

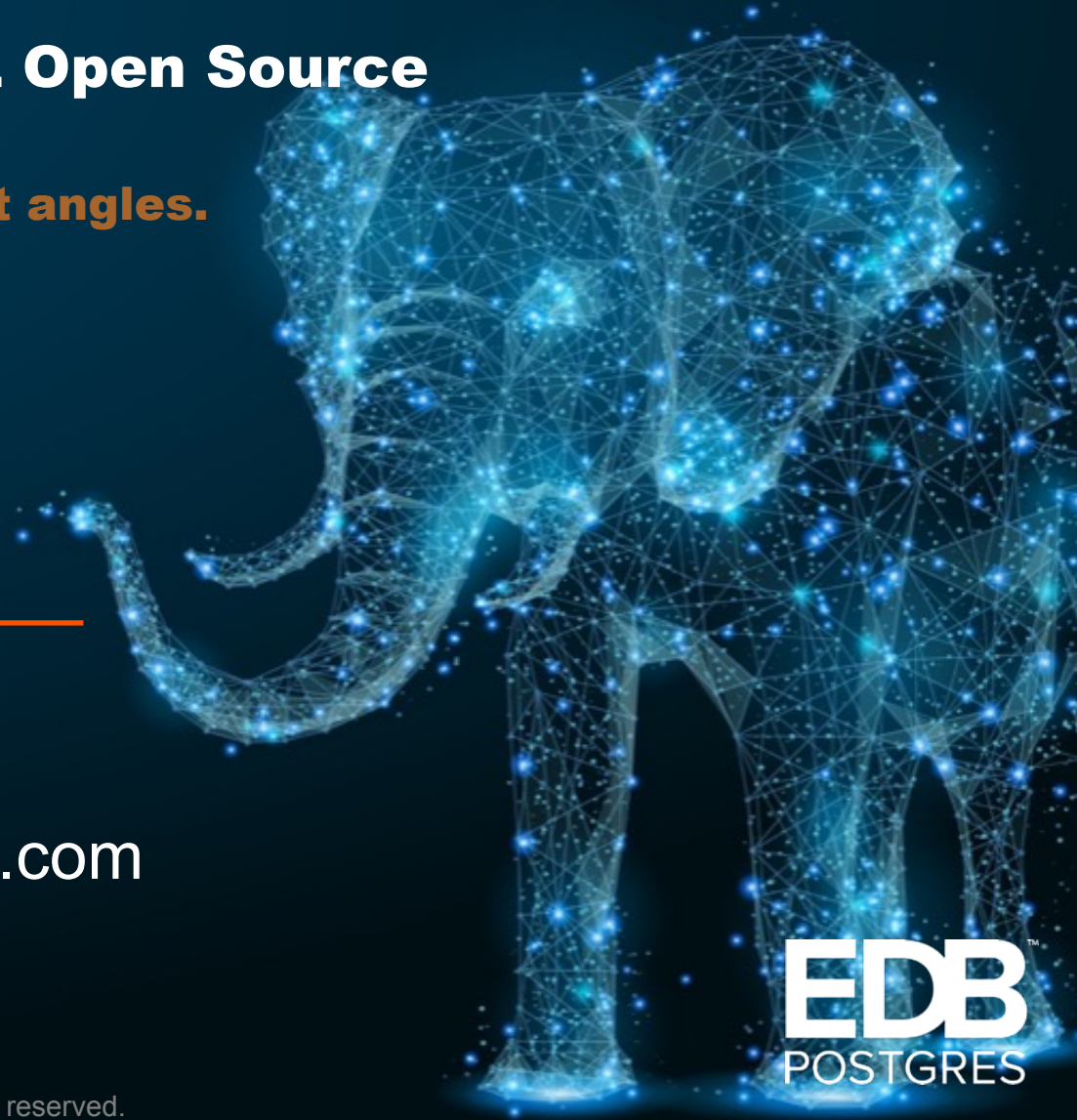
Contributing to PostgreSQL Open Source Community:

Viewing PostgreSQL from different angles.

28-Feb-2020 @ PGConf.IN

Rushabh Lathia

rushabh.lathia@enterprisedb.com



About me

- Rushabh Lathia

- **PostgreSQL Contributor**
 - Bug fixes and features.
 - Patch review.
 - Bug reporting.

- **Working at EnterpriseDB as a Database Architect**
 - Lead the database development team @ Pune, India.
 - Proprietary database features development.
 - Helping customers.

Contents

- About PostgreSQL.
- From where to start contribution?
- How to start contribution?
- Important coding guidelines.
- Feedback Process.
- Case study.
- References.
- Questions?

About OpenSource PostgreSQL Database

- Core database engine.
- World wide investment.
- PostgreSQL Code of conduct.

<https://www.postgresql.org/about/policies/coc/>

- Free to use – BSD Like.
- Very steady and well designed development process.
- Open for new ideas.
- Highly pluggable.
- High code quality.

Development Cycle

- PostgreSQL Committer – 29.
 - <https://wiki.postgresql.org/wiki/Committers>
- 1 year, September to September, until GA.
- Divided into two phase.
 - New development and big/small feature submission July~April.
 - Stability April~September.

PostgreSQL Minor Release

- One STABLE branch per release version.
 - REL_12_STABLE
 - REL_11_STABLE
 - REL_10_STABLE
 - ...
- No new feature, only bug fixes.
- No catalog changes, block format changes, WAL changes on minor version.
- EOL'd after 5 years.

PostgreSQL Mailing list

- PostgreSQL mailing list is one of the primary way to interacts with PostgreSQL Community.
- Track ongoing PostgreSQL Development.
- General support.
- Product release and announcements.
- And many more.
- PostgreSQL Mailing list: <https://www.postgresql.org/list/>

List	Description
pgsql-bugs	If you find any bug, report here.
pgsql-general	General discussion area for users. All non-bug related questions regarding PostgreSQL's version of SQL, and all installation related questions that do not involve bugs or failed compiles, should be restricted to this area.
pgsql-hackers	The PostgreSQL developers team lives here. Discussion of current development issues, problems and bugs, and proposed new features.
pgsql-sql	Any SQL related discussion.
pgsql-committers	Notification of git commits are sent to this list.

Start review/contribution?

- To start one should know the available resource.
- Developer guide: https://wiki.postgresql.org/wiki/Developer_FAQ
- Blogs: <https://planet.postgresql.org/>
- IRC: <https://www.postgresql.org/community/irc/>
- Slack: <https://postgres-slack.herokuapp.com/>
- PostgreSQL conference talks on Youtube.
- Must subscribe to pgsql-hackers for patch/feature submit or review.

Why one should contribute?

- Because database are fun!
- Get name as contributor to the open source community.
- Understand how PostgreSQL works.
- Getting up to date about PostgreSQL new development.
- Become better programmer.
- Commercial opportunities.

How can we contribute?

- Goal: Minimize the committer time.
- Your view on the patch - is it really helpful?
- Review the patch or part of a patch.
- Test the patch.
- Review the documentation.
- No contribution is too small.

Where to start?

Commitfest App: <https://commitfest.postgresql.org/>

Commitfests

The following commitfests exist in the system. New patches should be submitted to commitfest [2020-03](#).

- [2020-07](#) (Future - 2020-07-01 - 2020-07-31)
- [2020-03](#) (Open - 2020-03-01 - 2020-03-31)
- [2020-01](#) (Closed - 2020-01-01 - 2020-01-31)
- [2019-11](#) (Closed - 2019-11-01 - 2019-11-30)
- [2019-09](#) (Closed - 2019-09-01 - 2019-09-30)
- [2019-07](#) (Closed - 2019-07-01 - 2019-07-31)
- [2019-03](#) (Closed - 2019-03-01 - 2019-03-31)
- [2019-01](#) (Closed - 2019-01-01 - 2019-01-31)
- [2018-11](#) (Closed - 2018-11-01 - 2018-11-30)
- [2018-09](#) (Closed - 2018-09-01 - 2018-09-30)
- [2018-07](#) (Closed - 2018-07-01 - 2018-07-31)
- [2018-03](#) (Closed - 2018-03-01 - 2018-03-31)
- [2018-01](#) (Closed - 2018-01-01 - 2018-01-31)
- [2017-11](#) (Closed - 2017-11-01 - 2017-11-30)

Commitfest 2020-03

Search/filter

Shortcuts ▾

New patch

Status summary: Needs review: 136. Waiting on Author: 22. Ready for Committer: 4. Committed: 6. Total: 168.

Active patches

Patch	↓ Status	Ver	Author	Reviewers	Committer	Num cfs	Latest activity	Latest mail
Bug Fixes								
Fix the optimization to skip WAL-logging on table created in same transaction	Needs review	13	Heikki Linnakangas (heikki), Michael Paquier (michael-kun), Kyotaro Horiguchi (horiguti)	Michael Paquier (michael-kun), Kyotaro Horiguchi (horiguti), Noah Misch (nmisch), satyanarayana Narlapuram (snarlap), Sameer Arora (sameera)	nmisch	19	2020-02-01 12:41	2020-01-27 10:28
Spurious "apparent wraparound" via SimpleLruTruncate() rounding (data loss)	Needs review	stable	Noah Misch (nmisch)			6	2020-02-01 12:41	2020-01-05 01:19
Fix unique join costings	Needs review	stable	David Rowley (davidrowley)	Tom Lane (tgl)		6	2020-02-01 12:41	2019-11-30 17:32
fix for BUG #3720: wrong results at using ltree	Waiting on Author		Filip Rembickowski (filiprem)	Ibrar Ahmed (ibrar), Benjie Gillam (benjie)		5	2020-02-01 11:11	2020-02-01 11:13
standby recovery fails when re-replaying due to missing directory which was removed in previous replay.	Ready for Committer	stable	Kyotaro Horiguchi (horiguti), Paul Guo (paulguo)	Anastasia Lubennikova (lubennikovaav)		5	2020-02-01 11:57	2020-01-27 15:24
pg_upgrade fails with non-standard ACL	Needs review		Anastasia Lubennikova (lubennikovaav), Arthur Zakirov (a.zakirov)	Grigory Smolkin (g.smolkin.postgrespro.ru)		4	2020-02-01 12:41	2019-12-17 08:10
Invalid permission check in pg_stats for functional indexes	Waiting on Author	12	Pierre Ducroquet (pinaraf)	Kuntal Ghosh (kuntal_ghosh)		4	2020-02-01 11:13	2019-12-26 23:46
Fix base backup checksum verification for random or zero page headers	Needs review		Michael Banck (mbanck)	Michael Paquier (michael-kun)		3	2020-02-01 12:41	2019-10-18 09:05

Commitfest patch category

- Bug Fixes.
- Clients.
- Code Comments.
- Documentation.
- Miscellaneous.
- Monitoring & Control.
- Performance.
- Procedural Languages.
- Refactoring.
- Replication & Recovery.
- SQL Commands.
- etc.

Spurious "apparent wraparound" via SimpleLruTruncate() rounding (data loss)

[Edit](#) [Comment/Review](#) [Change Status](#)

Title

Spurious "apparent wraparound" via SimpleLruTruncate() rounding (data loss)

Topic

Bug Fixes

Created

2019-02-26 05:41:04

Last modified

2020-02-01 12:41:20 (3 days, 22 hours ago)

Latest email

2020-01-05 01:19:16 (1 month ago)

Status

2020-03: Needs review

2020-01: Moved to next CF

2019-11: Moved to next CF

2019-09: Moved to next CF

2019-07: Moved to next CF

2019-03: Moved to next CF

Target version

stable

Authors

Noah Misch (nmisch)

Reviewers

Become reviewer

Committer

Links

Emails

Spurious "apparent wraparound" via SimpleLruTruncate() rounding ✕

First at 2019-02-02 08:38:22 by Noah Misch <noah at leadboat.com>

Latest at 2020-01-05 01:19:16 by Noah Misch <noah at leadboat.com>

Latest attachment (slru-truncate-module-v2.patch) at 2019-02-17 04:09:13 from Noah Misch <noah at leadboat.com> +

Re: Spurious "apparent wraparound" via SimpleLruTruncate() rounding ✕

First at 2019-07-24 08:27:18 by Kyotaro Horiguchi <horikyota.ntt at gmail.com>

Latest at 2019-07-25 03:45:48 by Noah Misch <noah at leadboat.com>

Attach thread

History

When	Who	What
2020-02-01 12:41:20	Tomas Vondra (fuzzycz)	Closed in commitfest 2020-01 with status: Moved to next CF

Identify the Feature/Patch

- Read through the whole thread.
- Understand the feature or purpose of the patch.
- Is this the feature/patch, I am interested in?
- Does it address all the mention concerned?
- The details about the design has been finalised?
- For a start, would suggest to pick some small bugs, documentation or test case patch for a review.

Apply the patch

Apply the patch on the PostgreSQL branch and compile it:

- git clone [git://git.postgresql.org/git/postgresql.git](https://git.postgresql.org/git/postgresql.git)
- cd postgresql
- ./configure
- git apply ~/Downloads/my_review.patch
- make install

Start review/contribution?

Ask questions about the patch

- Does the patch work as specified?
- Apply the patch to the sources and test.
- Is it really useful, feel free to share your opinion.
- Does patch includes the test case?
- Does patch includes the documentation changes?
- Does patch has any performance overhead on new or existing feature?
- Does it follow the SQL standards?
- Does it follow the community standards?

Start review/contribution?

- Check if patch follows the coding standards
 - <https://www.postgresql.org/docs/devel/source.html>
- More info about reviewing a PostgreSQL patch.
 - https://wiki.postgresql.org/wiki/Reviewing_a_Patch
- More info about submitting a PostgreSQL patch.
 - https://wiki.postgresql.org/wiki/Submitting_a_Patch

Feedback Process

- Make sure update the commitfest status (Author & reviewer)
- Need Review: You can add your self as reviewer for the patch.
- Waiting for Author: After sharing the review, reviewer should change the status to waiting for Author.
- Ready for committer: If reviewer and Author both thinks that patch is now ready to commit, mark it as ready for committer.
- Other status: Committed, Returned with Feedback, Rejected.
- Generally, a patch have to go through the multiple round of reviews and revision before it's get committed, unless it's a very simple patch like typo's or some trivial doc or bug fix patch.

Conclusion

- Be patient
- Help others, try to understand others point of view.
- Write new feature, fix bug, review patches
- Remain polite
- Respect others opinion

Commitfest statistics.

2020-01	Committed: 49. Moved to next CF: 144. Withdrawn: 5. Rejected: 5. Returned with Feedback: 12. Total: 215.
2019-11	Committed: 35. Moved to next CF: 148. Withdrawn: 4. Rejected: 6. Returned with Feedback: 28. Total: 221.
2019-09	Committed: 39. Moved to next CF: 150. Withdrawn: 11. Rejected: 1. Returned with Feedback: 27. Total: 228.
2019-07	Committed: 64. Moved to next CF: 146. Withdrawn: 11. Returned with Feedback: 8. Rejected: 2. Total: 231.
2019-03	Committed: 100. Moved to next CF: 84. Withdrawn: 5. Rejected: 3. Returned with Feedback: 15. Total: 207.

Case-study

commit f0e44751d7175fa3394da2c8f85e3ceb3cdbfe63

Author: Robert Haas <rhaas@postgresql.org>

Date: Wed Dec 7 13:17:43 2016 -0500

Implement table partitioning.

Amit Langote, reviewed and tested by Robert Haas, Ashutosh Bapat, Amit Kapila, Rajkumar Raghuwanshi, Corey Huinker, Jaime Casanova, Rushabh Lathia, Erik Rijkers, among others. Minor revisions by me.

Case-study

commit 72b6460336e86ad5cafd3426af6013c7d8457367

Author: Alexander Korotkov <akorotkov@postgresql.org>

Date: Sat Mar 16 12:15:37 2019 +0300

Partial implementation of SQL/JSON path language

SQL 2016 standards among other things contains set of SQL/JSON features for JSON processing inside of relational database. The core of SQL/JSON is JSON path language, allowing access parts of JSON documents and make computations over them. This commit implements partial support JSON path language as separate datatype called "jsonpath". The implementation is partial because it's lacking datetime support and suppression of numeric errors. Missing features will be added later by separate commits.

Code was written by Nikita Glukhov and Teodor Sigaev, revised by me.
Documentation was written by Oleg Bartunov and Liudmila Mantrova. The work was inspired by Oleg Bartunov.

Case-study

commit 72cf7f310c0729a331f321fad39835ac886603dc

Author: Alvaro Herrera <alvherre@alvh.no-ip.org>

Date: Wed Apr 11 15:29:31 2018 -0300

Fix ALTER TABLE .. ATTACH PARTITION ... DEFAULT

If the table being attached contained values that contradict the default partition's partition constraint, it would fail to complain, because CommandCounterIncrement changes in 4dba331cb3dc coupled with some bogus coding in the existing ValidatePartitionConstraints prevented the partition constraint from being validated after all -- or rather, it caused to constraint to become an empty one, always succeeding.

Author: Álvaro Herrera. Tests written by Amit Langote

Reported-by: Rushabh Lathia

Diagnosed-by: Kyotaro HORIGUCHI, who also provided the initial fix.

Reviewed-by: Kyotaro HORIGUCHI, Amit Langote, Jeevan Ladhe

ARE YOU AWESOME?

WE'RE HIRING

Connect with
us at EDB Booth

Department

Position Title

Location

Product Development	Database Internals	Pune/Bangalore
Product Development	Java/Kafka/JDBC	Pune
Product Development	Devops - Build & Release	Pune
Product Development	Scrum Master	Pune
Product Development	Technical Architect - Java	Pune
Technical Services	Oracle/Postgres DBA's	Pune
Professional Services	Sales Engineer - Oracle/Postgres	Remote
IT	Drupal	Pune
Sales	AE - Sales	Remote
Customer Success Technical	Customer Success Specialist	Remote

References

- [A journey from Postgres enthusiast to committer.](#) - Amit Kapila
- [How to get a feature committed](#) - Michael Paquier
- [Reviewing PostgreSQL Patches for Fun and Profit](#) - David Steele

The background features a series of overlapping, horizontal, chevron-like shapes in shades of orange and red, pointing towards the right. Scattered throughout the background are numerous semi-transparent circles of varying sizes, also in orange and red tones, creating a bokeh effect.

QUESTIONS & DISCUSSION

THANK YOU

info@enterprisedb.com
www.enterprisedb.com



EDB
POSTGRES