Populating Redshift

Russ Thomas
DATA ARCHITECT

@sqljudo www.sqljudo.com

Module Prerequisites

Redshift Cluster with empty table schema

Variances in Distribution Keys / Sort Keys is ok

Access to download content for the course

The Copy Command

Logical approach

Data formats

Command syntax

Source locations

Integrating Redshift into Your ETL

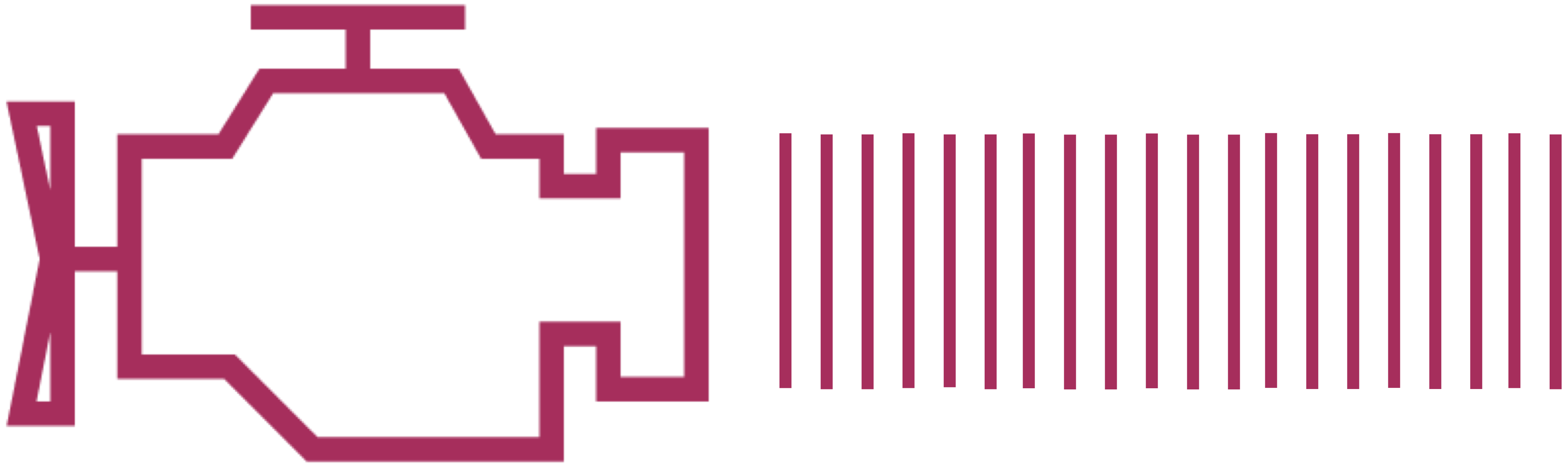
Leveraging Copy

Staging Tables

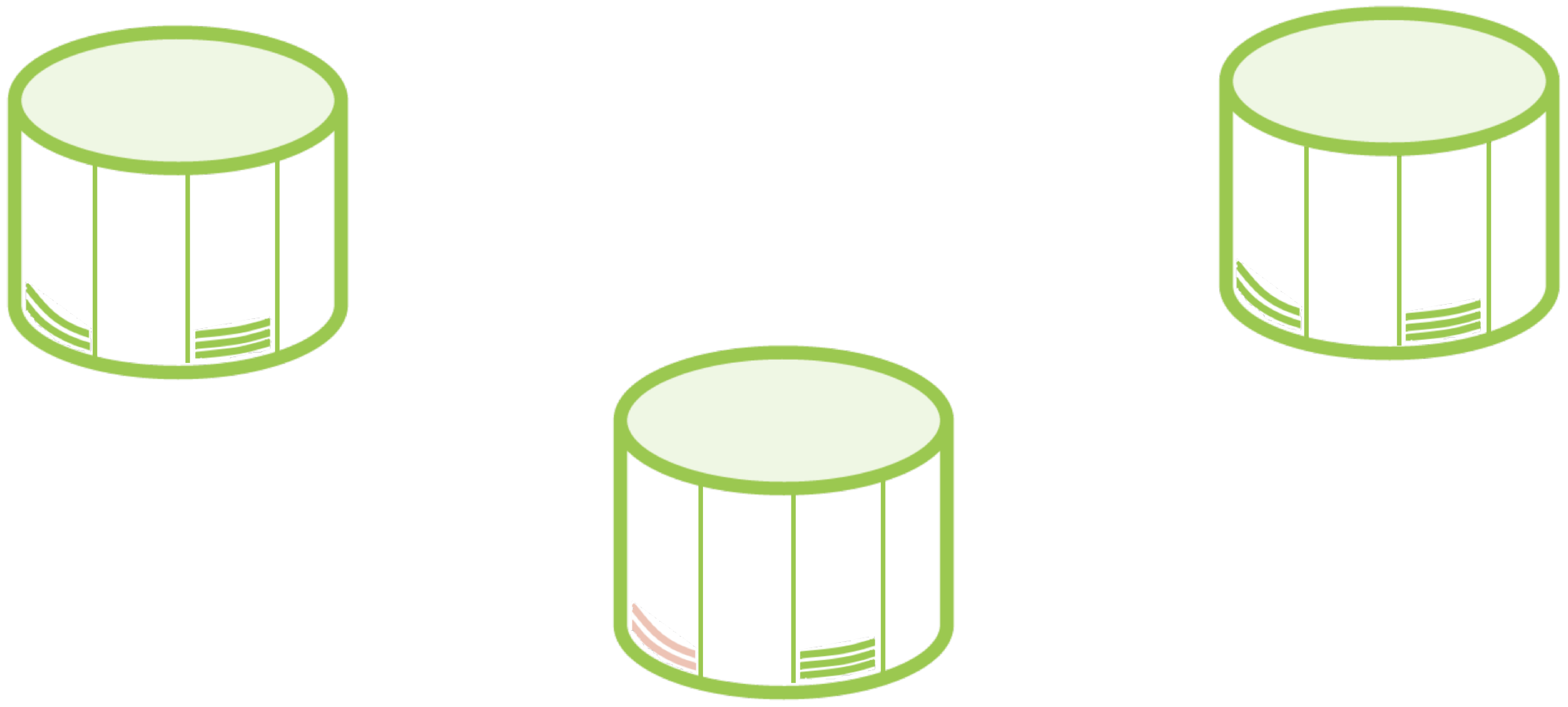
Transformations

Automated Loads

Leveraging Data Streaming



Loading into Nodes and Node Slices



ETL Approaches to Consider

Upserts

Overhead of major update / insert activity

Traditional ETL approach

Let each of your tools do what they do best!

A scripting approach does not preclude:

- Parallel tasks
- Multiple files
- Advanced approaches through SDK

The Copy command is the de facto standard for bulk loading data and ongoing population of your cluster

Overhead of Major Update / Insert Activity

Updates act as deletes followed by an insert

Inserts are not automatically sorted per sort key

Deletes are marked as deleted but continue to use space on disk

Vacuum Process

- Resorts rows per the sort key
- Reclaims space from deleted rows
- Includes option to re-index
- Substantially different from Postgres vacuum process

Copy Command

- Sources include:
  - S3 storage
  - External SSH sources
  - AWS databases such as DynamoDB
  - Kinesis Firehose

- Authorization
  - IAM Roles
  - Access Keys

- Options include:
  - GZIP and other zip technologies
  - Multiple formats and delimiters
  - Manifest files for parallel loads