

# Crazy Things You Can Do With PostgreSQL Indexes

Chris Travers

March 28, 2023

# About Me

I have around twenty four years of experience with PostgreSQL and databases in many roles – developer, database hacker, dba, and more. I love PostgreSQL's versatility.

# About This Talk

- Theory and practice
- Three scenarios with solutions
- Discussions of costs on these scenarios

# Theory: Functions

Discrete Math Definitions Made Simple:

- Relations are sets of corresponding facts. They are **correlations**.
- Functions are relations where the domain is unique. They are **inferences**.

Relational databases are, as Codd put it, inferential systems so they effectively work based on functions. Second Normal Form means that all tables in a database are **functions**.

## Derived Data

Although tables can be functions, some functions can also be calculated. These include data derived from data in a row:

- area of a postgis geometry type can be calculated: ST\_Area()
- length of a string can be calculated.
- a lower case string is a function of an input string
- defined substrings can also be calculated.

# Referential Transparency and Immutable Functions

To be used in this way a function must be referentially transparent:

- It must be deterministic, and
- It cannot have side effects

# BTrees

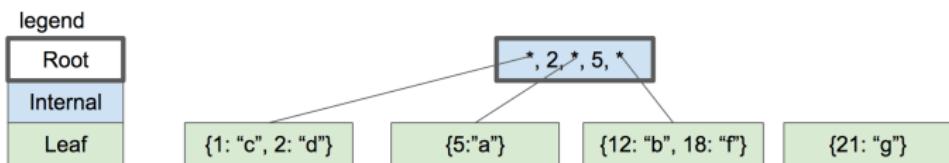


Image taken from cstack's db tutorial  
([https://github.com/cstack/db\\_tutorial/](https://github.com/cstack/db_tutorial/))

# Advanced Indexing Capabilities in PostgreSQL

- Expression or functional indexes
- Partial indexes
- Index-only scans
- Many more index structures besides btrees
- queries with order by and limit can use indexes for this

# Rodering on Derived Data

- Large table of documents, each document has a "content" field and a "database" field
- Each document's content has a format depending on the value of the "database" field
- Want to sort documents in the 'patent' database based on a date extracted from the "content" field
- Only around 10% of records need this

# Options that Don't Work

- Reprocess whole table with a worker
- Sort documents in the front end

# With a Function and Index

- PL/Perl immutable function that extracts and returns the date signature is `patent_date(document)` returns date
- Index the output of the function
- Integrate this into the front-end

# Example

```
CREATE OR REPLACE FUNCTION patent_date(document)
RETURNS date LANGUAGE PLPERL AS ...;
```

```
CREATE INDEX doc_patent_date on document
(protein_id, patent_date(document))
where doctype = 'patentd';
```

# Downsides to Approach

- Requires integration with the ORM layer
- Modifying the function requires a reindex operation

# Leveraging Index-Only Scans in Join Traversal

- Life sciences database
- Want to implement row-level security to reduce access
- Permissions added on two tables.
- Enforcing permissions requires traversing joins against up to five tables

# Diagram of main tables



# Disregarded options

- Accept poor performance
- Replicate data to all tables

# Solution using Index Only Scans

- Identify join order needed for enforcement
- For each traversed table, create index with (foreign key, primary key)

# Downsides of This Solution

- Difficult to reason about
- You are tied to a join order
- Accommodating multiple join orders incurs extra write overhead

# Scenario

- Large distributed data warehouse environment
- On point of entry nodes, data gets digested and sent only once
- But on updates we need the difference.....
- Processing is based on batches

# Options Ruled Out

- Indexing all of the data (multiple TB)

# Solution with Partial Index

- Trigger updates a "materialized" flag
- Partial index where "materialized" is false
- vacuum keeps index small and clean

# Downsides to this Solution

- Need to be aware the index is only for one application, not for the whole data.
- Otherwise worked well

# Conclusions

- Advanced index functions are really useful
- They allow you to do things you couldn't do before
- But they complicate troubleshooting and performance awareness
- As always, power tools are inelegant where hand tools will suffice

# Thanks to

- Novozymes AB and Adjust GmbH for where I got this experience

# Questions

Questions?