A comprehensive guide to

Redshift Optimization

Copyright 2023.Agilisium

Index

All Rights Reserved | Copyright 2023. Agilisium

Building an AWS well-architected (WAF) data warehouse on Redshift

With thousands of customers, Redshift is the most widely adopted Enterprise Data Warehouse. In the last 18 months alone, over 200 new features have been added to Redshift, helping it maintain an edge over its competition in terms of performance and predictable cost. Having worked closely with the product team as an AWS Advanced Consulting Partner for Data and Analytics, it is apparent that Redshift offers a gamut of advantages. However, in our interactions with clients, we found that maintaining the initial enthusiasm of when they migrated to Redshift is challenging for most organizations.

The primary cause of this drop in enthusiasm is that today, architects play a crucial role in designing enterprise solutions on the cloud. However, this architect community struggles to keep up with the rapid pace of innovation due to limited time and bandwidth; they cannot experiment, learn new features, and the latest best practices. Subsequently, an organization's ability to extract the maximum value from their existing Redshift investments is severely curtailed.

In this two-part blog series on Redshift Optimization, we will detail, 1) the critical design considerations for building an AWS well-architected (WAF) data warehouse on Redshift and 2) how organizations can optimize their EDW in the face of rapidly changing business needs without compromising the WAF pillars.

Key Design Considerations for a Redshift EDW built leveraging WAF

While working within a Redshift ecosystem, be it a completely new implementation or reviewing an existing setup, it must incorporate the five fundamental design principles laid out by AWS as part of the well-architected framework. Considering the WAF helps to build stable and efficient systems, which organizations can leverage to be agile and keep up with changing business needs. As listed below, the five pillars provide a consistent approach to evaluate and implement designs that can scale with an organization's application needs over time.

- Cost Optimization
- Performance Efficiency
- Security
- Reliability
- Operational excellence

Let us take a brief look at how an architect approaches an EDW's design, taking each of these five pillars into consideration.

Cost Optimization

The key to eliminating or avoiding unnecessary costs comes from understanding how Redshift clusters consume resources. This will mean answering questions like,

- How much cloud-capacity would be needed?
- Are you choosing the right size for the cluster?
- Is there compensation in performance?
- Are the choices cost-optimal?

It is essential to understand consumption as the Redshift environment set up is heavily influenced by it. An architect may choose to apply one or several of the strategies listed below based on consumption and workload patterns,

- Right-sizing cluster for optimal cost or speed rather than peak workloads
- Pricing strategies like choosing reserved capacity for regular workloads
- Ensuring the right type of nodes, and increasing or decreasing node type according to changing workloads

An architect will also turn to benchmarking to arrive at optimal requirements and use the most cost-effective resources. Choosing appropriate instances and resources is the first step in cost saving.

Performance Efficiency

The performance efficiency pillar provides tips on measuring the performance of the workload. While evaluating performance for a Redshift EDW, an architect tries to understand running workloads and answer questions like –

- How many workloads are run in parallel?
- How efficiently are they using the cluster?
- Is the data and workload distribution optimal?
- How is memory utilization?
- Are there any unutilized or under-utilized resources?

The overall idea is to measure the workload's performance and optimize resources and scale based on demand. As demand changes, regular performance review can help unearth issues that need attention. The focus is on a data-driven approach, making acceptable tradeoffs to arrive at a high-performing architecture.

Security

The security pillar aims to protect the information, systems, and assets in the cloud. From creating an identity foundation and enabling data traceability, to ensuring security at every layer, the aim is to protect data in transit and rest.

A critical security consideration is to ensure that an organization's network and data security design are well thought out based on their policy and performance needs. Organizations must be able to control who can do what, prevent security incidents, and identify and take immediate action in security breaches. This requires a well-defined process, one that maintains confidentiality and complies with regulatory requirements.

An architect also takes strategic advantage of the AWS Shared Responsibility Model, where AWS maintains the cloud's security, and an organization takes care of security in the cloud. Another critical security component is the continuous monitoring and auditing of cloud deployment.

Reliability

Business continuity is the key issue addressed through the Reliability pillar. While designing a Redshift architecture for reliability, an architect ensures that the following considerations are fulfilled.

- Is the Redshift ecosystem resilient towards both internal, external destructions?
- Is the solution highly available?
- Are relevant disaster recovery and backup procedures in place?
- Are the recovery measures tested?

Any ecosystem must be able to recover from infrastructure or service disruptions. A well-architected Redshift EDW is designed to detect most failures and automatically heal itself, creating a 'Reliable' ecosystem.

Operational Excellence

The Operational Efficiency pillar of the Well-Architected Framework focuses on ensuring continuous operation and management of an organization's ecosystem. An architect incorporates several techniques, processes, and strategies to achieve this – from performing operations as code, making small, frequent, reversible changes, to refining procedures frequently. The aim is to achieve the smooth functioning of all processes.

Manual processes cannot keep up with today's pace of change. Hence, an architect leverages modern and automated processes like DevOps and CI/CD processes and evolves said processes as requirements change. Centralized monitoring and logging are built into the ecosystem to help study processes in failure or error and provide necessary recommendations. Learning from prior performance, validating procedure, and anticipating failure take prime importance in implementing a successful process.

Proven strategies to optimize Amazon Redshift for cost and performance

According to a survey conducted by TDWI in partnership with a cloud services company, one of the top reasons that companies migrate to cloud data warehousing is to take advantage of the flexible costs they offer. However, cost optimization is the most common challenge enterprises face as changing workloads affect the cost and performance of even the most well-built data warehouse. In this blog, let's talk about proven optimization strategies that can help enterprises get the most out of their Amazon Redshift investment.

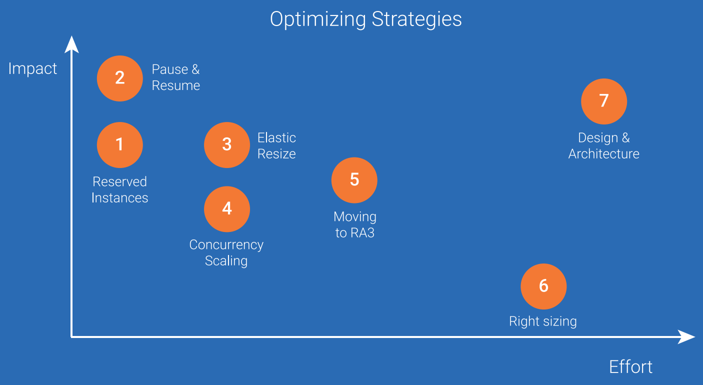
The Optimization Processes

Before we can look at the strategies, we need to look at the optimization process. we laid out how to design a well-architected Redshift ecosystem. The natural question that follows is 'How' to optimize it as an enterprise grows.

The foremost thing to remember is that optimization is a continuous process. For instance, today, the cluster can have optimal balance between costs and performance. As mentioned earlier, as the organization grows, there could be many factors that disrupt this balance, including but not limited to,

- Increase in data volume
- Change in the Business demand
- Technology evolution

Either way, regularly revisiting enterprise data warehouse set up is crucial to an optimized system.



Optimization strategies plotted on an impact vs. effort matrix

Let us now look at some of the practices that can make your Redshift ecosystem more efficient and cost-optimized. The optimization strategies are discussed in terms of the complexity of effort involved vs. the impact on performance. While some of them are easy and straightforward, others require significant planning and may include architecture changes.

However, it is essential to remember that optimizing for cost alone is not the goal. A performant system with strategic cost optimizations is the ideal outcome.

1. Using Reserved instances

AWS allows us to purchase reserve nodes for steady-state workloads by choosing the duration of commitment between 1 to 3 years. There is also the option of choosing the cost mode, which can be

- No upfront
- Partial Upfront
- Or All upfront

The underlying AWS cost principle here simply is,"The more you reserve and the more you pay upfront, the higher the discounts." Simply put, maximum discounts (up to 70 % compared to On-Demand instances) can be obtained when opting for 3-year reserved instances with all upfront payment.
This costing is ideal for enterprises with immense workloads. However, medium-scale establishments, or startups on the path of growth, typically start with on-demand instances. Depending on the duration of use and demand, such smaller companies can periodically revisit their cost strategy and make a gradual move to reserved instances for fixed workloads, while still on-demand instances for all other applications.

The other important point to note here is that reserved instances are ideal when used for always-on instances. Here's a real-life use case we've seen,

One of our clients spent around $110K for on-demand DS2.8XLarge, with two node instances for the past eight months. We saved 25% of their cost merely by moving to reserved instances with one year of commitment and no upfront charges. It is that simple!

2. Pause and Resume

Pause and resume is another simple yet effective approach to optimization.

Consider this scenario – one of our clients was using on-demand DC2.Large, 24*7 as part of their development instance. When we found that the CPU utilization dropped to 10% during non-business hours /weekends or holidays compared to 60% during business hours, we suggested halting these resources when not in use. Just enabling the 'pause and resume' option helped them save cost by 50%.

What is happening here? Why does pausing an instance correspond to cost savings? When on-demand instances are paused, only storage charges are applicable! You do not have to pay for compute when no computation is happening. Organizations using on-demand instances exclusively will stand to gain the most by using this option.

3. Elastic Resize

Organizations prefer a data warehouse that is faster to scale and do not want to compromise between performance and concurrency. The perfect solution to that is elastic resize. Elastic resize allows you to scale a cluster up or down within minutes.

Suppose organizations have currently planned their resources for their peak capacity of clusters by just enabling the elastic resize option. In that case, they can obtain significant cost savings, to the order of 15%.

However, you need to keep a few factors in mind while using the elastic resize option

- You cannot scale more than double the number of existing nodes using elastic resize. You will need to use the classic resize option.
- Scaling down of instances to the previous size of clusters or less than half the number of clusters may or may not be possible depending on the storage requirement.

4. Concurrency Scaling

Another interesting feature that impacts Redshift performance is the Concurrency Scaling, which is enabled at the workload management (WLM) queue level. The WLM allows users to manage priorities within workloads in a flexible manner. With the help of this feature, short, fast-running queries can be moved to the top of long-running queues.

The concurrency scaling feature automatically spins up transient clusters, serves the requests in the queue, and automatically scales down the clusters. You can support virtually unlimited concurrent users and queries, without compromising on query performance. What we need to know here is that Concurrency scaling is used only for reporting purposes.

For every 24 hours, the cluster is in use; the account is credited with 1 hour of free concurrency scaling by AWS. These free concurrency hours can be accumulated and used effectively during predictable workloads. On exceeding the accumulated hours, concurrency scaling is billed at per second cost of cluster price

5. Using the latest RA3 instances

One of the most welcome Amazon Redshift features in recent times has been the introduction of RA3 instances. RA3 instances allow us to scale compute and storage, independently, which means that the pricing is also loosely coupled with compute and storage.

In terms of benefits, we have seen up to 200% improvement in performance during the benchmarking exercise, which we recently conducted between RA3 vs. DS2.Xlarge. Since RA3 nodes are based on a large cache capacity, with high-performance SSD backed by S3, it is good to validate your instances to see if the RA3 can be a better fit for your storage and compute needs.

6. Right-sizing

Currently, Redshift provides 3 flavors of instances,

- Dense Compute – ideal for high reporting and minimal storage demands
- Dense storage – used for high storage
- RA3 instances

Choosing the optimal size of production instances helps in lowering the cost between 15-20%. It is always good to start sizing based on workloads depending on CPU, I/O, disk, and network requirements.

7. Resolving long-running queries

The last optimization strategy is the most complex of the lot because it impacts the current design & architecture of the existing system.

The most common issue that organizations face today is long-running queries and inability to achieve parallelism. The underlying factors that cause this are:

Data skew across nodes.
A high amount of I/O
Data not being compressed optimally

To resolve this, we need to look at the key strategies to improve performance and pay attention to distribution keys, sort keys, and encoding aspects.

- Use proper distribution keys, which help minimize data movement across nodes.
- AUTO distribution can be considered for tables with less than 5 Million records.
- For Optimal performance, choose columns used in Joins/filters as sort key.
- Be cautious in using Interleaved Sort Keys as it will add more overhead.

A new encoding type AZ64 has been included. AZ64 is Amazon's proprietary compression encoding algorithm targets high compression ratios and better processing of queries. Using the AZ64, we see close to 30% storage benefits and a 50% increase in performance compared with LZO and ZSTD coding methods.

Along with these best practices, we can also look at the benefits of Redshift materialized views to improve long-running queries.

Bonus: Leverage Auto Management along with optimization strategies

Having discussed optimization strategies, we have an additional tip. Leveraging Redshift's Auto workload management (WLM) feature in addition to the optimization strategies mentioned above leads to a more effective system. Redshift is gradually working towards Auto Management, where machine learning manages your workload dynamically.

Auto WLM involves applying machine learning techniques to manage memory and concurrency, thus helping maximize query throughput. Through WLM, Redshift manages memory and CPU utilization based on usage patterns. Auto WLM lets you prioritize your queries, ensuring that high-priority queries execute first as queries are continually submitted.

To leverage Auto Management,

- Ensure that AUTO ANALYZE, AUTO SORT, and AUTO VACUUM are enabled.
- If you are using interleaved sort keys, you need to run VACUUM REINDEX.

Nevertheless, if you choose to manage your workloads manually, some of the best practices are as follows.

- Do not create more than four queues
- Use QWR to monitor performance from bad queries
- Do not have more than 15 concurrent users

It is always a good practice to leave at least 5% memory unallocated.

Conclusion

The first step for an organization to utilize Redshift to its fullest is to set up an ecosystem that is stable, agile, and robust by following AWS's well-architected framework. An architect(s) spend time and effort carefully considering the 5 WAF to achieve the same. However, even the most well-architected EDW suffers from increased cost and performance issues as it evolves to keep up with changing business needs.

With an ongoing, well thought out optimization plan, an organization should and can keep its Redshift DW performant and its costs down and even as the DW evolves. By choosing Agilisium's Redshift Optimization Program, organizations can take a holistic look at their Redshift workloads to identify challenges and address them thoroughly and rapidly through a phased approach, extracting more value from AWS Redshift in the process.

That's not all; maintaining the balance between cost, agility, and performance is built into the program allowing organizations to reap continual benefits.

Competencies

aws PARTNER Advanced Tier Services

- Data & Analytics Services Competency
- Migration & Modernization Services Competency
- Immersion Day
- DevOps Services Competency
- Well-Architected Partner Program
- Microsoft Workloads Consulting Competency

SDPs

aws PARTNER Advanced Tier Services

- Amazon EMR Delivery
- Amazon RDS Delivery
- AWS Lambda Delivery
- Amazon Kinesis Delivery
- Amazon DynamoDB Delivery
- AWS Glue Delivery
- Amazon Redshift Delivery
- Amazon Quicksight Delivery
- Amazon EC2 for Windows Server Delivery
- Amazon OpenSearch Service Delivery

Top 3 Global Redshift Advocates for AWS customers

First AWS Partner across the world to achieve all AWS Data & Analytics competencies & SDPs

1st

Boutique AWS Partner– QuickStart solutions, Chatbots, Migration accelerators, Optimization inspector etc.

aws database freedom

Preferred Partner for AWS Database freedom program for legacy DB migrations to AWS

sales@agilisium.com

Agilisium is the fastest-growing Cloud Transformation & Data Analytics company with strong expertise in Data lake solutions, Data Warehouse Engineering, Data Migration & Modernization, Data Visualization, and Cloud Optimization services. Agilisium is an AWS Advanced Consulting Partner who helps companies architect, build, migrate, and manage their application workloads to accelerate their journey to the agile cloud, achieve desired business outcomes, and reach new emerging global markets. Learn More at www.agilisium.com.