BRINGING POSTGRES TOWARDS ZERO DOWNTIME MIGRATION

Matthieu Rigal, EuroPython 2015, Bilbao
INTRO

• Speaker: Matthieu Rigal, Frenchman living since 8 years in Berlin. 4 years using Python in Geo-informatics, since 3 years as a POS Data Backend

• Company: orderbird, mobile-based Point of Sales for restaurants and bars, making live data sync, push notifications, external payments, data analysis and much more in a pure-Python backend. We operate 50+ servers for our stack
MIGRATION NEEDED

Two main migration cases:

- Feature extension: classic schema migration (ADD/ALTER TABLE/COLUMN/CONSTRAINT)
  - no big problem, small downtime or hot migration

- New business logic, strategy change: eventually requiring a data migration
  - could require a downtime of several (dozens) of minutes
POSTGRESQL 9.4 IS HOT

- Logical decoding (http://www.postgresql.org/docs/current/static/logicaldecoding.html)

- You can set it as replication level

- Streams the modifications performed via SQL

- Can be extended with output plugins

- Flexible replication slots (http://www.postgresql.org/docs/current/static/high-availability.html)

- Creation, start, stop, replay check
OUR TOOLBOX

• **wal2json** ([https://github.com/ethansf/wal2json](https://github.com/ethansf/wal2json))
  - Converts WAL output to JSON to allow easy manipulation afterwards, written in C

• **pg-migrator**, Python and psycopg2 powered
  - **physical_logical_syncer**: Switching a physical-replicating slave to a logical-replicating second master

• **json_receiver**: Parsing the JSON and applying transformation (migration) on the fly between both masters
WORKFLOW

• Start from classical master-slave replication
• Switch to logical replication and pause it
• Apply migration to logical replicating master
• Replay replication transforming data on the fly
• Deploy application and write directly to new master
LIVE DEMO
FUTURE

• Develop and open-source pg-migrator package

• Integrate with python frameworks

• Django package to automatically transform django migrations into Transformer objects for the Json receiver
SPECIAL THANKS
Ethan Snyder-Frey

QUESTIONS?
PG_MIGRATOR@ORDERBIRD.COM