



EDB™

CUSTOMER
SUCCESS
OFFICE HOURS

Oracle Migration Best Practices

Date: August 31, 2022

Presented by: Ron Privett | Customer Success Solution Engineer

AGENDA

- Intro to Migration
- Methods of Migration
- EPAS vs Oracle
- Best Practices
- Migration Portal
- Migration Portal Demo
- Q&A

INTRO TO MIGRATION

INTRODUCTION TO MIGRATION

- **What is Database Migration?**
 - Process of moving Schema, Data, and Stored Procedures from one database platform to another
 - Migration to Postgres will involve several steps: Decide, Plan, Migrate, Optimize, and Cutover
- **Why migrate from Oracle to Postgres?**
 - Cost – License cost, Feature use cost, etc...
 - Flexibility – Postgres has open-source licensing, Available from several vendors, reduced risk of vendor lock-in
 - Customization – many open-source extensions available



5 EASY STEPS ON THE MIGRATION JOURNEY



1. **Decide** – Business Case, Organizational alignment
2. **Plan** – Open-source vs proprietary, Cloud vs On-prem, Applications, Replatform or Restructure, Define Requirements, Estimate Effort, Develop Rollback plan
3. **Migrate** – Schema, Data, APIs, Embedded SQL, Applications, Reports, DBA Scripts/Utilities
4. **Optimize** – Validate Data/Function/Performance, DB Tuning, Query Tuning, HA / DR, Security, Authentication / Authorization
5. **Cutover** – Sync in-flight data, Rollback setup, Go/No-Go

Source: [The database migration "journey"](#)

METHODS OF MIGRATION

METHODS OF MIGRATION: TOOLS

- Open-Source
 - [Ora2pg](#)
 - [Ora_migrator](#)
 - [Orafce](#)
- [EDB Migration Portal](#)
 - Free, Sign-in required
 - Assessment, conversion, reporting
- [EDB Migration Tool Kit](#)
 - Command line tool to migrate data
- Replication Server
- LiveCompare
- EDB Postgres Distributed
- Migration Assessment Service
 - Professional Services engagement
 - For complex migrations, or short-staffed teams

MIGRATION METHODS

1. Snapshot – Everything at once
 2. Snapshot in parallel – chunks of data moved over time
 3. Change Data Capture (CDC) or replication – data continuously loaded
-
- Options 1 & 2 require an outage or downtime to move all data and cutover
 - Option 3 might be best for environments where smaller downtime windows are required

MIGRATION METHODS: SNAPSHOT

- Pros

- All data moved at one time
- No Data Type challenges (LoBs)
- After snapshot, applications can start accessing the target database
- No special configuration required, Easy to manage
- READ ONLY users may access source database (in some cases)

- Cons

- The application will be down during the snapshot
- Interruptions require restart

MIGRATION METHODS: SNAPSHOT IN PARALLEL

- **Pros**

- Data moved at one time (parallel = less downtime)
- Data moved in parallel - table by table or a large table in small sets
- Can be controlled via the EDB Migration Toolkit

- **Cons**

- Application downtime required (less compared to snapshot approach)
- For large tables broken into small sets, primary key or unique row identifiers are mandatory
- Scripting/Programming required to adjust the parallel approach
- Interruptions require restart

MIGRATION METHODS: CHANGE DATA CAPTURE (CDC)

● Pros

- Data loaded continuously to target database (after initial snapshot)
- User can access source database while data loads to target database
- Data sync control (if interrupted, it can be resumed)
- Partial replication (Set of tables can be replicated)
- Partial to small application downtime (switchover time)

● Cons

- Need replication software
- In trigger-based CDC, there could be a slight performance overhead
- No Large Objects support
- Partial to small application downtime (switchover time)
- Only commercial/free to use tools available, no open source

TOOLS FOR DATA MIGRATION

	Snapshot	Snapshot in Parallel	Change Data Capture	Target
EDB Replication Server	●	●	●	Postgres Advanced Server/PostgreSQL
EDB MTK	●	●	●	Postgres Advanced Server/PostgreSQL
AWS SCT	●	●	●	PostgreSQL
Ora2Pg	●	●	●	PostgreSQL
Ora_FDW (extension)	●	●	●	PostgreSQL
Dblink_ora(extension)	●	●	●	Postgres Advanced Server

EPAS VS ORACLE

EPAS COMPATIBILITY WITH ORACLE

- Areas of compatibility?
 - Oracle-specific and syntax-compatible database object types
 - Oracle-specific data types
 - Oracle-specific SQL extensions
 - Oracle PL/SQL support as a built-in native procedural language
 - Oracle data dictionary views (i.e., ALL, DBA, USER_ views)
 - Oracle built-in PL/SQL packages
- Benefits of compatibility?
 - Reduce time & effort to migrate from Oracle (compatibility = less conversion/rewrite)
 - Eases transition for Oracle DBA and developers (familiar database and tools)

EPAS & ORACLE: TERMINOLOGY

Oracle	EDB Postgres Advanced Server
Table or index	Table, index, or relation
Row	Row or tuple
Column	Column or attribute
Data block	Page – When block is on disk
Page	Buffer – When block is in memory

EPAS & ORACLE: GENERAL CAPABILITIES

General Capabilities	Oracle	EPAS
Design origin	Commercial implementation based on IBM's original research for System R	Academic implementation (UC Berkeley) based on IBM's original research for System R
Continuous development	Since 1979	PostgreSQL development started in 1986. EPAS development started in 2004. EPAS is based on PostgreSQL and continuously merged.
Object relational database	Yes	Yes
Processing architecture	Process & thread based	Process based
Full ACID compliance	Yes	Yes

EPAS & ORACLE: GENERAL CAPABILITIES

General Capabilities	Oracle	EPAS
Multiversion concurrency control	Yes	Yes
Multitenant architecture	Yes	Yes
Automatic workload management	Yes	No
Enterprise database management	Oracle Enterprise Manager	EDB Postgres Enterprise Manager
Multicore support	Yes	Yes
Write-ahead durability	Redo logs	Write-ahead log
Disk-read buffering	Yes	Yes

EPAS & ORACLE: CAPACITIES

Capacities	Oracle	EPAS
Max. database size	Unlimited	Unlimited
Max. table size	4 GB x Block Size	32 TB
Max. row size	4 TB	1.6 TB
Max. field size	For BLOB: (4 GB - 1) x DB_BLOCK_SIZE initialization parameter	1 GB
Max. rows per table	Unlimited	Unlimited
Max. columns per table	1000	250 - 1600 depending on column types
Max. indexes per table	Unlimited	Unlimited

EPAS & ORACLE: MORE DETAILS

- EDB's [Migration Overview](#) documentation provides an exhaustive comparison of the following key areas:
 - [Database features](#) (tables & partitioning, data types, SQL capabilities, SQL extensions)
 - [Database operations-related capabilities](#) (high availability, performance/scalability, security, encryption, integration, management, catalog views, deployment options)
 - [Application development capabilities](#) (API support, object-oriented capabilities, package support)
 - [Nonrelational data support](#) (spatial, JSON, key value store, XML, Hadoop, Mongo, cube/rollup/grouping sets, full text search)

EPAS & ORACLE: NOTABLE DIFFERENCES

Oracle	EPAS
MERGE	No. Postgres UPSERT statements can be used but have different syntax from MERGE. MERGE syntax planned for PostgreSQL 15
Pipelined functions	No. Pipelined functions are used for table functions. Table functions can be implemented in Postgres via SETOF returning functions. In Postgres, data is returned only after the function completes.
Empty string = NULL	No. Empty string = !NULL
Performs many implicit data type conversions such as a number to a string	Partial. Many data types need to be explicitly cast to the other data type or an error occurs.

EPAS & ORACLE: NOTABLE DIFFERENCES

- **Constraints**

- Oracle allows users to disable and enable constraints as often as you want
- In Postgres, constraints are created as deferrable, and the SET CONSTRAINTS command is used to defer them. If the Oracle constraint is not deferrable, it will need to be dropped and re-created as deferrable, though it is sometimes possible to alter the constraint without having to drop it.

- **DELETE statements**

- The FROM clause is required with DELETE statements in Postgres but not in Oracle.

- More details found here: Blog: [The Complete Oracle to Postgres Migration Guide](#)

BEST PRACTICES

MIGRATION BEST PRACTICES

- Follow each step in the migration journey: Plan, Identify functional & non-functional requirements, Complete the solution design early to understand IT resource needs
- Get Developers & SysAdmins involved early to obtain feedback on the plan
- Utilize Development & Test environments to resolve issues prior to production migration.
- Test using workloads that simulate production usage.

MIGRATION BEST PRACTICES

- Obtain performance benchmarks for key processes under old environment to compare with the new environment
- Engage with support early on performance issues
- Develop a comprehensive rollback plan, especially for complex migrations
- If your Migration Portal Compatibility score is low, consider the help of EDBs Professional Services

MIGRATION PORTAL

MIGRATION PORTAL

- Migration Portal helps you:
 - Assess your source schema and understand what is and is not compatible with Postgres
 - Determine if how to convert objects that fail to convert automatically using an in-app wiki and knowledge base
 - Export and launch EPAS with 1-click using pgAdmin, PSQL client, or deploy directly to EPAS
 - Review your compatibility score using a report that details which objects converted without change, with modification, or require a manual update
- <https://migration.enterprisedb.com/>

EXTRACTING DDL FROM ORACLE



- Before starting with the Migration Portal, you need to extract the schema from your Oracle database. You can use one of 2 methods:
 - [EDB DDL Extractor](#) (recommended method)
 - [Oracle Data Pump utilities](#)
- When you start a new Migration Project in the portal, you can download the SQL file named 'edb_ddl_extractor.sql'
- The script uses Oracle's DBMS_METADATA built-in package to extract DDLs for different schemas you specify.

MIGRATION PORTAL DEMO

REFERENCES

- Blog: [The Complete Oracle to Postgres Migration Guide](#)
- Docs: [Factors to consider when migrating](#)
- Docs: [Comparison of EDB Postgres Advance Server with Oracle](#)

- Webinar: [Oracle Migrations Made Easy](#)
- Webinar: [Break free from Oracle: Proven strategies for migrating databases to Postgres](#)
- Success Story: [Metasphere Unlocks New Opportunities by Choosing Postgres and EDB](#)

Oracle Migration Webinar Series



- [Sep 13](#): Why customers are migrating from Oracle to Postgres and Challenges & considerations before embarking on a database migration
- [Sep 20](#): Tools, capabilities and resources EDB can offer to make Oracle-to-Postgres database migrations easier
- [Sep 27](#): Oracle Migration Success Stories

[REGISTER NOW](#)

Q&A

THANKS