Upgrade your PostgreSQL Database: Why and how?

Devrim Gündüz

Principal Systems Engineer @ EnterpriseDB

devrim.gunduz@EnterpriseDB.com

Twitter : @DevrimGunduz

About me

- Who is this guy?
  - Using PostgreSQL since 1998.
  - Responsible for PostgreSQL YUM repository (http://yum.PostgreSQL.org) where we host 150+ software
  - Fedora / CentOS packager for 50+ packages.
    - Used to break website, but recently gave up (until I can do more Python)
  - Working at EnterpriseDB.
  - Istanbul, Turkiye.

Social media

- Please tweet!
  - #pgconfeu
  - @pgconfeu

PostgreSQL version numbers

- Current: 2 digit major, 1 digit minor
  - 9.5.3 : 3rd minor version of 9.5 (major version)
- Upcoming: 1 digit major, 1 digit minor
  - 10.0, 10.1 ...
  - 11.0, 11.1...
  - Avoids confusion (no more “I’m using PostgreSQL 9”)
- Major version incompatibilities
- Changes between minor versions and changes between major versions.

Minor version updates

- Stays on the same major version:
  - 9.6.0→ 9.6.1
- Easy!
- Quarterly release:
  - https://www.postgresql.org/developer/roadmap/
- Package manager?
  - Automatic restart!
- In general, a restart is enough, rarely requires extra attention.
  - Release notes!
  - http://www.postgresql.org/docs/devel/static/release.html

Major version upgrades

- Changes the major version:
  - 9.5.5 → 9.6.1
- Data files are not compatible
- Needs some attention
- This talk is mostly about major version upgrades!

Issues

- Time (downtime)
  - Who loves it?
- Application compatibility
- Extension compatibility: Check before starting upgrade process
- (Lifetime) support
  - PGDG supports 5 releases.
- Platform changes
- ...

Why should you keep up2date?

- Security fixes
- Bug fixes
  - Sometimes fixes to avoid data loss or corruption.
- New features (only for major releases)
- Platform support
- Lifetime support

So, how to perform the upgrades?

Installing new major version

- Sure, we need to install the new major version first!

- First step: Install the new major version.
- Source code installation: Use a new directory
- Package installation: Depends
  - Debian/Ubuntu: Allows parallel major version installation
  - Red Hat/Fedora own RPMs: Only one major version, but supports upgrading.
    - Recently more than one major version available, but still not parallel.
  - PGDG RPMs: Allows parallel major version installation
  - Windows: Allows parallel major version installation

The old school way: Dump and restore

- Supposed to work all the times
  - Except when the data is corrupt
    - Duplicate constraints, etc.
- pg_dump or pg_dumpall
  - Parallel pg_dump in 9.3+: Not a big gain for large data sets.
- Tip: Use pg_dump of higher major version
- Cons: May take a long time.
  - pg_restore -j : May not be a big gain for large data sets.
- Pros: Works across different OSes (mostly) (usually) (well, probably).

What about on-disk upgrade?

- pg_upgrade!
  - Not needed for minor version updates
- 9.0+
- --check (-c) option!
  - Checks binary compatibility
- Dumps and restores structures (system tables)
- link mode! (-k), Parallel mode (-j)

What about on-disk upgrade?

- Make sure you delete old cluster and update statistics on new one.
- Pros: Fast
- Cons: No rollback!
- SR slaves: rsync
- Red Hat / Fedora: provides code for upgrading the database using pg_upgrade on OS upgrades.
- PGDG RPMs: Broken code. The RPM maintainer needs to fix it. Don’t know who he/she is :P

(Trigger based) replication and upgrade

Slony, Londiste, Bucardo

- Uses triggers: Overhead (duplicate writing)
- Installation: Needs extra attention
  - Get list of all tables
  - PK requirement (should be anyway)
- Major version upgrades
- Uses separate tables to track changes

(Trigger based) replication and upgrade

Slony, Londiste, Bucardo

- Less downtime (compared to pg_dump/pg_restore)
- Cons: No large object replication
- Slony: Different OS’es
- Maintenance overhead: Schema changes
- **May** take too long to sync

Logical replication framework

9.4+

- A framework for online upgrades
- “Stream the modifications performed via SQL”
- Replication slots
  - From docs: “Changes are sent out in streams identified by logical replication slots. Each stream outputs each change exactly once.”
- Replication sets:
  - Per docs: “Provides a mechanism to control which tables in the database will be replicated and which actions on those tables will be replicated.”
- Much performant than trigger based solutions!
- Allows per-database (or table, etc.) replication.

Logical replication framework

9.4+

- Works across different platforms (PPC on Linux to x86_64 on OS-X) .
- Works only with streaming, not archiving of WAL files.
- Compatibility issues?
- Faster in 9.5+
- wal_level = logical
- Does not (yet) replicate DDL
- Still improving

Tools using logical replication

9.4+

- BDR
- pglogical
- xDB
- Slony (WIP)

Upgrading with pglogical

- pglogical: Replication *and* upgrade solution for PostgreSQL
- Open source: https://2ndquadrant.com/en/resources/pglogical/
- “Extension” to PostgreSQL, submitted for 10.0.
- Temp and unlogged tables cannot be replicated, so no upgrades!
- Upgrade from 9.4+ to 9.5+

Upgrading with pglogical

- No DDL replication (pglogical.replicate_ddl_command)
- shared_preload_libraries=’pglogical’
- Can use track_commit_timestamps = on
- No GUI (yet?)

Upgrading with pglogical

- SELECT pglogical.create_node(

  node_name := 'pgconfeu16',

  dsn := 'host=masternode port=5432 dbname=pgconfeu2016'

  );
- Add all tables in a schema:
  - SELECT pglogical.replication_set_add_all_tables('default', ARRAY['public']);
- Tip: Try creating replication sets before subscribing: Saves time in initial replication (not a must).

Upgrading with pglogical

- Subscriber

- SELECT pglogical.create_node…
- SELECT pglogical.create_subscription ...

Upgrading with pglogical

Finishing the upgrade

- Shut down master
- Redirect your apps to new server
- Use a middleware!

Upgrading with xDB

- xDB: Replication *and* upgrade solution for PostgreSQL and PPAS.
- A product by EnterpriseDB:

  http://www.enterprisedb.com/products-services-training/products-overview/xdb-replication-server-multi-master
- Temp and unlogged tables cannot be replicated, so no upgrades!
- Upgrade from 9.4+ to 9.5+
- Nice GUI, also a command line option.

Upgrading with xDB

- DDL replication:
  - Explicitly execute DDL command via xDB (using GUI "Replicate DDL" option or Rep CLI "replicateDDL" command)
  - Example: To add a new column in published table, choose “Replicate DDL” option on MDN node, and xDB will apply it.
- Option for Trigger based upgrades
  - 9.1+
- No patches needed to PostgreSQL or PPAS.
- Requires test_decoding extension on publication node (for single master replication) or on all nodes (for multi master replication)

Upgrading with xDB

- Setting up replication with GUI:
  - Publisher node
  - Subscriber node
  - Select tables
  - Start replication
- Conflict resolution

Lessons learned:

- Always keep your servers up2date!
  - Minor version updates are relatively cheap
- Upgrade to a new major version to use new features. Do it. Use it, test it!
- pg_dump is not good for large databases
- Keep downtime minimal
- Test, test, test!

Questions, comments?

Upgrade your PostgreSQL Database: Why and how?

Devrim Gündüz

Principal Systems Engineer @ EnterpriseDB

devrim.gunduz@EnterpriseDB.com

Twitter : @DevrimGunduz