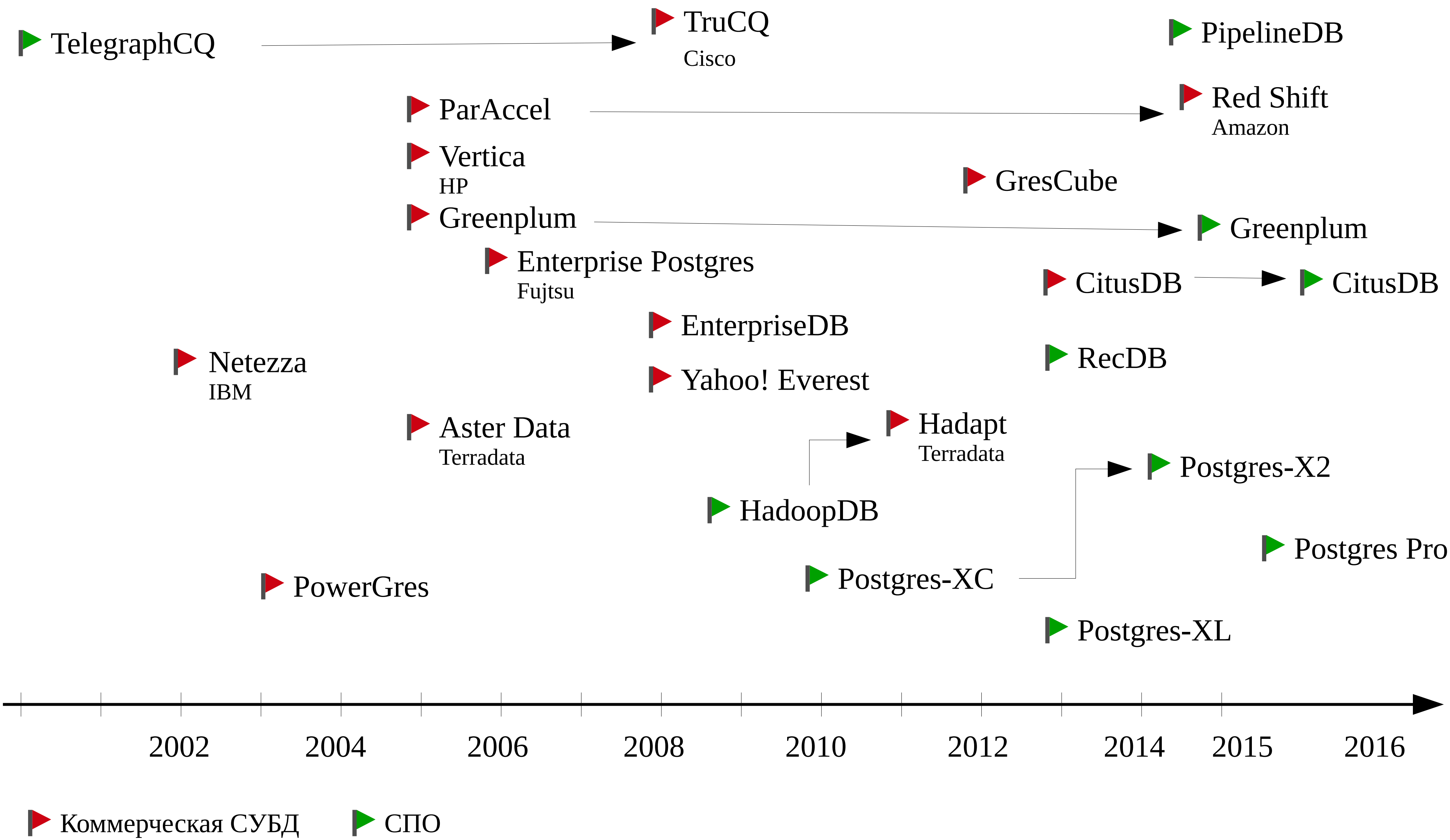
ПРФ, Московская Область, РосАтом, МО, Роснефть, МКС, РИА

История PostgreSQL

Michael Stonebreaker
Turing Award, 2015



Форки PostgreSQL



https://wiki.postgresql.org/wiki/PostgreSQL_derived_databases

Этапы развития PostgreSQL

Стабилизация работы	Совместимость с SQL стандартами	Возможности уровня Enterprise / простота использования

1996 1998 2001 2016

Базовая функциональность JDBC MVCC Optimizer Stats PL/pgSQL **Стабилизация** Исправление сбоев в работе Очистка кода Культура совершенства	**Стандарты** SQL 92 Joins Prepared queries Foreign Keys **Функциональность ядра** Write Ahead Log Prepared Queries Info. Schema Auto Vacuum	**Возможности уровня Enterprise** Потоковая репликация Производительность Вертикальное масштабирование PITR pg_upgrade NoSQL BDR Параллелизм **Простота использования** Портирование на Windows pg_basebackup Различные инструменты

Этапы развития PostgreSQL

Совместимость с SQL стандартами	Возможности уровня Enterprise / простота использования	Горизонтальное масштабирование (Шардинг)
1998	2001	31.10.2015

1998

Стандарты
- SQL 92 Joins
- Prepared queries
- Foreign Keys

Функциональность ядра
- Write Ahead Log
- Prepared Queries
- Info. Schema
- Auto Vacuum

2001

Возможности уровня Enterprise
- Потоковая репликация
- Производительность
- Вертикальное масштабирование
- PITR
- pg_upgrade
- NoSQL
- BDR
- Параллелизм

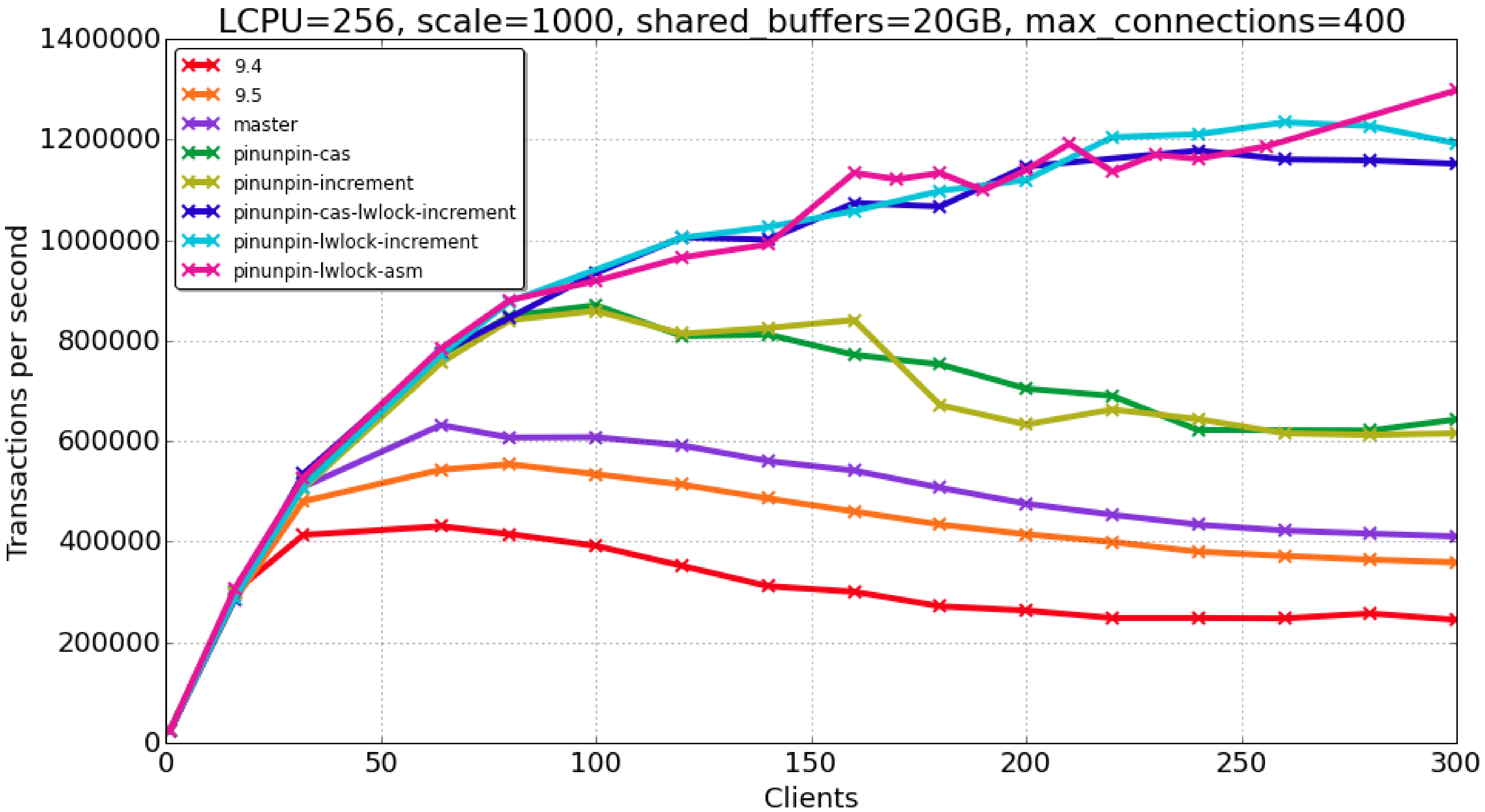
Простота использования
- Портирование на Windows
- pg_basebackup
- Различные инструменты

31.10.2015

Цели
- Read-Write scalability
- High availability

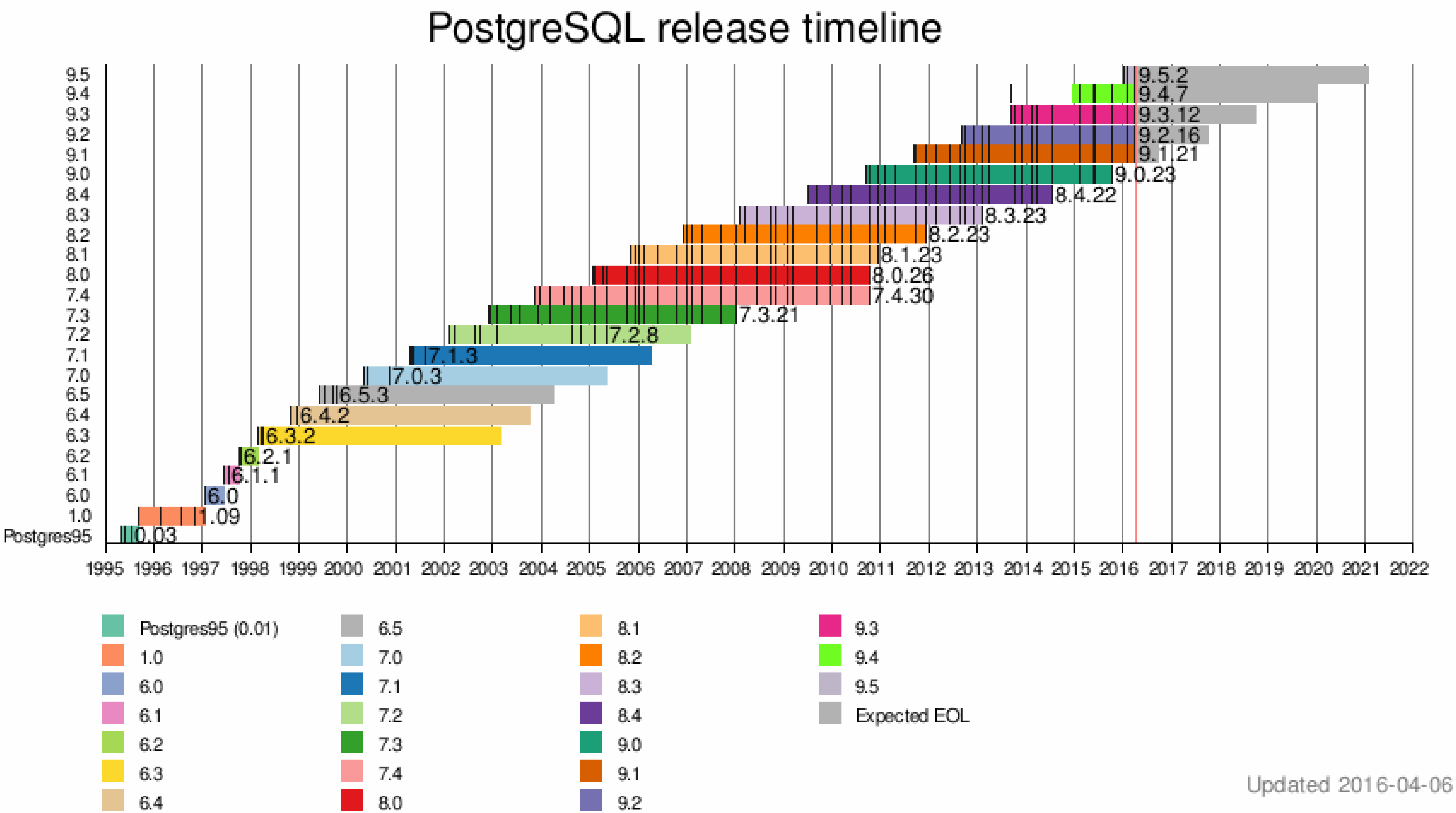
Задачи
- Управление распределенными транзакциями
- Планировщик и исполнитель распределенных запросов
- Обнаружение распределенных взаимных блокировок
- Онлайн перераспределение данных
- Поддержка глобальных ограничений
- Онлайн добавление и удаление узлов
- Средства администрирования

Вертикальная масштабируемость



http://habrahabr.ru/company/postgrespro/blog/270827/

Версии PostgreSQL (~300)



Цикл разработки

- 1-30 сентября 2015 : CF 1
- 1-30 ноября 2015 : CF 2
- 2 января — 8 февраля 2016 : CF 3
- 7 января 2016 : 9.5 Release
- 1-31 марта 2016 : Финальный CF для 9.6
- **8 апреля : Feature Freeze**
- Июнь 2016 : 9.6 beta
- Сентябрь 2016 : 9.6 release
- 1-30 сентября 2016 : CF 1 для 9.7 ….

Свежие новости PostgreSQL

9.5

- INSERT … ON CONFLICT (aka Upsert)
- Row Level Security
- BRIN index
- Ускоренная сортировка (abbreviated keys)
- GROUP BY … GROUPING SETS, CUBE, ROLLUP (многоуровневые агрегаты)
- Улучшение FDW (JOIN pushdown, import schema)
- SELECT … SKIP LOCKED
- SELECT … TABLESAMPLE
- Vacuumdb -j (выполнять вакуум параллельно)

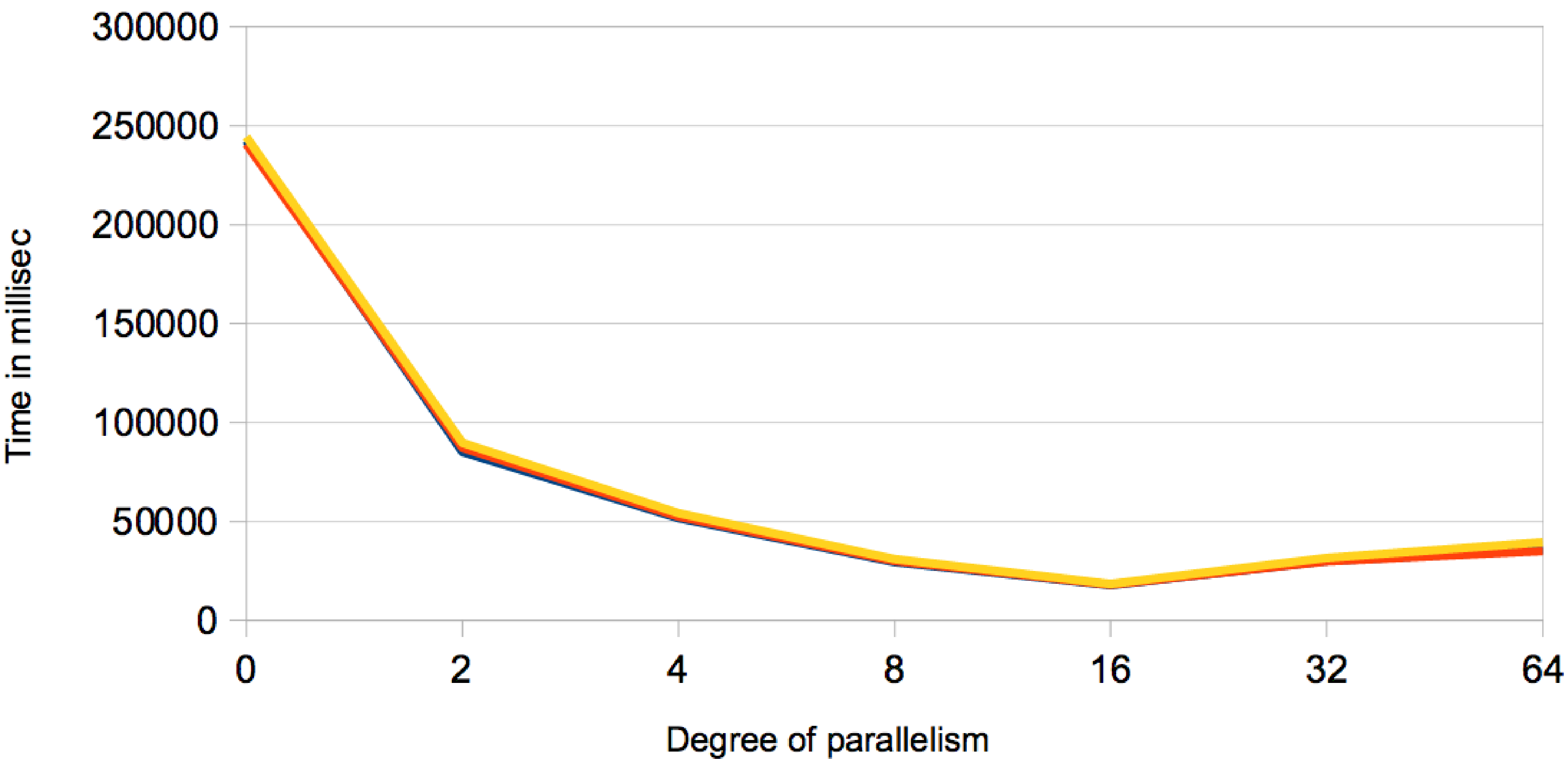
UNFORK

- Citus DB – распределенная СУБД в виде extension
- GreenPlum – на пути к этому

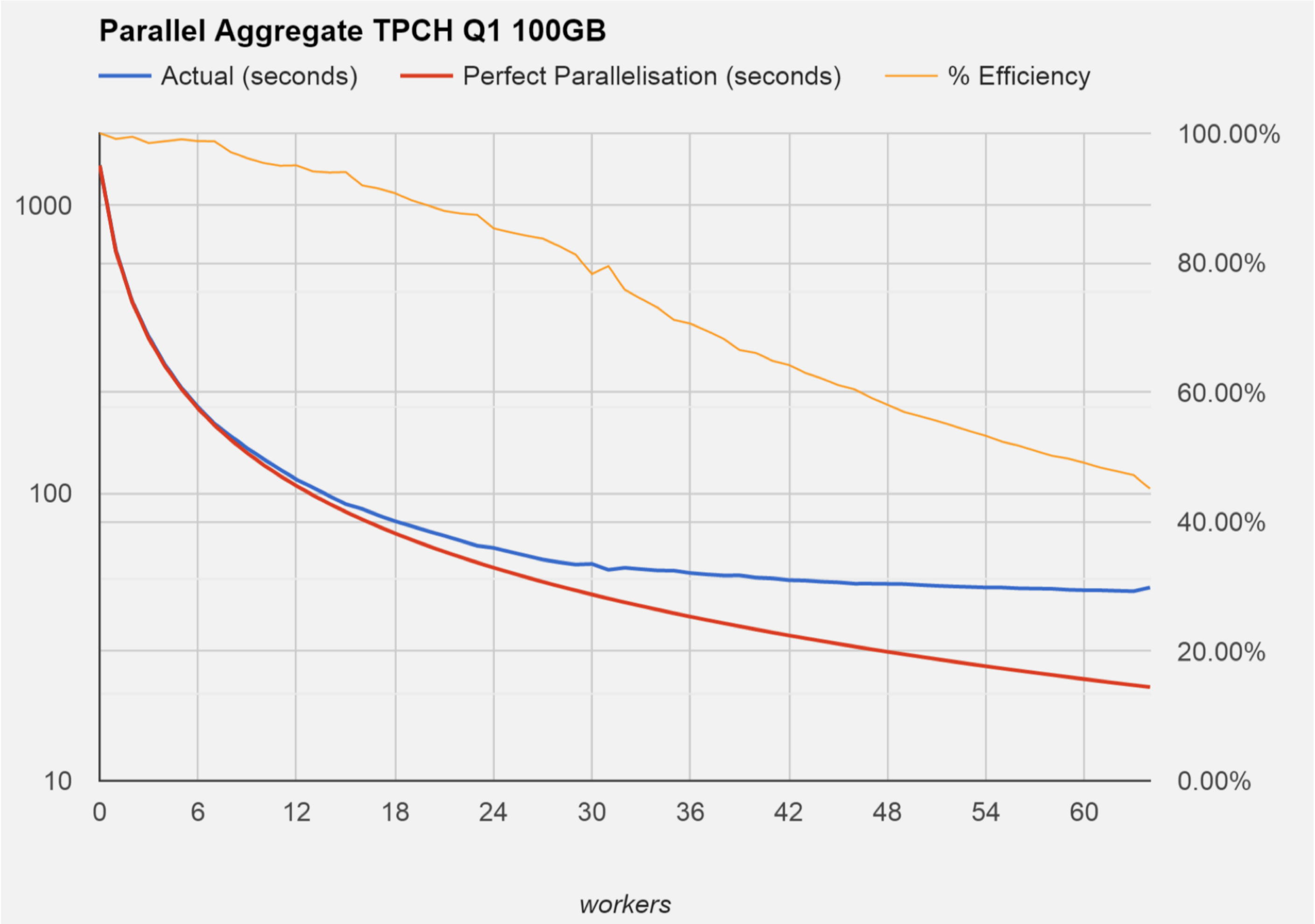
Что нас ждет «завтра» 9.6 ?

- Параллельное исполнение запросов (sequence scan, join, aggregate)
- Улучшение работы VACUUM с большими таблицами
- Extend relations multiple blocks at a time to improve scalability
- Улучшение FDW (push down join, DML, sort)
- Indexing boxes and polygons with SP-GiST
- CREATE ACCESS METHOD, GENERIC WAL
- Улучшение полнотекстового поиска — поиск фраз, улучшена поддержка словарей, функции для манипулирования с tsvector
- \ev, \sv — редактирование и показ представлений
- KNN для CUBE
- Улучшение статистики: FK, GROUP BY
- Комбинирование агрегатов
- IOS для частичных индексов
- Wait monitoring
- Масштабирование на большое количество ядер
- Покрывающие (covering) индексы -> 9.7
- Параллельное создание GIN-индексов → 9.7

Завтра. Параллельный seqscan



Завтра. Параллельная агрегация



Завтра (9.6). DBA

- Немного счастья для DBA
  - VACUUM FREEZE не будет трогать уже «замороженные» блоки — сильное облегчение для больших и нагруженных проектов
  - idle_in_transaction_session_timeout - решает проблему «idle in transaction"
  - a generic command progress reporting facility
  - simple VACUUM progress reporting — можно следить за выполнением вакуума pg_stat_progress_vacuum
  - per-tablespace effective_io_concurrency
  - Поддержка **нескольких** синхронных реплик
  - Новый режим синхронной репликации - 'remote_apply'

Завтра (9.6). DBA

Представление pg_config

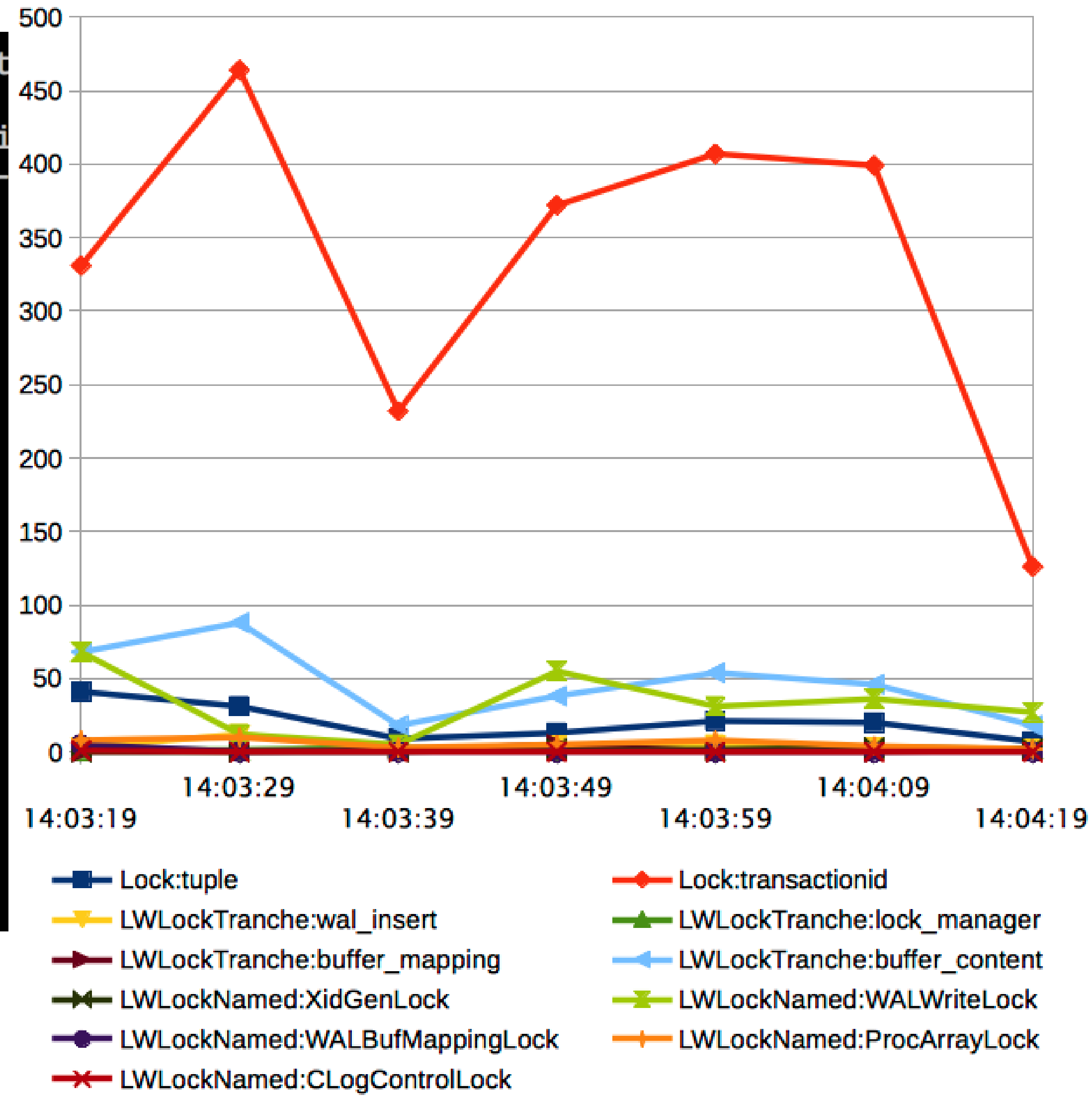
```
[local]:5432 postgres@postgres=# select * from pg_config;
       name        |
           setting

-------------------+-------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------
---------
 BINDIR            | /usr/local/pgsql-head/bin
 DOCDIR            | /usr/local/pgsql-head/share/doc
 HTMLDIR           | /usr/local/pgsql-head/share/doc
 INCLUDEDIR        | /usr/local/pgsql-head/include
 PKGINCLUDEDIR     | /usr/local/pgsql-head/include
 INCLUDEDIR-SERVER | /usr/local/pgsql-head/include/server
 LIBDIR            | /usr/local/pgsql-head/lib
 PKGLIBDIR         | /usr/local/pgsql-head/lib
 LOCALEDIR         | /usr/local/pgsql-head/share/locale
 MANDIR            | /usr/local/pgsql-head/share/man
 SHAREDIR          | /usr/local/pgsql-head/share
 SYSCONFDIR        | /usr/local/pgsql-head/etc
 PGXS              | /usr/local/pgsql-head/lib/pgxs/src/makefiles/pgxs.mk
 CONFIGURE         | '--prefix=/usr/local/pgsql-head' '--enable-depend' '--with-libs=/usr/local/lib'
 CC                | gcc
 CPPFLAGS          | -DFRONTEND
 CFLAGS            | -Wall -Wmissing-prototypes -Wpointer-arith -Wdeclaration-after-statement -Wendif-labe
ls -Wmissing-format-attribute -Wformat-security -fno-strict-aliasing -fwrapv -Wno-unused-command-line-argu
ment -O2
 CFLAGS_SL         |
 LDFLAGS           | -L../../src/common -L/usr/local/lib -Wl,-dead_strip_dylibs
 LDFLAGS_EX        |
 LDFLAGS_SL        |
 LIBS              | -lpgcommon -lpgport -lz -lreadline -lm
 VERSION           | PostgreSQL 9.6devel
(23 rows)
```

- Wait monitoring

```
[local]:5432 postgres@postgres=# \d pg_stat_activit
          View "pg_catalog.pg_stat_activity"
     Column      |           Type           | Modi
-----------------+--------------------------+-----
 datid           | oid                      |
 datname         | name                     |
 pid             | integer                  |
 usesysid        | oid                      |
 usename         | name                     |
 application_name | text                    |
 client_addr     | inet                     |
 client_hostname | text                     |
 client_port     | integer                  |
 backend_start   | timestamp with time zone |
 xact_start      | timestamp with time zone |
 query_start     | timestamp with time zone |
 state_change    | timestamp with time zone |
 wait_event_type | text                     |
 wait_event      | text                     |
 state           | text                     |
 backend_xid     | xid                      |
 backend_xmin    | xid                      |
 query           | text                     |
```

- pg_wait_sampling



Что будет в 10.0 ?

- BDR — двунаправленная репликация http://2ndquadrant.com/en/resources/bdr/
- Pglogical (5x быстрее slony, londiste3) http://2ndquadrant.com/en/resources/pglogical/
- Declarative partitioning (+pg_pathman)
- Highly Available multimaster «из коробки»
- Инкрементальный бэкап на уровне блоков
- FDW pushdown aggregates
- Миллисекундный полнотекстовый поиск
- In-memory хранилище
- Хранение временных объектов в памяти
  - временные таблицы на слейве

Postgres «наш» ?

Российские разработчики

Вадим Михеев, Красноярск (PostgreSQL CORE MEMBER)

PostgreSQL CORE

- MVCC
- WAL
- Subselects
- Vacuum
- Transactions Isolation
- Triggers

Российские разработчики

Олег Бартунов, Федор Сигаев, Александр Коротков

PostgreSQL CORE

- Locale support
- PostgreSQL extendability: GiST(KNN), GIN, SP-GiST
- Full Text Search (FTS)
- NoSQL (hstore, jsonb)
- Indexed regexp search
- Custom AM & Generic WAL
- VODKA access method (WIP)

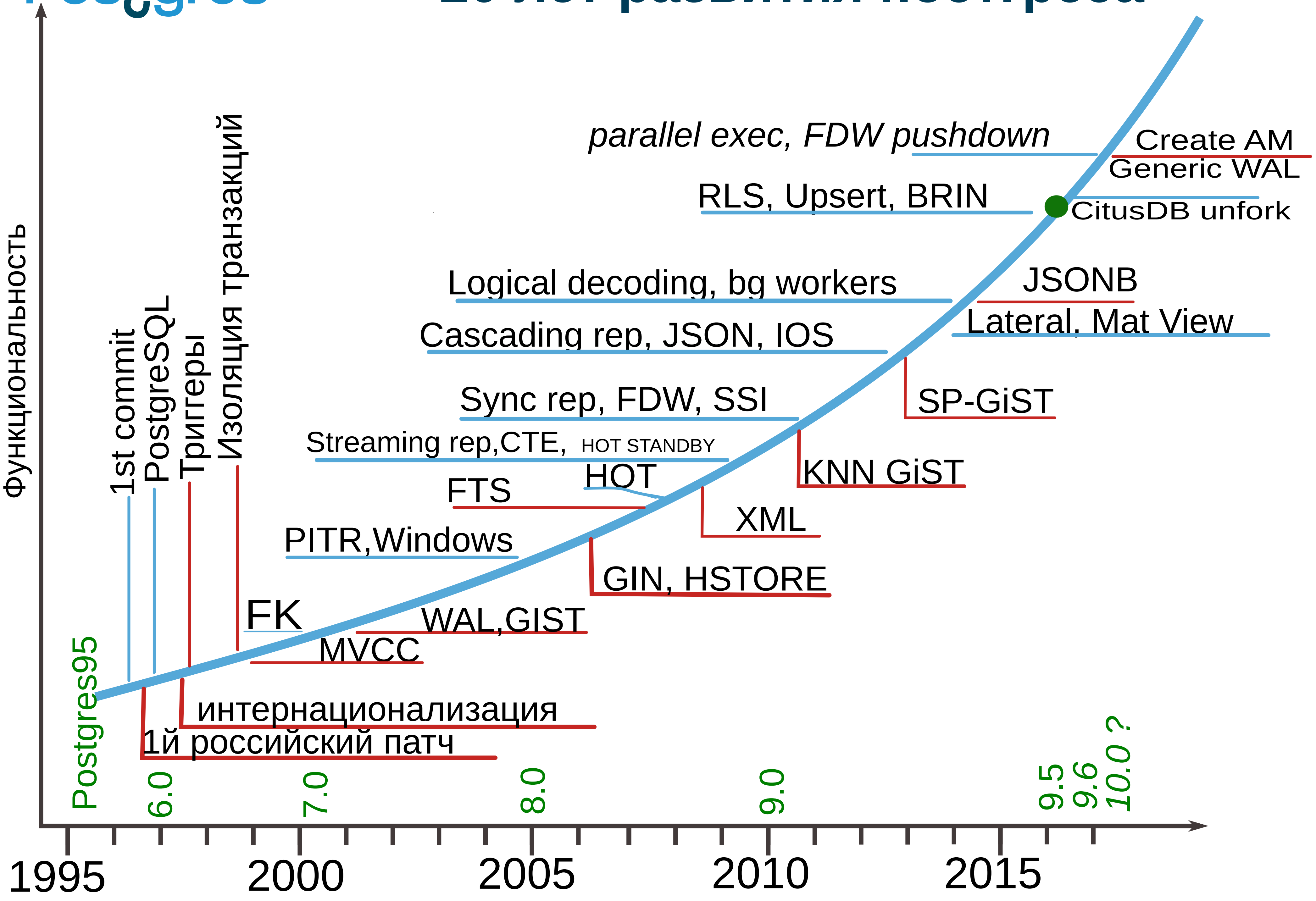
Расширения:
- Intarray
- Pg_trgm
- Ltree
- Hstore
- plantuner

- Докладчики PGCon, PGConf: 20+ докладов
- Менторы GSoC
- Коммитеры PostgreSQL (1+1 in progress)
- Организаторы конференций
- 50+ лет экспертизы PostgreSQL: разработка, аудит, консалтинг
- Novartis, Raining Data, Heroku, Engine Yard, WarGaming, Rambler, Avito, 1c

PostgreSQL Developers Meeting, 2015, Ottawa, Canada

Трое российских разработчиков участвуют в совещании разработчиков

20 лет развития постгреса



Создание компании Postgres Professional

Январь-апрель 2015 г.

Раньше:
Postgres использовали
на свой страх, риск,
и в удовольствие.

Теперь:
Есть российская компания — вендор,
обладающая компетенцией разработчиков.

Postgres Professional

Российский вендор PostgreSQL в России

- Поддержка, разработка, консалтинг, обучение
- Тему технологической независимости СУБД пропагандируем с 2011 г.
- Члены международного сообщества
- Участники и спонсоры международных конференций (Канада, Австрия, Бразилия, Китай, Беларусь)
- PgConf.Russia 2015 и 2016 - крупнейшие в мире конференция по PostgreSQL

В направлениях, где мы ведем разработку, PostgreSQL является лидером* среди РСУБД

- геоинформационные системы, слабоструктурированные данные, полнотекстовый поиск, расширяемость

Все российские ключевые международно признанные разработчики PostgreSQL работают в нашей компании

В нашей команде 4 кандидата наук: 3 – по PostgreSQL и технологиям БД.

Сотрудничаем с МГУ и СПбГУ.

Postgres PROFESSIONAL

Postgres Professional NY 2016

Импортозамещение

~~Импортозамещение~~

Технологическая независимость

Технологическая независимость

Мировая тенденция к **открытым** решениям

Пирамида «импортозамещения»

PROFESSIONAL Postgres

Год жизни компании

2015

Февраль. Регистрация компании

Апрель. Начало работы

Июнь. **Первое место** среди проектов по импортозамещению СУБД

Первая крупная разработка — мониторинг ожиданий

Июль. Сборки под Windows и 1С

Июль. Готов курс DBA1.

Октябрь. Достижение 2 млн транзакций в секунду на Power 8

2016

Январь. Готов курс DBA2

Выпуск **PostgresPro** 9.5.0.1

Февраль. PgConf.Russia 2016

Март. Выпуск PostgresPro 9.5.1.2

Вхождение в Единый реестр российского ПО

Апрель. Выпуск PostgresPro 9.5.2.1

7 докладов на международных конференциях (PGCon, PGConf)

20 докладов на российских Конференциях

65 патчей в ядро PostgreSQL 9.6

Российская СУБД Postgres Pro

Postgres Pro = PostgreSQL + …

1) Более ранний доступ к новым разработкам (в первую очередь, российским)
2) Поддержка со стороны российской компании-разработчика
3) Присутствие в Едином Реестре Российского ПО

Открытая версия

- Open Source
- BSD-подобная лицензия

Закрытая версия

- Соответствие российским требованиям ИБ
- Сертификация ФСТЭК (5 НСД, 4 НДВ) — в процессе
- Более высокие уровни — в планах

Что такое Postgres Pro ?

1. Российский форк PostgreSQL, вошедший в Реестр Российского ПО
2. Наши разработки ядра PostgreSQL, которые ещё не успели войти в релиз, но закоммичены в апстрим
3. Бэкпорты из 9.6, которые мы считаем полезными
4. Полезные расширения (не только наши)
5. Возможность оперативно реагировать на запросы клиентов

Postgres Pro предоставляет доступ к новой функциональности и улучшениям раньше, и позволяет быстрее реагировать на запросы клиентов

Что еще есть в Postgres Pro ?

- Увеличение производительности на многоядерных системах:
- Улучшение полнотекстового поиска:

  поиск фраз, словарь=extension,

  словари в shared memory
- Покрывающие индексы. Поддержка конструкции INCLUDES в CREATE INDEX. https://pgconf.ru/2016/89847
- Переносимость: ICU
- Модуль pg_trgm: нечеткий поиск подстроки
- Модуль pageinspect: доступ к внутреннему представлению данных
- Модуль sr_plan: сохранение планов запросов
- Модуль dump_stat: дамп и восстановление статистики
- Модуль JSQuery
- Libpq с поддержкой failover

Скоро в Postgres Pro

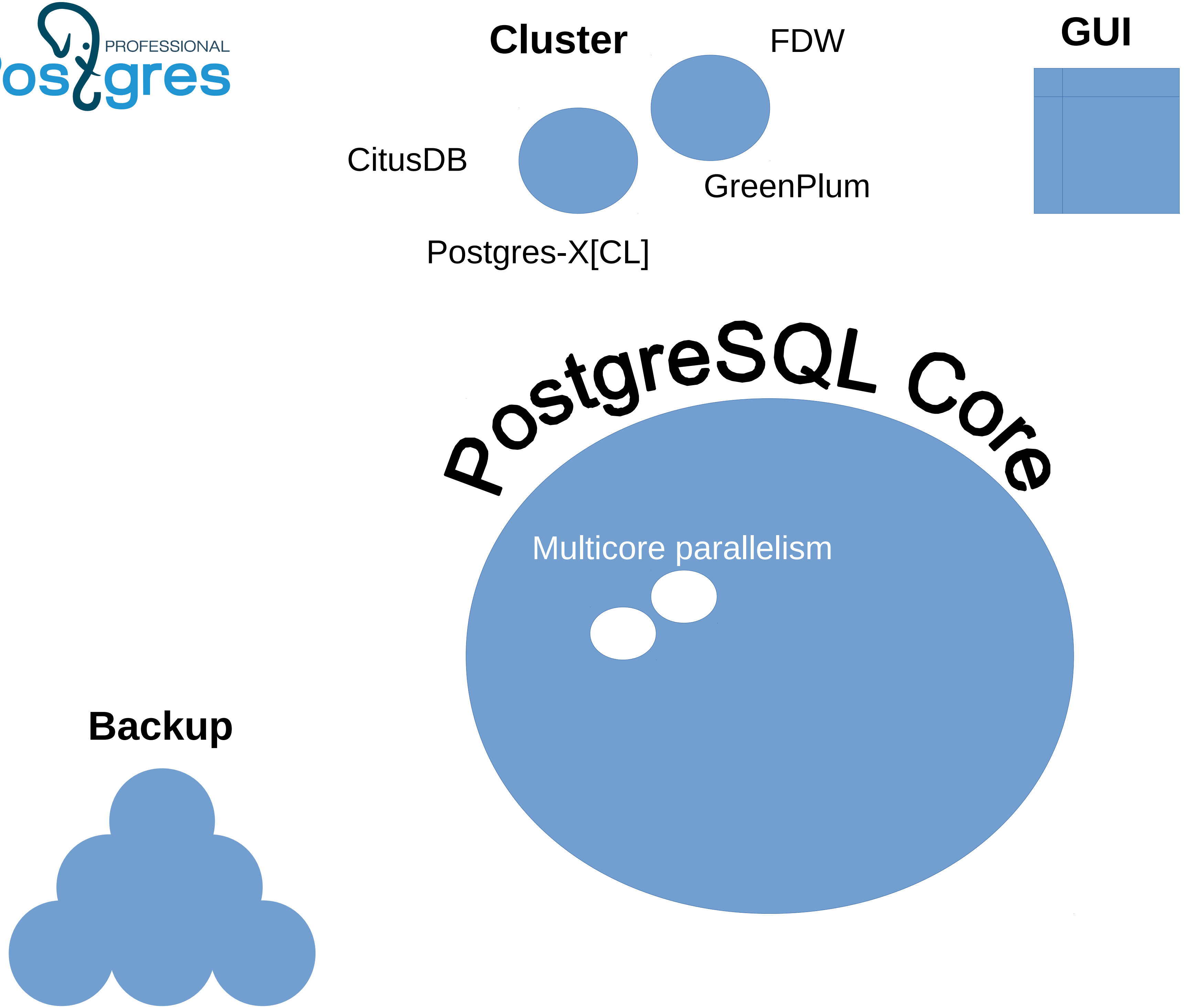
- Секционирование таблиц без накладных расходов
- Инкрементальный бэкап на уровне блоков
- Компрессия B-Tree индексов
- Миллисекундный полнотекстовый поиск (на основе CREATE ACCESS METHOD и GENERIC WAL)

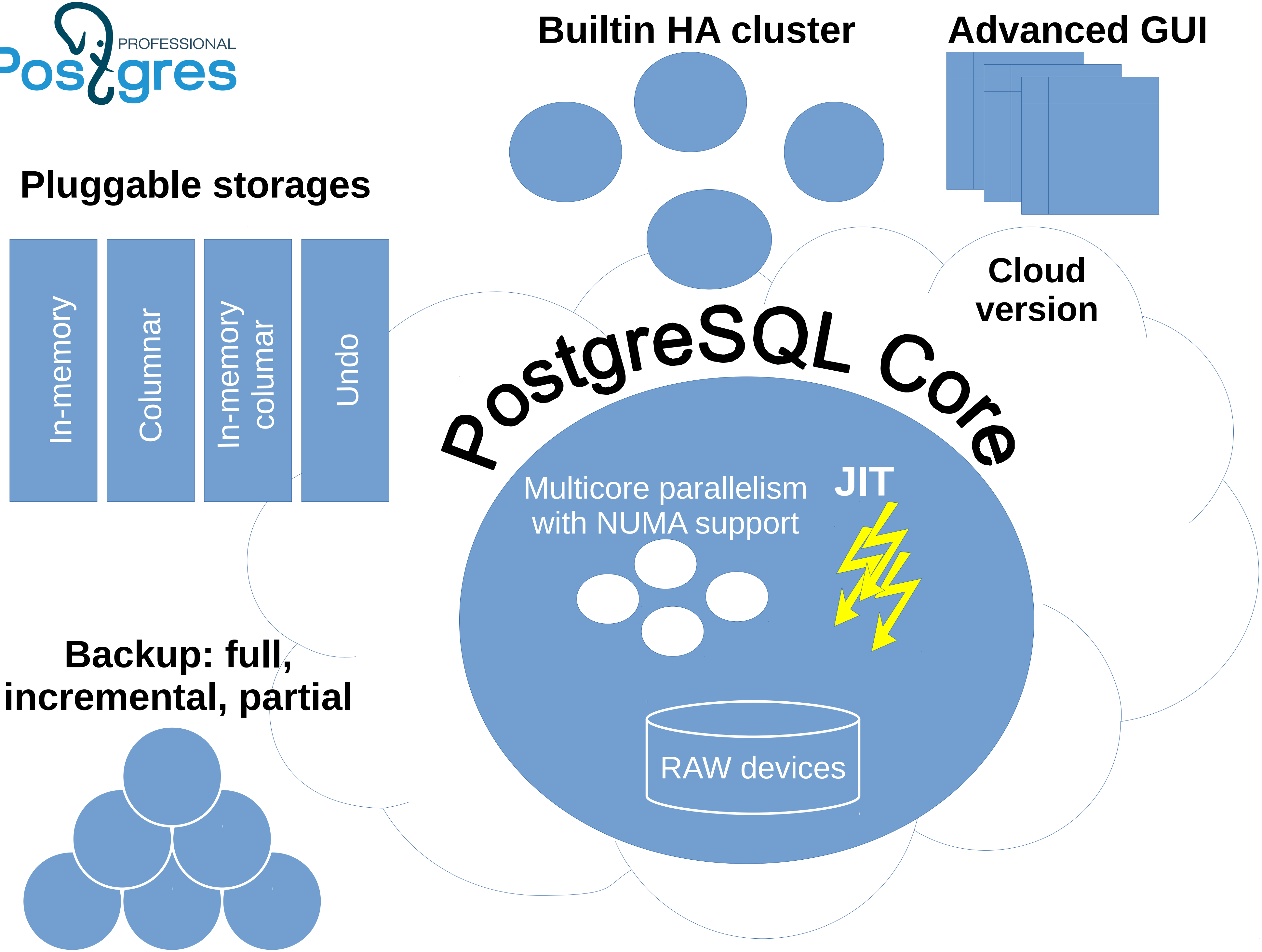
https://wiki.postgresql.org/images/2/25/Full-text_search_in_PostgreSQL_in_milliseconds-extended-version.pdf

- Мониторинг ожиданий

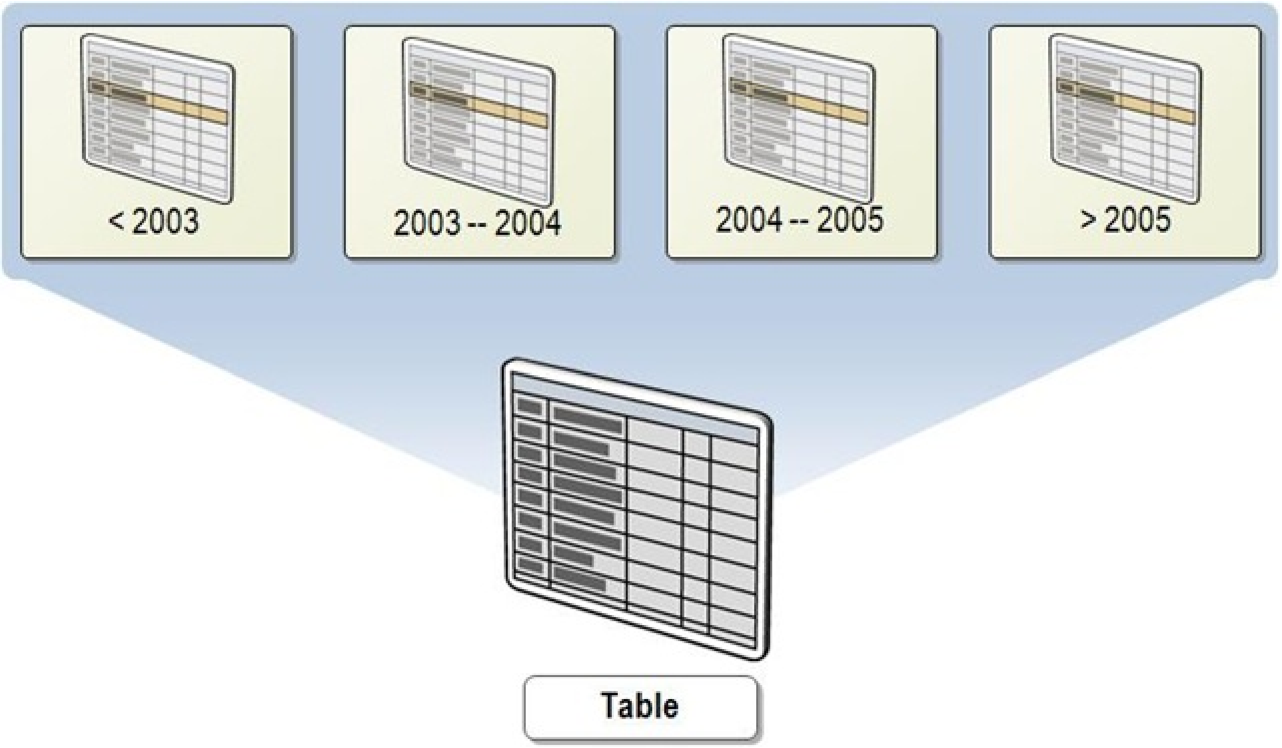
http://akorotkov.github.io/blog/2016/08/26/wait_monitoring_9_6/

http://www.postgresql.org/docs/devel/static/monitoring-stats.html#WAIT-EVENT-TABLE


Cluster
FDW
GUI
CitusDB
GreenPlum
Postgres-X[CL]
PostgreSQL Core
Multicore parallelism
Backup


Builtin HA cluster
Advanced GUI
Pluggable storages
In-memory
Columnar
In-memory columar
Undo
Cloud version
PostgreSQL Core
Multicore parallelism with NUMA support
JIT
RAW devices
Backup: full, incremental, partial

Секционирование



Секционирование

До сих пор:

- Делается через наследование таблиц (мега-костыль ?)
- Всегда сканируется родительская таблица (зачем?)
- Планирование медленное, т. к. механизм constraint_exclusion пытается работать в общем виде
- Не поддерживается HASH-секционирование
- Нет выбора секций на этапе выполнения (очень медленное выполнение, когда секции не известны на этапе планирования: nested loop joins, subqueries)

Уже сделано в Postgres Pro (расширение pg_pathman):

- HASH и RANGE partitioning
- Хранение метаданных секций с бинарным поиском для RANGE partitioning'а при планировании запроса
- Упрощение условий фильтрации, которые попадают в секции
- Select, Insert, Update, Delete
- Нода RuntimeAppend, которая выбирает партиции на этапе выполнения.

Секционирование (планы)

pg_pathman

- Стабилизация работы
- Оптимизация Hash join при секционировании по ключу JOIN
- LIST-partitioning
- Subpartitioning
- Online (re)partitioning

PostgreSQL 9.7 core:

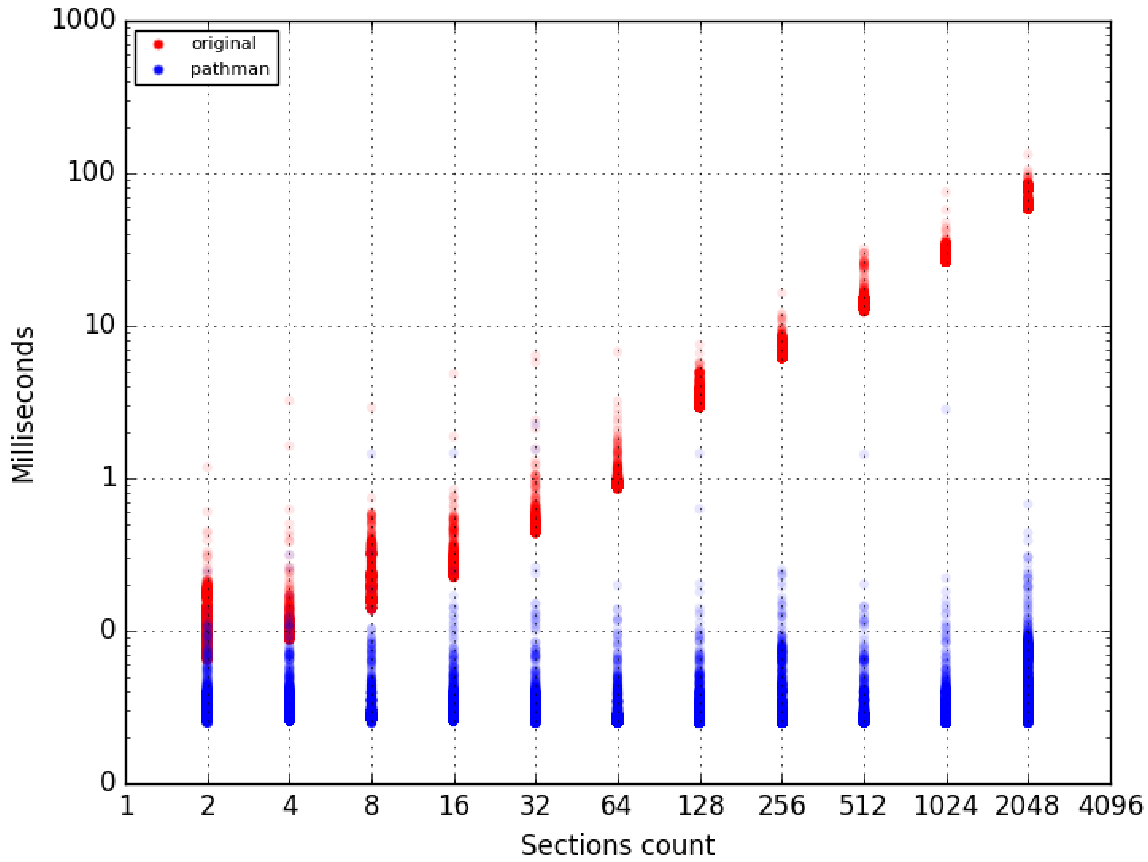
- Декларативный синтаксис (совместно с NTT)
- Поэтапное портирование фич pg_pathman

Секционирование

Время планирования при попадании в план одной секции.

Обычная реализация на основе наследования таблиц страдает большими накладными расходами.

pg_pathman свободен от них.

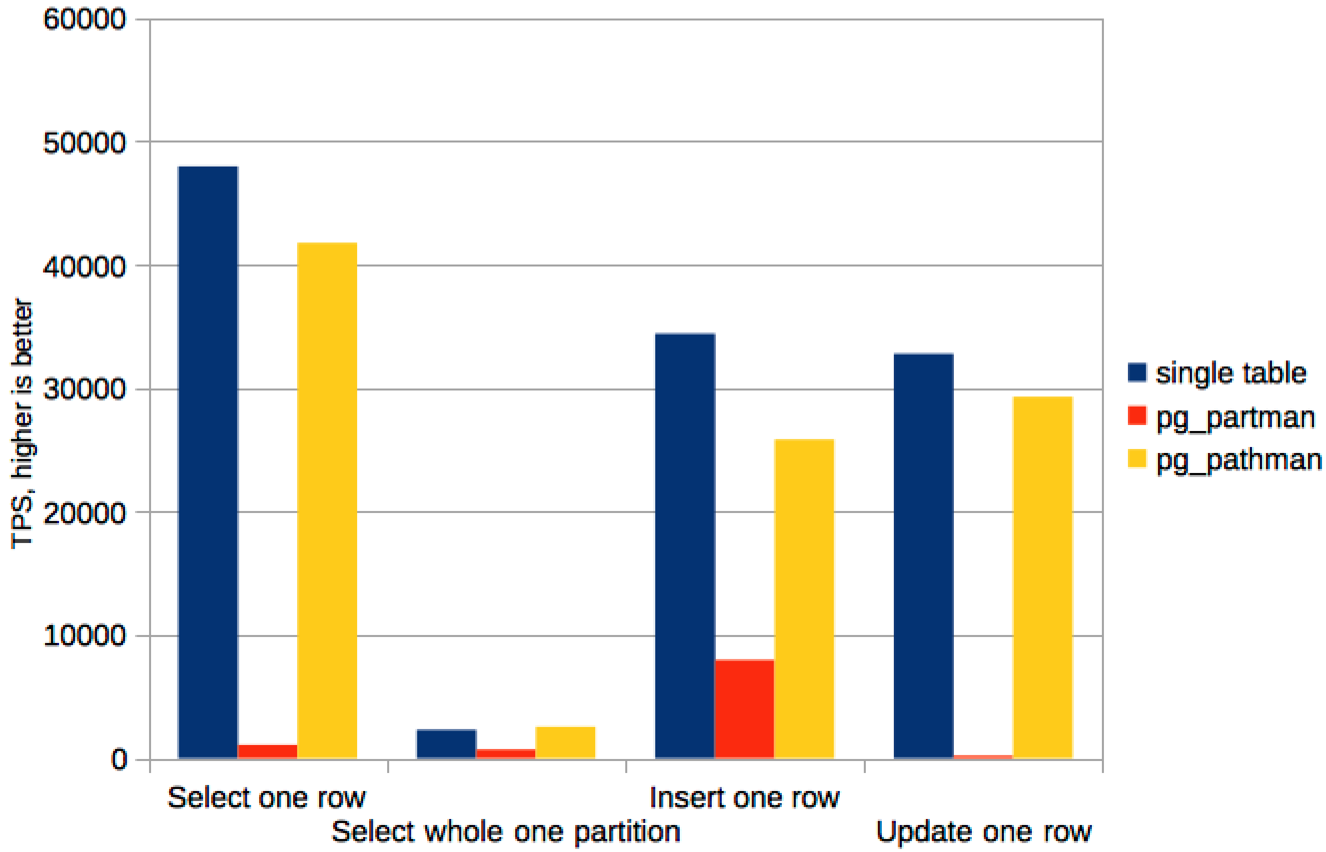


Секционирование

Производительность
SELECT, INSERT, UPDATE

pg_partman — наследование таблиц

pg_pathman — разработка Postgres Pro, использование хуков планировщика



Подробности:

https://pgconf.ru/2016/89633

http://akorotkov.github.io/blog/categories/partitioning/

http://postgrespro.ru/blog/pgsql/pg_pathman

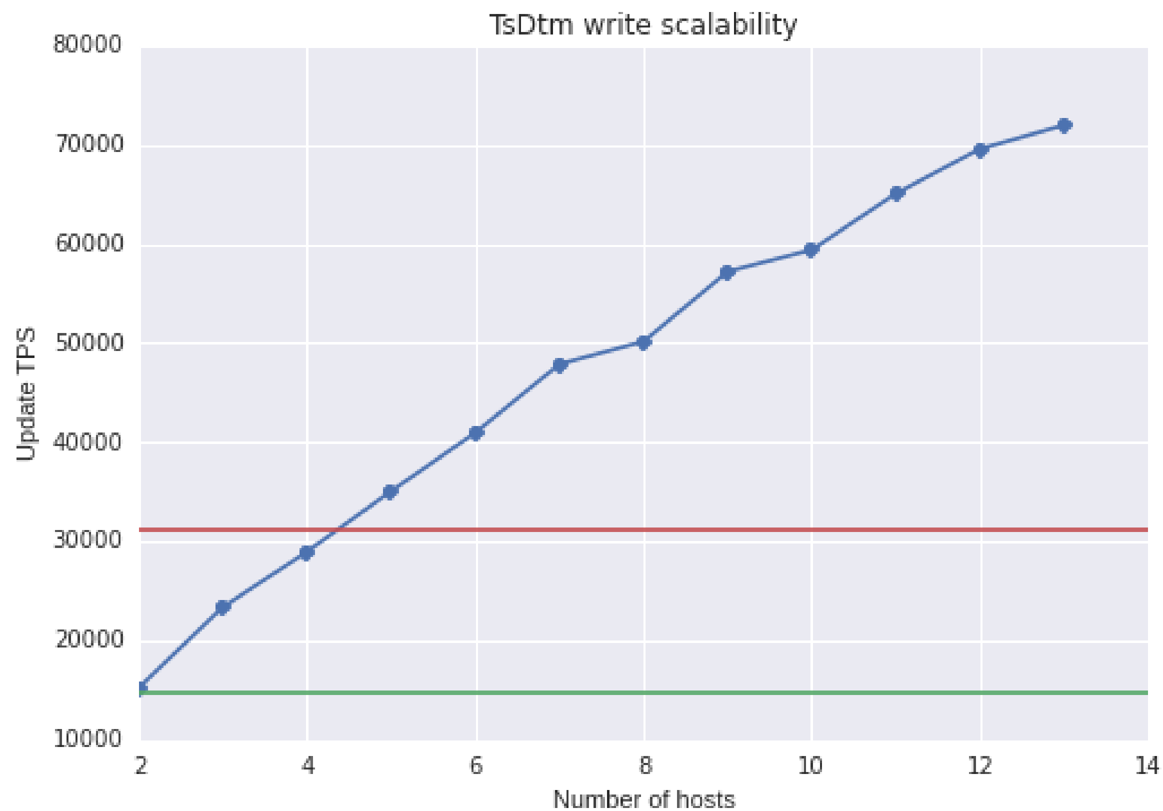
Кластер

Уже сделано:

- Менеджер распределённых транзакций с atomic commit и atomic visibility
- Multimaster на базе logical decoding
- Node management

TODO:

- Стабилизация работы
- Sharding



Backup

Уже сделано:

- Битовая маска изменённых блоков
- Использование битовой маски в pg_arman для инкрементального бэкапа

TODO:

- Тестирование и стабилизация работы
- Частичный backup (отдельная БД, схема, таблица)

Управление планами

Уже сделано:

- Прибивание плана запроса «гвоздями» (сериализация/десериализация): sr_plan

TODO:

- Перенос планов между базами данных
- Перенос планов между major версиями
- Гибкое управление планами (сохранение хинтов)

Pluggable storages

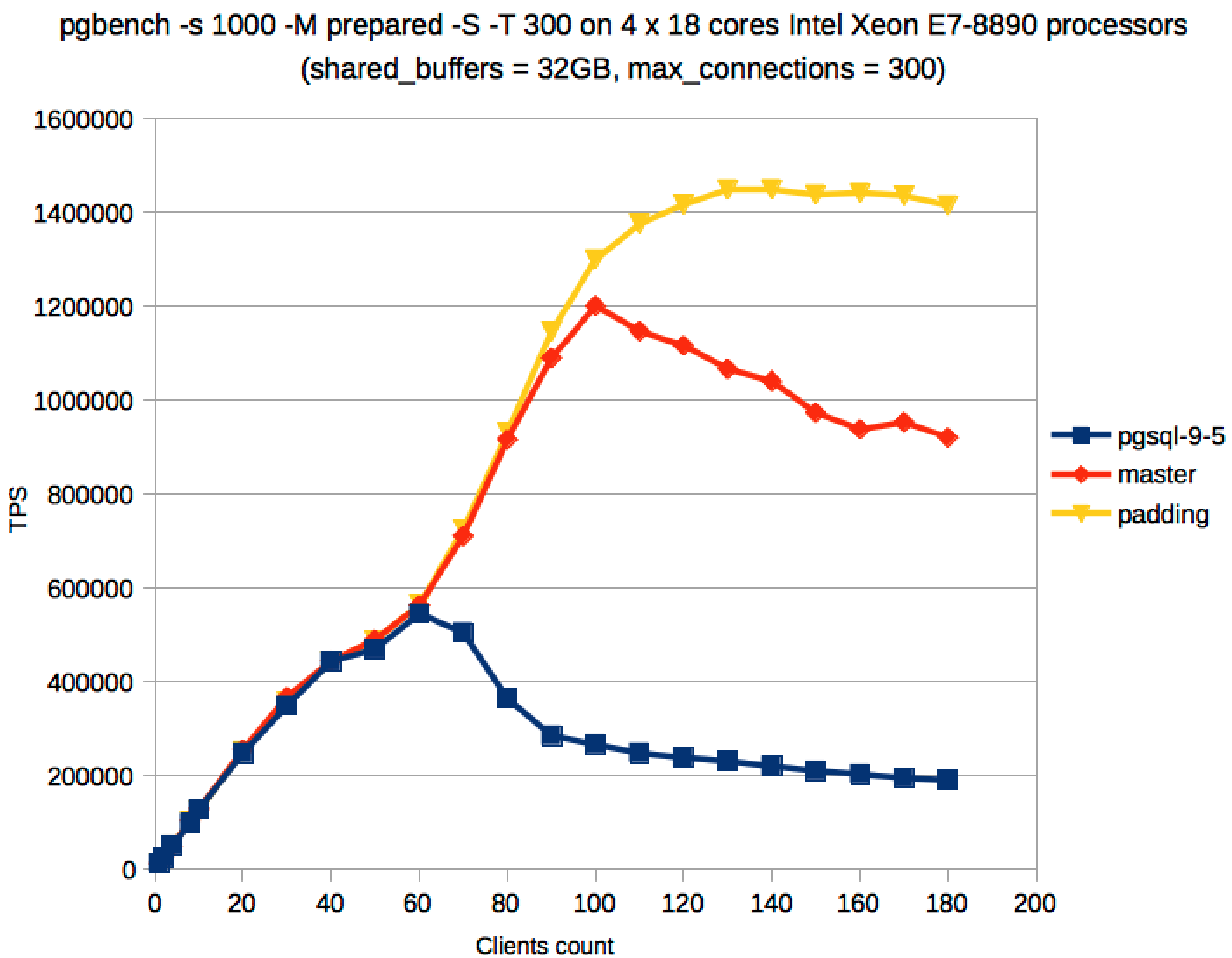
- In-memory persistent row-oriented storage
  - Базовая версия
  - Поддержка транзакций
  - Поддержка persistency
- Columnar storage
- In-memory columnar storage
- On-disk storage with undo log

JIT-компиляция запросов

- План запроса – это дерево, которое интерпретируется (очень высокая степень indirection)
- Когда запрос упирается в CPU (все данные в памяти или дисковая подсистема мощная), то можно многое сэкономить, за счёт JIT-компиляции.
- Актуально для OLAP, in-memory storage и пр.

Multicore scalability

Уже сделано:



TODO:

- NUMA support

Better indexing

Btree

- Covering indexes (done)
- Duplicate compression – размер btree примерно такой же, как и у GIN (in-progress)
- KNN support
- Full index scan

VODKA

- Container for nesting other index access methods
- Container for complex graphs on other index access methods

Образовательная деятельность

DBA1

Июль,
Август Москва
12-14.10 Барнаул
10-12.11 Ульяновск
2-4.12 Омск
16-18.12 Тверь
23-25.12 Улан Удэ
12-14.03.2016 Казань,
28-30.03 Новосибирск

Тюмень
Санкт-Петербург
Челябинск

УЦ «Форс»

DBA2

9,16.02 Яндекс
25.02 ГАИШ

Hacking PostgreSQL

С 25.02.2016 Яндекс

DEV1, DEV2, сертификация ….

Технологии СУБД

2016-2017 уч. Год СПбГУ
Заинтересовались: МГУ, Киров, Омск, Челябинск, Барнаул, ВШЭ, МЭСИ, УрФУ, Красноярск

https://www.postgrespro.ru/education

План разработок Postgres Pro :2019

- HA Мультимастер
- Доработки безопасности для сертификации на высокие уровни секретности
- Инкрементальный и частичный бэкап
- Секционирование (partitioning)
- Управление планами запросов
- Планировщик заданий
- Расширяемые индексные методы доступа
- Эффективное хранение временных объектов
- Компрессия данных
- Сбор детальной статистики и метрик мониторинга
- Средства упрощения миграции с Oracle
- Средства управления СУБД
- CREATE INDEX USING RUM (GIN2)
- Advanced multicore parallelism
- Incremental materialized views
- Облачная версия СУБД
- Запись и воспроизведение нагрузки
- Удобный и расширяемый синтаксис для поиска и обновления JSON
- Эффективная работа с массивами, интервалами и иерархическими данными
- Распараллеливание запросов в мультимастере
- Улучшения планировщика (корреляции, планирование на этапе выполнения)
- Улучшения сборки и тестирования (CMake, unit-тесты)

План разработок Postgres Pro :2019

- Advanced FTS (быстрый поиск, обратная задача, конф. парсер, TF/IDF)
- Индексы с быстрой вставкой и обновлением
- Адаптивное построение запросов с помощью машинного обучения
- JIT компиляция
- Sharding
- Эффективная работа с неточными, битемпоральными, сферическими, timeseries данными
- Индесная поддержка для расширяемого синтаксиса JSON
- Эффективная поддержка NUMA
- Распределённая система тестирования
- Альтернативные табличные движки: columnar, in-memory
- Сжатие передаваемых данных
- Оптимизация работы с диском (double buffering, SSD)
- Система тестирования производительности (TPC-* из коробки)
- CREATE INDEX USING VODKA
- In-memory columnar storage
- Sharding с автоматическим распределением по узлам
- Мониторинг кластера
- Асинхронный протокол взаимодействия
- Встроенный балансировщик нагрузки и пул коннектов
- Поддержка тета-соединений
- Эффективная встроенная очередь
- Оптимизация работы с SSD

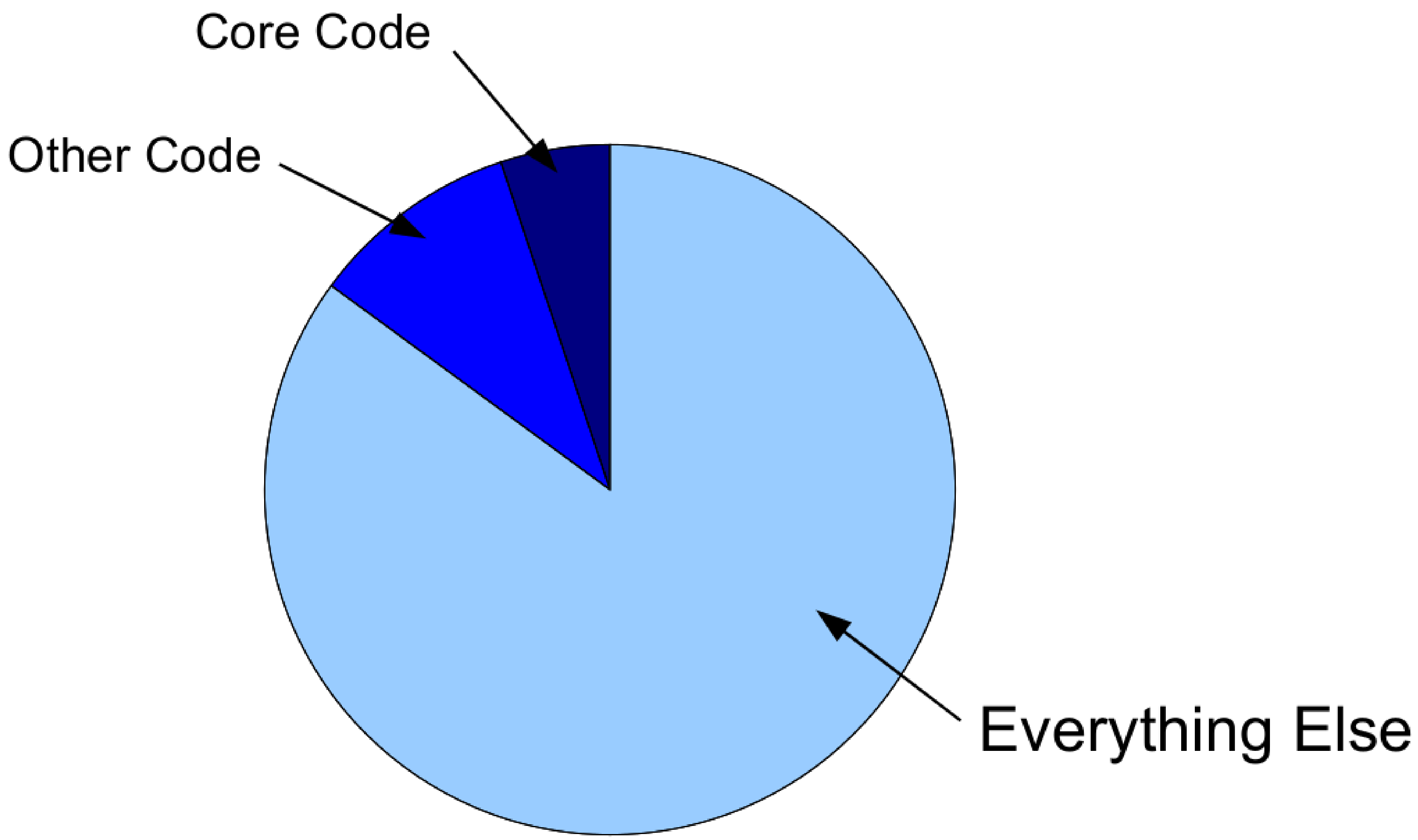
Мы ищем таланты

- Разработчики, инженеры, QA, PM, технические писатели, стажеры (студенты), люди науки
  - Работать в команде, «жить» в сообществе
  - Уметь и любить учиться
  - Любить вызовы
  - Держать цель
- Возможна удаленная работа

Российское сообщество

- Самое организованное — несколько тысяч человек
- Митапы при поддержки крупных компаний
- Крупнейшие в мире конференции по постгресу:
  - летом PGDay.ru в Санкт-Петербурге (2014, 2015, 2016)
  - Зимой PGConf.ru в Москве (2015, 2016)
- Секции и квартирники на крупнейших конференциях
  - Highload++, RIT, Codefest, Stachka
- Участвуем в международных конференциях
  - PGConf.EU, PGCon.org
- Свободные курсы DBA1, DBA2, «Hacking Postgres» от Postgres Professional

50 способов помочь сообществу



50 способов помочь сообществу

Ядро
Разработка, review, тестирование, reporting bugs

Экосистема
Расширения, драйверы, ORM, средства мониторинга… поддержка Pg в прикладном ПО
Создание дистрибутивов, пакетирование

Документация
Улучшение, перевод, публикация статей, книг, учебных, маркетинговых материалов…блоггинг!

Расскажите о своей истории с PostgreSQL!

Общение, образование
Создание локальных сообществ
Проведение конференций, митапов, семинаров, учебных курсов.

Внедрите PostgreSQL!
В Вашей компании. Запустите учебный курс в Вашем ВУЗе

Спонсорство
Спонсируйте разработку нужной Вам функциональности.
Спонсируйте мероприятие.

Если вы с Open Source

Свобода — необходимое условие для творчества.
Идеи рождаются в свободном творчестве.

Главные свойства

- Возможность бесплатного тиражирования
- Доступ к исходным кодам

Основные свободы по FSF/GNU:

- Выполнять программу
- Изучать и модифицировать программу
- Передавать копии программы
- Передавать копии модифицированной программы

«Безрассудная» свобода MIT/BSD:

- Создавать закрытую программу на основе открытой

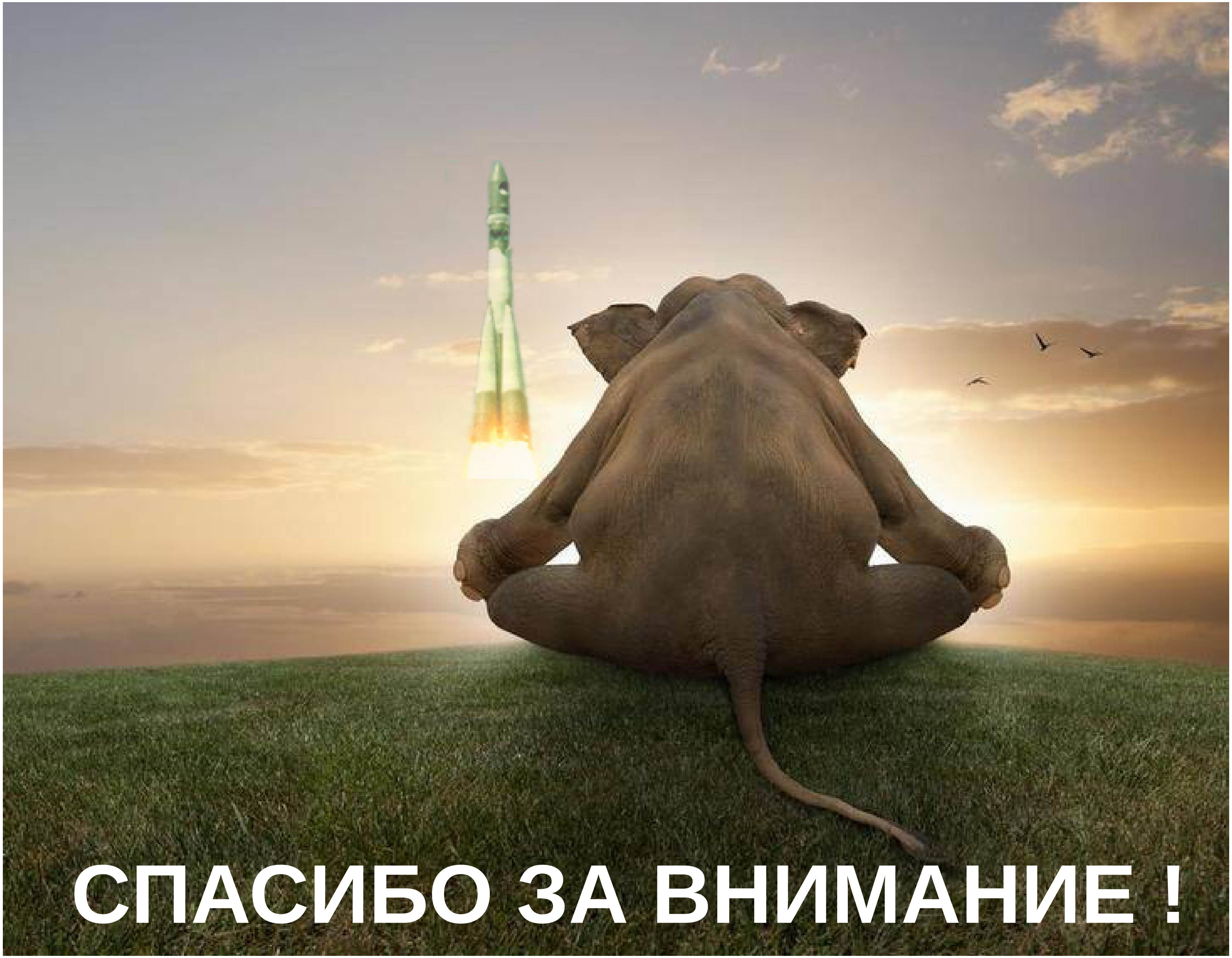
Если вы захотели стать разработчиком OPEN SOURCE проекта

Что дает участие в Open Source

- Причастность к большому проекту, большому сообществу
- Реализация как разработчика
- Влияние на развитие проекта
- Независимость от компании, репутация в сообществе
- Карьера в сообществе коррелирует с карьерой в компании
- Возможность жить и работать в удобном месте — дОма (no Piter, no Moscow) !
- Удовлетворение — help the World !

Требования к разработчику

- Знание и владение основными инструментариями
  - Язык[и] программирования
  - Git, треккеры, вики, средства документирования
- Совместимость с сообществом
  - Знание английского языка (разные)
  - Умение вести переписку
  - Не пропадать надолго
  - Следовать стилю кодирования
  - Синхронизоваться с циклом разработки
  - Следовать принятым сценариям разработки
  - Принимать участие в жизни сообщества


СПАСИБО ЗА ВНИМАНИЕ !