



Amazon Web Services

RedShift

Contents

- ✓ Data Warehousing Concepts
- ✓ Columnar Databases - Concepts
- ✓ Amazon Redshift
- ✓ Creating Redshift Cluster
- ✓ Working with Redshift
- ✓ Redshift Snapshots



Data Warehouse

- A Data Warehouse stores data from multiple sources for analytical purposes.
- Data Warehouses are designed for OLAP which is used to integrate copies of transactional data from other systems and use it for analytical purposes.



Amazon Redshift

- Amazon Redshift is a fully managed, petabyte-scale **data warehouse** service in the cloud. Amazon RedShift is a **columnar database**
- An Amazon Redshift data warehouse is a collection of computing resources called **nodes**, which consists of a **leader** node and **one or more compute nodes**, which are organized into a group called a **cluster**.
 - The type and number of compute nodes that you need depends on the size of your data, the number of queries you will execute, and the query execution performance that you need.
 - Each cluster runs an Amazon Redshift engine and contains one or more databases.
- Amazon Redshift integrates with various data loading and ETL tools and BI reporting, data mining, and analytics tools.
- Amazon Redshift is **based on industry-standard PostgreSQL..**



Columnar Database

name	age	sex	zipcode
thomas	18	male	1416
martin	33	male	1645
bob	25	male	1613

name	age	sex	zipcode
thomas	18	male	1416
martin	33	male	1645
bob	25	male	1613

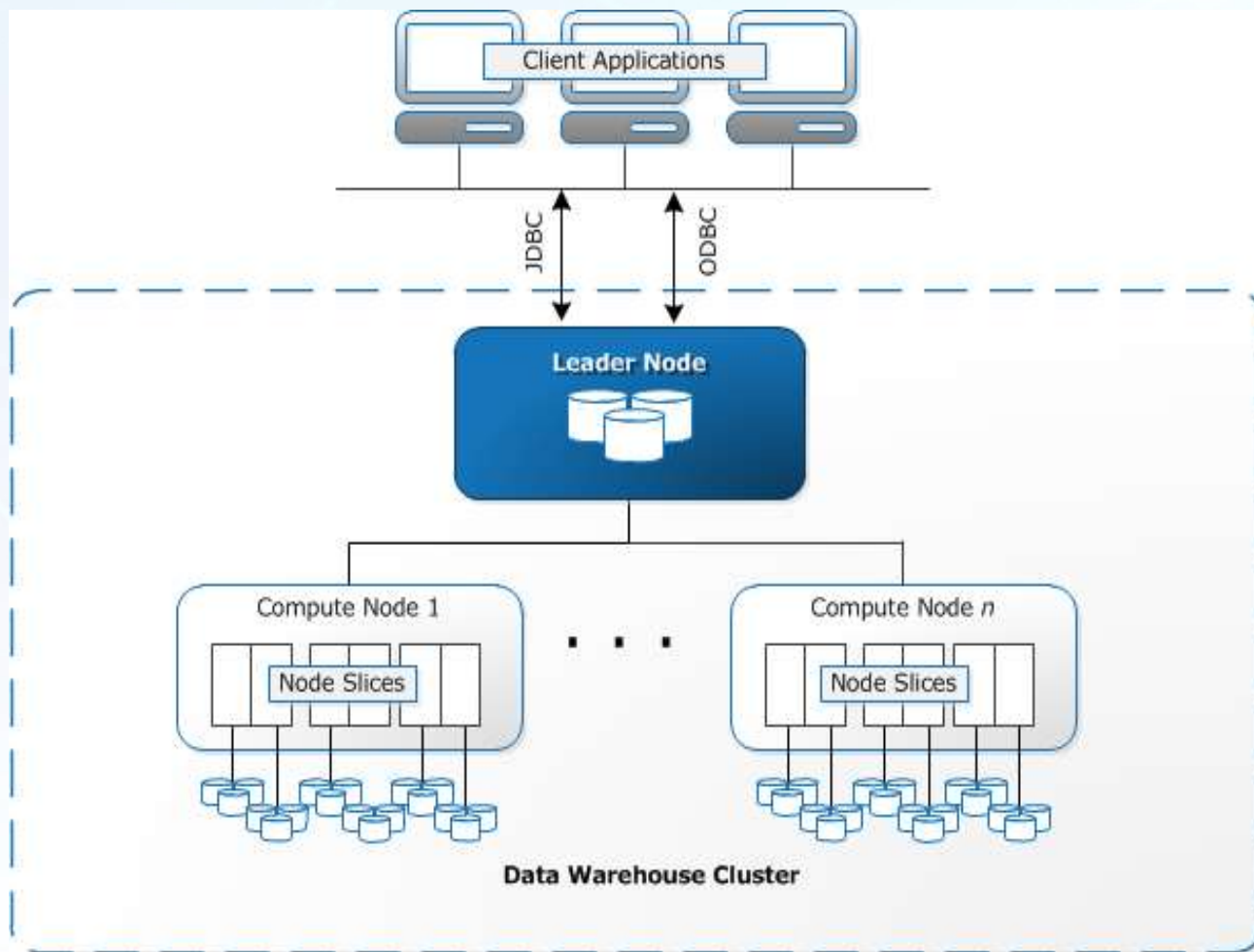


Amazon Redshift Cluster

- The core infrastructure component of an Amazon Redshift data warehouse is a cluster. A cluster is composed of one or more compute nodes.
- If a cluster is provisioned with two or more compute nodes, an additional leader node coordinates the compute nodes and handles external communication.
- Your client application interacts directly only with the leader node. The compute nodes are transparent to external applications.



Amazon Redshift Cluster



Amazon Redshift Cluster - Leader & Compute Nodes

- **Leader Node**

- The leader node manages communications with client programs and all communication with compute nodes.
- It parses and develops execution plans to carry out database operations, in particular, the series of steps necessary to obtain results for complex queries. Based on the execution plan, the leader node compiles code, distributes the compiled code to the compute nodes, and assigns a portion of the data to each compute node.
- The leader node distributes SQL statements to the compute nodes only when a query references tables that are stored on the compute nodes. All other queries run exclusively on the leader node.



Amazon Redshift Cluster - Leader & Compute Nodes

- **Compute Node**

- The leader node compiles code for individual elements of the execution plan and assigns the code to individual compute nodes. The compute nodes execute the compiled code and send intermediate results back to the leader node for final aggregation.
- Each compute node has its own dedicated CPU, memory, and attached disk storage, which are determined by the node type. As your workload grows, you can increase the compute capacity and storage capacity of a cluster by increasing the number of nodes, upgrading the node type, or both.
- Redshift provides several node types for your compute and storage needs – RA3, DC2 (and DS2 which generally is not recommended)



Create Redshift Cluster - Create IAM Role

- **Create an IAM role**
 - Select IAM service
 - Select Roles (option in left side menu)
 - Select Redshift (as a Service)
 - Select Redshift Customizable (as Use case)
 - Click next to go to Permissions
 - Select AmazonS3ReadOnlyAccess as a policy
 - Click next to add Tags
 - No need to add any Tags here. Just click next Review
 - Add a Role name (ex: DemoRoleRedshift)
 - Click on create Role button.



Create Redshift Cluster - Create a Security Group

- **Create a Security Group**
 - Select EC2 service
 - Select 'Security Groups' option
 - Click on 'Create Security Group' Button
 - Give a name to your security group (ex: DemoSecurityGroupRedshift)
 - Add an Inbound Rule
 - Allow all traffic from Redshift

Inbound rules Info						
Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
sgr-093fe8197a20b3256	Redshift ▼	TCP	5439	Custom ▼	<input type="text" value="0.0.0.0/0"/>	<input type="text" value=""/>
				<input type="button" value="Delete"/>		



Create Redshift Cluster - Launch Cluster

- Go to Redshift dashboard & click on **Launch Cluster**
- Fill the Cluster Details
 - Cluster identifier (ex: demo-cluster-redshift)
 - Type (Select Free Trail)
 - Database Configuration
 - Admin User Name
 - Admin Password
- Fill Node Configuration details
 - Node Type – dc2large
 - Cluster Type – Single Node
- Fill Additional Configuration details
 - Select the security group
 - AIM role – select the one created in previous steps.
 - Leave all others ad defaults
- Review and Launch Cluster



Working with Redshift cluster - Query Editor

- Launch the Query Editor
 - NOTE: Query editor can be used to launch queries that can be completed within 2 minutes.
 - Queries would timeout automatically after 2 minutes.
- Run a few SQL commands:
 - create table students (id int, name varchar(30), age int);
 - insert into students values
(100, 'Raju', 45),
(101, 'Ramesh', 40),
(102, 'Vijay', 35);



Public S3 buckets

- There are a few public s3 buckets that we can use to practice loading data to Redshift.
- Query the following S3 buckets using AWS CLI commands.

```
aws s3 ls s3://awssampleduswest2/ticket/
```

```
aws s3 ls s3://awssampleduswest2/ssbgz/
```



COPY command

```
create database ctsdemo;

CREATE TABLE category_pipe( catid smallint, catgroup
varchar(10), catname varchar(10), catdesc varchar(50));

copy category_pipe
from
    's3://awssampledbuswest2/tickit/category_pipe.txt'
Credentials
    'aws_iam_role=IAM_ARN'
region
    'us-west-2';
```

NOTE: Replace IAM_ARN with the actual value of the IAM ARN attached to the cluster.



Splitting data into multiple files

- While loading very large files into Redshift cluster, it is advised to split files into multiple splits. By having multiple splits we can leverage the parallel processing of files using multiple slices of the compute nodes.
 - You can split the files by having common prefix for the filenames or by explicitly listing the files in a manifest file.
- Divide your data file into splits in such a way that the number of splits is a multiple of number slices in your cluster.
 - For example, ds2.xl has two slices per compute node
 - If your cluster has two ds2.xl nodes, split your data into 4 files or multiple of 4.



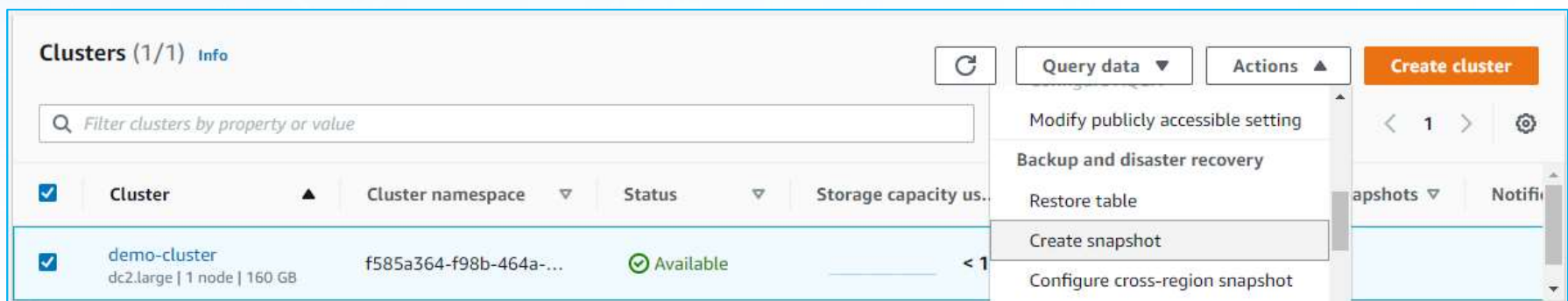
Creating a snapshot

- Snapshots are point-in-time backups of a cluster. There are two types of snapshots: ***automated*** and ***manual***.
- Amazon Redshift stores these snapshots internally in Amazon S3 by using an encrypted SSL connection.



Creating snapshot of a cluster

- You can create a snapshot of a cluster either explicit or while deleting the cluster.
- To create a snapshot, select the cluster, go to Actions menu and select create snapshot option. Mention the retention period and other details and create.
- Other way to create a snapshot is while deleting the cluster. Mention a snapshot name while deleting the cluster. The snapshot will be created and then the cluster is deleted.



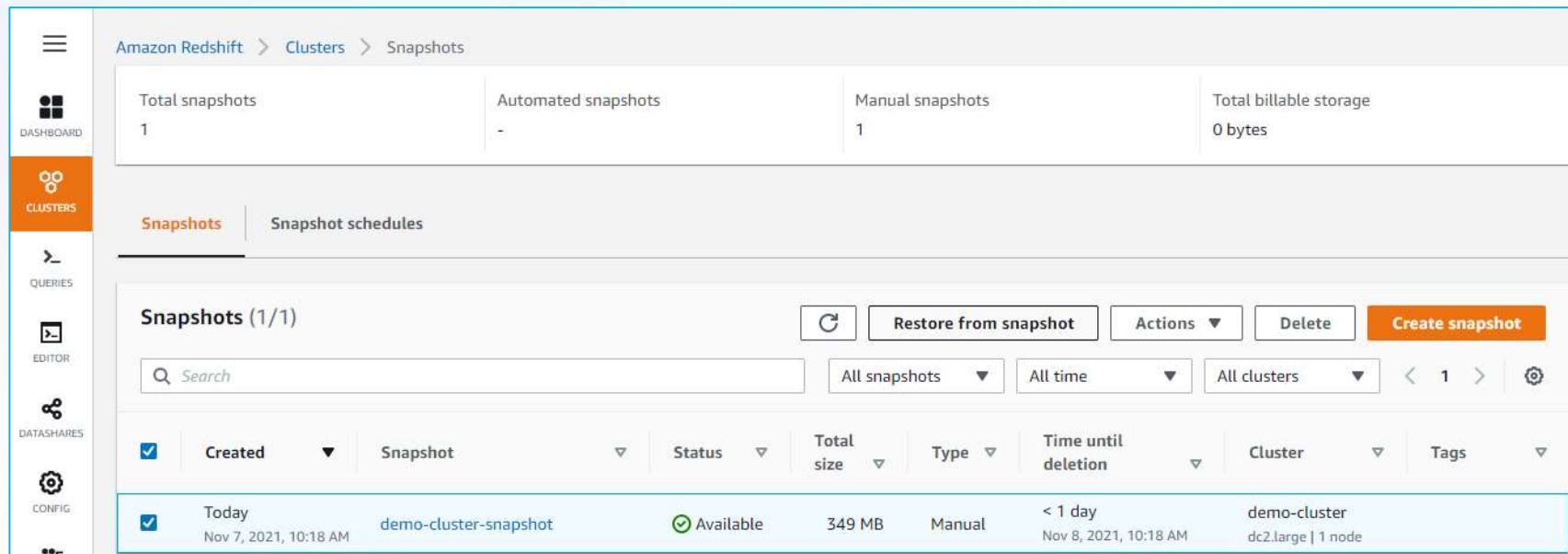
The screenshot shows the Amazon EMR console interface. At the top, there's a header 'Clusters (1/1) Info'. Below it is a search bar 'Filter clusters by property or value'. A table lists the clusters, with one cluster 'demo-cluster' selected. The cluster details are: 'demo-cluster', 'dc2.large | 1 node | 160 GB', 'f585a364-f98b-464a-...', and 'Available'. To the right of the table is an 'Actions' menu with options: 'Query data', 'Modify publicly accessible setting', 'Backup and disaster recovery', 'Restore table', 'Create snapshot', and 'Configure cross-region snapshot'. The 'Create snapshot' option is highlighted. On the far right, there's a 'Create cluster' button and a pagination control showing '1'.

Cluster	Cluster namespace	Status	Storage capacity us...
<input checked="" type="checkbox"/> demo-cluster dc2.large 1 node 160 GB	f585a364-f98b-464a-...	Available	< 1



Restoring a cluster from snapshot

- Go to Redshift Dashboard. Select Clusters >> Snapshots option from the menu.
- Select the snapshot and click on **Restore from snapshot** button.



The screenshot shows the Amazon Redshift console interface. The breadcrumb navigation at the top reads "Amazon Redshift > Clusters > Snapshots". A summary section at the top provides the following statistics:

Total snapshots	Automated snapshots	Manual snapshots	Total billable storage
1	-	1	0 bytes

Below the summary, the "Snapshots" tab is selected, showing "Snapshots (1/1)". The interface includes a search bar, a refresh button, and buttons for "Restore from snapshot", "Actions", "Delete", and "Create snapshot". Filter dropdowns are set to "All snapshots", "All time", and "All clusters". A table lists the snapshot details:

<input checked="" type="checkbox"/>	Created	Snapshot	Status	Total size	Type	Time until deletion	Cluster	Tags
<input checked="" type="checkbox"/>	Today Nov 7, 2021, 10:18 AM	demo-cluster-snapshot	Available	349 MB	Manual	< 1 day Nov 8, 2021, 10:18 AM	demo-cluster dc2.large 1 node	





**THANK
YOU**