

# Operations with DynamoDB

---



**Ivan Mushketyk**

@mushketyk   brewing.codes



# Overview



**Monitoring DynamoDB**

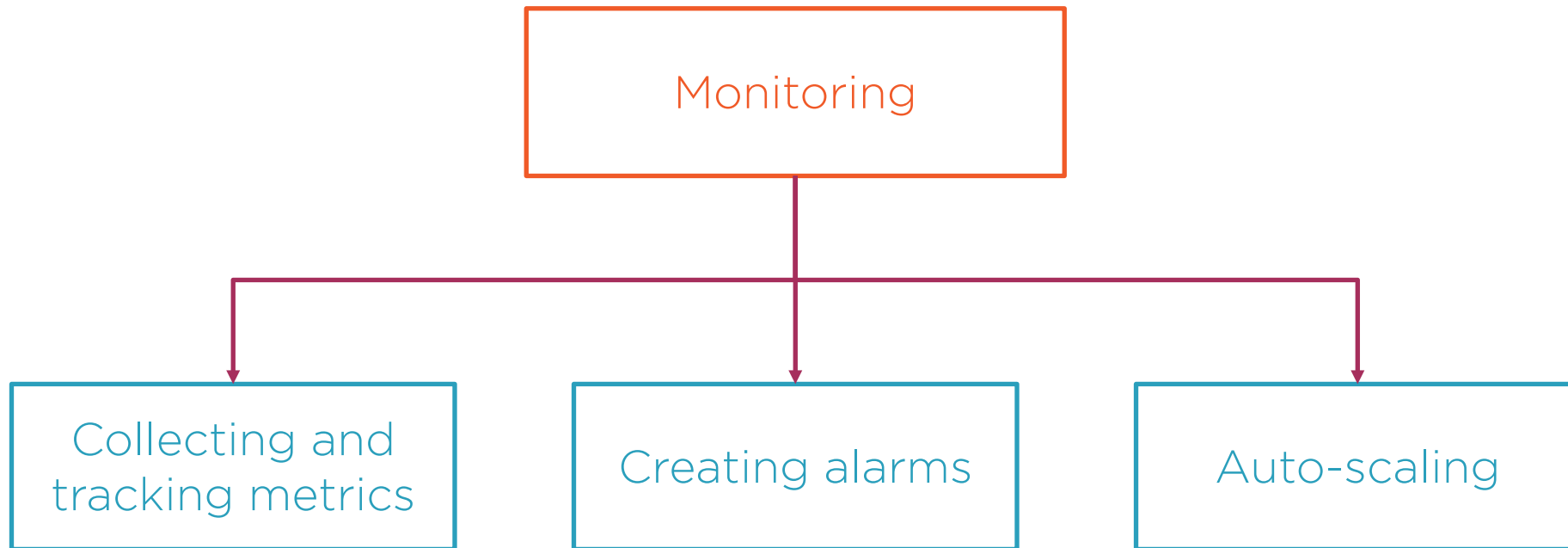
**Caching with DAX**

**Back ups with DynamoDB**

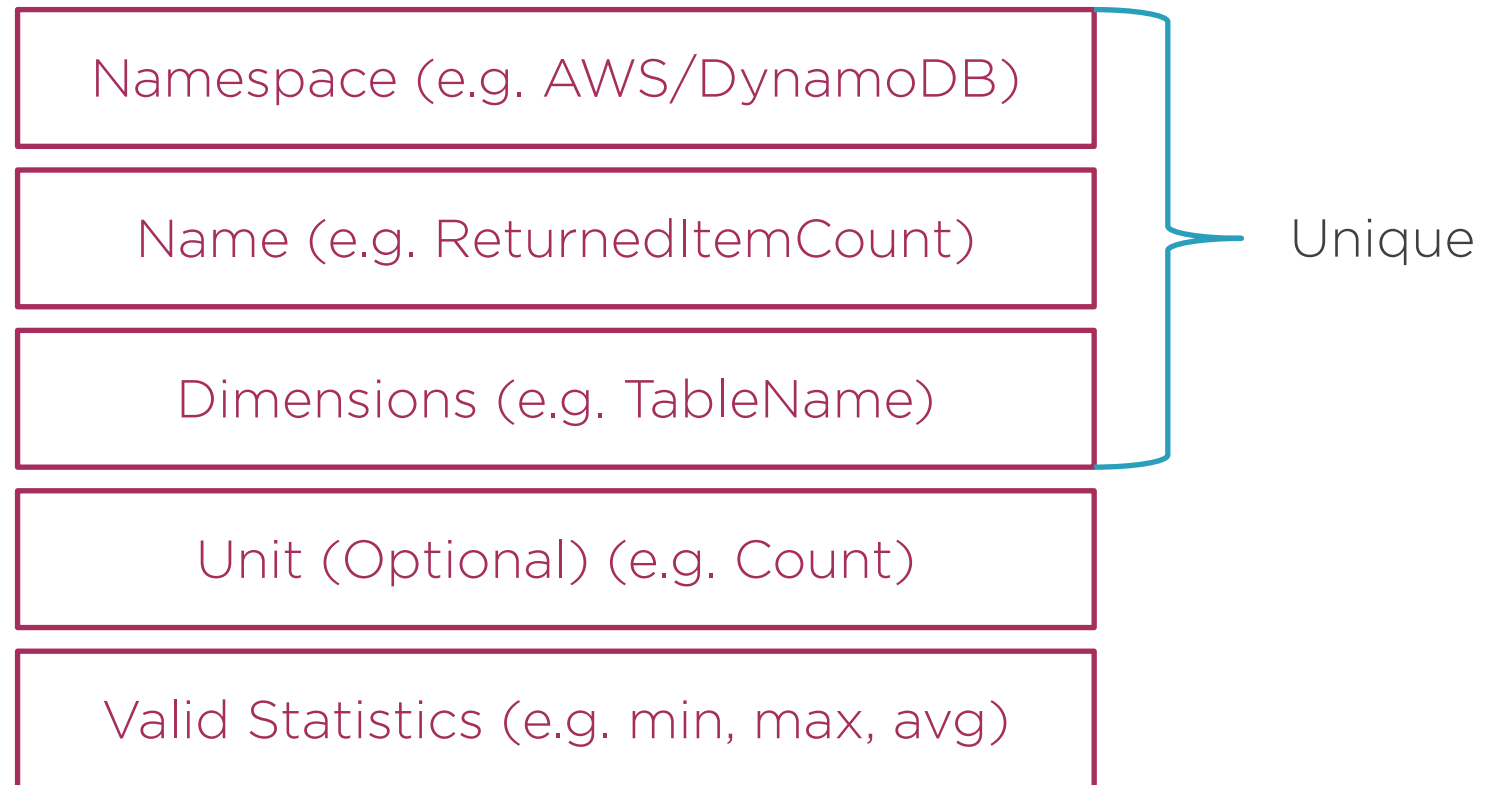
**Cross-region replication**



# AWS CloudWatch



# CloudWatch Metric



# DynamoDB Metrics

	Metrics
Throughput	Provisioned/Consumed Read/Write CapacityUnits, Read/Write ThrottleEvents, ThrottledRequests



# DynamoDB Metrics

	Metrics
Throughput	Provisioned/Consumed Read/Write CapacityUnits, Read/Write ThrottleEvents, ThrottledRequests
Data Returned	ReturnedRecordsCount, ReturnedItemCount, ReturnedBytes



# DynamoDB Metrics

	Metrics
Throughput	Provisioned/Consumed Read/Write CapacityUnits, Read/Write ThrottleEvents, ThrottledRequests
Data Returned	ReturnedRecordsCount, ReturnedItemCount, ReturnedBytes
Failures	ConditionalCheckFailedRequests, UserErrors, SystemErrors



# DynamoDB Metrics

	Metrics
Throughput	Provisioned/Consumed Read/Write CapacityUnits, Read/Write ThrottleEvents, ThrottledRequests
Data Returned	ReturnedRecordsCount, ReturnedItemCount, ReturnedBytes
Failures	ConditionalCheckFailedRequests, UserErrors, SystemErrors
Other	SuccessfulRequestLatency, TimeToLiveDeletedItemCount





# What to Do with Metrics?



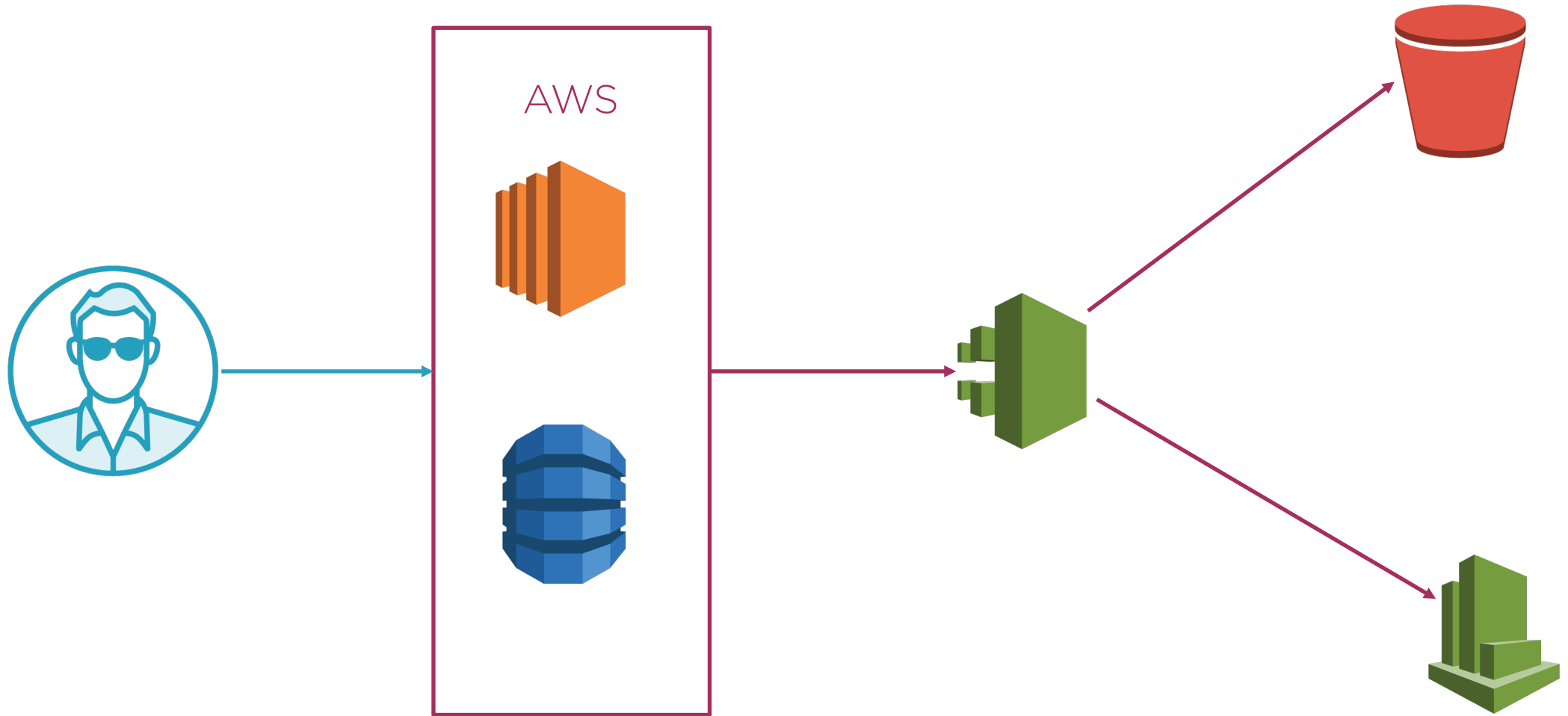
**Store custom metrics (e.g. request latency)**

**Review service/custom metrics**

**Create dashboards**

**Create alarms**

# AWS CloudTrail



# Information Logged by CloudTrail



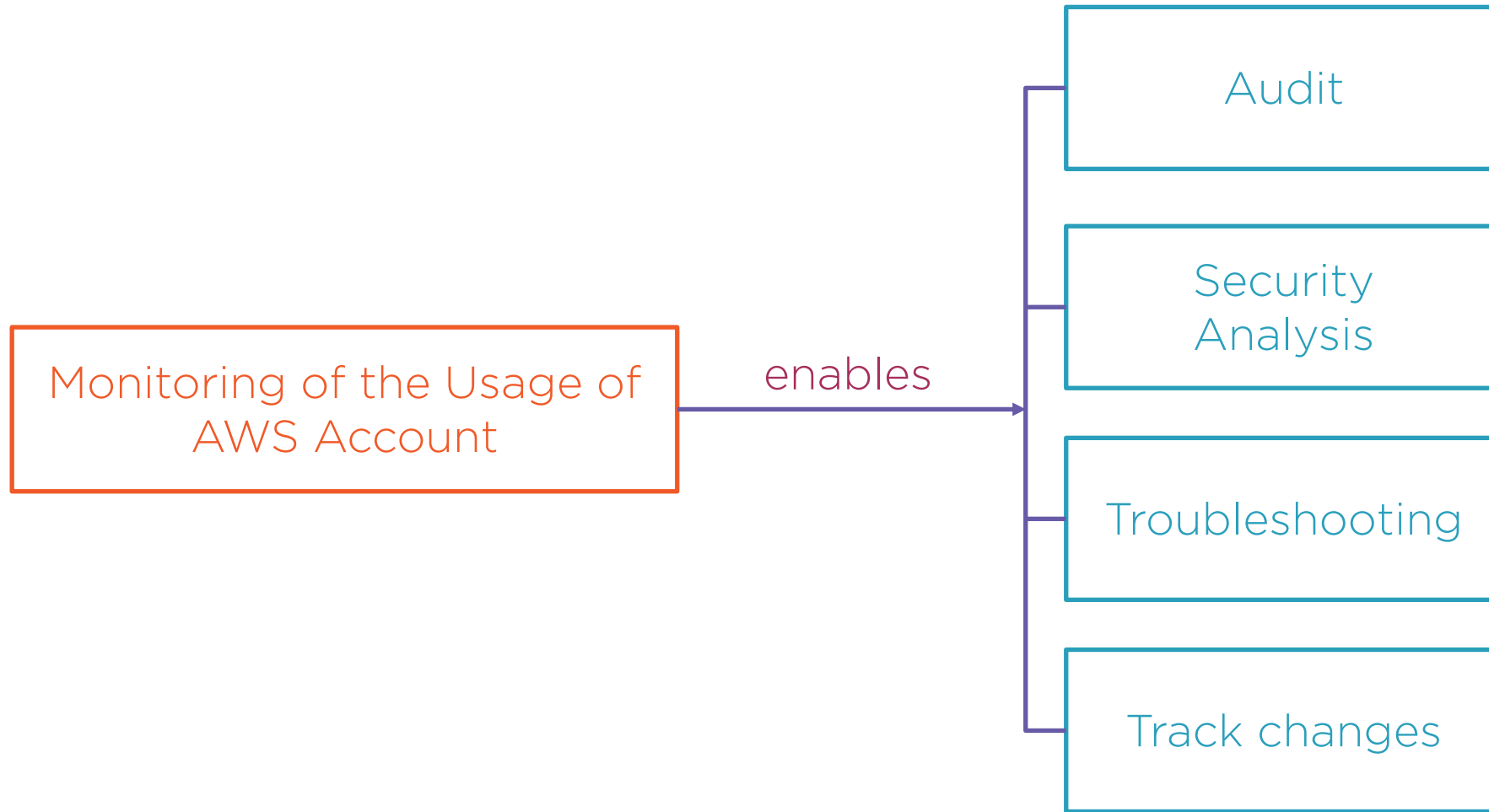
**User who made API call**

**API method**

**Timestamp**

**Call parameters**

# AWS CloudTrail



# What to Do with CloudTrail Events?



**Search for events in CloudTrails (up to 7 days)**

**Store log events to S3 or CloudWatch logs**

**Configure SNS notifications**

**Use third-party solutions**

# Why Do We Need Caching?



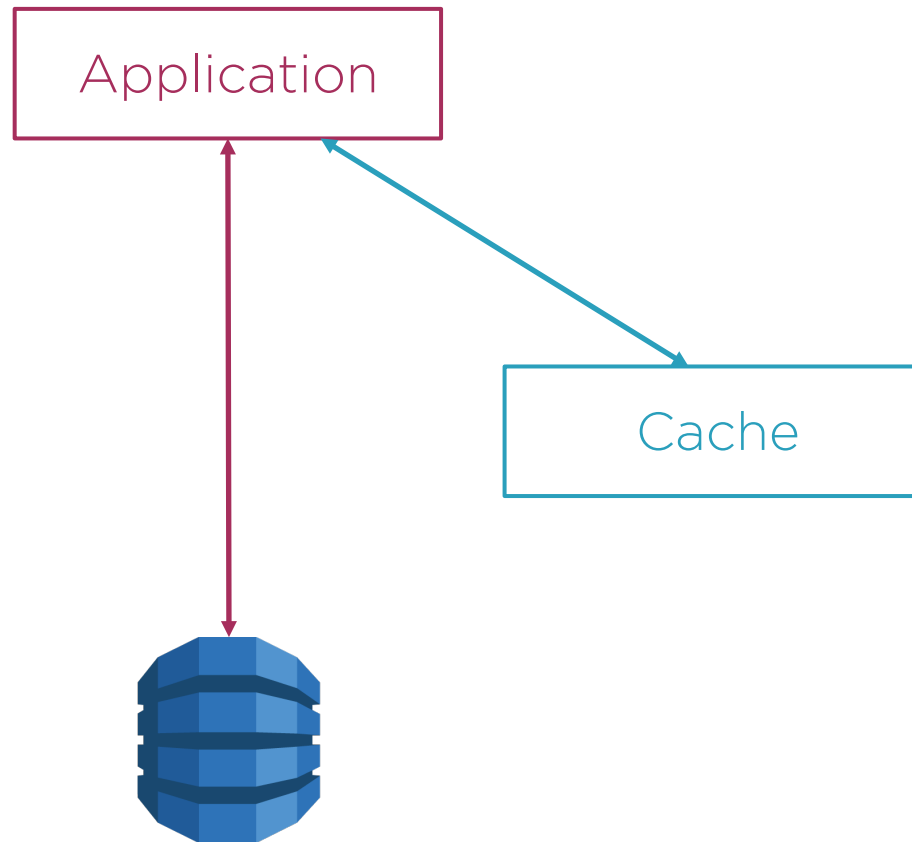
**Avoid hot partitions**

**Decrease latency**

**Read-intensive applications**

**Save money on provisioned capacity**

# Cache Before DAX



# What Is DAX?



**In-memory Acceleration**

**Submillisecond latency**

**Fully-managed/flexible**

**Scalable, up to 10 nodes**

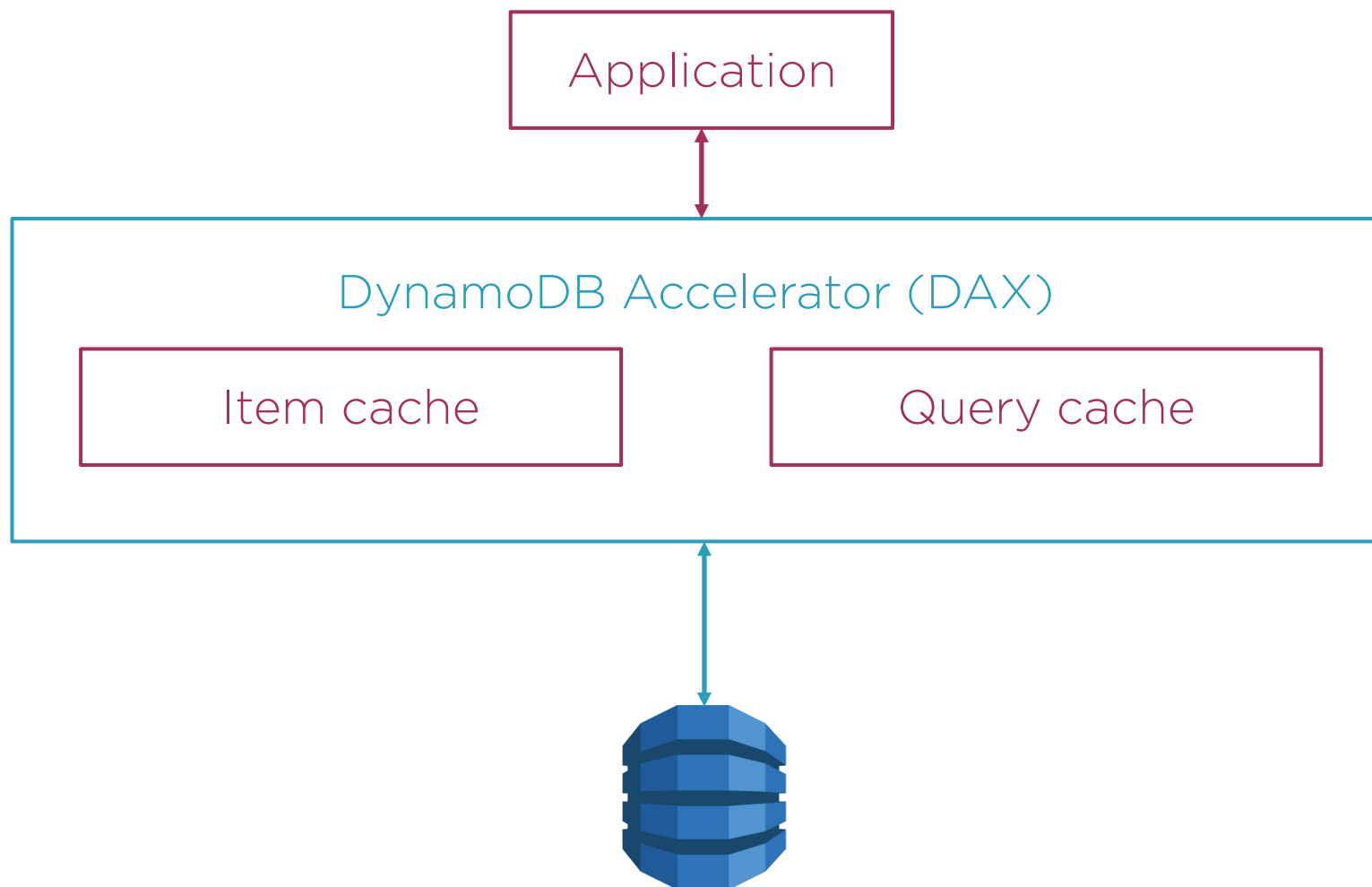
**”Sits” between DynamoDB and application**

**Easy to use. Same API**

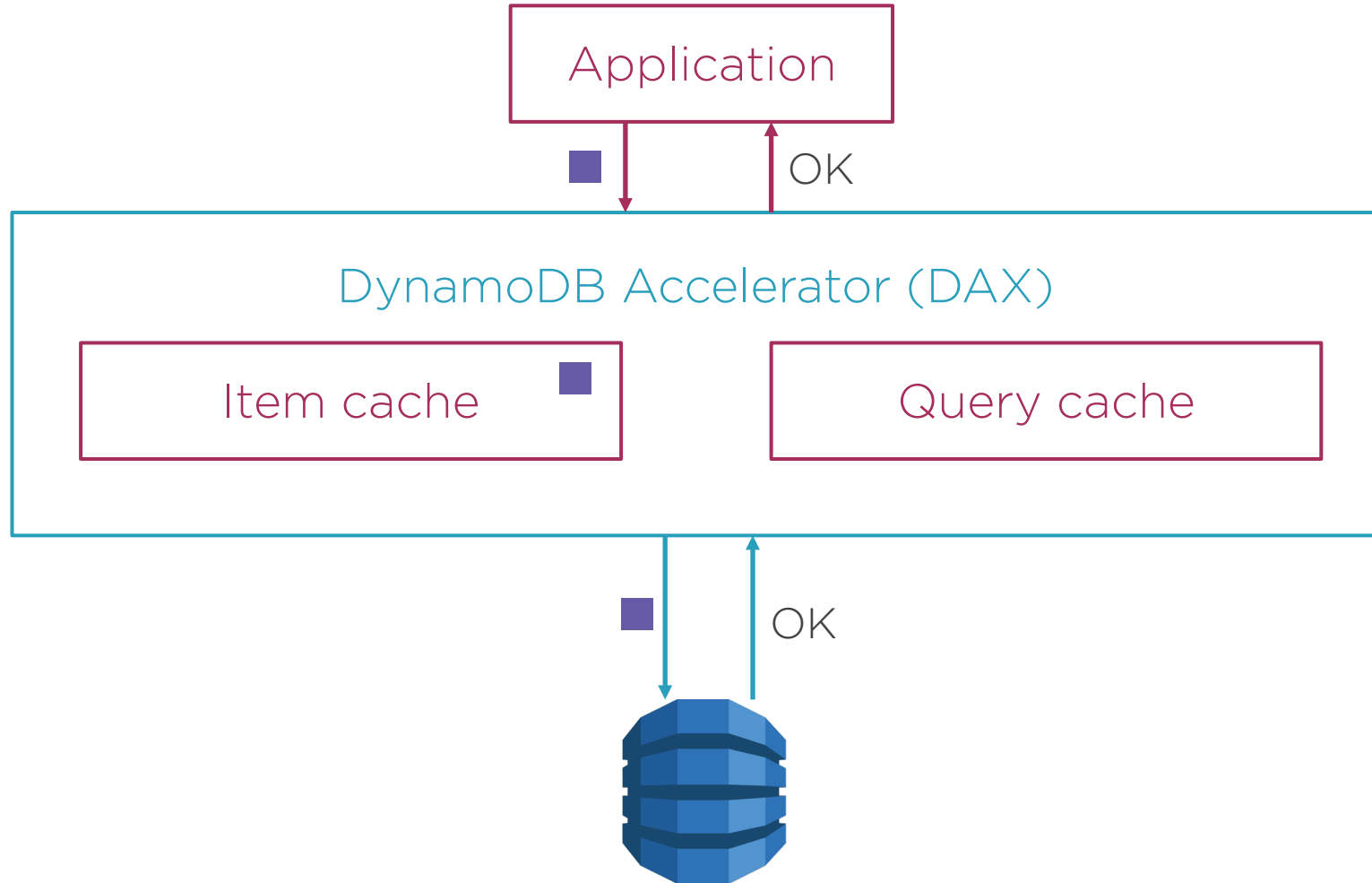




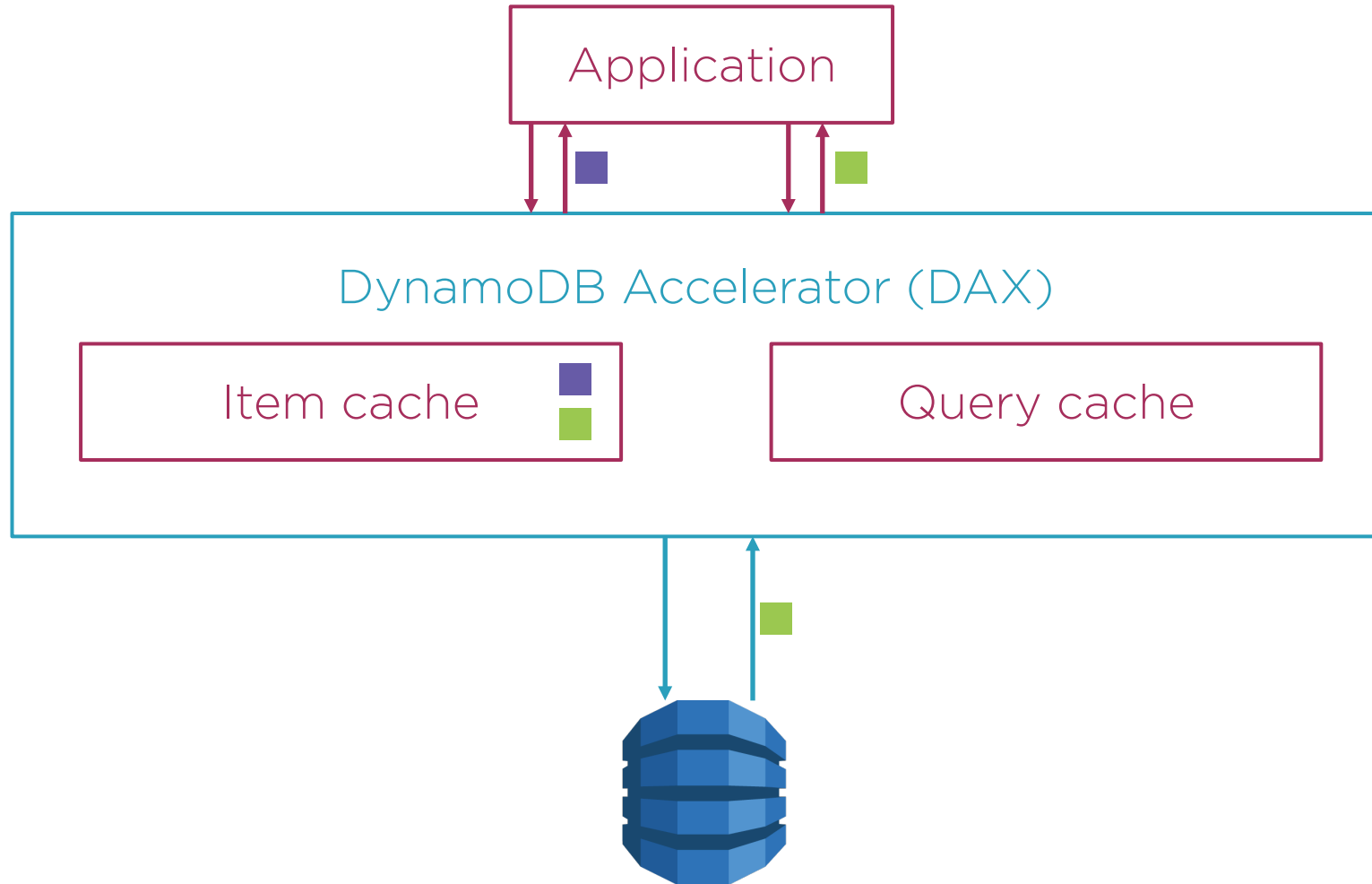
# How DAX Works



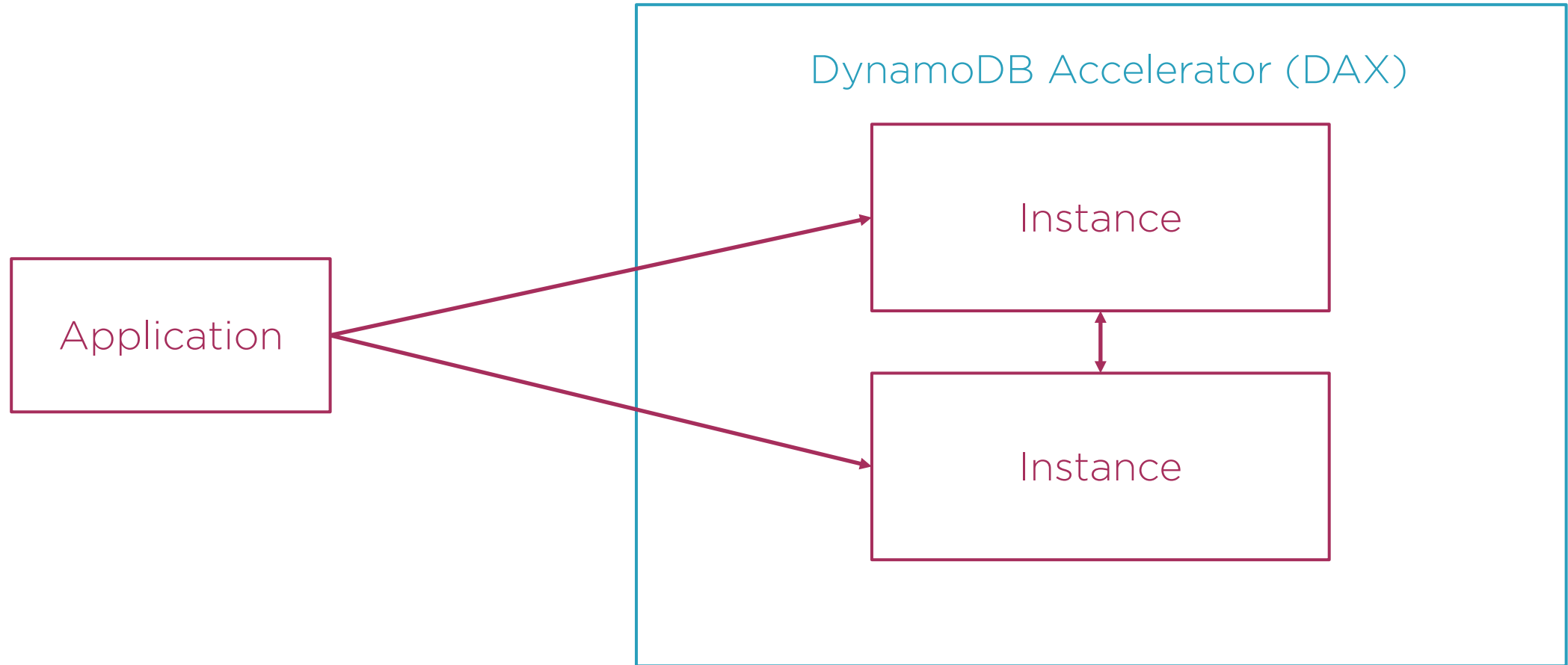
# Writing to DAX



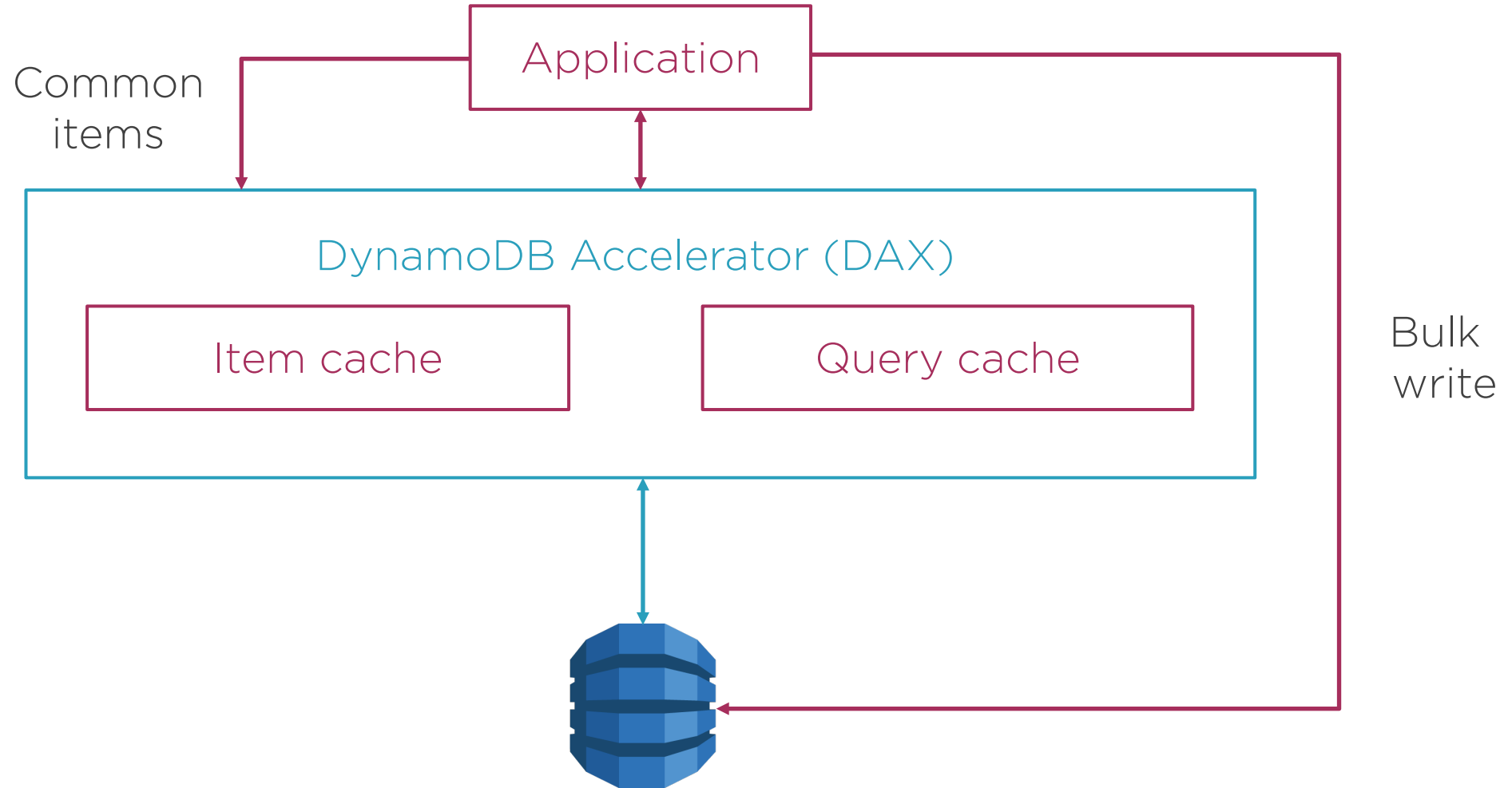
# Reading from DAX



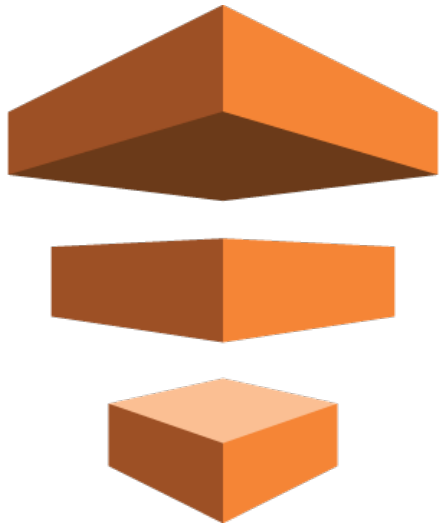
# Data Consistency in DAX



# Write-Around



# AWS Data Pipeline



**System to define ETL jobs**

**Declarative**

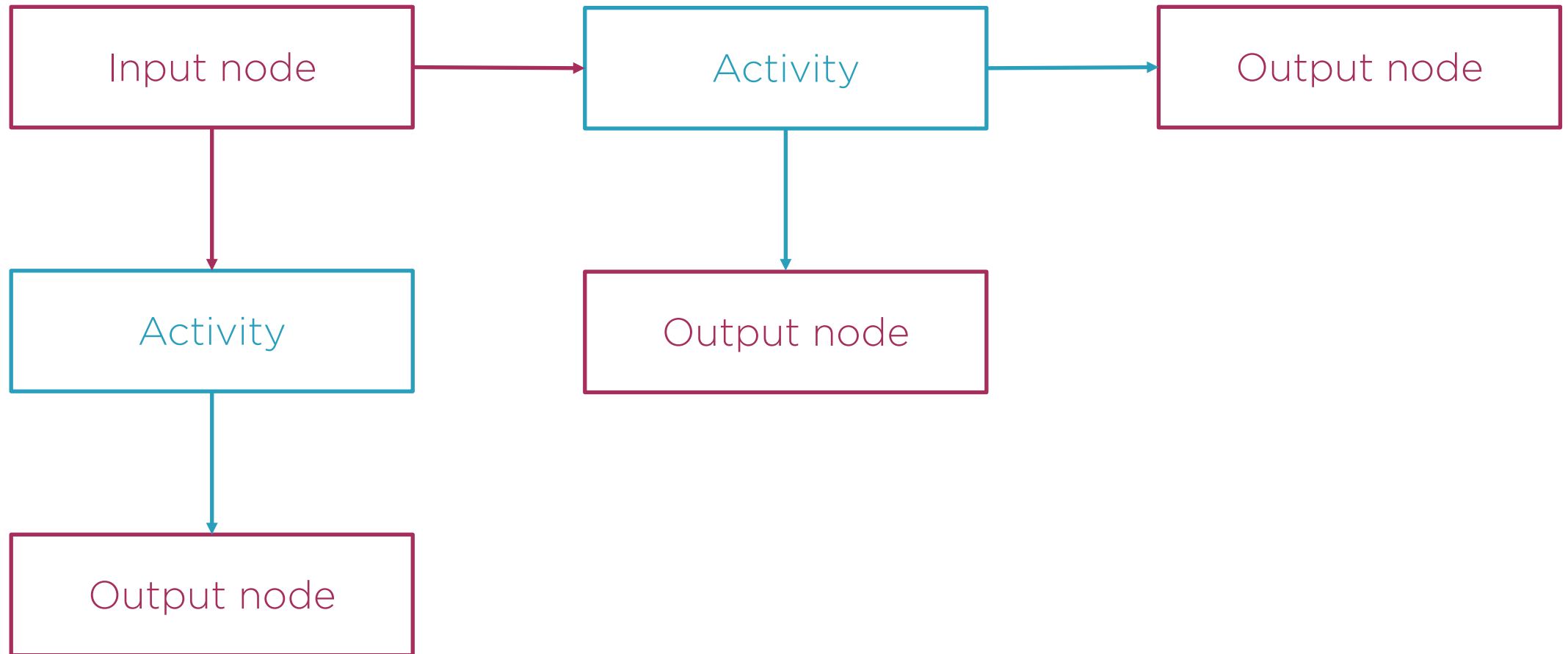
**Resilient**

**Allows to save money**

**Integrated with AWS services**



# Data Pipeline in a Nutshell



# Inputs/Outputs for Data Pipeline



DynamoDB

S3

RDS/Redshift

JDBC queries/custom scripts



# Predefined Templates



**DynamoDB import/export**

**RDS copy/restore**

**Copy RDS to Redshift**

**Load S3 to Redshift**

# Cross-Region Replication

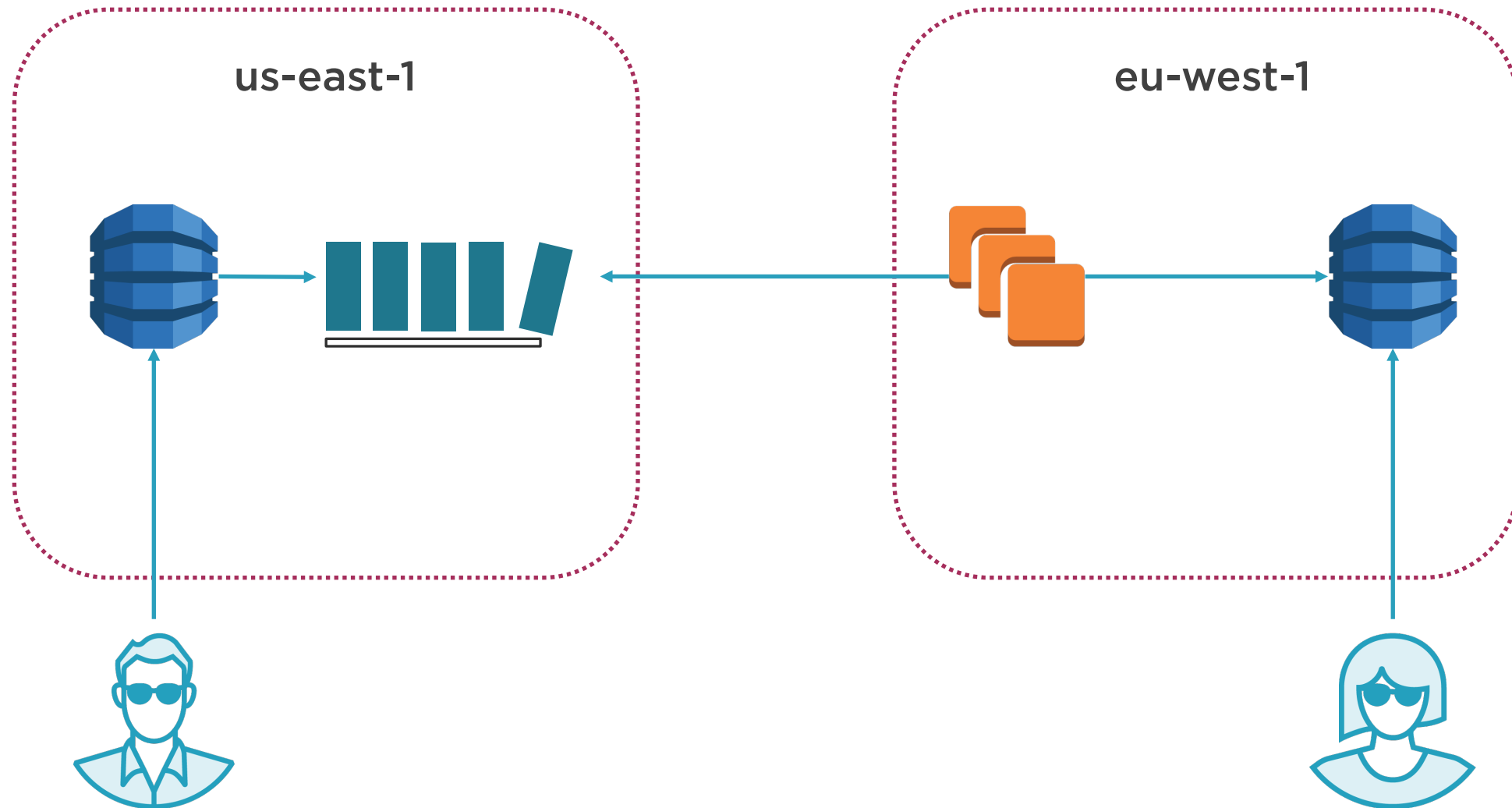


**Low-latency access for global applications**

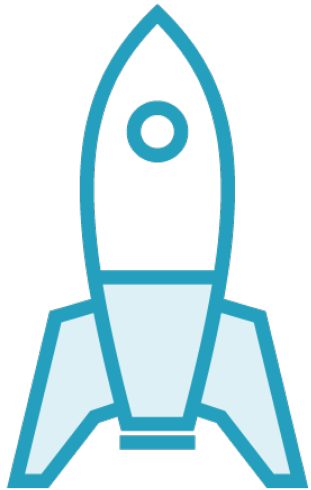
**Maintain copy of your data**



# Cross-Region Replication



# How to Enable Replication



**Install replication tool**

**Enable stream for original table**

**Create initial copy**

**Run tool continuously**



# Summary



**Covered DynamoDB operations**

**Monitor DynamoDB tables**

**Use DAX to implement cache**

**Perform back-ups**

**Enable cross-region replication**

