Cloud Native: AKS & OSS Database Architectural Guidance

PROJECT UNLEASH TEAM

Agenda

Cloud Native (Kubernetes & AKS)

- Building microservices on Azure
- Microservices architecture on Azure Kubernetes Service (AKS)
- Building a CI/CD pipeline for microservices on Kubernetes
- Monitoring a microservices architecture in Azure Kubernetes Service (AKS)
- Cluster operator and developer best practices to build and manage applications on Azure Kubernetes Service (AKS)

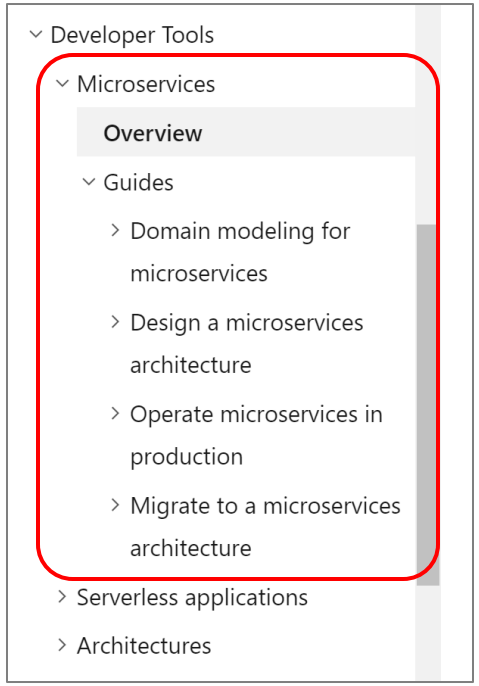
Open Source Database

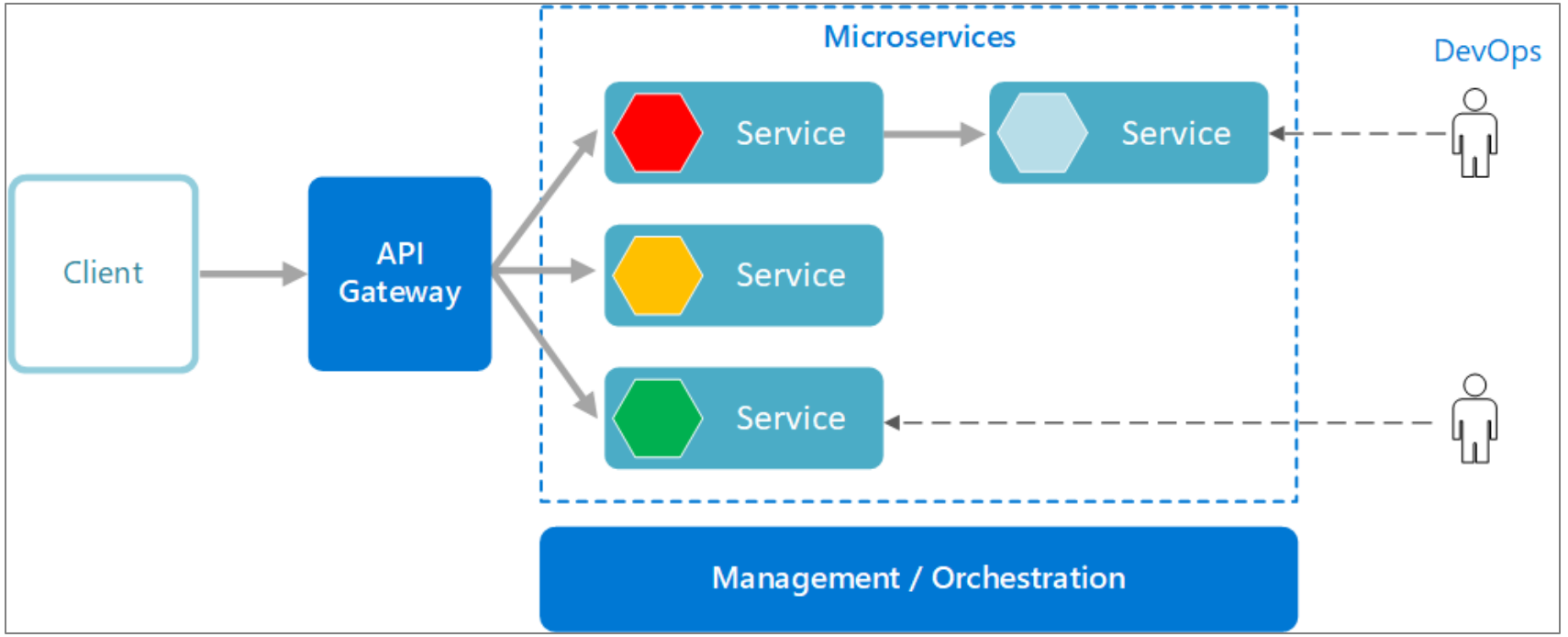
- Reference Architecture: Digital Marketing using Azure Database for MySQL
- Reference Architecture: Finance management apps using Azure Database for MySQL
- Reference Architecture: Gaming using Azure Database for MySQL
- Azure Database Migration Guide
- Migration Guide: Migrate Oracle to Azure Database for PostgreSQL

kubernetes

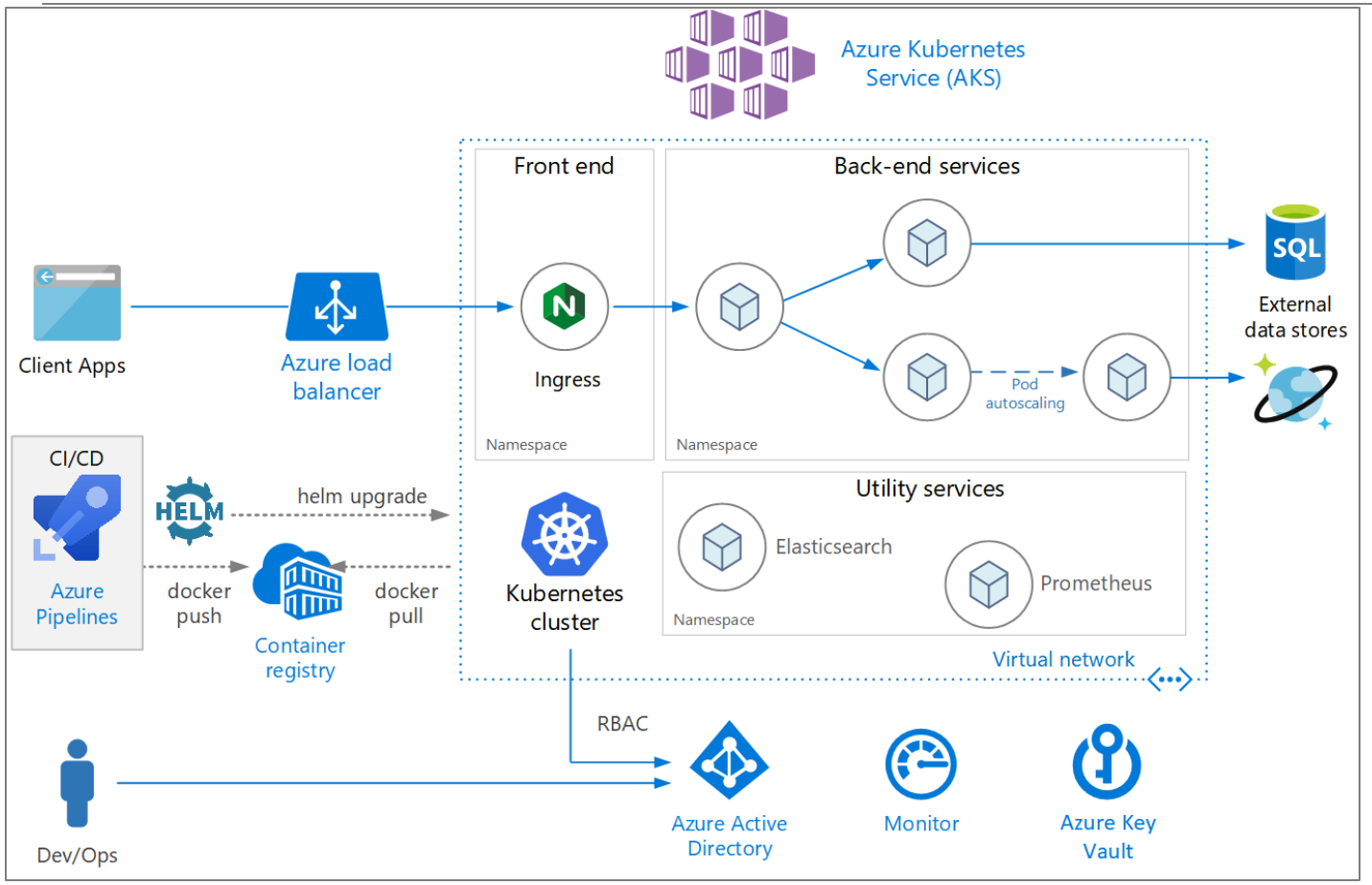
Cloud Native (Kubernetes & AKS)

Building microservices on Azure

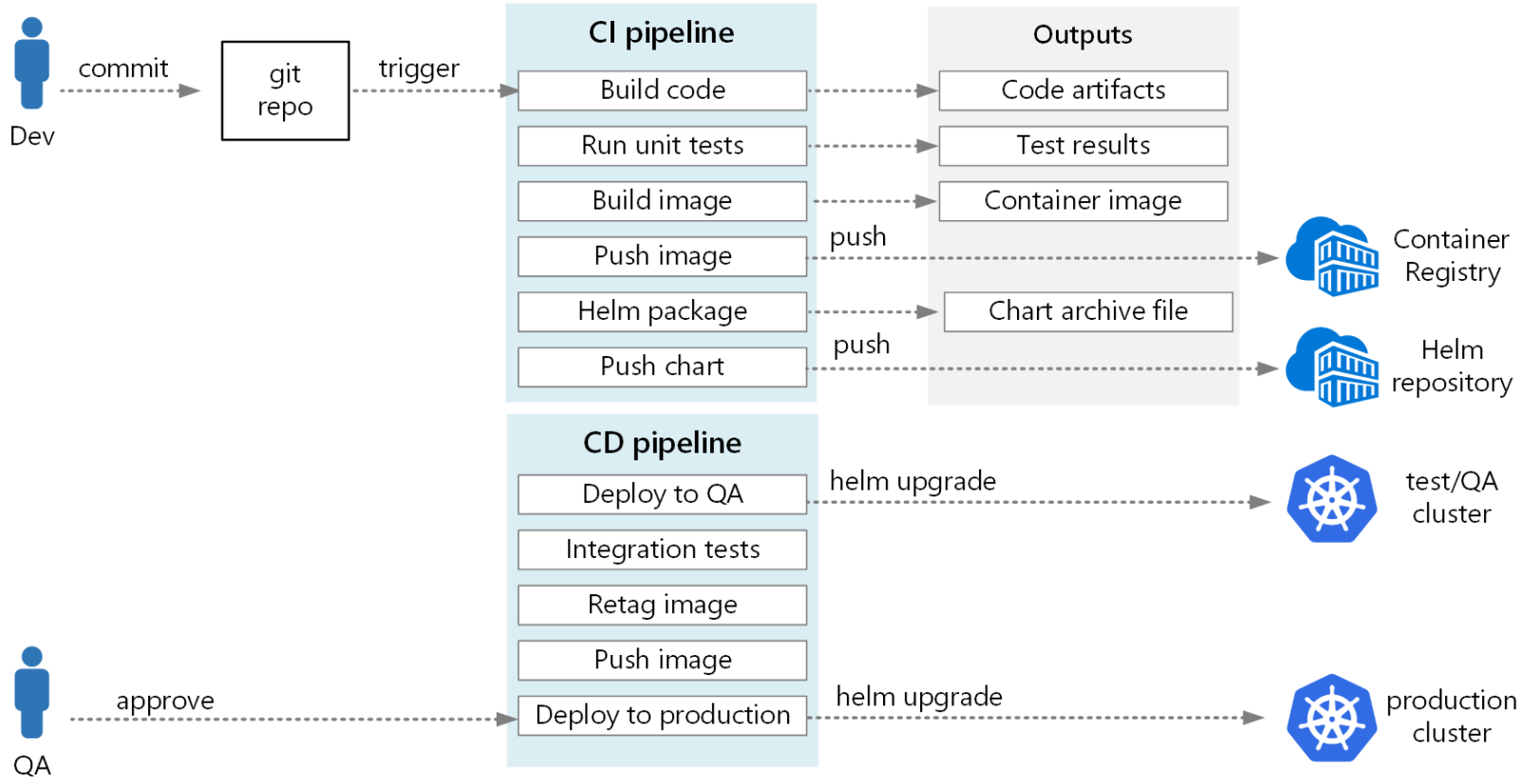




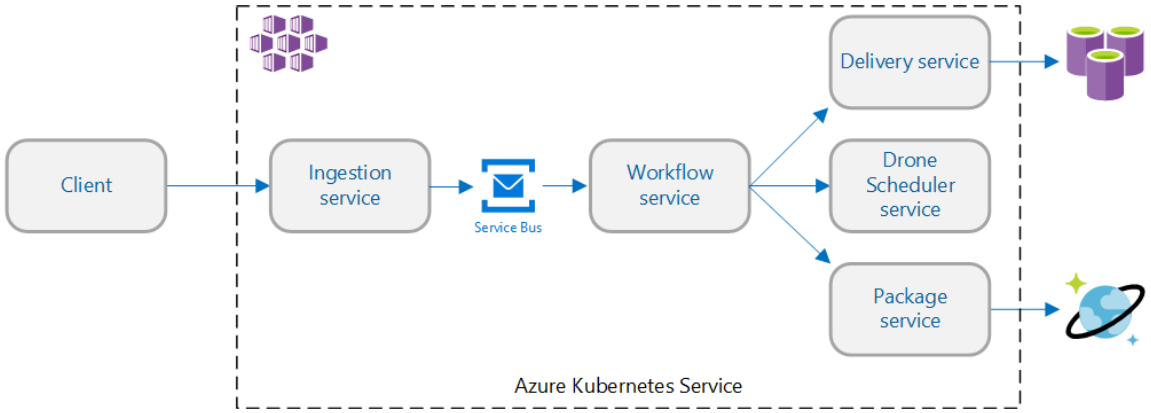
Microservices architecture on Azure Kubernetes Service (AKS)

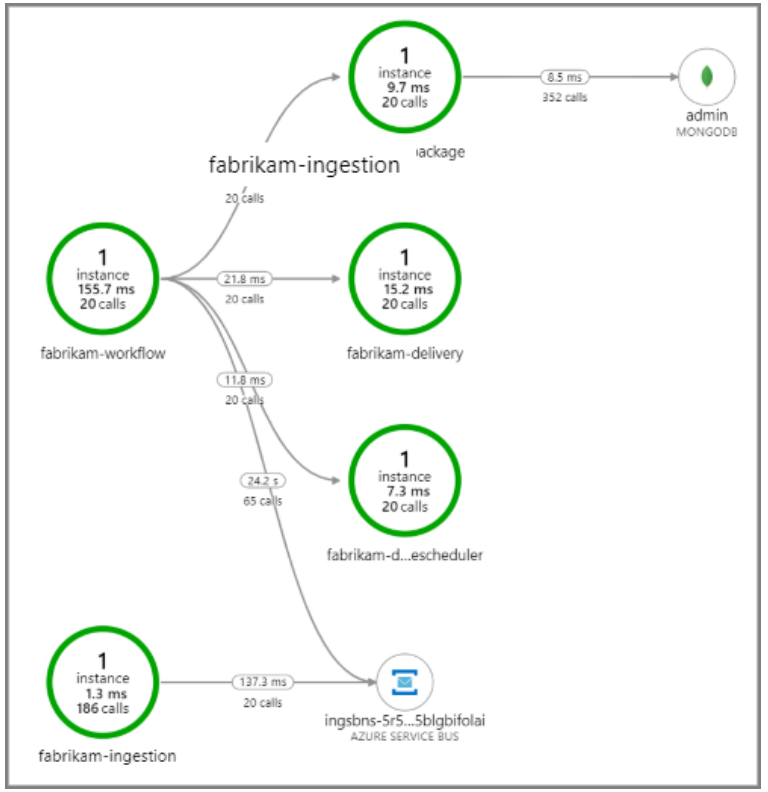


Building a CI/CD pipeline for microservices on Kubernetes



Monitoring a microservices architecture in Azure Kubernetes Service (AKS)





Cluster operator and developer best practices to build and manage applications on Azure Kubernetes Service

Cluster operator best practices

Multi-tenancy

- Best practices for cluster isolation
- Best practices for basic scheduler features
- Best practices for advanced scheduler features
- Best practices for authentication and authorization

Security

- Best practices for cluster security and upgrades
- Best practices for container image management and security
- Best practices for pod security

Network and storage

- Best practices for network connectivity
- Best practices for storage and backups

Running enterprise-ready workloads

- Best practices for business continuity and disaster recovery

Developer best practices

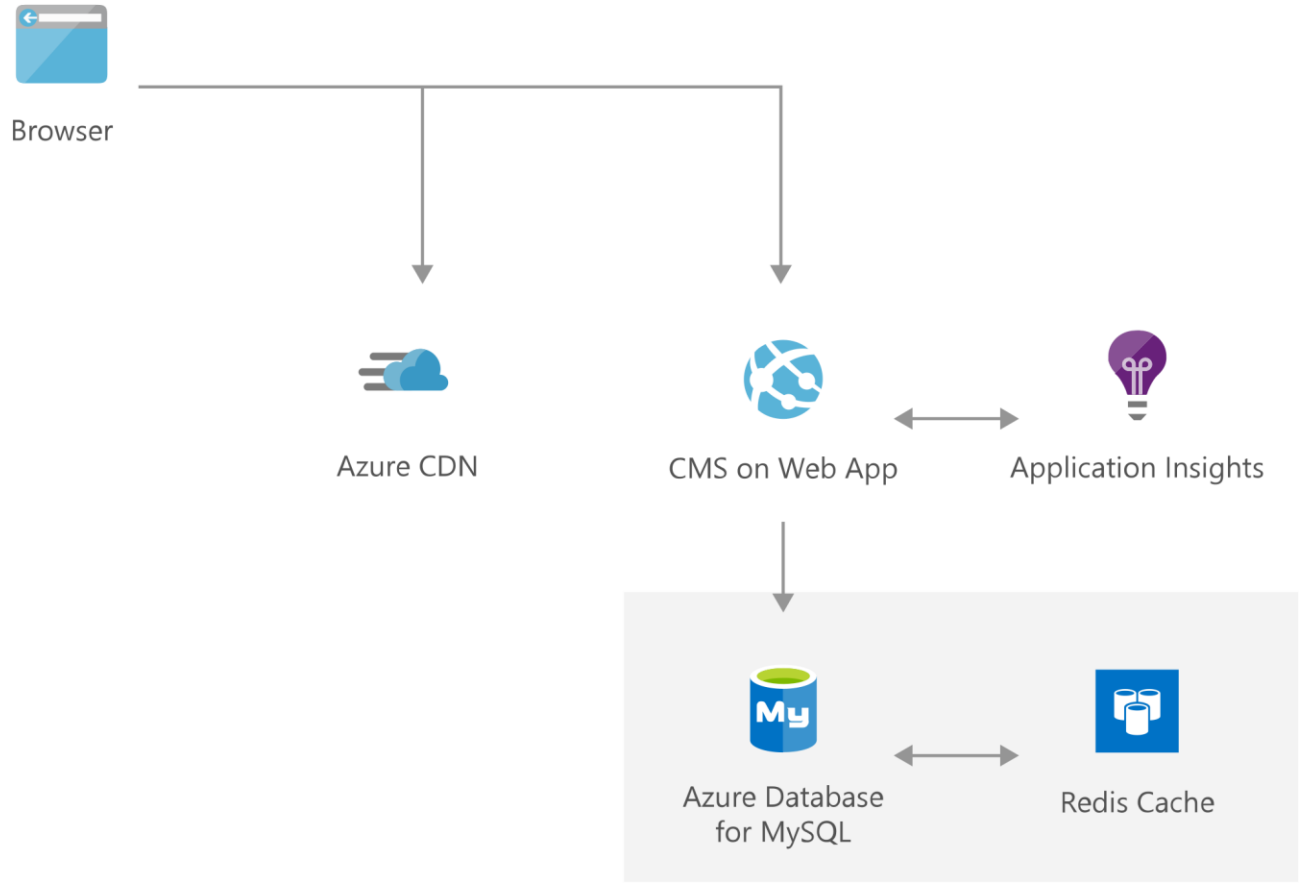
- Best practices for application developers to manage resources
  - Includes defining pod resource requests and limits, configuring development tools, and checking for application issues.
- Best practices for pod security
  - Includes securing access to resources, limiting credential exposure, and using pod identities and digital key vaults.

Azure Open Source Database

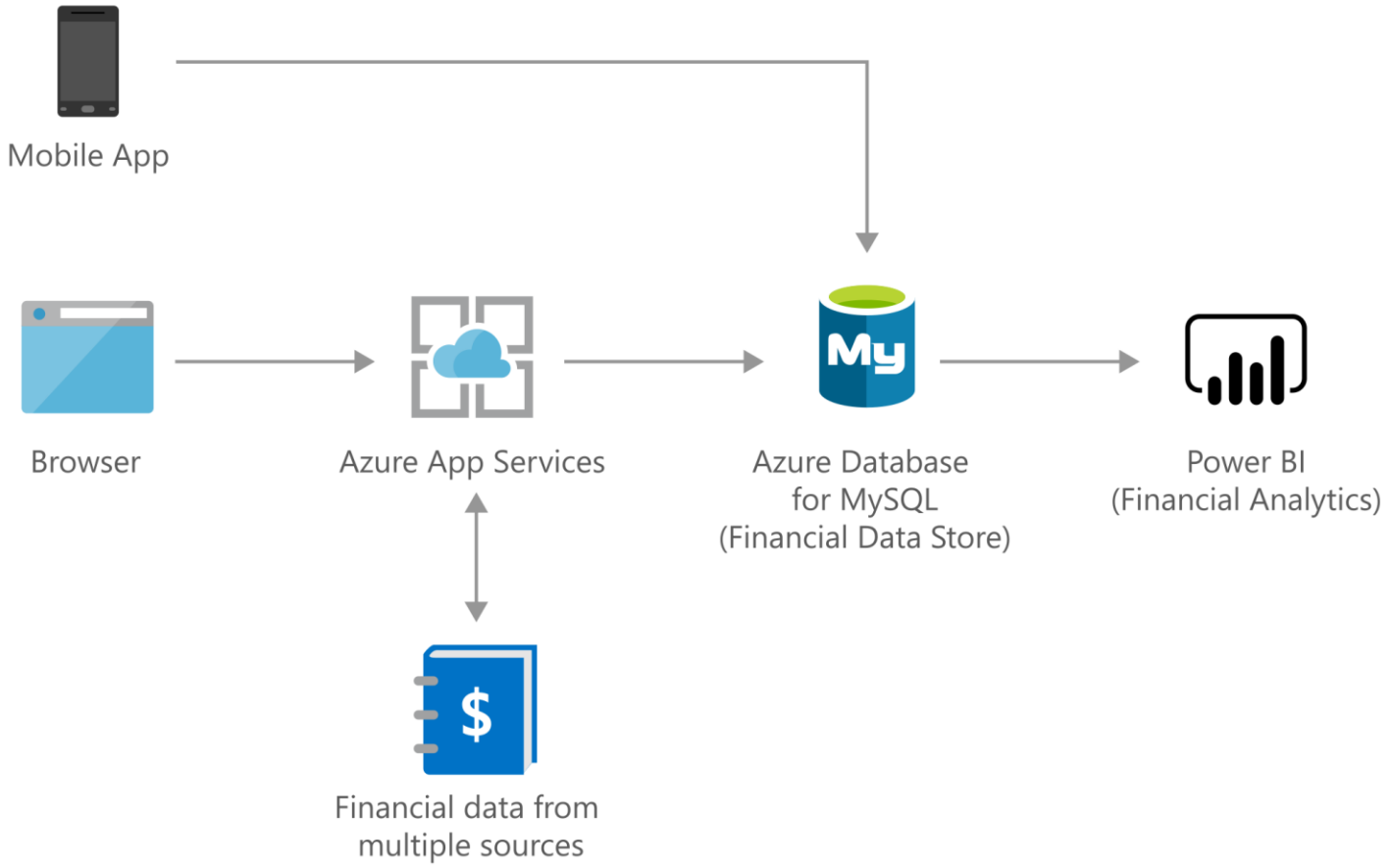
AZURE DATABASE FOR:

POSTGRESQL, MYSQL, MARIADB

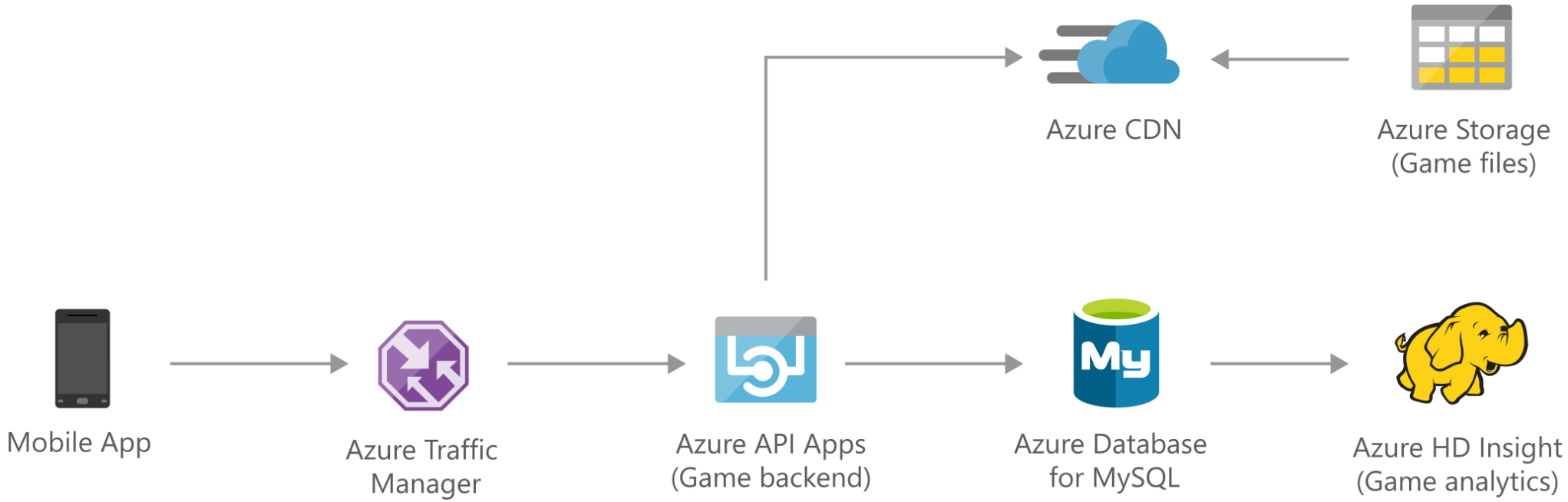
Reference Architecture: Digital Marketing using Azure Database for MySQL



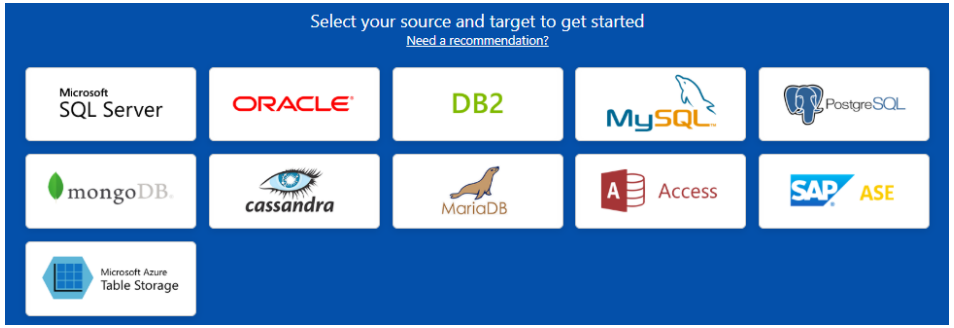
Reference Architecture: Finance management apps using Azure Database for MySQL



Reference Architecture: Gaming using Azure Database for MySQL



Azure Database Migration Guide



Microsoft migration tools and services

Azure Migrate
A central hub for starting, executing, and tracking your Azure migration.

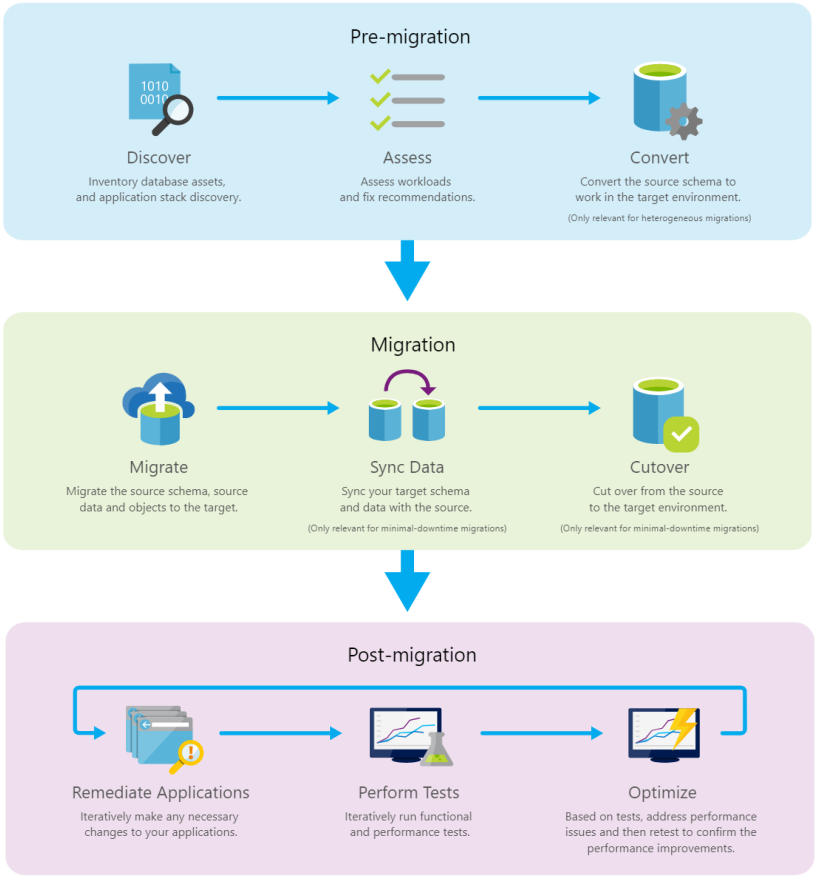
Azure Database Migration Service
An end-to-end solution for migrating multiple sources to cloud database platforms at scale.

Data Migration Assistant
A tool for assessing source SQL databases for potential compatibility issues on your target platform.

SQL Server Migration Assistant
A tool for assessing competitor source databases for potential compatibility issues on your target platform.

Database Experimentation Assistant
An experimentation solution for evaluating a targeted version of SQL Server for a specific workload.

Data Access Migration Toolkit (Preview)
A VS Code extension for analyzing Java and .Net source code and detecting data access API calls and queries.



HTTPS://DATAMIGRATION.MICROSOFT.COM/

Migration Guide: Migrate Oracle to Azure Database for PostgreSQL

Top reasons why migrating from Oracle to PostgreSQL:

- Economic reason: Oracle license
- PostgreSQL is completely open source database
  - No lock-in experience
  - Open source culture
- Maturity of PostgreSQL
- Innovation

Migrate Oracle to Azure Database for PostgreSQL

1. Before you start | 2. Pre-migration | 3. Migration | 4. Post-migration

Preparing for database migration

As you prepare for migrating your database to the cloud, verify that your source environment is supported and that you have addressed any prerequisites. This will help to ensure an efficient and successful migration.

Overview

PostgreSQL is one of world's most advanced open source databases. This article describes how to use the free ora2pg utility to migrate an Oracle database to PostgreSQL. You can use ora2pg, a free tool, to migrate an Oracle or MySQL database to a PostgreSQL compatible schema. The utility connects your Oracle database, scans it automatically, and extracts its structure or data. Afterwards ora2pg generates SQL scripts that you can load into your PostgreSQL database. ora2pg can be used for tasks from reverse engineering an Oracle database, performing a huge enterprise database migration, or simply replicating some Oracle data into a PostgreSQL database. It's easy to use and doesn't require any Oracle database knowledge other than the ability to provide the parameters needed to connect to the Oracle database.

Note: For more information about using the latest version of ora2pg, see the ora2pg documentation.

Typical ora2pg migration architecture

After provisioning the VM and Azure Database for PostgreSQL, two configurations are needed for enabling connectivity between them: "Allow Azure Services" and "Enforce SSL Connection", depicted as follows:

- "Connection Security" blade -> Allow access to Azure Services -> ON
- "Connection Security" blade -> SSL Settings -> Enforce SSL Connection -> DISABLED

HTTPS://DATAMIGRATION.MICROSOFT.COM/SCENARIO/ORACLE-TO-AZUREPOSTGRESQL?STEP=1