SQL Server on AWS Best Practices

Robert Rozas Navarro
Premier Field Engineer
Apps Domain

Agenda

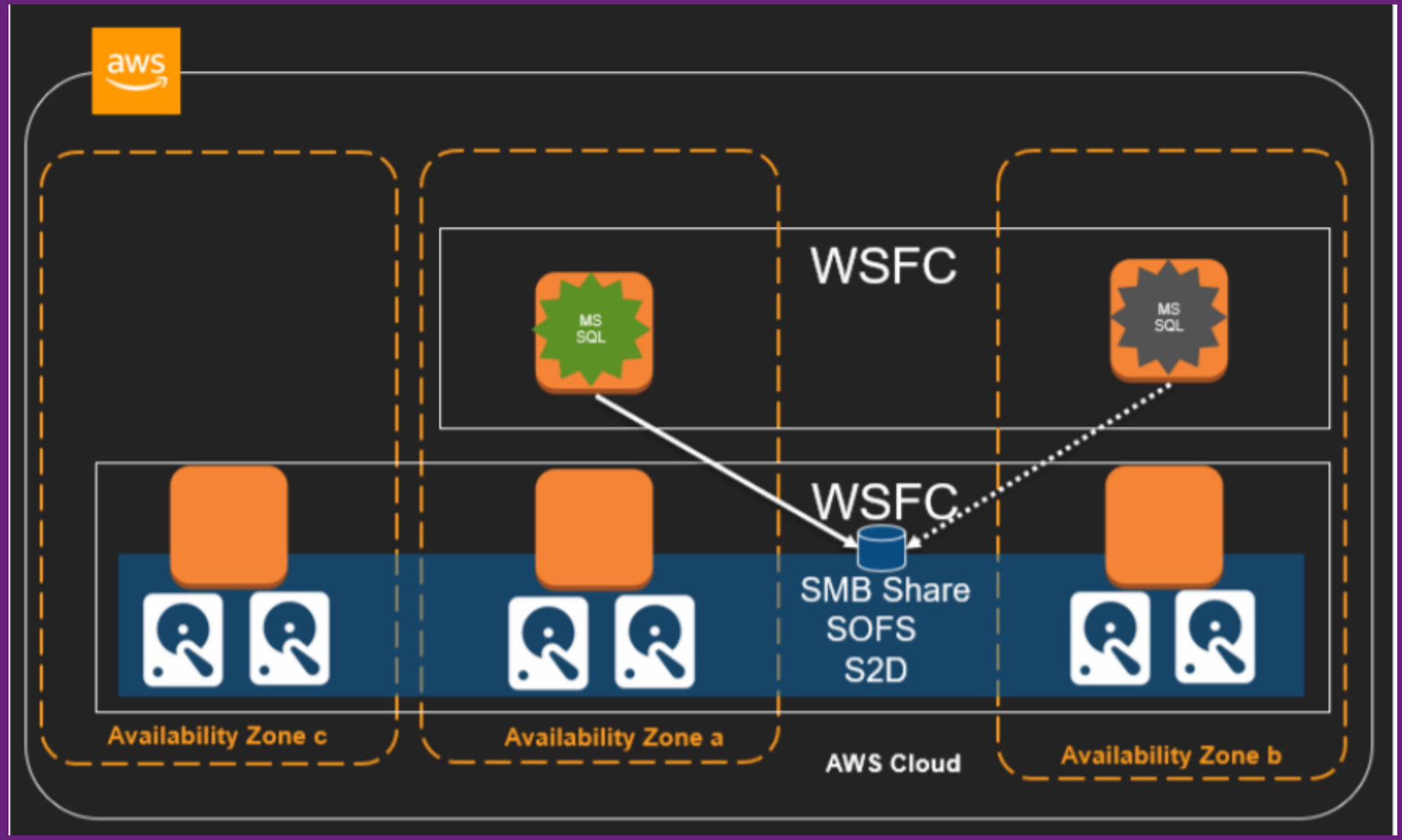
1. Introduction
2. Performance Optimization
3. Security Optimization
4. Cost Optimization
5. Q&A

Intro

Microsoft SQL Server offers several High Availability/Disaster Recovery (HA/DR) solutions, each suitable for specific requirements.

- Log Shipping
- Mirroring (Deprecated, use Availability Groups instead)
- Always On Availability Groups (Enterprise Edition)
- Always On Basic Availability Groups (Standard Edition)
- Always On Failover Cluster Instances
- Distributed Availability Groups

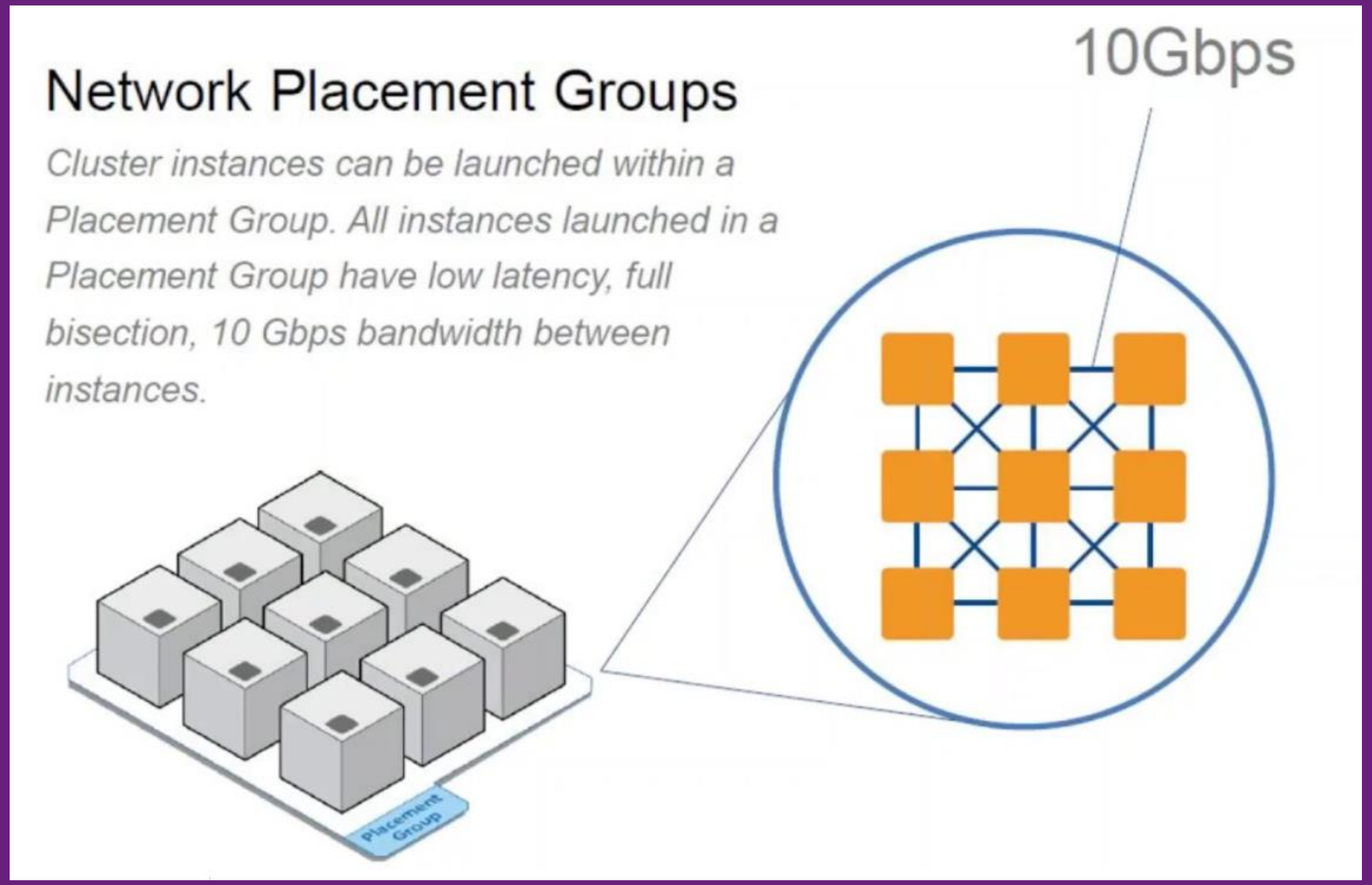
Availability Zones and Multi-AZ Deployment



Availability Zones and Multi-AZ Deployment

- AWS Availability Zones (AZs) are designed to provide separate failure domains, while keeping workloads in relatively close proximity for low latency inter-communications.
- AZs are a good solution for synchronous replication of your databases using Mirroring, Always On Availability Groups, or Basic Availability Groups.
- SQL Server provides zero data-loss and, when combined with the low-latency infrastructure of AWS Availability Zones, provides high performance.

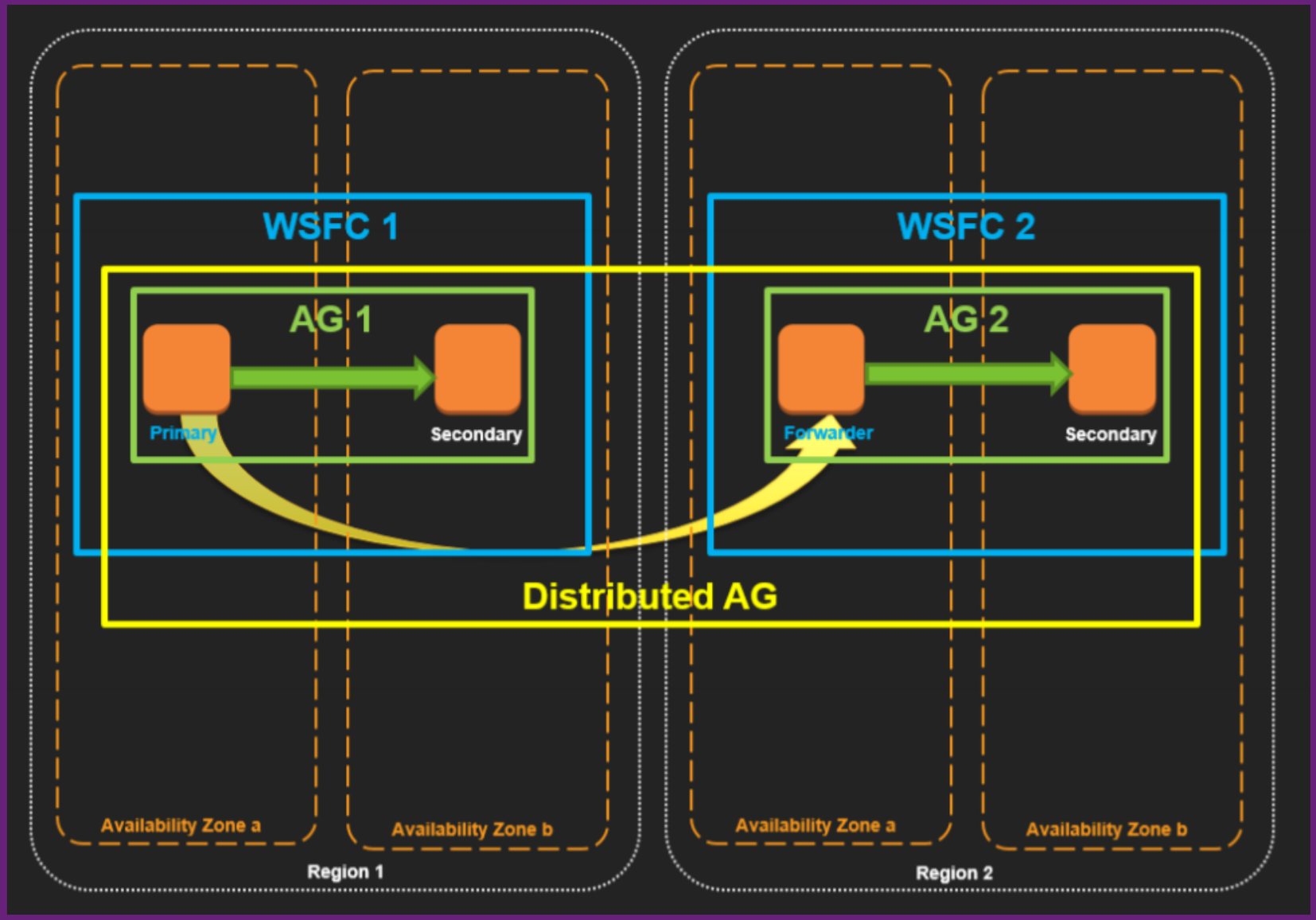
Cluster Placement Groups and Enhanced Networking



Cluster Placement Groups and Enhanced Networking

- Amazon EC2 enables you to deploy a number of EC2 instances inside a cluster placement group.
- To gain the highest bandwidth on AWS, you can leverage enhanced networking and Elastic Network Adapter (ENA)
- To minimize latency, you can deploy Always On Failover Cluster Instances, or Always On Availability Groups on instances that run inside an EC2 cluster placement group.

Multi-Region Deployment



Multi-Region Deployment

- For those workloads that require even more resilience against unplanned events, you can leverage the global scale of AWS to ensure availability under any circumstances.
- If you have applications or users that are deployed in remote regions which need to connect to your SQL Server instances, you can leverage the AWS Direct Connect feature that provides connectivity from any Direct Connect connection to all AWS regions.

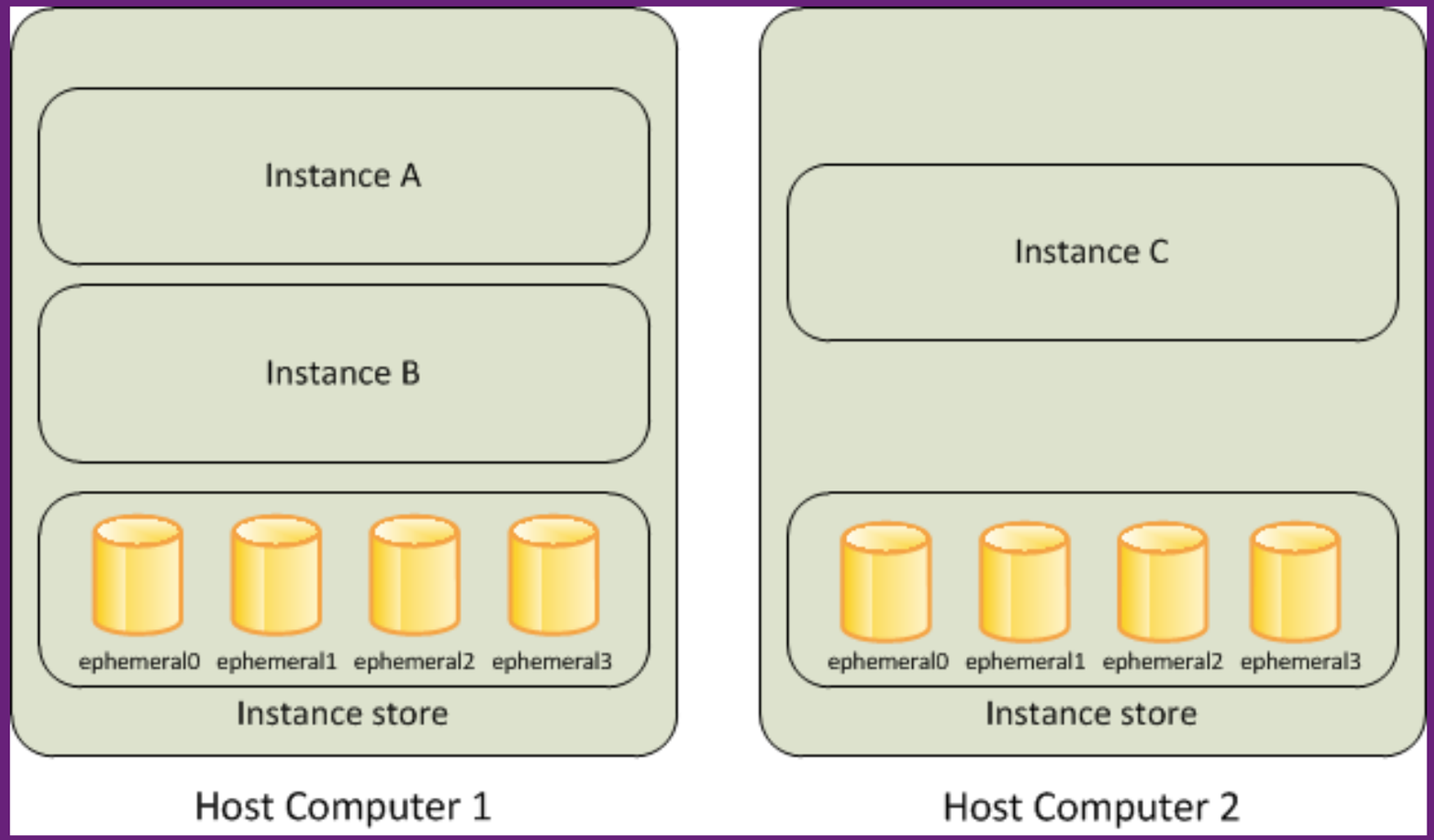
Performance Optimization

Using Amazon Elastic Block Store (Amazon EBS)

Using Amazon Elastic Block Store (Amazon EBS)

- Amazon EBS is a single-AZ block storage service with various flexible options, catering for diverse requirements.
- One point to remember is to use EBS-optimized EC2 instance types.
- You can use AWS Systems Manager Run Command to take application-consistent EBS snapshots of your online SQL Server files at any time, with no need to bring your database offline or in read-only mode.

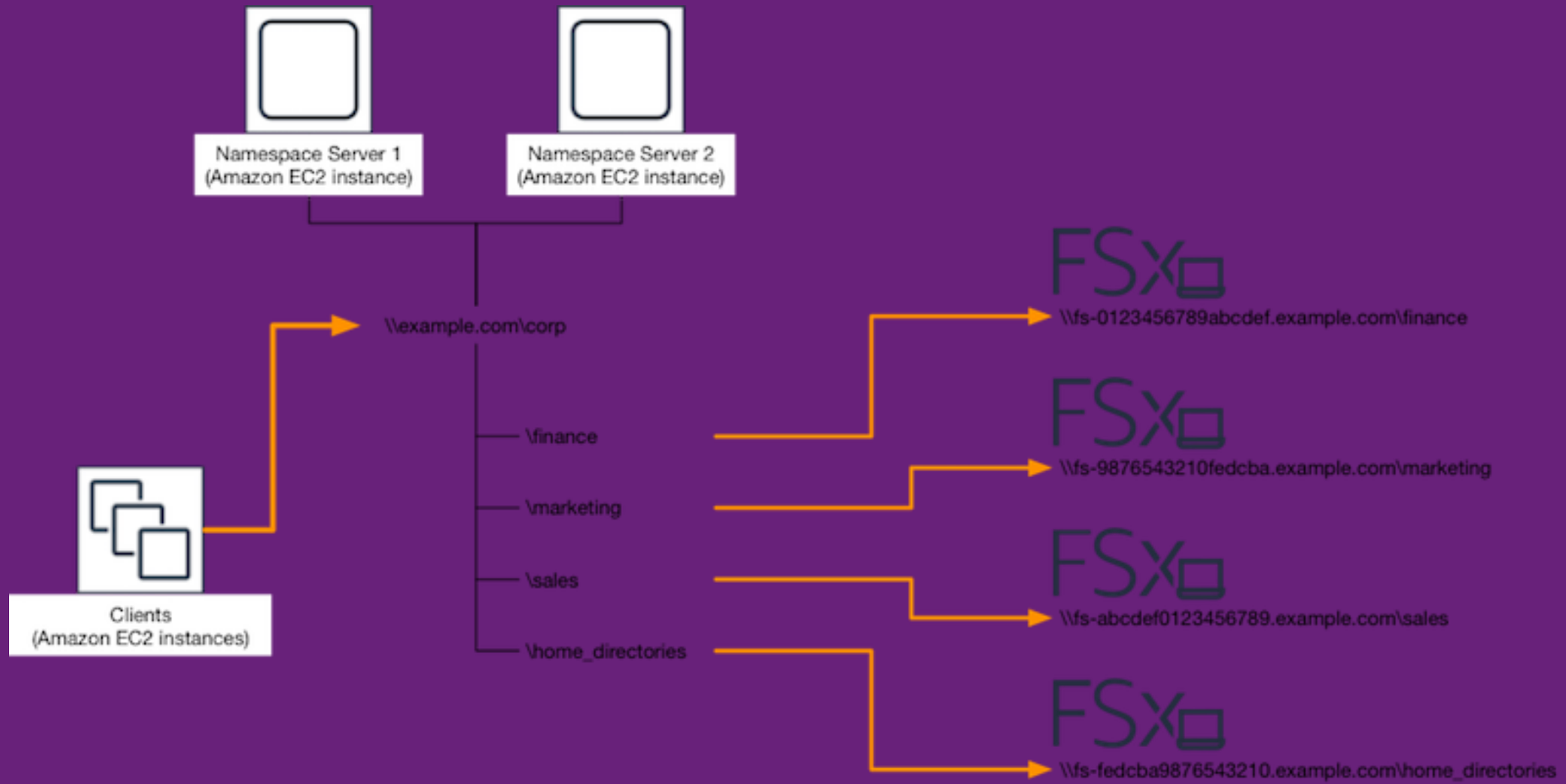
Instance Storage



Instance Storage

- Storage optimized EC2 instance types8 use fixed-size local disks and a variety of different storage technologies are available.
- Among these, Non-Volatile Memory express (NVMe) is the fastest technology with the highest IOPS and throughput.
- Instance disks are ephemeral and only live as long as their associated EC2 instance lives.
- Another use for EC2 instance storage is the buffer pool extension.

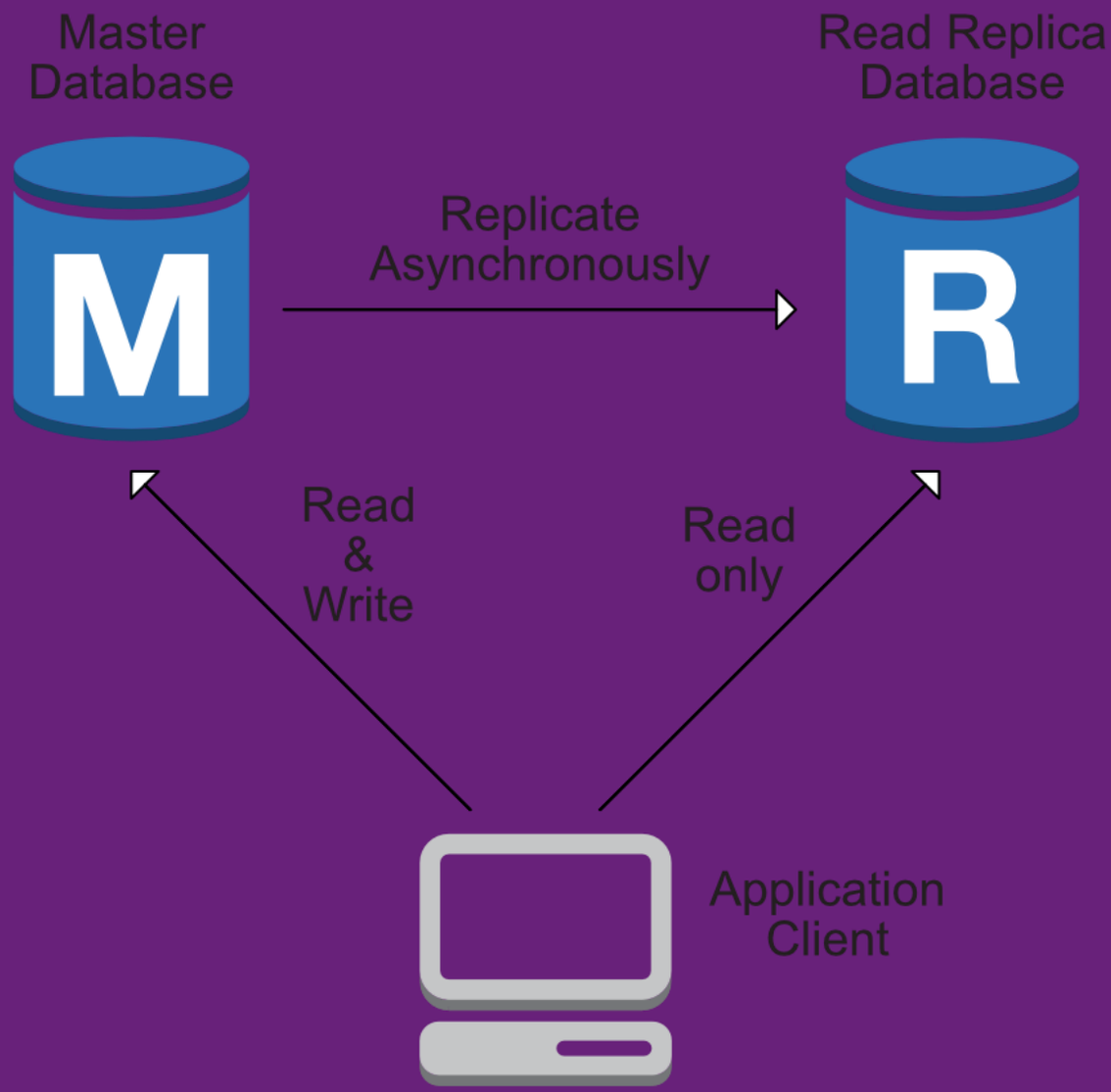
Scale-Out File Server



Scale-Out File Server

- Windows Server 2016 introduced a new service called Storage Spaces Direct (S2D)
- Using Amazon EC2 Windows instances along with S2D solves both problems of durability and scale.
- When tuning for performance, it is important to remember the difference between latency and bandwidth.

Read Replicas

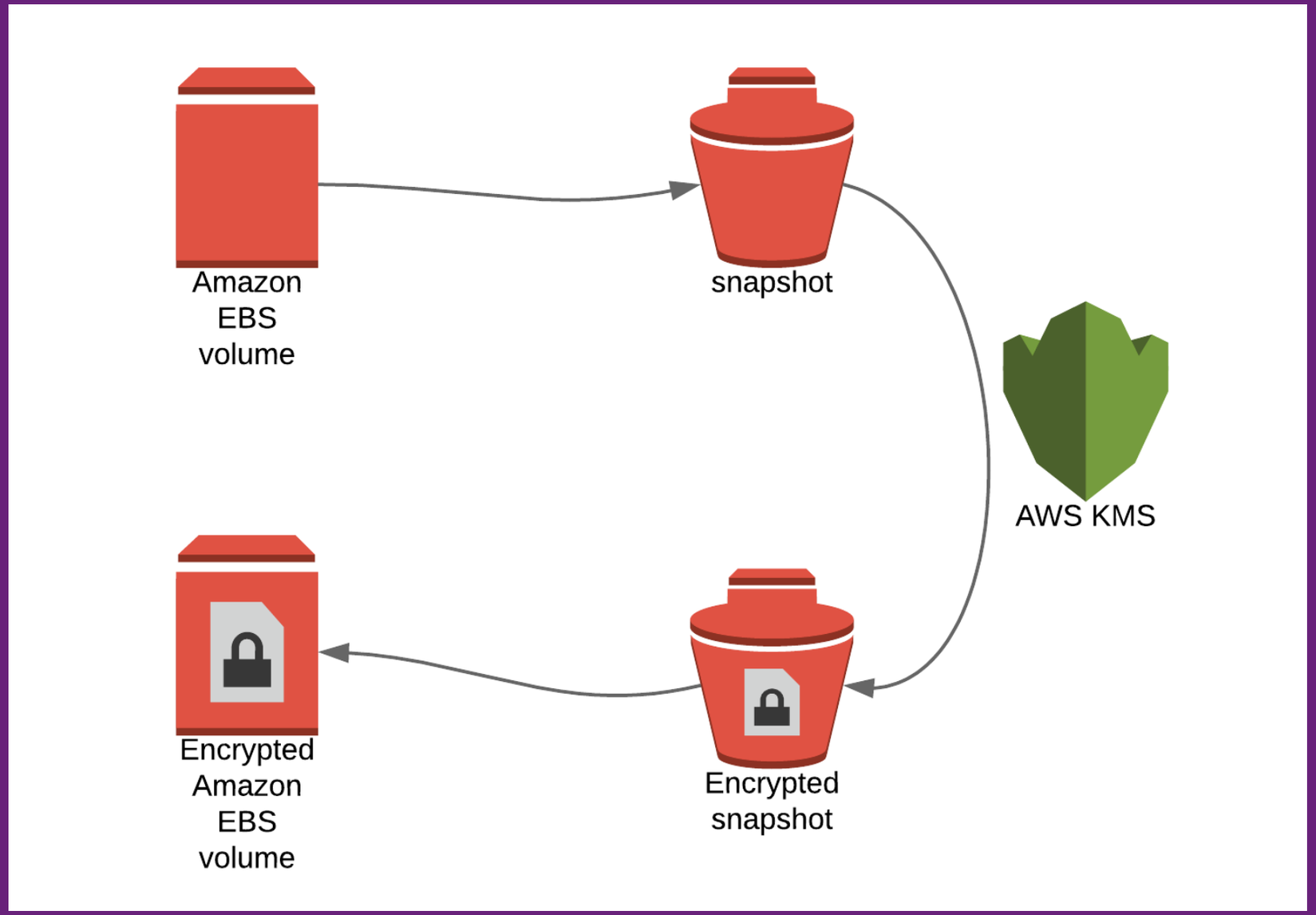


Read Replicas

- You may determine that many of your DB transactions are read-only queries, and that the sheer number of incoming connections is flooding your database.
- When using availability group (AG) listeners, you can mark your connection strings as read-only.
- There may be cases where you have users or applications connecting to your databases from geographically dispersed locations. If latency is a concern, you can locate read replicas in close proximity to your users and applications.

Security Optimization

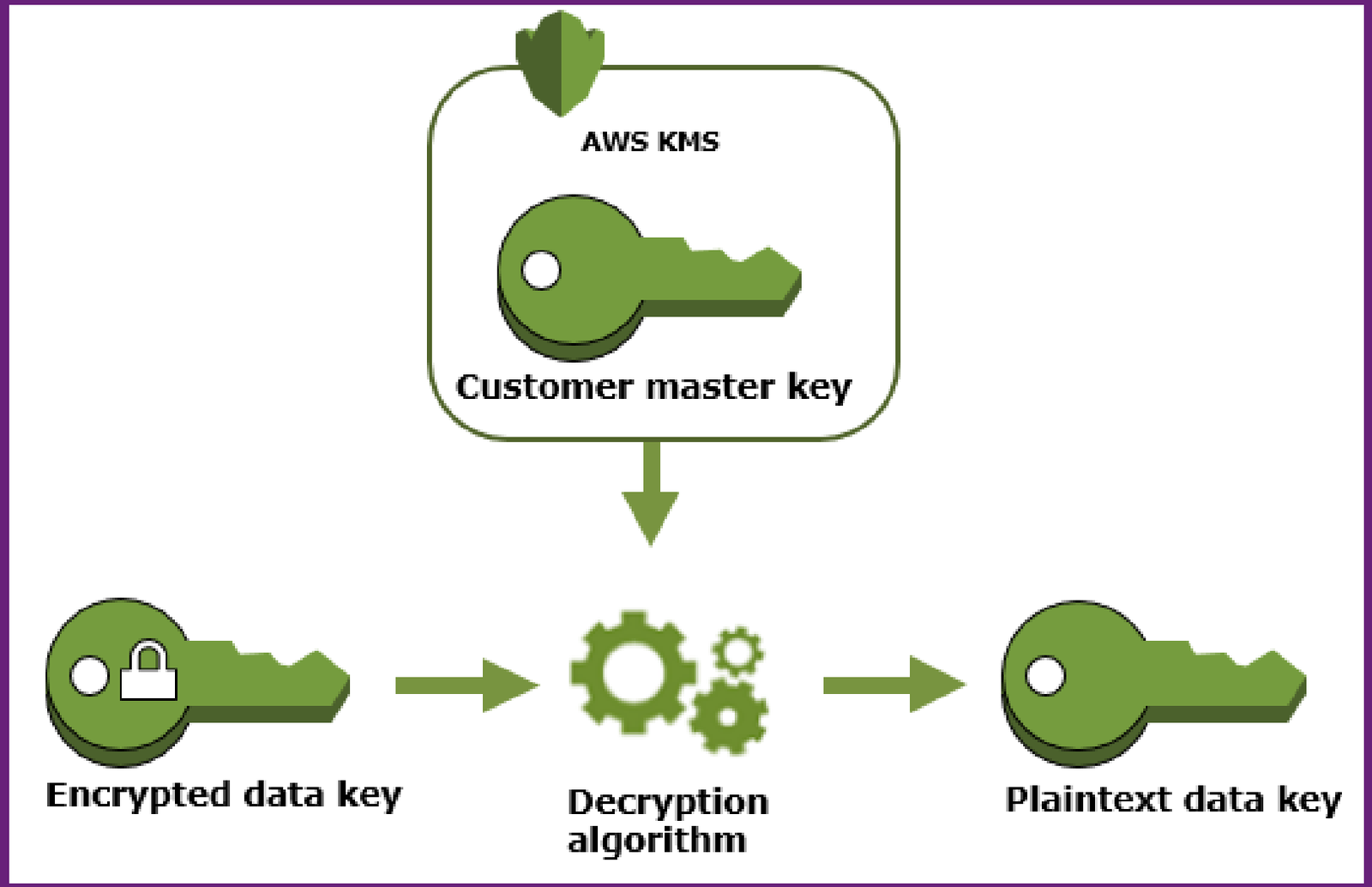
Amazon EBS Encryption



Amazon EBS Encryption

- If you are using EBS volumes to store your SQL Server database files, you have the option to enable block-level encryption.
- Amazon EBS transparently handles encryption and decryption for you. This is available through a simple checkbox, with no further action necessary.

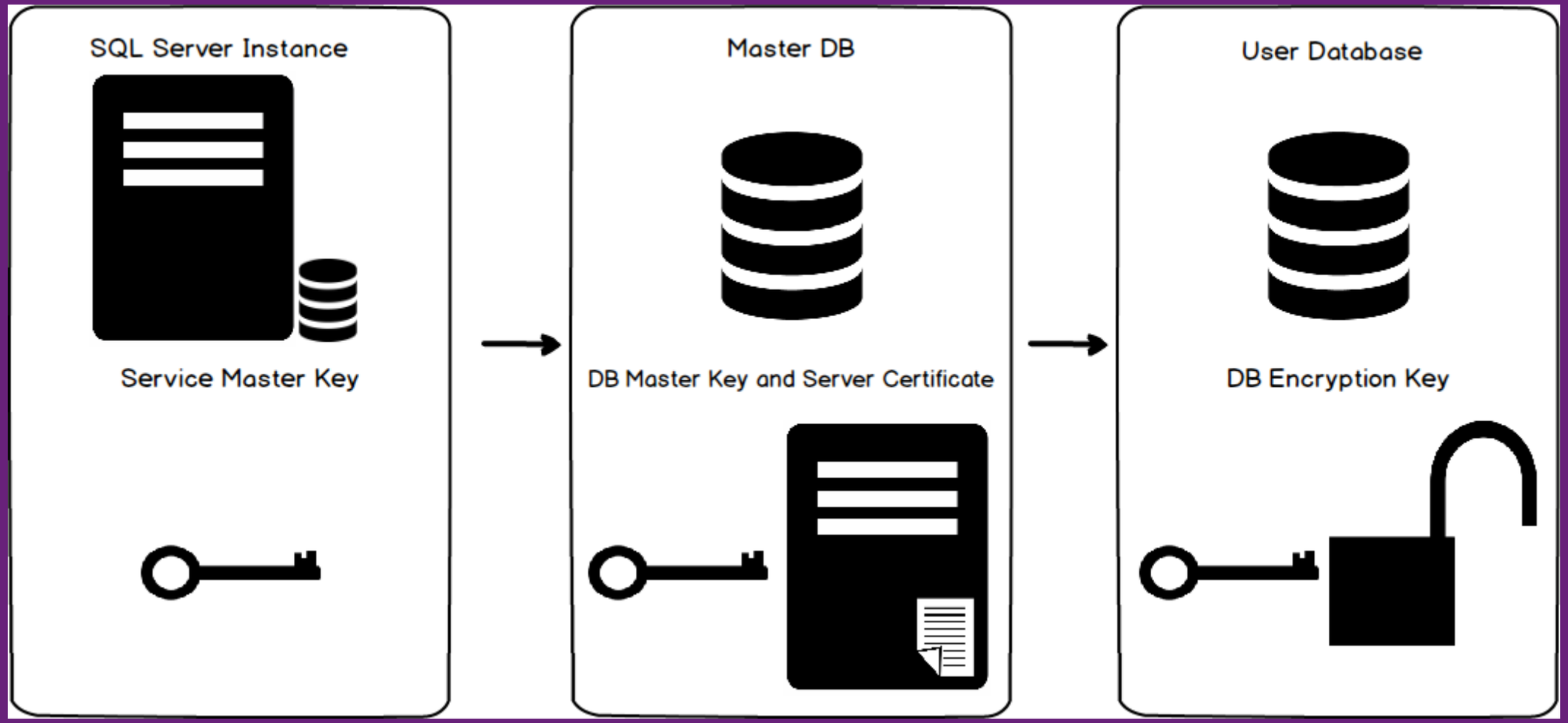
AWS Key Management Service (KMS)



AWS Key Management Service (KMS)

- AWS KMS is a fully managed service to create and store encryption keys.
- You can use KMS-generated keys or bring your own keys.
- In either case, keys never leave KMS and are protected from any unauthorized access.
- You can use KMS keys to encrypt your SQL Server backup files when you store them on Amazon S3, Amazon Glacier, or any other storage service.
- Amazon EBS encryption also integrates with AWS KMS.

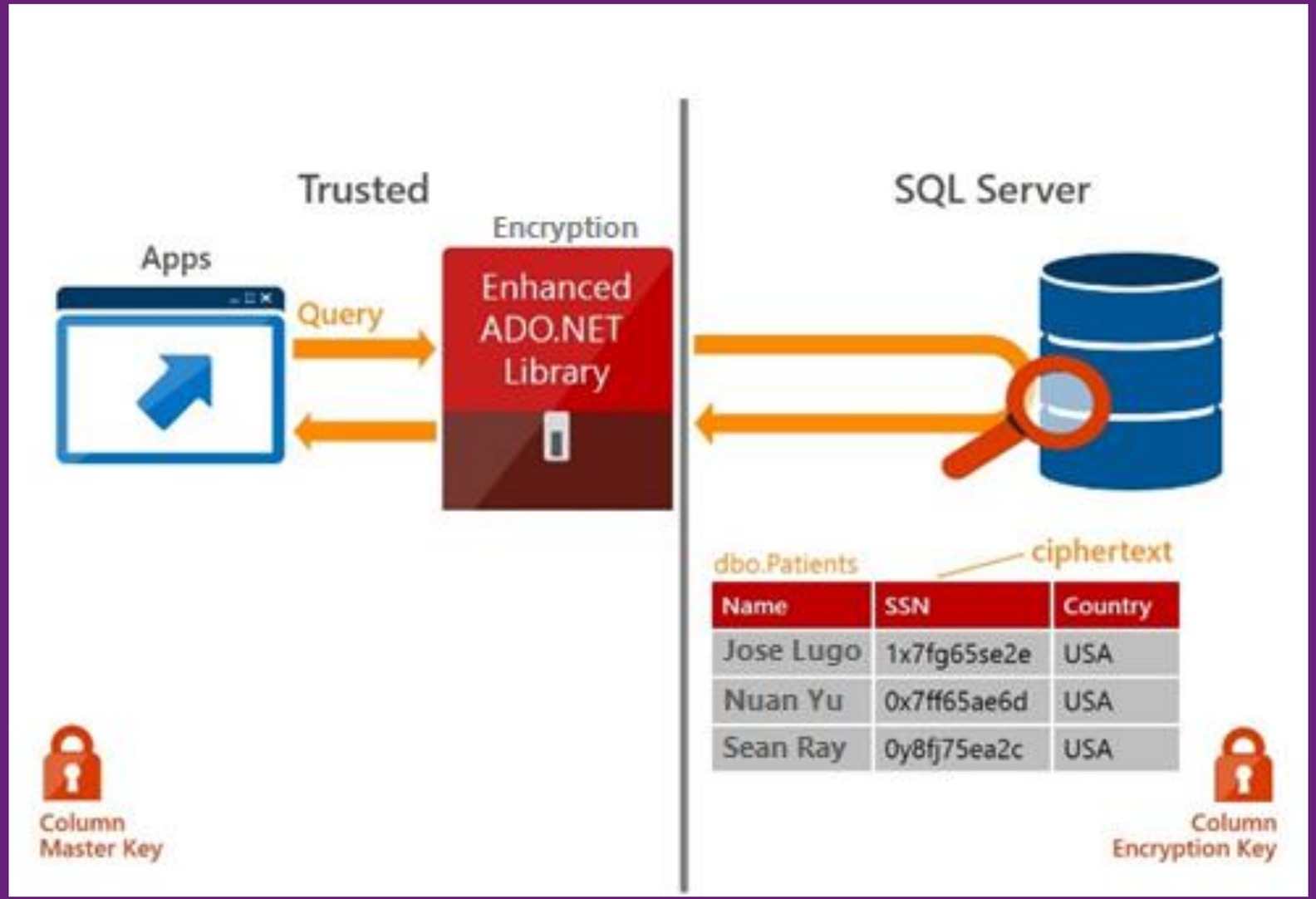
Transparent Data Encryption (TDE)



Transparent Data Encryption (TDE)

- TDE is a feature available in Microsoft SQL Server that provides transparent encryption of your data at rest.
- TDE is available on Amazon RDS for SQL Server, and you can also enable it on your SQL Server workloads on EC2 instances.
- This feature is only available on SQL Server Enterprise Edition.
- However, if you want to have encryption-at-rest for your database files on Standard Edition, you can use EBS encryption instead.

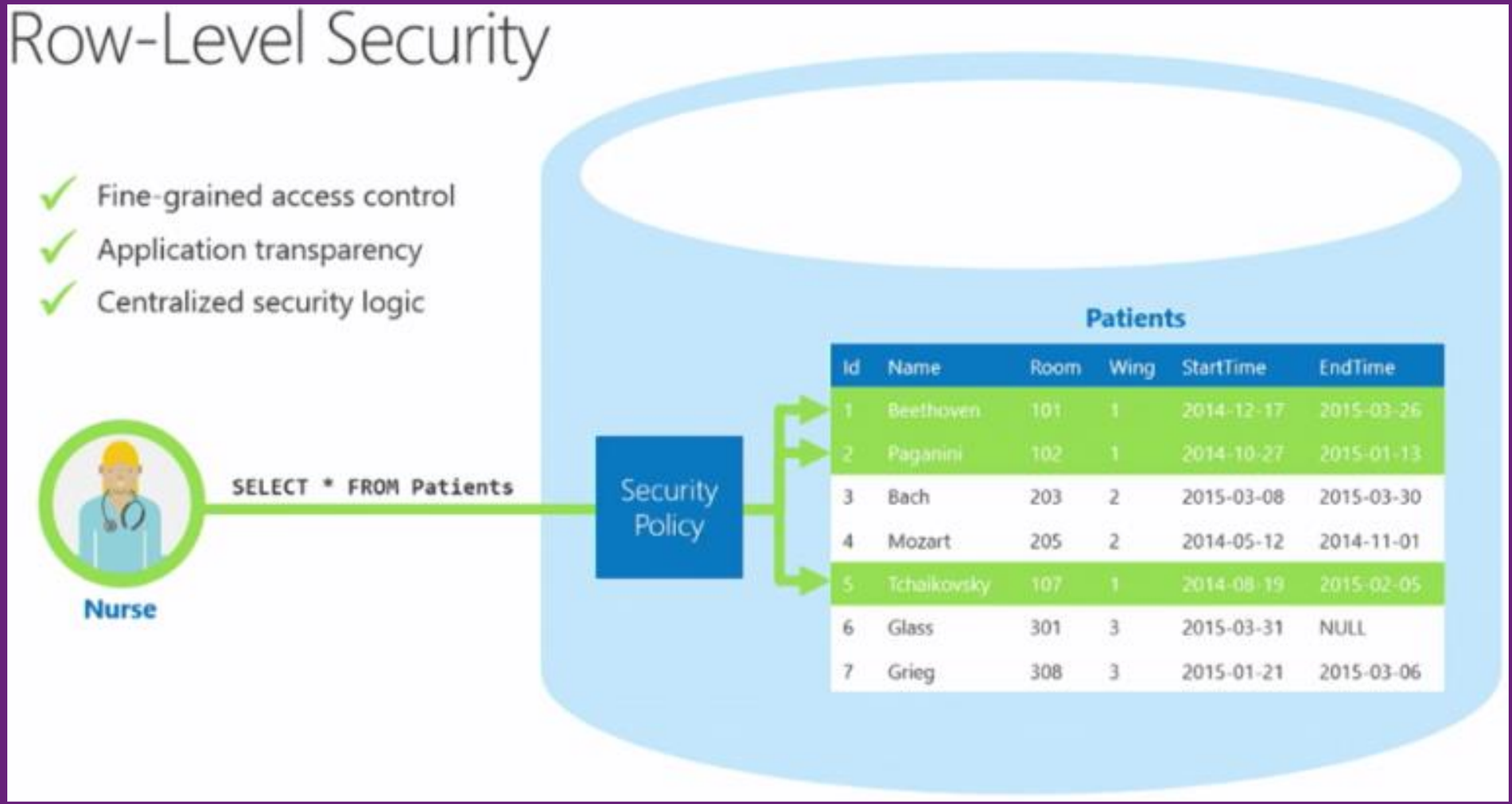
Always Encrypted



Always Encrypted

- Always Encrypted is a feature that allows separation between data owners and data managers.
- Sensitive data that is stored in Microsoft SQL Server using Always Encrypted, stays encrypted even during query processing.
- Encryption keys remain with the data owners and are not revealed to the database engine.
- Available on SQL Server 2016

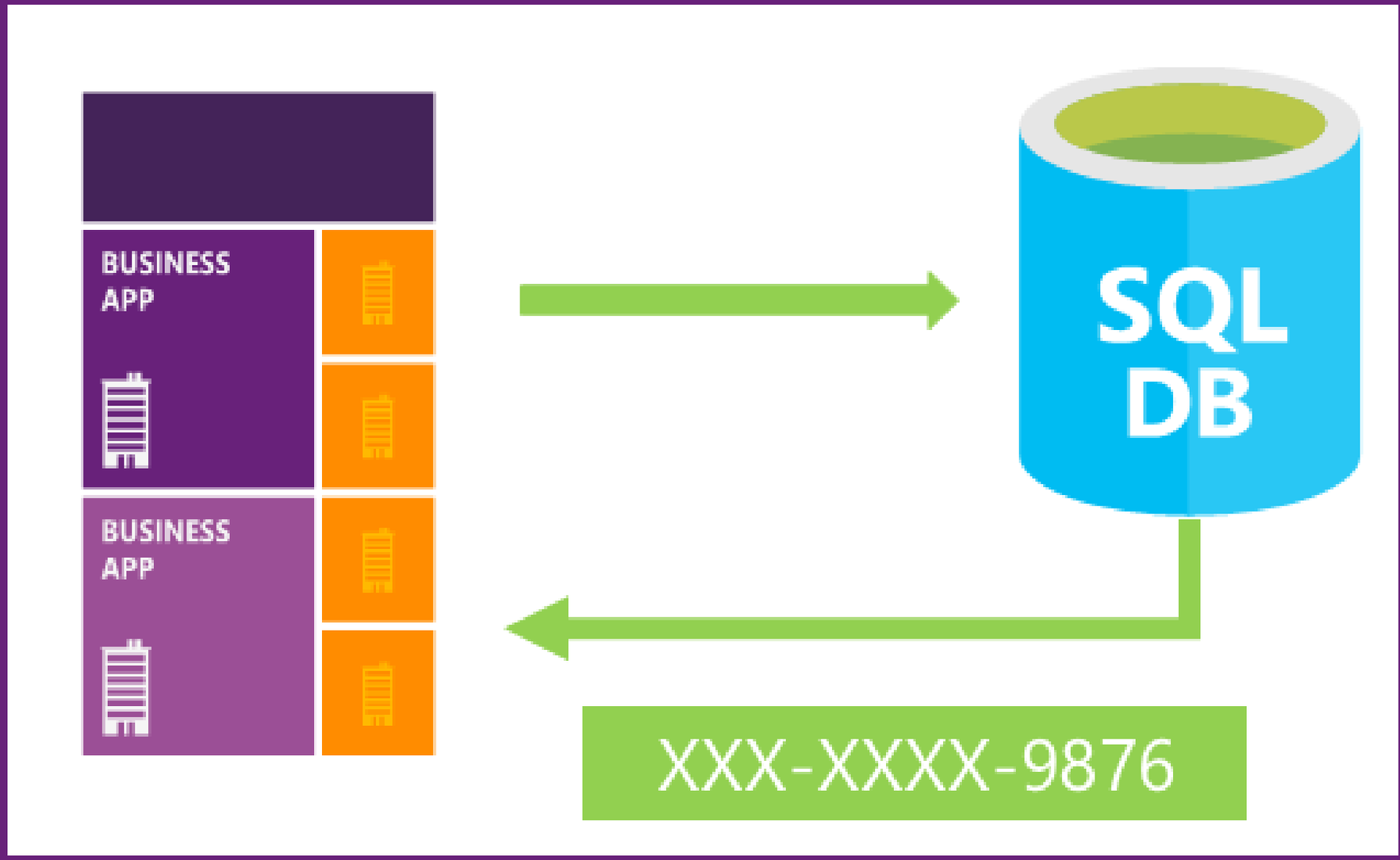
Row-Level Security



Row-Level Security

- Row-Level Security (RLS) in SQL Server enables you to control database access at the row level.
- This feature reduces your attack surface by filtering out all unauthorized access attempts, originating from any layer of your application, directly from the database.
- It could potentially simplify your applications, but you need to design your applications in a way that differentiates users at the database level.

Dynamic Data Masking



Dynamic Data Masking

- Dynamic Data Masking (DDM) is another feature that can simplify design and implementation of security requirements in applications.
- You can use DDM for partially or fully masking certain fields when they are returned as part of query results.
- You can leverage central policies to apply DDM on sensitive data.
- Available on SQL Server 2016

Amazon VPC

Amazon VPC

- There are many features in Amazon VPC that help you to secure your data in transit.
- You can use security groups to restrict access to your EC2 instances and only allow whitelisted endpoints and protocols.
- You can also use network access control lists to blacklist known sources of threats.

Whitelisting

Whitelisting

- You can leverage Windows Server Group Policies to whitelist your SQL Server software, and possibly other known applications, on your EC2 Windows instances.
- This ensures that nothing but your whitelisted applications can run on those servers.
- This is one of the most effective ways to eliminate the possibility of having malware infect your instances.
- This way, anything other than legitimate applications are filtered at the Windows kernel level.

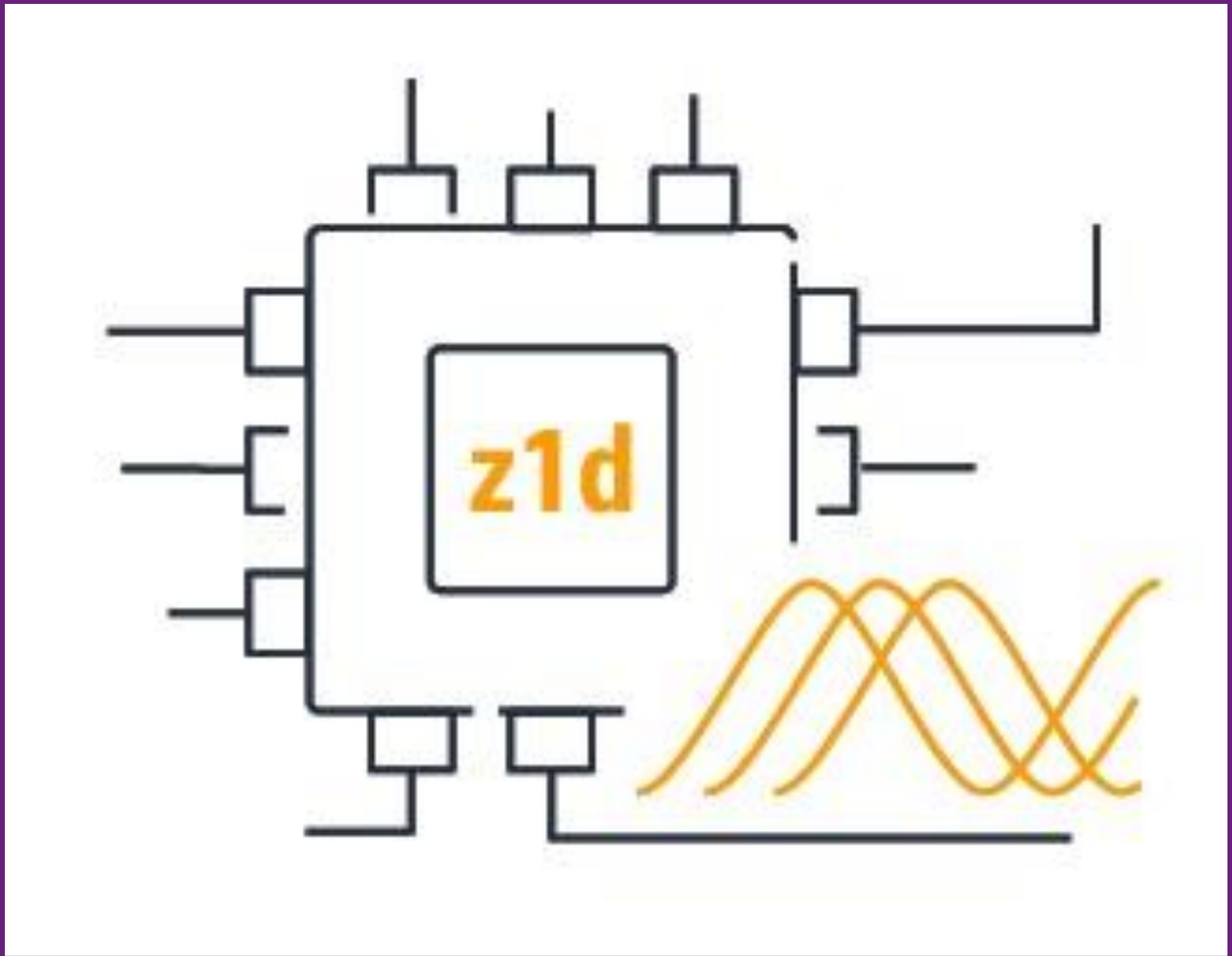
Cost Optimization

Total Cost of Ownership (TCO)

Total Cost of Ownership (TCO)

- Customers running Microsoft SQL Server database workloads on Amazon EC2 can break down their TCO into two components: infrastructure cost and software license cost.
- The infrastructure cost is the cost of the server or EC2 instance type used to run the database (M4, R4, R5, etc.) and will vary based on the amount of compute, memory, networking, and storage available.
- The software license cost is often charged on a per-core basis and varies based on the type of license, with Microsoft SQL Server Enterprise being one of the most expensive options.

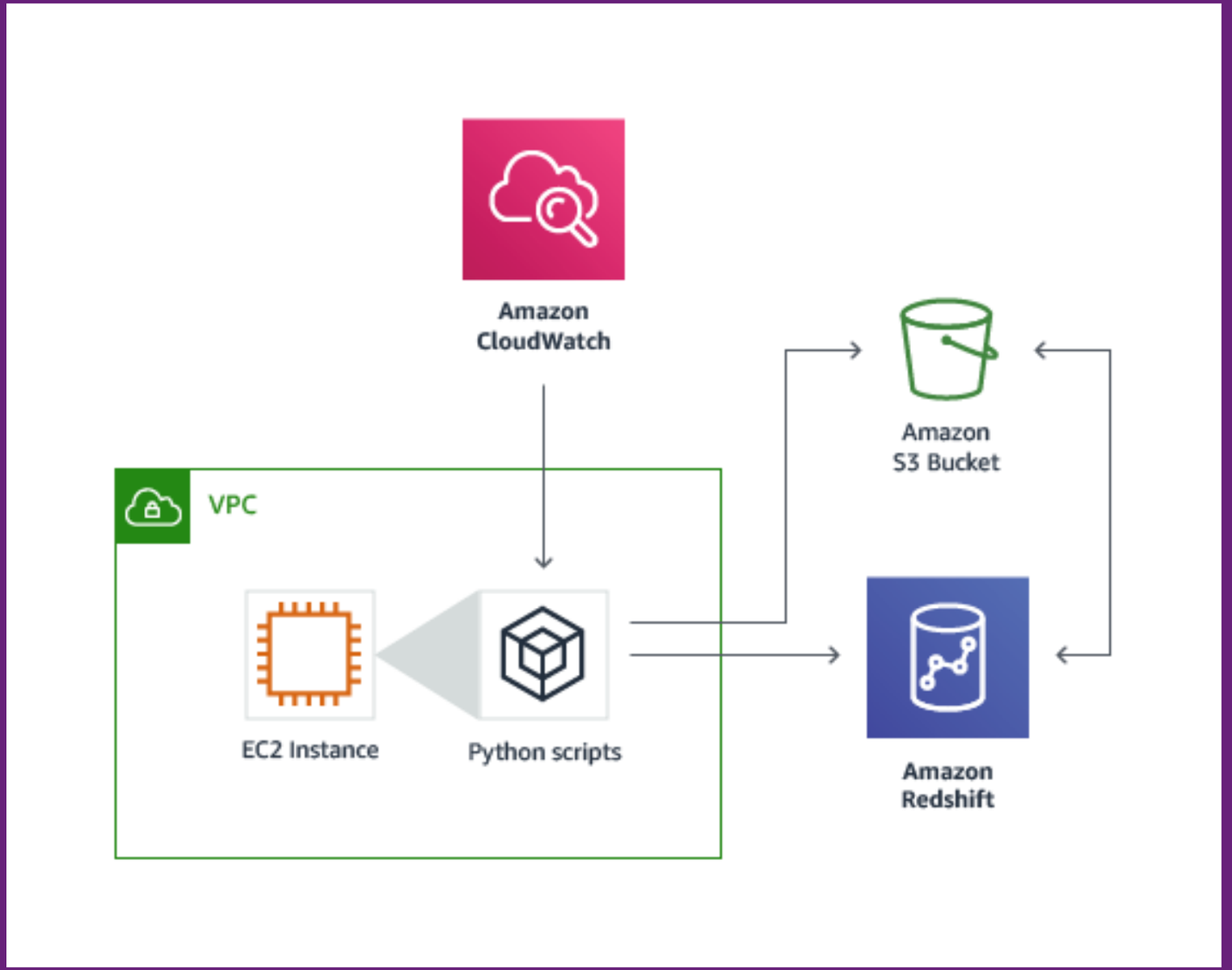
z1d Instance Type



z1d Instance Type

- The high performance z1d instance is optimized for workloads that carry high licensing costs, such as Microsoft SQL Server, and Oracle databases.
- Moving an SQL Server Enterprise workload running on an r4.4xlarge to a z1d.3xlarge can deliver up to 24% in savings as a result of licensing fewer cores

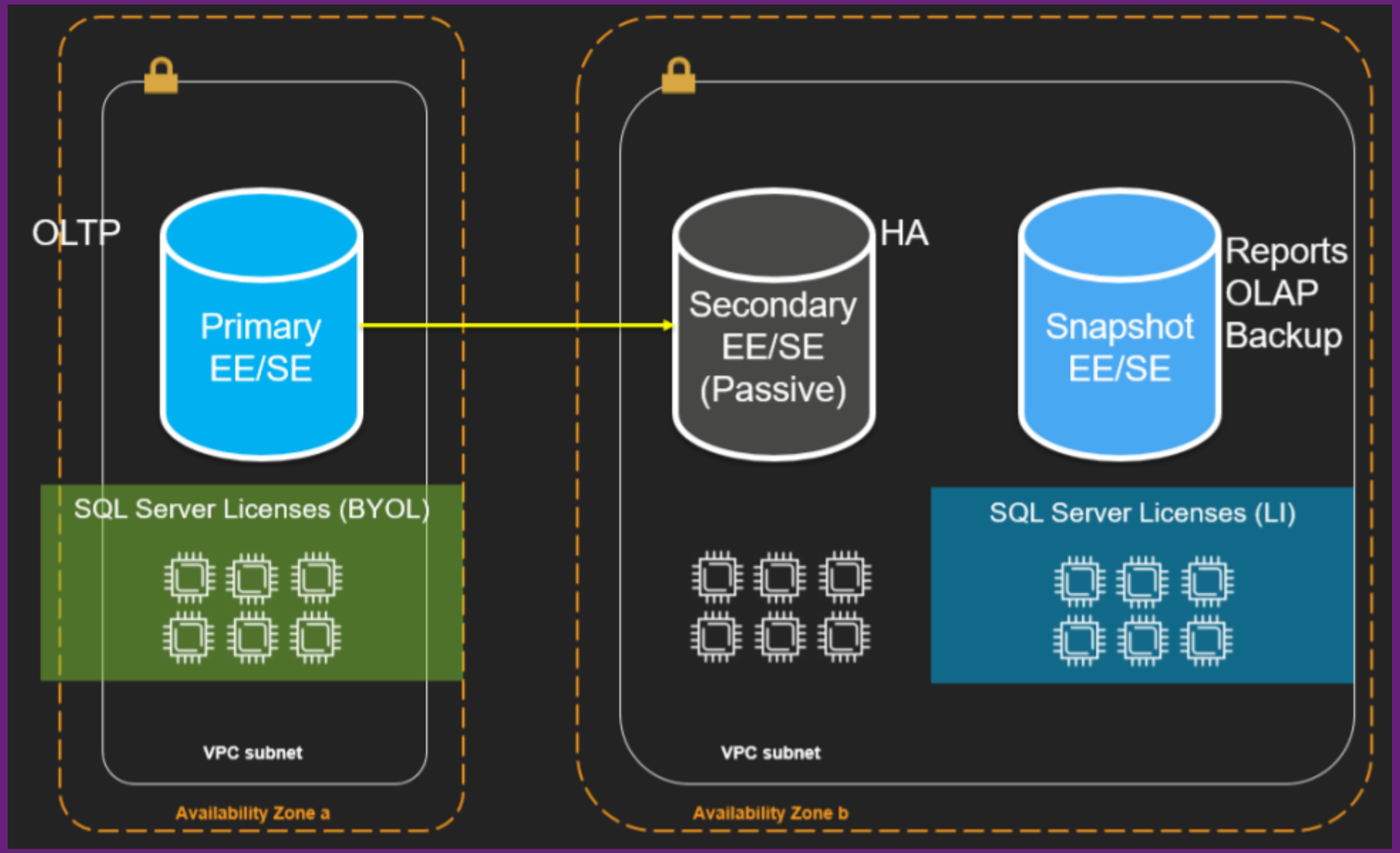
EC2 CPU Optimization



EC2 CPU Optimization

- The z1d instance types provide the maximum CPU power, allowing to reduce number of CPU cores for compute-intensive SQL Server deployments.
- You can use EC2 CPU optimization to reduce number of cores available to an EC2 instance and avoid unnecessary licensing costs.

Eliminating Active Replica licenses

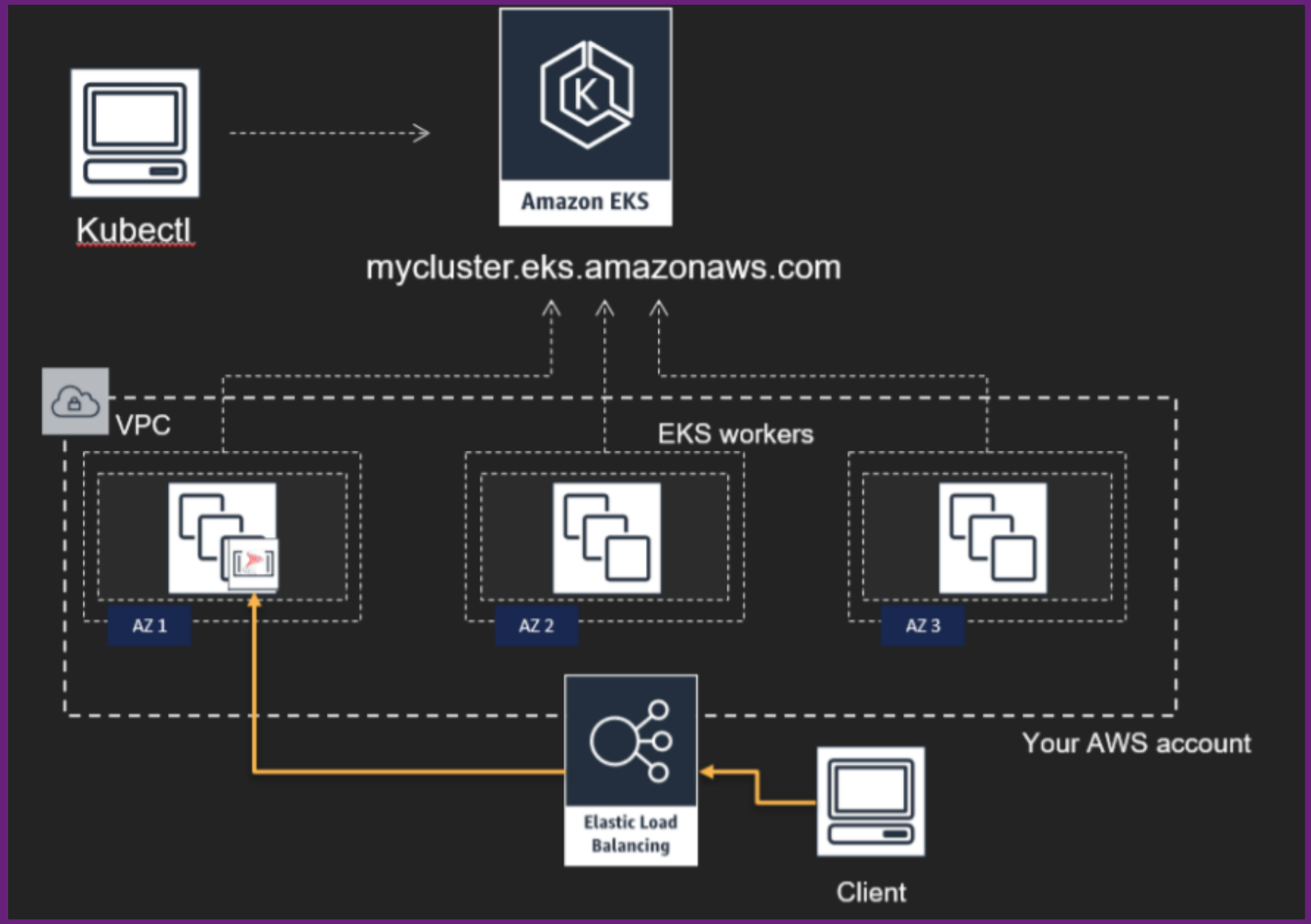


Eliminating Active Replica licenses

- One of the best opportunities for cost optimization in the cloud is through applying a combination of BYOL and LI models.
- A common use case is SQL Server Always On Availability Groups with active replicas.
- Active replicas are used primarily for:

  Reporting, Backup, OLAP Batch jobs, HA
- Out of the above four operations, the first three are often performed intermittently.

SQL Server inside Docker Container



SQL Server inside Docker Container

- Although containerization is a new feature introduced in SQL Server 2017, but instances of SQL Server 2017 can be used to host databases with compatibility levels of earlier versions.
- Businesses who wish to use BYOL with SQL Server licenses on AWS, often have Software Assurance available.
- SA gives them the benefit of license mobility, as well as upgrade to SQL Server 2017.

Q&A?