zebanza

Postgres Upgrade Strategies

11/06/2020

Pablo Hendrickx@Zebanza.be

Contents of presentation

- **Update policy**
- **What benefits?**
- **Choose your update strategy**

  -->About 20 minutes

Postgres update policy

- **Major release every September**
- **Minor release every 3 months**
- **Wait for first minor update before installing new major version**

Postgres update policy

- **Versioning Numbering**
- **Old:**
  - 9.6.3
  - Major.Major.Minor
  - Up to version 9.6
- **New:**
  - 12.2
  - Major.Minor
  - Version 10 onwards

Benefits

- **Why should I bother with upgrades?**

Benefits

- **Benefits of new major versions**
  - More security
  - Performance
  - New features for developers
  - New SQL implementations
  - Better management features
  - EOL after 5 years
  - Frequently upgrading makes it easier

Benefits

- **Benefits of new minor versions**
  - *Low risk and simple procedure*
  - *Only fixes bugs, security issues, and data corruption problems*
  - *“For minor releases, the community considers not upgrading to be riskier than upgrading.”*

Upgrading minor version

- Update Postgres
- Make sure to update ALL pg packages
- Check the release notes

Upgrading minor version

1. Stop connections
2. Take a backup
3. Stop the cluster
4. Replace the Postgres binaries
5. Restart the cluster

```
[pablo@sol ~]$ dnf list installed "postgres*"
Installed Packages
postgresql-jdbc.noarch                          42.2.12-1.f31                   @pgdg-common
postgresql12.x86_64                             12.3-1PGDG.f31                  @pgdg12
postgresql12-contrib.x86_64                     12.3-1PGDG.f31                  @pgdg12
postgresql12-libs.x86_64                        12.3-1PGDG.f31                  @pgdg12
postgresql12-server.x86_64                      12.3-1PGDG.f31                  @pgdg12
[pablo@sol ~]$ sudo dnf update postgresql12 postgresql12-server postgresql12-contrib postgresql12-libs postgresql-jdbc
```

Upgrading major version

- **3 options:**
  - Pg_dump/restore
  - In-place upgrade with pg_upgrade
  - Online upgrade

Upgrading major version

- **Pg_dump/restore**
- **pg_dumpall > pg_dump**
  - pg_dumpall also gets global objects

Upgrading major version

- **Pg_dump/restore**
  - Get the new software
  - Initiate an empty database on new path
  - Use psql to restore the dump

```
bash-5.0$ pg_dumpall -l postgres -h localhost -p 5432 -U postgres > dump.sql
bash-5.0$
bash-5.0$ psql -d postgres -h new_host -p 5432 -U postgres < dump.sql
```

Upgrading major version

- **Pg_dump/restore**
- **Pro’s:**
  - Works since v7.0
  - Simplicity
  - Robust
  - Small DB → perfect solution

Upgrading major version

- **Pg_dump/restore**
- **Contra’s:**
  - Larger database → slow, larger downtime
  - Requires double space

Upgrading major version

- **In-place upgrade with pg_upgrade**
  - Detaches pg_clog and pg_catalog
  - Moves data files
  - Reattaches to new binaries

Upgrading major version

- **In-place upgrade with pg_upgrade**
  - Get new software (also for extensions!)
  - Create empty db of new postgres version
  - Stop old postgres db
  - Start pg_upgrade command
  - Start new database
  - Start analyze statistics script

```
/usr/pgsql-12/bin/initdb /tmp/pg12
systemctl stop postgresql-11
/usr/pgsql-12/bin/pg_upgrade -b /usr/pgsql-11/bin -B /usr/pgsql-12/bin -d /var/lib/pgsql/11/data -D /var/lib/pgsql/12/data
systemctl start postgresql-12
./analyze_new_cluster.sh
```

Upgrading major version

- **In-place upgrade with pg_upgrade**

```
Upgrade Complete
----------------
Optimizer statistics are not transferred by pg_upgrade so,
once you start the new server, consider running:
    ./analyze_new_cluster.sh

Running this script will delete the old cluster's data files:
    ./delete_old_cluster.sh
```

Upgrading major version

- **In-place upgrade with pg_upgrade**
  - Hardlink option (-k)
  - Blazing fast
  - Old database is lost
  - Data dirs must be in same filesystem
  - No double space

Upgrading major version

- **In-place upgrade with pg_upgrade**
- **Pro’s**
  - Much faster for large databases
  - Even faster with hardlink option
  - Not too complicated

Upgrading major version

- **In-place upgrade with pg_upgrade**
- **Contra’s**
  - Double space (unless using hardlink option)
  - Risky if using hardlink option
  - Buggy in containers

Upgrading major version

- **Online upgrade**
  - Setup new database cluster
  - Setup replication from old one to new one
  - Perform failover

Upgrading major version

- **Online upgrade**
  - Replicating from old to new can only be done with logical replication
  - Else, third party tool required!

Upgrading major version

- **Online upgrade**
  - Enterprisedb replication server
  - Slony-I

Upgrading major version

- **Online upgrade**
- **Pro’s**
  - Very low down time
  - Safe! Can switchover standby

Upgrading major version

- **Online upgrade**
- **Contra’s**
  - Complexity
  - Double space

Sources

- Ilya Kosmodemiansky

  https://fosdem.org/2020/schedule/event/postgresql_an_ultimate_guide_to_upgrading_your_postgresql_installation/
- https://www.postgresql.org/support/versioning/
- https://www.enterprisedb.com/edb-docs/d/postgresql/reference/manual/9.4.24/pgupgrade.html
- https://www.percona.com/blog/2019/04/04/replication-between-postgresql-versions-using-logical-replication/
- https://www.cybertec-postgresql.com/en/a-primer-on-postgresql-upgrade-methods/

Questions?

Pablo.Hendrickx@zebanza.be