
aws
SUMMIT
LONDON | APRIL 27, 2022

AN-02

Get quicker and more valuable insights from your data, without having to manage your own data warehouse infrastructure

Carlos Contreras
Big Data and Analytics Prototyping Architect
Amazon Web Services

Toby Ayre
Head of Data & Analytics
Rail Delivery Group

Agenda

Data Warehouse: Use the right tool for the right job

Why Amazon Redshift

Performance on Amazon Redshift

New features:

- @ Data Sharing
- @ Auto { MViews | WLM | … }

Amazon Redshift Serverless

Rail Delivery Group: Customer story + Demo

“Does data lose value over time?”

Some data may lose value over time…

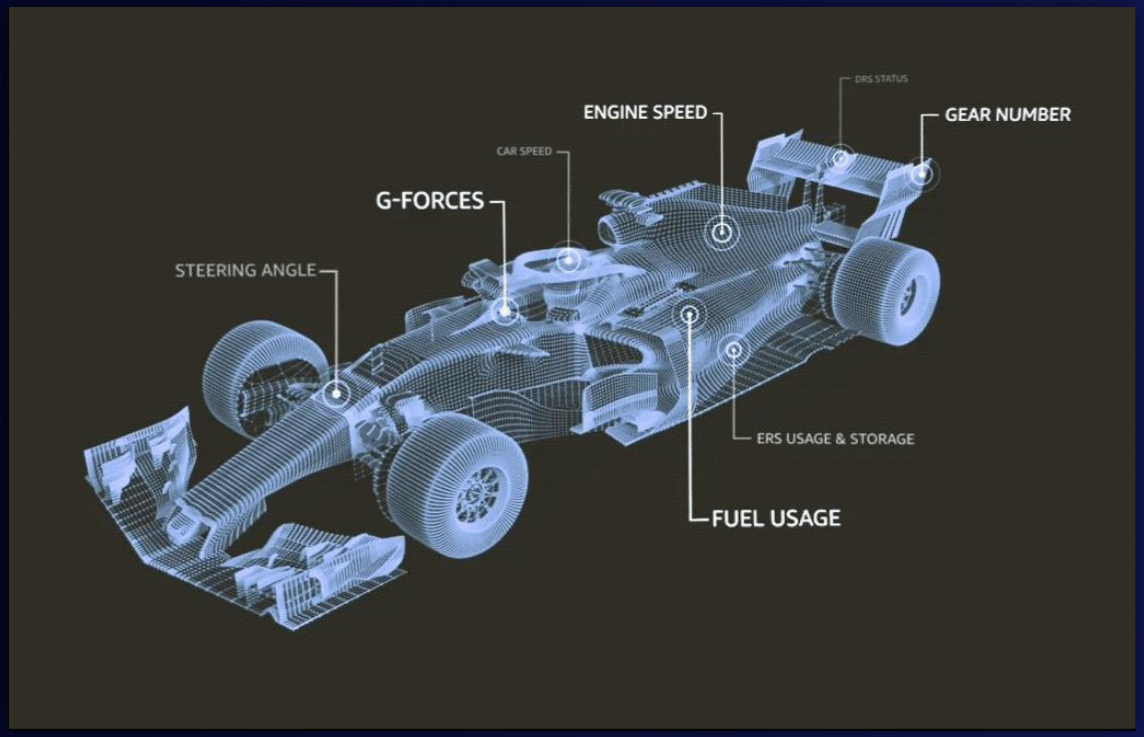
Video source: aws.amazon.com/f1

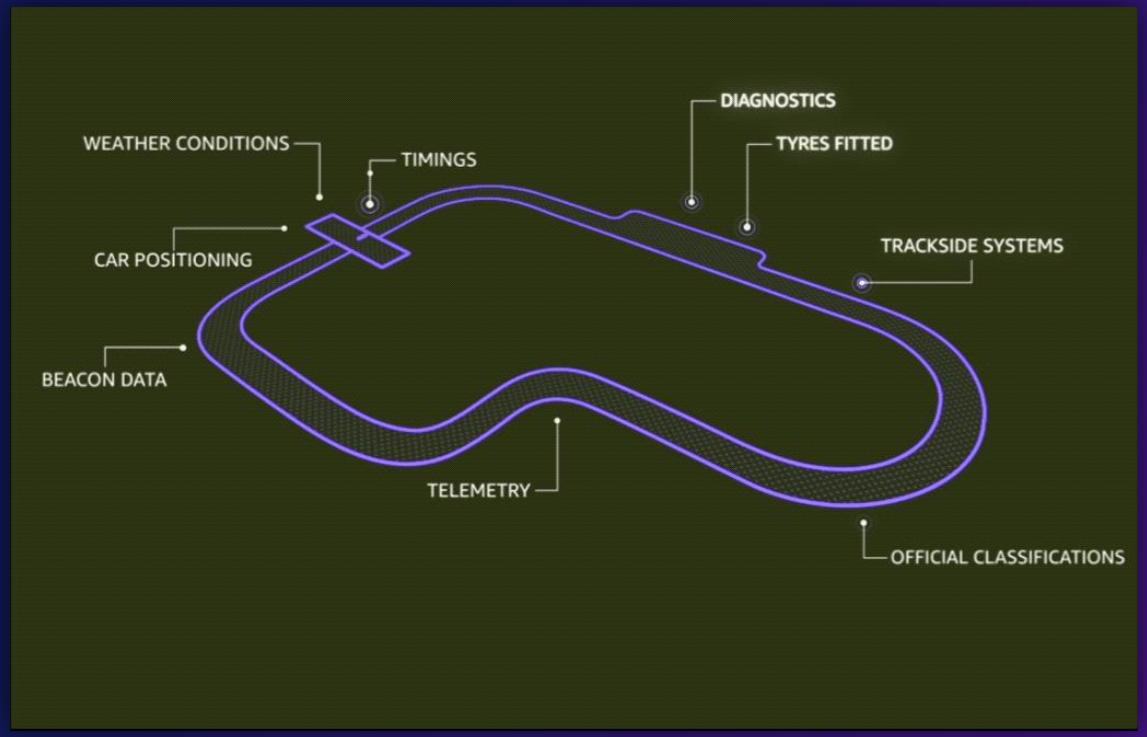
Other (historical) data may gain value over time

*"Our F1 car can record **more than 250 channels of data** simultaneously*
*…We could generate more than one **billion of data points in a race**"*

— F1 Fantastical Facts. McLaren, 2013 —

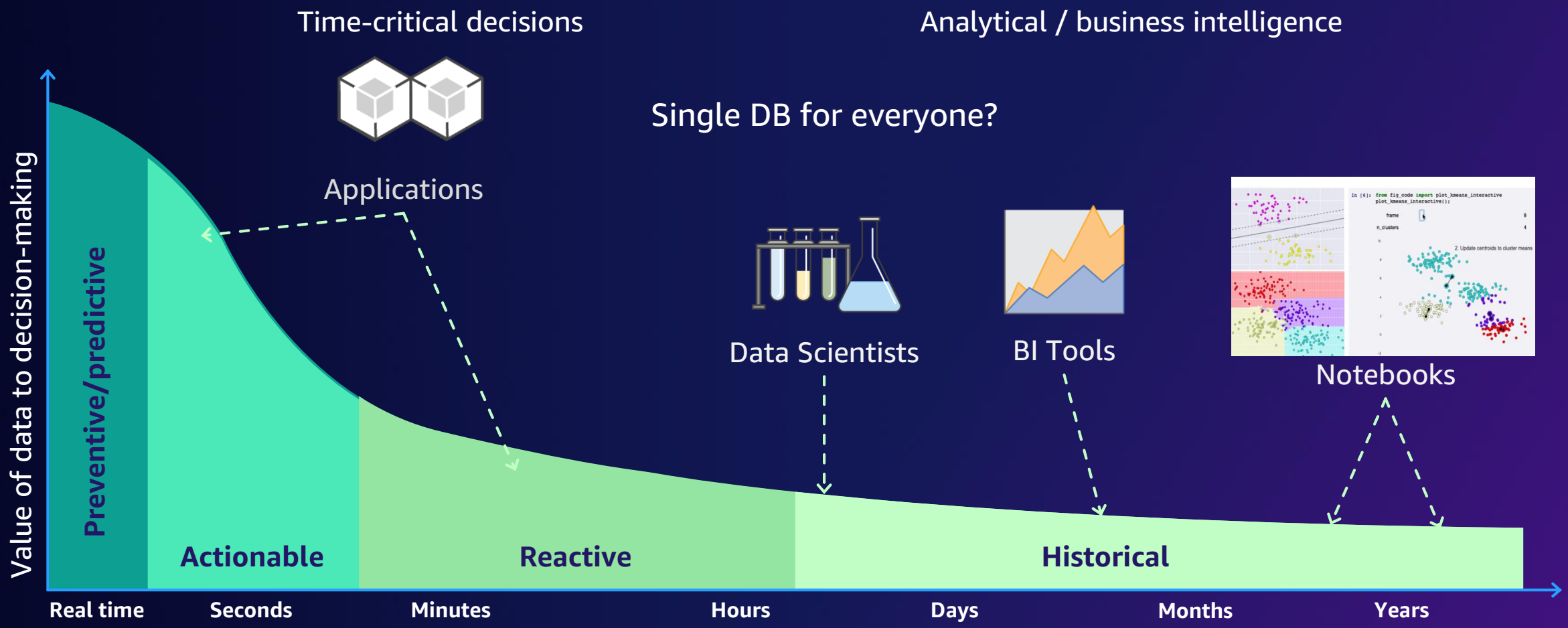
300+ sensors/car
50+ TB of data/week





Images source: aws.amazon.com/f1

Challenge: Data generated vs. Data available for analysis

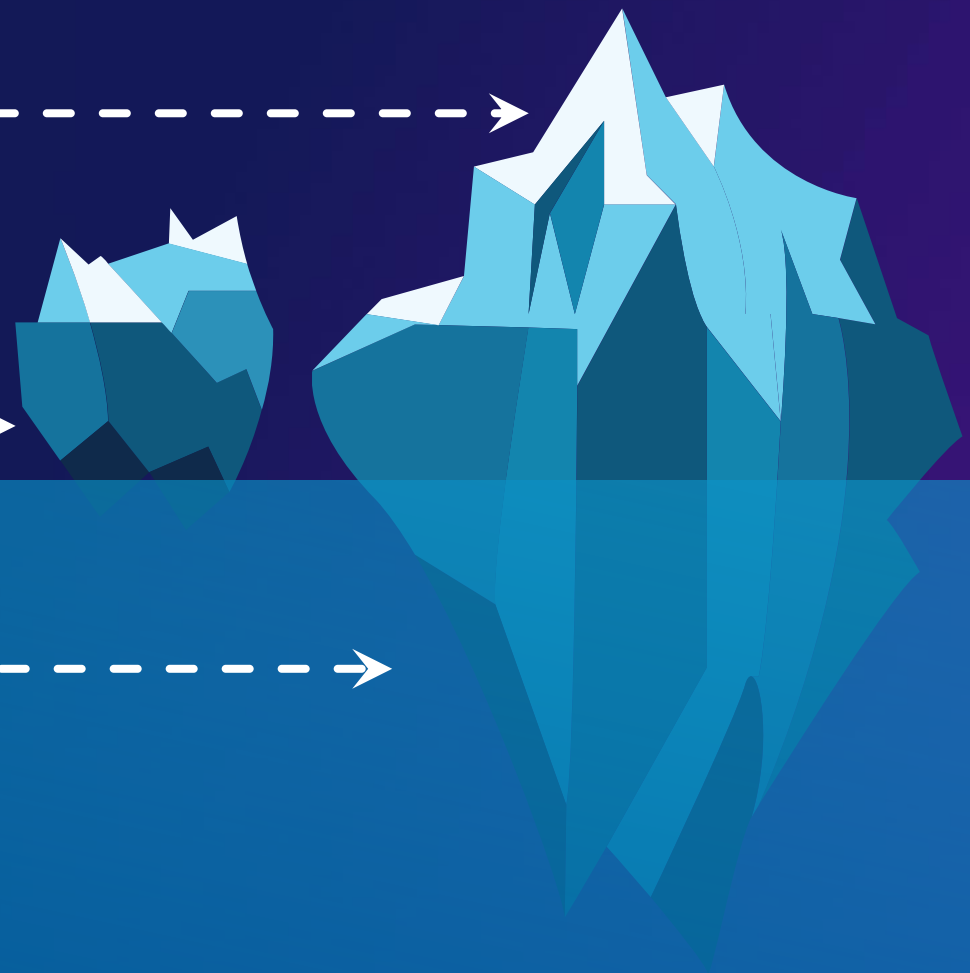


Graph visual: Perishable insights, Mike Gualtieri, Forrester

Challenges working with data

Data Warehouse

Operational Database



```
@ DELETE fact_hist_finance
    WHERE txn_date < ( now() - 5y )
```

Why Amazon Redshift for your data needs?

Why Amazon Redshift for your data needs?

Secure, Fast, Easy-to-implement and scale, cloud data warehouse

MPP-aware Query Optimizer | Columnar storage | Data Compression

Easy analytics for everyone

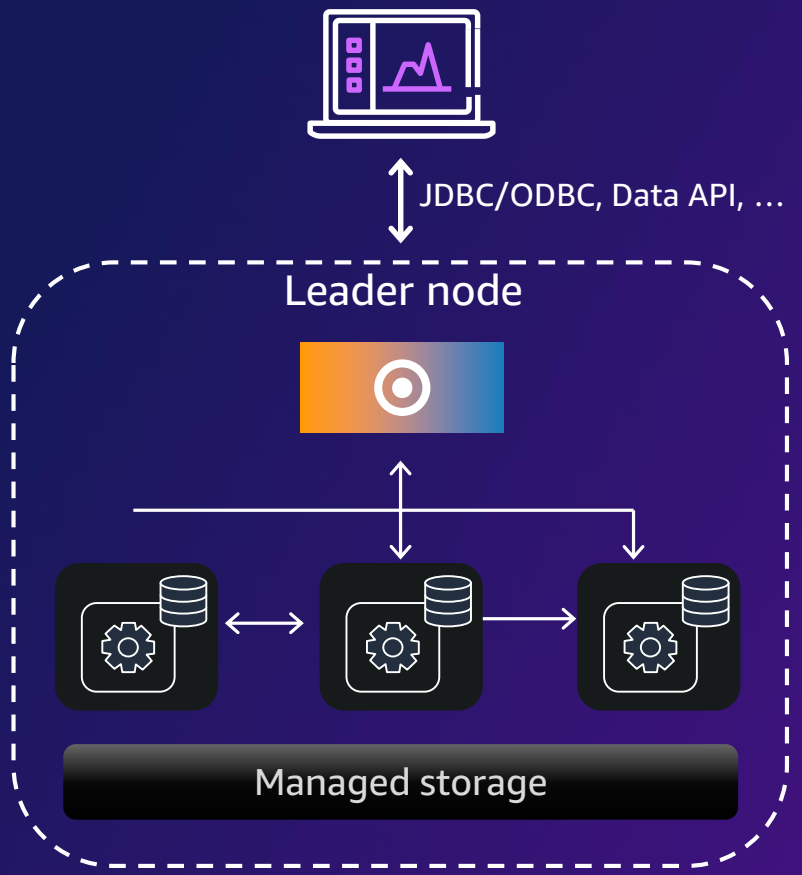
Insights in seconds, without managing your data warehouse

Analyze all your data

Remove **data silos** and run **real-time analytics**

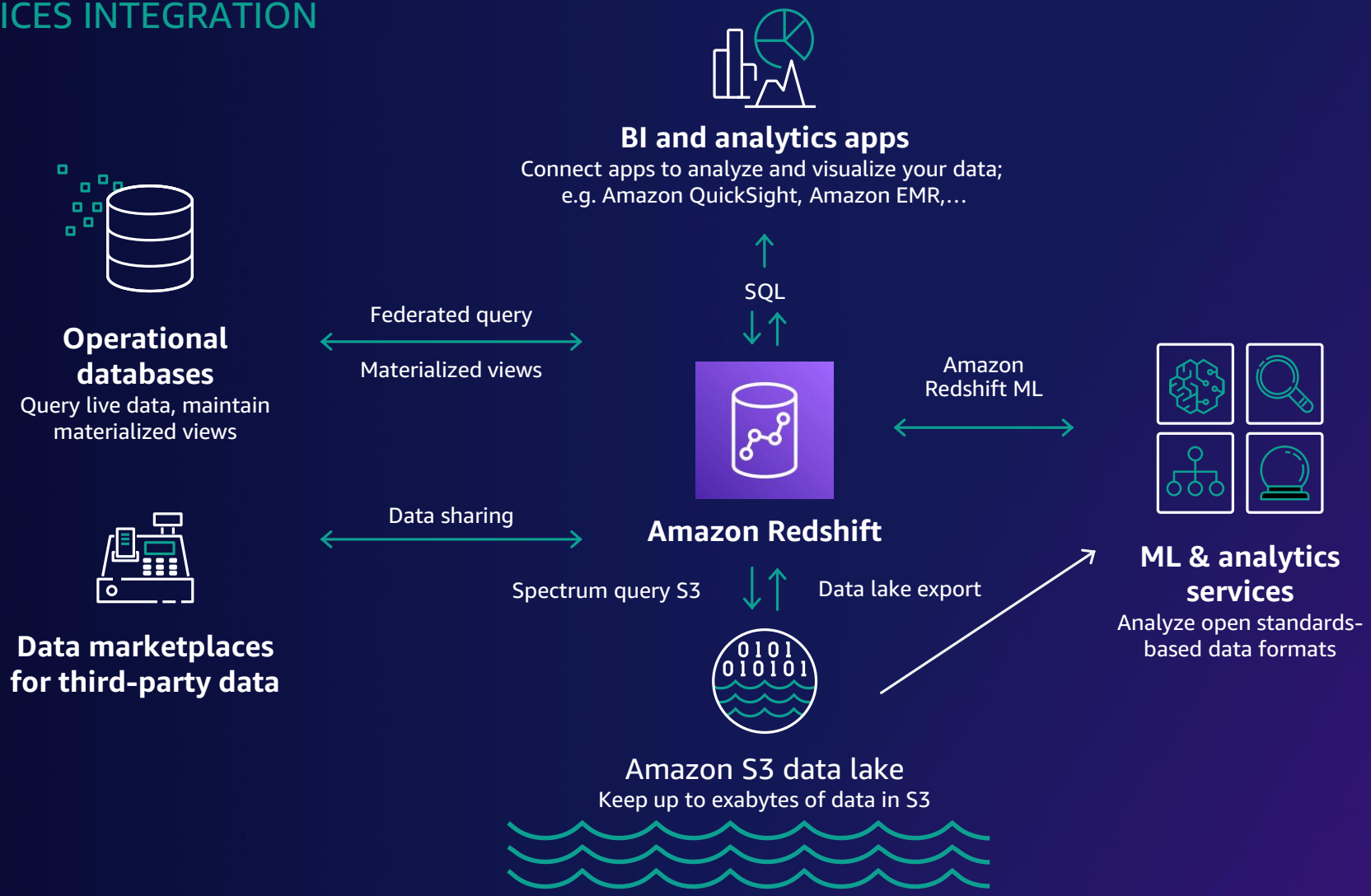
Performance at any scale

Dynamically scale



Analyze all your data

WITH AWS SERVICES INTEGRATION



RA3 nodes with managed storage

SCALE COMPUTE AND STORAGE INDEPENDENTLY

Managed storage

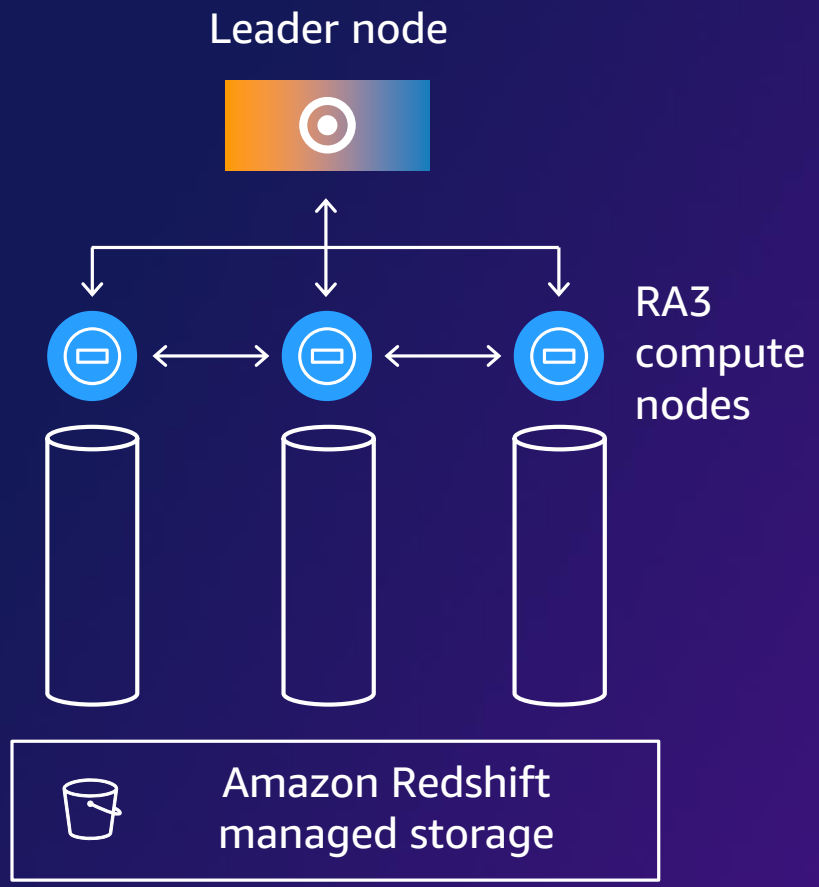
Large high-speed cache

High-bandwidth networking

Scale and pay independently for compute and storage

No need to manage storage

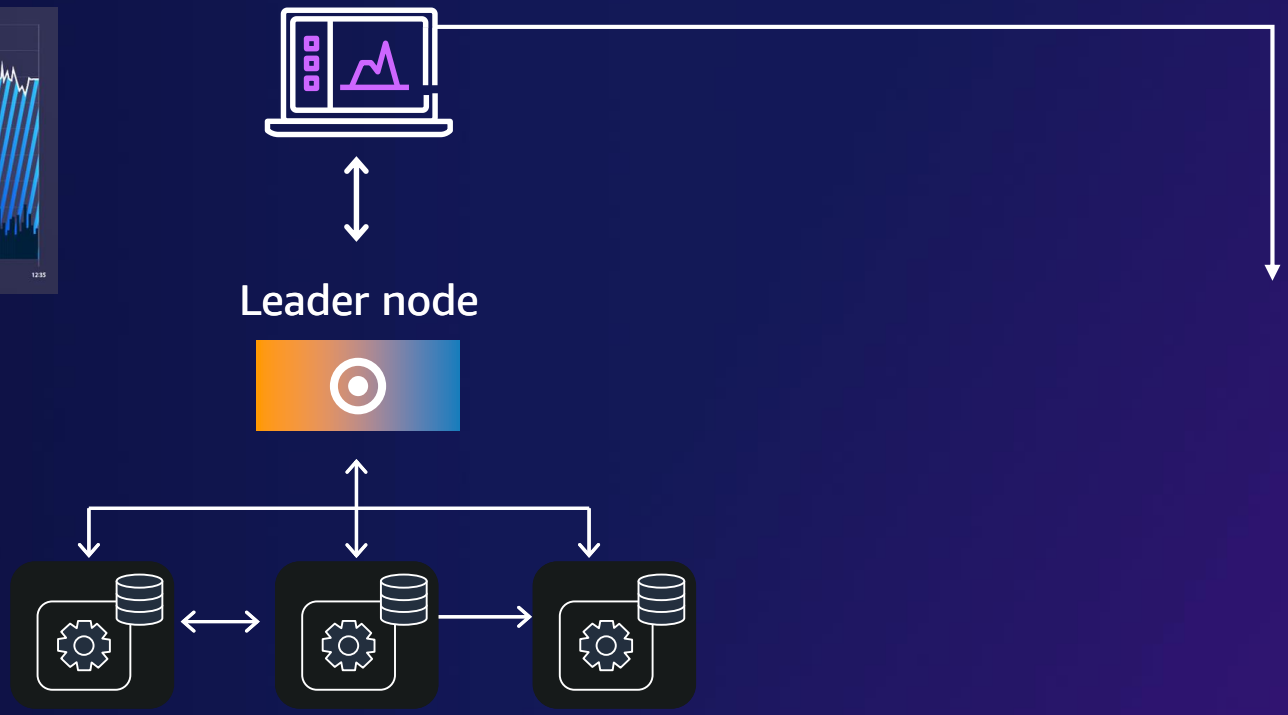
Available in small, medium, and large workloads



Performance at any scale

Amazon Redshift workload management (WLM)

WLM enables users to manage priorities within workloads so that short, fast-running queries won't get stuck in queues behind long-running queries.

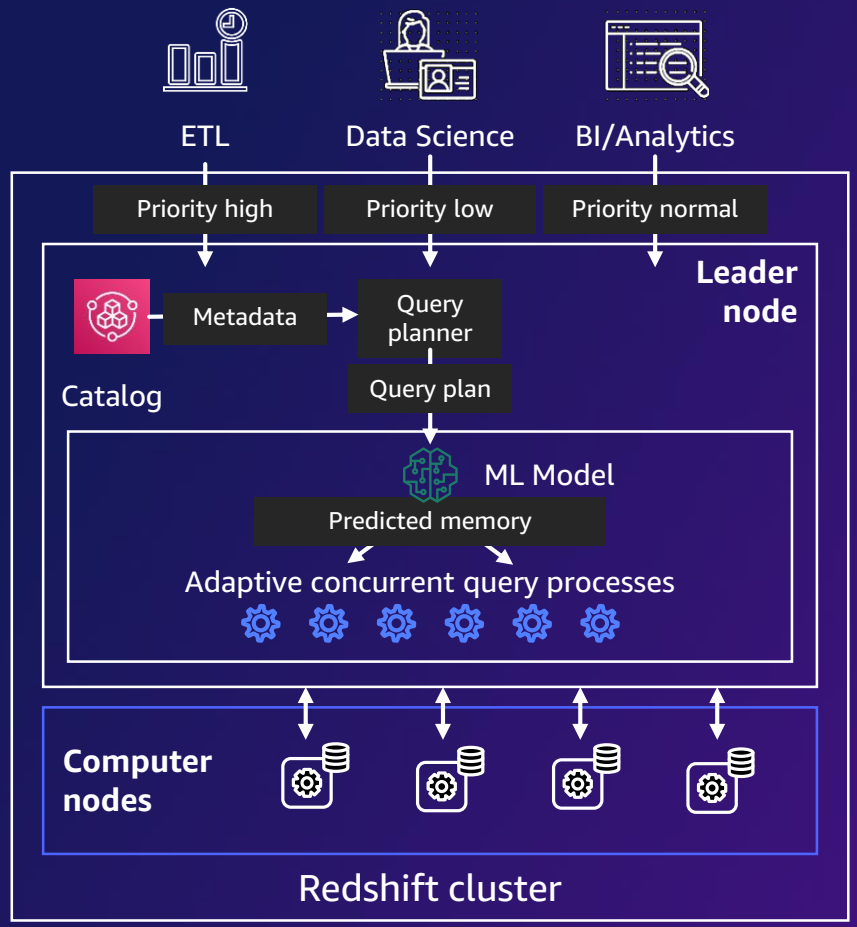


Enhanced Auto WLM – adaptive concurrency

Based on the query traffic and resource utilization, it determines the number of concurrent queries to optimize query throughput

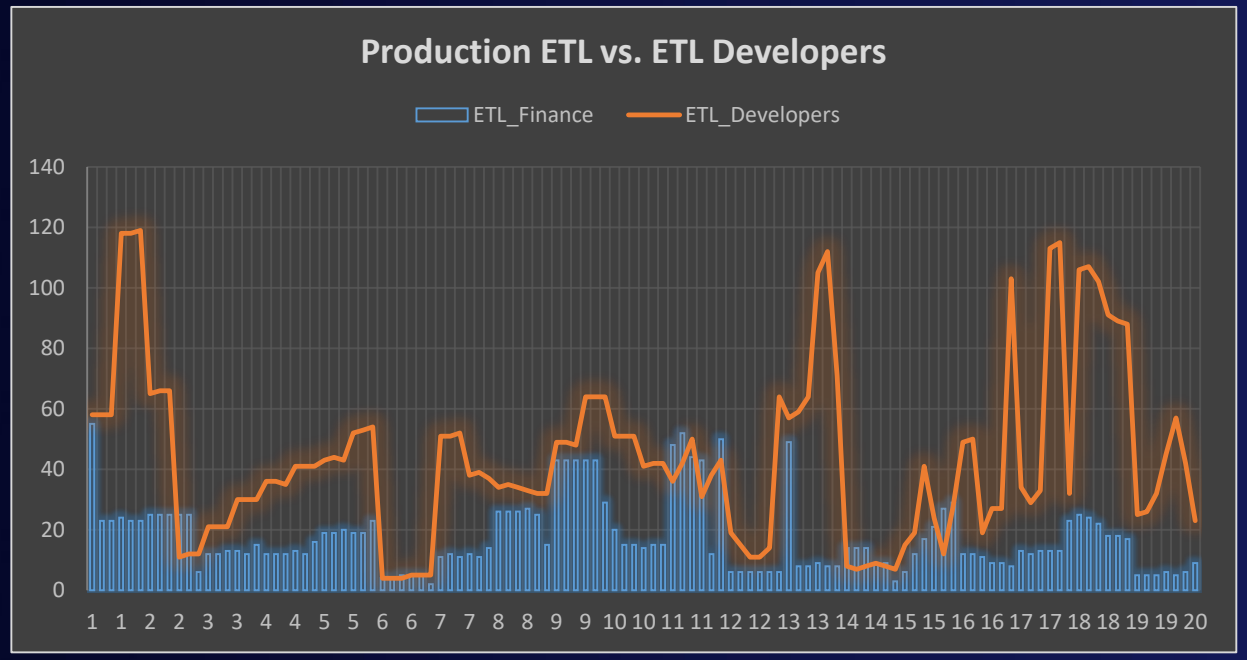
Auto WLM : create user or query group queues based on priority

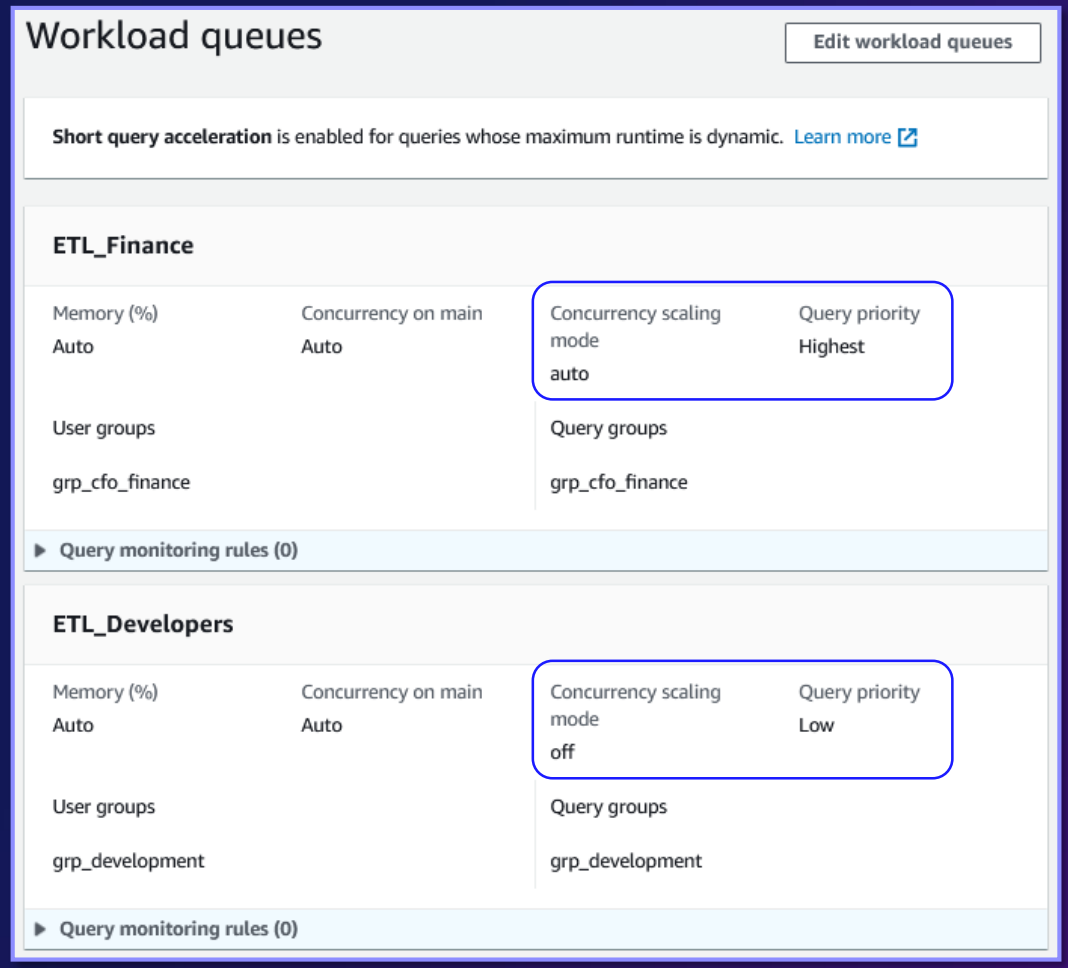
Concurrency Scaling based on Business Priority and Cost to ensure SLA



Enhanced Auto WLM – adaptive concurrency

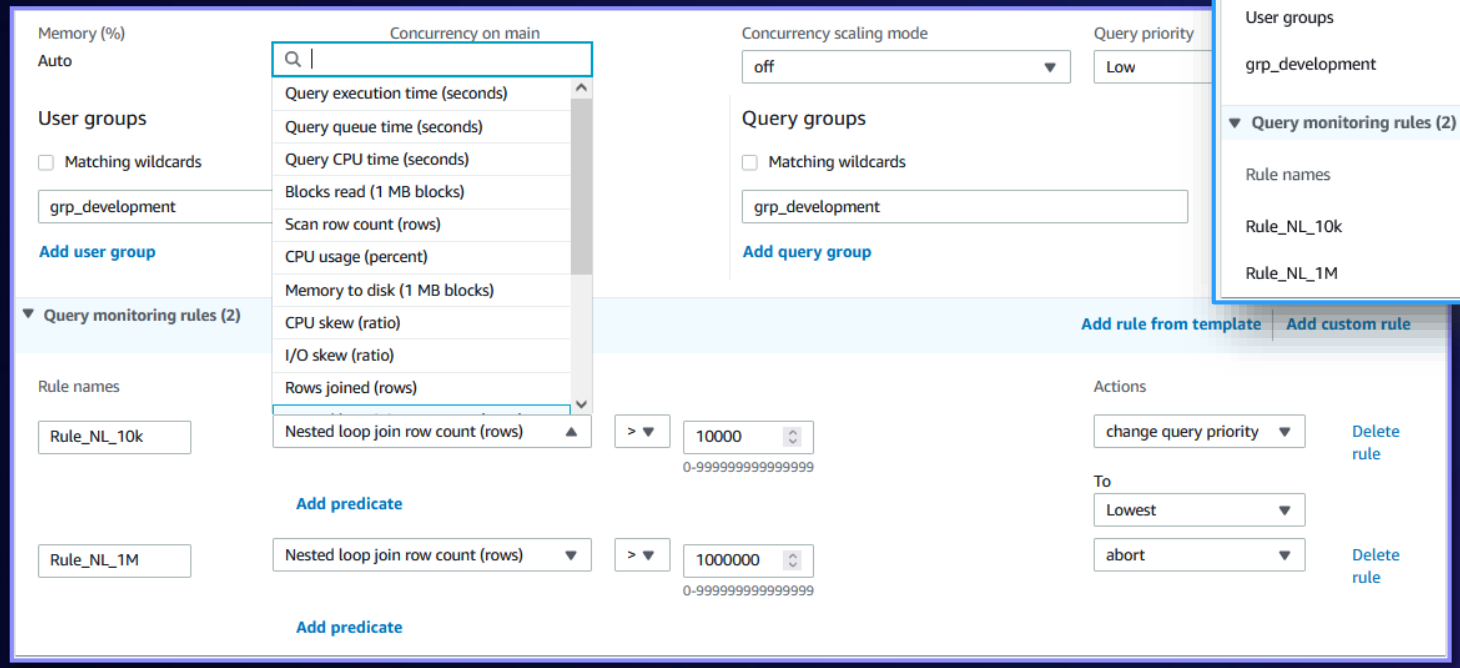
Real-world example, running a TPC benchmark to help a customer in defining the best **Cost vs. Performance balance**

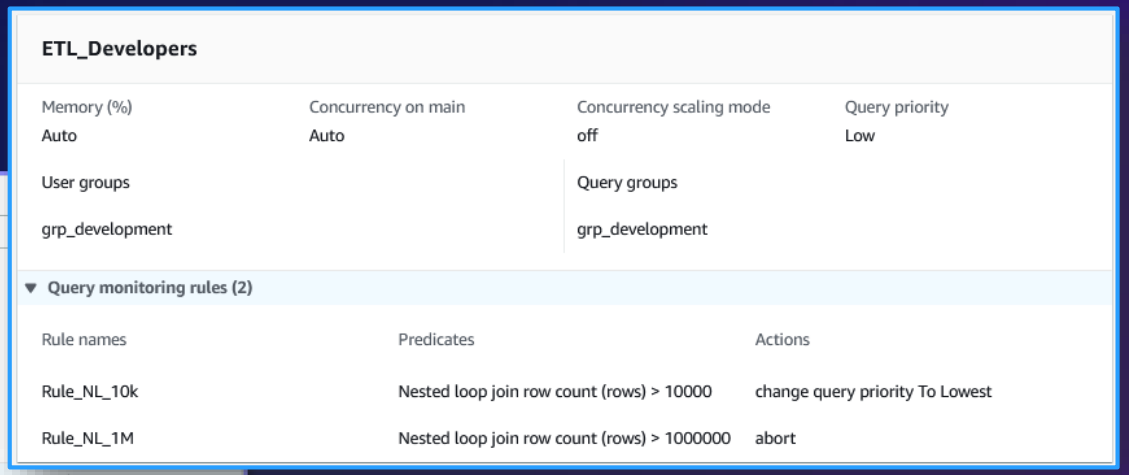




WLM – Using Query Monitoring Rules

Help your Developers to code their reports, to follow best practices





WLM – Using QMR / Query Monitoring Rules

POWER OF QMR THROUGH A SIMPLE EXAMPLE

Why should we detect Nested Loops?

Merge join will scan the data once

High I/O == Columnar vs. Row-based

```
SELECT loc.store_town_location, sum(s.txn_amount)
FROM fact_sales s
   INNER JOIN dim_store_locations loc
       ON (s.store_id = loc.store_id)
WHERE loc.store_city in ('Los Angeles')
GROUP BY loc.store_town_location;
```

snappify.io

Amazon Redshift Grafana Plugin

NEW

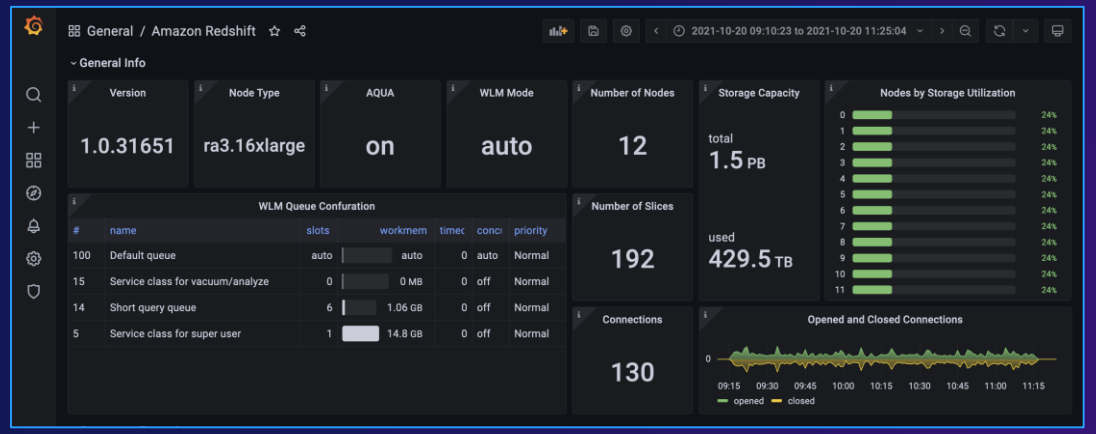
QUERY YOUR OPERATIONAL AND BUSINESS DATA DIRECTLY FROM GRAFANA

Available now in Amazon Managed Grafana and OSS Grafana

Visualize operational metrics and business insights from Redshift on your Grafana dashboards

Query System Views directly for in-depth system metrics beyond CloudWatch metrics

A default Amazon Redshift operational dashboard is available out-of-the-box



Redshift Spectrum: Partition Pruning

Example: Spectrum query to sum up data, for a specific application and date

- Table structure on S3 data lake

```
s3://fin_bucket/data/marketing/impressions/application=<application name>/date=<yyyymmdd>
```

- Query to execute

```
select sum(hits)
from marketing_impressions
where application = 'flux_capacitor'
  and date = to_char(current_date,'yyyymmdd')
```

- Outcome

  One prefix is read on S3 — less I/O — performance and cost

Recently released features

Automated Materialized Views

NEW PREVIEW

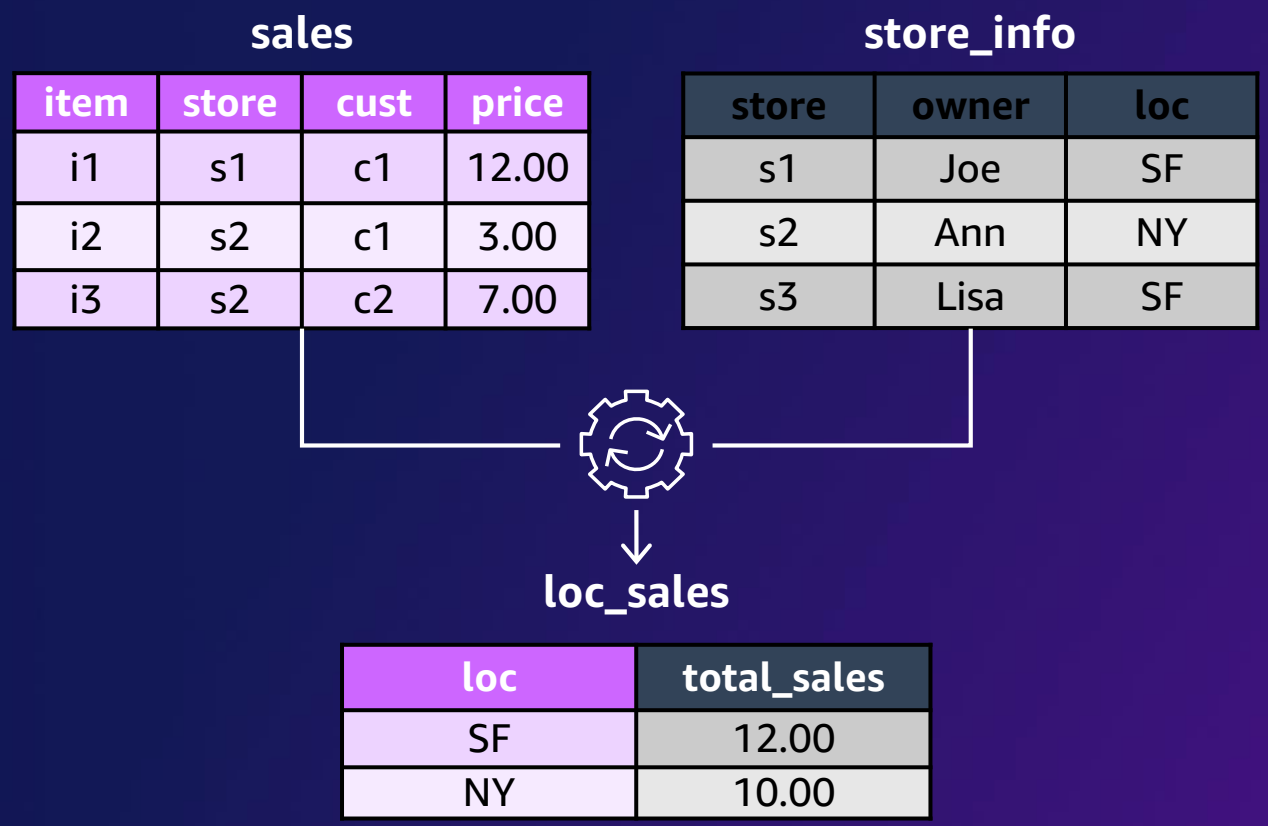
Automated creation and deletion of materialized views, with incremental refresh

Up to orders of magnitude query performance improvement

Continual monitoring to identify queries that will benefit from having a MV and automatically create

Automated query rewrite to leverage MV

Automatically deletes MVs that are no longer useful



Automated performance tuning

ML-BASED OPTIMIZATIONS TO GET STARTED EASILY AND GET THE FASTEST PERFORMANCE QUICKLY

Automatic vacuum delete

ATO: Automatic distribution keys

ATO: Automatic sort keys

Auto workload manager

Automatic table sort

ATO: Automatic column encoding

Auto Analyze

Auto refresh & re-write Materialized Views

Automates physical data design and optimization

Optimizes for peak performance as data and workloads scale

Leverages machine learning to adapt to shifting workloads

Amazon Redshift Data API

SIMPLIFIES DATA ACCESS FROM WEB SERVICES BASED APPLICATIONS

Simplifies data access from web-services. No configuring drivers or connection pools required!

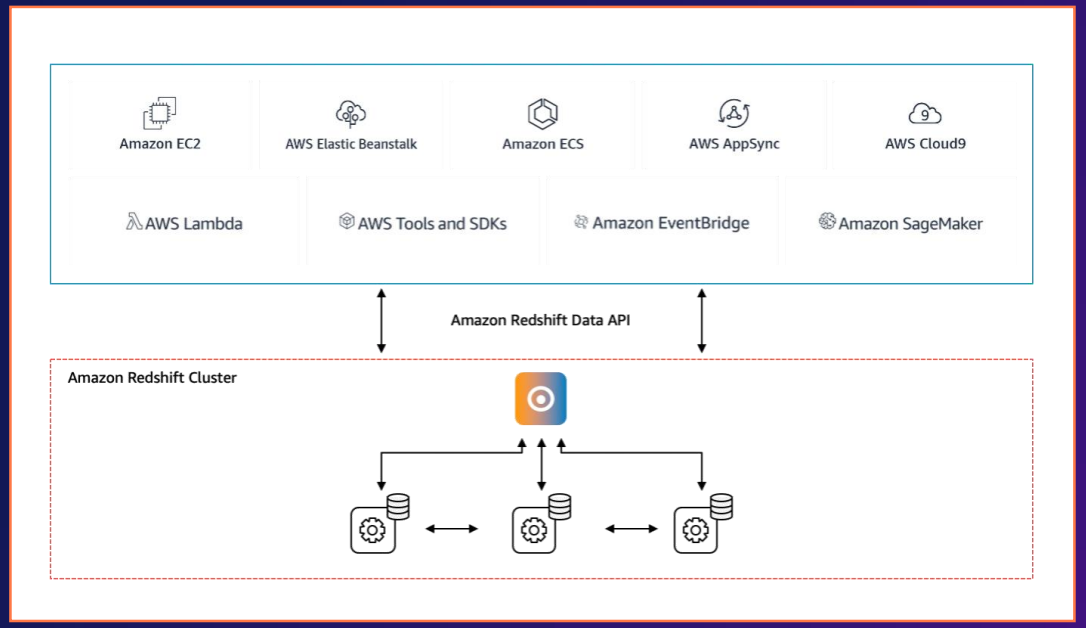
Build your ETL pipeline with Lambda, AWS step functions

Build event-driven apps with EventBridge and Lambda

Simpler access from DS tools; e.g. SageMaker and Jupyter notebooks

Schedule SQL scripts

```
aws redshift-data execute-statement
--database [DATABASE]
--query [QUERY]
--secret-arn [CREDENTIALS_ARN]
```



Amazon Redshift Data API

SIMPLIFIES DATA ACCESS FROM WEB SERVICES BASED APPLICATIONS

Or simply call the Data API **from your favorite SDK, Step functions, etc.**

```python
def run_query(rs_cluster, rs_db_name, db_user, sql_input, query_id, sync_exec=False, aws_region='eu-west-1'):
    """
    function to execute an SQL statement, against a Redshift cluster

    rs_cluster: str
    [...]
    """

    # AWS settings
    redshift_client = boto3.client('redshift-data', region_name=aws_region)

    try:
        if sql_input:
            # Run Redshift query; Secrets Manager is also compatible.
            redshift_response_run = redshift_client.execute_statement(
                ClusterIdentifier=rs_cluster,
                Database=rs_db_name,
                DbUser=db_user,
                Sql=sql_input,
                StatementName='SQL Query'
            )
```

snappify.io

Amazon Redshift and AWS Data Exchange (ADX) integration

NEW PREVIEW

Offer your Redshift data as data products, with pricing terms & conditions

Data subscribers can search and subscribe in AWS Data Exchange

Get 'live access' to the data in your Redshift cluster or data lake. No ETL or data ingestion

Welcome to Serverless!

Amazon Redshift Serverless

Automatically and intelligently provisions and scales data warehouse capacity

All of Amazon Redshift's benefits with an easy get started experience

NEW PREVIEW

No compromises

SQL features | Performance at scale | Scalability | Analyze all your data | Security

Easy to get started

Developer, business analyst

Enable Amazon Redshift serverless for your AWS account

BI tool

Connect from your favorite BI tool or Amazon Redshift Query Editor

Amazon Redshift

Amazon Redshift Serverless executes queries by automatically provisioning capacity

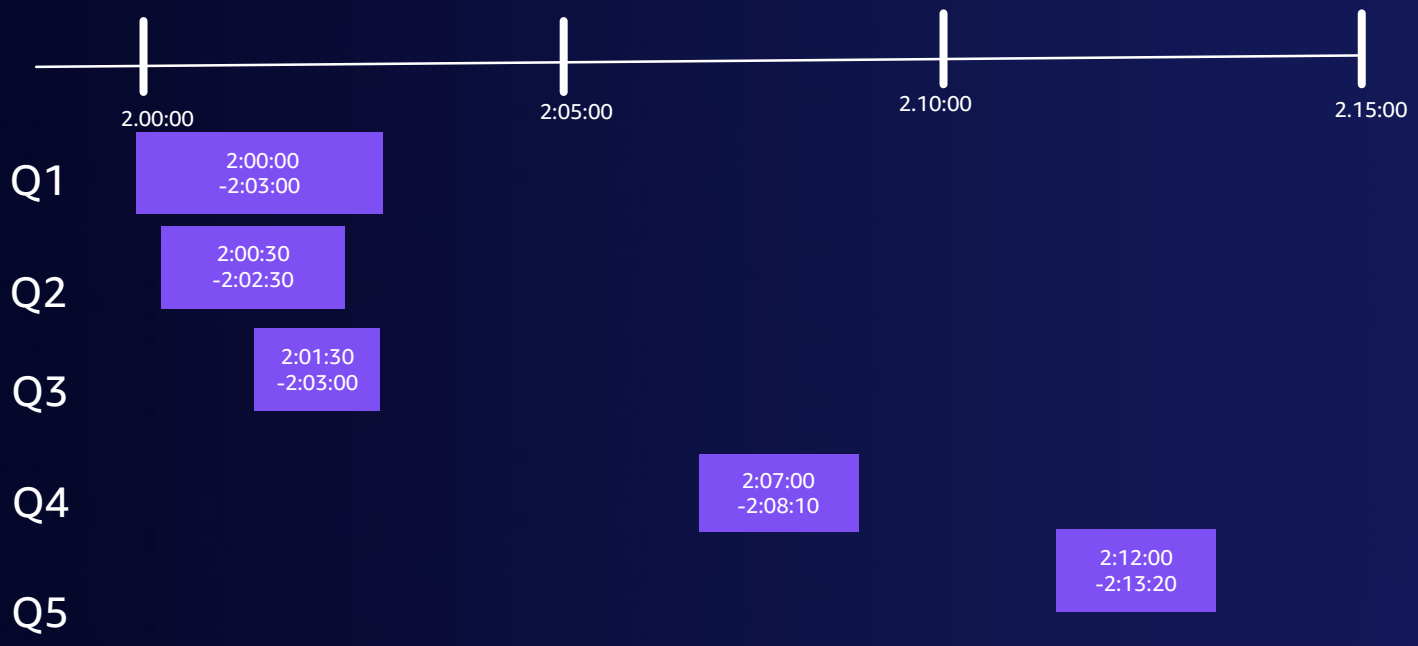
Pay

Pay for compute and storage used during analysis

Pay for use

NEW PREVIEW

Pay for the compute capacity **only** for the workload duration (metered on a per-second basis)

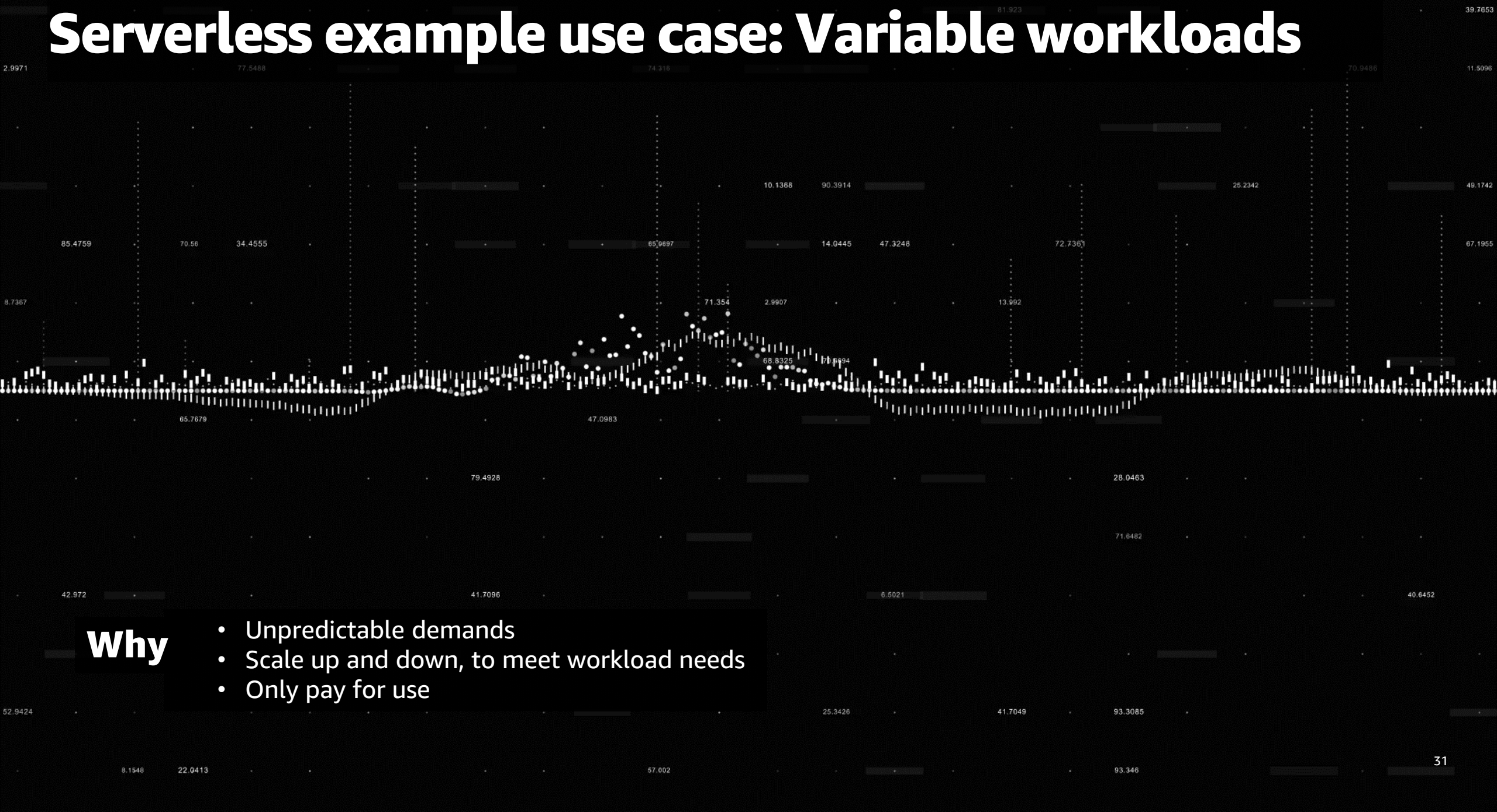


Billing duration	Query execution time
@2:03	3 minutes (for Q1, Q2, Q3)
@2:09	1 minute 10 seconds (for Q4)
@2:14	1 minute and 20 seconds (for Q5)
Total charges	5 minutes and 30 seconds

No charges for idleness; i.e.

When:
*** No ETL activity**
*** No more paying for Dev clusters at night**

Serverless example use case: Variable workloads



Why

- Unpredictable demands
- Scale up and down, to meet workload needs
- Only pay for use

Amazon Redshift Data Sharing

Data sharing

A SECURE AND EASY WAY TO SHARE DATA ACROSS AMAZON REDSHIFT CLUSTERS

Instant access without data copies/movement

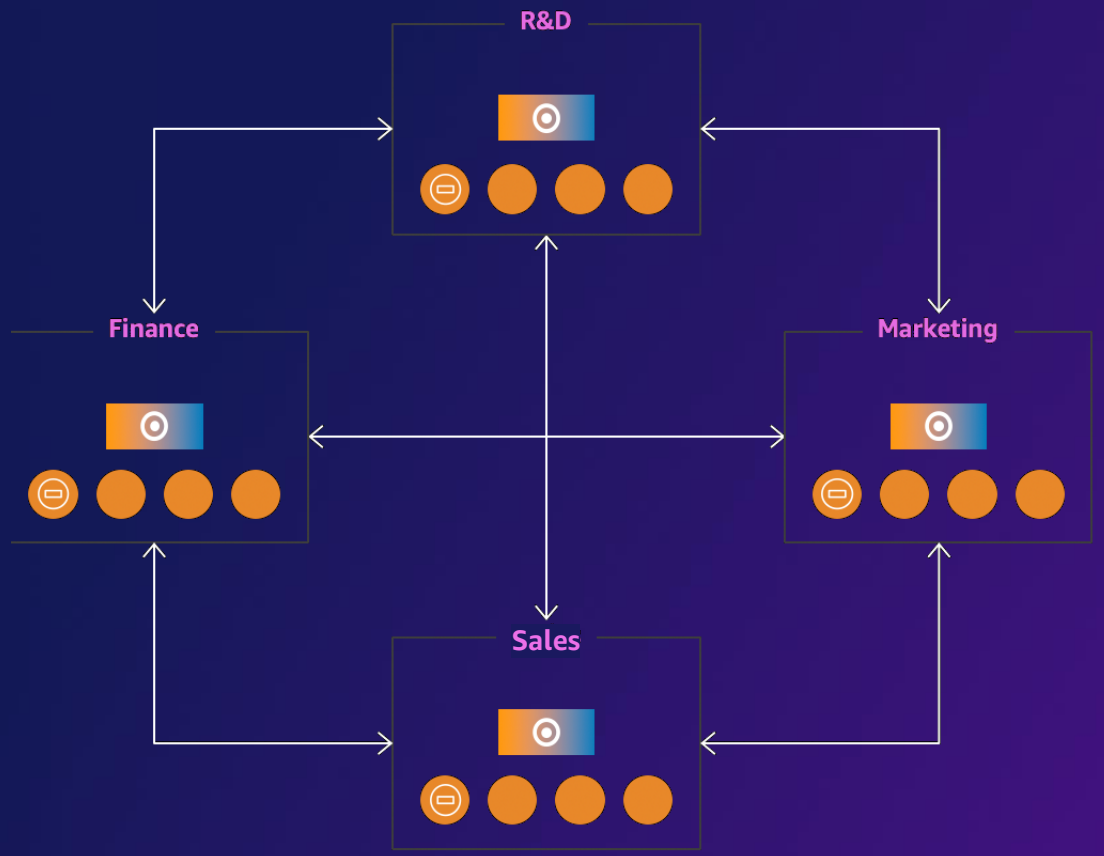
Live data, even when Producer is paused

Secure and governed collaboration

Isolated workloads

Cross account & Cross Region data sharing

Result Caching & Concurrency scaling for consumers



Data sharing

EXAMPLE: SEGREGATION OF WORKLOADS & PII DATA

1. Create second database and Share it;
   i.e. all schema or only with Tables and/or Views and excluding PII

Producer

Consumer

2. Create database and grant user groups

3. Refresh MViews every 5 minutes
   (Data from views will be immediately available)

4. Use data without impacting Production;
   e.g. additional complex aggregations

Producer - Finance

```
create database tenant1_silodb;
create database tenant2_silodb;
```

snappify.io

Producer - Finance

```
CREATE DATASHARE tenant1_silodbshare;
ALTER DATASHARE tenant1_silodbshare ADD SCHEMA tenant1_siloschema;
ALTER DATASHARE tenant1_silodbshare ADD ALL TABLES IN SCHEMA tenant1_siloschema;
```

snappify.io

Consumer - Marketing

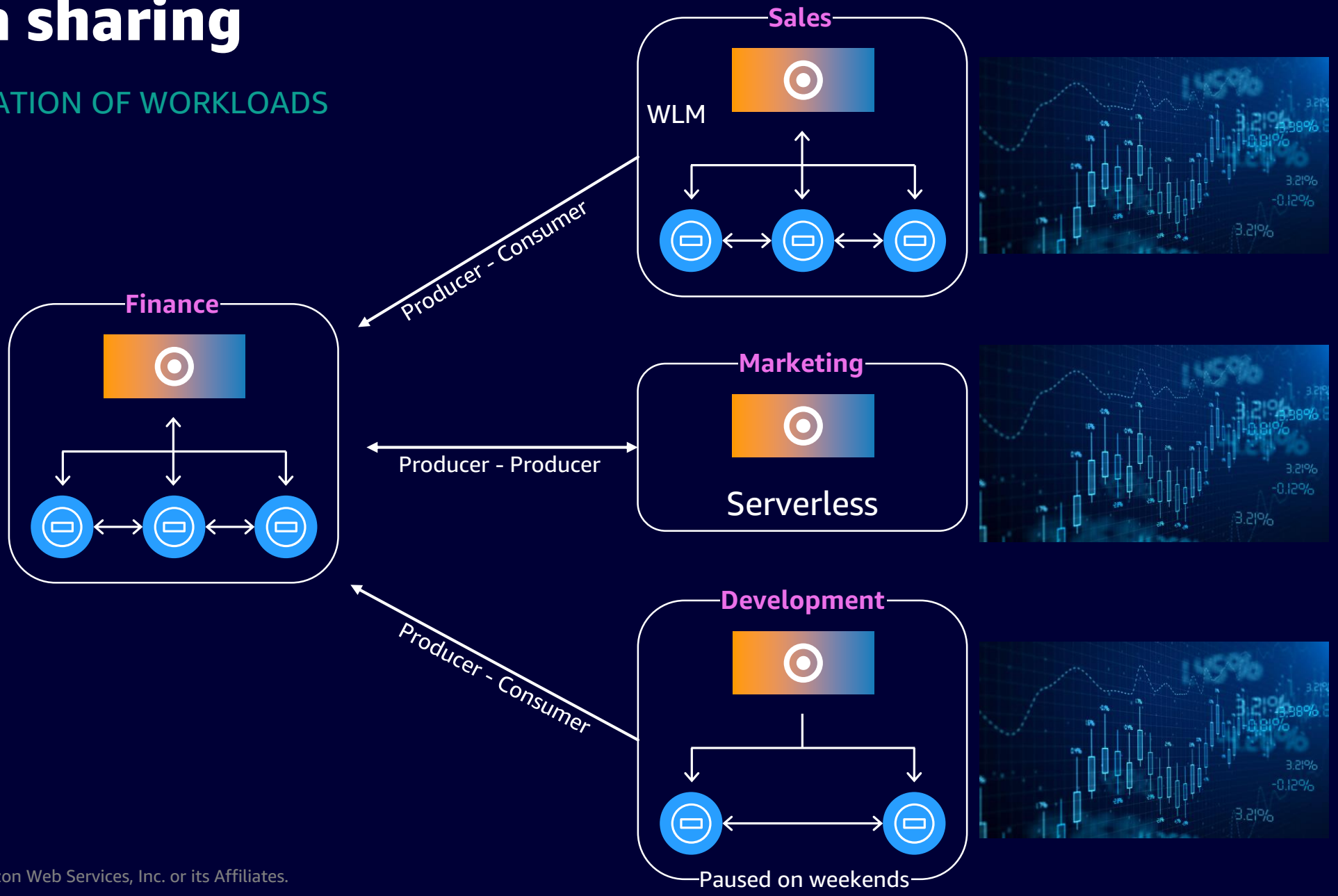
```
CREATE DATABASE tenant1_silodb FROM DATASHARE tenant1_silodbshare
OF NAMESPACE '<producercluster_namespace>';

select * from tenant1_silodb.tenant1_siloschema.customer;
```

snappify.io

Data sharing

SEGREGATION OF WORKLOADS



Rail Delivery Group

National Rail

Rail Delivery Group

Toby Ayre

Rail Delivery Group | Head of Data & Analytics

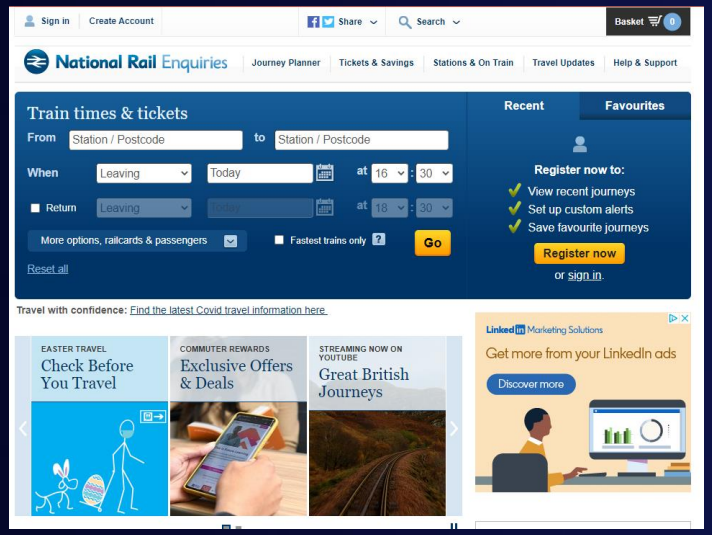
Rail Delivery Group

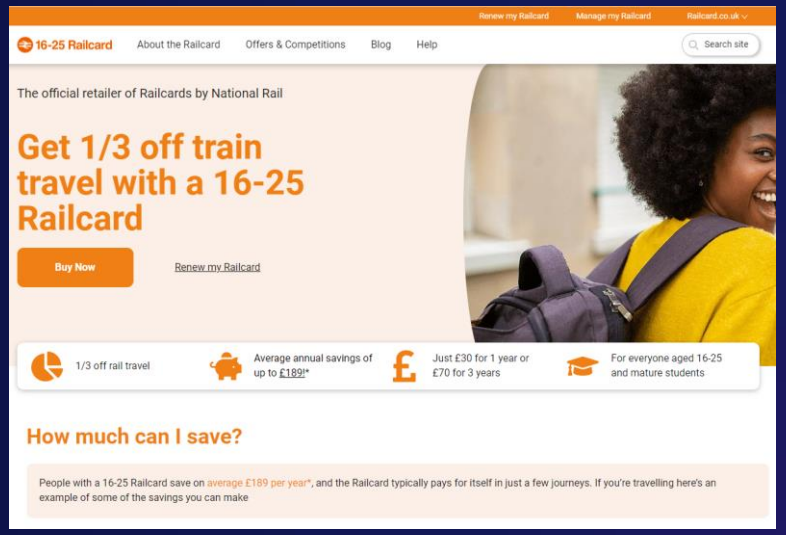
Rail Delivery Group
National Rail

National Rail

Railcard

LET'S GET BACK ON TRACK



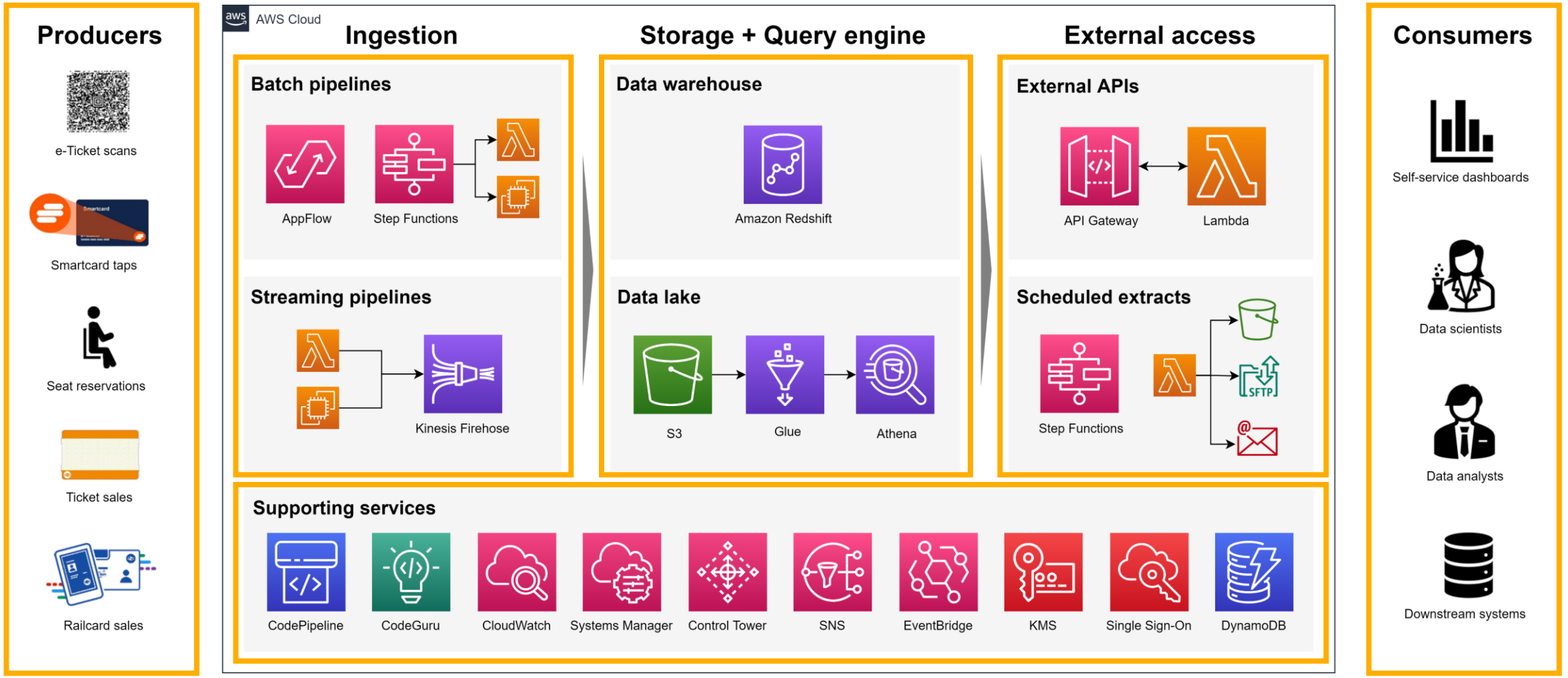


We bring together the companies which run Britain's railways

Data Platform

Rail Delivery Group

National Rail



Example Outputs

Rail Delivery Group

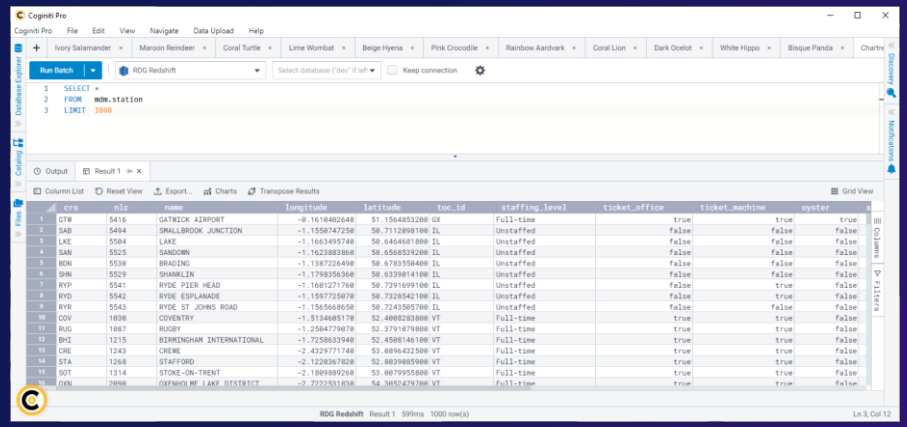
National Rail

1. File-based extracts

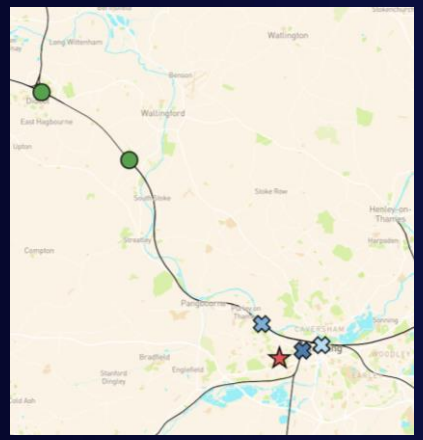
Name	Type	Last modified	Size	Storage class
services-AW2021-09-26.csv.gz	gz	September 27, 2021, 02:13:15 (UTC+01:00)	6.6 MB	Standard
services-AW2021-09-27.csv.gz	gz	September 28, 2021, 02:13:36 (UTC+01:00)	6.6 MB	Standard
services-AW2021-09-28.csv.gz	gz	September 29, 2021, 02:13:03 (UTC+01:00)	6.5 MB	Standard
services-AW2021-09-29.csv.gz	gz	September 30, 2021, 02:13:24 (UTC+01:00)	6.5 MB	Standard
services-AW2021-09-30.csv.gz	gz	October 1, 2021, 02:12:58 (UTC+01:00)	6.5 MB	Standard

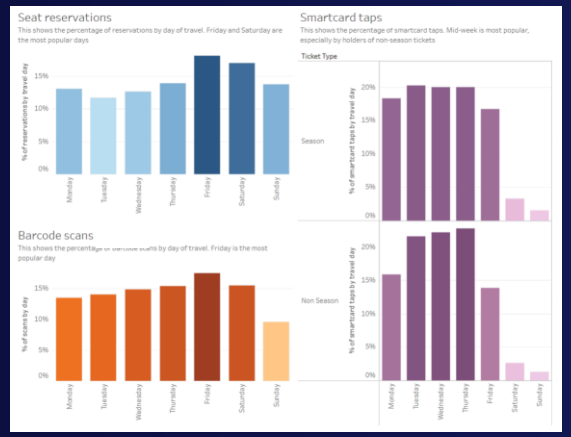
service_name	service_date	service_type	service_property_1	service_property_2	service_property_3	service_property_4	departure_station	departure_time	arrival_station	arrival_time
AW653300	28/09/2021	TRAIN	TC_S	TL_S	PC_25444001	PT_DMU	EBB	28/09/2021 10:38	CDF	28/09/2021 1
AW205600	28/09/2021	TRAIN	TC_S	PC_25429000	PT_DMU	TL_E	GCR	28/09/2021 20:00	MST	28/09/2021 2
AW654400	28/09/2021	TRAIN	TC_S	PC_25444001	PT_DMU	TL_A	EBB	28/09/2021 22:38	CDF	28/09/2021 2
AW679700	28/09/2021	TRAIN	TC_S	PC_22342000	TL_S	PT_DMU	WRX	28/09/2021 06:32	BID	28/09/2021 0
AW655500	28/09/2021	TRAIN	TC_S	PC_22338000	TL_S	PT_DMU	LLD	28/09/2021 08:02	LLI	28/09/2021 0
AW401400	28/09/2021	TRAIN	TC_S	PC_22345000	PT_DMU	TL_E	MIA	28/09/2021 11:35	LLD	28/09/2021 1
AW305700	28/09/2021	TRAIN	TC_S	PC_22340000	PT_DMU	TL_E	HHD	28/09/2021 11:48	BHI	28/09/2021 1
AW402500	28/09/2021	TRAIN	TC_S	PC_22340000	PT_DMU	TL_E	LLD	28/09/2021 14:40	MIA	28/09/2021 1
AW960200	28/09/2021	TRAIN	TC_S	PC_25441000	PT_DMU	TL_A	ABA	28/09/2021 08:21	BYI	28/09/2021 0
AW951500	28/09/2021	TRAIN	TC_S	PC_25447000	PT_DMU	TL_A	CDF	28/09/2021 08:05	TRB	28/09/2021 0
AW963600	28/09/2021	TRAIN	TC_S	PC_25447000	PT_DMU	TL_A	RDR	28/09/2021 23:14	BYI	28/09/2021 2
AW940500	28/09/2021	TRAIN	TC_S	PC_25443000	PT_DMU	TL_A	PEN	28/09/2021 07:18	RHY	28/09/2021 0
AW952600	28/09/2021	TRAIN	TC_S	PC_25447000	PT_DMU	TL_A	CDF	28/09/2021 14:05	TRB	28/09/2021 1
AW941600	28/09/2021	TRAIN	TC_S	PC_25443000	PT_DMU	TL_A	PEN	28/09/2021 17:18	RHY	28/09/2021 1
AW953700	28/09/2021	TRAIN	TC_S	PC_25447000	PT_DMU	TL_A	CDF	28/09/2021 19:05	TRB	28/09/2021 2

2. JDBC access for ad-hoc analysis



3. JDBC access for interactive dashboards





4. APIs

```
{
  "service": "XC112000",
  "travel_date": "2020-03-04"
}
```

Amazon Redshift

Rail Delivery Group

National Rail

2 x
RA3.4Xlarge nodes

85 billion
records in the largest table

5+ years
since launched at Rail Delivery Group

45,000
queries per day on average

300+
Tableau dashboards connected

700+
tables and views

Rail Delivery Group

National Rail

Demo

Amazon Redshift | Data Sharing

Rail Delivery Group

Amazon Redshift provisioned clusters

Redshift serverless New

Provisioned clusters dashboard

Clusters
- Reserved nodes
- Snapshots

Query editor
Query editor v2
Queries and loads

Datashares

Configurations

Advisor
AWS Marketplace
Alarms

Query data using Redshift query editor

Use the query editor v2 to run queries in your Redshift cluster.

Query data

Work with your client tools

You can connect to Amazon Redshift from your client tools, such as SQL clients, business intelligence (BI) tools, and extract, transform, load (ETL) tools, using JDBC or ODBC drivers.

Cluster

Cluster identifier

Copy JDBC URL

Copy ODBC URL

Choose your JDBC or ODBC driver

Use JDBC or ODBC drivers to connect to Amazon Redshift from your client tools, such as SQL clients, BI tools, and ETL tools. We recommend using the new Amazon Redshift-specific drivers for better performance and scalability.

Driver

JDBC 4.2 without AWS SDK (.jar)

Download driver

Clusters (2) Info

Query data | Actions | Create cluster

Filter clusters by property or value

1

Cluster	Cluster namespace	Status	Storage capacity us...	CPU utilization	Snapshots
redshift-cluster-1 ra3.xlplus | 1 node | 4 TB	adcc6cd7-1d15-4758-...	Available	< 1%	4%	3 snapshots
redshift-cluster-2 ra3.xlplus | 1 node | 4 TB	98689f23-808f-4902-...	Available	< 1%	4%	1 snapshot

Thank you!

Carlos Contreras

AWS – Big Data & Analytics
EMEA Prototyping Labs

linkedin.com/in/carloscontreras

Toby Ayre

Rail Delivery Group
Head of Data & Analytics

linkedin.com/in/tobyayre/

Please complete the session survey