



Why you shouldn't really..



<https://learn.cantrill.io>



adriancantrill

- **Admin overhead** - managing EC2 and DBHost
- **Backup / DR Management**
- EC2 is **single AZ**
- **Features** - some of AWS DB products are amazing
- EC2 is **ON** or **OFF** - no serverless, no easy scaling
- **Replication** - skills, setup time, monitoring & effectiveness
- **Performance**....AWS invest time into optimisation & features

and implement it on EC2,



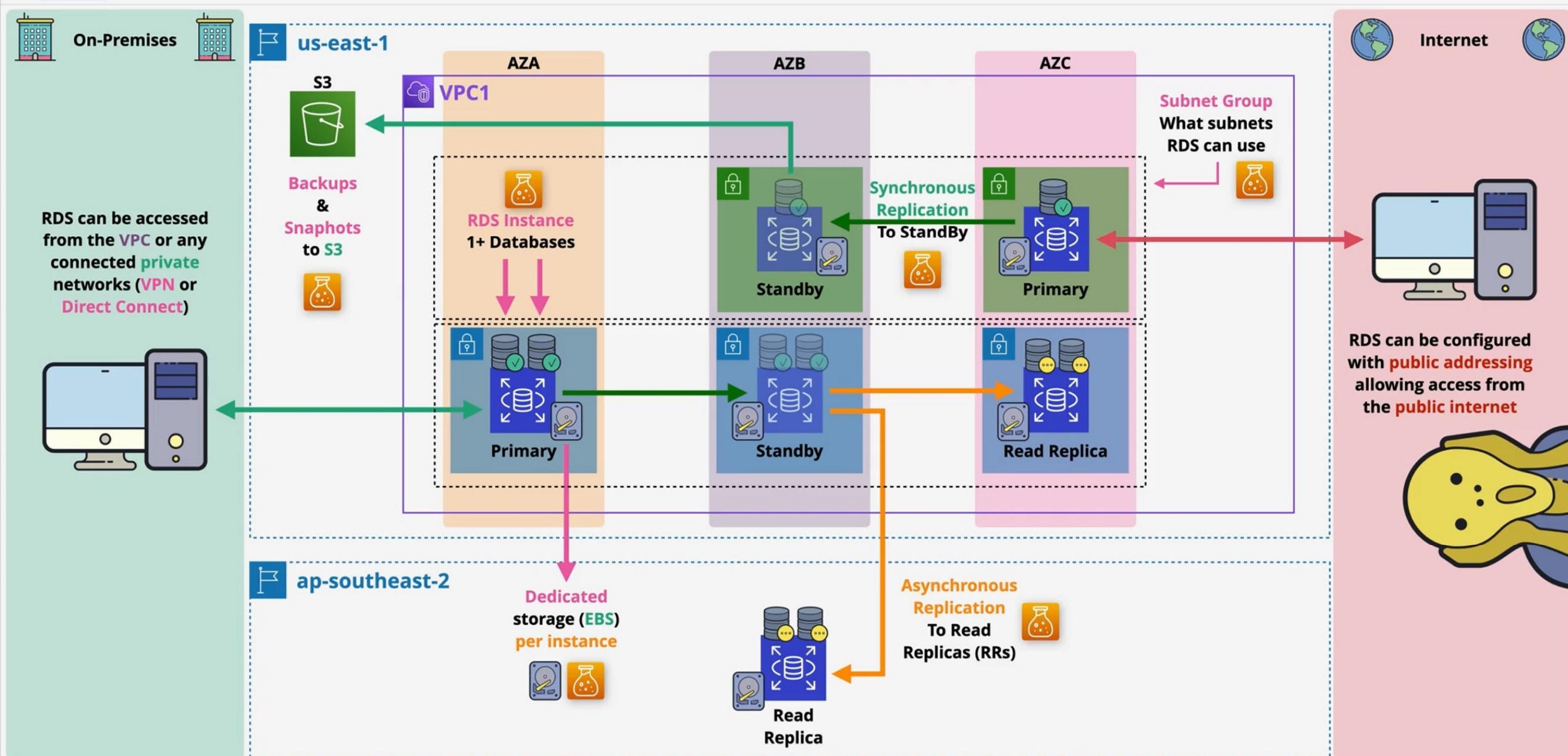
RDS - Architecture



<https://learn.cantrill.io>

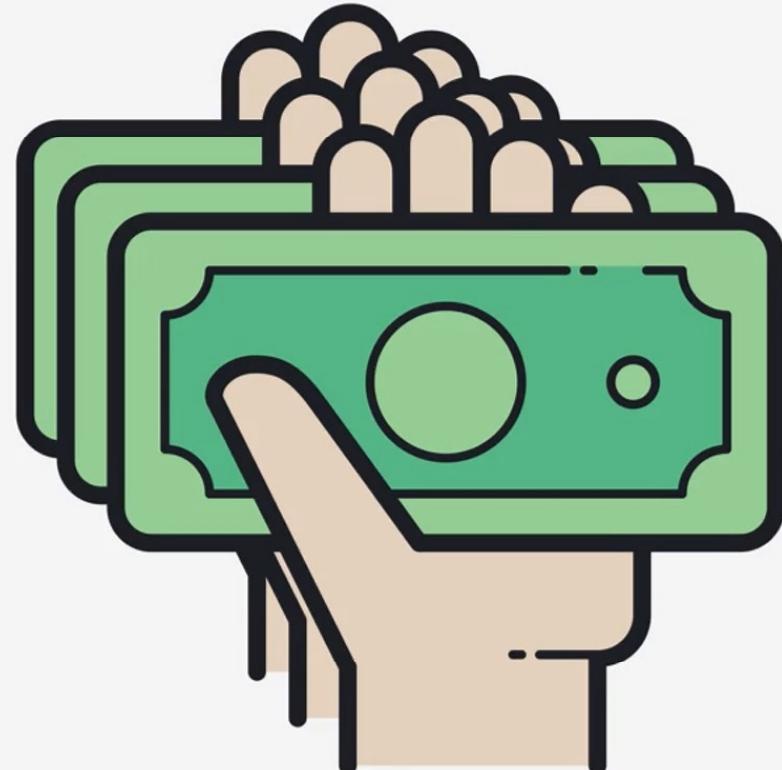


adriancantrill





- Cost #1 - **Instance Size & Type**
- Cost #2 - **Multi AZ** or not
- Cost #3 - **Storage type & amount**
- Cost #4 - **Data** Transferred
- Cost #4 - Backups & Snapshots
- Cost #5 - Licensing* (if applicable)





- Data => Primary **AND** Replicated to StandBy = **Committed (Synchronous)**
- **Not free tier** ... extra cost for replica
- **ONE** StandBy replica **ONLY**
- .. which **can't be used** for reads or writes
- **60-120** seconds failover
- **Same region only** ... different AZs in the same region
- **Backups** taken from **standby** to **improve performance**
- AZ Outage, Primary Failure, Manual failover, Instance type change and software patching



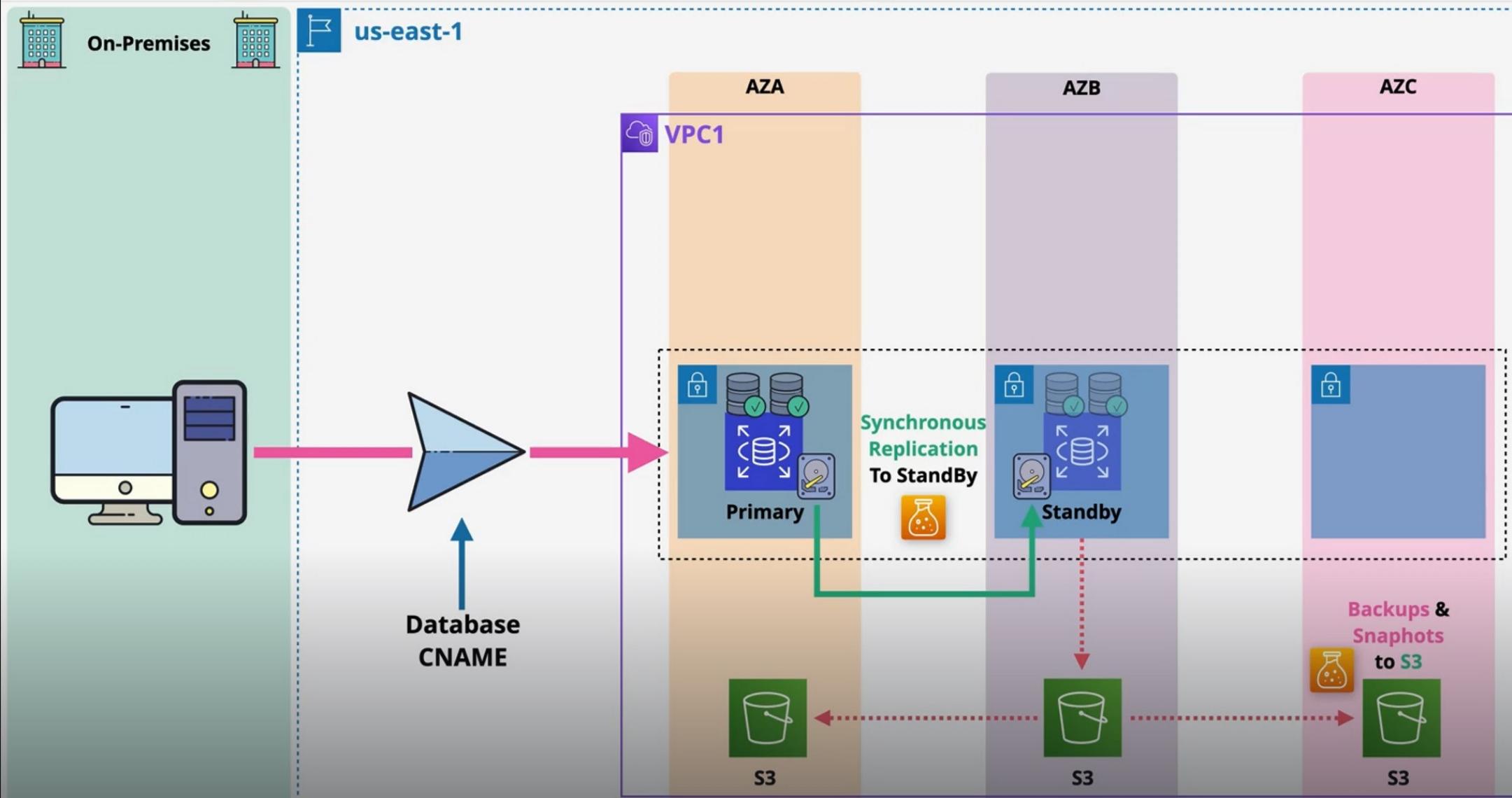
RDS - Multi AZ - Instance



<https://learn.cantrill.io>



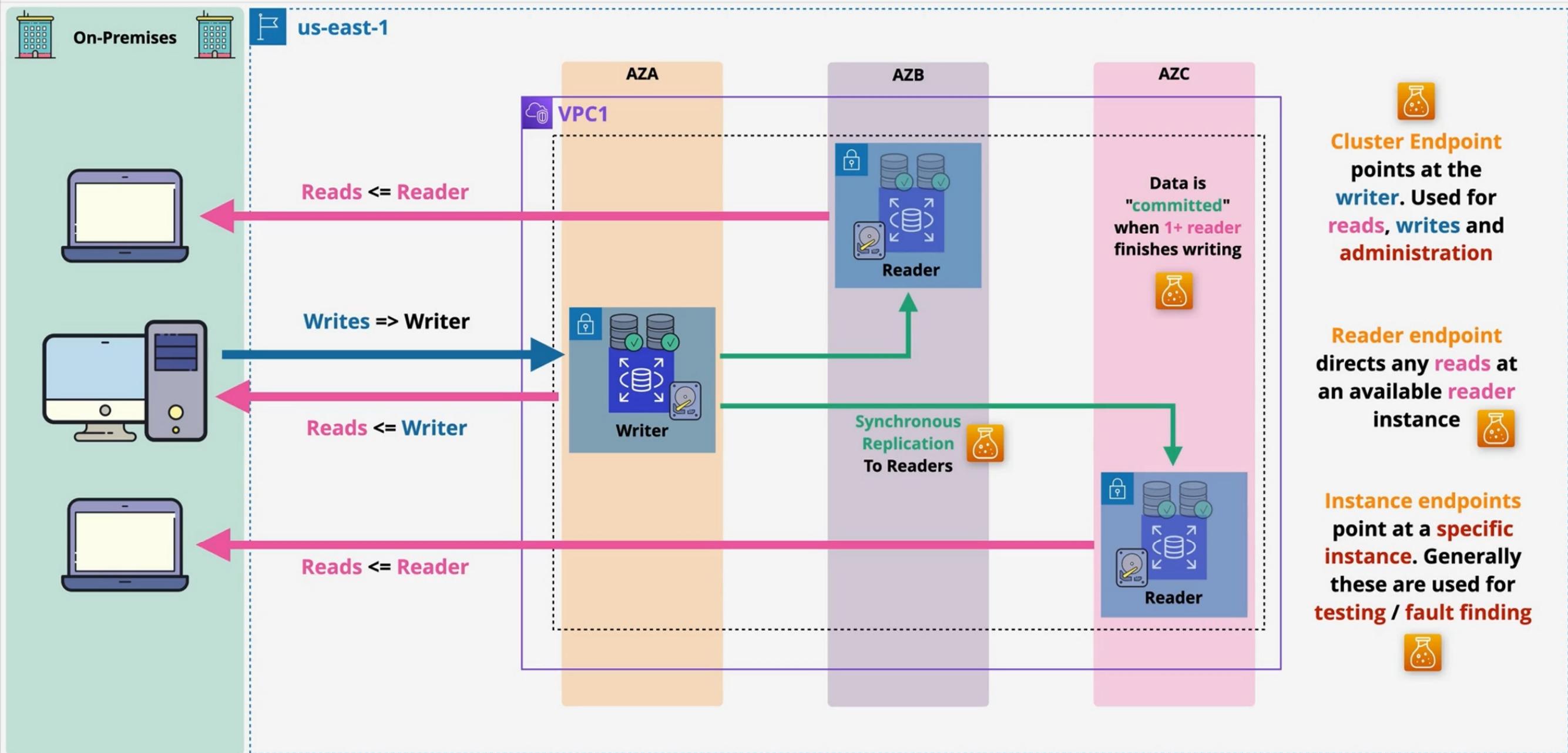
adriancantrill



RDS - Multi AZ - Cluster

<https://learn.cantrill.io>

adriancantrill



- **1 Writer** and **2 Reader** DB instances (different AZs)
- Much faster hardware, Graviton + local **NVME SSD Storage**
- Fast writes to **local storage => flushed to EBS**
- **Readers** can be used for **reads** ... allowing some **read scaling**
- Replication is via transaction logs ... more efficient
- Failover is faster **~35s** + **transaction log apply**
- Writes are "**committed**" when **1 reader** has confirmed



RDS - Backups - General

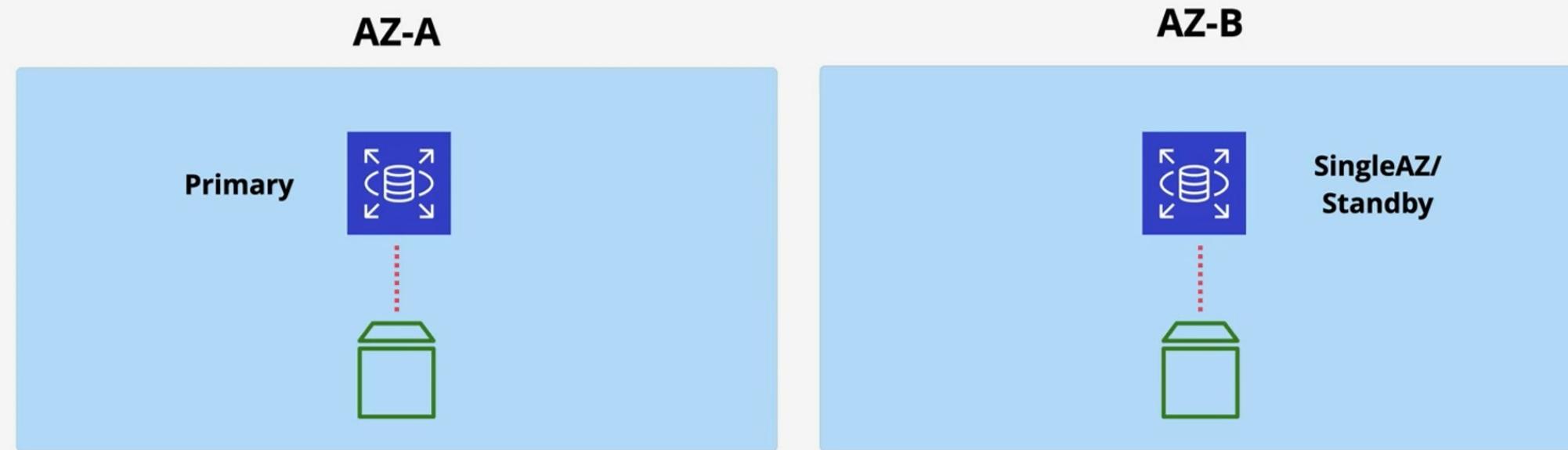


<https://learn.cantrill.io>



adriancantrill

F us-east-1



Automated Backups

Snapshots

AWS Managed S3 Buckets





RDS - Backups - General



<https://learn.cantrill.io>



adriancantrill



us-east-1

First Snap is FULL

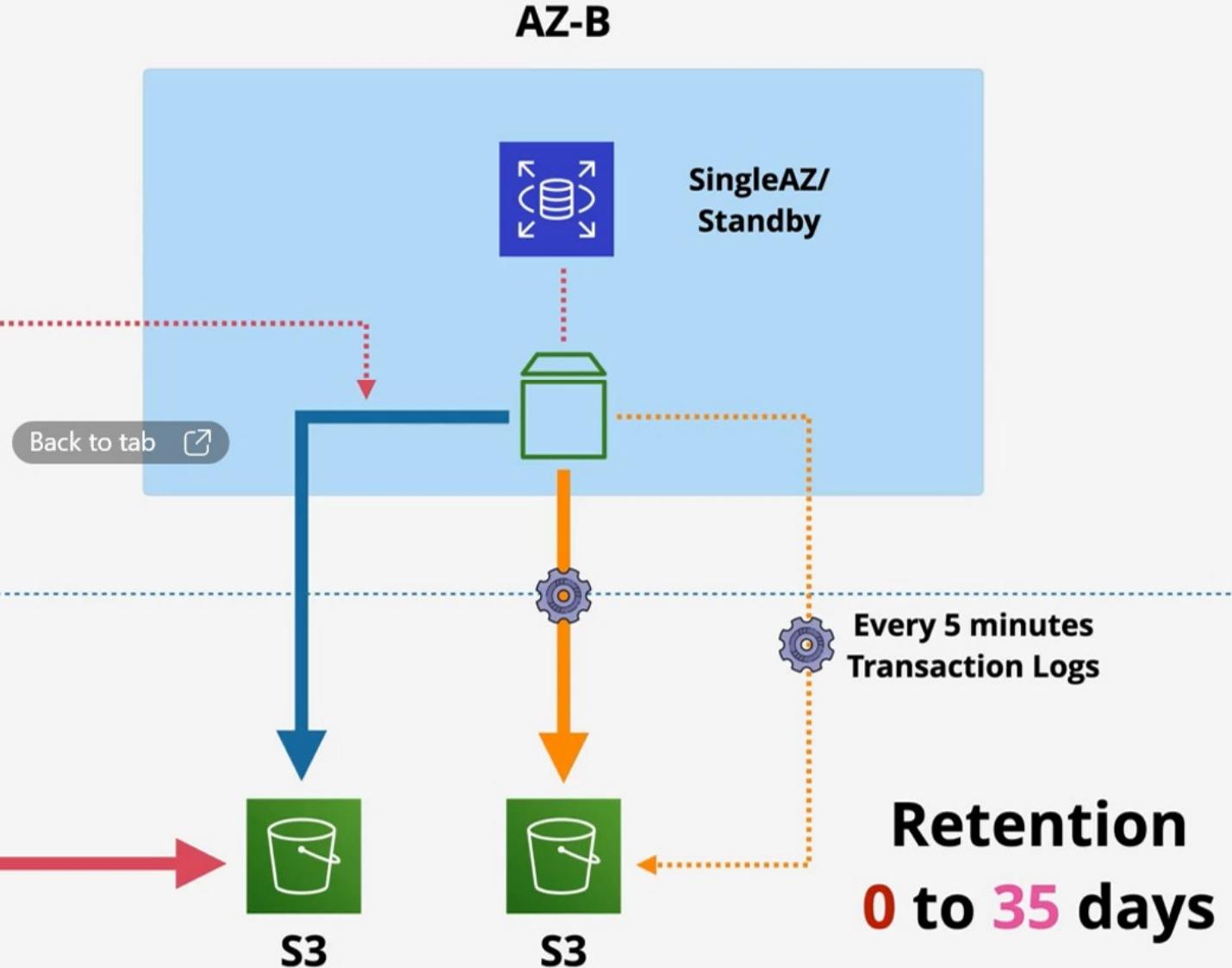
Size of consumed data

Then onward = Incremental

Automated Backups

Manual Snapshots

AWS Managed S3 Buckets





RDS - Backups - Cross-Region



<https://learn.cantrill.io>



adriancantrill



- RDS can **replicate** backups to **another region**
- .. both **snapshots** and **transaction logs**
- Charges apply for the **cross-region data copy**
- ... and the **storage** in the **destination region**
- **NOT DEFAULT** ..configured  within automated backups



- Creates a **NEW RDS Instance - new address**
- Snapshots = **single point in time**, creation time
- Automated = **any 5 minute point in time**
- Backup is restored and transaction logs are 'replayed' to bring DB to desired point in time (**GOOD RPO**)
- Restores **aren't fast** - Think about **RTO (*RR's)**



(read) Performance Improvements



<https://learn.cantrill.io>



adriancantrill

- **5x** direct read-replicas per DB instance
- Each providing an **additional instance of read performance**
- Read-Replicas can have read-replicas - **but lag starts to be a problem.**
- **Global** performance improvements



RPO/RTO Improvements



<https://learn.cantrill.io>



adriancantrill

- Snapshots & Backups Improve **RPO**
- **RTO's are a problem**
- RR's offer **nr. 0 RPO**
- RR's can be **promoted quickly - low RTO**
- **Failure only** - watch for data corruption
- **Read only - until promoted**
- **Global availability improvements ... global resilience**



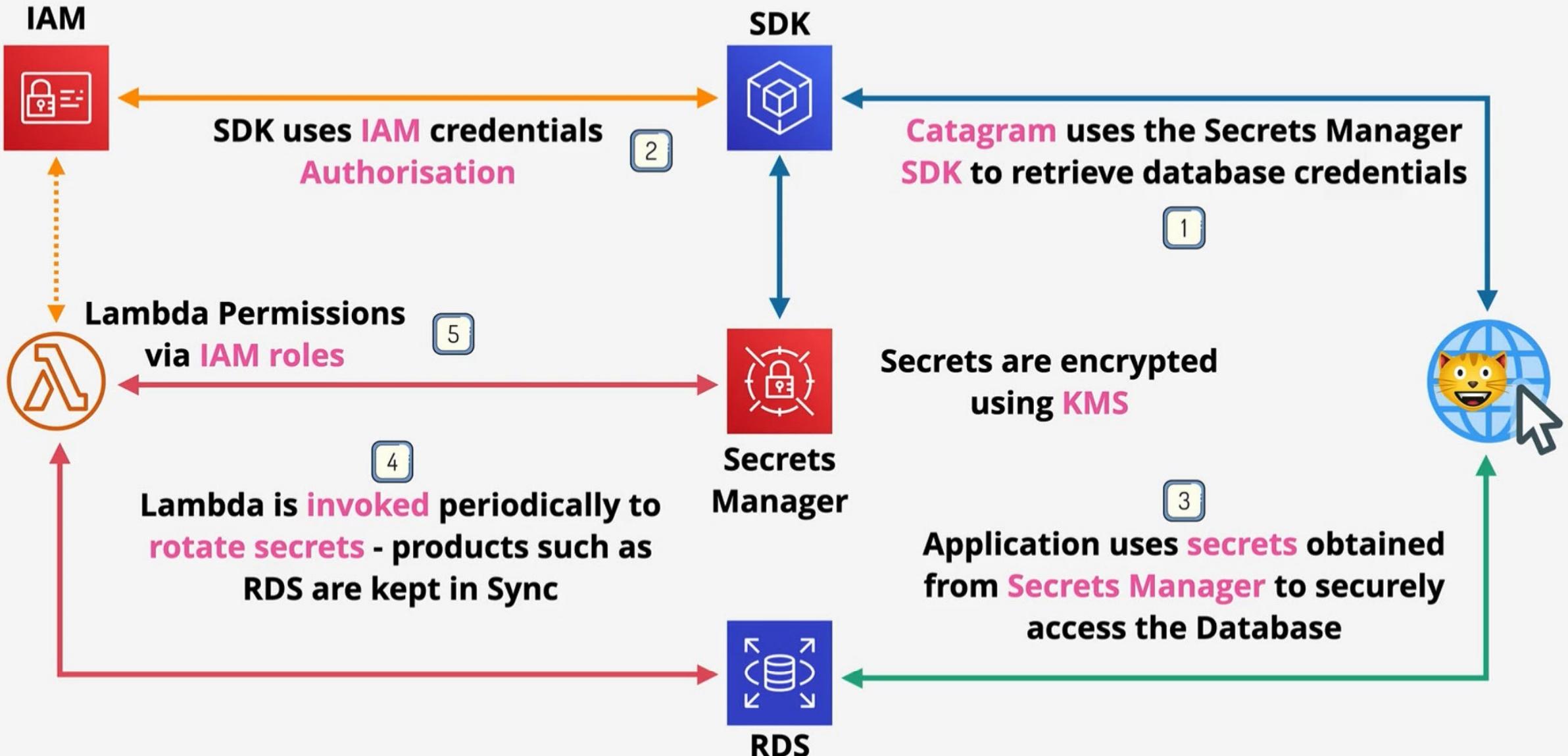
AWS Secrets Manager



<https://learn.cantrill.io>



adriancantrill





AWS Secrets Manager



<https://learn.cantrill.io>



adriancantrill

- It **does share functionality** with Parameter Store
- Designed for **secrets** (.. passwords, API KEYS..)
- Usable via **Console, CLI, API** or **SDK's** (*integration*)
- Supports **automatic rotation** ... this uses **lambda**
- Directly **integrates** with some AWS products (..**RDS**)





(DMS) & Snowball



<https://learn.cantrill.io>



adriancantrill

- Larger migrations might be multi-TB in size
- ... moving data over networks takes time and consumes capacity
- DMS can utilise snowball ...
- **Step 1** : Use SCT to extract data locally and move to a snowball device
- **Step 2** : Ship the device back to AWS. They load onto an S3 bucket.
- **Step 3** : DMS migrates from S3 into the target store
- **Step 4** : Change Data Capture (CDC) can capture changes, and via S3 intermediary they are also written to the target database



Schema Conversion Tool (**SCT**)



<https://learn.cantrill.io>



adriancantrill

- SCT is used when converting **one database** engine to **another**.



- .. including **DB** -> **S3** (Migrations using DMS)

- SCT is **not used when migrating between DB's of the same type**



- .. On-premises MySQL -> RDS MySQL

- Works with **OLTP** DB Types (MySQL, MSSQL, Oracle)

- And **OLAP** (Teradata, Oracle, Vertica, Greenplum..)

- e.g. On-Premises **MSSQL** -> RDS **MySQL**

- e.g. On-premises **ORACLE** -> **Aurora**



Database Migration Service (DMS)



<https://learn.cantrill.io>

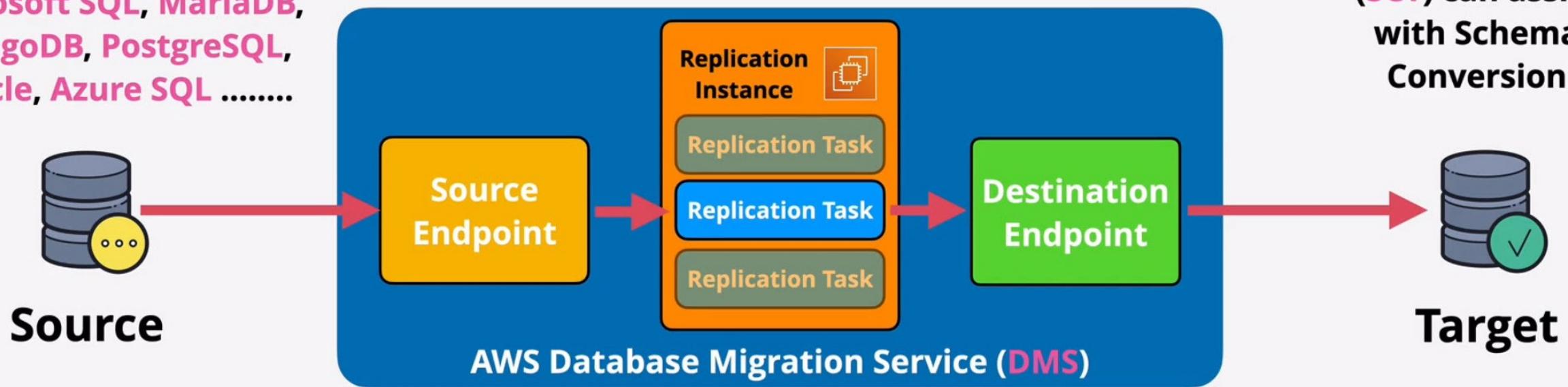


adriancantrill

Common DB Support
MySQL, Aurora,
Microsoft SQL, MariaDB,
MongoDB, PostgreSQL,
Oracle, Azure SQL

Replication instance performs the migration between
Source and Destination endpoints which store
connection information for source and target databases

Schema
Conversion Tool
(SCT) can assist
with Schema
Conversion



Jobs can be **Full load** (one off migration of all data), **Full Load + CDC** (Change data capture) for ongoing replication which captures changes or **CDC Only** (if you want to use an alternative method to transfer the bulk DB data ... such as **native tooling**)



Database Migration Service (DMS)



<https://learn.cantrill.io>



adriancantrill

- A managed database migration service
- Runs using a **replication instance**
- **Source** and **Destination Endpoints** point at ...
- **Source** and **Target** Databases
- **One endpoint MUST be on AWS**



RDS - Proxy - Key Facts



<https://learn.cantrill.io>



adriancantrill

- Fully Managed DB Proxy for RDS/Aurora
- ... auto scaling, highly available by default
- Provides connection pooling - reduces DB Load
- ONLY accessible from a VPC
- Accessed via Proxy Endpoint - no app changes
- Can enforce SSL/TLS
- Can reduce failover time by over 60%
- Abstracts failure away from your applications



RDS - Proxy - When?



<https://learn.cantrill.io>



adriancantrill

- Too many connections errors...
- DB Instances using T2/T3 (i.e smaller/burst) instances
- AWS Lambda ... time saved/connection reuse & IAM Auth
- Long running connections (SaaS apps) - low latency
- Where resilience to database failure is a priority...
- ... RDS proxy can reduce the time for failover
- ... and make it transparent to the application



RDS - Proxy - Architecture



<https://learn.cantrill.io>



adriancantrill

us-east-1

RDS Proxy => DB Instance connections
can be reused ... avoiding the lag of
establishment, usage & termination
for each invocation



Abstracts client away
from DB failure or
failover events

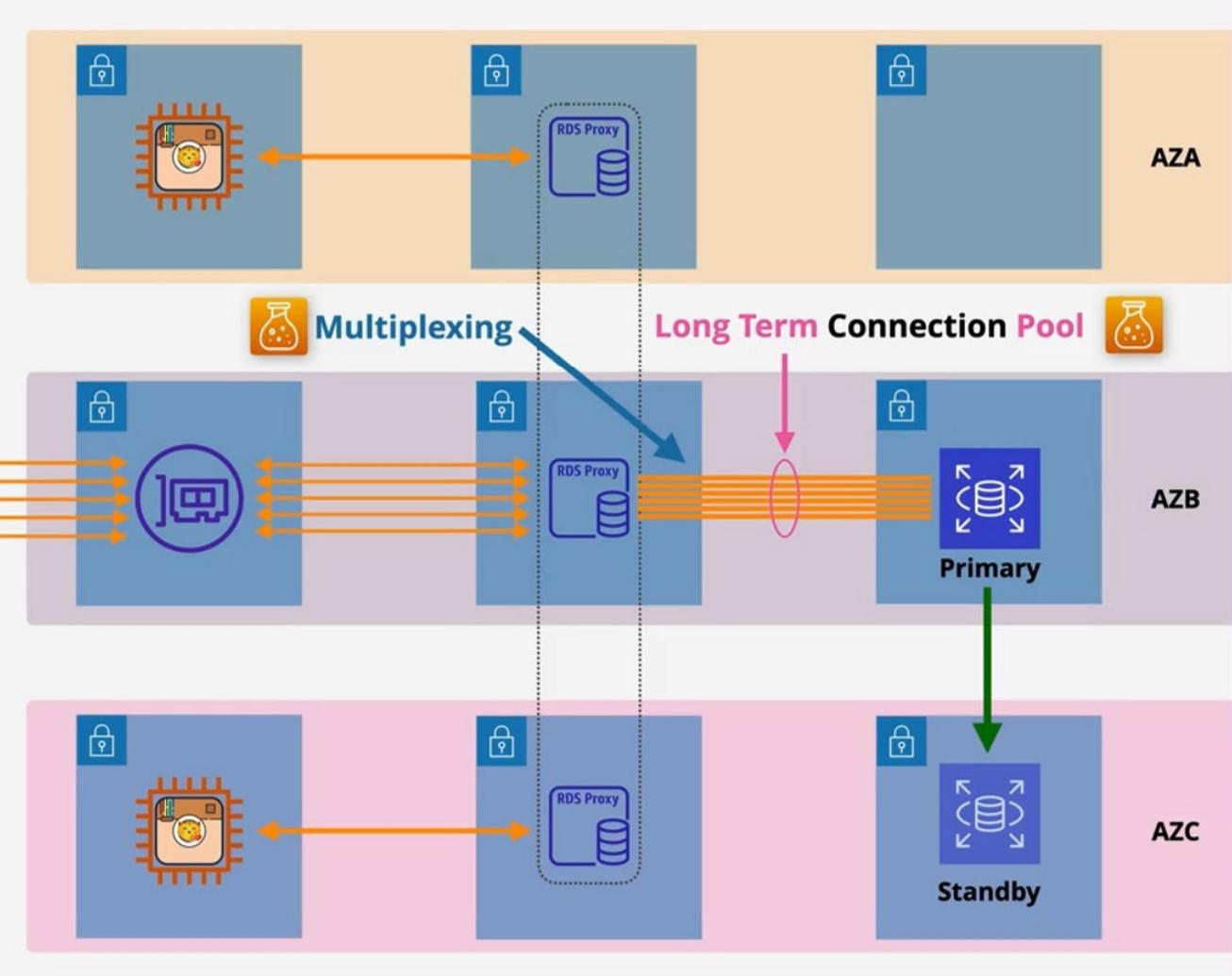


Client => Proxy connection is
established and WAITS even if
the Target DB is unresponsive



Each Lambda function connects to the Proxy
This is MUCH quicker to establish vs direct to database

VPC1





RDS - Proxy - Why?



<https://learn.cantrill.io>



adriancantrill

- Opening and Closing Connections consume resources
-It takes time .. which creates latency
- With serverless .. every lambda opens and closes ?
- Handling failure of Database instances is hard ...
- .. doing it within your application adds risks
- DB Proxies help ... managing them is not trivial (scaling/resilience)
- Application(s) => Proxy (connection pooling) => Database



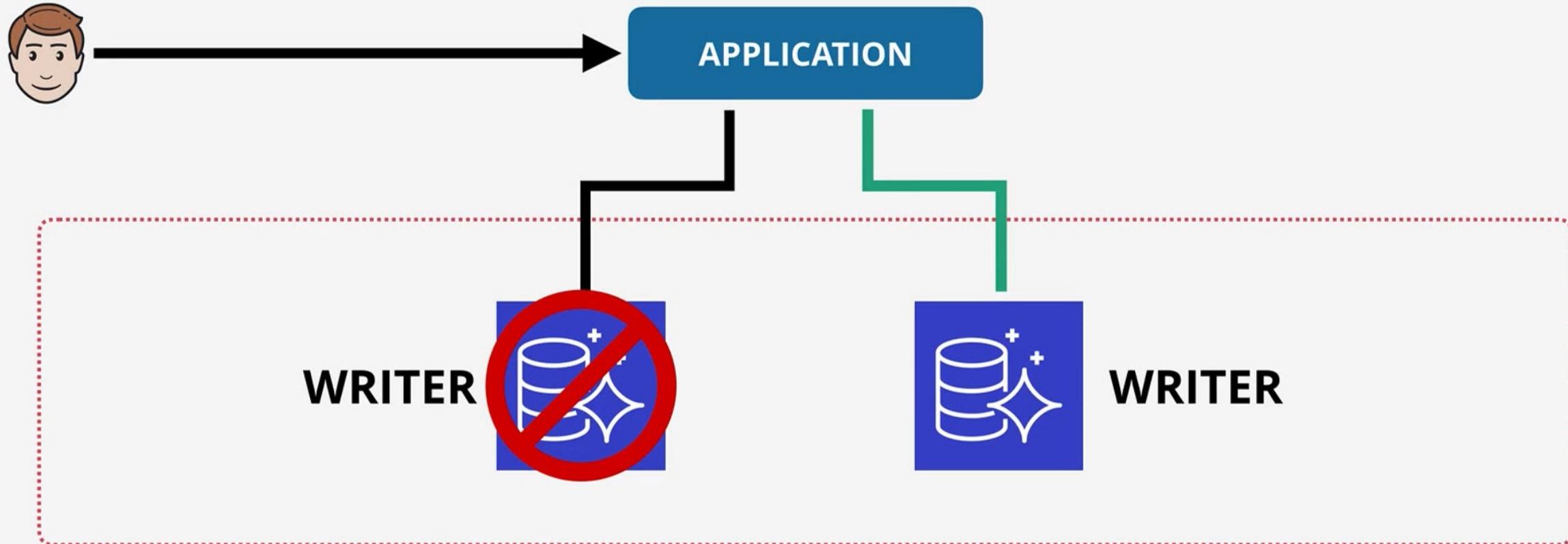
Aurora Multi-Master



<https://learn.cantrill.io>



adriancantrill





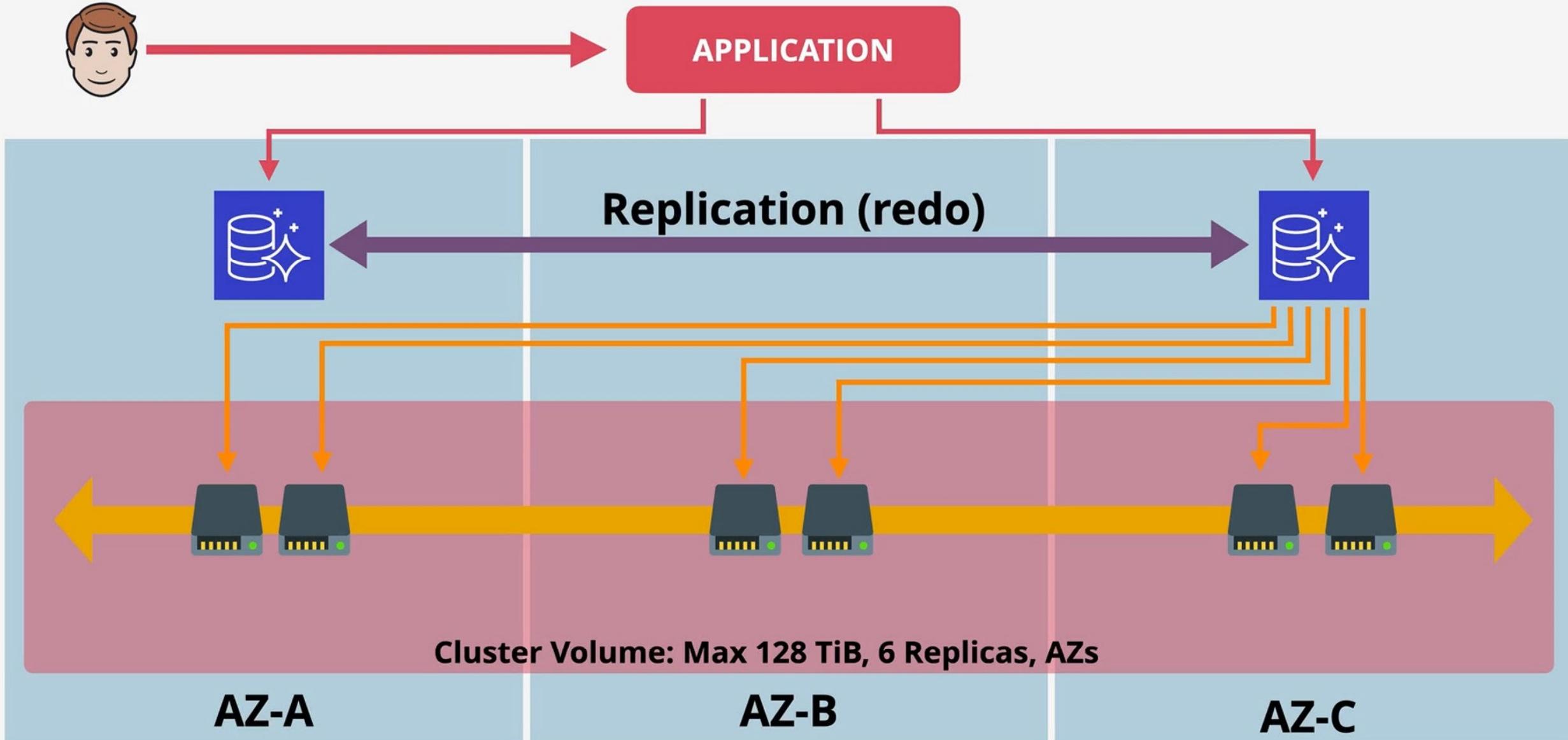
Aurora Multi-Master



<https://learn.cantrill.io>



adriancantrill





Aurora Multi-Master



<https://learn.cantrill.io>



adriancantrill

- Default Aurora mode is **Single-Master**
- **One R/W** and **0+ Read Only** Replicas
- Cluster Endpoint is used to write, read endpoint is used for load balanced reads
- Failover takes time - replica promoted to R/W
- In Multi-Master mode **all instances are R/W**



Aurora Global Database



<https://learn.cantrill.io>



adriancantrill

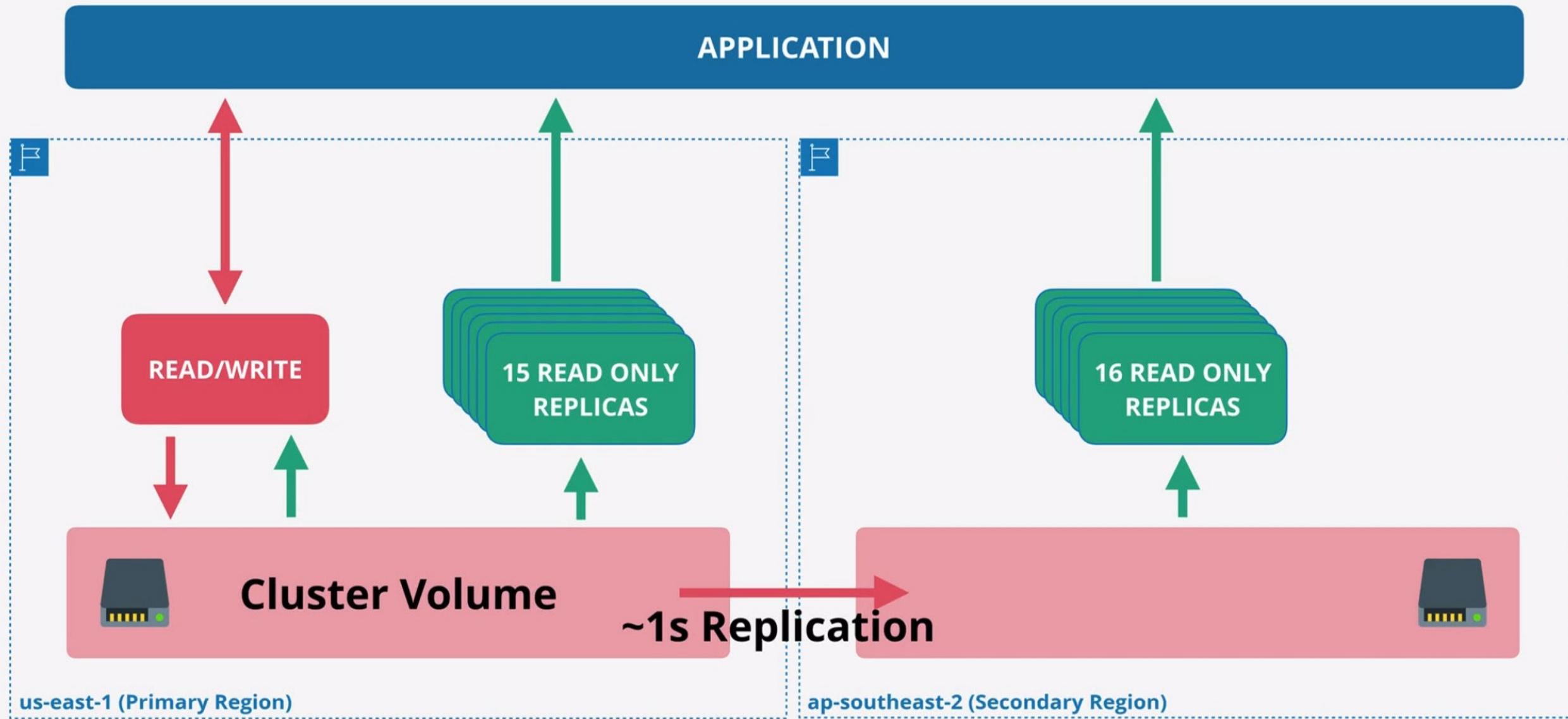
- **Cross-Region DR and BC**
- **Global Read Scaling - low latency performance improvements**
- **~1s or less** replication between regions
- **No impact** on DB performance
- Secondary regions can have **16 replicas**
- .. Can be promoted to R/W
- Currently MAX 5 secondary regions...



Aurora Global Database

<https://learn.cantrill.io>

adriancantrill





- **Infrequently** used applications
- **New** applications
- **Variable** workloads
- **Unpredictable** workloads
- **Development** and **test** databases
- **Multi-tenant** applications



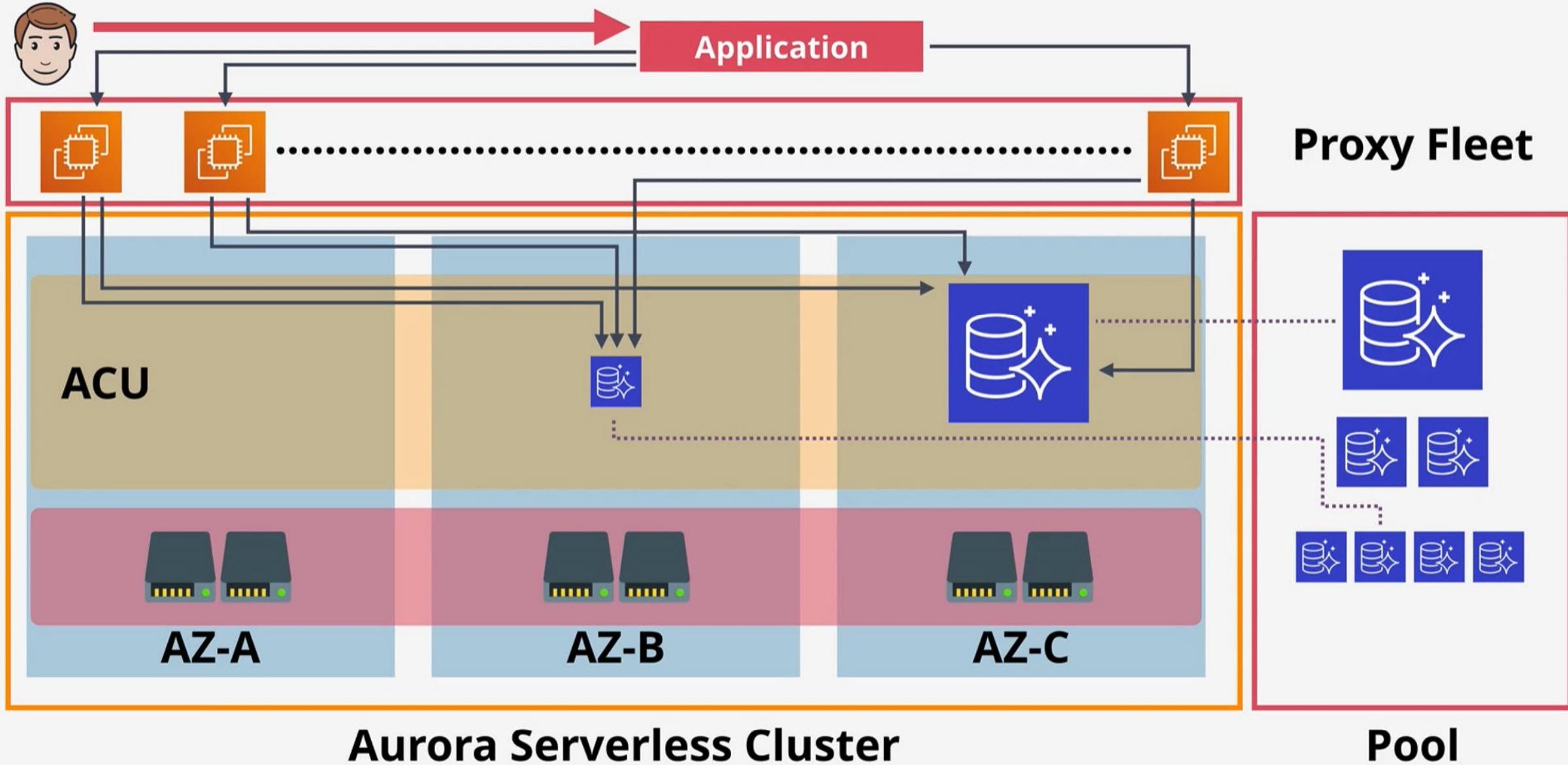
Aurora Serverless Architecture



<https://learn.cantrill.io>



adriancantrill





Aurora Serverless Concepts



<https://learn.cantrill.io>



adriancantrill

- Scalable - **ACU** - Aurora Capacity Units
- Aurora Serverless cluster has a **MIN & MAX ACU**
- Cluster adjusts based on load
- Can go to **0** and be **paused**
- Consumption billing per-second basis
- Same resilience as Aurora (6 copies across AZs)



Aurora Restore, Clone & Backtrack



<https://learn.cantrill.io>



adriancantrill

- Backups in Aurora work in the same way as RDS
- Restores create a **new cluster**
- Backtrack can be used which allow **in-place rewinds** to a previous point in time
- Fast clones make a new database MUCH faster than copying all the data - **copy-on-write**



Cost



<https://learn.cantrill.io>



adriancantrill

- **No free-tier option**
- Aurora doesn't support Micro Instances
- Beyond RDS singleAZ (micro) Aurora offers better value
- Compute - hourly charge, per second, 10 minute minimum
- Storage - GB-Month consumed, IO cost per request
- 100% DB Size in backups are included



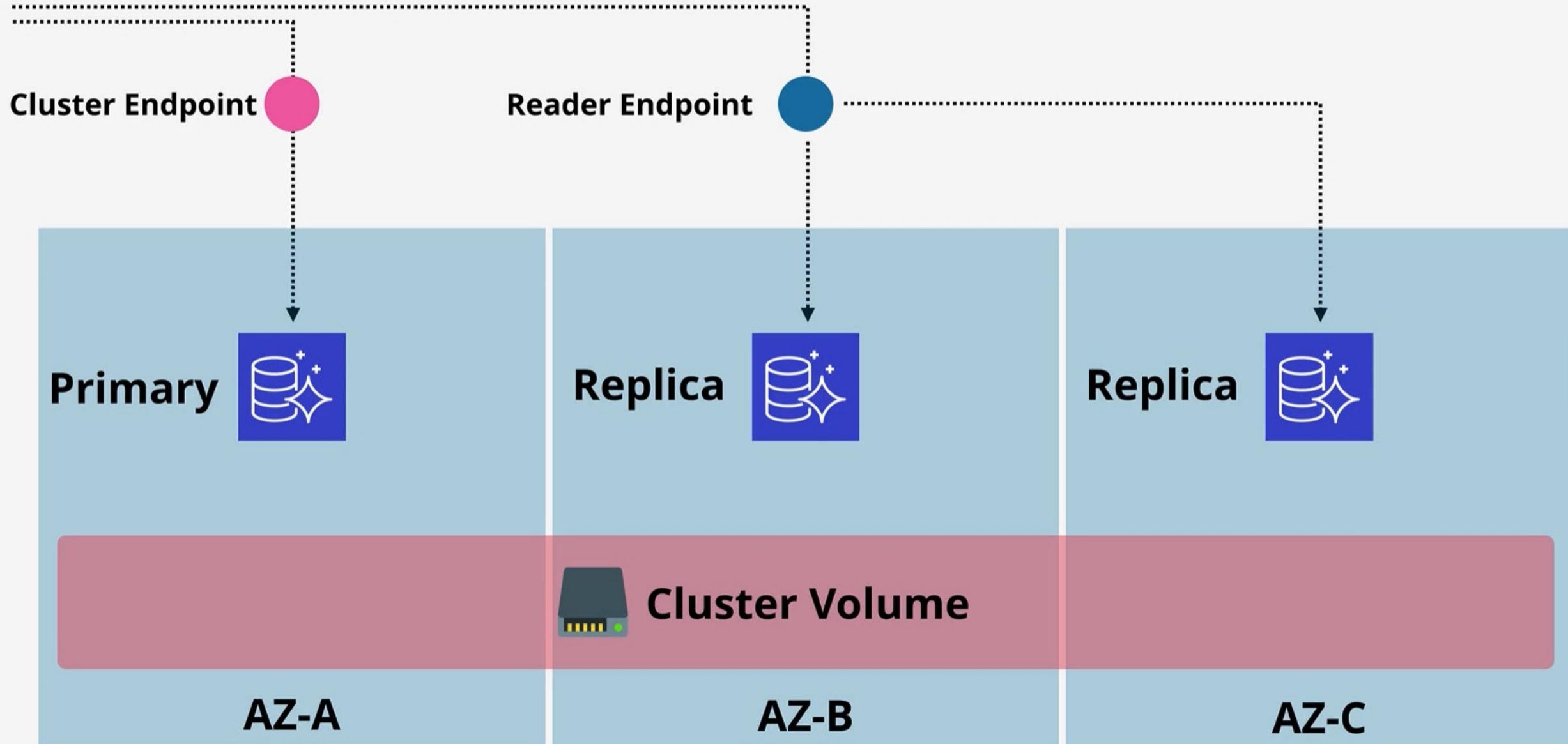
Endpoints



<https://learn.cantrill.io>



adriancantrill





Aurora Storage Architecture



<https://learn.cantrill.io>



adriancantrill

- All SSD Based - **high IOPS, low latency**
- Storage is billed based on **what's used**
- **High water mark** - billed for the most used
- Storage which is freed up can be re-used
- Replicas can be added and removed without requiring storage provisioning.



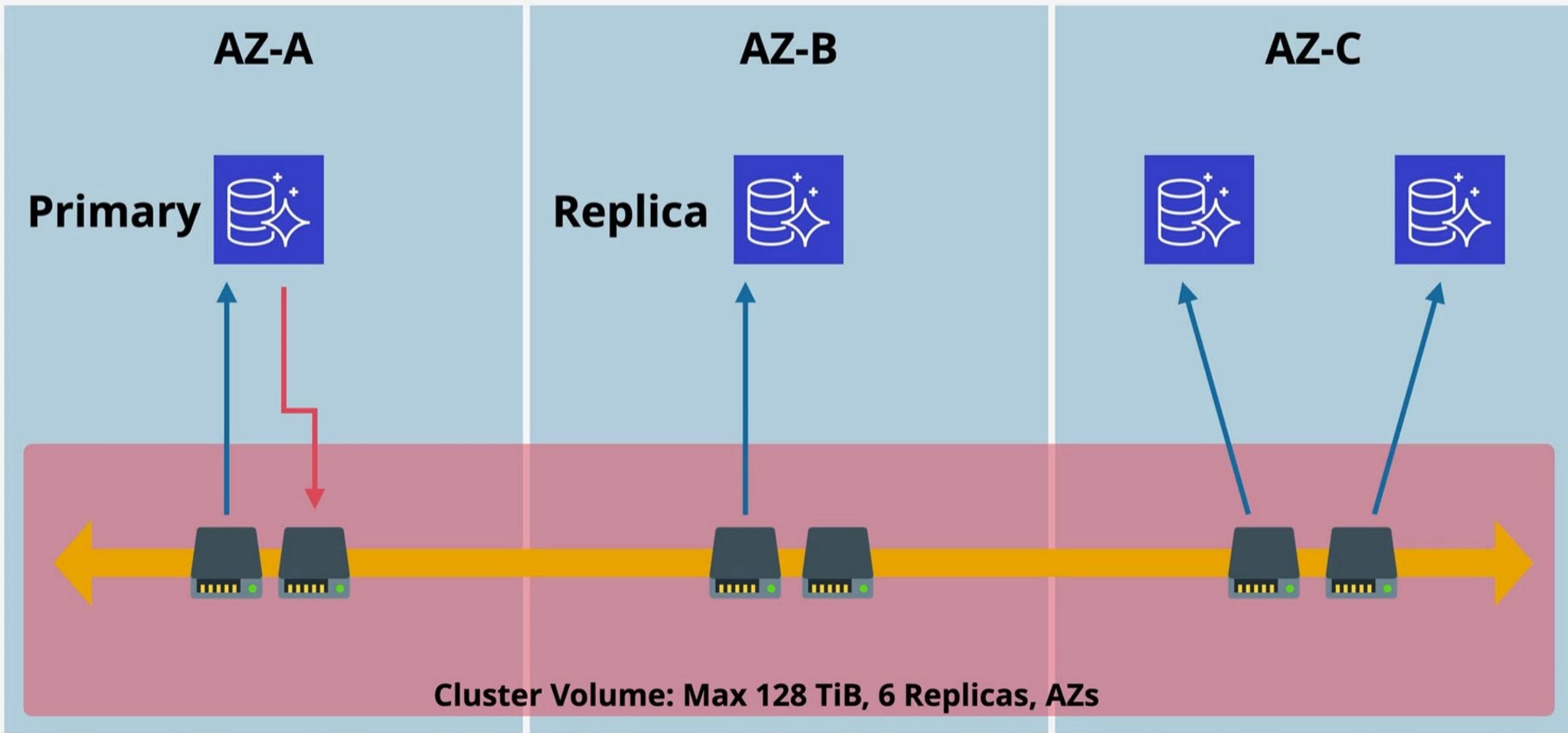
Aurora Storage Architecture



<https://learn.cantrill.io>



adriancantrill





Aurora Key Differences



<https://learn.cantrill.io>



adriancantrill

- Aurora architecture is **VERY** different from RDS...
- ...Uses a “**Cluster**”
- A single **primary** instance + **0** or more **replicas**
- No local storage - uses **cluster volume**
- Faster provisioning & improved availability & performance

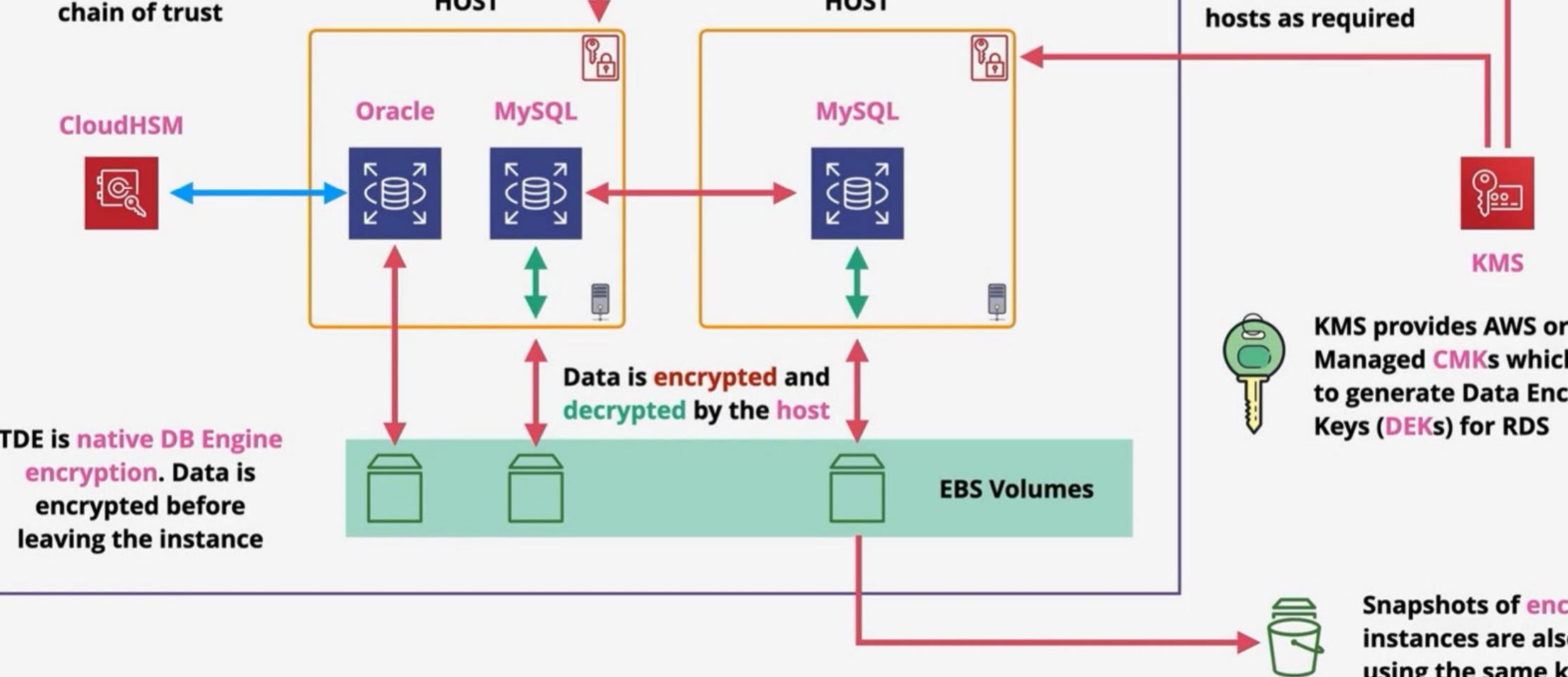


Amazon RDS KMS Encryption & TDE

<https://learn.cantrill.io>

adriancantrill

With RDS Oracle - keys can be provided via CloudHSM
- removing AWS from the chain of trust





Amazon RDS Security



<https://learn.cantrill.io>



adriancantrill

- RDS **MSSQL** and RDS **Oracle** Support **TDE** 
- .. **Transparent Data Encryption**
- Encryption handled **within the DB engine**
- RDS **Oracle** supports integration with **CloudHSM** 
- Much stronger key controls (even from AWS)



Amazon RDS Security

<https://learn.cantrill.io>

adriancantrill

- **SSL/TLS (in transit)** is available for RDS, can be **mandatory**
- RDS supports **EBS volume** encryption - **KMS**
- Handled by **HOST/EBS**
- AWS or Customer Managed CMK generates **data keys**.
- **Data keys** used for **encryption operations**
- **Storage, Logs, Snapshots & replicas** are encrypted
-**encryption can't be removed**





RPO/RTO Improvements



<https://learn.cantrill.io>



adriancantrill

- Snapshots & Backups Improve **RPO**
- **RTO's are a problem**
- RR's offer **nr. 0 RPO**
- RR's can be **promoted quickly - low RTO**
- **Failure only** - watch for data corruption
- **Read only - until promoted**
- **Global availability improvements ... global resilience**



(read) Performance Improvements



<https://learn.cantrill.io>



adriancantrill

- **5x** direct read-replicas per DB instance
- Each providing an **additional instance of read performance**
- Read-Replicas can have read-replicas - **but lag starts to be a problem.**
- **Global** performance improvements



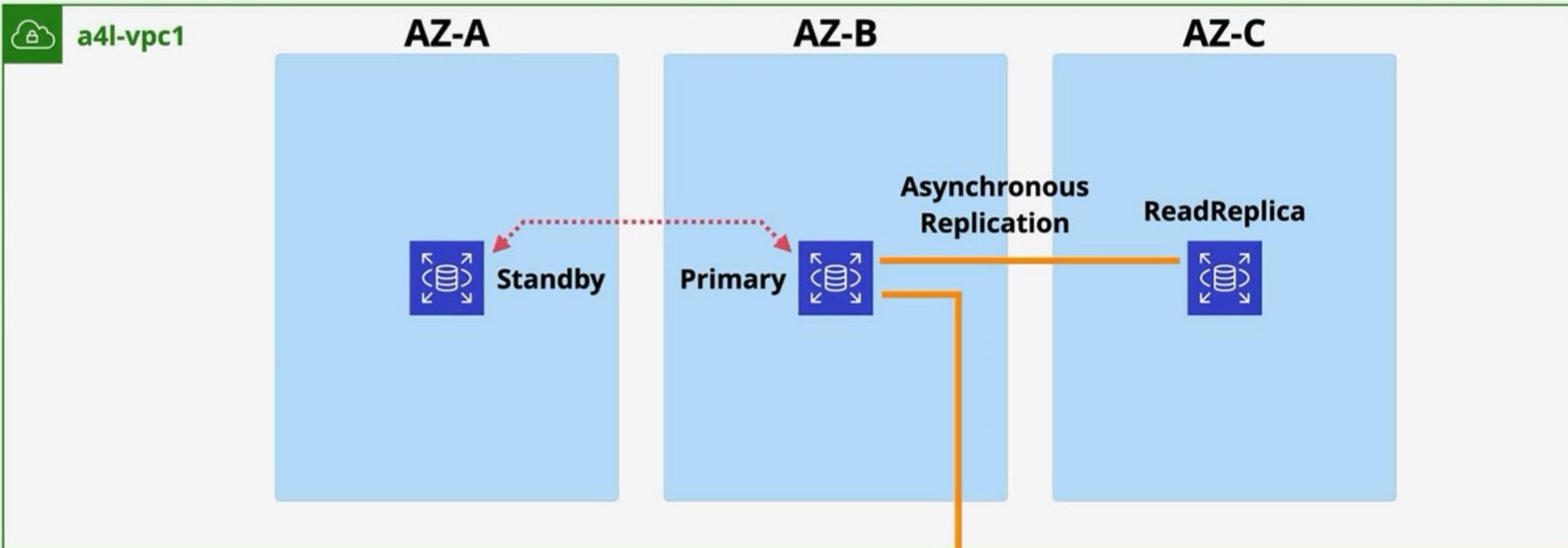
RDS - Read Replicas

<https://learn.cantrill.io>

adriancantrill

us-east-1

**Read Only
DB Replicas**



ap-southeast-2





- Creates a **NEW RDS Instance - new address**
- Snapshots = **single point in time**, creation time
- Automated = **any 5 minute point in time**
- Backup is restored and transaction logs are 'replayed' to bring DB to desired point in time (**GOOD RPO**)
- Restores **aren't fast** - Think about **RTO (*RR's)**



RDS - Backups - Cross-Region

 <https://learn.cantrill.io>

 adriancantrill 

- RDS can **replicate** backups to **another region**
- .. both **snapshots** and **transaction logs**
- Charges apply for the **cross-region data copy**
- ... and the **storage** in the **destination region**
- **NOT DEFAULT** ..configured  within automated backups



RDS - Backups - General



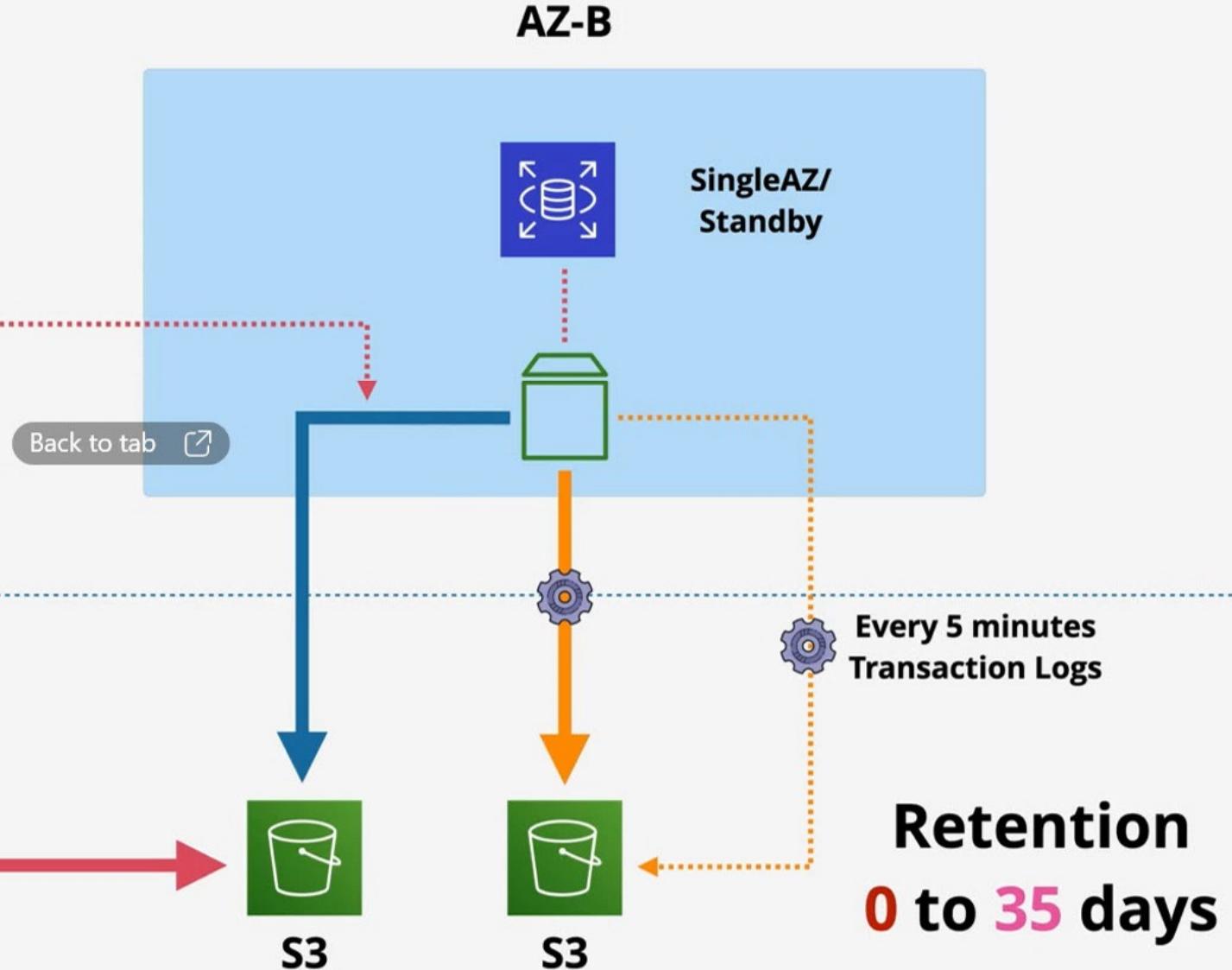
<https://learn.cantrill.io>



adriancantrill

us-east-1

First Snap is FULL
Size of consumed data
Then onward = Incremental





RDS - Backups - General

<https://learn.cantrill.io>

adriancantrill

F us-east-1



Automated Backups

Snapshots

AWS Managed S3 Buckets





Why might you do it...



<https://learn.cantrill.io>



adriancantrill

- Access to the DB Instance **OS**
- **Advanced DB Option tuning** ... (DBROOT)
- ... Vendor demands.. 
- **DB or DB Version AWS don't provide..**
- Specific **OS/DB Combination** AWS don't provide
- Architecture AWS don't provide (replication/resilience)
- Decision makers who '**just want it**'