Sizing Cloud Data Warehouses

Recommended Guidelines to Sizing a Cloud Data Warehouse

January 2019

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents AWS's current product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS's products or services are provided "as is" without warranties, representations, or conditions of any kind, whether express or implied. AWS's responsibilities and liabilities to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Abstract

This whitepaper describes a process to determine an appropriate configuration for your migration to a cloud data warehouse. This process is appropriate for typical data migrations to a cloud data warehouse, such as Amazon Redshift. The intended audience includes database administrators, data engineers, data architects, and other data warehouse stakeholders.

Whether you are performing a PoC (proof of concept) evaluation, a production migration, or are migrating from an on-premises appliance or another cloud data warehouse, you can follow the simple guidelines in this whitepaper to help you increase the chances of delivering a data warehouse cluster with the desired storage, performance, and cost profile.

Introduction

One of the first tasks of migrating to any data warehouse is sizing the data warehouse appropriately by determining the appropriate number of cluster nodes and their compute and storage profiles. Fortunately, with cloud data warehouses such as Amazon Redshift, it is a relatively straightforward task to make immediate course corrections to resize your cluster up or down. However, sizing a cloud data warehouse based on the wrong type of information can lead to your PoC evaluations and production environments being executed on suboptimal cluster configurations. Resizing a cluster might be easy, but repeating PoCs and dealing with change control procedures for production environments can potentially be more time consuming, risky, and costly, which puts your project milestones at risk.

Migrations of several petabytes to exabytes of uncompressed data typically benefit from a more holistic sizing approach that factors in existing data warehouse properties, data profiles, and workload profiles. Holistic sizing approaches are more involved and fall under the category of professional services engagement. For more information, contact AWS.

Sizing Guidelines

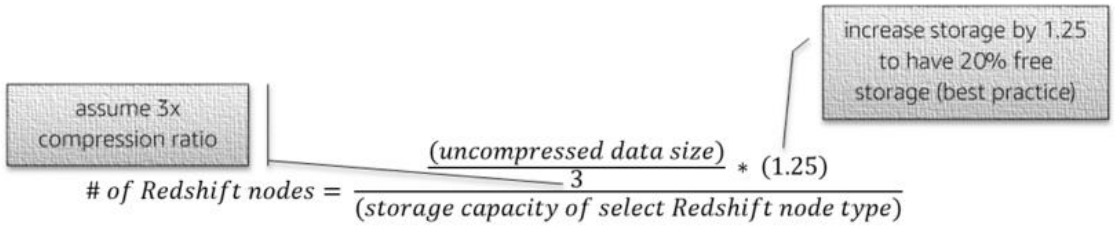
For migrations of approximately one petabyte or less of uncompressed data, you can use a simple, storage-centric sizing approach to identify an appropriate data warehouse cluster configuration.

With the simple-sizing approach, your organization's uncompressed data size is the key input for sizing your Redshift cluster. However, you must refine that size a little.

Redshift typically achieves 3x–4x data compression, which means that the data that is persisted in Redshift is typically 3–4 times smaller than the amount of uncompressed data.

In addition, it is always a best practice to maintain 20% of free capacity in a Redshift cluster, so you should increase your compressed data size by a factor of 1.25 to ensure a healthy amount (20%) of free space.

The simple-sizing approach can be represented by this equation:



This equation is appropriate for typical data migrations, but it is important to note that sub-optimal data modelling practices could artificially lead to insufficient storage capacity.

Amazon Redshift has four basic node types—or instance types—with different storage capacities. For more information on Redshift instance types, see the Amazon Redshift Clusters documentation.

Basic Redshift cluster information

Instance Family	Instance Name	vCPUs	Memory	Storage	# Slices
Dense-storage	ds2.xlarge	4	31GiB	2TB HDD	2
	ds2.8xlarge	36	244GiB	16TB HDD	16
Dense-compute	dc2.large	2	15.25GiB	160GB SSD	2
	dc2.8xlarge	32	244GiB	2.56TB SSD	16

In an example scenario, the fictitious company, Example.com, would like to migrate 100TB of uncompressed data from its on-premises data warehouse to Amazon Redshift. Using a conservative compression ratio of 3x, you can expect that the compressed data profile in Redshift wil decrease from 100TB to approximately 33TB. You factor in an additional 20% size increase to ensure a healthy amount of free space, which will give you approximately 42TB of storage capacity in your Redshift cluster.

You now have your target storage capacity of 42TB. There are multiple Redshift cluster configurations that can satisfy that requirement. The Example.com VP of Data Analytics wants to start out small, select the least expensive option for their cloud data warehouse, and then scale up as necessary. With that extra requirement, you can configure your Redshift cluster using the dense-storage, *ds2.xlarge* node type which has 2TB of storage capacity. With this information, your simple-sizing equation is:

$$\#\ of\ Redshift\ ds2.xlarge\ nodes = \frac{\frac{100\ TB}{3} * (1.25)}{2\ TB} = \frac{42\ TB}{2\ TB}$$
$$= \mathbf{21\ ds2.xlarge\ nodes}$$

You should also consider the following information about this example Redshift cluster configuration:

Cluster Type	Instance Type	Cluster Capacity				Cost ($/month)
		Nodes	Memory	Compute	Storage	
Dense-storage	ds2.xlarge	21	651Gb	84 Cores	42TB	$x
	ds2.8xlarge	3	732Gb	108 Cores	48TB	$1.2x
Dense-compute	dc2.8xlarge	17	4,148Gb	544 Cores	44TB	$4.52x

If initial testing shows that the Redshift cluster you selected is under or over powered, you can use the straightforward resizing capabilities available in Redshift to scale the Redshift cluster configuration up or down for the necessary price and performance.

Redshift Cluster Resize

After your data migration from your on-premises data warehouse to the cloud is complete, over time it is normal to make incremental node additions or removals from your cloud data warehouse. These changes help you to maintain the cost, storage, and performance profiles you need for your data warehouse. With Amazon Redshift, there are two main methods to resize your cluster:

- **Elastic resize** – Your existing Redshift cluster is modified to add or remove nodes, either manually or with an API call. This resize typically requires approximately 15 minutes to complete. Some tasks might continue to run in the background, but your Redshift cluster is fully available during that time.
- **Classic resize** – Enables a Redshift cluster to be reconfigured with a different node count and instance type. The original cluster enters read-only mode during the resize, which can take multiple hours.

In addition, Amazon Redshift supports concurrency-based scaling, which is a feature that adds transient capacity to your cluster during concurrency spikes. This, in effect, temporarily increases the number of Amazon Redshift nodes processing your queries. With **concurrency scaling**, Redshift automatically adds transient clusters to your Redshift cluster to handle concurrent requests with consistently fast performance. This means that your Redshift cluster is temporarily scaled up with additional compute nodes to provide increased concurrency and consistent performance.

For more information about resizing a Redshift cluster, see:

- **Resizing a Cluster (Redshift Documentation)**
  https://docs.aws.amazon.com/redshift/latest/mgmt/working-with-clusters.html#cluster-resize-intro
- **Elastic Resize (Redshift Documentation)**
  https://aws.amazon.com/about-aws/whats-new/2018/11/amazon-redshift-elastic-resize/
- **Elastic Resize (Blog Post)**
  https://aws.amazon.com/blogs/big-data/scale-your-amazon-redshift-clusters-up-and-down-in-minutes-to-get-the-performance-you-need-when-you-need-it/
- **Concurrency Scaling (Blog Post)**
  https://www.allthingsdistributed.com/2018/11/amazon-redshift-performance-optimization.html

Conclusion

It is important that you size your cloud data warehouse using the right information and approach. Although it is easy to resize a cloud data warehouse (such as Amazon Redshift) up or down to achieve a different cost or performance profile, the change control procedures for modifying a production environment, repeating a PoC evaluation, etc. could pose significant challenges to project milestones. You can follow the simple sizing approach outlined in this whitepaper to help you identify the appropriate cluster configurations for your data migration.

Contributors

Contributors to this document include:

- Asser Moustafa, Solutions Architect Specialist, Data Warehousing
- Thiyagarajan Arumugam, Solutions Architect Specialist, Data Warehousing

Document Revisions

Date	Description
January 2019	First publication