

Subscribe now »

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

[« Prev](#)

[Next »](#)

Postgres Weekly

Issue 3

March 27, 2013

Featured

[Batch Update](#)

SQL is much better at handling many rows in a single statement than handling single rows across many statements. [Dmitri Fontaine](#) of 2nd Quadrant France shows how to take better advantage of batch updates within Postgres.

[Scaling PostgreSQL at Braintree: Four Years of Evolution](#)

Braintree has run a variety of data stores over the year including Riak, MongoDB, MySQL, and others. They highlight here their four year journey with Postgres which includes sharding their data and now running an automated failover setup.

From our sponsor

[Try New Relic at no charge](#)

Deploy New Relic and get your Nerd Life t-shirt. Monitor your Ruby, PHP, Python, Java and .NET web apps in less than 2 minutes.

([Sponsor this newsletter](#))

The rest for this week

[PostgreSQL, Python, and Squid](#)

In his talk from PyPgDay Christophe Pettus of PG Experts shows the flexibility of Postgres with custom datatypes. In this example he creates a custom squid datatype using Python inside Postgres.

[Autoupdating Full Text Search in Ruby](#)

Postgres' Full Text Search is powerful and directly built-in. This simple migration for Ruby creates a trigger to automatically update your index, and the same principle could be applied in any language.

[Sharding with the Django ORM](#)

Sharding, the process of horizontally partitioning rows of a table across multiple DB instances, is a point that so many applications hope to reach. This post along with [slides](#) from [PyPgDay](#) and an [example application](#) do a great job covering how Disqus has implemented this.

[UPSERT/MERGE using Writable CTE in PostgreSQL 9.1](#)

Upsert or merge is a common feature request of Postgres. While we can wait and hope for it to come in future versions with writeable CTEs (Common Table Expressions) a form of this functionality is available today. Read how to begin using or if a rubyist check out the [library we included last week](#)

[« Prev](#)

[Next »](#)

Subscribe now »

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