RDS Maintenance: Strategies for Patching and Version Upgrades

Karen Ng
Senior Technical Account Manager

Anita Singh
Database Specialist Principal Solutions Architect

Agenda

- RDS PostgreSQL Maintenance
- PostgreSQL Engine Version upgrade
- Downtime for PostgreSQL
- RDS PG Major Version upgrade – Prerequisites
- Database Upgrade Approaches: From In-Place to near Zero-Downtime
- In-Place Major Version upgrade
- RDS Blue/Green Deployment – Interim state
- Best Practices for performing database engine update
- Extended Major Version Support

RDS PostgreSQL maintenance

- Operating system or Amazon RDS software patches are usually performed without restarting databases
- Database engine upgrades require downtime
    - Minor version upgrades—automatic (AMVU) or manually applied
    - Major version upgrades—manually applied
- You can view upcoming maintenance events available for your DB instance by using the RDS console, the AWS CLI, or the Amazon RDS API.

PostgreSQL Engine Version Upgrades

- Minor version upgrades
  - Patches to the binaries
  - No new functionality
  - May contain important security fixes
  - Auto-minor version upgrade (AMVU)

- Major version upgrades
  - Tracks the community yearly release cycle
  - Introduces new functionality
  - May change system catalogs and page formats
  - Supports skip version in-place upgrade

Current source version	Newest upgrade target	Available upgrade targets								
14.1	14.2	14.2								
13.5	14.2	14.2	14.1	13.6						
13.4	14.2	14.2	14.1	13.6	13.5					
13.3	14.2	14.2	14.1	13.6	13.5	13.4				
12.10	14.2	14.2	13.6							
12.9	14.1	14.1	13.6	13.5	12.10					
12.8	13.6	13.6	13.5	13.4	12.10	12.9				
12.7	13.6	13.6	13.5	13.4	13.3	12.10	12.9	12.8		
11.15	14.2	14.2	13.6	12.10						
11.14	14.1	14.1	13.5	12.10	12.9	11.15				
11.13	13.4	13.4	12.10	12.9	12.8	11.15	11.14			
11.12	13.3	13.3	12.10	12.9	12.8	12.7	11.15	11.14	11.13	
10.20	14.2	14.2	13.6	12.10	11.15					
10.19	14.1	14.1	13.5	12.9	11.15	11.14	10.20			
10.18	13.4	13.4	12.8	11.15	11.14	11.13	10.20	10.19		
10.17	13.3	13.3	12.7	11.15	11.14	11.13	11.12	10.20	10.19	10.18
9.6.24	14.1	14.1	13.5	12.9	11.14	10.20	10.19			
9.6.23	13.4	13.4	12.8	11.13	10.20	10.19	10.18	9.6.24		
9.6.22	13.3	13.3	12.7	11.12	10.20	10.19	10.18	10.17	9.6.24	9.6.23

Upgrades of the RDS for PostgreSQL DB engine

Upgrading Amazon Aurora PostgreSQL DB clusters

Downtime for PostgreSQL

	Downtime: OS Update	Downtime: Minor Version Upgrade	Downtime: Major Version Upgrade
RDS Single-AZ	Yes, in minutes.	Yes, in minutes.	Yes. Full DB Upgrade. Use B/G deployment.
RDS Multi-AZ Instance (2 instances)	Yes, during failover. Perform maintenance on Standby, Promote Standby to Primary. Then, perform maintenance on old Primary, which becomes the new standby. If have Replica, OS Patch for Replica can be applied at separate time.	Yes, during restart. Primary & Standby upgrade together, so both unavailable. If have Replica, need to upgrade first before Primary.	Yes. Full DB Upgrade. Upgrade all instances. No failover during Major version upgrade. Use B/G deployment. B/G deployment doesn't support Cross-region read replica.
RDS Multi-AZ Cluster (3 instances)	Yes, during failover. Perform offline patch on 1st Reader, make sure other 2 instances are in-sync. Perform offline patch on 2nd Reader. Failover current Writer to Reader that was already patched. Make Writer a new Reader. If have Replica, OS Patch for Replica can be applied at separate time.	Yes, during failover. Apply update and change one of the Reader to be new Writer. Then, upgrade all other readers before upgrade old Writer instance and make it a new Reader. If have Replica, need to upgrade first before upgrade RDS PostgreSQL Multi-AZ cluster (AMVU handle that automatically).	Yes. Full DB Upgrade. Upgrade all instances. No failover during Major version upgrade. B/G Deployment is not supported. If have Replica, must delete and recreate the read replicas after the upgrade completes.
Aurora PostgreSQL	Yes, rolling upgrade that upgrade Reader then Writer.	Yes, unless ZDP is used. [1] Default: AMVU enabled, apply in cluster.	Yes, Full DB Upgrade. Use B/G deployment.
Aurora PostgreSQL Global Database	Yes, rolling upgrade in each region.	Yes, in primary region. Upgrade secondary clusters before Primary cluster. Add instance for headless cluster, remove after upgrade. ZDP [1] and AMVU [3] are not supported for Aurora Global Database.	Yes, Apply major version on Global DB Cluster level & upgrade all regions. Need to be on same major, minor & patch level for managed switchover & failover to work. [2] Use B/G deployment.

[1] Minor version upgrade and ZDP and Limitation of ZDP.
[2] Patch level compatibility for managed cross-region switchover & failover.
[3] AMVU for Aurora clusters.

Differences between Major and Minor version upgrades

	Minor Version Upgrade	Major Version Upgrade
Multi-AZ DB Instance	Simultaneously upgrades the primary and any standby instances.	Simultaneously upgrades the primary and any standby instances.
Multi-AZ DB Cluster	Amazon RDS Upgrades the Read Replica one at a time. Reader instance switches to new Writer instance. Amazon RDS then upgrades the old Writer instance (which is now a reader instance). This typically reduce the downtime of minor version upgrades to approximately 35 seconds.	Amazon RDS doesn't upgrade Multi-AZ DB Cluster read replicas. Manually delete and recreate read replicas after the primary upgrade completes.
In-Region Read Replica	First upgrade all of the read replicas before upgrading the source instance.	Replicas upgraded along with the primary DB instance.
Needs new custom parameter group for upgraded instance	No	Yes
Upgrades automatically (Provided RDS is configured with Auto Minor Version Upgrades)	Yes	No
Updates Database Data Files	No	Yes
Copies table statistics to upgraded instance	Yes	No for PostgreSQL 17 and below. Yes for PostgreSQL 18. Use vacuumdb --all --analyze-in-stages --missing-stats-only
Is always backward compatible	Yes	No
Extensions Requiring Upgrade	No	Yes

RDS PG Major version upgrades - Prerequisites

- Check Choosing a major version for an RDS PostgreSQL upgrade.
- Prepare version-compatible custom parameter group.
- Test an upgrade of your Production DB cluster to a new major version.
- Check for unsupported database classes, see Supported DB engines for DB instance classes.
- Check for unsupported usages (i.e. Prepared transactions, data type, invalid database).
- Extension – may need to drop before major version upgrade / upgrade extension before/ after upgrade.

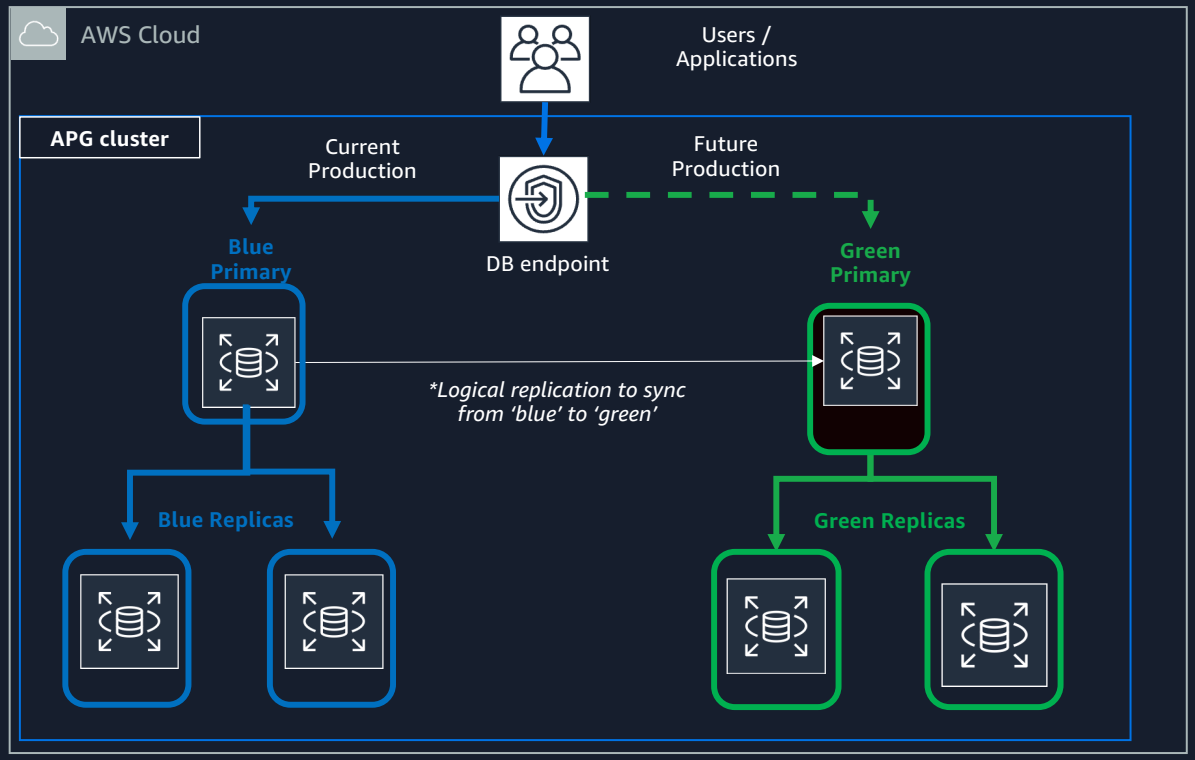
Database Upgrade Approaches: From In-Place to near Zero-Downtime

1. **In-Place Major Version Upgrade**
   - Direct upgrade path with automatic pre/post snapshots
   - Requires downtime during upgrade process
2. **Snapshot Migration**
   - Isolated testing environment
   - Risk-free validation of new version
3. **Database Migration Service (DMS)**
   - Minimal downtime approach
   - Active replication between different versions
   - Suitable for critical applications
4. **Logical Replication**
   - Native PostgreSQL replication or pglogical
   - Flexible version compatibility
   - Works with Aurora Fast Cloning
5. **Blue/Green Deployments – Recommended**
   - Managed switchover process
   - Minimal risk upgrade path

In-Place Major version upgrade

- Upgrades can skip major versions with multi major version upgrade
- The time needed to upgrade depends more on the number of database objects than the database size
- Snapshots taken before & after the upgrade if backup retention period > 0. Recommend to take manual snapshot.
- Statistics are not upgraded so ANALYZE is needed on the new version
- A new parameter group is needed for the new version – will be automatically created for you - default

RDS Blue/Green Deployments – Interim state



What it does:

- Creates a mirrored copy of the current production environment (blue) as the green environment (future production)
- Sets up logical replication between blue primary and green primary.
- Modify green, add/remove replicas, and test changes in green environment before switchover

Tasks in Green before switchover:

- Monitor replication lag.
- Run Analyze in Green (major version).

- RDS PostgreSQL - Default Physical Replication, use logical replication for major version in Green environment.
- Aurora PostgreSQL – Logical replication

Best Practices for Performing Database Engine Update

- Update DB engine, when a new release becomes available (recommended)
- Choose automatic minor upgrades vs. manual based on need
- Set an appropriate maintenance window
- Required updates are scheduled automatically during a maintenance window
- Physical Read replica can have different minor version than primary, but not major
- Use pglogical or native logical replication for minimum downtime major version upgrade
- Test engine update process in a pre-prod/testing environment (Blue/Green, Aurora Fast Clone or DMS)
- Run Analyze post major version upgrade.

Extended Major Version Support

- PostgreSQL 12 is supporting 3 years past community end-of-life

- Fully supported by Amazon Aurora and Amazon RDS during above period
  - Patches for critical CVEs/bug fixes in the DB engine
  - Ongoing infrastructure (hardware/OS) patches
  - Access to AWS Support
  - Availability SLA

https://aws.amazon.com/blogs/database/upgrade-strategies-for-amazon-aurora-postgresql-and-amazon-rds-for-postgresql-12/

Thank you!