# Oracle to PostgreSQL Migration Workflow

## 1. Assessment & Planning

* • Identify applications and databases using Oracle.
* • Evaluate data size, schemas, and PL/SQL dependencies.
* • Determine compatibility gaps between Oracle and PostgreSQL.
* • Plan downtime window, rollback strategy, and cutover approach.

## 2. Schema Conversion

* • Extract schema from Oracle (DDL).
* • Use tools like ora2pg to convert schema to PostgreSQL-compatible format.
* • Resolve datatype differences (NUMBER vs NUMERIC, CLOB vs TEXT, etc.).
* • Test converted schema in a dev/staging PostgreSQL instance.

## 3. Data Migration

* • Choose appropriate migration tool (ora2pg, AWS DMS, custom scripts).
* • Export data from Oracle and load into PostgreSQL.
* • Perform batch migration to minimize downtime (if required).
* • Validate row counts and data integrity.

## 4. Code & Application Refactoring

* • Convert PL/SQL to PL/pgSQL or equivalent logic.
* • Update application connection strings, drivers, and queries.
* • Test performance and tune queries for PostgreSQL optimizer.

## 5. Testing & Validation

* • Functional Testing: Verify application functionality.
* • Performance Testing: Compare query performance before/after migration.
* • User Acceptance Testing (UAT): Involve business users to sign-off.

## 6. Cutover & Go-Live

* • Schedule final data sync.
* • Perform switchover during approved downtime window.
* • Monitor application behavior post-cutover.

## 7. Post-Migration Activities

* • Monitor database performance, connections, and errors.
* • Enable backups, monitoring, and alerting.
* • Train teams on PostgreSQL administration and troubleshooting.

# Enhanced Workflow Diagram with Decision Points

