

# INNERSOURCE- JPMC

#### ACCOUNT TECHNOLOGY OFFICE

# Migrate Apache Cassandra workloads to Amazon Keyspaces by using AWS Glue (User story)

## Created By

Emp ID - Emp Name

Emp ID - Emp Name

Emp ID - Emp Name

Emp ID - Emp Name

### JUNE 2024

# INDEX

## Contents

|  |  |
| --- | --- |
| 1. Specifications | 3 |
| 1. Summary | 3 |
| 1. Prerequisites | 3 |
| 1. Limitations | 3 |
| 1. Architecture | 4 |
| 1. Technology Stack | 4 |
| 1. Target Architecture | 4 |
| 1. Tools | 4 |
| 1. AWS Services and Tools | 4 |
| 1. Code | 5 |
| 1. Best Practices | 5 |
| 1. Approach | 5 |
| 1. Troubleshooting | 6 |
| 1. Resources | 6 |

# Migrate Apache Cassandra workloads to Amazon Keyspaces by using AWS Glue (User story)

## Specifications:

|  |  |  |
| --- | --- | --- |
| **Environment:**Dev | **Source:**Cassandra | **Target:**Amazon Keyspaces |
| **R Type:**N/A | **Workload:**Application | **Technologies:**Analytics; Migration; Serverless; Big data |
| **AWS services:**AWS Glue; Amazon Keyspaces; Amazon S3; AWS CloudShell |  |  |

## Summary

This pattern shows you how to migrate your existing Apache Cassandra workloads to Amazon Keyspaces (for Apache Cassandra) by using CQLReplicator on AWS Glue. You can use CQLReplicator on AWS Glue to minimize the replication lag of migrating your workloads down to a matter of minutes. You also learn how to use an Amazon Simple Storage Service (Amazon S3) bucket to store data required for the migration, including [Apache Parquet](https://parquet.apache.org/) files, configuration files, and scripts. This pattern assumes that your Cassandra workloads are hosted on Amazon Elastic Compute Cloud (Amazon EC2) instances in a virtual private cloud (VPC).

### 3. Prerequisites

* Cassandra cluster with a source table.
* Target table in Amazon Keyspaces to replicate the workload.
* S3 bucket to store intermediate Parquet files that contain incremental data changes.
* S3 bucket to store job configuration files and scripts.

### Limitations

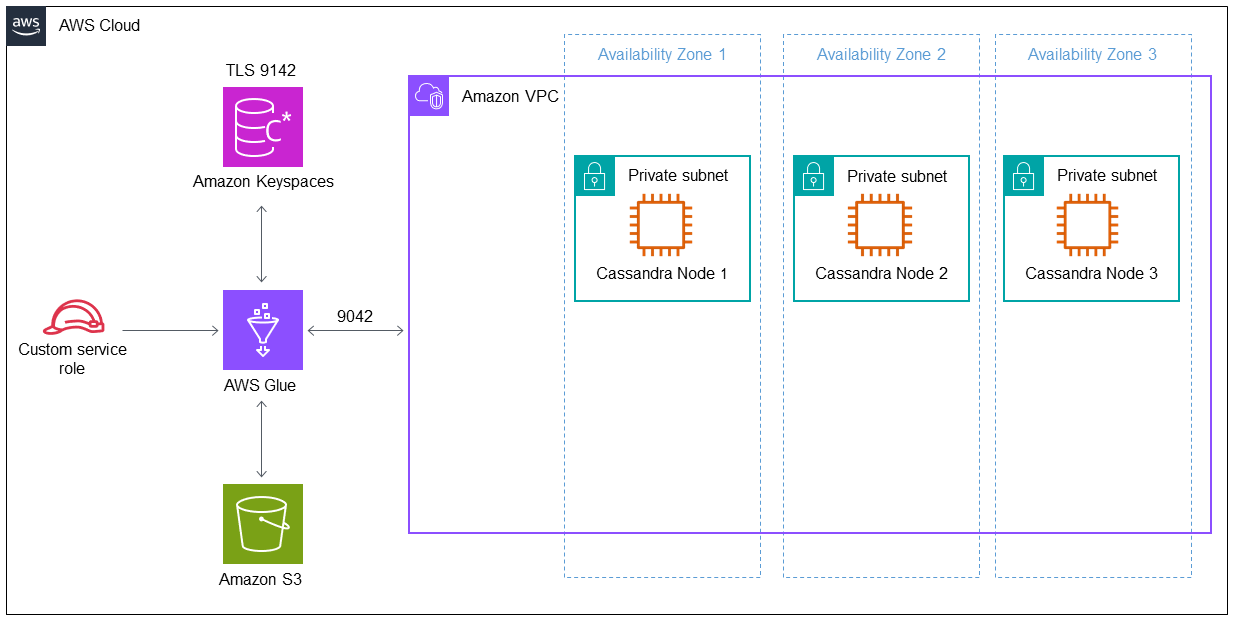
* CQLReplicator on AWS Glue requires some time to provision Data Processing Units (DPUs) for the Cassandra workloads. The replication lag between the Cassandra cluster and the target keyspace and table in Amazon Keyspaces is likely to last for only a matter of minutes.

## Architecture

### 5.1 Technology Stack:

* Apache Cassandra
* DataStax Server
* ScyllaDB
* Amazon Keyspaces

### 5.2 Target Architecture: \*\*\*If required\*\*\*



## Tools: (\*\*Tools used to implement User Story\*\*)

### 6.1 AWS services and tools

* [AWS Command Line Interface (AWS CLI)](https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-welcome.html) is an open-source tool that helps you interact with AWS services through commands in your command-line shell.
* [AWS CloudShell](https://docs.aws.amazon.com/cloudshell/latest/userguide/welcome.html) is a browser-based shell that you can use to manage AWS services by using the AWS Command Line Interface (AWS CLI) and a range of preinstalled development tools.

### 6.2 Code

The code for this pattern is available in the GitHub [CQLReplicator](https://github.com/aws-samples/cql-replicator/tree/main/glue) repository.

## Best practices

* To determine the necessary AWS Glue resources for the migration, estimate the number of rows in the source Cassandra table. For example, 250 K rows per 0.25 DPU (2 vCPUs, 4 GB of memory) with 84 GB disk.
* Pre-warm Amazon Keyspaces tables before running CQLReplicator. For example, eight CQLReplicator tiles (AWS Glue jobs) can write up to 22 K WCUs per second, so the target should be pre-warmed up to 25-30 K WCUs per second.

## Approach

|  |  |  |
| --- | --- | --- |
| **Task** | **Description (Procedure & Code Snippets)** | **Skills Required** |
| Create an IAM role for the AWS Glue job. | Create a new AWS service role named glue-cassandra-migration with AWS Glue as a trusted entity.  **Note:** The glue-cassandra-migration should provide read and write access to the S3 bucket and Amazon Keyspaces. The S3 bucket contains the .jar files, configuration files for Amazon Keyspaces and Cassandra, and the intermediate Parquet files. For example, it contains the AWSGlueServiceRole, AmazonS3FullAccess, and AmazonKeyspacesFullAccess managed policies. | AWS DevOps |
| Modify the reference configuration files. | Copy CassandraConnector.conf and KeyspacesConnector.conf to the ../glue/conf directory in the project folder. | AWS DevOps |

## Troubleshooting

|  |  |
| --- | --- |
| **Issue** | **Solution** |
| AWS Glue jobs failed and returned an Out of Memory (OOM) error. | 1. Change the worker type (scale up). For example, change G0.25X to G.1X or G.1X to G.2X. Alternatively, increase the number of DPUs per AWS Glue job (scale out) in CQLReplicator. 2. Start the migration process from the point where it was interrupted. To restart failed CQLReplicator jobs, rerun the --state run command with the same parameters. |

## Resources

* [CQLReplicator with AWS Glue README.MD](https://github.com/aws-samples/cql-replicator/blob/main/glue/README.MD)
* [AWS Glue documentation](https://docs.aws.amazon.com/glue/latest/dg/what-is-glue.html)