QBurst

Seamless Migration from IBM Db2 to PostgreSQL

Project Overview

A global leader in HR services, on its digital transformation journey, sought to adopt cloud-native solutions to enhance operational efficiency. To align with their data and cloud strategy, they aimed to modernize their on-premises IBM Db2 databases to a fully managed cloud service with zero downtime. AWS was chosen for its robust infrastructure, security, compliance, and seamless migration capabilities.

Client Profile

Based in Japan, our client is a global leader in human resource services providing flexible HR solutions tailored to the evolving labor market.

Business Challenges

A significant challenge faced by the client is the high cost associated with IBM Db2 licenses. Additionally, with IBM planning to discontinue support for Db2 versions up to 10.5 within a year, there were both financial and operational implications that needed to be addressed to ensure uninterrupted database functionality.

Solution

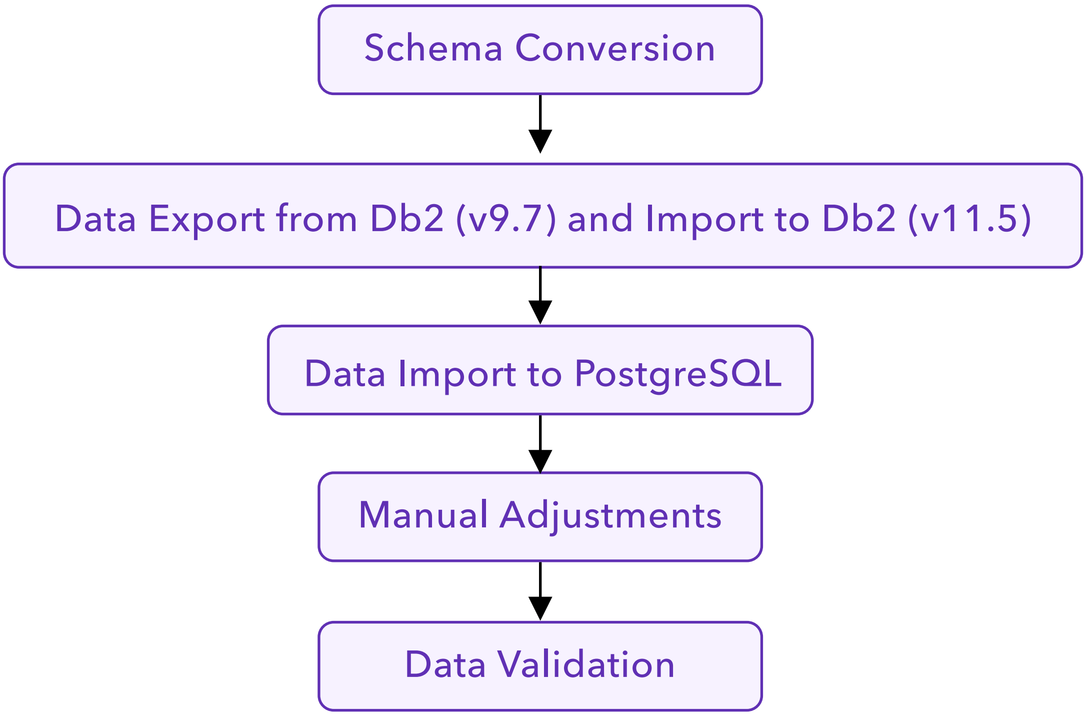
We proposed migrating the IBM Db2 database to AWS RDS for PostgreSQL, leveraging PostgreSQL's robust features and cost-effectiveness to address the client's needs.

Project Goal

Our objective was to successfully migrate approximately 500 GB of data across 600 tables from IBM Db2 (version 9.7) to AWS RDS for PostgreSQL (version 16.2) while ensuring:

- Seamless migration of schema, triggers, functions, and data.
- Preservation of schema compatibility and data integrity.
- Zero data loss throughout the process.

Migration Process



Schema Conversion

Data Export from Db2 (v9.7) and Import to Db2 (v11.5)

Data Import to PostgreSQL

Manual Adjustments

Data Validation

1. Schema Conversion

- Created a dummy Db2 database in UTF-8 encoding and restored the original Db2 DDL (Database Definition Language) to this database.
- Used AWS Schema Conversion Tool (SCT) tool to migrate the Db2 schema to PostgreSQL.

2. Export and Import of ~500GB Data from DB2 9.7 to DB2 11.5

- Created a database with suitable encoding in RDS Db2 to restore the DDL and data in Db2 11.5 since a direct migration to PostgreSQL 16.2 was not possible.
- Db2 data was exported table-by-table using command-line tools and Bash scripts.
- Employed utility commands to extract data dumps for easier import.
- Shell scripts were developed to process data in batches, optimizing migration time and efficiently managing large transactions.

3. Data Import to PostgreSQL

- Migrated the data from Db2 to PostgreSQL using AWS DMS.
- Configured DMS to batch-process the migration to handle large data volumes effectively.
- Disabled triggers in the PostgreSQL database during the migration to prevent issues with automatic row creation.

4. Manual Adjustments

- Db2 triggers and functions were not converted directly to PostgreSQL, requiring manual modification and validation to ensure proper functionality.
- Resolved schema differences, especially those involving sequences and encoding-related errors.

5. Data Validation

- Performed data validation using AWS DMS to ensure data integrity.
- Encountered mismatches in some tables due to Chinese characters and encoding differences, which were resolved by character length-based validations.

Project Challenges and Resolutions

1. Encoding Mismatches

- We initially attempted to restore the Db2 RDS database using UTF-8 encoding, which resulted in rejected rows during the data import process due to encoding mismatches.
    - To resolve this, we recreated the Db2 RDS database using IBM-943 encoding, ensuring compatibility and enabling a successful data import.

2. Database Dump and Restore

- The initial Db2 database dump (binary image file) failed to restore in the newer Db2 version (11.5) due to compatibility and encoding issues.
    - A new database was created with the required encoding to address compatibility concerns. The DDL was restored using IBM Data Studio.

3. Import Optimization

- The initial use of Db2 Import commands caused performance bottlenecks, generating extensive logs which caused the data import to crash.
    - We converted scripts using Db2 Import commands to Load commands for faster imports and reduced log sizes.

4. Data Validation Issues

- During post-migration data validation, AWS DMS validation failed due to encoding differences between the source database (Db2 with IBM-943) and the target database (PostgreSQL with UTF-8).
    - We bypassed AWS DMS validation for encoding mismatches and implemented a custom validation approach by comparing data based on character length instead of direct content matching.

Tools Used

IBM Data Studio

AWS Schema Conversion Tool (SCT)

AWS Database Migration Service (DMS)

DBeaver

Conclusion

The migration of IBM Db2 9.7 to AWS RDS for PostgreSQL 16.2 was successfully executed, ensuring full schema and data compatibility. The process required resolving encoding mismatches, modifying triggers manually, and validating the migrated data effectively. From this experience, several learnings emerged that would optimize future migrations:

1. Prepare for Encoding Challenges

Use consistent encoding formats during setup to minimize compatibility issues.

2. Batch Processing

Utilize batch scripts to manage large data sets effectively and reduce migration time.

3. Batch Processing

Anticipate the need for manual schema and trigger adjustments as automated tools may not fully cover all scenarios.

4. Validation Strategy

Employ a robust validation strategy, particularly for special characters and non-UTF-8 encoded data.