PGCon-2016: Последние новости

Oleg Bartunov, Alexander Korotkov
Postgres Professional, Moscow University

RIT++, June 1, 2017
PostgreSQL Developers Meeting, May 17, 2016, Ottawa, Canada

- Oleg Bartunov, PostgreSQL Professional
- Josh Berkus, Red Hat
- Joe Conway, Crunchy Data
- Jeff Davis, AWS
- Andrew Dunstan
- Peter Eisenraut, 2nd Quadrant
- Andres Freund, Citus Data
- Stephen Frost, Crunchy Data
- Etsuro Fujita, NTT
- Kevin Grittner, EnterpriseDB
- Robert Haas, EnterpriseDB
- Magnus Hagander, Redpill Linpro
- Heikki Linnakangas, Pivotal
- Amit Kapila, EnterpriseDB
- Alexander Korotkov, PostgreSQL Professional
- Tom Lane, Crunchy Data
- Noah Misch
- Dave Page, EnterpriseDB
- Michael Paquier, VMWare
- Simon Riggs, 2nd Quadrant
- Masahiko Sawada, NTT
- Teodor Sigaev, PostgreSQL Professional
Oleg Bartunov, PostgreSQL Professional
Josh Berkus, Red Hat
Joe Conway, Crunchy Data
Jeff Davis, AWS
Andrew Dunstan
Peter Eisentraut, 2nd Quadrant
Andres Freund, Citus Data
Stephen Frost, Crunchy Data
Eturo Fujita, NTT
Kevin Grittner, EnterpriseDB
Robert Haas, EnterpriseDB
Magnus Hagander, Redpill Linpro
Heikki Linnakangas, Pivotal
Amit Kapila, EnterpriseDB
Alexander Korotkov, PostgreSQL Professional
Tom Lane, Crunchy Data
Noah Misch
Dave Page, EnterpriseDB
Michael Paquier, VMWare
Simon Riggs, 2nd Quadrant
Masahiko Sawada, NTT
Teodor Sigaev, PostgreSQL Professional
Developers meeting

Agenda

<table>
<thead>
<tr>
<th>Time</th>
<th>Item</th>
<th>Presenter</th>
</tr>
</thead>
<tbody>
<tr>
<td>09:00 - 09:10</td>
<td>Welcome and introductions</td>
<td>Dave Page</td>
</tr>
<tr>
<td>9:10 - 9:25</td>
<td>Contributors</td>
<td>Robert Haas</td>
</tr>
<tr>
<td>9:25 - 9:40</td>
<td>Core team</td>
<td>Robert Haas</td>
</tr>
<tr>
<td>9:40 - 9:55</td>
<td>Release management team</td>
<td>Robert Haas/Simon Riggs</td>
</tr>
<tr>
<td>9:55 - 10:05</td>
<td>9.7 Schedule</td>
<td>All</td>
</tr>
<tr>
<td>10:05 - 10:20</td>
<td>Shared testing infrastructure</td>
<td>Amit Kapila</td>
</tr>
<tr>
<td>10:20 - 10:30</td>
<td>Postgres Pro Roadmap</td>
<td>Alexander Korotkov</td>
</tr>
<tr>
<td>10:30 - 10:45</td>
<td>Coffee break</td>
<td>All</td>
</tr>
<tr>
<td>10:45 - 11:00</td>
<td>Sharding/clustering</td>
<td>Simon Riggs</td>
</tr>
<tr>
<td>11:00 - 11:15</td>
<td>Cross-node transactional consistency</td>
<td>Simon Riggs</td>
</tr>
<tr>
<td>11:15 - 11:30</td>
<td>Logical replication</td>
<td>Simon Riggs</td>
</tr>
<tr>
<td>11:30 - 12:00</td>
<td>Any other business</td>
<td>Dave Page</td>
</tr>
<tr>
<td>12:00</td>
<td>Lunch</td>
<td></td>
</tr>
</tbody>
</table>

- Как номинировать на Major contributors
  Команда волонтеров займется этим (Josh Stephen, Joe, Dave)
- Emeritus list для долгонеактивных
- Core team медленно меняется
- Нет ясного механизма
- RMT — хорошая практика, так держать
- За качество отвечают коммитеры!
- Next Release Schedule
  * CF1: September 1 to 30th
  * CF2: November 1 to 30th
  * CF3: January 1 to 31st.
  * CF4: March 1 to 31st.
  * Feature Freeze: March
  * 31st May 16th: Beta

- Нужен общий пул железа для тестирования
- Спор вокруг шардинга
  * FDW sharding? EDB отреклась - это все Брюс, это его проект.
- Distributed transactions в ядро?
- Postgres-XL и Postgres Professional будут вести отдельные сессии на Developers Unconference
- Pglogical форкнут к 1 августа для включения в ядро
- CFM — Commitfest Management Team (2-3 человека) на каждый CF за три недели до старта (CORE team задача)

Version Numbering

* 8 голосов за X.Y.Z
* 13 голосов за X.Y
  X меняется каждый год
  Y — минорный релиз
* 9.7 → 10.0

https://wiki.postgresql.org/wiki/PgCon_2016_Developer_Meeting
PostgreSQL Roadmap

- На самом деле планов развития у постгреса нет
  https://www.postgresql.org/developer/roadmap/
  "PostgreSQL is a non-commercial, all volunteer, free software project, and as such there is no formal list of feature requirements required for development. We really do follow the mantra of letting developers scratch their own itches."

- Однако, постгрес давно используется большим бизнесом и вокруг него сложились компании, которые занимаются поддержкой, разработкой как самого постгреса, так и приложений. Всем нужны планы!

- В России сложились специфические условия для развития постгреса, нас даже заметил Оракл :) Пришло много бизнеса, много новых людей.

- Поэтому Postgres Professional поставил вопрос о планах
On 04/12/2016 01:07 PM, Oleg Bartunov wrote:

> Our roadmap http://www.postgresql.org/developer/roadmap/ is the problem.
> We don't have clear roadmap and that's why we cannot plan future feature
> full release.

As someone who's worked at multiple proprietary software companies, having a roadmap doesn't magically make code happen.

> There are several postgres-centric companies, which have
> most of developers, who do all major contributions. All these companies
> has their roadmaps, but not the community. I think 9.6 release is
> inflection point, where we should combine our roadmaps and release the
> one for the community. Than we could plan releases and our customers
> will see what to expect. I can't say for other companies, but we have
> big demand for many features from russian customers and we have to
> compete with other databases. Having community roadmap will helps us to
> work with customers and plan our resources.

It would be good to have a place for the companies who do PostgreSQL feature work would publish their current efforts and timelines, so we at least have a go-to place for "here's what someone's working on". But only if that information is going to be *updated*, something we're very bad at. And IMHO, a "roadmap" which is less that 50% accurate is a waste of time.

There's an easy way for you to kick this off though: have PostgresPro publish a wiki page or Trello board or github repo or whatever with your roadmap and invite other full-time PostgreSQL contributors to add their pieces.
**Postgres Professional Roadmap**

1. Multimaster cluster with sharding
2. JIT-compilation of queries
3. Pluggable storages
4. Effective partitioning
5. Adaptive query planning
   - 5.1 Query plans management
   - 5.2 Machine learning
   - 5.3 Execution-time planning
6. Backup
   - 6.1 Block-level incremental backup
   - 6.2 Backup validation
   - 6.3 Partial backup and partial restore
7. Connection pooling
8. Page-level data compression
9. Wait events monitoring
10. Better temporary tables
    - 10.1 Temporary tables outside of system catalog
    - 10.2 Temporary tables on standbys
11. Native querying for jsonb with indexing support

**EnterpriseDB database server roadmap**

1. Parallelism
   - Hash node
   - Bitmap Heap Scan
   - Index Scan
   - Vacuum
2. Partitioning
   - Declarative partitioning
3. Foreign Data Wrappers
   - Aggregate pushdown to postgres_fdw
   - Async execution to executor
   - Pluggable API for heap in core
4. Replication
   - Causal reads
5. Vertical Scalability
   - Speedup CLOG access
   - Cache MVCC snapshots-reduce contention
6. Performance
   - Asynchronous, Vectorized execution
   - Hash indexes
7. SQL Features

### Developers Unconference

5 sessions out of 12 by Postgres Professional

<table>
<thead>
<tr>
<th>Time</th>
<th>Session Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>11:15</td>
<td>PARTITIONING</td>
</tr>
<tr>
<td>11:30</td>
<td>LOGICAL REPL W CORE</td>
</tr>
<tr>
<td>11:45</td>
<td>Database schema validation</td>
</tr>
<tr>
<td>12:00</td>
<td>Ephemeral Relations</td>
</tr>
<tr>
<td>12:15</td>
<td>PAGE COMPRESSION</td>
</tr>
<tr>
<td>13:30</td>
<td>IMPRINTS</td>
</tr>
<tr>
<td>13:45</td>
<td>TRANSACTION MANAGER AD MASTER</td>
</tr>
<tr>
<td>14:00</td>
<td>HOW TO GET STARTED REVIEWS</td>
</tr>
<tr>
<td>15:00</td>
<td>EXECUTOR PERFORMANCE</td>
</tr>
<tr>
<td>15:15</td>
<td>PARALLEL INDEX BUILDING</td>
</tr>
<tr>
<td>15:30</td>
<td>ASYNCHRONOUS EXECUTION</td>
</tr>
</tbody>
</table>

[https://wiki.postgresql.org/wiki/PgCon_2016_Developer_Unconference](https://wiki.postgresql.org/wiki/PgCon_2016_Developer_Unconference)
Developers Unconference

5 sessions out of 12 by Postgres Professional

Unconference schedule by Time and Room

<table>
<thead>
<tr>
<th>Time</th>
<th>DMS1160</th>
<th>DMS1110</th>
<th>DMS1120</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wed 11:15-12:15</td>
<td>Partitioning</td>
<td>Database schema validating tools</td>
<td>Ephemeral relations</td>
</tr>
<tr>
<td>Wed 13:15-14:15</td>
<td>Logical replication in core</td>
<td>Wait events monitoring</td>
<td>Page compression 📚</td>
</tr>
<tr>
<td>Wed 14:30-15:30</td>
<td>IO improvements + buffer cache scaling</td>
<td>Transaction manager + multimaster</td>
<td>How to get started reviewing 📚</td>
</tr>
<tr>
<td>Wed 16:00-17:00</td>
<td>Executor performance</td>
<td>Parallel index building</td>
<td>Async execution</td>
</tr>
</tbody>
</table>

https://wiki.postgresql.org/wiki/PgCon_2016_Developer_Unconference
<table>
<thead>
<tr>
<th>Topic</th>
<th>Speaker(s)</th>
<th>Track</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parallel Query Has Arrived! Where We’ve Been, Where We Are, and Where We’re Going</td>
<td>Robert Haas</td>
<td>9.6 Features</td>
</tr>
<tr>
<td>Yandex.Mail success story Experience of a big migration from Oracle to PostgreSQL</td>
<td>Vladimir Borodin</td>
<td>Track DBA</td>
</tr>
<tr>
<td>In-memory column store</td>
<td>Hari Babu</td>
<td>Track Hacking</td>
</tr>
<tr>
<td>ERROR: snapshot too old (why having the option of seeing that is a good thing)</td>
<td>Kevin Grittner</td>
<td>9.6 Features</td>
</tr>
<tr>
<td>Unveiling pg global kv A distributed key value store built on top Postgres</td>
<td>Matthew Kelk</td>
<td></td>
</tr>
<tr>
<td>Index Internals</td>
<td>Heikki Linnakangas</td>
<td>Track Hacking</td>
</tr>
<tr>
<td>B-Tree – explore the heart of PostgreSQL.</td>
<td>Anastasia Lubennikova</td>
<td>Track Hacking</td>
</tr>
<tr>
<td>Autovacuum, explained for engineers</td>
<td>Ilya Kosmodemiansky</td>
<td>Track DBA</td>
</tr>
<tr>
<td>pg_paxos: Paxos-based table replication</td>
<td>Marco Slot</td>
<td></td>
</tr>
<tr>
<td>PostgreSQL High-Availability and Geographic Locality using consul...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The website may be down, but the database is not.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Run Simple Query Faster</td>
<td>Dilip Kumar</td>
<td></td>
</tr>
<tr>
<td>FTS is dead? Long live FTS! New features of Full-text search in PostgreSQL</td>
<td>Oleg Bartunov, Teodor Sigaev</td>
<td></td>
</tr>
<tr>
<td>9.4 The Hard Way Version upgrades with pg_upgrade and rsync using hard links</td>
<td>James Miller, Samantha Billington</td>
<td></td>
</tr>
<tr>
<td>PostgreSQL extendability: Origins and new horizons Towards pluggable storage engines</td>
<td>Alexander Korotkov</td>
<td>Track Hacking</td>
</tr>
<tr>
<td>Experience with PostgreSQL Research and with PostgreSQL</td>
<td>Sean Chittenden</td>
<td>Track Scaling Out</td>
</tr>
<tr>
<td>Lightning talks Short sharp descriptions of short topics</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Parallel Query Architecture

- Gather: nodeGather.c
- Parallel Executor Support: execParallel.c
- Parallel-Aware Executor Nodes: nodeSeqScan.c, nodeForeignScan.c, nodeCustom.c
- Tuple Queue Reader and DestReceiver: tqueue.c
- Parallel Context: parallel.c
- State Synchronization: dfmgr.c, guc.c, combocid.c, snapmgr.c, xact.c
- Dynamic Background Workers: bgworker.c
- Group Locking: lock.c
- Dynamic Shared Memory: dsm.c, dsm_impl.c
- Shared Memory Message Queue: shm_mq.c
- Error/Notice Forwarding: pqmq.c
- Shared Memory Table of Contents: shm_toc.c
- Group Locking: lock.c

TPC-H Testing: Results

- With parallel query enabled, 17 plans used parallelism. The other 5 query plans did not change.
- Of the 17 queries that used parallelism, 15 got faster, 1 was unchanged, and 1 got slower. (Increasing work_mem fixed the regression.)
- 3 queries ran at least 4x faster; 11 queries ran at least 2x faster.
- Full results at rhaas.blogspot.com
Ideal join strategy is a merge join, which we don't know how to do in parallel: Q2, Q13, Q15

Need parallel hash table build: Q3, Q5, Q7, Q8, Q21

Need parallel bitmap index scan: Q4, Q5, Q6, Q7, Q14, Q15, Q20

Need parallel subquery handling: Q2, Q15, Q16, Q22

Need gather merge: Q17

Not sure: Q3, Q9, Q10, Q11, Q18

Already looks ideal: Q1, Q12, Q19

In-memory column storage by Fujitsu
• Чтобы не накапливался большой bloat, нужно обрывать транзакции, которые видят старые данные.
• Параметр old_snapshot_threshold в интервале 1 мин. – 60 дн.
• Обрываем транзакцию только тогда, когда она действительно пытается обратиться к изменявшимся данным.
• Осторожно! Значительный overhead на многоядерных системах.
• В будущем, можно применить этот механизм для standby.
Ускорение простых запросов

Оптимизации:

- Push Down Scan key
- Save Expression Initialization for targetlist and qual
- Save Scan slot
- Save Executor State
- Save Expression Context

Это даёт:

- 25% ускорение INSERT
- 20% ускорение SELECT
- 20% ускорение UPDATE

<table>
<thead>
<tr>
<th>Track Performance</th>
<th>Track DBA</th>
<th>Track Hacking</th>
<th>Track Hacking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster PostgreSQL: Improved Writes</td>
<td>Audit Logging for PostgreSQL</td>
<td>MLS PostgreSQL</td>
<td>How I Learned to Stop Worrying and Love Failover Automated HA on a container stack</td>
</tr>
<tr>
<td>Amit Kapila</td>
<td>David Steele</td>
<td>Joe Conway</td>
<td>Josh Berkus</td>
</tr>
<tr>
<td>Track Performance</td>
<td>Track DBA</td>
<td>Track Hacking</td>
<td>Track Scaling Out</td>
</tr>
<tr>
<td>WAL Reduction Lose WAL size, gain performance</td>
<td>Building a BaaS using PostgreSQL From schema to REST API instantly</td>
<td>Beyond EXPLAIN: Query Optimization From Theory To Code</td>
<td></td>
</tr>
<tr>
<td>Rahila Syed</td>
<td>Lakshmi Narasimhan Parthasarathy</td>
<td>Ryoji Kawamichi, Yuto Hayamizu</td>
<td></td>
</tr>
<tr>
<td>PostgreSQL's buffer manager – Problems &amp; Improvements</td>
<td>The PCI Compliant Database... yours probably is not.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Andres Freund</td>
<td>Christophe Pettus</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Track Hacking</td>
<td>Track Applications</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| Queues in PostgreSQL  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Can elephants queue?</em></td>
<td>Non-volatile Memory Logging</td>
<td>PoWA 3: towards a computer assisted DBA</td>
</tr>
<tr>
<td>Thomas Munro</td>
<td>Takashi Horikawa</td>
<td>Ronan Dunklau</td>
</tr>
<tr>
<td>Track Applications</td>
<td>Track Performance</td>
<td>Track DBA</td>
</tr>
</tbody>
</table>

| Overlooked Data Types  
|---|---|---|
| *A study of useful lesser-known data types.* | A Challenge of Huge Billing System Migration | Not Easy Being Green  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><em>The Long Tail of Buildfarm Failures</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peter van Hardenberg</td>
<td>Tatsuro Yamada</td>
<td>Noah Misch</td>
</tr>
<tr>
<td>Track Applications</td>
<td>Track Case Studies</td>
<td>Track Hacking</td>
</tr>
</tbody>
</table>

| Benchmarking Databases  
|---|---|---|
| *Performance Evaluation* | Lessons in Building a Distributed Query Planner  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Why Some SQL Queries are Hard to Scale</em></td>
<td>Managing OS Provided PostgreSQL Packages</td>
<td></td>
</tr>
<tr>
<td>Jan Wieck</td>
<td>Ozgun Erdogan</td>
<td>Keith Fiske</td>
</tr>
<tr>
<td>Track Performance</td>
<td>Track Scaling Out</td>
<td>Track DBA</td>
</tr>
</tbody>
</table>

| Closing sessions  
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><em>prizes, auctions, fun, games</em></td>
</tr>
<tr>
<td>Dan Langille</td>
</tr>
<tr>
<td>Track Social</td>
</tr>
</tbody>
</table>
Multicore scalability

`pgbench -s 1000 -M prepared -S -T 300 on 4 x 18 cores Intel Xeon E7-8890 processors
(shared_buffers = 32GB, max_connections = 300)`
Групповой коммит: 30% ускорение для 64 клиентов и 133% ускорение для 256 клиентов на read-write pgbench тесте.

Увеличение clog buffers с 32 до 128: 90% ускорение на unlogged таблицах для 256 клиентов на read-write pgbench тесте.

Параметры checkpointers_flush_after, bgwriter_flush_after и backend_flush_after предотвращают накопление слишком большой очереди на запись в кэше ОС.
Проблемы buffer manager'a

- Медленная хэш-таблица. Нужно применять lock-free hash-table или prefix tree.

- Алгоритм вытеснения не масштабируется. Нужно менять на другой.

- Checkpoint'ы нужно специально «размазывать», чтобы выдержала дисковая подспистама.
<table>
<thead>
<tr>
<th>Time</th>
<th>Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:45</td>
<td>Code of Conduct</td>
</tr>
<tr>
<td>1:00</td>
<td>Multi Master + Demo</td>
</tr>
<tr>
<td>2:15</td>
<td>Almost Ready (by Frank)</td>
</tr>
<tr>
<td>3:30</td>
<td>Planning + Web team</td>
</tr>
<tr>
<td></td>
<td>Revising SQL</td>
</tr>
<tr>
<td></td>
<td>Figures + Pics + Docs</td>
</tr>
<tr>
<td></td>
<td>PGCAC + Tidbitiales</td>
</tr>
<tr>
<td></td>
<td>Portfolio on Free BSD</td>
</tr>
</tbody>
</table>

User Unconference

https://wiki.postgresql.org/wiki/Pgcon2016userunconference
# User Unconference

<table>
<thead>
<tr>
<th>Time</th>
<th>DMS 1160</th>
<th>DMS 1110</th>
<th>DMS 1130</th>
</tr>
</thead>
<tbody>
<tr>
<td>10am</td>
<td>Session Pitches and Scheduling - Coffee and snacks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:45AM</td>
<td>PostgreSQL Project Code of Conduct discussion</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:00 Noon</td>
<td>Lunch!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1:00PM</td>
<td>Multi-Master Demo</td>
<td>User Complaints</td>
<td>Postgres on FreeBSD</td>
</tr>
<tr>
<td>2:15PM</td>
<td>Bug Tracker Discussion</td>
<td>Replaying SQL</td>
<td>PGCAC Annual Meeting</td>
</tr>
<tr>
<td>3:30PM</td>
<td>PgInfra &amp; Web Team</td>
<td>Figures &amp; Pics in Docs</td>
<td>Running Postgres in Containers</td>
</tr>
</tbody>
</table>

https://wiki.postgresql.org/wiki/Pgcon2016userunconference
PostgreSQL Community Code of Conduct
Draft May 20, 2016

Introduction
The PostgreSQL Project prides itself on the quality of our code and our work, and the technical and professional accomplishments of our community. We expect everyone who participates to conduct themselves in a professional manner, acting with common courtesy and in the common interest, with respect for all users and developers.

To that end, we have established this Code of Conduct for community interaction and participation in the project’s work and the community at large.

Inclusivity and Appropriate Conduct
The PostgreSQL Project is open to participation regardless of their level of experience with the development and contributions from all users.

We encourage thoughtful, constructive discussions and suggestions for development among our community and related technology, projects and infrastructure.

Personal attacks and comments on personal characteristics are not welcome.

Personal characteristics include, but are not limited to:
- age
- race
- |ethnicity
- color
- gender

GIN diagram — ASCII
Charity Auction

220$ Слон в облаке

350$ Сломанный значок

350$ Зубная Щетка Oracle

60$ Зубная Щетка Oracle
NEW FEATURES of FTS

Oleg Bartunov, Teodor Sigaev
Postgres Professional
Inverted Index in PostgreSQL

No positions in index!
Improving GIN
Create access methods as extension! Let's call it RUM

- Храним в addinfo
  - Координаты слов
    - Ускоряем поиск с ранжированием
    - Ускоряем phrase search
  - Timestamp
    - Ускоряем поиск с упорядочиванием по времени
- Ветки query tree
  - Ускоряем "обратный" поиск Full Query Search

Thanks !