



Что нового в PostgreSQL 8.3?

Николай Самохвалов,
«Постгресмен»

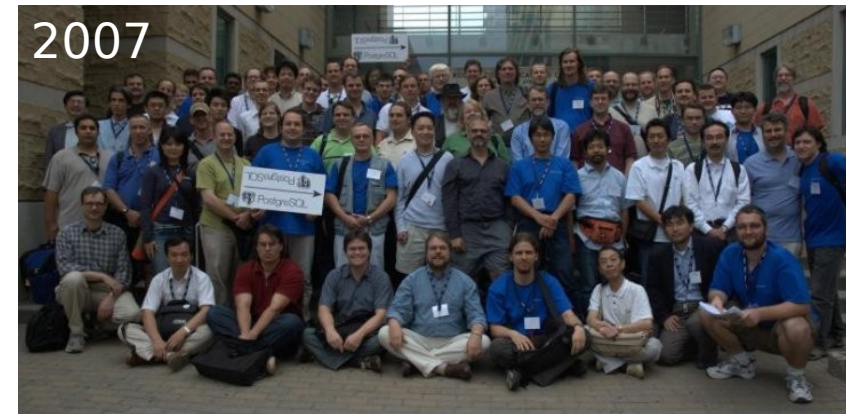
октябрь 2007

PostgreSQL – это...

- ...самая развитая СУБД с открытым кодом
- ...надежность и устойчивость на любых нагрузках
- ...кроссплатформенность (FreeBSD, Linux, Solaris, Windows, OS X)
- ...высокий уровень соответствия ISO/ANSI SQL 92, 99, 2003
- ...интерфейсы для Tcl, Perl, C, C++, PHP, ODBC, JDBC, Embedded SQL in C, Python, Ruby, ...
- ...развитые административные утилиты (pgAdmin, phpPgAdmin, ...)

11 лет развития PostgreSQL

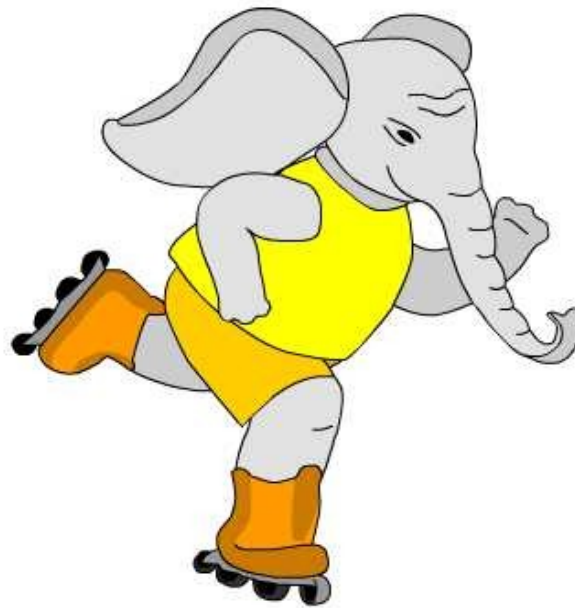
- ...1995 – «прошлая жизнь» (Postgres)
- 1996...1998 – стабильность
- 1998...2001 – стандарт SQL
- 2001...н. в. – выход на уровень Enterprise
 - производительность
 - простота администрирования
 - 24/7



Новое в PostgreSQL 8.3

- Производительность
- Разработчикам баз данных
- Администраторам баз данных
- Проекты сторонних разработчиков

Производительность



The 1st PostgreSQL Enterprise-level Benchmark (SPEC*)

Josh Berkus: «...a good day for Open Source»

PostgreSQL 8.2 – 813.73 JOBS

- SPECjAppServer2004 2x Sun Fire X4200 appservers (8 cores, 4 chips) and 1 Sun Fire T2000 DB server (8 cores, 1 chips) with PostgreSQL 8.2.4
- HW: ~\$60,000; SW: \$0

Oracle 10g – 874.17 JOBS

- SPECjAppServer2004 1 HP rx2660 appserver (4 cores, 2 chips) and 1 rx2660 DB server (4 cores, 2 chips) with Oracle Database 10g Enterprise Edition Release 10.2.0.2
- HW: ~\$74,000; SW: ~\$110,000

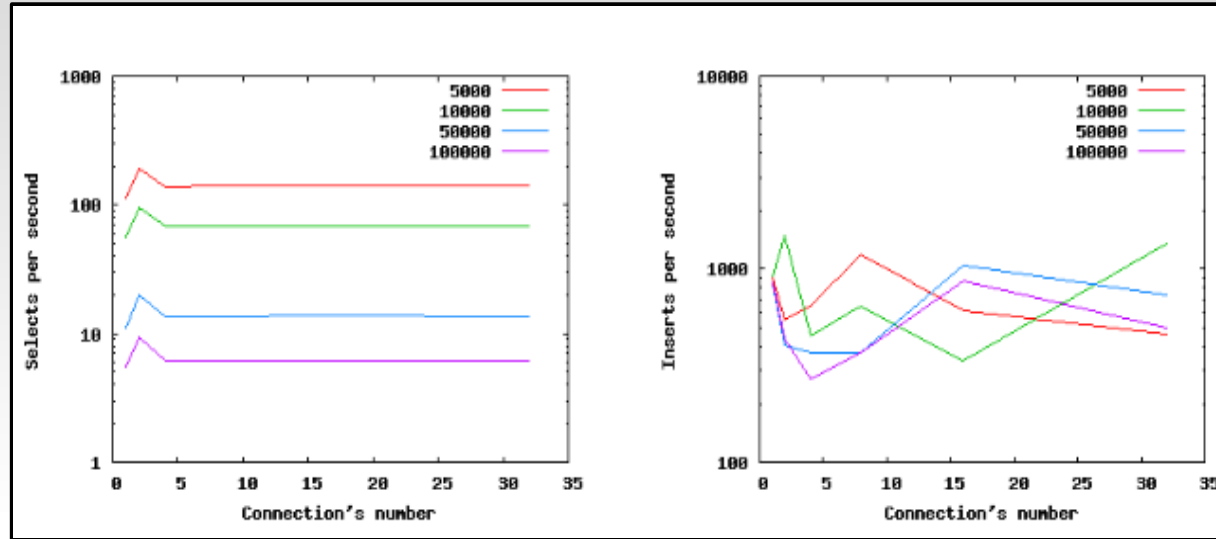
MySQL 5.0 – 720.56 JOBS

- SPECjAppServer2004 3x Sun Fire X4100 appservers (12 cores, 6 chips) and 1 Sun Fire X4100 DB server (4 cores, 2 chips) with MySQL 5.0.27 64-bit

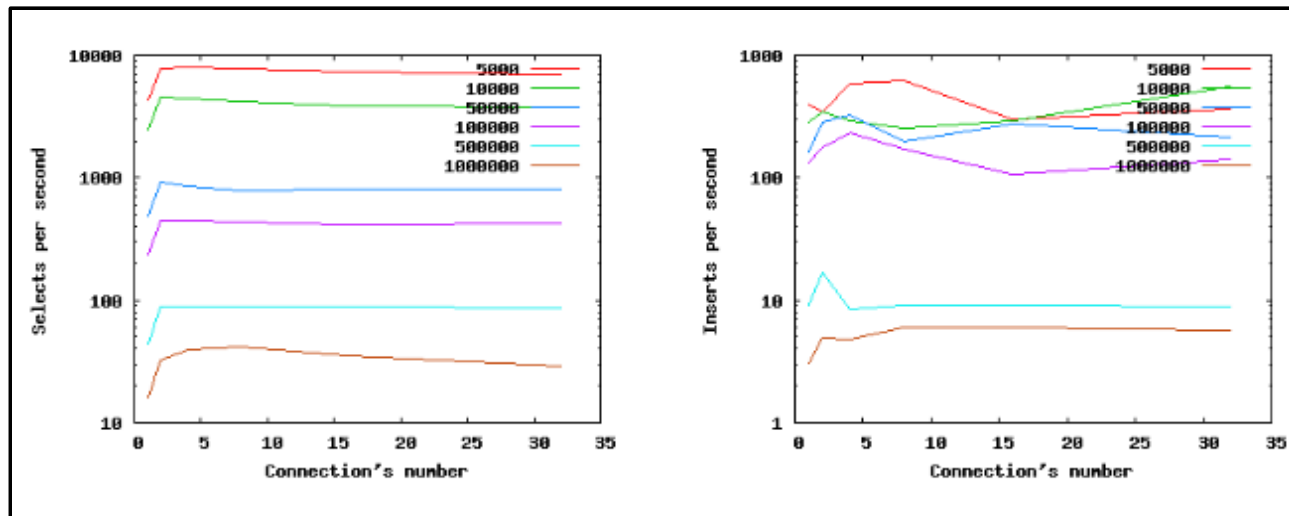
*) SPEC – Standard Performance Evaluation Corporation, <http://spec.org>

Индексы: GiST и GIN

GiST



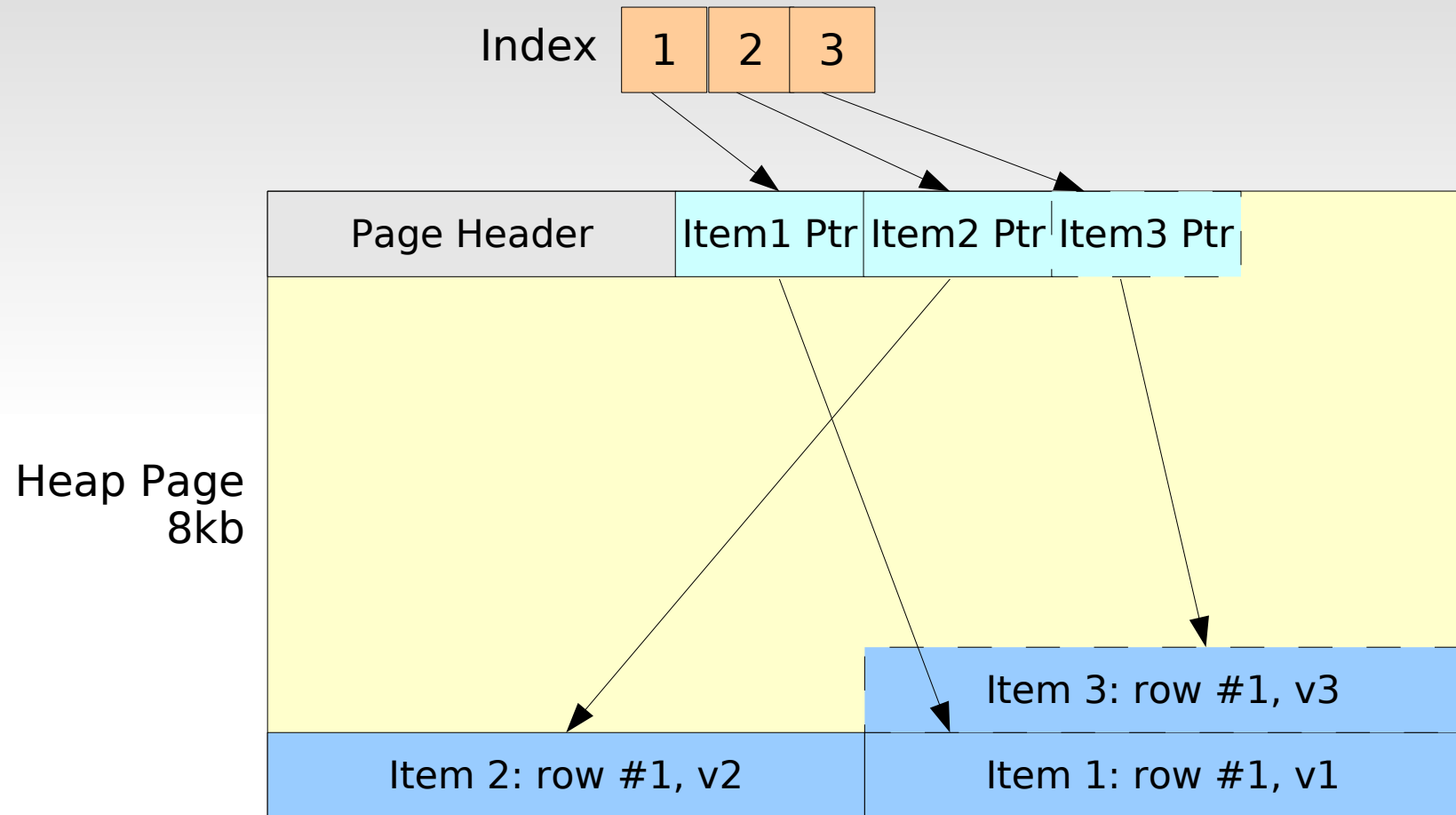
GIN



HOT (Heap Only Tuples)

- При обновлении (UPDATE) индекс **не** меняется, если:
 - UPDATE не затрагивает проиндексированные столбцы,
 - новая версия строки записывается на той же странице данных (heap).
- «Моментальное» повторное использование места в heap без ожидания процесса VACUUM.

Обновление без HOT



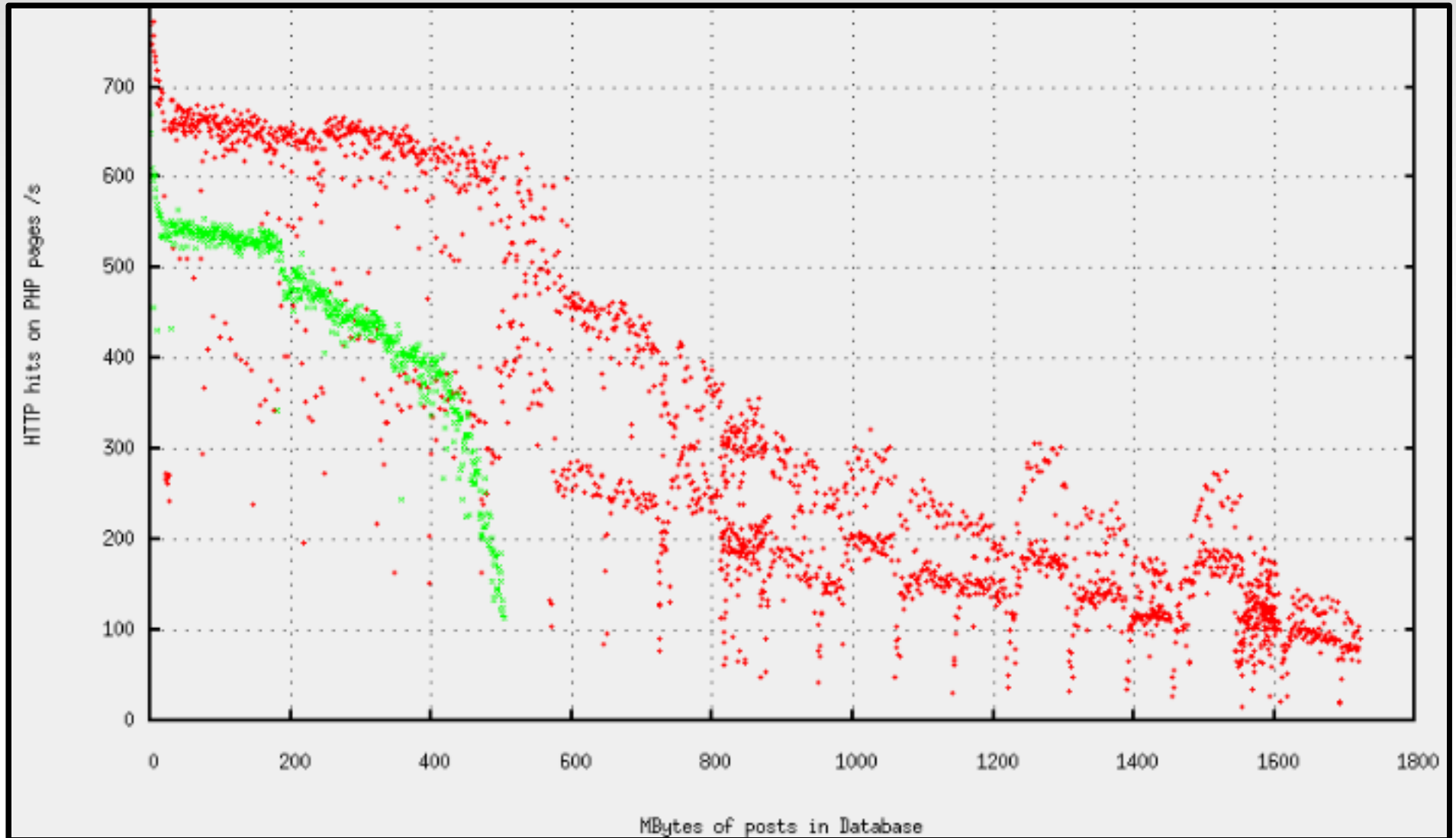
Asynchronous Commit

```
SET synchronous_commit TO OFF;
```

Позволяет отложить fsync на небольшое время (COMMIT без ожидания fsync)

- возможна потеря небольшой порции данных (порядка 0.2...1.0 сек)
- нарушение целостности данных невозможно! (в отличие от fsync=off)

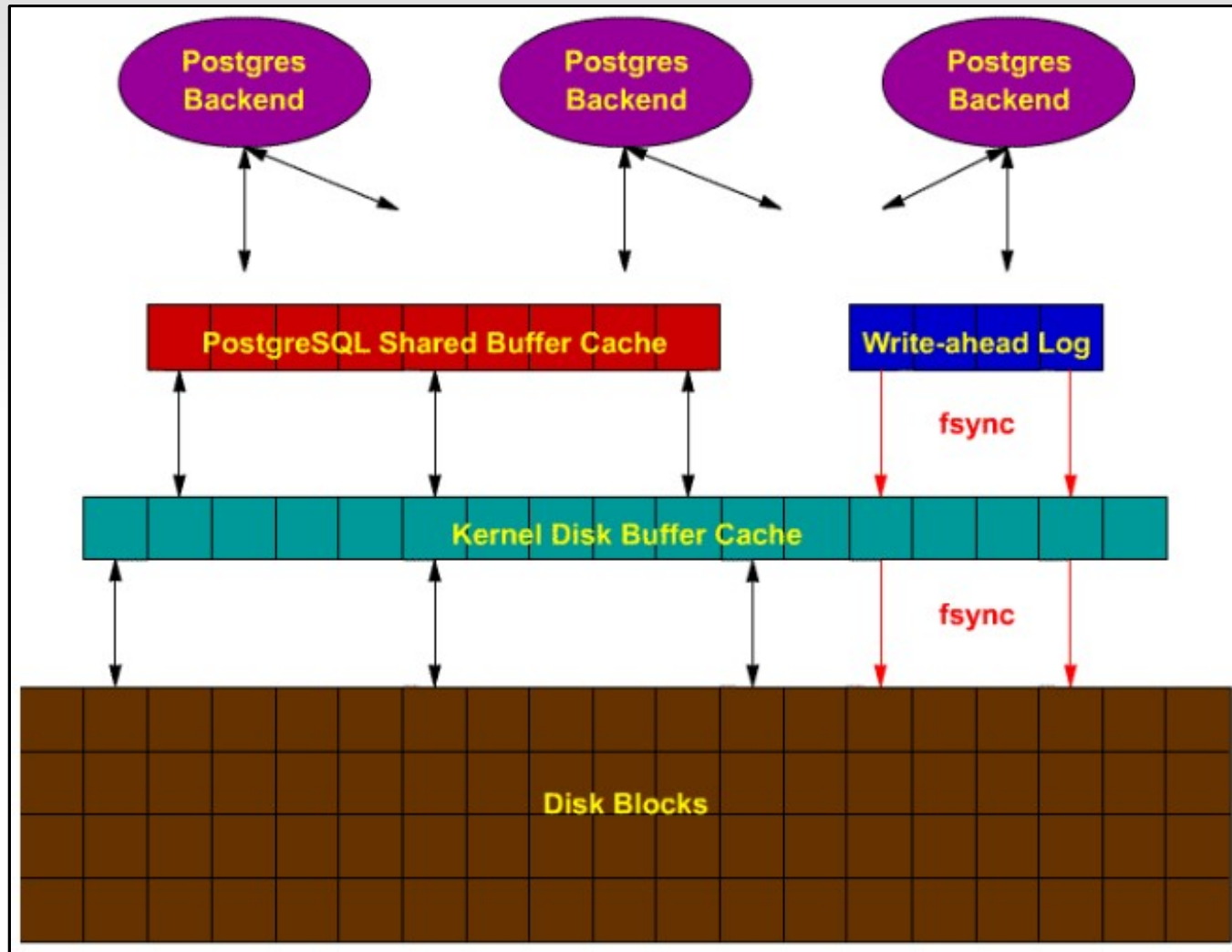
Надёжность производительности



Сглаживание контрольной точки (checkpoint smoothing)

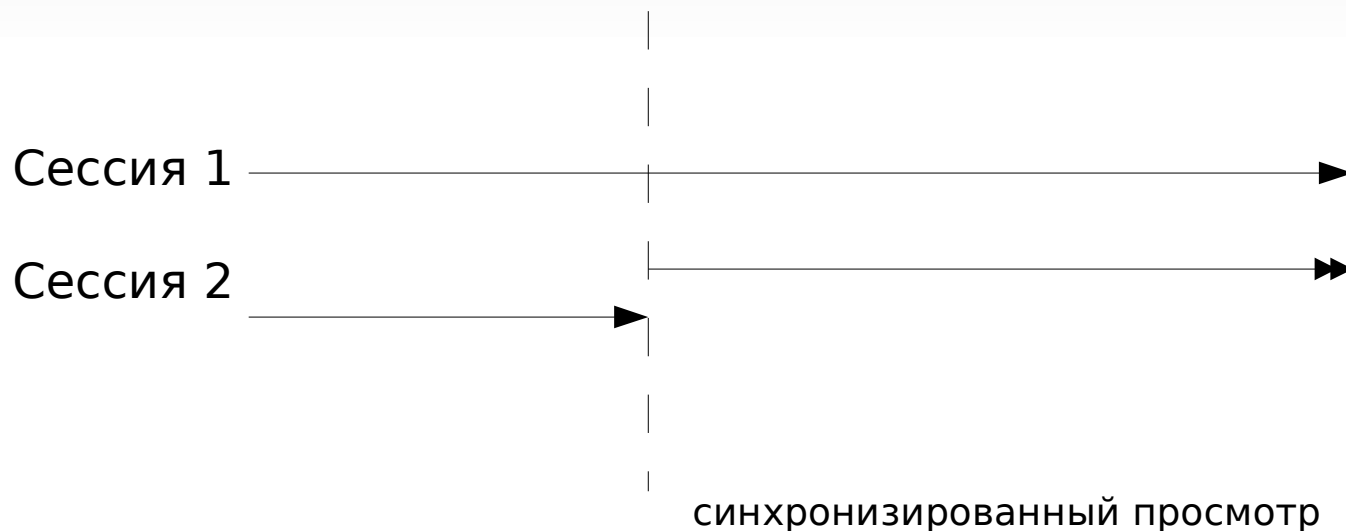
- «Размазывание» процесса checkpoint во времени, уменьшение интенсивности I/O-операций.
- Работает только для background checkpoints (при выключении сервера – полная скорость I/O-операций).

Устройство PostgreSQL



Synchronized Scans

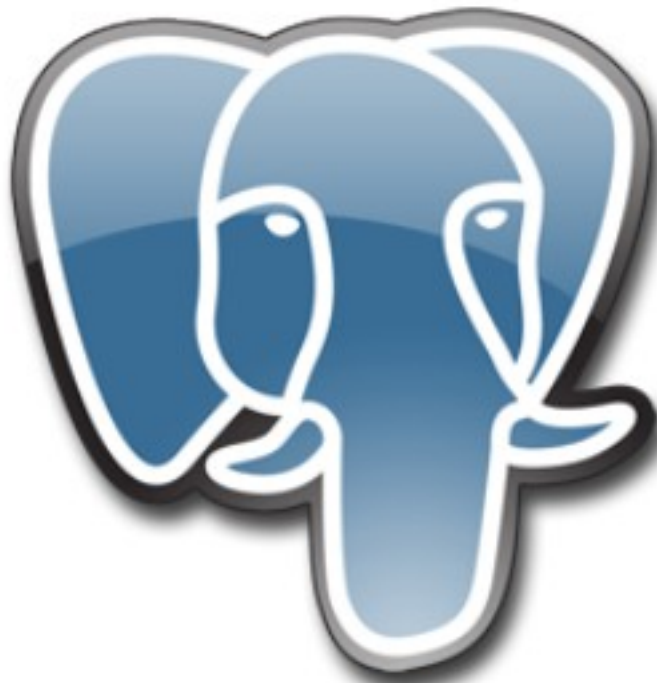
- Позволяет процессу «подсесть» к уже выполняющейся операции SeqScan другого процесса



Кроме того

- Autovacuum включён по умолчанию
- Несколько параллельных процессов autovacuum
- Экономия пространства (varvarlena)
 - VARLENA-значения менее 128 байт: **-6 байт!**
 - заголовок строки: **-4 байта!**
- При «читающих» транзакциях xid не меняется
- Автонастройка bgwriter
- ORDER BY ... LIMIT w/o sorting («Top N» sorting)

Разработчикам баз данных



Полнотекстовый поиск

- contrib/tsearch2 --> ядро системы
- Новый синтаксис
- Упрощённое создание индекса (без триггера)
- Простая миграция и апгрейд версии PostgreSQL



Полнотекстовый поиск

```
CREATE TEXT SEARCH DICTIONARY thesaurus_astro (  
    TEMPLATE = thesaurus,  
    DictFile = thesaurus_astro,  
    Dictionary = english_stem  
);  
  
ALTER TEXT SEARCH CONFIGURATION russian  
    ADD MAPPING FOR lword, lhword, lpart_hword  
    WITH thesaurus_astro, english_stem;  
  
CREATE INDEX pgweb_idx ON pgweb  
    USING gin(to_tsvector('russian', title || body));  
  
SELECT ts_rank_cd(textsearch_index, q) AS rank, title  
FROM pgweb, plainto_tsquery('supernova star') q  
WHERE q @@ textsearch_index  
ORDER BY rank DESC LIMIT 10;
```

Поддержка XML

- Новый тип данных: `xml`
- Стандартные (ISO/ANSI SQL) функции публикации SQL/XML
- DTD-валидация
- XPath: функция `xpath()` (с поддержкой Namespaces)
- Альтернативные функции экспорта объектов БД в XML
- B-tree и GIN-индексы для XML-данных, полнотекстовый поиск

Поддержка XML

SQL:

```
SELECT XMLROOT (  
  XMLELEMENT (  
    NAME 'some',  
    XMLATTRIBUTES (  
      'val' AS 'name',  
      1 + 1 AS 'num'  
    ),  
    XMLELEMENT (  
      NAME 'more',  
      'foo'  
    )  
  ),  
  VERSION '1.0',  
  STANDALONE YES  
);
```

Результат:

```
<?xml version='1.0'  
  standalone='yes' ?>  
<some name='val'  
  num='2'>  
  <more>foo</more>  
</some>
```

Поддержка XML

```
SELECT *
FROM table1
WHERE ((xpath('//person/@smpl:pid', xdata,
  ARRAY[ARRAY['smpl', 'http://example.com']])) [1])::text = '111';

CREATE INDEX i_table1_xdata ON table1 USING btree(
  xpath('//person/@name', xdata)
);
```

Типы данных

- GUID/UUID
- enum
- МАССИВЫ СОСТАВНЫХ ТИПОВ

Кроме того

- Автоматическая инвалидация плана («чистка» кэша функций)
- Table Function
- Updatable Cursors
- ORDER BY NULLS FIRST/LAST (ISO/ANSI SQL)
- Поддержка NULL-значений в GiST-индексах

Администраторам баз данных



EXPLAIN: больше информации

QUERY PLAN

```
-----  
Sort (cost=34.38..34.42 rows=13 width=176) (actual time=0.946..0.948 rows=6  
loops=1)  
  Sort Key: obj2tag.o2t_tag_name  
  Sort Method: quicksort Memory: 18kB  
  -> Hash Join (cost=19.19..34.14 rows=13 width=176) (actual time=0.812..0.835  
rows=6 loops=1)  
    Hash Cond: (obj2tag.o2t_tag_obj_id = tag.obj_id)  
    -> Bitmap Heap Scan on obj2tag (cost=4.43..19.21 rows=13 width=172)  
(actual time=0.079..0.093 rows=6 loops=1)  
      Recheck Cond: (o2t_obj_obj_id = 179313440)  
      Filter: (obj_status_did = 1)  
      -> Bitmap Index Scan on u_obj2tag_ids (cost=0.00..4.43 rows=24  
width=0) (actual time=0.037..0.037 rows=24 loops=1)  
        Index Cond: (o2t_obj_obj_id = 179313440)  
    -> Hash (cost=9.89..9.89 rows=389 width=12) (actual time=0.700..0.700  
rows=389 loops=1)  
      -> Seq Scan on tag (cost=0.00..9.89 rows=389 width=12) (actual  
time=0.012..0.396 rows=389 loops=1)  
    Total runtime: 1.064 ms  
(13 rows)
```

Кроме того

- `pg_standby` – создание Warm Standby (Linux, Windows)
- Удаление пользователя вместе со всеми принадлежащими ему объектами
- GSSAPI
- Улучшенная сборка под Win32 (MS VC++)

Кроме того

- Оценки стоимости вызовов функций:

```
ALTER FUNCTION log_data(text)  
    COST 100000 ROWS 1;
```

- Per-function GUC settings:

```
ALTER FUNCTION log_data(text)  
    SET synchronous_commit TO OFF;
```

- Создание таблиц по «подобию»:

```
CREATE TABLE dict2 (LIKE dictionary INCLUDING INDEXES);
```

Проекты сторонних разработчиков



PL/pgSQL Debugger

The screenshot shows the PL/pgSQL Debugger window titled "Debugger - adm.grant_on_functions". The main pane displays the source code of the function, with the line `num := 0;` highlighted in green and a red stop icon on the left margin, indicating the debugger is paused at line 11. The code includes a variable declaration, a BEGIN block, a comment, and a FOR loop with a SELECT statement. The output pane at the bottom shows a table with four rows of data.

```
num integer;
BEGIN
num := 0;
/* select all function's names in given schema
FOR func IN
SELECT
    proname, proargtypes
FROM
    pg_proc AS pro
JOIN pg_namespace AS ns ON pro.proname:
WHERE
    ns.nspname = schema_name
    AND proname LIKE mask
```

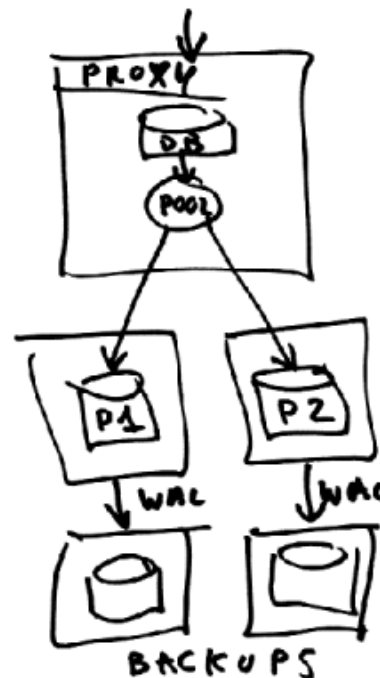
Name	Type	Value
role_name	text	nikolay
permission	text	read
mask	text	*
schema_nar	text	public

Parameters Local Variables Results

Paused at line 11 Ln 16 Col 32 Ch 511

Продукты Skуре

- PL/Proxy – прокси-язык для удаленного вызова процедур и партицирования данных между разными базами.
- PgBouncer – простой и быстрый менеджер соединений.
- SkyTools
 - WalMgr
 - Londiste
 - PgQ



Кроме того

- pgSNMP – SNMP-агент для PostgreSQL (мониторинг)
- SEPostgres – расширение, основанное на модели SELinux (безопасность)
- phrrgadmin: настройка Slony и полнотекстового поиска, автовакуума
- Index Advisor



Вопросы?

<http://postgresql.org>

<http://postgresmen.ru>

http://developer.postgresql.org/index.php/Feature_Matrix

nikolay@postgresmen.ru,

Skype: postgresmen

Special thanks to: Bruce Momjian, Josh Berkus, Teodor Sigaev, «PFC»