UDR – Uni-Directional Replication

PGConf.EU 2014

Petr Jelínek

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 318633.
What is UDR

Master-Slave replication
What is UDR

Logical replication

• Allows write queries (TEMP Tables)
• Per-DB replication (in fact per table)
• Cross-version replication
• Different indexes on primary and slave
• Different roles/privileges on primary and slave
Why UDR

• 9.4 introduces Logical Decoding
  – Low overhead
  – Better integration

• Extension
  – Runs/configures/logs under Postgres

• Async or sync apply – per transaction control

• Performance
Performance

pgbench scale 50, -c 32 -j 32
Performance

Custom workload throttled to <2s replication latency

- HotStandby
- UDR
- BDR
- Londiste3
- Slony

Sustainable TPS (higher is better)

© 2ndQuadrant 2014
Performance

pgbench scale 50, -c 32 -j 32

[Bar chart showing performance comparison between Master and Slave for different systems: HotStandby, UDR, BDR, Londiste3, Slony.]
Setup

• Master/Slave
  – shared_preload_libraries = 'bdr'

• Master
  – wal_level = logical
  – max_wal_senders, max_replication_slots

• Slave
  – bdr.connections = 'mymaster'
  – bdr.mymaster_dsn = 'host=myhost dbname=mydb'
Setup

master -> slave
Setup

master → slave1 → slave2
Setup

master1 ----> slave1

master2 ----> slave1
Zero downtime upgrades

• Logical – uses pg_dump internally
  – Similar to londiste/Slony
• Combined – pg_upgrade + logical catchup
  – Fast!
Zero downtime upgrades

- pg_basebackup
- bdr_init_copy
- pg_upgrade
- UDR cachup
Zero downtime upgrades

- **master 9.4** to **slave 9.4** via UDR
- **slave 9.4** to **master 9.4** via UDR
- **master 9.4** to **slave 9.4** via UDR
- **slave 9.5** to **master 9.5** via UDR
- **master 9.4** to **master 9.5** via UDR catchup
- **master 9.5** to **master 9.5** via pg_upgrade
Additional features

- Replication sets
- Delayed apply
- User defined action on conflict
- PK column update
- New nodes can be setup using pg_basebackup
- You can still have physical replication from UDR node
What about DDL?

- `bdr.bdr_replicate_ddl_command(ddl)`
- DDL outside of this function disabled by default
  - `bdr.permit_unsafe_ddl_commands`
Future

- Action based filtering
- Full configuration inside database
- Sequence replication
- Transparent DDL
- Integration into PostgreSQL
Development

• Maintained by 2ndQuadrant
• Patches welcome!
• PostgreSQL license
• http://git.postgresql.org/gitweb/?p=2ndquadrant_bdr.git
• Questions for 2ndquadrant: bdr@2ndquadrant.com
• Questions about general usage: pgsql-general@postgresql.org
Thanks

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 318633.