



Running a Proof-of-Concept (POC) on Amazon Redshift

Test, prove out, and adopt features



Table of contents

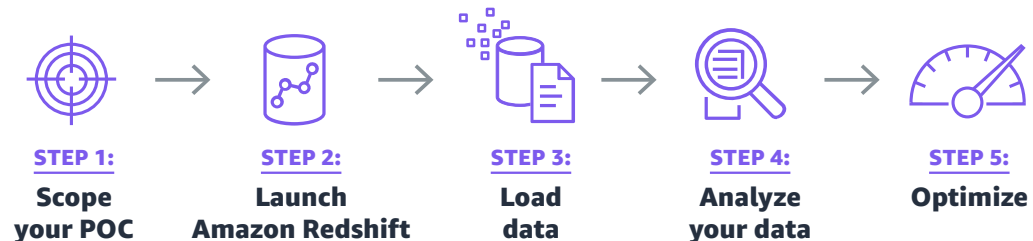
Running a Proof-of-Concept (POC) on Amazon Redshift	3
Step 1: Scope your POC	4
Step 2: Set up Amazon Redshift Serverless	5
Step 3: Load Data	6
Option 1: Load Sample Data	6
Option 2: Bring your Own Data	6
Step 4: Analyze Your Data	8
Option 1: Query Using Query Editor	8
Option 2: Run a Load Test	9
Step 5: Optimize	11
Option 1: Test Drive	11
\$300 Redshift Serverless credit trial here	12

Running a Proof-of-Concept (POC) on Amazon Redshift

Tens of thousands of customers use [Amazon Redshift](#) every day to modernize their data analytics workloads and deliver insights for their businesses, AI/ML/Gen-AI applications, complex data processing jobs, powerful dashboards, and more. Amazon Redshift is a popular cloud data warehouse, offering a fully managed cloud-based service that seamlessly integrates with an organization's Amazon Simple Storage Service (Amazon S3) data lake, real-time streams, machine learning (ML) workflows, transactional workflows, and much more—all while providing up to 6x better price-performance than other cloud data warehouses.

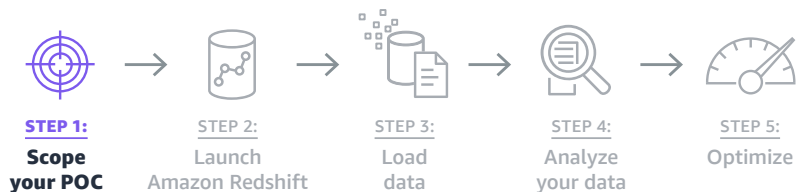
Amazon Redshift Serverless makes it easy for you to run and scale analytics without managing data warehouse infrastructure. The serverless architecture intelligently and efficiently scales underlying compute resources to deliver high performance for even the most demanding and unpredictable workloads. This guide is intended to make the process of doing a proof-of-concept (POC) on Amazon Redshift Serverless easy, organized, and aligned to your business goals in 5 simple steps. It will help you set goals for your POC, as well as take advantage of tools that can automate provisioning and configuration of services for your POC.

Test, prove out, and adopt features ranging from best-in-class security capabilities, elastic scaling, easy integration and ingestion, and flexible decentralized data architecture options.



For steps 1-5, consider bringing in your own data. If that is not possible you can use sample data. Timeline for the entire POC is 2+ weeks depending on complexity and volume of data and tests.

- Best-in-class security
- Elastic scaling
- Easy integration & ingestion
- Flexible decentralized data architecture options.



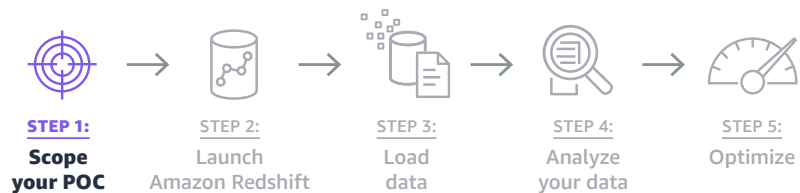
STEP 1: Scope your POC

- 1.1 Identify your business and functional requirements,** then work backwards. Common examples are: *faster performance, lower costs, test a new workload or feature, or comparison between Amazon Redshift and another data warehouse.*
- 1.2 Set specific targets** which will become the success criteria for the POC. For example, from ‘faster performance’, come up with a list of the top 5 processes you wish to accelerate, and include the current run times along with your required runtime. These can be reports, queries, ETL processes, data ingestion, or whatever your current pain points are.
- 1.3 Bring Your Own or Load sample data**
To test your own data, come up with the minimum viable list of data artifacts which will be required to test for your success criteria. For example, if your current data warehouse has 200 tables, but the reports you want to test only need 20, your POC can be executed faster by using only the smaller subset of tables.

Your notes:

Your notes:

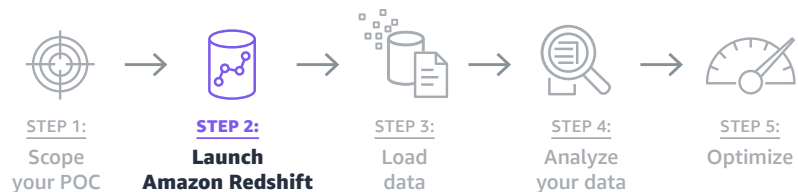
Your notes:



- If you don't have your own datasets ready, you can still get started with doing a POC on Amazon Redshift by using the industry-standard benchmark datasets such as [TPC-DS](#) or [TPC-H](#) and run sample benchmarking queries to harness the power of Amazon Redshift. These datasets can be easily accessed from within your Amazon Redshift warehouse once it is created within a few clicks. For detailed instructions on how to access these datasets and sample queries, please navigate to the section **'Load Data'**.

1.4 Identify the specific scope & artifacts needed to run the tests. What datasets do you need to **migrate or continuously ingest into Amazon Redshift**, and what queries / processes are needed to **run the tests** to measure against the success criteria? There are following two ways to do this:

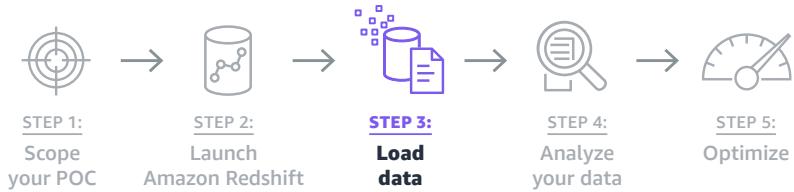
Your notes:



STEP 2: Set up Amazon Redshift Serverless

[Amazon Redshift Serverless](#) is a cloud data warehouse without any infrastructure maintenance and overhead hassles. Amazon Redshift Serverless is more intelligent than ever before with AI-driven scaling and optimizations, with the capability to learn customer workload patterns based on dimensions such as query complexity, frequency, seasonality, and continuously adjusting resources throughout the day to proactively scale to meet your price-performance objectives.

- The first time you use Amazon Redshift Serverless, the console will talk you through the steps required to launch your warehouse.
- You will also receive a \$300.00 credit towards your Serverless usage in this account.
- Please follow the steps in [the Amazon Redshift Serverless Getting Started Guide](#) to create a data warehouse with Amazon Redshift Serverless.
 - o If you do not have a dataset you would like to load, the guide also contains steps on how to load a sample data set.
- If you have previously launched Amazon Redshift Serverless in this account, please follow the steps on how to [create a workgroup with a namespace](#).



STEP 3: Load Data

Option 1: Load Sample Data

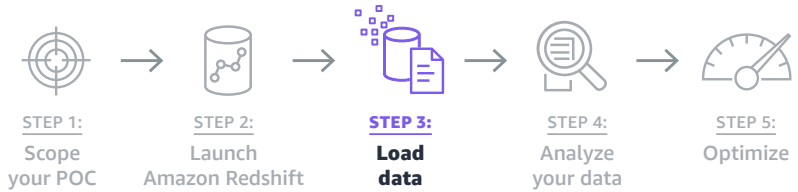
If you don't have your own datasets ready, you can opt to load sample data available in Amazon Redshift. Once you connect to your cluster using Query Editor, simply choose the icon associated with the sample data you want to load. The Query Editor then loads the data into a schema in database `sample_data_dev` and creates a folder of saved notebooks in your **Notebooks** folder. See this [link](#) for description of the sample datasets available.

Option 2: Bring your Own Data

To bring your own data, you can **upload a simple CSV** file, ingest **semi-structured data from S3**, or **stream data** directly into Amazon Redshift.

To proceed, simply choose the method you prefer below, and follow those detailed steps.





OPTION 1: Upload a Local File

Amazon Redshift Query Editor is a web-based analyst workbench for you to securely explore, share, and collaborate on data with your teams using SQL within a common notebook interface. For quick ingestion and analysis, you can use Amazon Redshift Query Editor to easily load data files from your local desktop. It has the capability to process files in various formats such as CSV, JSON, AVRO, PARQUET, ORC, and more. To enable your users to load data from a local desktop using Query Editor, as an administrator, you have to specify a common S3 bucket, and the user account must be [configured with proper permissions](#). You can follow [this section](#) of the [blog](#) for step by step guidance.

OPTION 2: Load Your Streaming Data

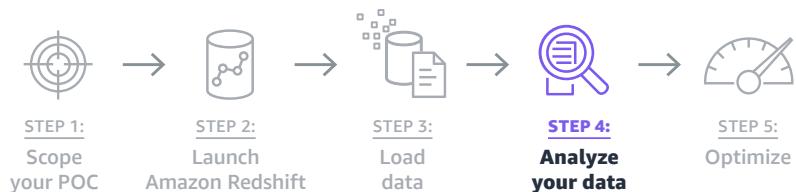
Streaming ingestion provides low-latency, high-speed ingestion of stream data from [Amazon Kinesis Data Streams](#) and [Amazon Managed Streaming for Apache Kafka](#) into Amazon Redshift. Amazon Redshift streaming ingestion uses a materialized view, which is updated directly from the stream utilizing [Auto Refresh](#). The materialized view maps to the stream data source. You can perform filtering and aggregations on the stream data as part of the materialized-view definition. You can follow documentation for step by step guidance to [load from a Kinesis Data Stream](#) or an [MSK topic](#).

OPTION 3A: Load from S3 Files

To load data from an S3 bucket into Amazon Redshift, begin by utilizing the [COPY command](#), specifying the source S3 location and target Amazon Redshift table. Ensure that the IAM roles and permissions are properly configured to allow Amazon Redshift access to the designated S3 bucket. Follow [this tutorial](#) for step-by-step guidance. You can also choose “Load Data” option in Query Editor to [directly load data from your S3 bucket](#).

OPTION 3B: Continuous Data Ingestion (AutoCopy)

[Auto-copy \(in Preview\)](#) is an extension of the [COPY](#) command and automates continuous data loading from Amazon S3 buckets. When you create a copy job, Amazon Redshift detects when new Amazon S3 files are created in a specified path, and then loads them automatically without your intervention. Amazon Redshift keeps track of the loaded files to verify that they are loaded only one time.



STEP 4: Analyze Your Data

After creating your cluster, and loading your data, you can immediately run queries by using the Query Editor on the Amazon Redshift console. You can use this to test query functionality or query performance against your own datasets.

Alternatively, if you want to run a load test as part of your POC, you can do this easily by following the steps described below, to install and run Apache Jmeter.



Query Using Query Editor

Access Query Editor from the [Amazon Redshift console](#).

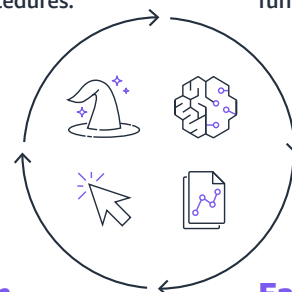
Refer to this [blog](#) for a complete easy guide on how to configure, connect, and run queries with Query Editor.

Visual Wizard

Browse database objects including tables, views, and stored procedures.

Generative SQL

Increase productivity with text to SQL functionality



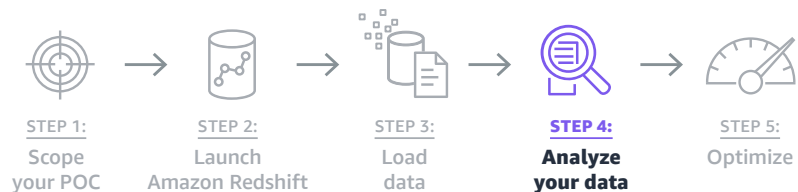
Navigation

Browse database objects including tables, views, and stored procedures

Easy collaboration

Collaborate with other SQL users using a common interface, with automatic version management





Run a Load Test

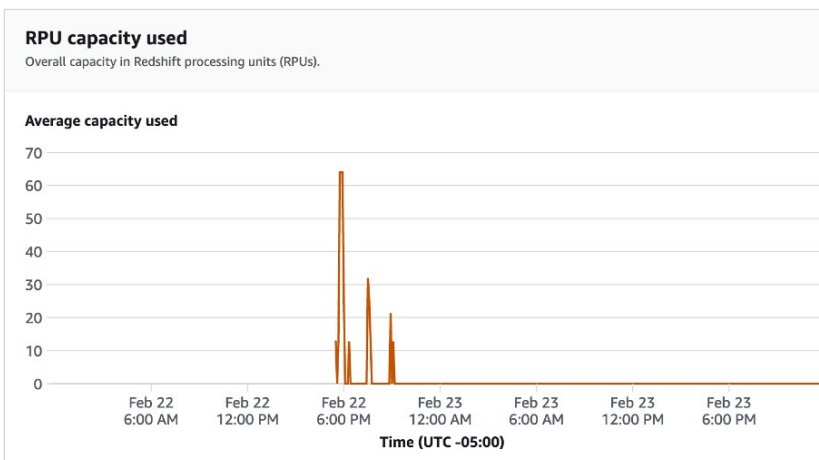
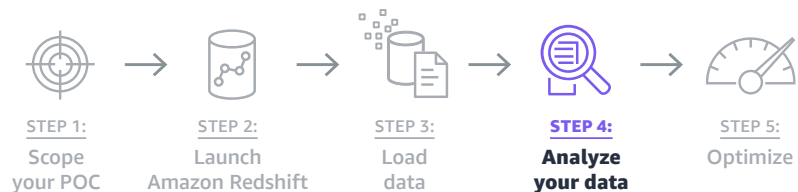
Run a Load test on Amazon Redshift using JMeter

To perform a load test to simulate “N” users submitting queries concurrently to Amazon Redshift, you can use [Apache JMeter](#), an open-source Java based tool.

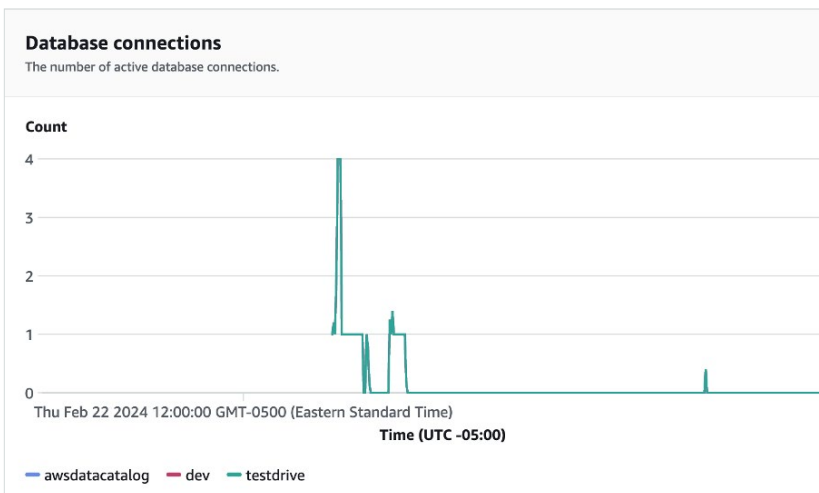
To install and configure Apache Jmeter to run against your cluster, please follow the instructions in this [blog](#). It uses the [AWS Analytics Automation toolkit \(AAA\)](#), an open-source utility for dynamically deploying Amazon Redshift solutions, to automatically launch these resources. If you have loaded your own data into Amazon Redshift, be sure to perform the Step #5 – Customize SQL option, to make sure you supply the appropriate SQL statements you would like to test against your tables. Test each of these SQL statements once using Query Editor (above) to make sure they execute without errors.

Once you complete customizing your SQL statements and finalizing your test plan, save and run your test plan against your cluster. To monitor the progress of your test, use the **Query history** tab in Query and Database monitoring in the [Amazon Redshift Serverless console](#).

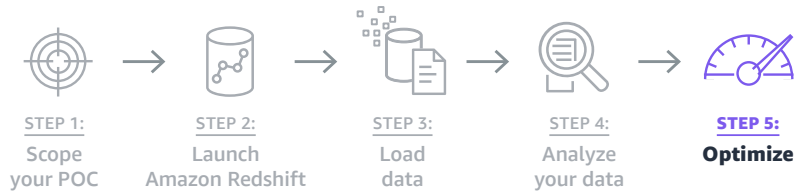
For cluster performance metrics, you can navigate to **Database performance** tab in **Query and Database monitoring** in the [Amazon Redshift Serverless console](#), to monitor metrics such as **Database Connections, RPU capacity used as shown below.**



Use this graph to monitor RPU capacity being used as well as observe how Redshift automatically scales to meet concurrent workload demands while the load test is running on your cluster.



Database Connections is another useful metric to monitor while running the load test to see how your cluster is handling numerous concurrent connections at a given time to meet the increasing workload demands.



STEP 5: Optimize

Amazon Redshift empowers tens of thousands of customers to process exabytes of data every day and power their analytics workloads by offering a variety of configurations and features to support individual use cases. When choosing between these options, customers are looking for tools that help them determine the most optimal Data Warehouse configuration to support their Amazon Redshift workload.

Test Drive

- You can use [Test Drive](#) to automatically replay your existing Amazon Redshift workload on potential configurations and analyze the corresponding outputs to evaluate the optimal target to migrate your workload to.
- Refer to [this blog](#) on using Test Drive to evaluate different Amazon Redshift configurations.

Amazon Redshift empowers thousands of companies to deploy analytics with confidence that your data is protected no matter where in the world you operate or how highly regulated your industries are. Amazon Redshift supports industry-leading security with built-in identity management and federation for single sign-on (SSO), multi-factor authentication, granular access control Amazon Virtual Private Cloud (Amazon VPC), and faster cluster resize. With Amazon Redshift, your data is encrypted in transit and at rest. All Amazon Redshift security features are offered at no additional cost to satisfy the most demanding security, privacy, and compliance requirements. AWS supports more security standards and compliance certifications than any other provider, including ISO 27001, SOC, HIPAA/HITECH, and FedRAMP.

Your notes:

To get started with your Amazon Redshift Serverless trial, take advantage of the [\\$300 Redshift Serverless credit trial here.](#)