AWS re:Invent

ANT 320R

What's new with Amazon Redshift featuring Yelp

Himanshu Raja
Sr. Product Manager
Amazon Redshift

Steven Moy
Software Engineer
Yelp

Tony Gibbs
Solutions Architect
Amazon Web Services

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

7 years ago …



AWS re:Invent 2012

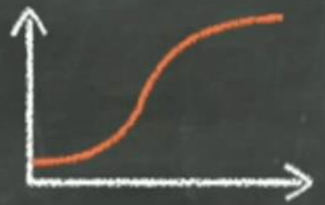
Andy Jassy
Senior Vice President, Amazon Web Services

What Would Data Warehousing Look Like If We Did It the AWS Way

Easy to provision and scale up massively



No upfront costs, pay as you go



Really fast performance at a really low price



Open & flexible with support for popular tools





AWS re:Invent 2012

Andy Jassy
Senior Vice President, Amazon Web Services

Introducing Amazon Redshift
Data Warehousing the AWS Way



- Easily and rapidly analyze petabytes of data
- 1/10 the cost of traditional data warehouses
- Automated deployment & administration
- Compatible with popular BI tools

7 years later:
Tens of thousands of customers use Redshift



Amazon Redshift

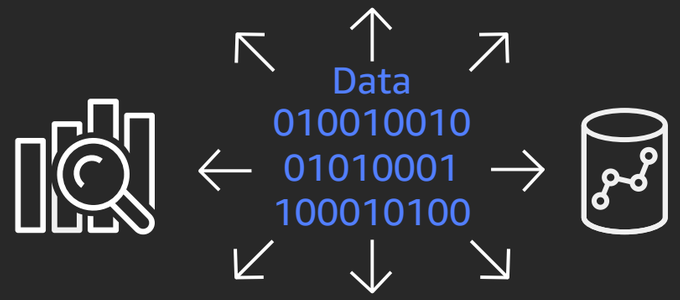
is the most popular cloud data warehouse

21st Century Fox · 23andMe · amazon · AMGEN · ancestry · AstraZeneca
CIVIS · coinbase · coursera · Dow Jones · edmunds · EA
EQUINOX · FT Financial Times · FOURSQUARE · HEARST · iflix · intuit
Jack in the box · Kimberly-Clark · Klarna · Liberty Mutual · London Stock Exchange · MANTHAN
MERCK · Nasdaq · The Ohio State University · OPTUM · PADDYPOWER betfair
Pfizer · Prezi · QANTAS · REDFIN · robinhood
SCHOLASTIC · SCP HEALTH · SCOPELY · SOUNDCLOUD · Sysco · tinder · yelp

Our customers are asking for …



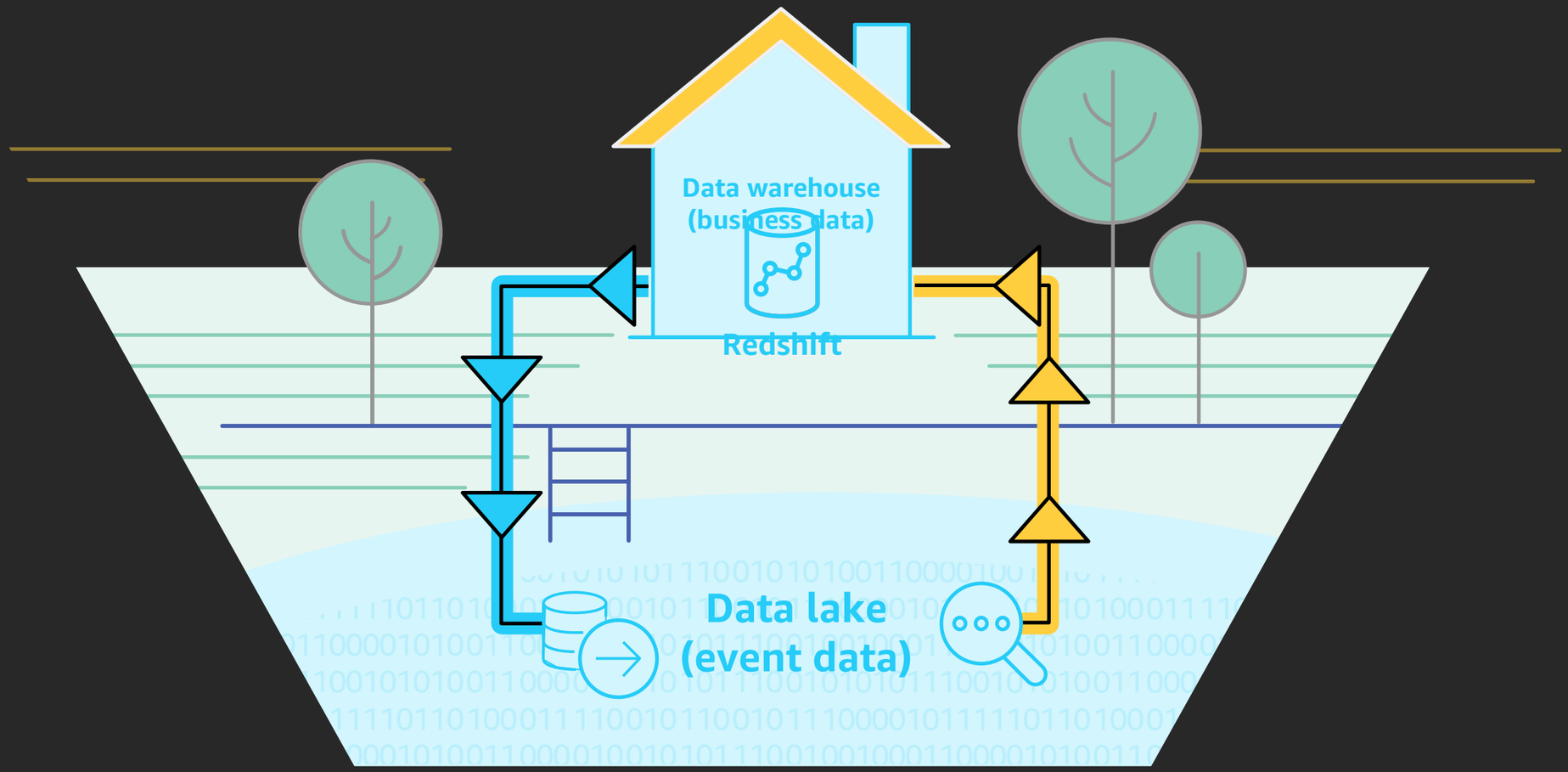
Migrations to the cloud



Exponential growth of event data



End-to-end insights from analyzing all your data



Customers moving to data lake architectures

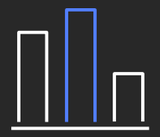
Redshift enables you to have a lake house approach

Amazon Redshift: Data warehouse of choice for



Data lake and AWS integrated

Lake Formation catalog and security, exabyte scale query (spectrum & federated), AWS integrated (DMS, CloudWatch)



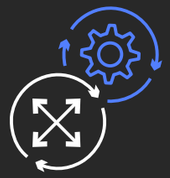
Best performance

Up to 3x faster than other cloud data warehouses



Lowest cost

Up to 75% less than other cloud data warehouses and predictable costs



Most scalable

Virtually unlimited elastic linear scaling



Most secure and compliant

AWS-grade security (e.g., VPC, encryption with KMS, Cloud Trail); certifications such as SOC, PCI, DSS, ISO, FedRAMP, HIPAA



Fully managed

Easy to provision and manage, automated backups, AWS support, 99.9% SLAs

Amazon Redshift has been innovating quickly

Robust result set caching

Large # of tables support ~20000

Copy command support for ORC, Parquet

IAM role chaining

Elastic resize

Groups

Redshift Spectrum: date formats, scalar json and ION file formats support, region expansion, predicate filtering

Auto analyze

Health and performance monitoring with Amazon Cloud watch

Automatic table distribution style

Cloud watch support for WLM queues

Performance enhancements — hash join, vacuum, window functions, resize ops, aggregations, console, union all, efficient compile code cache

Unload to CSV

Auto WLM

~25 Query Monitoring Rules (QMR) support

AQUA

Concurrency Scaling

200+

new features and enhancements in the past 18 months

Manage multi-part query in AWS console

Auto analyze for incremental changes on table

DC1 migration to DC2

Resiliency of ROLLBACK processing

Spectrum Request Accelerator

Apply new distribution key

Redshift Spectrum: Row group filtering in Parquet and ORC, nested data support, enhanced VPC Routing, multiple partitions

Faster Classic resize with optimized data transfer protocol

Performance: Bloom filters in joins, complex queries that create internal table, communication layer

Redshift Spectrum: Concurrency scaling

Amazon Lake Formation integration

Auto-Vacuum sort, Auto-Analyze and Auto Table Sort

Auto WLM with query priorities

Snapshot scheduler

Performance: join pushdowns to subquery, mixed workloads temporary tables, rank functions, null handling in join, single row insert

Advisor recommendations for distribution keys

AZ64 compression encoding

Console redesign

Stored procedures

Spatial processing

Column level access control with AWS lake formation

RA3

Performance of Inter-Region Snapshot Transfers

Federated Query

Materialized Views

Manual pause and resume

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

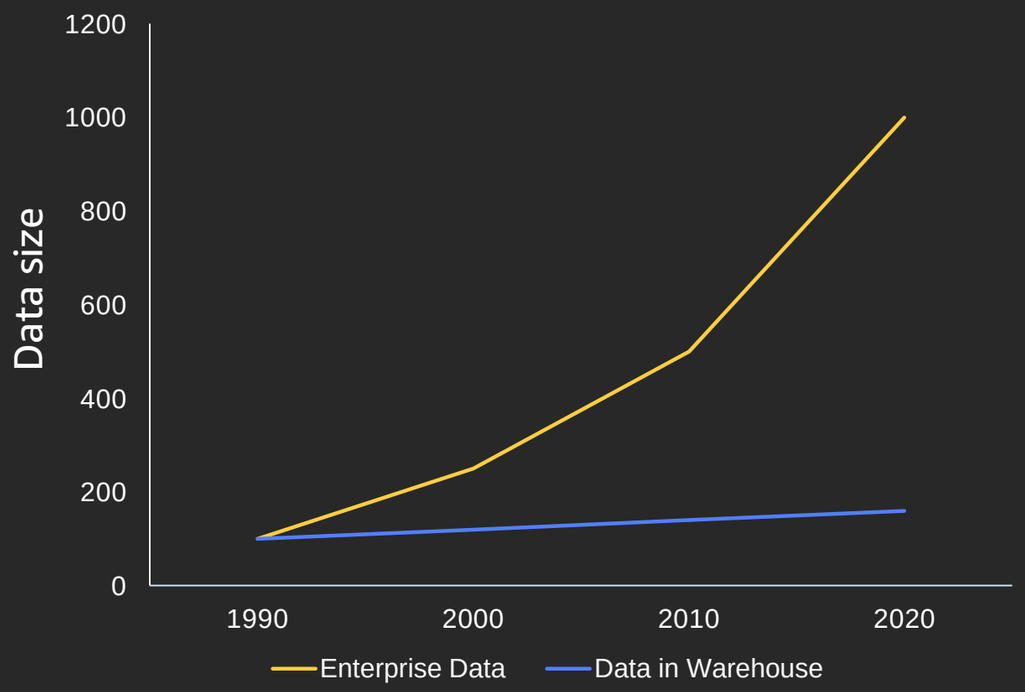
NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Exponential data growth causing management, performance and scaling challenges

What about data growth?



Compute and storage needs can vary independently

Compute scaling:

- To handle increased query load during peak times
- To ingest data fast and make available for querying in near real time

Storage scaling:

- To store data for querying or auditing
- To pay separately for data storage at cheaper rates, independent of compute usage

New third generation Redshift compute instance: RA3

Managed storage

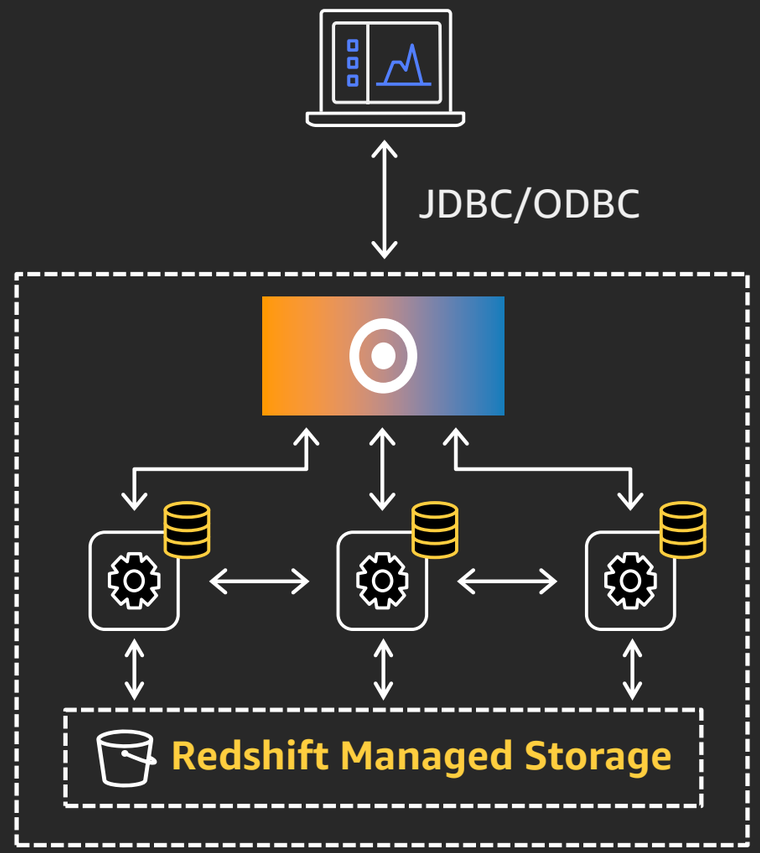
High-speed cache

High-bandwidth networking

Scale data warehouse only based on steady state compute needs

Pay separately for storage and compute

Automatic, no changes to any workflows, no need to manage storage



RA3: Unmatched performance at unbeatable price

RA3.16xl

On demand price—$13.04/node/hr

For storage pay as you use at $0.024/GB-month

Can scale to tens of PB of data (8PB compressed)

Coming soon
RA3.4xl

2x performance and 2x storage capacity compared to DS2.8XL for the same price

Up to 3x price-performance compared to any other Cloud DW

The minimum size RA3.16XL cluster scales up to 128TB (compressed), size only for your compute needs

Customer experience with RA3

Western Digital

"Our data is nearly **doubling** every year and we run **6 Redshift clusters with total 78 nodes and 631+ TB compressed data** stored to get insights that our business analysts and leadership depend on. The new Redshift **RA3 instances** offer us the ability to process our growing data more cost-effectively while we **doubling our storage capacity** **compared to our previous Redshift cluster."**

Fayaz Syed, Sr. Manager, Big Data Platform



"We load **billions of events per day** into Amazon Redshift and have **hundreds of terabytes of data that is expected to double every year**. While we store and process all our data, most of the analysis only **uses a subset of the data**. The new Redshift RA3 instance with managed storage **delivered 2x performance** for most of our queries compared to our previous DS2 instance based Redshift cluster."

Jonathan Burket, Senior Software Engineer



"Thrilled with the new RA3 instance type. We have observed a **1.9x performance improvement over DS2** and **1.5x performance improvement over DC2** in our workload while **keeping the same costs and providing scalable managed storage**. This allows us to keep pace with explosive data growth and have the necessary fuel to train our machine learning systems."

Stephen Moy, Software Engineer

Amazon Redshift: Key 2019 innovations



Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling



Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration



Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

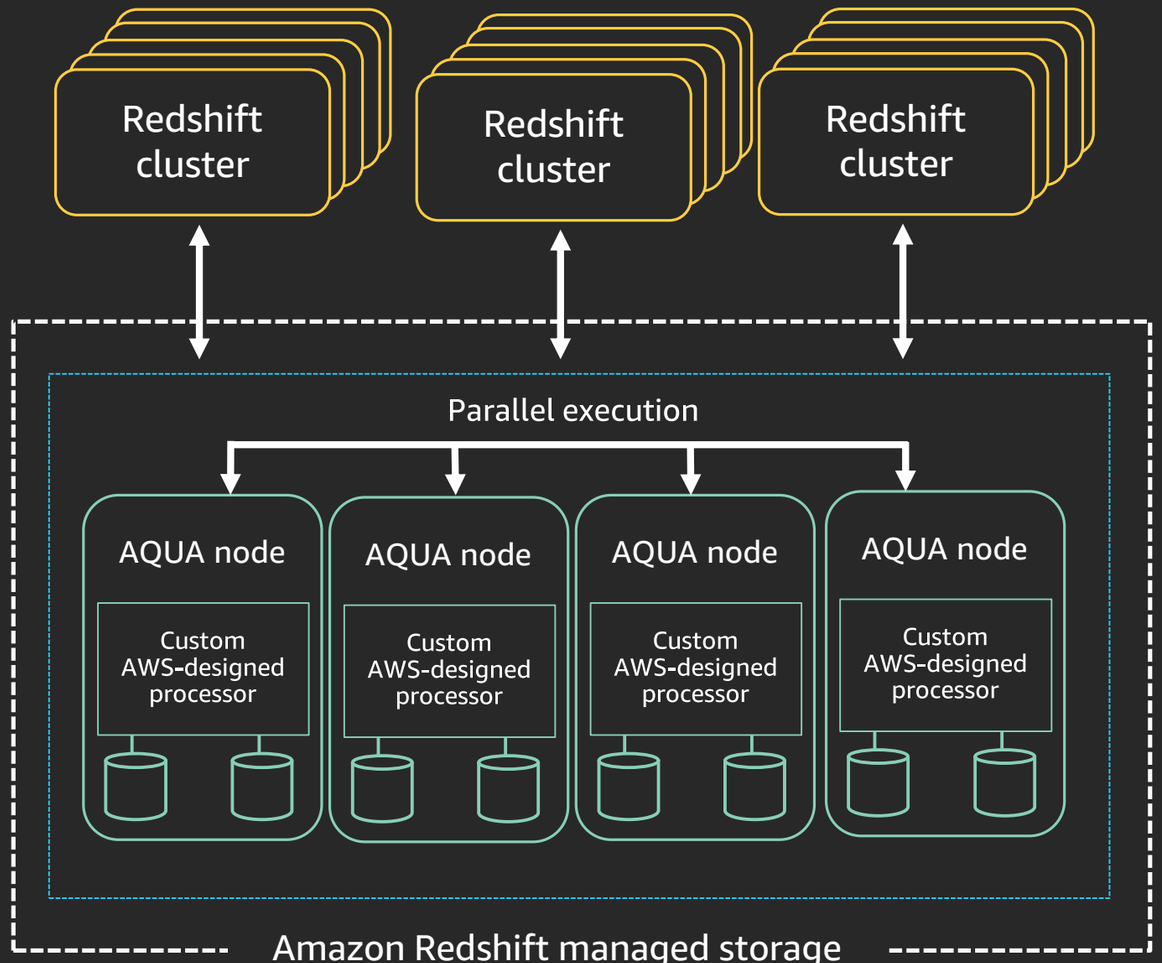
NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

AQUA (Advanced Query Accelerator) for Amazon Redshift



New distributed and hardware-accelerated processing layer.

With AQUA, Amazon Redshift is up to **10x faster** than any other cloud data warehouse, no extra cost.

AQUA nodes with custom AWS-designed analytics processors to make operations (compression, encryption, filtering, and aggregations) faster than traditional CPUs.

Available in Preview with RA3.
No code changes required.

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! Out of the box performance with AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

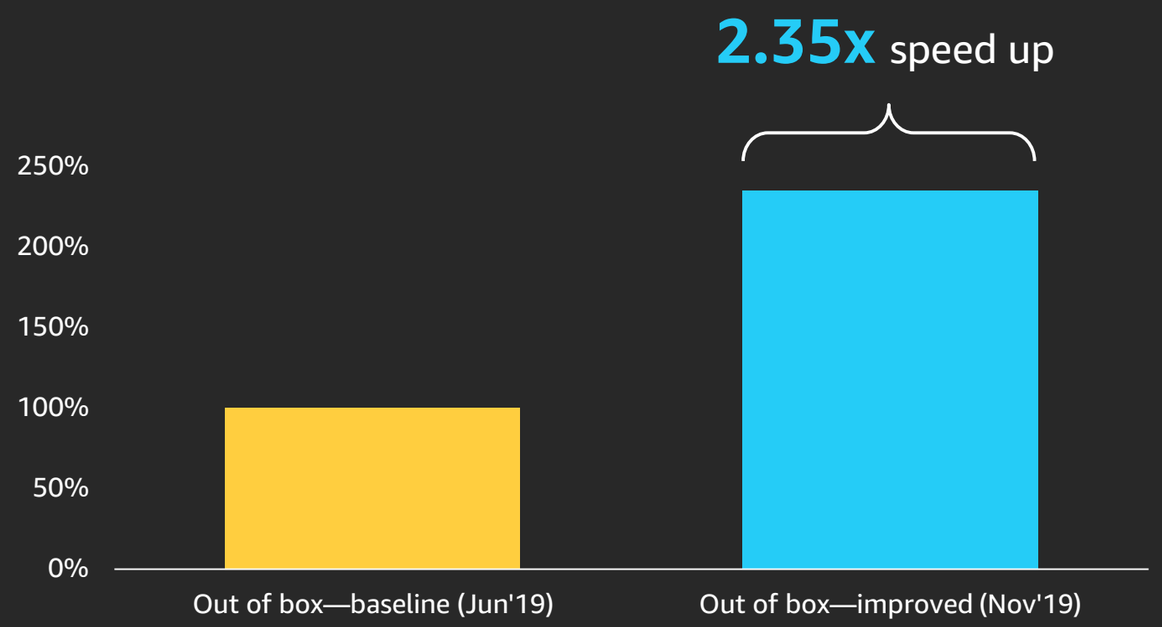
NEW! Elastic resize

Redshift out of the box performance is now >2x faster

Performance innovations:

- AZ64 encoding
- Bloom filters for collocated and broadcast JOIN queries
- Enhanced planner for modern hardware CPUs and networking
- HLL (HyperLogLog) statistics
- Cache-optimized aggregation and join processing

***Cloud DW benchmark, 30TB**
Normalized queries/hour (QPH)
(Higher is better)



* Cloud DW benchmark is based on TPC-DS (v2.10) with no query modifications done

AZ64 — a new compression encoding for optimized storage and high query performance

High compression ratio and improved query performance

Efficient compression on small groups of data

Leverages SIMD instructions for parallel processing

Applicable for all numeric and date/time datatypes

	AZ64 storage savings	AZ64 performance speed ups
RAW	60–70% less storage	25–30% faster
LZO	35% less storage	40% faster
ZSTD	Comparable footprint	70% faster

*Cloud DW benchmark, 30TB

* Cloud DW benchmark is based on TPC-DS (v2.10) with no query modifications done

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Introducing Materialized Views:
Compute once, query many times

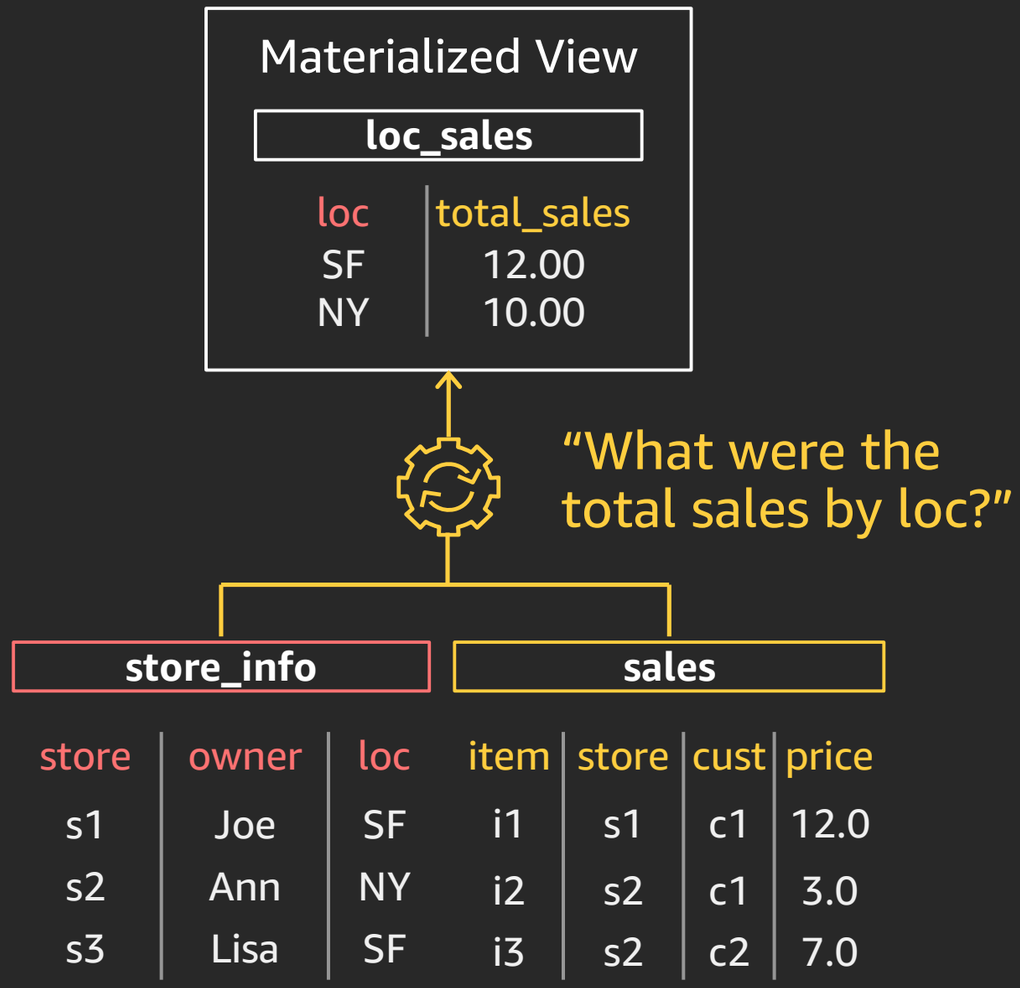
Speed up queries by orders of magnitude

- Joins, filters, aggregations, and projections

Simplify and accelerate ETL/BI pipelines

- Incremental refresh
- User triggered maintenance

Easier and faster migration to Redshift



Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW!
RA3 with Redshift managed storage

PREVIEW!
AQUA

NEW!
AZ64

PREVIEW!
Materialized views

NEW!
Spatial data support

NEW!
Concurrency Scaling

Data lake & AWS integration

PREVIEW!
Federated query across Redshift & RDS/Aurora

NEW!
Data lake export in Parquet

NEW!
Spectrum Request Accelerator

NEW!
Amazon Lake Formation integration

Fully managed

NEW!
New management console

NEW!
Auto WLM: Query priorities

NEW!
Elastic resize scheduler

NEW!
Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW!
Stored procedures

NEW!
Distribution and sort key advisor

NEW!
Cross-instance restore

NEW!
Faster cross regional copy and change

NEW!
Auto Data Distribution

NEW!
Deferred Maintenance

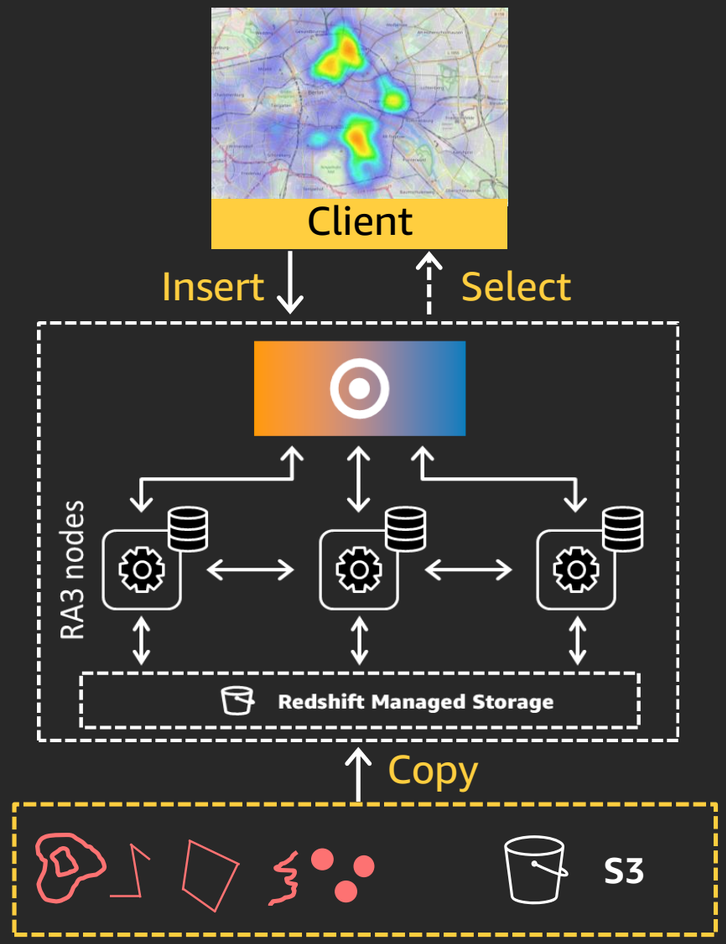
NEW!
Elastic resize

New data type Geometry:
Ingest, store, and analyze spatial data

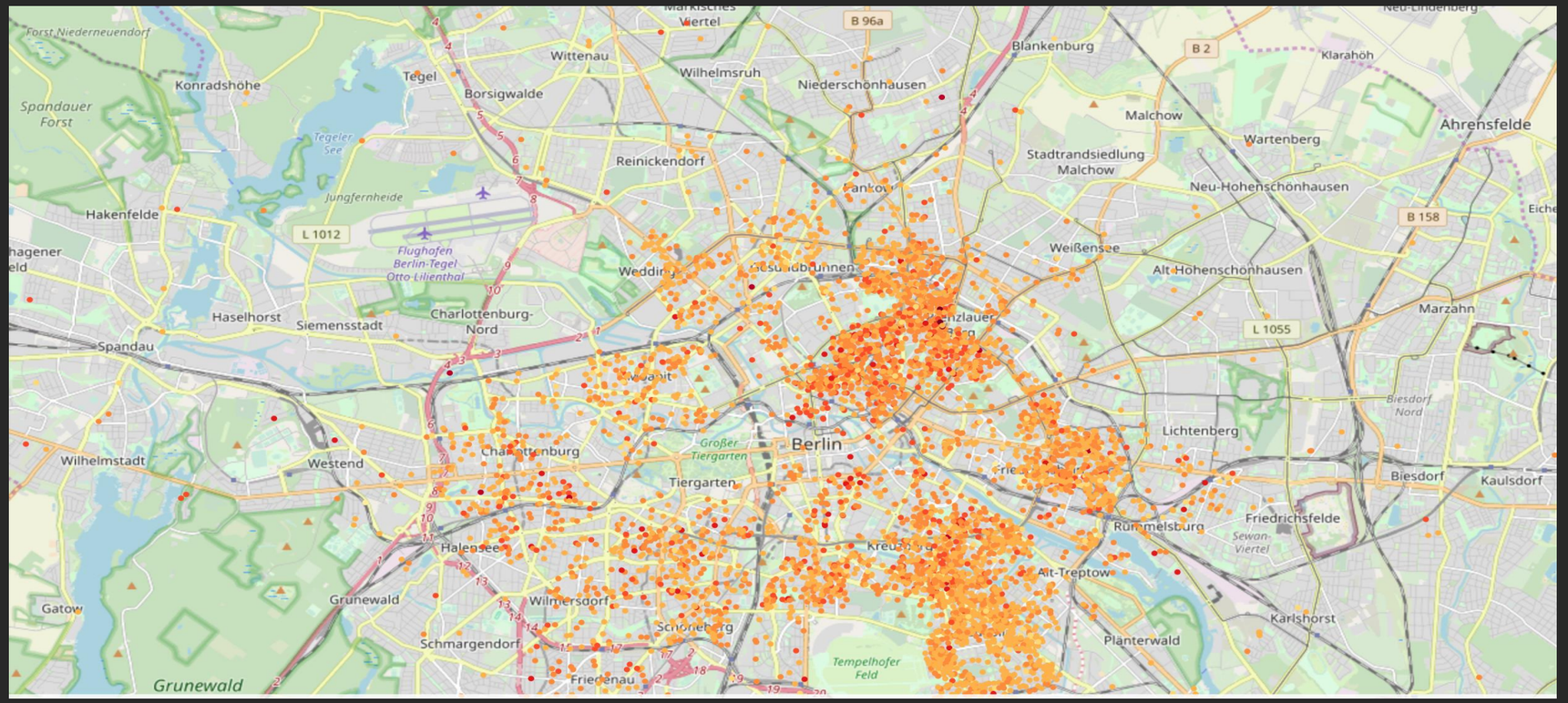
Seamlessly integrate spatial and business data

Geometry data types supports shapes such as MultiPoint, MultiLinestring, MultiPolygon, and GeometryCollection

40+ SQL spatial functions to construct geometric shapes, import, export, access and process the spatial data



Spatial processing — sample query




```
SELECT name, ST_X(shape) as lng, ST_Y(shape) as lat, price
FROM accommodations
WHERE ST_Within(shape, ST_GeomFromText( 'POLYGON((13.111839294433596
52.4285942596063, 13.111839294433596 52.60117089057946, 52.4285942596063))', 4326))
LIMIT 5000
```

Data Types

GEOMETRY
Point, Linestring, Polygon,
MultiPoint, MultiLinestring,
MultiPolygon,
GeometryCollection

Spatial Accessors

ST_NumGeometries,
ST_GeometryType,
ST_Dimension, …

Spatial Predicates

ST_Covers, ST_Equals,
ST_Within, ST_DWithin,…

Spatial Functions

ST_Distance,
ST_Azimuth, …

Spatial Formats

WKT/WKB, EWKT/EWKB, GeoJSON
Ingestion: CSV

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

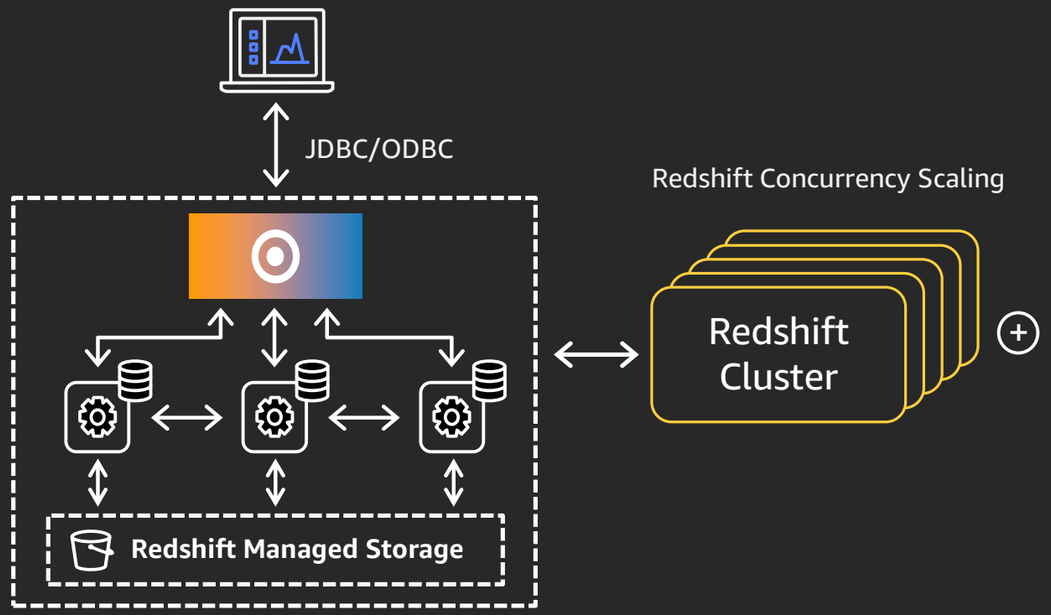
NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Compute elasticity and scalability with Concurrency Scaling



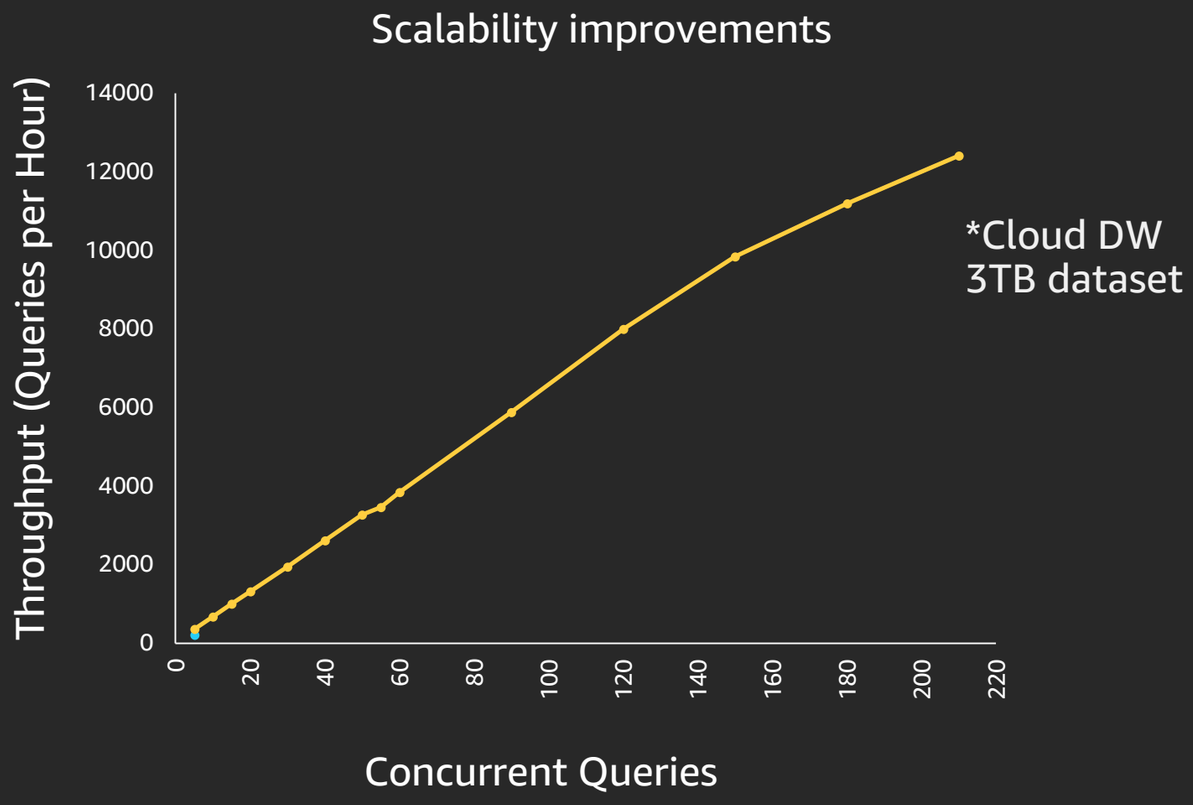
Scale-out to multiple Redshift clusters from a single endpoint in seconds

Support virtually unlimited concurrent users and queries while maintaining SLAs

Per-second billing for additional clusters used

Free 1 hour usage per day (free for 97% of clusters)

>35x improvement in throughput in 2019



* Cloud DW benchmark is based on TPC-DS (v2.10) with no query modifications done

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Amazon Redshift federated query (preview)

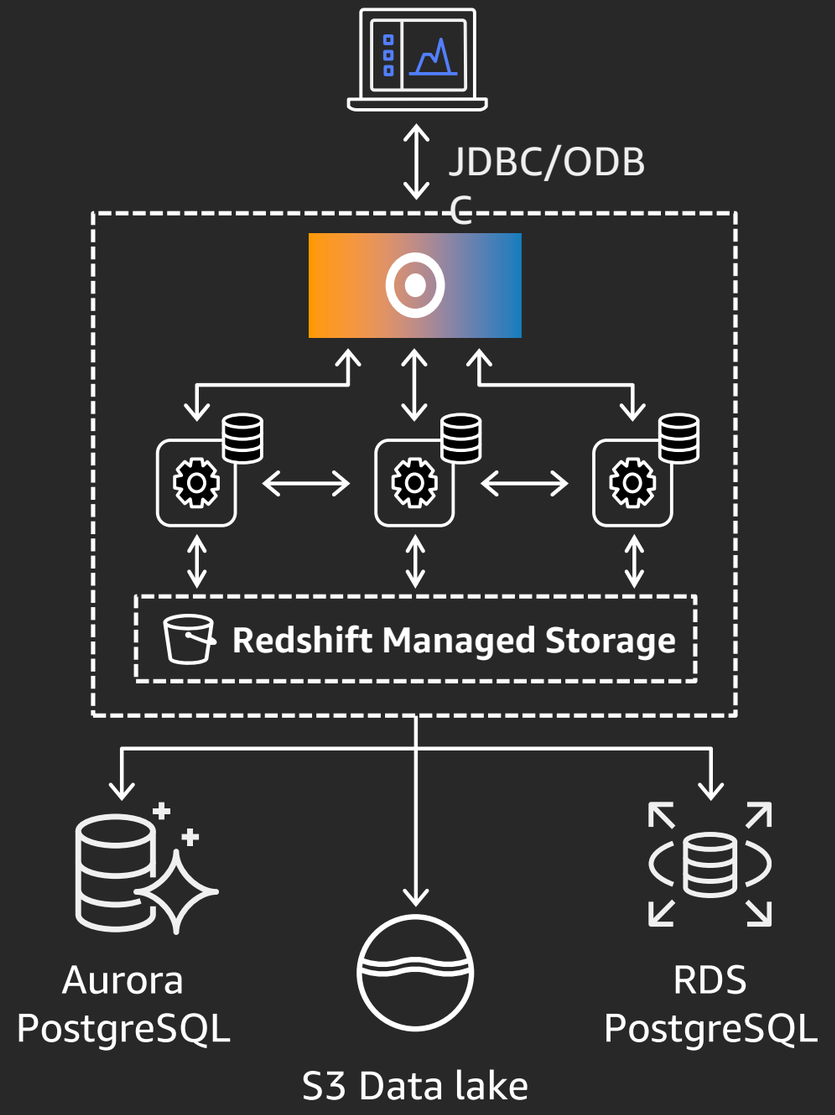
Queries on RDS and Aurora PostgreSQL databases

Analytics on live data without data movement

Unified analytics across data warehouse, data lake and operational databases

Flexible and easy way to ingest data

Performant and secure access to data



Amazon Redshift federated query:
Most up to date data, no ETL required

```
-- Aurora Postgres has Hot Data (2019)
-- Redshift has Recent Data (2016-2018)
-- S3 has Archival Data (1992-1998)
-- Declare a view across all backends

CREATE VIEW lineitem_all AS
SELECT * FROM s3.lineitem_1t_part
UNION ALL
SELECT * FROM public.lineitem
UNION ALL
SELECT * FROM apg.lineitem
WITH NO SCHEMA BINDING
```

```
-- Find #sales with 1 item in Jan of each year
-- Predicates are being pushed down
-- Partition pruning on the S3 data
-- Aggregates are being pushed down
-- Very intuitive syntax

SELECT EXTRACT(year FROM l_shipdate) AS year,
       EXTRACT(month FROM l_shipdate) AS month,
       COUNT(*) AS orders
FROM lineitem_all
WHERE extract(month FROM l_shipdate) = 1
AND l_quantity < 2
GROUP BY 1,2
ORDER BY 1,2;
```

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

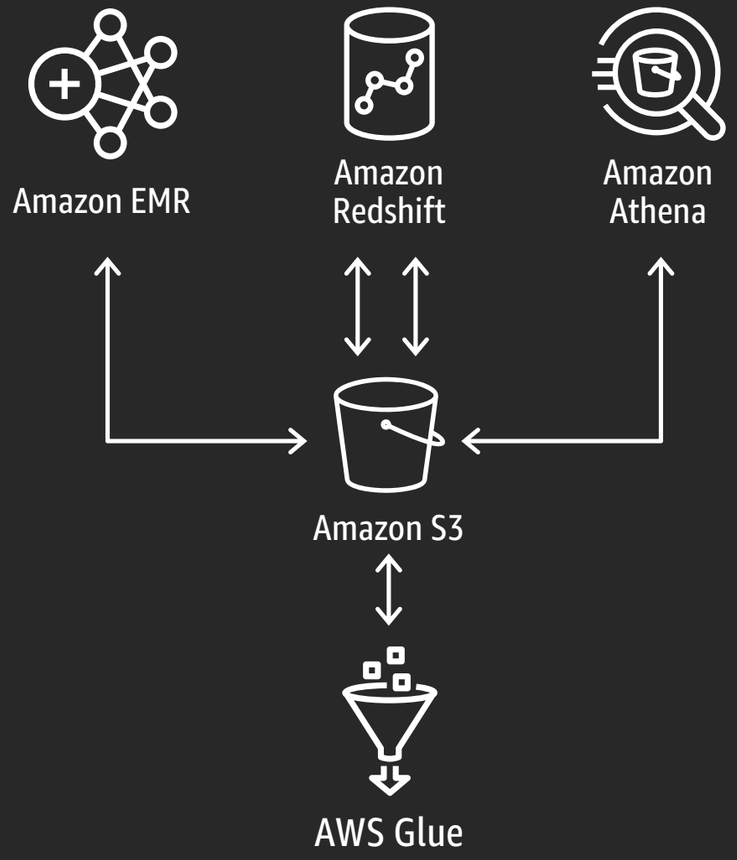
NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Data lake export: share data in Parquet format



Parquet is efficient open columnar storage format for analytics

Analyze your data with Redshift Spectrum and other AWS services such as Amazon Athena and Amazon EMR

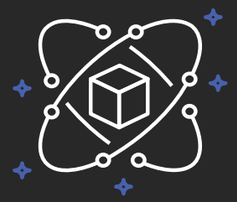
```
UNLOAD
('select * from lineitem')
TO
's3://mybucket/unload/lineitem/'
FORMAT as PARQUET
PARTITION BY (l_shipdate);
```

Self optimizing and easy to maintain



Fully managed

Easy to provision and manage, automated backups, AWS support, 99.9% SLAs



Automatic optimizations

Self-learning and self-optimizing to turbo charge performance based on usage patterns



Easy to customize

Get the best out of the cluster hardware and spend where it matters to the business

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Redshift turbo charges query performance with machine learning-based automatic optimizations

Automates table maintenance

Optimizes for peak performance as data and workloads scale

Leverages machine learning

Offers prescriptive recommendations with ability to apply changes dynamically



Automatic Analyze



Automatic Table Distribution Style



Distribution/Sort key advisors



Automatic Vacuum Delete



Automatic Table Sort

Automatic Table Sort

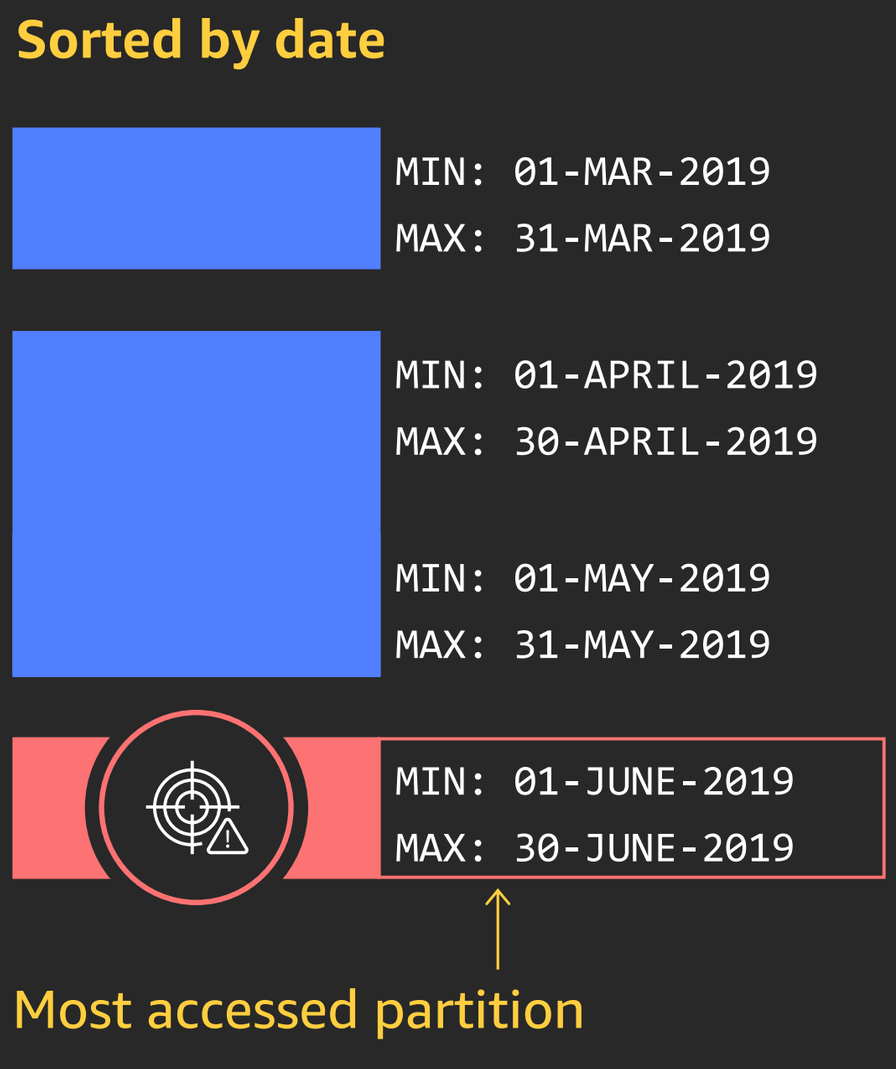
Eliminates DBA-managed sorting

Automates sort-order maintenance

Maximizes efficient use of compute

Leverages machine learning

No impact to performance or accessibility



Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

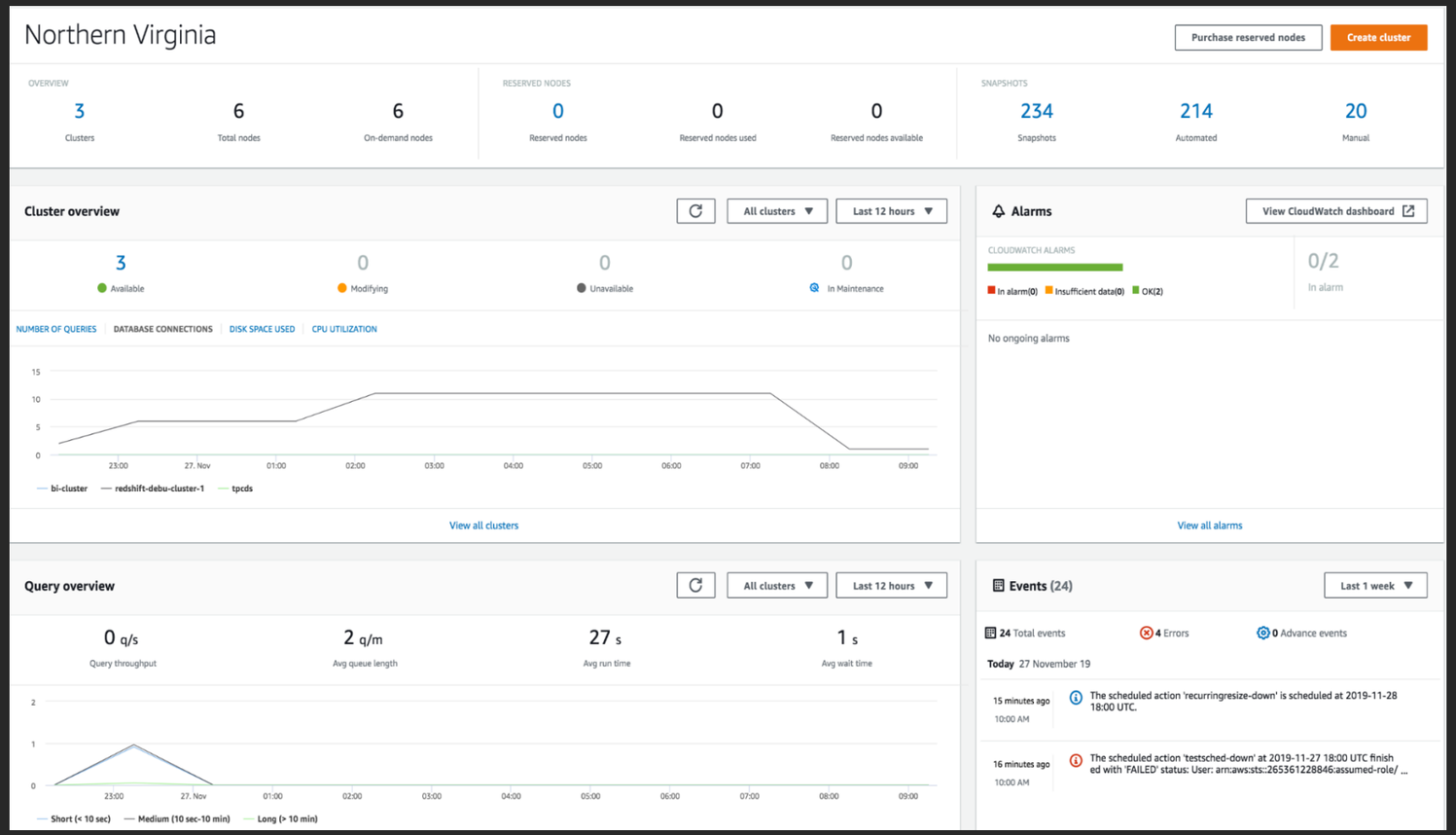
NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

New management console:
Intuitive, fast, and feature-rich



- Manage multiple data warehouses with command center-like dashboard
- Reduced time to diagnose when unexpected happens
- Share Query Editor with non-admin users
- Visually analyze your query results right in the query editor

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

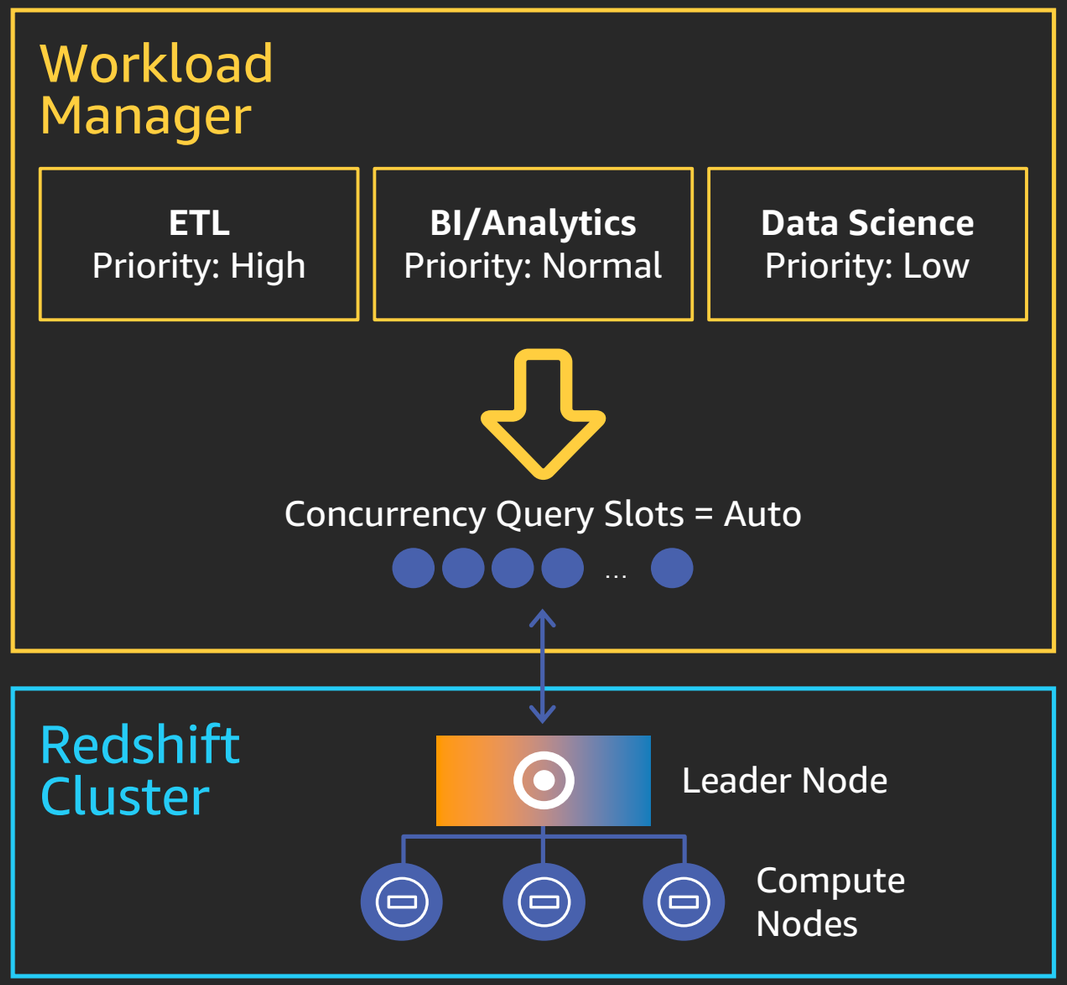
NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Efficiency with automatic workload management and query priorities



- Manages concurrency dynamically
- Supports efficient sharing of cluster resources
- Maximizes throughput and performance
- Leverages machine learning to classify queries based on resources needs
- Ability to influence workload performance based on business priorities
- Ensures low priority queries make progress

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

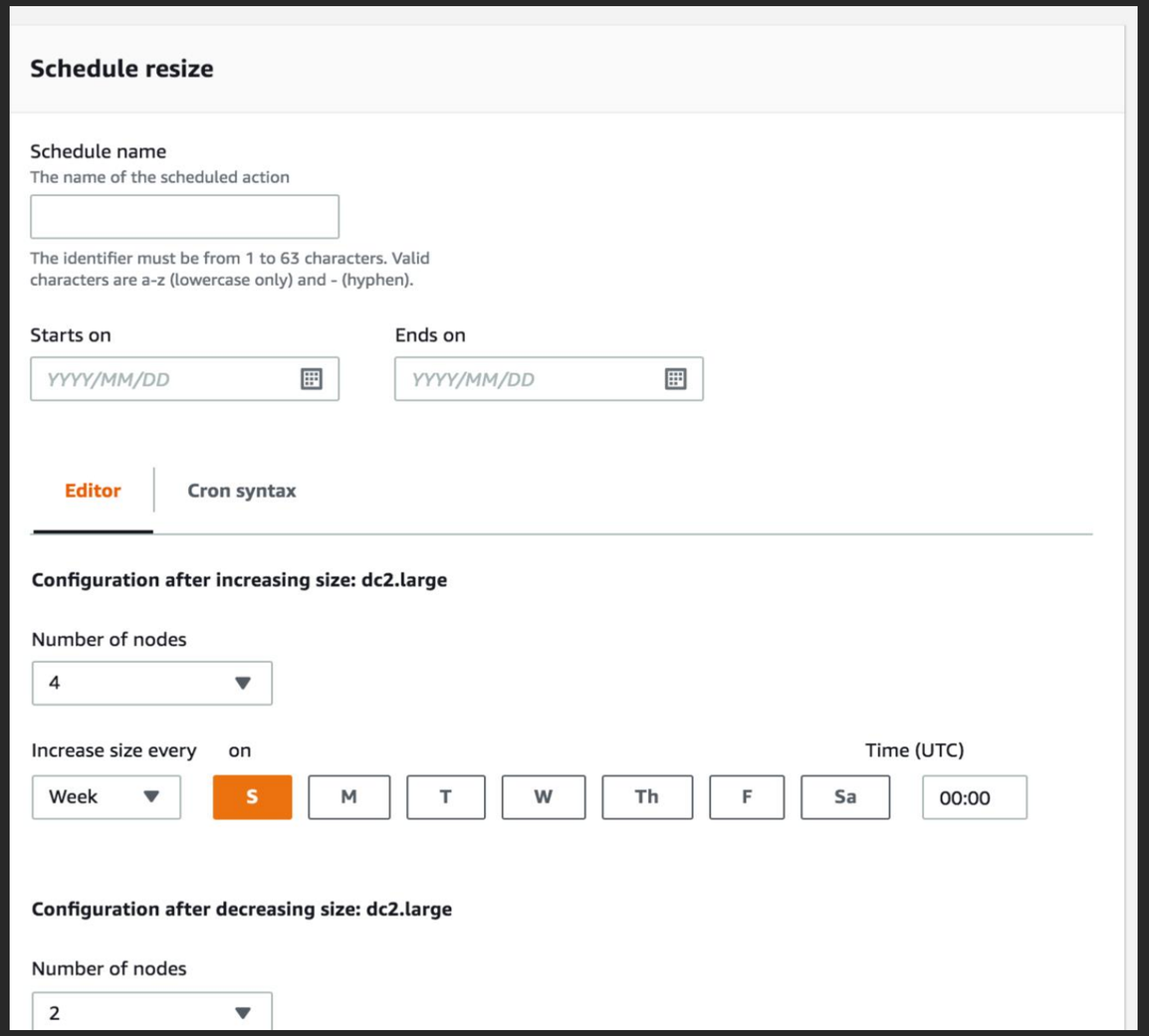
Scheduler for elastic resize: Optimize cost easily

Add or remove nodes in minutes using elastic resize

Scheduled cluster resize using management console or API

Removes dependency on external function and failures

Optimize cost and plan ahead for peak demand



Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

Stored procedures support to simplify migrations



Bring your existing **stored procedures** to Amazon Redshift

Migrating to Redshift is even easier!

Redshift supports stored procedures in PL/pgSQL format

Support for stored procedures provides the ability to run code where the data is to efficiently run ETL, data validation, and custom business logic

```
CREATE OR REPLACE PROCEDURE test_spl(f1 int,
f2 varchar)
AS $$
BEGIN
  RAISE INFO 'f1 = %, f2 = %', f1, f2;
END;
$$ LANGUAGE plpgsql;

Call test_spl(5, 'abc');
INFO: f1 = 5, f2 = abc
CALL
```

Amazon Redshift: Key 2019 innovations

Performance & scalability

NEW! RA3 with Redshift managed storage

PREVIEW! AQUA

NEW! AZ64

PREVIEW! Materialized views

NEW! Spatial data support

NEW! Concurrency Scaling

Data lake & AWS integration

PREVIEW! Federated query across Redshift & RDS/Aurora

NEW! Data lake export in Parquet

NEW! Spectrum Request Accelerator

NEW! Amazon Lake Formation integration

Fully managed

NEW! Auto-Vacuum, Auto-Analyze & Auto Table Sort

NEW! New management console

NEW! Auto WLM: Query priorities

NEW! Elastic resize scheduler

NEW! Stored procedures

NEW! Distribution and sort key advisor

NEW! Cross-instance restore

NEW! Faster cross regional copy and change

NEW! Auto Data Distribution

NEW! Deferred Maintenance

NEW! Elastic resize

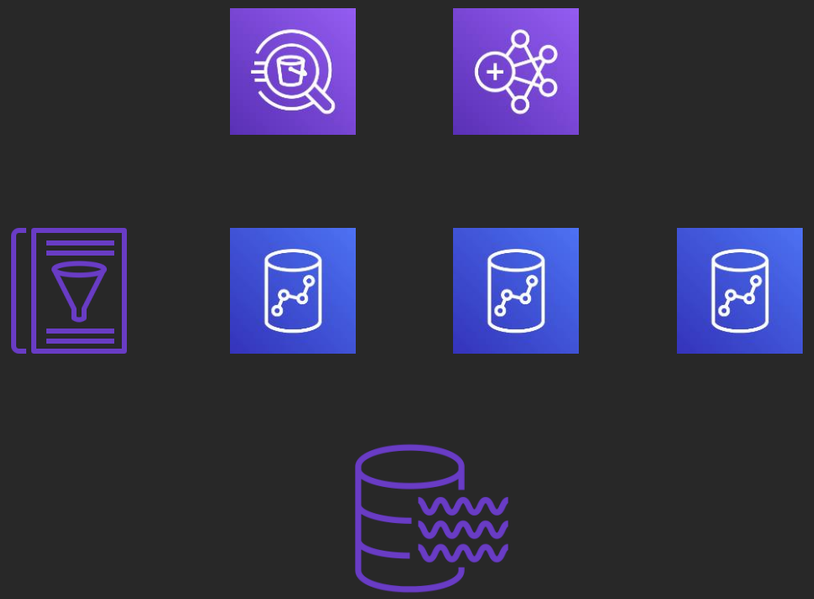
Yelp – building analytics at scale with Amazon Redshift

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

Hello, my name is Steven



Clickstream sessionization

Data warehouse management

Data lake management

SQL-on-anything

Agenda

Yelp's analytics needs for both external and internal customers

Background on Yelp's data ecosystem

How Amazon Redshift played a key role in Yelp's ecosystem

What's next in Yelp's data ecosystem

Who is the Yelp Analytics audience?

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Our mission: Connecting people with great local businesses

4.8 million
Active claimed local business locations[1, 2]

199 million
Cumulative reviews[1]

38 million
App unique devices[3]

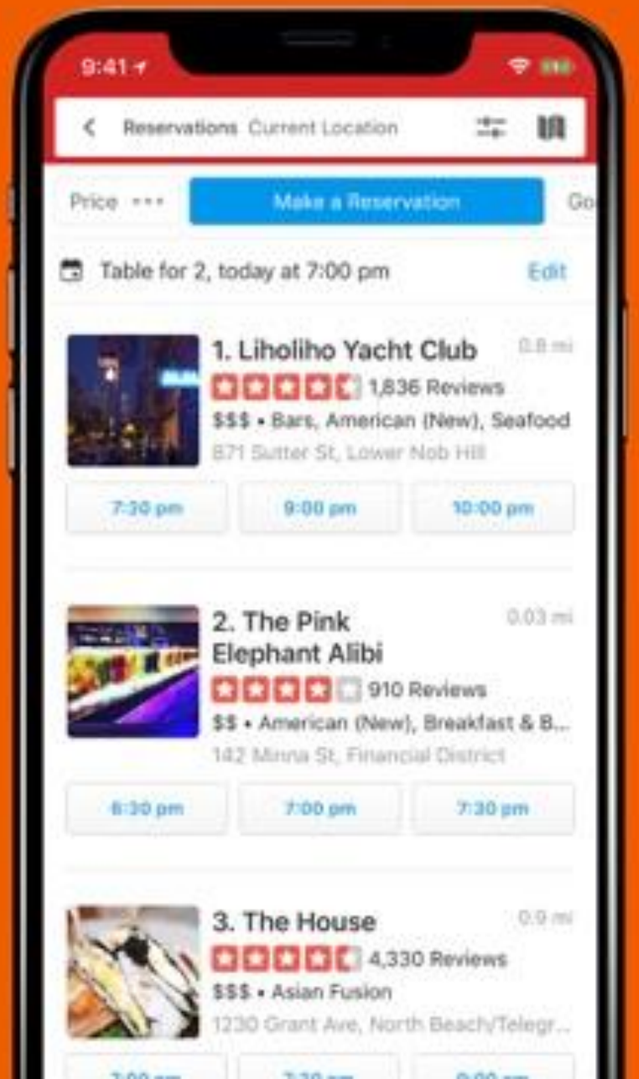
563,000
Paying advertising locations[4]

2.6 million
Seated diners via Yelp[3]

2.2 million
Request-a-quote leads delivered[3]

1. As of September 30, 2019
2. Represents the number of claimed local business locations that are both (a) active on Yelp and (b) associated with an active business account as of a given date. We consider a claimed local business to be active if it has not closed, been removed from our platform, or merged with another claimed local business location.
3. Monthly average for Q3 2019
4. All business locations associated with a business account from which Yelp recognized advertising revenue in a given month, averaged over Q3 2019.

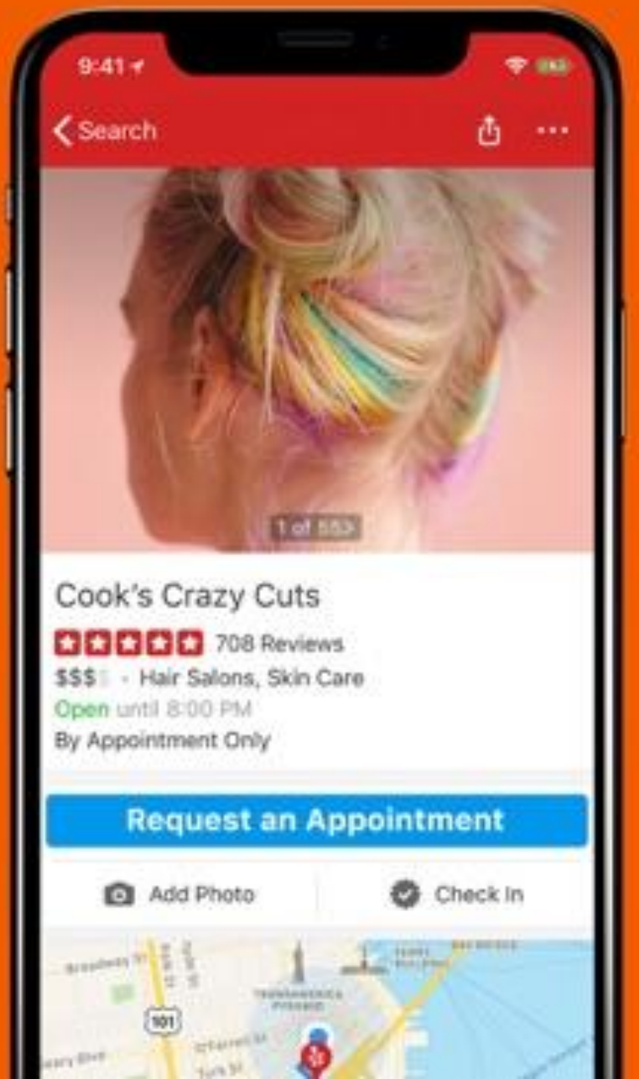
Make a reservation



Hire a home pro



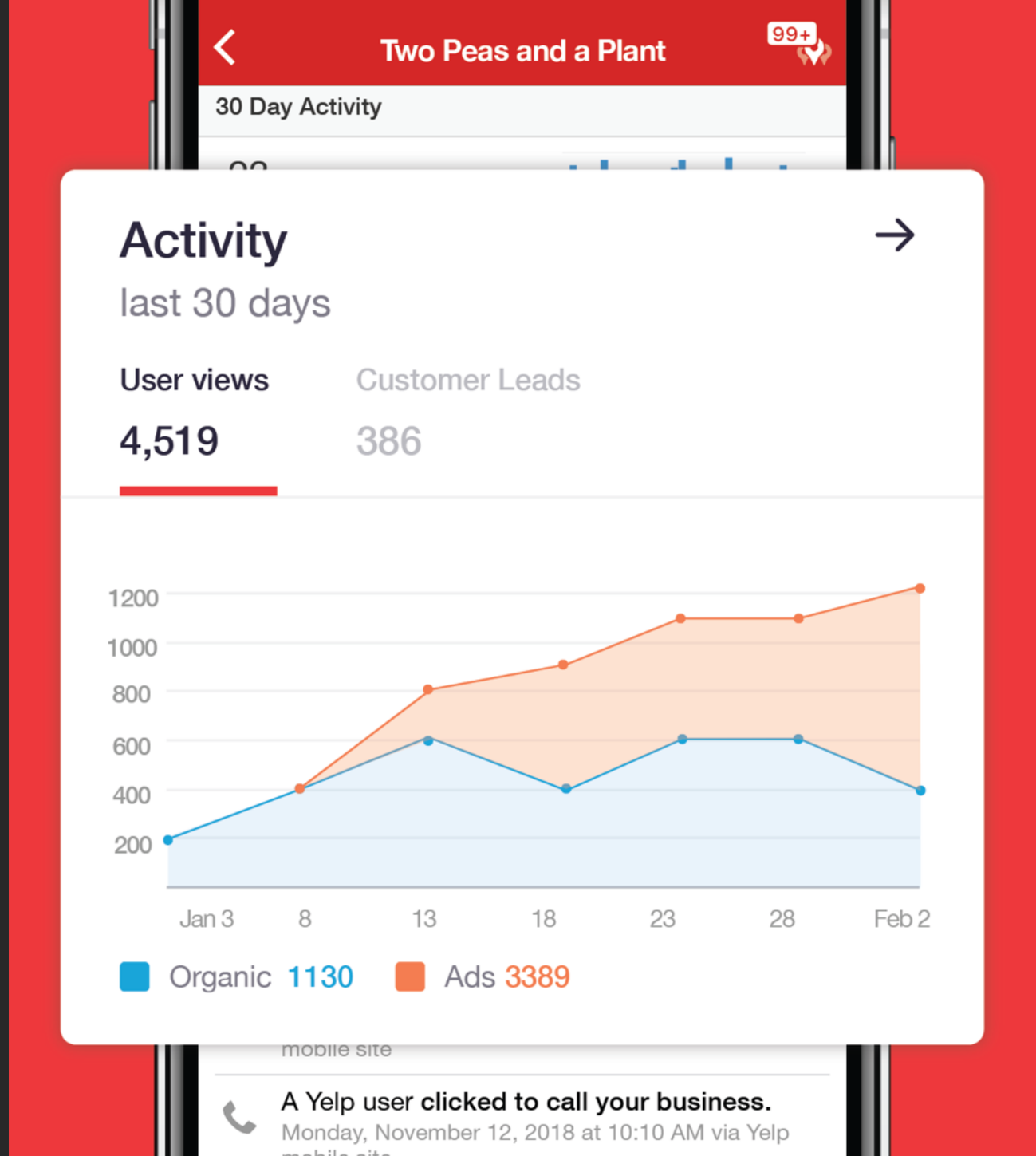
Book an appointment



Empowering local business owners

Business owners can claim their Yelp Business Page for free to engage the Yelp community

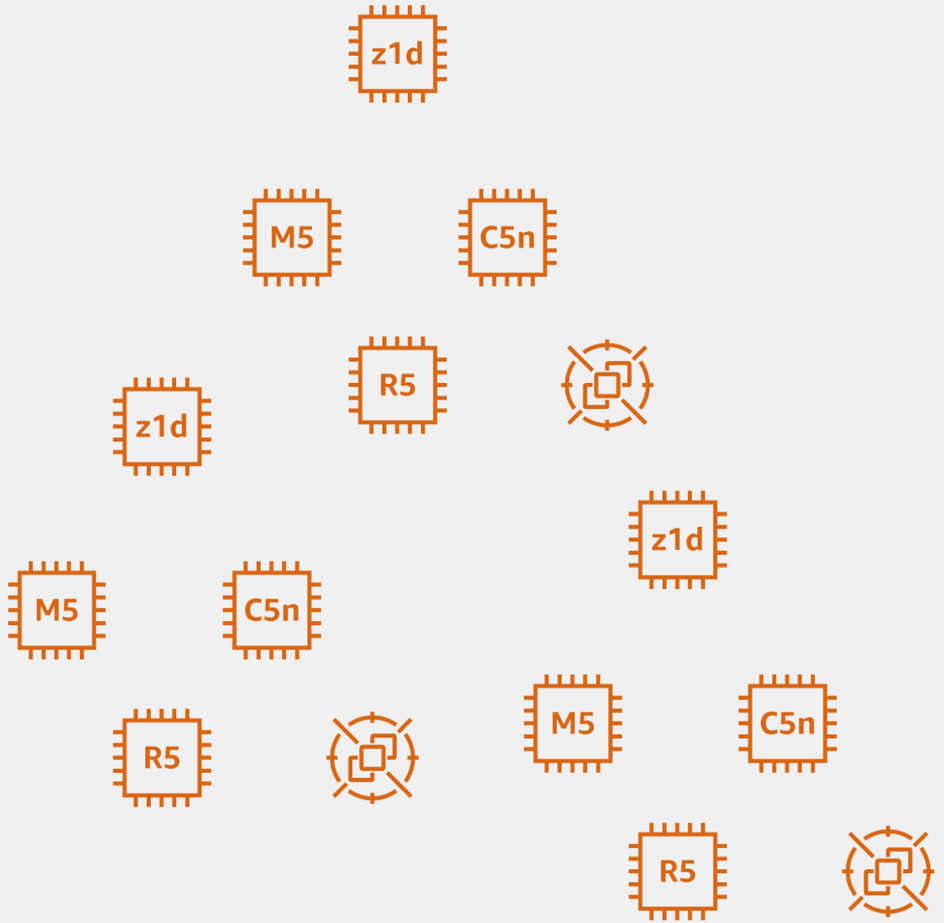
In addition, they can track user and customer views in their business owner portal



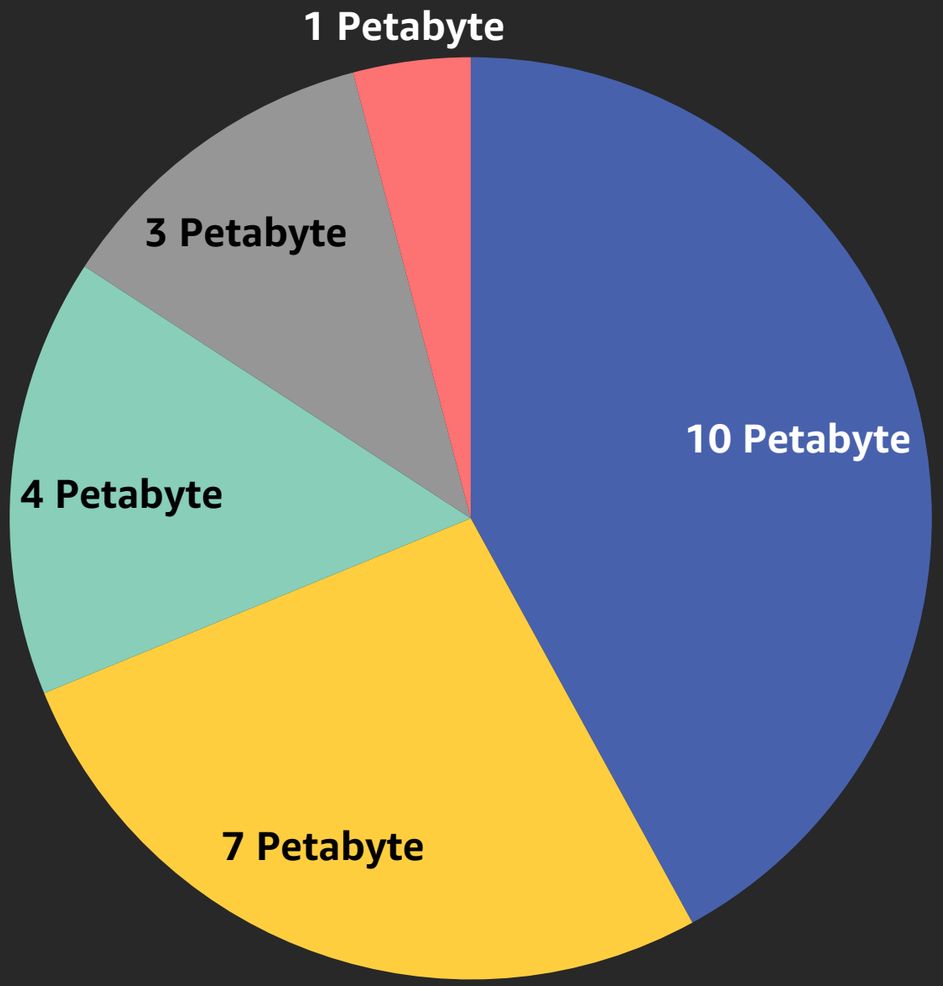
Compute at scale

4.8 million location times,
38 million app uniques
naively arrives at 182
trillion computation

We need to use all the
data processing techniques
available to compute these
numbers efficiently



Storage at scale



Yelp’s data lake is primarily on intelligent tiering

- Infrequent access
- Standard storage
- Glacier
- Intelligent tier infrequent access
- Intelligent tier standard

Moreover, analytics audience continues to expand



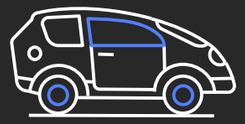
Analysts and data scientists

Deducing relationships between variables, and figuring out different interactions to build predictive models



Product manager

Leveraging internal dashboards to drive product feature decision and monitor product health



Engineers

Using data pipeline to power all kinds of user- and business-facing features and machine-learning models

Yelp’s data ecosystem

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

One size fits all

An idea whose time has come and gone



Michael Stonebraker
ACM Turing Award 2014

"One Size Fits All": An Idea Whose Time Has Come and Gone

Michael Stonebraker
Computer Science and Artificial Intelligence Laboratory, M.I.T., and StreamBase Systems, Inc.
stonebraker@csail.mit.edu

Uğur Çetintemel
Department of Computer Science Brown University, and StreamBase Systems, Inc.
ugur@cs.brown.edu

Abstract

The last 25 years of commercial DBMS development can be summed up in a single phrase: "One size fits all". This phrase refers to the fact that the traditional DBMS architecture (originally designed and optimized for business data processing) has been used to support many data-centric applications with widely varying characteristics and requirements.

In this paper, we argue that this concept is no longer applicable to the database market, and that the commercial world will fracture into a collection of independent database engines, some of which may be unified by a common front-end parser. We use examples from the stream-processing market and the data-warehouse market to bolster our claims. We also briefly discuss other markets for which the traditional architecture is a poor fit and argue for a critical rethinking of the current factoring of systems services into products.

1. Introduction

Relational DBMSs arrived on the scene as research prototypes in the 1970's, in the form of System R [10] and INGRES [27]. The main thrust of both prototypes was to surpass IMS in value to customers on the applications that IMS was used for, namely "business data processing". Hence, both systems were architected for on-line transaction processing (OLTP) applications, and their commercial counterparts (i.e., DB2 and INGRES, respectively) found acceptance in this arena in the 1980's. Other vendors (e.g., Sybase, Oracle, and Informix) followed the same basic DBMS model, which stores relational tables row-by-row, uses B-trees for indexing, uses a cost-based optimizer, and provides ACID transaction properties.

Since the early 1980's, the major DBMS vendors have steadfastly stuck to a "one size fits all" strategy, whereby they maintain a single code line with all DBMS services. The reasons for this choice are straightforward — the use of multiple code lines causes various practical problems, including:

- *a cost problem*, because maintenance costs increase at least linearly with the number of code lines;
- *a compatibility problem*, because all applications have to run against every code line;
- *a sales problem*, because salespeople get confused about which product to try to sell to a customer; and
- *a marketing problem*, because multiple code lines need to be positioned correctly in the marketplace.

To avoid these problems, all the major DBMS vendors have followed the adage "put all wood behind one arrowhead". In this paper we argue that this strategy has failed already, and will fail more dramatically off into the future.

The rest of the paper is structured as follows. In Section 2, we briefly indicate why the single code-line strategy has failed already by citing some of the key characteristics of the data warehouse market. In Section 3, we discuss stream processing applications and indicate a particular example where a specialized stream processing engine outperforms an RDBMS by two orders of magnitude. Section 4 then turns to the reasons for the performance difference, and indicates that DBMS technology is not likely to be able to adapt to be competitive in this market. Hence, we expect stream processing engines to thrive in the marketplace. In Section 5, we discuss a collection of other markets where one size is not likely to fit all, and other specialized database systems may be feasible. Hence, the fragmentation of the DBMS market may be fairly extensive. In Section 6, we offer some comments about the factoring of system software into products. Finally, we close the paper with some concluding remarks in Section 7.

2. Data warehousing

In the early 1990's, a new trend appeared: Enterprises wanted to gather together data from multiple operational databases into a data warehouse for business intelligence

Right tool for the right job

Lower latency
(ms)











Amazon Simple
Storage Service



Amazon Redshift



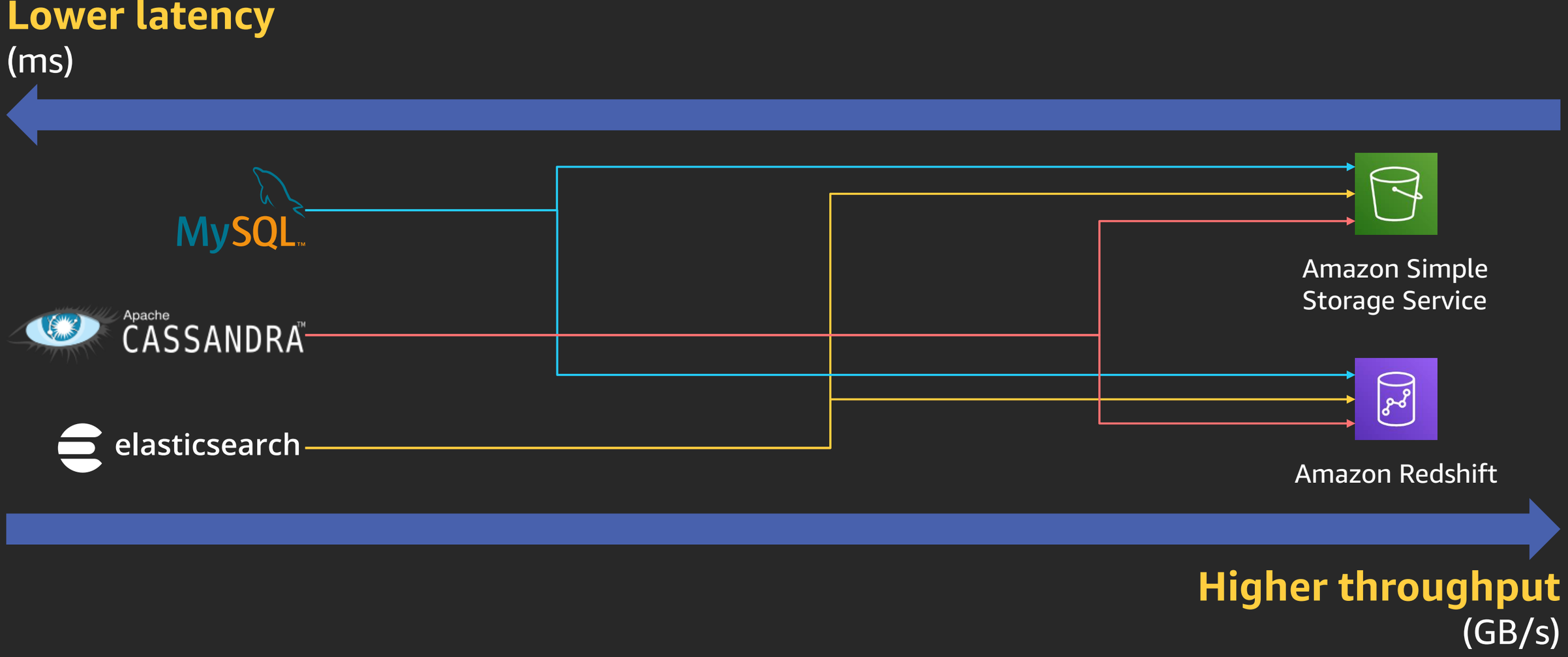
Higher throughput
(GB/s)

Great, what about data silos?



Bad idea

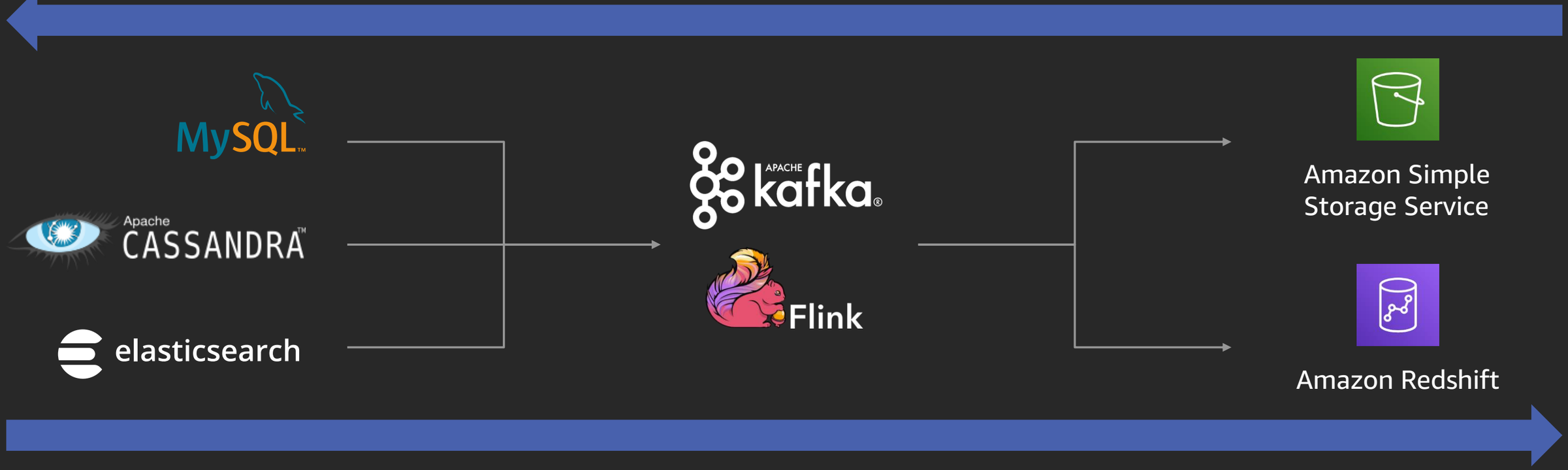
Developing n x m batches to move data



Good idea

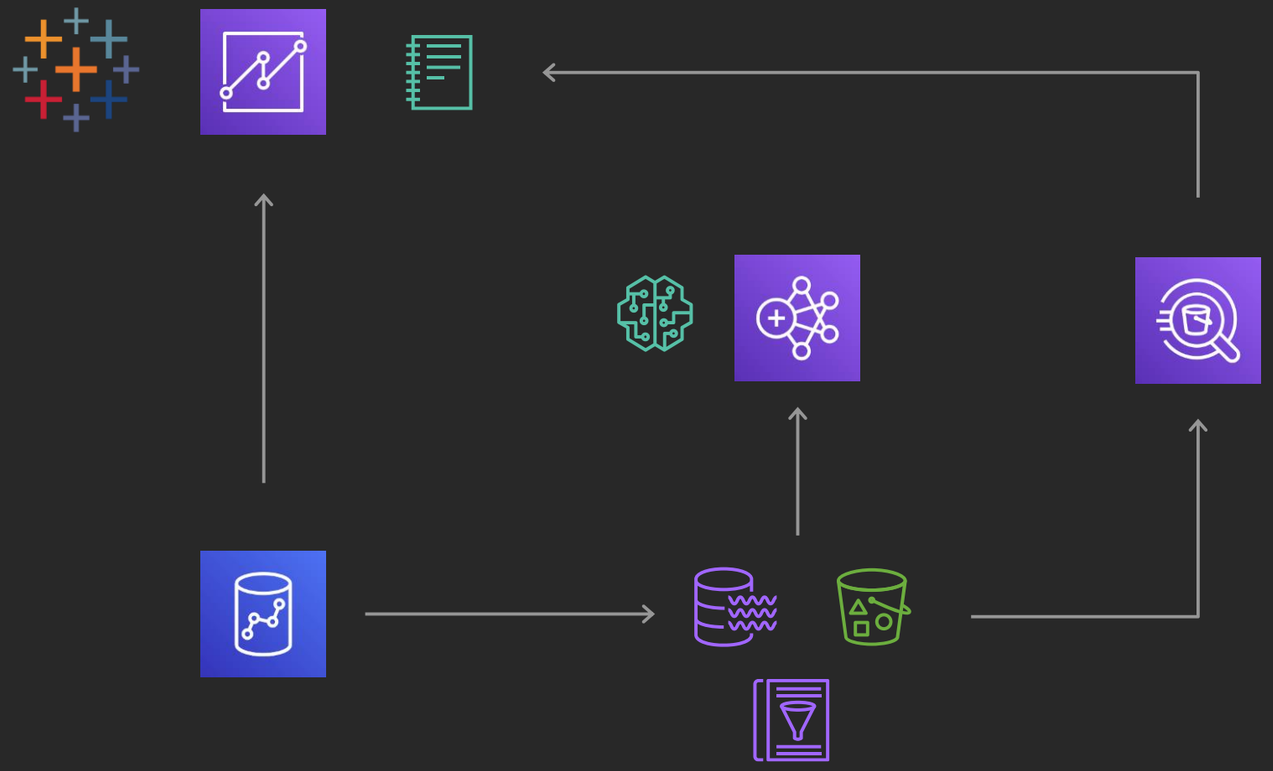
Connector ecosystem

Lower latency
(ms)



Higher throughput
(GB/s)

Analytics powered by Redshift



Visualization Tier

Processing Tier

Data Tier

How did Amazon Redshift play a key role in Yelp's ecosystem?

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we are looking for in a modern data warehouse

State-of-the-art optimizer

ANSI SQL

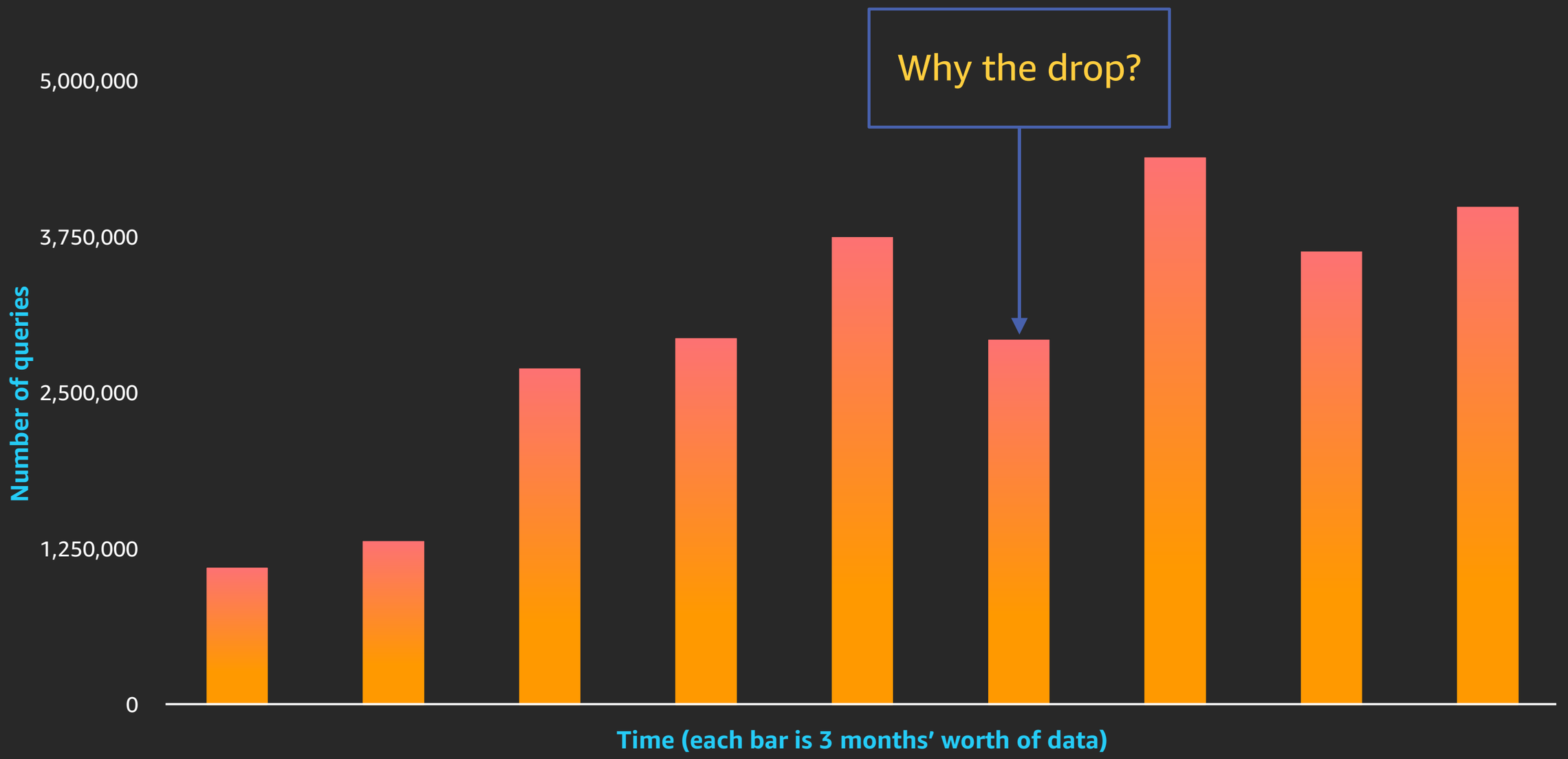
Elastic

Columnar storage engine

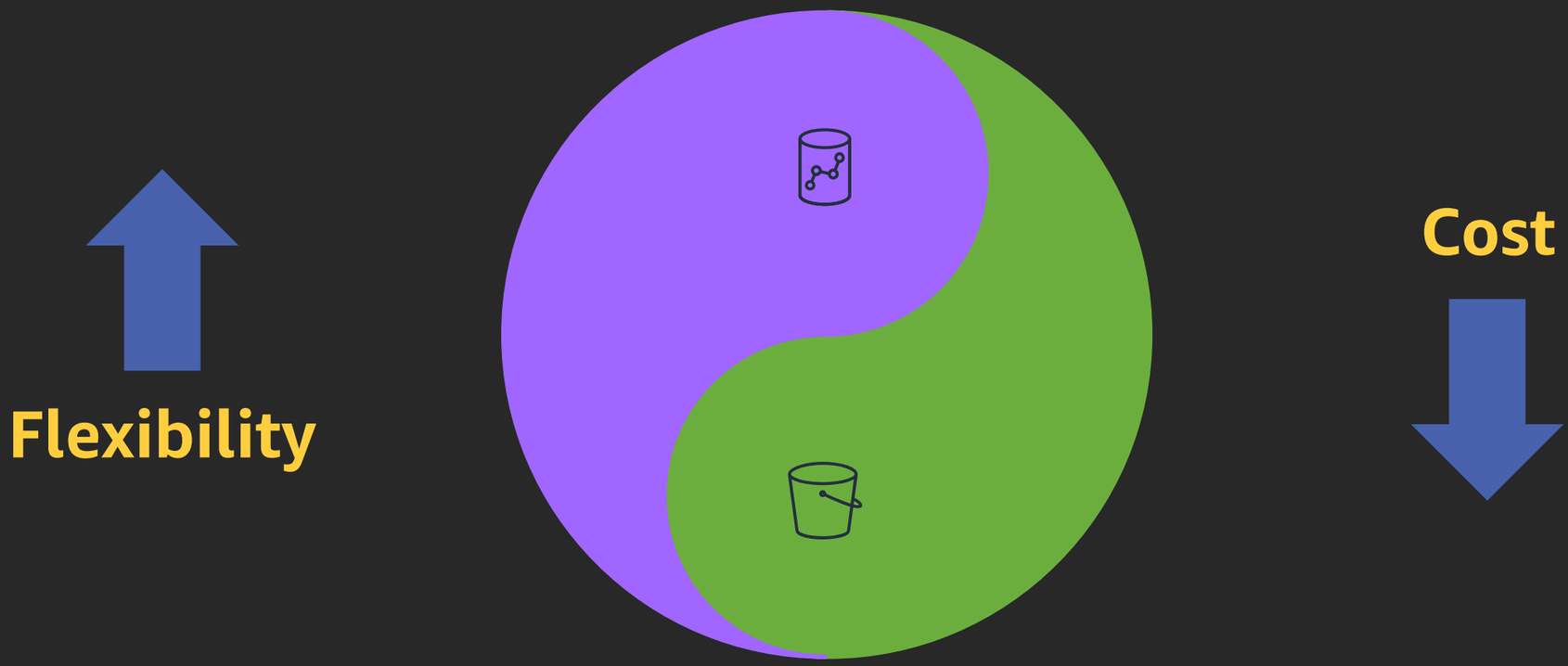
SIMD support

Auto-scale

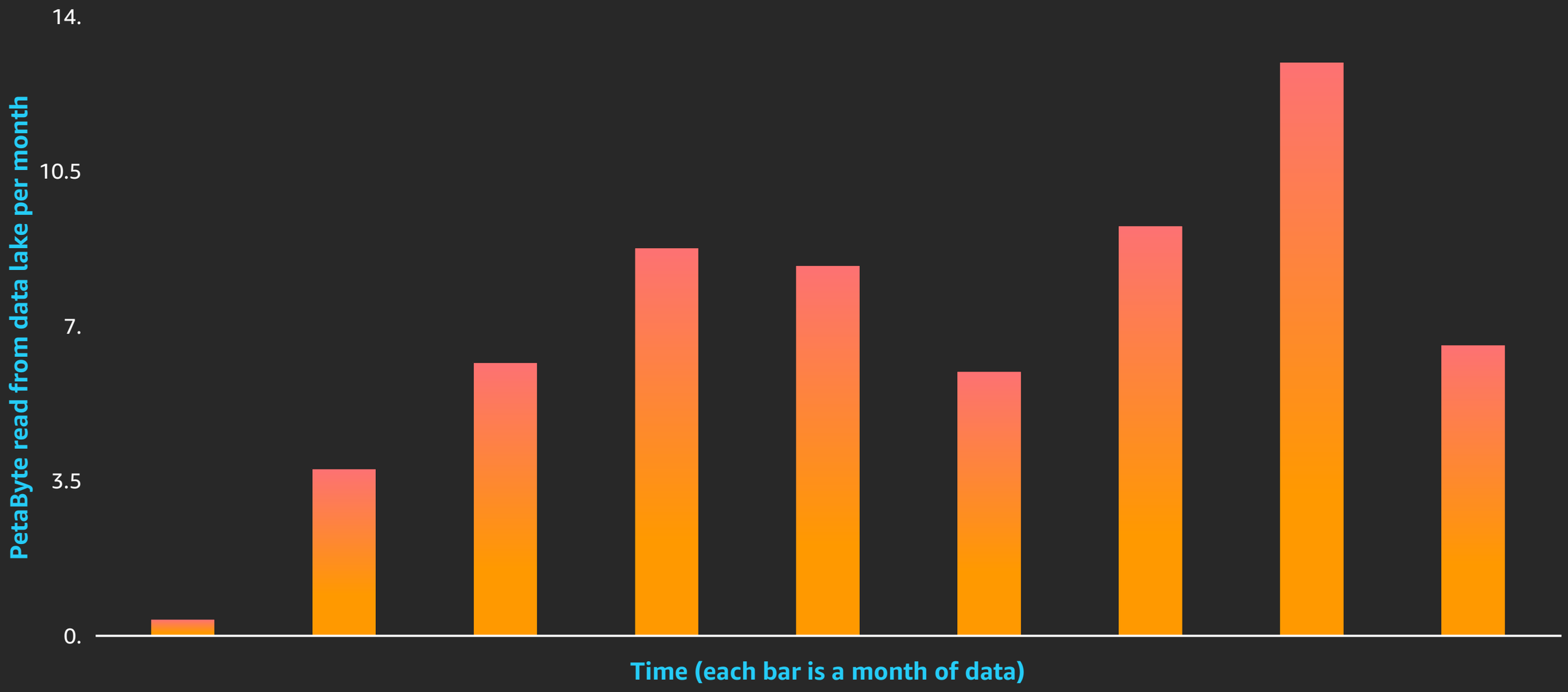
Redshift queries growth (last 2 years)



Along the way, Redshift team surprises us with Spectrum

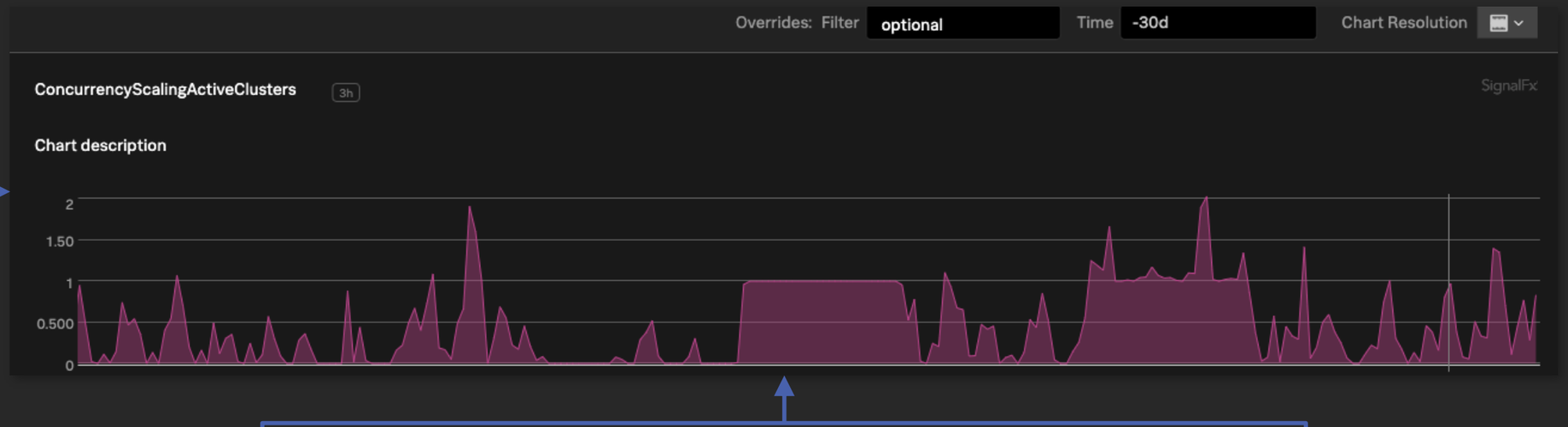


Redshift Spectrum punches above the cluster's weight



Concurrency scaling allows compute auto-scaling

Y-axis is an average of extra concurrent-scaled cluster driven by query workload



X-axis is time — in this graph you have about 30 days of concurrency scaling usage

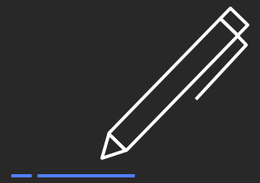
Have you ever thought about how many resources your query needs?



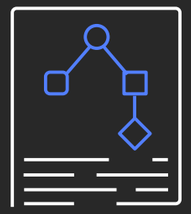
```
XN Merge  (cost=1002815366604.92..1002815366606.36 rows=576 width=27)
  Merge Key: sum(sales.pricepaid)
  ->  XN Network  (cost=1002815366604.92..1002815366606.36 rows=576 width=27)
        Send to leader
        ->  XN Sort  (cost=1002815366604.92..1002815366606.36 rows=576 width=27)
              Sort Key: sum(sales.pricepaid)
              ->  XN HashAggregate  (cost=2815366577.07..2815366578.51 rows=576 width=27)
                    ->  XN Hash Join DS_BCAST_INNER  (cost=109.98..2815365714.80 rows=172456 width=27)
                          Hash Cond: ("outer".eventid = "inner".eventid)
                          ->  XN Seq Scan on sales  (cost=0.00..1724.56 rows=172456 width=14)
                          ->  XN Hash  (cost=87.98..87.98 rows=8798 width=21)
                                ->  XN Seq Scan on event  (cost=0.00..87.98 rows=8798 width=21)
```

Auto WLM to the rescue!

Auto WLM with query priority



Writer
High priority
for fresh data



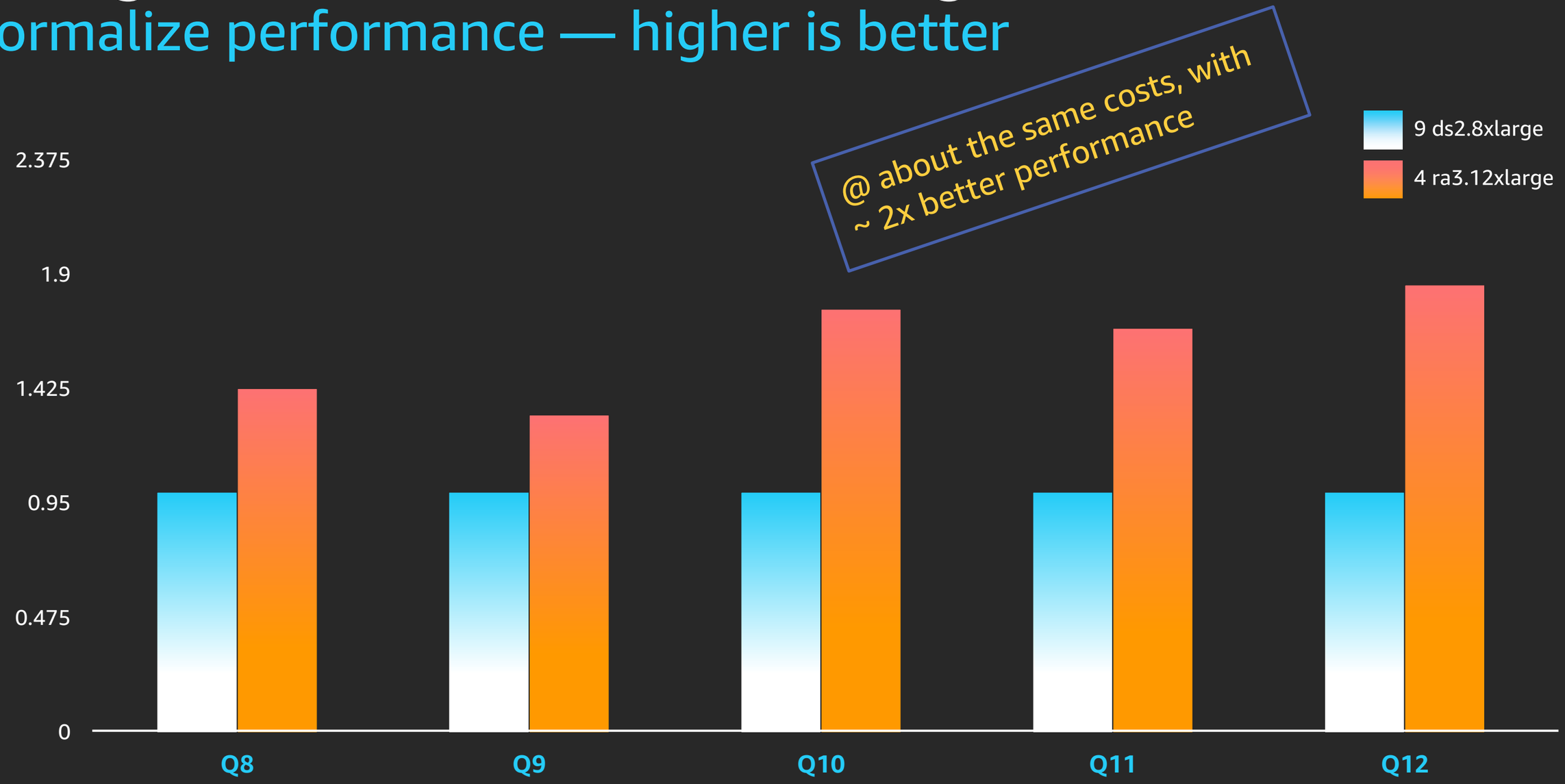
Reports
Normal priority
Fast: <10min



Adhoc query
Low priority
>30min

Next generation data warehousing

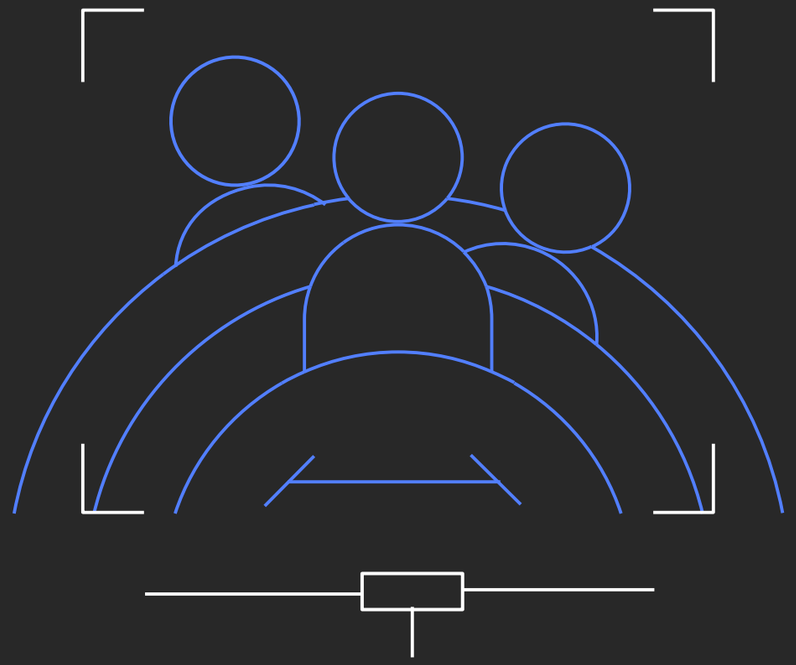
Normalize performance — higher is better



Looking forward

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Contribution to Redshift community



Redshift and Spark community

Leading a community fork of spark-redshift (https://github.com/spark-redshift-community)

Redshift is a popular data source for machine learning developers

Compatible with Spark 2.4.x

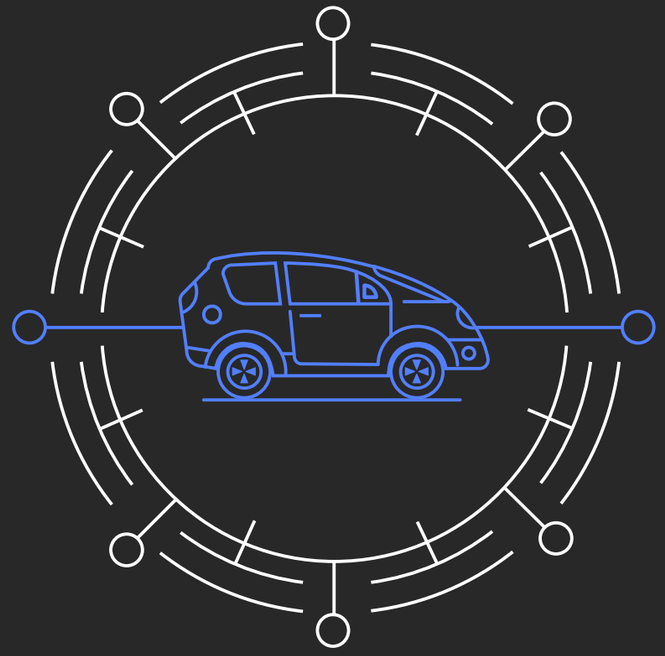
Yelp has done the initial set of patches to make it 2.4 compatible

Looking forward to upcoming Spark 3.x compatibility

Community driven

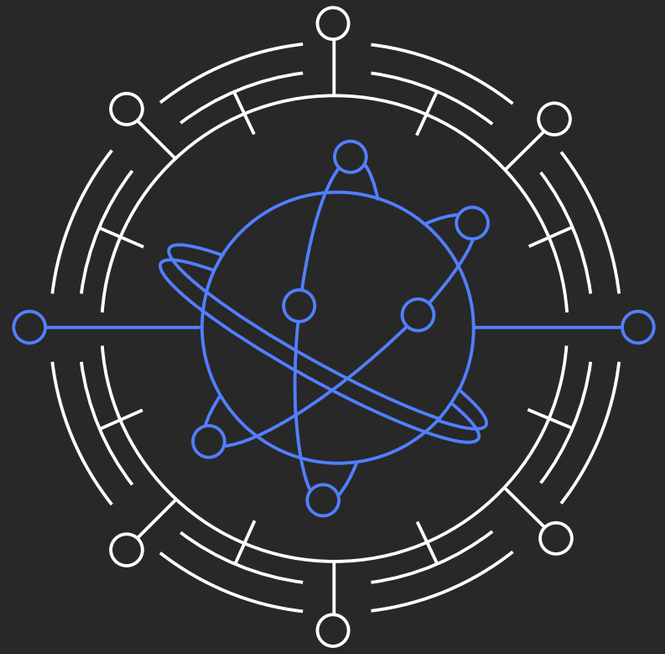
We are seeking others' input to make this a vibrant community

Yelp data ecosystem in 2020 and beyond



Self-driving data systems, more machine learning and optimization

Yelp data ecosystem in 2020 and beyond



Data provenance across all data systems


We’re hiring!
www.yelp.com/careers/

Thank you!

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws



Please complete the session survey in the mobile app.

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.