

# HOW TO KEEP YOUR DATA LAKE "MATCH FIT"

Phil Anderson, Head of Product Management & Strategy

## PROBLEM STATEMENT

An Australian energy resources company engaged RoZetta to lead a review and redevelopment of its data lake.

The data lake has grown exponentially over recent years with the introduction of new data sources and an increasing emphasis on using data scientists to provide insights into operations and opportunities.

The data and technology architecture had not been reviewed since the inception of the data lake and was now a roadblock to progressing key decisions on resource allocation, pricing and assessment of strategic investments.

The client had the data it needed to improve its operations and business decisions but conflicts caused by the current technology design had become a roadblock.

The business as usual activities could not be disrupted during the review, and any agreed changes were implemented. The data lake had become mission critical for the business.

## PROJECT COMMENCEMENT STATE

The client collected huge amounts of data from every operational system used to manage the enterprise. This included legacy systems, production systems, administration systems, logistic applications and over 30,000 IoT sensors deployed on its infrastructure. The data lake supported the data and information needs of more than 2,000 users and many of the enterprise's administrative systems.

All these feeds were ingested into a data lake with two key components, a time series database, HBase, and a relational database, AWS Aurora.

The system had several issues caused by the data lake getting bigger without evolving the underlying architecture, causing several issues that affected the efficiency and effectiveness of the data lake:

- 1. Requesting large extracts via the time series API was inefficient because the original time series database was optimised for spot data retrieval, very small packets, rather than bulk data extracts.
- 2. Persistent data gaps existed in the production data, and in some instances, duplicate values were loaded.
- 3. The production system data feed had a mix of raw and interpolated data caused by re-using historical extracts.
- 4. A growing need to ingest alternate data types like multidimensional time-series data and video, which the current system does not support.
- 5. The original system used an expensive HBase cluster, hard to scale, and tied to a single dataset.

## PROJECT OUTCOME

#### **Operational Effectiveness:**

Evidence of the system's effectiveness is that a broad user base, not just the data science team but operational teams, executed up to 15 million queries per month without processing conflicts or delays. The data lake had become a more trusted source of information able to deliver significant access to data and empower the digitisation of the organisation through the scalable platform.

#### Continual Improvement:

The client now has a lower cost, more efficient, data lake and an established network of mentors that can be called on to maintain the innovation in a dynamic data management environment.

## ENGAGEMENT PRINCIPLES

There were key principles in the engagement with RoZetta:

- RoZetta would manage the project.
- RoZetta's role would work with the existing technology and data science teams to ensure any changes to architecture made the data lake more resilient for future use cases.
- RoZetta would take on a mentor role for the client's technology team to ensure knowledge transfer was evident.
- RoZetta and the client shared any emerging intellectual property.

## SOLUTION - TECHNICAL ELEMENTS

RoZetta made a number of recommendations on the technical design and application mix to streamline processes and improve performance and reduce the overall running cost of the data lake.

#### <sup>1</sup>. Align Data Storage with Data Demand

Utilise a combination of S3 and Redis as hot and cold storage.

The S3-based Data Lake component serves as the permanent store of all data. Data is served up via AWS Athena. Redis plays the role of a performance layer. An Elasticache Redis cluster stores a subset of data that is anticipated to be queried frequently.

#### 2. Single Channel for External Access

A Time Series API became the sole external access point to the data.

he API implements request routing so that, when appropriate, data can is retrieved from Redis, and any other data is retrieved from S3 via Athena.

This provided the end-users an abstraction of the underlying data without needing to know the intricacies of how the data is stored.

1. HBase is a column-oriented non-relational database management system that runs on top of the Hadoop Distributed File System (HDFS).

<sup>2.</sup> Amazon Aurora is a global-scale relational database service built for the cloud with full MySQL and PostgreSQL compatibility.

RoZetta configured and tuned the system such that approximately 95% of API requests are served by data in Redis. while the remaining requests are served by a combination of Redis and Athena or Athena alone.

#### 3. Ingestion Efficiencies

All data feeds are placed in a designated S3 bucket and triggers SNS notifications.

The frequency of updates varies from a few seconds, as with IoT sensor generated data, while other files arrive every minute, and yet others may arrive at longer intervals, such as data from administrative systems.

Each dataset has an instance of the ingestion system to allow for scaling and tuning per dataset. The ingestion system has a pair of SQS queues subscribing to the SNS notifications, these are, in turn, subscribed to a pair of ingestion Lambda functions, which read the incoming data and write it to the appropriate data store. The S3 ingestion functions write all data to S3 as partitioned ORC files.

To manage latency requirements, the system is now configured to invoke an instance of the ingestion function for each incoming file at the highest possible concurrency. This ensures that when a large number of input files arrive at once, they are processed in parallel as quickly as possible.

The parallel runs of the Redis ingestion function help attain low latency, minimizing the gap between when the file is placed in S3 and its contents written to Redis.

The system is designed to ingest such data into Redis within 30 seconds (99.9th percentile) with a benchmark performance of less than 10 seconds.

The parallel runs of the S3 ingestion function ensure high throughput of data, especially historical data, which come in large batches. The design will ingest up to 64 million records with a backlog latency of less than 5 minutes.

#### 4. Preventing File Fragmentation

The parallel runs of the S3 ingestion can result in large numbers of ORC files being written to S3. This can cause fragmentation and slow Athena query speeds and consequently increase AWS costs.

To prevent fragmentation RoZetta implemented a compactor service that frequently compacts the files in S3, ensuring that the data is consolidated into a smaller number of larger files. The compactor service is implemented as a Glue Job that runs on a configurable schedule.

### TECHNOLOGY COMPONENTS

The solution is flexible and highly extensible and now ingests new data sources in almost any format. From a user perspective, the data is now more accessible by an expanding user group and far more efficient in supplying extracts to data users and operational sub-systems.

Using an S3 as the basis for the data lake allows for highly scalable, durable and low-cost data storage. Removing the reliance on the HBase cluster simplified support and maintenance by implementing an integrated AWS toolset.

The design and implementation of a metadata API service allows for easier onboarding of new datasets. Added advantages were gained by separation of computing and data storage. Compute capacity now scales without having to redistribute data to new nodes as the cluster grows, while compute capacity is scaled on-demand using transient resources such as AWS Spot instances. This reduces the cost of operating the total platform.

Utilising S3 means that data is accessed from a wide and everincreasing toolset providing future proofing for the data lake. These tools enabled ingestion to scale dynamically, keeping costs low.

### BUSINESS BENEFITS

- Easy to onboard new datasets.
- The client has reduced data compute and storage costs.
- All user groups have a rapid response rate to queries for large data volumes.
- All data types, including video capture, can be loaded and accessed easily.
- Automated extracts form management information systems, operational reporting and decision systems.
- The client's staff acquired new knowledge and skills in maintaining and developing the data lake.

©2022 RoZetta Technology. All rights reserved. 3 of 4

<sup>1.</sup> ElastiCache Fully Managed Redis & Memcached Open-Source Database w/ Scalability.

Since Jables you to send notifications directly to your customers.
 Since Jables you to decouple and scale microservices, distributed systems.
 Lambda runs code on a high-availability compute infrastructure and performs all of the administration of the compute resources.
 Optimized Row Columnar (ORC) file format provides a highly efficient way to store Hive data.

## WHY ROZETTA?

RoZetta Technology has a deep knowledge and extensive experience in developing large, complex systems critical to business operations.

RoZetta has developed these systems in a range of industry verticals and brings this broader view to introducing best practices in data management and technology implementation.

Reviewing just the technology or the database design limits the business benefits a review like this can deliver. RoZetta's approach is that data management encompasses the underlying data architecture, the technology platform and the tools used in the end-to-end process of receiving and utilising data in an enterprise.

## EXAMPLE OF DIRECT BUSINESS BENEFIT



One of the subsystems RoZetta developed as part of this engagement was a fully integrated KPI management system.

The KPI System made it easier for employees to access KPI metrics and the underlying performance data. A more detailed report on the KPI system is available here.

#### REFERENCES

1. HBase is a column-oriented non-relational database management system that runs on top of the Hadoop Distributed File System (HDFS).

- 2. Amazon Aurora is a global-scale relational database service built for the cloud with full MySQL and PostgreSQL compatibility.
- 3. ElastiCache Fully Managed Redis & Memcached Open-Source Database w/ Scalability.
- SNS enables you to send notifications directly to your customers.
  Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems.
  Lambda runs code on a high-availability compute infrastructure and performs all of the administration of the compute resources.
- 7. Optimized Row Columnar (ORC) file format provides a highly efficient way to store Hive data.
- 8. Glue Job encapsulates a script that connects to source data, processes it, and then writes it out to the data target.
- 9. Spot instances are spare compute capacity in the AWS cloud available at steep discounts compared to On-Demand price.

©2022 RoZetta Technology. All rights reserved. 4 of 4