

## Redshift Slides

03:55 PM

# Amazon Redshift

Fully-managed, petabyte-scale data warehouse

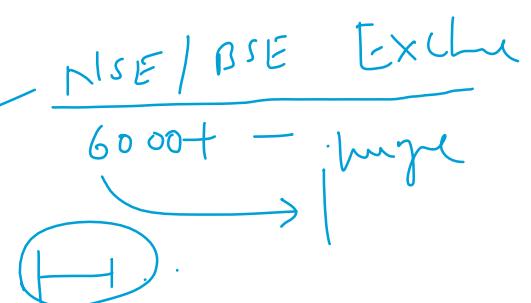


## What is Redshift?

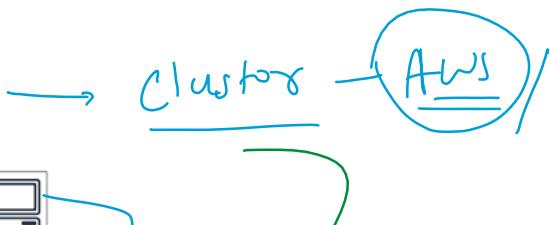
- Fully-managed, petabyte scale data warehouse service
- 10X better performance than other DW's
  - Via machine learning, massively parallel query execution, columnar storage
- Designed for OLAP, not OLTP
- Cost effective
- SQL, ODBC, JDBC interfaces
- Scale up or down on demand
- Built-in replication & backups
- Monitoring via CloudWatch / CloudTrail

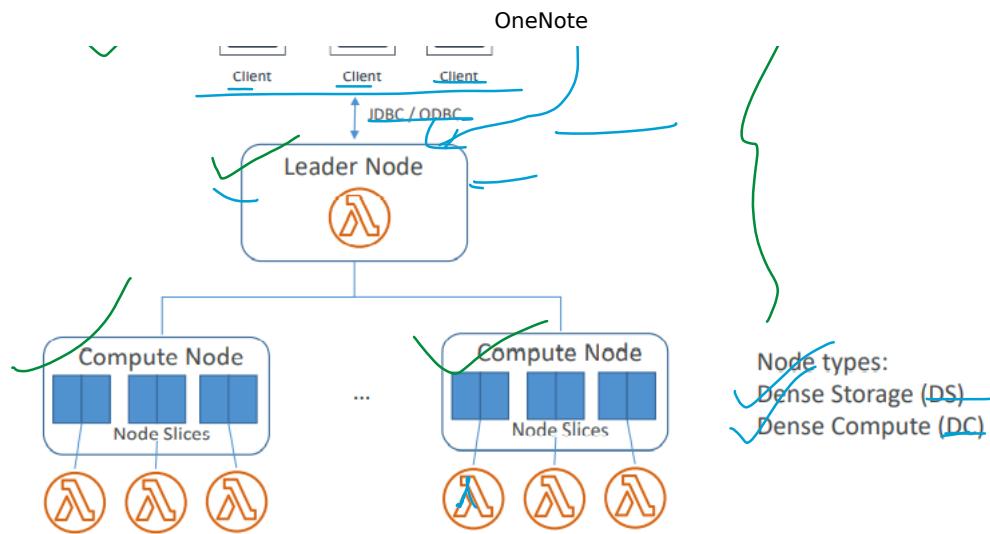
## Redshift Use-Cases

- Accelerate analytics workloads
- Unified data warehouse & data lake
- Data warehouse modernization
- Analyze global sales data
- Store historical stock trade data
- Analyze ad impressions & clicks
- Aggregate gaming data
- Analyze social trends



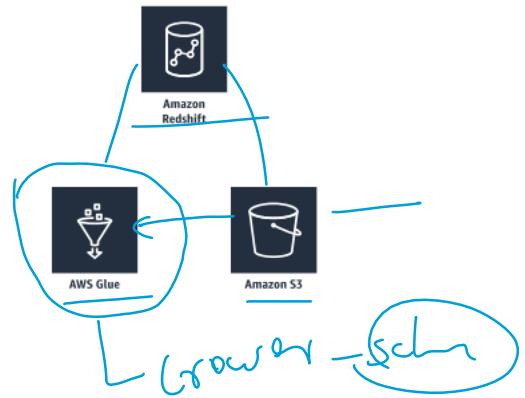
## Redshift architecture





## ✓ Redshift Spectrum

- Query exabytes of unstructured data in S3 without loading
- Limitless concurrency
- Horizontal scaling
- Separate storage & compute resources
- Wide variety of data formats
- Support of Gzip and Snappy compression



## ✓ Redshift Performance

- Massively Parallel Processing (MPP)
- Columnar Data Storage
- Column Compression



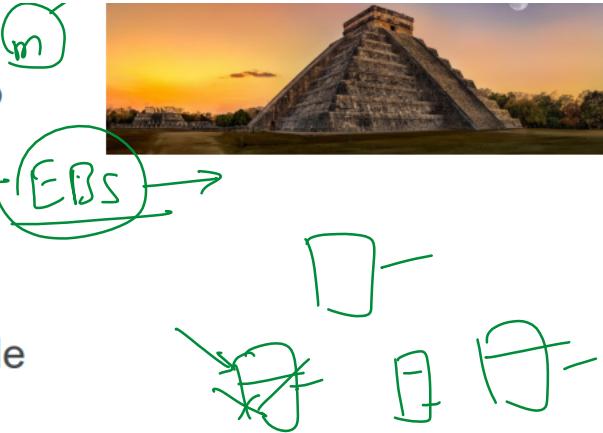
## Redshift Durability

ENG

- Replication within cluster



- ✓ Backup to S3
  - Asynchronously replicated to another region
- ✓ Automated snapshots
- ✓ Failed drives / nodes automatically replaced
- ✓ However – limited to a single availability zone (AZ)



## Scaling Redshift

- ✓ Vertical and horizontal scaling on demand
- ✓ During scaling:
  - A new cluster is created while your old one remains available for reads
  - CNAME is flipped to new cluster (a few minutes of downtime) →
  - Data moved in parallel to new compute nodes

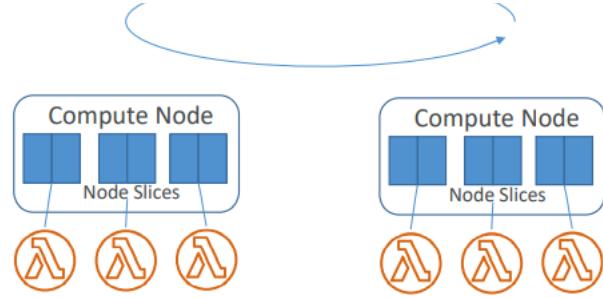


## Redshift Distribution Styles

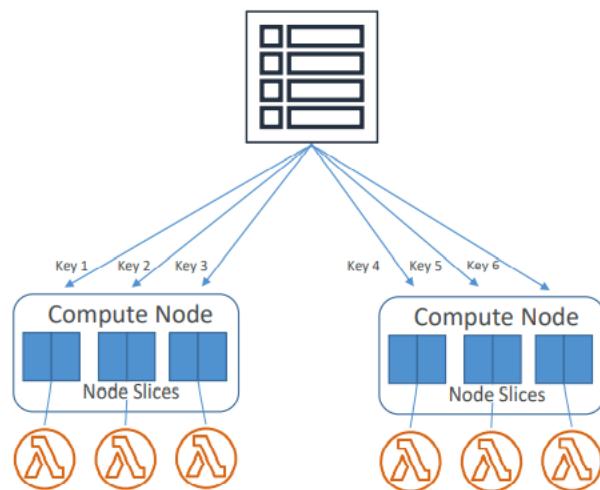
- AUTO
  - Redshift figures it out based on size of data
- EVEN
  - Rows distributed across slices in round-robin
- KEY
  - Rows distributed based on one column
- ALL
  - Entire table is copied to every node

## EVEN distribution

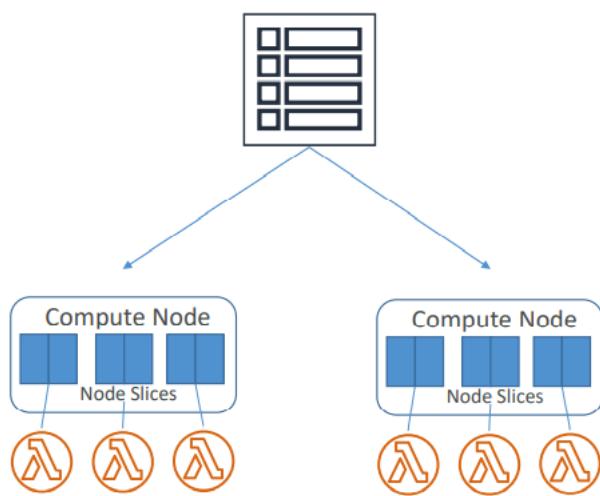




## KEY distribution



## ALL distribution



## Redshift Sort Keys

- Rows are stored on disk in sorted order based on the column you designate as a sort key
- Like an index
- Makes for fast range queries





- Choosing a sort key
  - Recency? Filtering? Joins?
- Single vs. Compound vs Interleaved sort keys

## Sort Keys: Single Column

Date	Genre	Movie
3/18/2019	Comedy	Monty Python and the Holy Grail
3/18/2019	Adventure	Indiana Jones and the Temple of Doom
3/18/2019	Drama	Interstellar
3/18/2019	Drama	The Dark Knight
3/19/2019	Fantasy	The Lord of the Rings
3/19/2019	Drama	12 Angry Men
3/19/2019	Adventure	Inception

## Sort Keys: Compound

Date	Genre	Movie
3/18/2019	Adventure	Indiana Jones and the Temple of Doom
3/18/2019	Comedy	Monty Python and the Holy Grail
3/18/2019	Drama	Interstellar
3/18/2019	Drama	The Dark Knight
3/19/2019	Adventure	Inception
3/19/2019	Drama	12 Angry Men
3/19/2019	Fantasy	The Lord of the Rings

## Sort Keys: Interleaved

Date	Genre	Movie	Date	Genre	Movie
3/18/2019	Adventure	Indiana Jones and the Temple of Doom	3/19/2019	Drama	12 Angry Men
3/18/2019	Comedy	Monty Python and the Holy Grail	3/19/2019	Adventure	Inception
3/18/2019	Drama	Interstellar	3/18/2019	Adventure	Indiana Jones and the Temple of Doom
3/18/2019	Drama	The Dark Knight	3/18/2019	Drama	Interstellar
3/19/2019	Adventure	Inception	3/18/2019	Comedy	Monty Python and the Holy Grail
3/19/2019	Drama	12 Angry Men	3/18/2019	Drama	The Dark Knight
3/19/2019	Fantasy	The Lord of the Rings	3/19/2019	Fantasy	The Lord of the Rings

Date	Genre	Movie
3/18/2019	Adventure	Indiana Jones and the Temple of Doom
3/19/2019	Adventure	Inception
3/18/2019	Comedy	Monty Python and the Holy Grail
3/18/2019	Drama	Interstellar
3/18/2019	Drama	The Dark Knight
3/19/2019	Drama	12 Angry Men
3/19/2019	Fantasy	The Lord of the Rings

# Importing / Exporting data

- COPY command
  - Parallelized; efficient
  - From S3, EMR, DynamoDB, remote hosts
  - S3 requires a manifest file and IAM role
- UNLOAD command
  - Unload from a table into files in S3
- Enhanced VPC routing



## COPY command: More depth

- Use COPY to load large amounts of data from outside of Redshift
- If your data is already in Redshift in another table,
  - Use INSERT INTO ...SELECT
  - Or CREATE TABLE AS
- COPY can decrypt data as it is loaded from S3
  - Hardware-accelerated SSL used to keep it fast
- Gzip, Izop, and bzip2 compression supported to speed it up further
- Automatic compression option
  - Analyzes data being loaded and figures out optimal compression scheme for storing it
- Special case: narrow tables (lots of rows, few columns)
  - Load with a single COPY transaction if possible
  - Otherwise hidden metadata columns consume too much space



## Integration with other services

- ~~S3~~
- ~~DynamoDB~~
- ~~EMR / EC2~~
- ~~Data Pipeline~~
- ~~Database Migration Service~~



Amazon S3



AWS Database Migration Service



Amazon DynamoDB



Amazon EMR

## Redshift Workload Management (WLM)

- Prioritize short, fast queries vs. long, slow queries
- Query queues
- Via console, CLI, or API

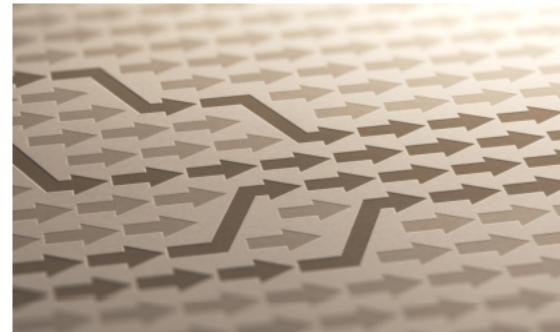
## Concurrency Scaling

- Automatically adds cluster capacity to handle increase in concurrent **read** queries
- Support virtually unlimited concurrent users & queries
- WLM queues manage which queries are sent to the concurrency scaling cluster



## Automatic Workload Management

- Creates up to 8 queues
- Default 5 queues with even memory allocation
- Large queries (ie big hash joins) -> concurrency lowered
- Small queries (ie inserts, scans, aggregations) -> concurrency raised
- Configuring query queues
  - Priority
  - Concurrency scaling mode
  - User groups
  - Query groups
  - Query monitoring rules



## Manual Workload Management

- One default queue with concurrency level of 5 (5 queries at once)
- Superuser queue with concurrency level 1
- Define up to 8 queues, up to concurrency level 50
  - Each can have defined concurrency scaling mode, concurrency level, user groups, query groups, memory, timeout, query monitoring rules
  - Can also enable query queue hopping

- Timed out queries “hop” to next queue to try again

## Short Query Acceleration (SQA)

- Prioritize short-running queries over longer-running ones
- Short queries run in a dedicated space, won’t wait in queue behind long queries
- Can be used in place of WLM queues for short queries
- Works with:
  - CREATE TABLE AS (CTAS)
  - Read-only queries (SELECT statements)
- Uses machine learning to predict a query’s execution time
- Can configure how many seconds is “short”

## ✓ Resizing Redshift Clusters

- Elastic resize
  - Quickly add or remove nodes of same type
    - (It \*can\* change node types, but not without dropping connections – it creates a whole new cluster)
  - Cluster is down for a few minutes
  - Tries to keep connections open across the downtime
  - Limited to doubling or halving for some dc2 and r4s node types.
- Classic resize
  - Change node type and/or number of nodes
  - Cluster is read-only for hours to days
- Snapshot, restore, resize
  - Used to keep cluster available during a classic resize
  - Copy cluster, resize new cluster



## ✓ VACUUM command

- Recovers space from deleted rows
- VACUUM FULL
- VACUUM DELETE ONLY
- VACUUM SORT ONLY
- VACUUM REINDEX



# New Redshift features for 2020+

- RA3 nodes with managed storage
  - Enable independent scaling of compute and storage
- Redshift data lake export
  - Unload Redshift query to S3 in Apache Parquet format
  - Parquet is 2x faster to unload and consumes up to 6X less storage
  - Compatible with Redshift Spectrum, Athena, EMR, SageMaker
  - Automatically partitioned

## Redshift anti-patterns

- Small data sets
  - Use RDS instead
- OLTP
  - Use RDS or DynamoDB instead
- Unstructured data
  - ETL first with EMR etc.
- BLOB data
  - Store references to large binary files in S3, not the files themselves.



## Redshift security concerns

- Using a Hardware Security Module (HSM)
  - Must use a client and server certificate to configure a trusted connection between Redshift and the HSM
  - If migrating an unencrypted cluster to an HSM-encrypted cluster, you must create the new encrypted cluster and then move data to it.
- Defining access privileges for user or group
  - Use the GRANT or REVOKE commands in SQL
  - Example: grant select on table foo to bob;