

ABDULLAH HANEFI ÖNALDI
& NAISILA PUKA
PGCON 2023

ADDING PGI5 SUPPORT TO CITUS EXTENSION: STRATEGIES AND LESSONS LEARNED

ABOUT US

Naisila Puka

- Software Engineer @ Microsoft
- Working on the Citus extension for 2+ years.
- npuka@microsoft.com
- From Albania

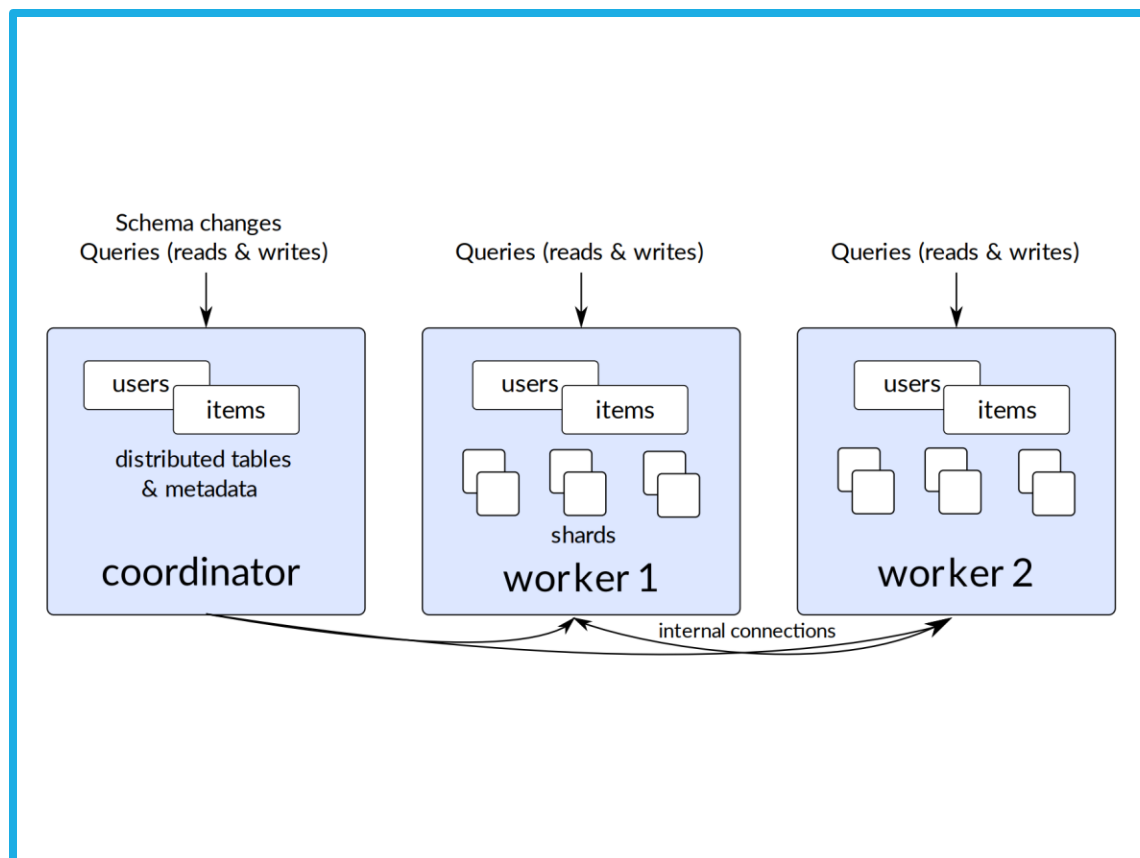
Abdullah Hanefi Önalđı

- Software Engineer @ Microsoft
- Working on the Citus extension for 4+ years.
- hanefi.onaldi@microsoft.com
- From Turkey

OUTLINE

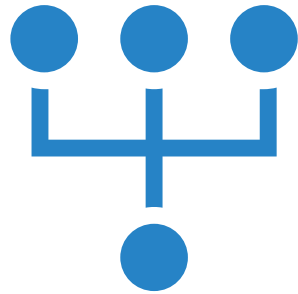
- Citus: Distributed PostgreSQL as an Extension
- Maintaining a Postgres extension like Citus
- PG15 Support Timeline
- PG15 Support Steps - Compilation
- PG15 Support Steps - Current logic sanity
- PG15 Support Steps - PG15 new features integration
- How we tracked new features in PG15
- Revisit PG15 Support Timeline
- New feature in PG15 – Is it supported in Citus?

CITUS: DISTRIBUTED POSTGRESQL AS AN EXTENSION



- Citus is a PostgreSQL extension that adds the ability to distribute and replicate PostgreSQL tables across a shared-nothing PostgreSQL cluster
- Open-source repo on GitHub: <https://github.com/citusdata/citus>
- Core component of Azure Cosmos DB for PostgreSQL
- Citus supports latest 3 PG releases in one codebase.
- We want to "demystify" what it means to keep Citus an updated extension of Postgres – and not just a fork!

PURPOSE OF THIS PRESENTATION



Share the strategies and lessons learned on dividing tasks & resources for PG15 support



Serve as a reference for future PG integrations in Citus and other extensions

MAINTAINING A POSTGRES EXTENSION LIKE CITUS

Citus is a HUGE extension of Postgres:

- Multi Node Postgres!
- Intercept Postgres's planner & executor
- Citus Utility hook and Columnar Utility hook before Postgres's standard utility process
- Extend database directory, extend the deparser
- Citus's own configuration parameters, own custom scan nodes
- Etc ...

Bigger Extension <-> More work on supporting new PG versions

PG 15 SUPPORT TIMELINE

- Real-time following PG updates step by step
- Supporting PG 15 was a moving target in a window of 5 months in 2022:
 - **Start** - PostgreSQL 15 Beta 1 – May 19th
 - PostgreSQL 15 Beta 2 – June 30th
 - PostgreSQL 15 Beta 3 – August 11th
 - PostgreSQL 15 Beta 4 – September 8th
 - PostgreSQL 15 RC 1 – September 29th
 - PostgreSQL 15 RC 2 – October 6th
 - **Finish** - PostgreSQL 15 – October 13th

FIRST – SUCCESSFUL COMPILATION

- Update CONFIGURE script to include PG 15
- Some variables no longer exist / have been replaced, e.g:
 - Value node struct has been removed, replaced by separate Integer, Float, String, and BitString node types
- Functions/Objects/Variables/Properties added/renamed/changed, e.g:

```
#if PG_VERSION_NUM >= PG_VERSION_15
#define RelationCreateStorage_compat(RelFileNode rnode, char relpersistence, bool register_delete)
    RelationCreateStorage(rnode, relpersistence, register_delete)
#else
#define RelationCreateStorage_compat(RelFileNode rnode, char relpersistence, bool register_delete)
    RelationCreateStorage(rnode, relpersistence)
#endif
```
- PG 15 specific: new `shmem_request_hook` hook introduced to request additional shared mem for extensions

SECOND – CURRENT LOGIC SANITY

- Successful compilation is NOT enough.
- Should update INTERNAL LOGIC accordingly to make sure current features function properly.

- Example:

Pre PG15, `tgisinternal = true` for a "child" trigger on a partition

In PG15, `tgisinternal = false` in that case.

In Citus, we used `tgisinternal` value to prevent trigger creation conflicts on partitions when adding a new node.

RESULT -> adding a new node FAILED in PG15, even though Citus compiled successfully.

SECOND – CURRENT LOGIC SANITY

- Updating internal logic sounds great – but where to start?
- First: PORT THE RULEUTILS CHANGES
- Ruleutils: Functions to convert stored expressions/querytrees back to source text
- We maintain a copy of ruleutils in Citus repo for static functions and shard-specific deparsing functions
- We port all static function changes one by one

SECOND – CURRENT LOGIC SANITY

NEXT:

- We can proudly say that Citus has an extensive and thorough test suite 😊
- Making sure our test suite passes with PG15 helps A LOT!

SECOND – CURRENT LOGIC SANITY

- Process: Simple `tgisinternal` flag example

1. Find the relevant PG commit breaking the current logic

[Create foreign key triggers in partitioned tables too · postgres/postgres@f456634 \(github.com\)](#)

`descr: ...this commit also changes what tgisinternal means in some cases...`

2. Fix the logic in Citus and put a reference to the PG commit in the commit description

[Prevent creating child triggers on partitions when adding new node](#)

`descr: Pre PG15, tgisinternal is true for a "child" trigger on a partition cloned from the trigger on the parent.`

`In PG15, tgisinternal is false in that case. However, we don't want to create this trigger on the partition since it will create a conflict when we try to attach the partition to the parent table:`

`ERROR: trigger "... " for relation "{partition_name}" already exists`

`Solution: add an extra check on whether the parent id is invalid to make sure this is not a child trigger`

`Relevant PG commit:`

`f4566345cf40b068368cb5617e61318da60676ec`

HOW WE TRACKED NEW FEATURES IN PG 15

- Official git repository for PostgreSQL, particularly REL_15_STABLE branch
- [PostgreSQL: Release Notes](#)
- [PostgreSQL: Feature Matrix](#)
- [PostgreSQL 15 New Features \(hpe.com\)](#) – initially released for beta1, updated for GA
- [pg15 – select * from depez;](#) - Great blog authored by Hubert "depez" Lubaczewski
- [Postgres 15 support - Missing features in Citus · Issue #6162](#) - Parent issue to track all PG 15 changes

REVISIT TIMELINE

- PostgreSQL 15 Beta 1 – May 19th
- PostgreSQL 15 Beta 2 – June 30th
- PostgreSQL 15 Beta 3 – August 11th
- PostgreSQL 15 Beta 4 – September 8th
- PostgreSQL 15 RC 1 – September 29th
- PostgreSQL 15 RC 2 – October 6th
- PostgreSQL 15 – October 13th

REVISIT TIMELINE

- PostgreSQL 15 Beta 1 – May 19th : SQL/JSON feature
- PostgreSQL 15 Beta 2 – June 30th : improvements to SQL/JSON feature
- PostgreSQL 15 Beta 3 – August 11th
- PostgreSQL 15 Beta 4 – September 8th : SQL/JSON feature is reverted, builds are broken
- PostgreSQL 15 RC 1 – September 29th : message wording change, tests are broken
- PostgreSQL 15 RC 2 – October 6th
- PostgreSQL 15 – October 13th : Introduces new function, builds are ok, tests are broken

PG 15 SUPPORT TIMELINE

- Important to carefully follow each beta and RC release: test Citus PG compatibility frequently.
 - Check builds everyday, run all regression tests for Citus by building PostgreSQL from source.
- The SQL/JSON features proposed for this PG release in beta1 have been removed as of beta4.
 - We also reverted our commits for these features.
- Cannot even trust Release Candidates: Function name conflict `ReplicationSlotName` after RC2!
 - We renamed our function to `ReplicationSlotNameForNodeAndOwner`.

VERSION SUPPORT MATRIX

- Supported
- Tests Fail
- Not Supported

	beta4	rc1	rc2	GA
11.1.0	Supported	Tests Fail	Tests Fail	Not Supported
11.1.1	Supported	Tests Fail	Tests Fail	Not Supported
11.1.2		Supported	Supported	Not Supported
11.1.3				Supported
11.1.4				Supported

NEW FEATURE IN PG 15 – IS IT SUPPORTED IN CITUS?

- Depending on the feature, we give 3 different answers to this question:
 1. Yes, and we have tests to prove it.
 2. Yes, and we tested it manually and we are confident that it will work.
 3. No, but we print meaningful error messages and hints.

NEW FEATURE IN PG 15 – IS IT SUPPORTED IN CITUS?

Yes, and we have tests to prove it

- NULLS NOT DISTINCT
- New regexp SQL functions,
- HEADER option in COPY,
- Numeric type with negative scale (only on non-distribution columns)

Yes, and we tested manually

- Prevent CREATE OR REPLACEVIEW from changing collation of output columns.
- log_destination=jsonlog
- Archiving via loadable modules


NEW FEATURE IN PG 15 – IS IT SUPPORTED IN CITUS?

No, but we print meaningful error messages and hints.

- CLUSTER commands on partitioned tables
- MERGE command on distributed tables
(implemented later in the following Citus 11.2 release)
- ALTER TABLE .. SET ACCESS METHOD
(implemented later in the following Citus 11.2 release)

NEXT - PGI 6

- Just recently we started working on supporting PGI 6
- PGI 6 Beta 1 released on May 25th – draft PR already open in Citus GitHub repo
- Focus Areas:
 - Update the planner hook with the new permission info entry; vars are outer-join-aware.
 - Update Columnar Extension to use the new relation physical identifiers
 - PGI 6 new feature: SQL/JSON constructors and identity functions
 - List will be updated as we further progress into it

- 
- Special thanks to Önder Kalacı for his contributions and guidance throughout this whole process.
 - onderk@microsoft.com Principal Software Engineer
 - Thank you all for your attention.
 - Q&A

REFERENCES

- [KEYNOTE: The Distributed PostgreSQL Problem & How Citus Solves it | Citus Con 2023 | Marco Slot - Speaker Deck](#)