Database Migration & Modernization

Comprehensive 8-week training program for transitioning from Oracle, DB2, Informix, and SQL Server to open source database platforms. Master enterprise-scale database modernization from assessment to production cutover.

Why Database Modernization Matters

Rising Costs

Enterprise database licensing fees continue to escalate, often consuming 20-40% of IT budgets. Oracle's per-core pricing model becomes increasingly expensive as organizations scale their infrastructure.

Vendor Lock-in

Proprietary systems create dependencies that limit flexibility and innovation. Organizations struggle to adapt quickly to changing business requirements when constrained by vendor-specific technologies.

Cloud-First Strategy

Modern cloud platforms offer superior scalability, managed services, and cost optimization. Open source databases integrate seamlessly with containerized architectures and DevOps workflows.

Target Audience & Prerequisites

Database Administrators

DBAs managing Oracle, DB2, MySQL, or SQL Server environments seeking to expand expertise in open source alternatives and cloud-native solutions.

Application Developers

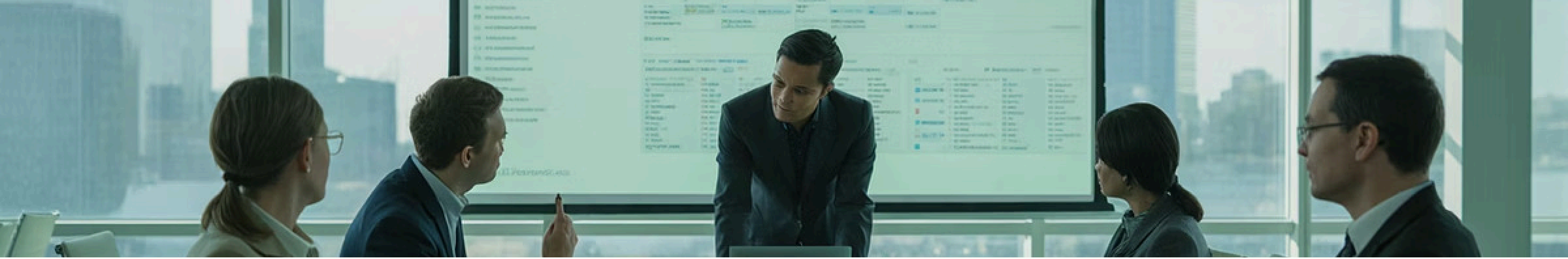
Developers working with Oracle PL/SQL, stored procedures, and database-centric applications who need to adapt code for new platforms.

IT Leadership

IT managers and project leaders responsible for strategic technology decisions and modernization roadmaps in enterprise environments.

Cloud Teams

Technical teams preparing for cloud transformation projects, hybrid deployments, and digital modernization initiatives.



Learning Objectives

1 Master Migration Planning

Develop comprehensive migration strategies from assessment through production cutover, including risk mitigation and rollback procedures.

2 Hands-on Tool Expertise

Gain practical experience with industry-standard migration tools including ora2pg, pgloader, EDB Migration Toolkit, and Striim for various scenarios.

3 Code Transformation Skills

Convert complex PL/SQL procedures, functions, and packages to modern alternatives while maintaining functionality and performance.

4 Zero-Downtime Strategies

Implement enterprise-grade cutover approaches that minimize business disruption and ensure data consistency throughout the migration process.

Training Format & Structure

Interactive Learning

- Weekly 2-3 hour online sessions
- Live demonstrations and Q&A
- Expert instructor guidance
- Peer collaboration opportunities

Practical Experience

- Hands-on laboratory environment
- Real database scenarios
- Migration tool practice
- Troubleshooting exercises

Applied Learning

- Weekly mini-projects
- Progressive skill building
- Case study development
- Final presentation capstone

Week 1: Assessment & Migration Planning

Understanding Migration Drivers

Explore the business and technical factors driving database modernization initiatives. Analyze cost structures, licensing models, and strategic technology alignment across different migration scenarios.

01

Inventory Analysis

Document database sizes, schema complexity, application dependencies, and integration points

02

Cost Evaluation

Calculate current licensing costs, maintenance fees, and projected savings from open source alternatives

03

Risk Assessment

Identify technical risks, compatibility issues, and business continuity requirements

Week 2: Target Architecture Design

Open Source Database Ecosystem

Compare leading open source platforms and their enterprise capabilities. Understand the strengths and use cases for PostgreSQL, MySQL, MariaDB, and MongoDB in different architectural contexts.

PostgreSQL

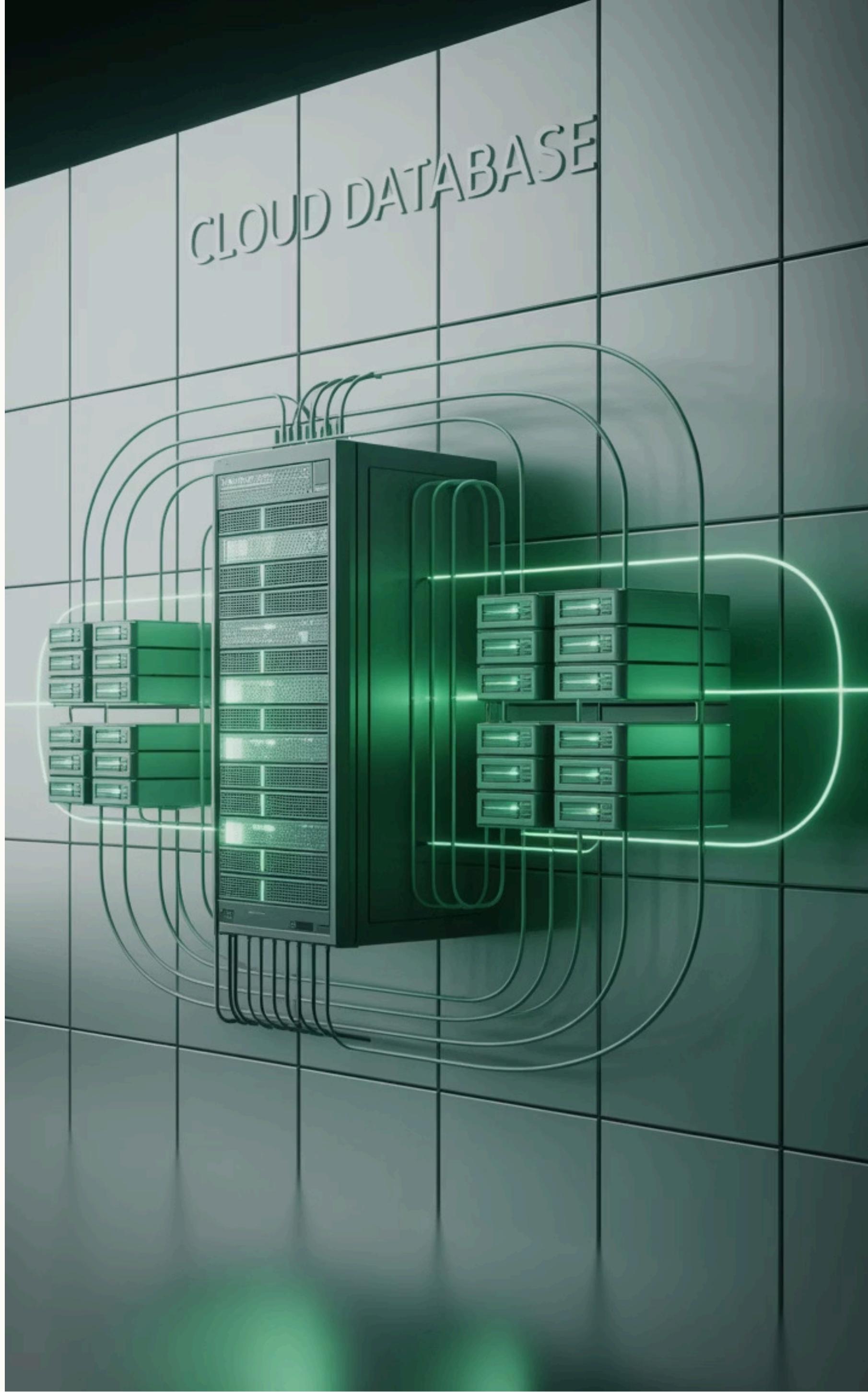
Advanced SQL features, JSON support, extensibility, and ACID compliance make it ideal for complex enterprise applications requiring robust data integrity.

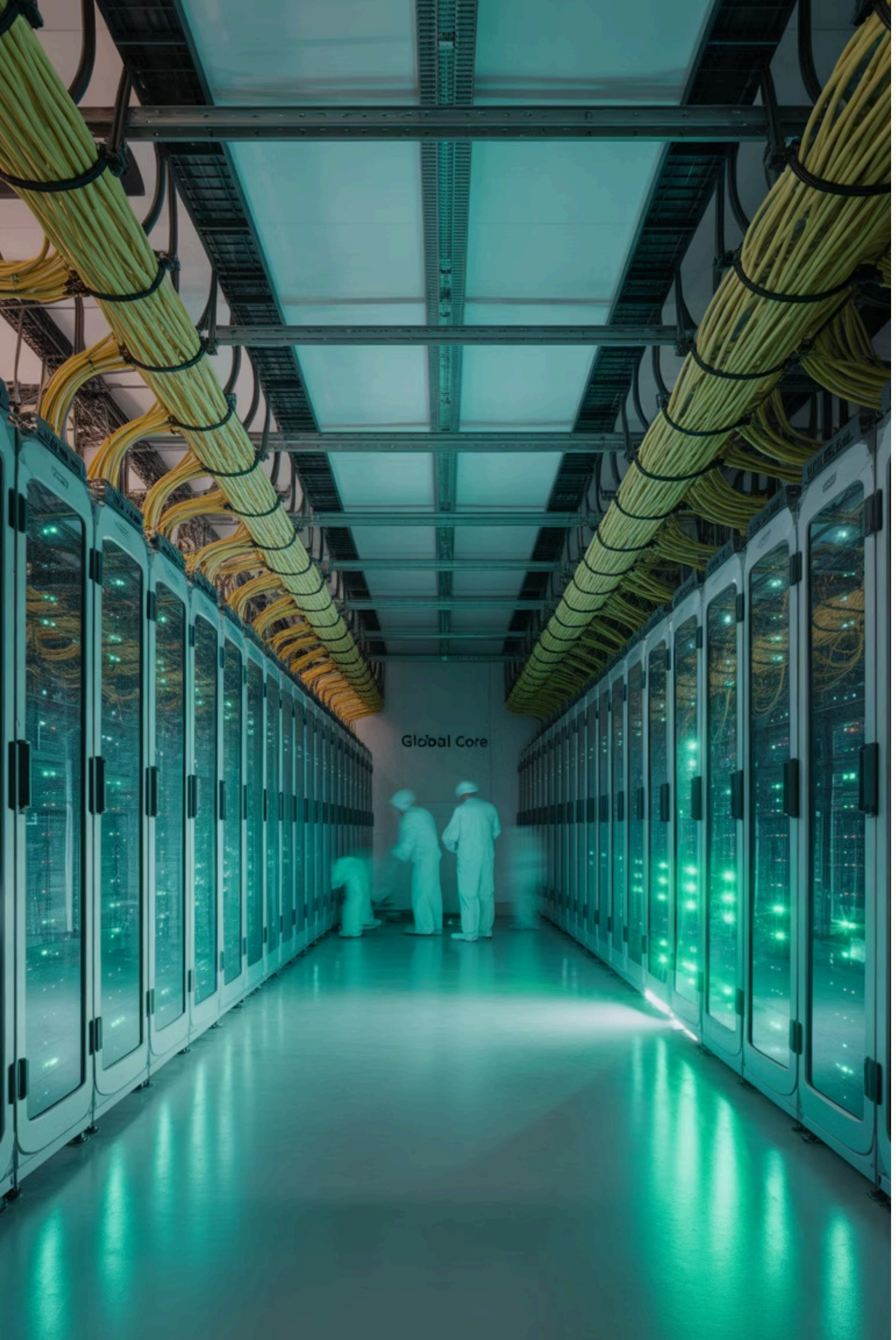
MySQL/MariaDB

High-performance OLTP workloads, web applications, and read-heavy scenarios benefit from optimized storage engines and clustering capabilities.

Cloud Options

Managed services like Cloud SQL, AlloyDB, and Aurora provide enterprise features without operational overhead, enabling faster migration paths.





High Availability & Disaster Recovery

1

Replication Setup

Configure streaming replication, logical replication, and read replicas for load distribution and failover protection.

2

Backup Strategies

Implement automated backup schedules, point-in-time recovery, and cross-region backup replication for comprehensive data protection.

3

Failover Testing

Validate failover procedures, RTO/RPO objectives, and automated recovery mechanisms before production deployment.

Week 3: Data Migration Strategies

Migration Approaches

Master different data migration methodologies based on system size, downtime requirements, and business constraints. Learn when to use full dump/load versus incremental approaches.

Full Migration

Complete database export and import, suitable for smaller systems or when extended downtime is acceptable. Simplest approach with clear cutover point.

Incremental Migration

Change Data Capture (CDC) enables continuous synchronization, minimizing downtime for large, mission-critical systems requiring near-zero disruption.

Migration Tools Comparison

Tool	Source DB	Target DB	Key Features
ora2pg	Oracle	PostgreSQL	Schema + Data + Code
pgloader	Multiple	PostgreSQL	High-speed loading
Striim	Multiple	Multiple	Real-time CDC
EDB Toolkit	Oracle	PostgreSQL	Enterprise features

Handling Large-Scale Data Migration

Partitioning & Batch Processing

Implement proven strategies for migrating terabyte-scale databases while maintaining system performance and data consistency. Learn parallel processing techniques and resource optimization.

1 **Pre-Migration**

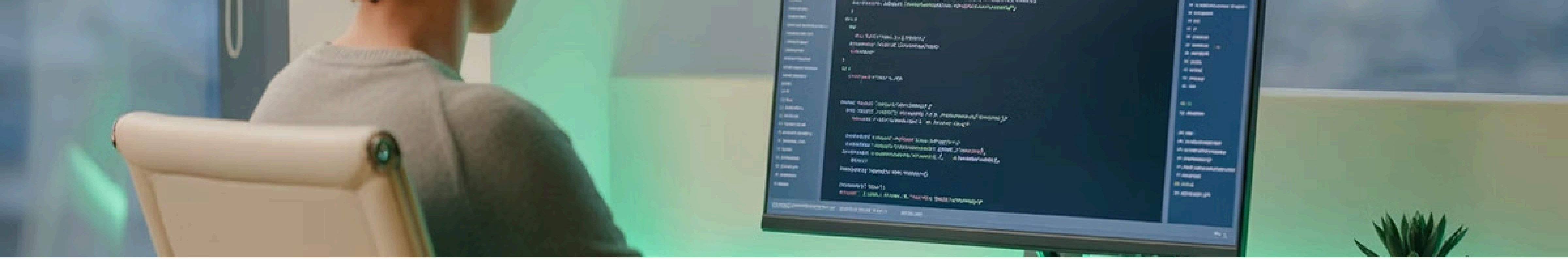
Schema validation, data profiling, and test migration execution to identify potential issues and optimize transfer performance.

2 **Batch Processing**

Partition large tables by date ranges or key values, enabling parallel migration streams and progress monitoring capabilities.

3 **Validation**

Automated row count verification, checksum comparisons, and sample data validation to ensure migration accuracy and completeness.



Week 4: Code Migration Fundamentals

PL/SQL to PL/pgSQL Transformation

Navigate the complexities of converting Oracle stored procedures, functions, and packages to PostgreSQL equivalents. Understand syntax differences, feature mapping, and performance considerations.

Automatic Conversion

- Basic procedure structure
- Simple data types
- Standard SQL constructs
- Common built-in functions

Manual Rewriting Required

- OUT parameter handling
- Package structures
- Dynamic SQL execution
- Oracle-specific functions

Advanced Code Conversion Techniques

1

Exception Handling

Convert Oracle's exception handling mechanisms to PostgreSQL's error management model, including custom exception types and error propagation.

2

Cursor Management

Transform cursor-based processing logic, accounting for differences in cursor syntax, loop structures, and memory management between platforms.

3

Trigger Migration

Adapt complex trigger logic, handling differences in trigger timing, NEW/OLD references, and multi-table trigger scenarios.

```
-- Oracle PL/SQL Example
CREATE OR REPLACE PROCEDURE update_salary(
 emp_id IN NUMBER,
 new_salary IN NUMBER,
 result OUT VARCHAR2
) IS
BEGIN
 UPDATE employees SET salary = new_salary
 WHERE employee_id = emp_id;
 result := 'Updated successfully';
EXCEPTION
 WHEN NO_DATA_FOUND THEN
 result := 'Employee not found';
END;
```

Week 5: Application Layer Migration

Driver & Connection Changes

Manage the transition from Oracle JDBC drivers to PostgreSQL or MySQL equivalents. Update connection strings, handle driver-specific behaviors, and optimize connection pooling configurations.

Driver Updates

Replace database drivers and update connection parameters

Configuration

Adjust connection pooling, timeout, and transaction settings

Testing

Validate functionality, performance, and error handling

Code Adaptation

Modify SQL queries and handle database-specific syntax differences

ORM Framework Adaptation

Hibernate & Framework Migration

Adapt popular ORM frameworks for new database targets. Address dialect changes, annotation updates, and configuration modifications required for seamless application functionality.

Hibernate Configuration

Update hibernate.dialect settings, connection properties, and database-specific optimizations for PostgreSQL or MySQL targets.

Entity Mapping

Modify JPA annotations, handle sequence generation differences, and adapt column type mappings between database platforms.

Query Optimization

Rewrite native SQL queries, optimize HQL statements, and leverage database-specific performance features in the new environment.

Week 6: Testing & Validation

Data Consistency Verification

Implement comprehensive testing methodologies to ensure data integrity throughout the migration process. Learn automated validation techniques and performance benchmarking approaches.

Data Validation

Automated row counts, checksum verification, and statistical sampling to confirm complete and accurate data transfer.

Application Testing

Unit tests, integration tests, and regression testing to validate application functionality against the new database platform.

Performance Benchmarking

TPC-C, JMeter, and sysbench testing to compare performance characteristics and identify optimization opportunities.

Cutover Strategy Planning

Migration Approaches Comparison

Parallel Run

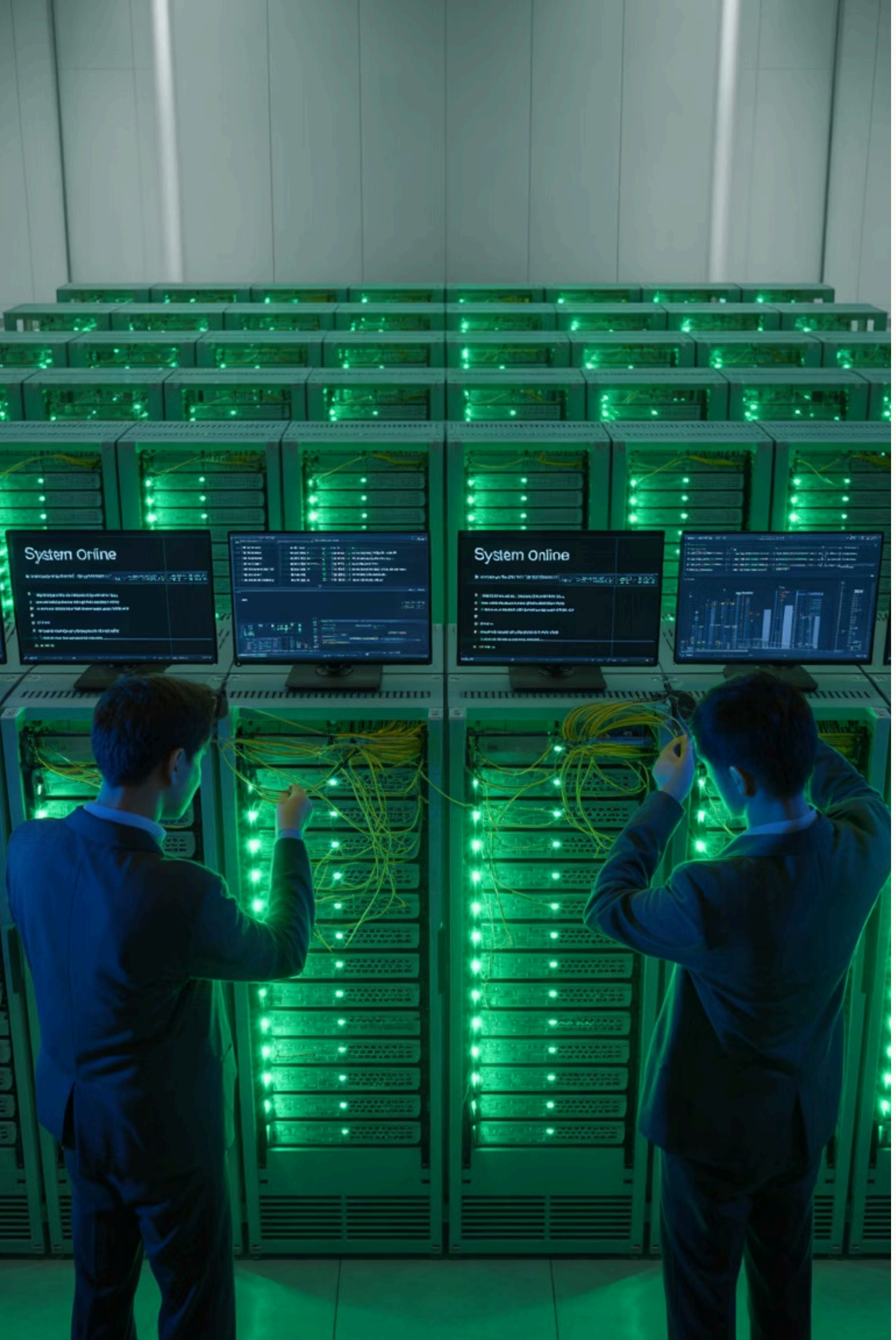
Operate both systems simultaneously, allowing gradual transition and easy rollback. Requires more resources but minimizes risk through extended validation period.

- Lower risk profile
- Extended validation time
- Higher resource requirements
- Complex data synchronization

Big Bang Approach

Complete cutover during planned maintenance window. Faster resource recovery but requires thorough preparation and testing to minimize failure risk.

- Faster completion
- Lower ongoing costs
- Higher risk concentration
- Requires comprehensive testing



Week 7: Go-Live Strategy

Zero-Downtime Migration Techniques

Master advanced cutover strategies that minimize business disruption. Learn real-time data synchronization, application failover procedures, and rollback mechanisms for mission-critical systems.

1

Pre-Cutover Sync

Establish continuous data synchronization using CDC tools, ensuring target database remains current with production changes.

2

Application Switchover

Coordinate application connection redirection, session management, and transaction completion during the cutover window.

3

Validation & Monitoring

Implement real-time monitoring, automated health checks, and rapid rollback procedures to ensure system stability.

Monitoring & Performance Optimization

Post-Migration Operations

Establish comprehensive monitoring and alerting systems for the new database environment. Compare performance metrics with baseline measurements and implement optimization strategies.

System Monitoring

Implement Prometheus, Grafana, or Zabbix for comprehensive database performance monitoring, alerting, and trending analysis.

Query Optimization

Utilize pg_stat_statements, slow query logs, and execution plan analysis to identify and resolve performance bottlenecks.

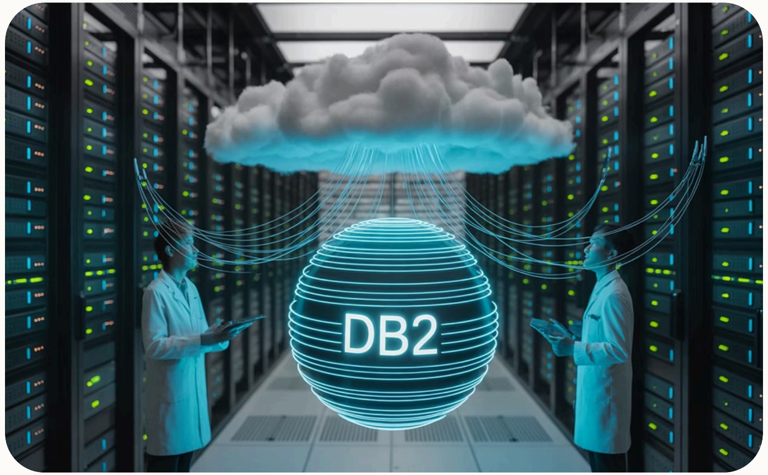
Security & Access

Configure role-based access control, SSL encryption, and audit logging to maintain enterprise security standards.

Week 8: Best Practices & Case Studies

Real-World Migration Examples

Analyze successful enterprise migrations with detailed case studies covering technical challenges, solutions implemented, and lessons learned from production deployments.



Oracle → PostgreSQL

Fortune 500 financial services company migrated 50TB Oracle data warehouse to PostgreSQL using ora2pg, achieving 60% cost reduction while improving query performance by 30%.

DB2 → Cloud SQL

Manufacturing company transitioned legacy DB2 systems to Google Cloud SQL, enabling global scalability and reducing maintenance overhead by 75% through managed services.

Your Migration Journey Starts Here

Next Steps & Continued Learning

Complete your capstone project by presenting a comprehensive migration plan for your organization's specific database environment. Apply the knowledge, tools, and strategies learned throughout this 8-week program.

01

Assessment Completion

Finalize your organization's database assessment and migration readiness evaluation

02

Tool Selection

Choose appropriate migration tools and create detailed project timeline

03

Pilot Implementation

Execute proof-of-concept migration with non-critical systems

04

Production Planning

Develop comprehensive production migration strategy with stakeholder buy-in

> Transform your organization's data infrastructure with confidence. Join the growing community of enterprises successfully modernizing their database platforms for cloud-native, cost-effective, and scalable solutions.