

[« Prev](#)[Next »](#)

Postgres Weekly

Issue 112 — July 1, 2015

Introducing HypoPG, Hypothetical Indexes for PostgreSQL

Want to know if an index will be useful before you create it? Here's an extension to help with just that, by hypothetically creating an index before it actually writes it out. Great to see if an index will be useful ahead of time!

POSTGRESQL.ORG [#NEWS](#)

How to Aggregate Data with PostgreSQL using Window Functions

If you're doing any level of complex or even moderately complex aggregation across your data, window functions are hugely useful. Here's an example of how you can use them.

ADRIAN SERAFIN [#TUTORIAL](#)

Showing 'Similar' Results in Postgres

Want your application's search to be a bit more flexible than perfectly matching what the user entered? Postgres has some built in flexibility for finding 'similar' words and sounds.

LEIGH HALLIDAY [#TUTORIAL](#)

VividCortex and PostgreSQL Monitoring

VividCortex is the easiest, safest, and most detailed way to monitor PostgreSQL. Unlike other tools that just capture a few status metrics, VividCortex measures every query, every process, and every metric on your entire system, down to the microsecond to improve performance.

VIVIDCORTX [#SPONSORED](#)

Postgres 9.5 Feature Highlight: archive_mode = always

MICHAEL PAQUIER [#NEWS](#)

Searching Serialized Fields in Rails Using Postgres Arrays

If you're using Arrays or other more unique datatypes, the default WHERE when searching isn't going to be your ideal. Fortunately there is a better option that works great in Postgres, here's how to take advantage of it in Rails directly.

VIGET.COM [#TUTORIAL](#)

Optimizing Full Text Search with Postgres tsvector Columns and Triggers

THOUGHTBOT [#TUTORIAL](#)

par_psql: Parallel psql

Parallel 'psql'. Run parallel queries inline in PostgreSQL's psql tool. Useful for 'slightly big data', GIS.

GRAEME B BELL [#TOOLS](#)

Holycorn: PostgreSQL Multi-purpose Ruby Data Wrapper

Holycorn makes it easy to implement a Foreign Data Wrapper for Postgres using Ruby. It's built on top of mruby.

FRANCK VERROT [#CODE](#)[« Prev](#)

[Next »](#)

Want to subscribe? Enter your address here

Subscribe now »

Easy to unsubscribe at any time. Your e-mail address [is safe](#) — here's [our privacy policy](#).