Migrating Your Databases to AWS

Akanksha Sheoran

Database GTM Specialist

Agenda

- Data Trends
- Fully-managed Database Service
- Purpose-built Databases
- Database Freedom Program
- Migration: Strategies, Tools and Partners
- Tungsten Network: Migration Journey to AWS

Unprecedented Data Growth Drives Innovation

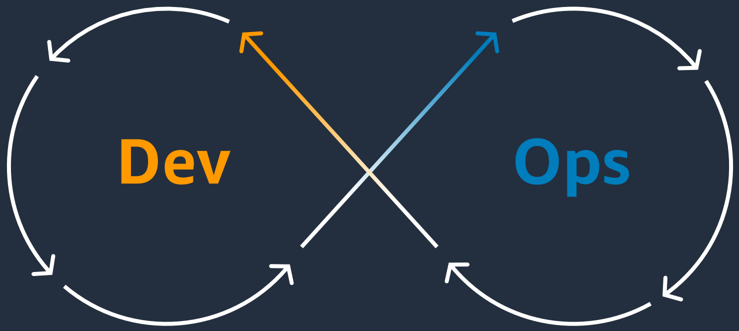
Explosion of data

Data grows 10x every 5 years driven by network-connected smart devices

Micro-services changes data and analytics requirements

Micro-services architecture decreases need for one-size-fits-all databases and increases need for real-time monitoring and analytics

Rapid rate of change driven by DevOps



Transition from IT to DevOps increases rate of change

To get more value from their data, customers are…

Moving to fully managed database services

Building modern applications with purpose-built databases

Breaking free from legacy databases

To get more value from their data, customers are..

Moving to fully managed database services

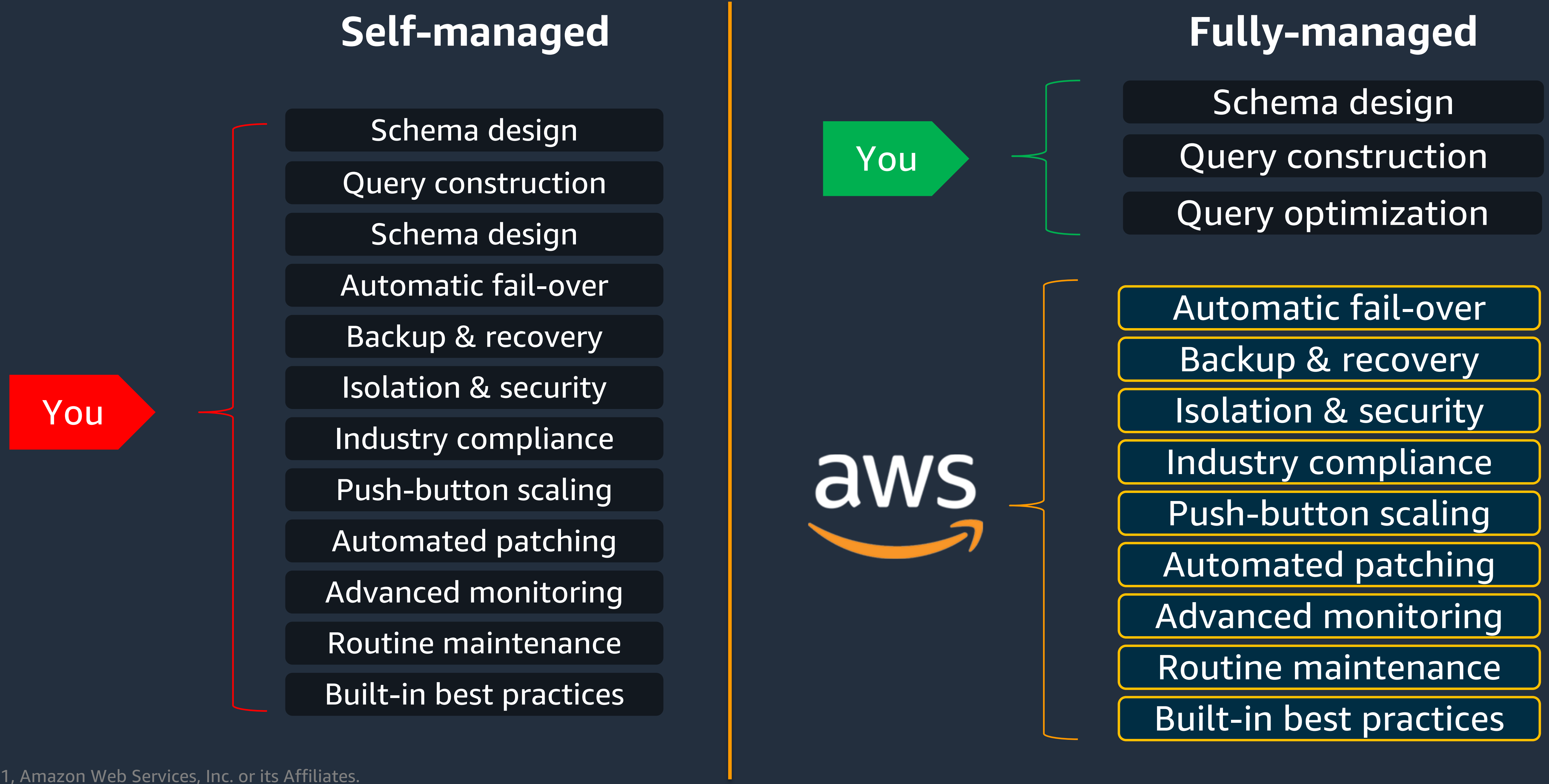
Building modern applications with purpose-built databases

Breaking free from legacy databases

Self-managed databases are complex, time consuming, and expensive

- → Hardware & software installation, configuration, patching, backups
- → Performance and high availability issues
- → Capacity planning and scaling clusters for compute and storage
- → Security and compliance

Fully-managed services on AWS



Self-managed

You

- Schema design
- Query construction
- Schema design
- Automatic fail-over
- Backup & recovery
- Isolation & security
- Industry compliance
- Push-button scaling
- Automated patching
- Advanced monitoring
- Routine maintenance
- Built-in best practices

Fully-managed

You

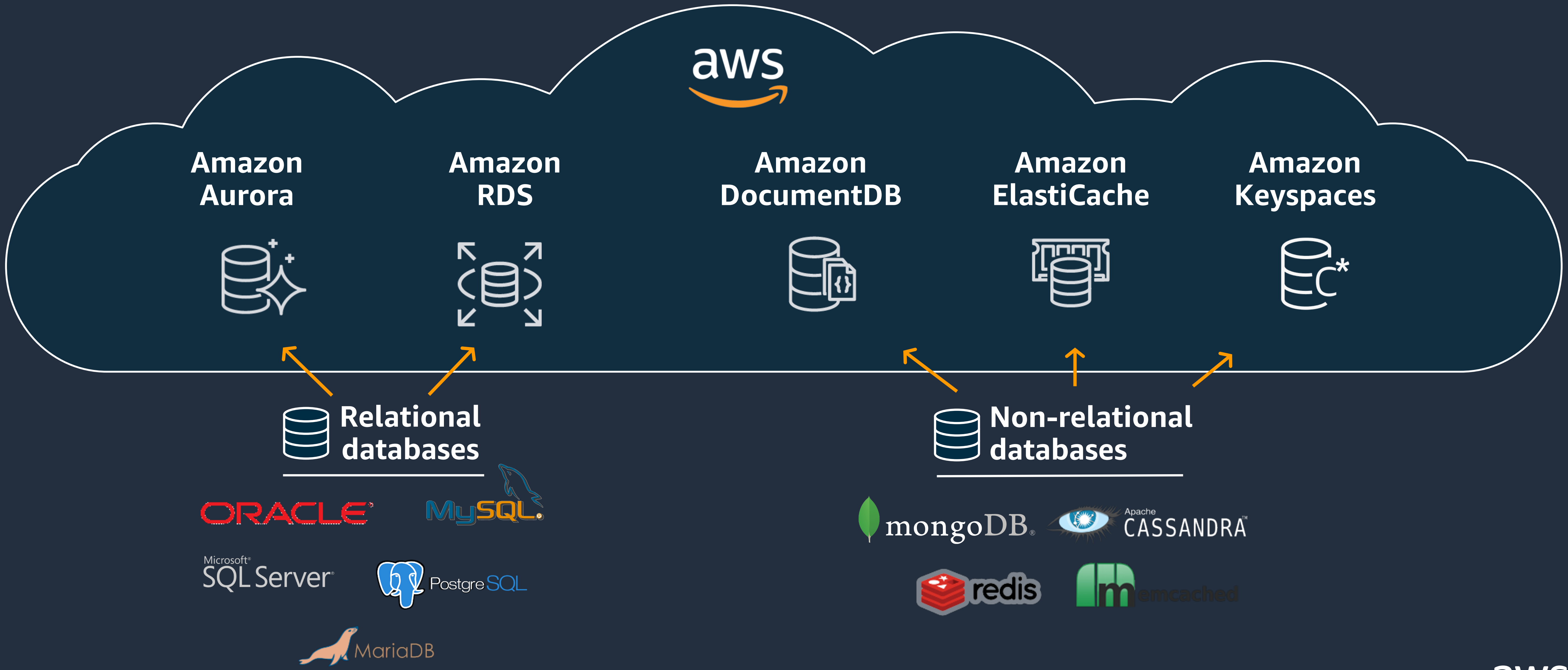
- Schema design
- Query construction
- Query optimization

aws

- Automatic fail-over
- Backup & recovery
- Isolation & security
- Industry compliance
- Push-button scaling
- Automated patching
- Advanced monitoring
- Routine maintenance
- Built-in best practices

Move to fully-managed databases

Migrate on-premises or self-managed databases to fully managed services



Amazon Aurora

Amazon RDS

Amazon DocumentDB

Amazon ElastiCache

Amazon Keyspaces

Relational databases

Non-relational databases

instacart

Instacart offers its customers a new way to order same-day delivery from local grocery stores online

Challenge:

Building and maintaining production database was challenging on PostgreSQL for a fast-growing startup. They wanted their engineering team to focus on building applications vs. managing infrastructure

Solution:

The company turned to AWS to run its database on Amazon RDS for PostgreSQL. RDS gives them an out-of-the-box, API-based, fully-managed, production-ready database with scalability, manageability, and security

Result:

Engineers spend less time managing and scaling the database, and more time building and improving their applications for better customer experience

PostgreSQL .... MOVE TO MANAGED → .... Amazon RDS for PostgreSQL

Davinci

Challenge:

Expedited growth with customers demanding cloud-based financial services.

Solution:

Migrated SQL Server database to **Amazon RDS** then to **Amazon Aurora** for PostgreSQL.

Result:

"Once we moved our SQL Server applications to Amazon Aurora PostgreSQL, we saw a 30 percent performance increase without needing to do any optimization."

—Peter Kobes,

Microsoft SQL Server MIGRATIONS → Amazon Aurora

aws

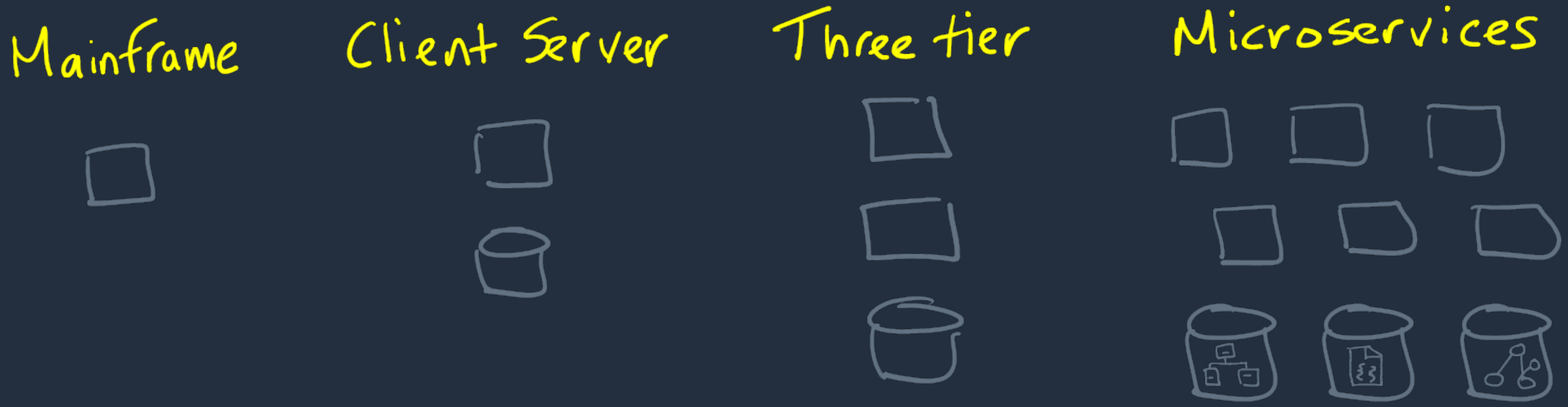
To get more value from their data, customers are..

Moving to fully managed database services

Building modern applications with purpose-built databases

Breaking free from legacy databases

App architectures & patterns have evolved over the years…



Modern Application Requirements

Requires more performance, scale, and availability

E-Commerce

Media streaming

Social media

Online gaming

Shared economy

Users	1M+
Data volume	Terabytes—petabytes
Locality	Global
Performance	Microsecond latency
Request rate	Millions per second
Access	Mobile, IoT, devices
Scale	Virtually unlimited
Economics	Pay-as-you-go
Developer access	Instance API access
Development	Apps & storage are decoupled

Purpose-built databases

	Relational	Key-value	Document	In-memory	Graph	Time-series	Ledger	Wide Column
	Referential integrity, ACID transactions, schema-on-write	High throughput, Low latency reads and writes, endless scale	Store documents and quickly access querying on any attribute	Query by key with microsecond latency	Quickly and easily create and navigate relationships between data	Collect, store, and process data sequenced by time	Complete, immutable, and verifiable history of all changes to application data	Scalable, highly available, and managed Apache Cassandra-compatible service
AWS Service(s)	Aurora RDS	DynamoDB	DocumentDB	ElastiCache	Neptune	Timestream	QLDB	Keyspaces Managed Cassandra
Common Use Cases	Lift and shift, ERP, CRM, finance	Real-time bidding, shopping cart, social, product catalog, customer preferences	Content management, personalization, mobile	Leaderboards, real-time analytics, caching	Fraud detection, social networking, recommendation engine	IoT applications, event tracking	Systems of record, supply chain, health care, registrations, financial	Build low-latency applications, leverage open source, migrate Cassandra to the cloud

To get more value from their data, customers are..

Moving to fully managed database services

Building modern applications with purpose-built databases

Breaking free from legacy databases

Legacy database providers are…

Very expensive

Proprietary

Lock-in

Punitive licensing

You’ve got mail

Customers are moving to open databases

Customers are moving to open databases

MySQL®　PostgreSQL　MariaDB

+

Commercial-grade performance and reliability?

Amazon Aurora

MySQL and PostgreSQL compatible relational database built for the cloud

Performance and availability of commercial-grade databases at 1/10th the cost

Performance and scalability

5x throughput of MySQL

3x throughput of PostgreSQL

15 read replicas

Scale out reads and writes across multiple data centers

Availability and durability

Fault-tolerant self-healing storage

Six copies of data across three AZs

Continuous backup to S3

Single Global database with cross-region replication

Highly secure

Network isolation

Encryption at rest/transit

Fully managed

Managed by RDS: no hardware provisioning, software patching, setup, configuration, or backups

Samsung migrates 1.1 billion users to Amazon Aurora with AWS Database Migration Service

Challenge

IT giant Samsung Electronics needed a more flexible, microservices-driven solution to replace its monolithic legacy internet data center, which was proving costly and unable to accommodate growing traffic.

Solution

The company migrated its 1.1 billion Samsung Account users to Amazon Aurora with minimal service disruption using AWS Database Migration Service, drastically reducing database costs and increasing scalability.

Results

- Underwent a global migration of a mission-critical workload in 18 months
- Enabled 60 ms latency or less 90% of the time
- Reduced monthly database costs by 44%

ORACLE MIGRATIONS → Amazon Aurora Amazon Database Migration Service Amazon DynamoDB aws

The Pokémon Company migrates to AWS purpose-built databases

Challenge:

The Pokémon Company International wanted to address the complexity of managing NoSQL and memory-caching systems for a database of more than 300 million users.

Solution:

The company uses Amazon Aurora as its main user database, Amazon DynamoDB to reduce bot login attempts, and Amazon ElastiCache to enable dynamic caching of user logins.

Results:

- Cuts monthly costs by tens of thousands of dollars
- Reduces number of nodes from 300 to 30
- Experiences zero hours of downtime or performance degradation after migration

Amazon Aurora

Amazon DynamoDB

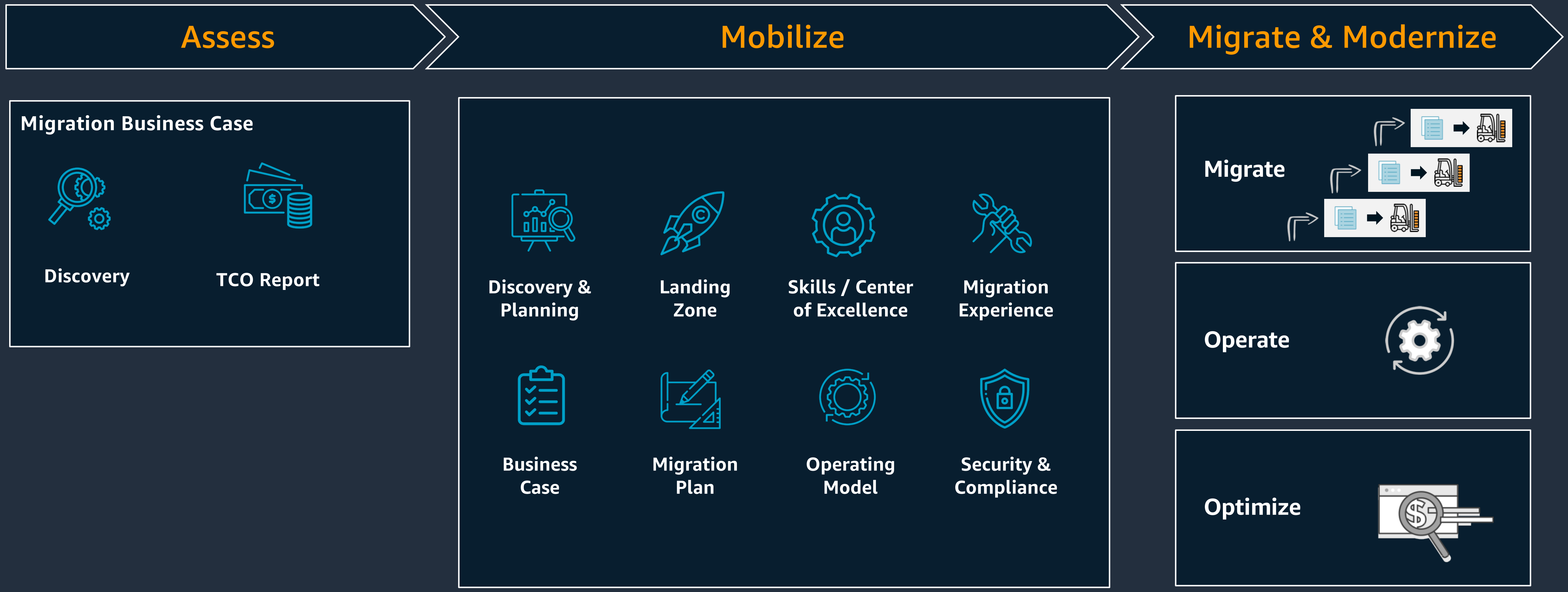
Amazon ElastiCache

Database Freedom: Break free from legacy databases

Database Freedom is an AWS database and analytics modernization initiative focused on accelerating enterprise migrations from Oracle, Microsoft SQL Server, Neteeza, Teradata, Cassandra, or MongoDB platforms to AWS cloud-native database services.

ORACLE Microsoft SQL Server cassandra mongoDB NETEZZA teradata.

Migration customer journey



Assess

Mobilize

Migrate & Modernize

Migration Business Case

Discovery

TCO Report

Discovery & Planning

Landing Zone

Skills / Center of Excellence

Migration Experience

Business Case

Migration Plan

Operating Model

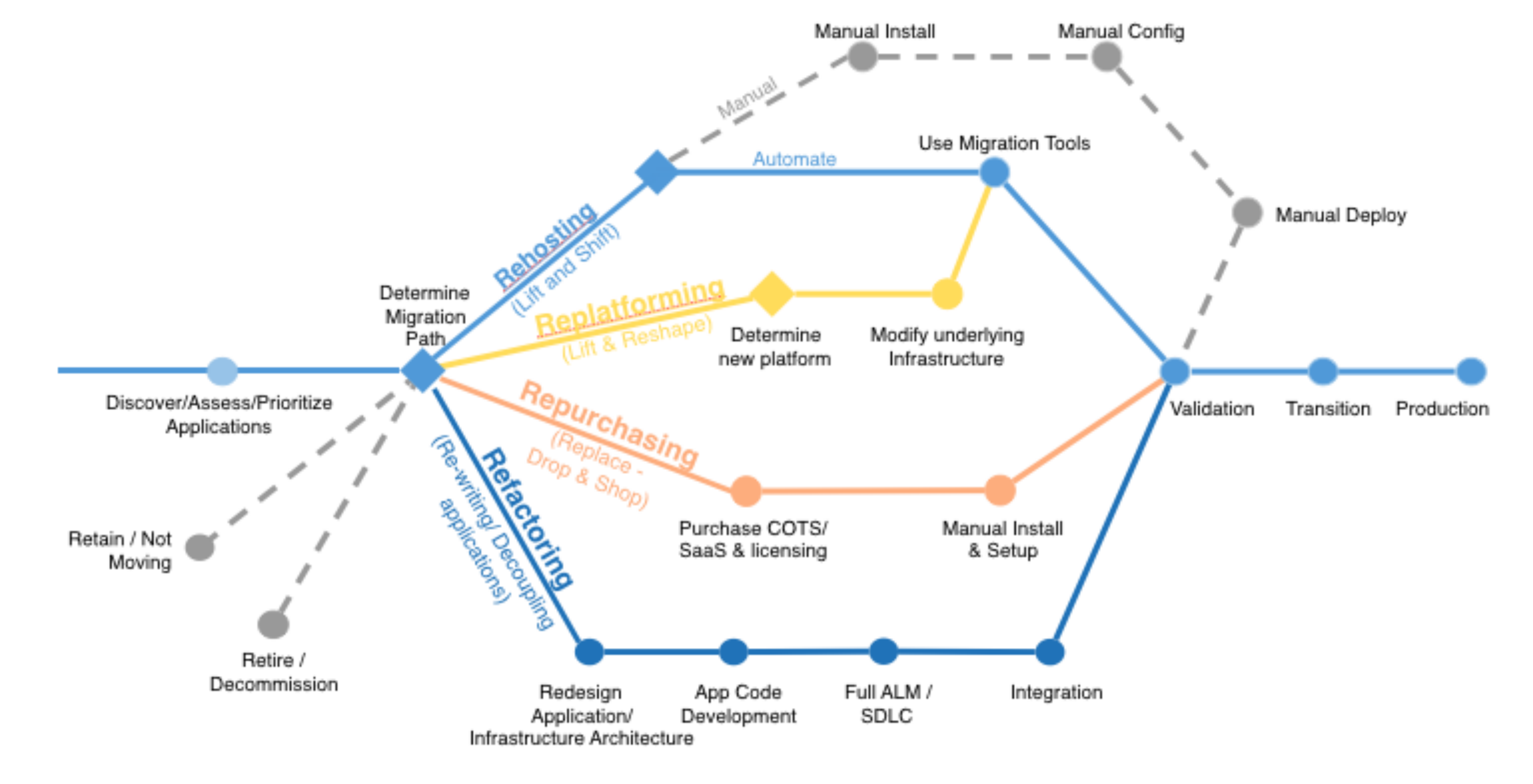
Security & Compliance

Migrate

Operate

Optimize

Application Migration Strategies



Which option is right for you?

Rehost: Run servers on Amazon Elastic Compute Cloud (Amazon EC2)

Replatform: Migrate to a managed service

Amazon Relational Database Service (Amazon RDS)

Refactor: Modernize application/migrate to cloud-native solution

Amazon Elastic Kubernetes Service (Amazon EKS), AWS Lambda, Amazon DynamoDB, and others)

Database Migration Service & Schema Conversion Tool

AWS DMS

Database migration and replication managed service

Benefit

- Easily and securely migrate and/or replicate your databases and data warehouses to AWS
- Migrate between different database engines
- Low cost and global availability

AWS SCT

Development tooling to convert schemas between databases and data warehouses

Benefit

- Automates schema conversion including database structure and code
- Minimizes manual effort of performing a schema re-write
- Allows conversion from commercial to open source platforms

Native tooling

Migration options included with the engine

Benefit

- Leverage a familiar environment
- Full support for native features

Open source

Ora2Pg and others

Benefit

- Tailored solution for a specific problem
- Multiple tools can offer a more complete solution than any one product
- Free

Commercial

Attunity, Golden Gate, Informatica, Talend, and others

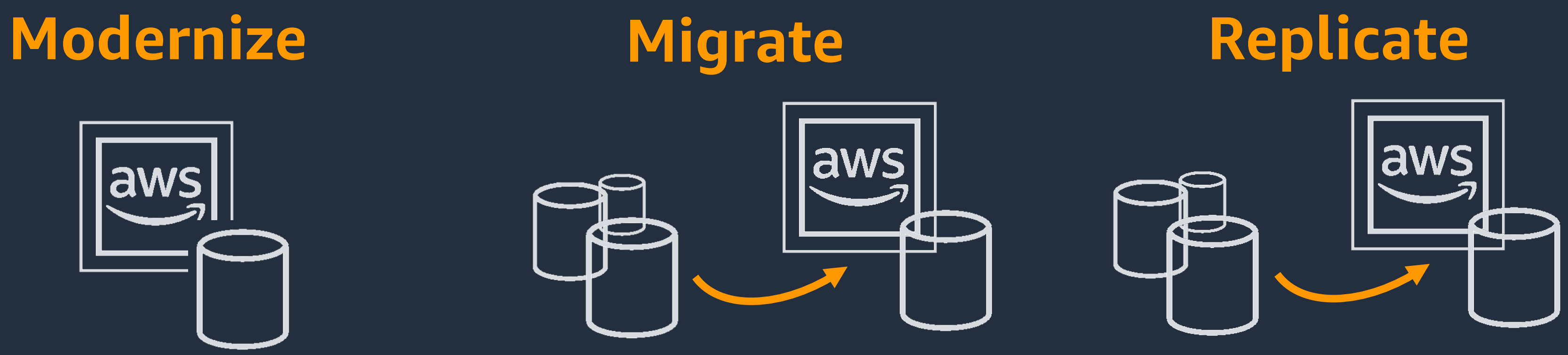
Benefit

- Commercial solutions are available for a wide range of migration sources and targets including legacy mainframe systems

AWS Database Migration Service & AWS Schema Conversion Tool

AWS Database Migration Service (AWS DMS) easily and securely migrates and/or replicates your databases *and* data warehouses to AWS

AWS Schema Conversion Tool (AWS SCT) converts your commercial database and data warehouse schemas to open-source engines or native AWS services, such as Amazon Aurora and Amazon Redshift



Modernize

Migrate

Replicate

Workload Qualification Framework (WQF)

Assesses and classifies OLTP workloads to determine the ease of migration, and which target is appropriate: AWS RDS or Aurora database

WQF classifies OLTP workloads into 5 categories:

🟩	Category 1	ODBC/JBDC workloads
🟩	Category 2	Light, proprietary feature workloads
🟩🟦	Category 3	Heavy, proprietary feature workloads
🟦🟪	Category 4	Engine-specific workloads
🟪	Category 5	Nonportable, high-risk, or lift-and-shift workloads

- Migration complexity assessments
- Workload migration strategy

Verisk

Challenge:

The Oracle and SQL databases could no longer meet the compliance and performance requirements of running analytics software in various regions around the world.

Solution:

Migrated Oracle and SQL databases to **Amazon Aurora** to improve speed, latency and processing times of their databases.

Result:

"Thanks to Amazon Aurora, we can now deliver a highly responsive, scalable and highly available intelligent compliance solution to our worldwide customers. The migration gave us a significant cost reduction and improved our ability to deliver value to our customers."

—Ashish Verma,

ORACLE Microsoft SQL Server MIGRATIONS → Amazon Aurora

aws

ProQuest

Challenge:

The Oracle database was monolithic, slow,
and complex to maintain.

Solution:

Migrated all Oracle workloads to **Amazon Aurora** in four hours. The migration helped save cost, scales as the customer base grows, and has simplified licensing.

Result:

"We've gotten so many benefits from migrating our databases from Oracle to Aurora. There is high availability out of the box, lower cost, easier scaling for reads, and far less licensing complexity."

—Suresh Karri,

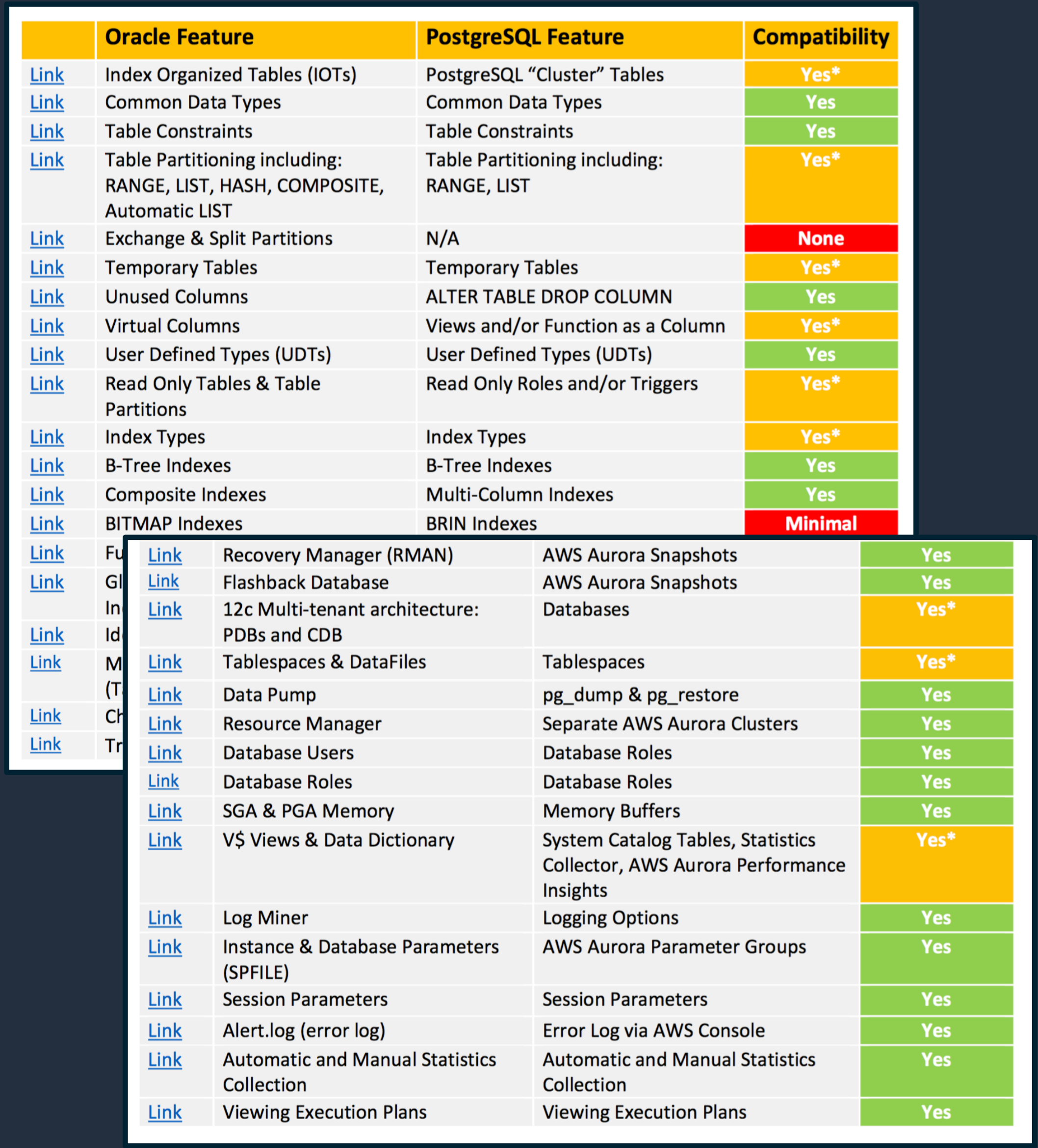
ORACLE MIGRATIONS → Amazon Aurora

aws

Legacy databases to AWS migration playbooks

Resources to help guide your migration

- Topic-by-topic overview of how to migrate databases and data warehouses to AWS services
- Covers all proprietary features and the different database objects
- Migration best practices
- Oracle to Aurora PostgreSQL
- SQL Server to Aurora MySQL
- SQL Server to Aurora PostgreSQL
- Oracle to Aurora MySQL
- Cassandra to DynamoDB, data warehouses to Amazon Redshift, and Db2 LUW to Aurora PostgreSQL



	Oracle Feature	PostgreSQL Feature	Compatibility
Link	Index Organized Tables (IOTs)	PostgreSQL "Cluster" Tables	Yes*
Link	Common Data Types	Common Data Types	Yes
Link	Table Constraints	Table Constraints	Yes
Link	Table Partitioning including: RANGE, LIST, HASH, COMPOSITE, Automatic LIST	Table Partitioning including: RANGE, LIST	Yes*
Link	Exchange & Split Partitions	N/A	None
Link	Temporary Tables	Temporary Tables	Yes*
Link	Unused Columns	ALTER TABLE DROP COLUMN	Yes
Link	Virtual Columns	Views and/or Function as a Column	Yes*
Link	User Defined Types (UDTs)	User Defined Types (UDTs)	Yes
Link	Read Only Tables & Table Partitions	Read Only Roles and/or Triggers	Yes*
Link	Index Types	Index Types	Yes*
Link	B-Tree Indexes	B-Tree Indexes	Yes
Link	Composite Indexes	Multi-Column Indexes	Yes
Link	BITMAP Indexes	BRIN Indexes	Minimal
Link	Fu		
Link	Gl In		
Link	Id		
Link	M (T		
Link	Ch		
Link	Tr		

Link	Recovery Manager (RMAN)	AWS Aurora Snapshots	Yes
Link	Flashback Database	AWS Aurora Snapshots	Yes
Link	12c Multi-tenant architecture: PDBs and CDB	Databases	Yes*
Link	Tablespaces & DataFiles	Tablespaces	Yes*
Link	Data Pump	pg_dump & pg_restore	Yes
Link	Resource Manager	Separate AWS Aurora Clusters	Yes
Link	Database Users	Database Roles	Yes
Link	Database Roles	Database Roles	Yes
Link	SGA & PGA Memory	Memory Buffers	Yes
Link	V$ Views & Data Dictionary	System Catalog Tables, Statistics Collector, AWS Aurora Performance Insights	Yes*
Link	Log Miner	Logging Options	Yes
Link	Instance & Database Parameters (SPFILE)	AWS Aurora Parameter Groups	Yes
Link	Session Parameters	Session Parameters	Yes
Link	Alert.log (error log)	Error Log via AWS Console	Yes
Link	Automatic and Manual Statistics Collection	Automatic and Manual Statistics Collection	Yes
Link	Viewing Execution Plans	Viewing Execution Plans	Yes

Schema + Data + Best practices

AWS SCT + AWS DMS + Playbook

AWS Migration Partners

slalom

NorthBay

epam

ONICA
a rackspace company

AGILISIUM

Cloudwick

wipro

NAVISITE

virtusa
Accelerating Business Outcomes

TEKSTREAM
The People to Power Your Success

MACTORES

SYSTECH

MLOGICA

DB BEST
TECHNOLOGIES

datAvail
BI/Analytics • Applications • Databases

TEKsystems
Own change

eCloud valley

License Advisory

PALISADE COMPLIANCE

House of Brick
TECHNOLOGIES

Redwood Compliance

softwareONE

M1RO CONSULTING

apps associates

aws

Learn databases with AWS Training and Certification

Resources created by the experts at AWS to help you build and validate database skills

25+ free digital training courses cover topics and services related to relational and nonrelational databases

The classroom offering, Planning and Designing Databases on AWS, features AWS expert instructors and hands-on activities

Validate expertise with the AWS Certified Database – Specialty exam

Visit the databases learning path at aws.amazon.com/training/path-databases

John Wilson – Tungsten Network

March 2021

Oracle Migration to AWS

Tungsten Network & me

- e-invoicing delivered as a service
- Bringing together businesses and their many suppliers
  - *Faster, more consistent processing of invoices and purchase orders*
  - *Compliancy with international mandates and regulations*
  - *Analysis of real time spend*
  - *Improved cashflow management*
- I joined Tungsten in late 2019 as Head of Technology Operations
- My background is SaaS/AWS delivery and service transformation
- My brief was to improve service quality, security and cost effectiveness

The situation analysis

- Where we were
  - *AWS as a managed service*
  - *35TB Oracle databases*
  - *Hybrid colocation in London and Slough*
  - *Limited in house capability*
  - *End of life infrastructure*
  - *Incomplete instrumentation*
  - *Suspicion over service performance*

- The challenges
  - *Low sense of technical ownership*
  - *Most capacity already consumed*
  - *Extra complexity and overhead*
  - *Little vision for the future*
  - *No refresh strategy*
  - *Hard to break down service quality*
  - *Believing we could perform better*

Looking for help

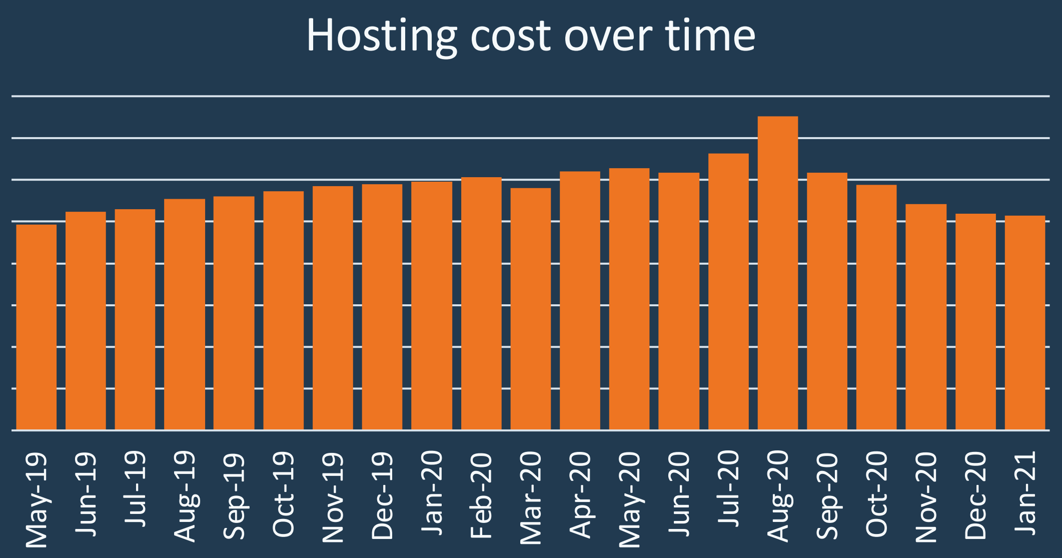
- Going “all in” with AWS a no brainer
- Contacting our AWS account manager
- Consultations and introductions
- A new AWS and Oracle partner in Navisite
- Developing the AWS and database roadmap
- Use increased capacity to drive an EDP
- Addressing questions and concerns
- Building and selling the business case

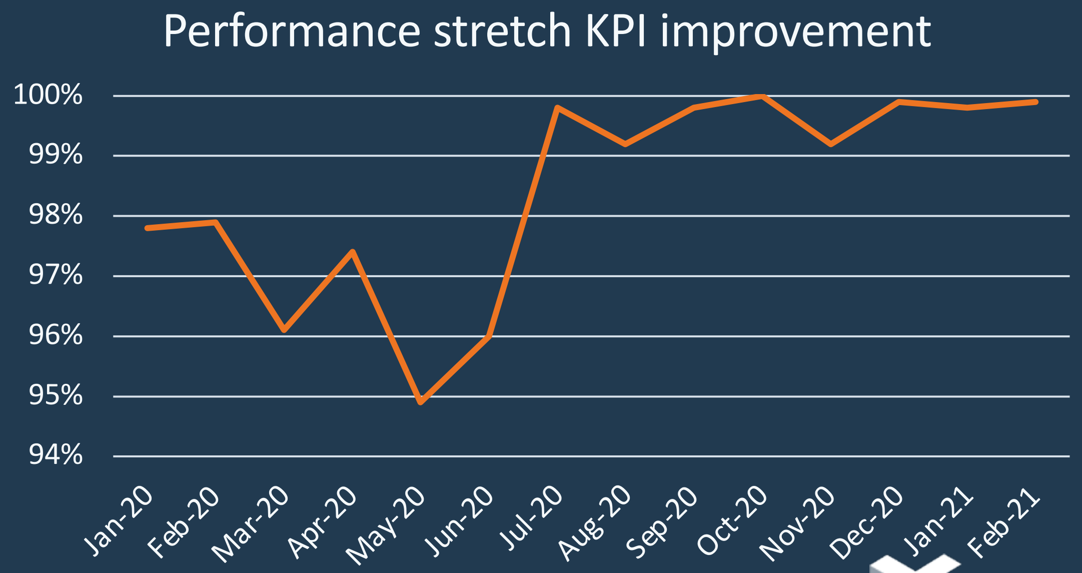
Transformation

- Tungsten, Navisite and AWS collective effort
- AWS capacity and Oracle DBA services
- AWS migration funding support
- Migrate now and optimise later
- Design validation and compliance check
- Oracle Active Data Guard supported migration
- A few final challenges but all addressed
- Performance checks
- Dry run, wet run and live run

Benefit realisation

- Simpler, better performing infrastructure
- Total DB infrastructure refresh
- Storage encryption for better security profile
- EDP driven lower price point for AWS capacity
- Improved recoverability
- Reduced running cost back to May 2019
- 40% faster query execution where it matters
- Stretch KPI goals no longer a stretch
- More engagement with partners and AWS





TUNGSTEN NETWORK

To summarise

- Running our Oracle workload in AWS was a great opportunity to improve performance, security, cost and service
- Approaching AWS direct was key to getting things moving
- Navisite as both AWS and Oracle partner gave us confidence we were heading in the right direction
- Working together we were able to realise the benefits that we promised to the business

Thank you

TUNGSTEN NETWORK