

# **QLIKVIEW INTEGRATION WITH AMAZON REDSHIFT**

---

**John Park**

**Partner Engineering**

June 2014

## Contents

---

Introduction.....	3
About Amazon Web Services (AWS) .....	3
About Amazon Redshift.....	3
QlikView on AWS.....	4
Recommended architecture .....	5
Getting Started with Amazon .....	7
Connecting QlikView and Redshift .....	8
Which drivers? .....	8
Amazon Documentation I: Redshift Drivers.....	8
ODBC Driver: Postgres ODBC Download.....	8
Configuring the connector .....	8
Tuning QlikView and Redshift for performance .....	10
Big Data, QlikView, and Redshift .....	11

## Introduction

---

This paper is a hands-on guide that explains how to run QlikView in the cloud with Amazon Redshift. In order to provide an adequate context, this paper provides background information on Amazon Web Services, Redshift, and QlikView. The main section of the whitepaper is a step-by-step guide on how to get you started.

## About Amazon Web Services (AWS)

---

Amazon Web Services is a collection of web services that collectively make up a cloud computing platform.

Compared to buying and building a physical server farm, the three key benefits of Amazon's cloud platform are:

- Ease of use – a platform that can be constructed in hours, unlike a physical server which may take days
- Flexibility – capacity can be grown or shrunk on demand
- Cost matching – the cost of a platform can be easily matched to the benefits gained

Under the AWS banner, Amazon offers a number of services, including:

- DynamoDB – NoSQL database
- EC2 – cloud-based servers running software
- RDS – relational database service
- Redshift – data warehouse as a service
- S3 – scalable cloud storage
- EMR – elastic map reduce(Hadoop as Service)

## About Amazon Redshift

---

From the Amazon website: “*Amazon Redshift is a fast, fully managed, petabyte-scale data warehouse service that makes it simple and cost-effective to efficiently analyze all your data using your existing business intelligence tools. It is optimized for datasets ranging from a few hundred gigabytes to a petabyte or more and costs less than \$1,000 per terabyte per year, a tenth the cost of most traditional data warehousing solutions.*”

<http://aws.amazon.com/redshift/>

Here are some of the benefits of using Redshift as opposed to physical hardware:

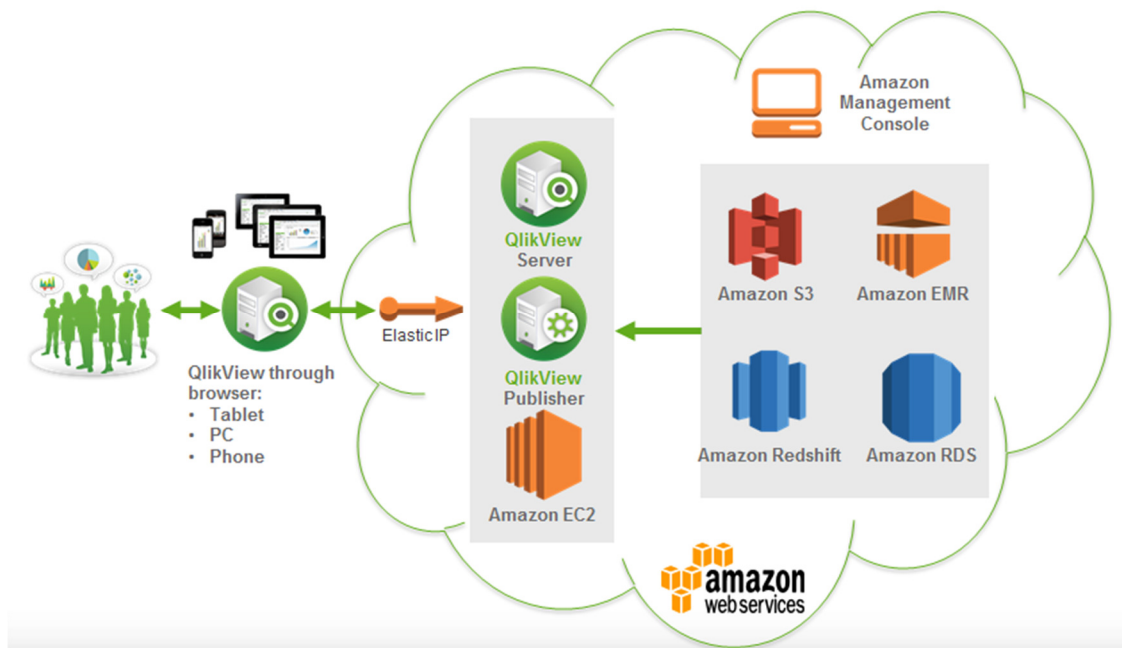
- Data Warehouse as a Service (DaaS) - no physical hardware needed and a pay-as-you-go model
- Fast, effective and low-cost data warehouse – columnar database built for analytical workloads
- Easy to use – one click deployment, easy to back up, easy to manage
- Scalability – allows resizing and clustering
- Fully managed – hardware and software upgrades are all managed by AWS

## QlikView on AWS

Since 2011, Qlik and AWS have been providing cloud-based business intelligence using cloud-based data. QlikView works well as a service and there are large number of Organizations and Partners that have deployed cloud-based solutions. Some of Qlik's partners base entire business lines around QlikView deployed in the cloud.

Ever since it was first released in 2013, Amazon Redshift has been adding the flexibility of a massively scalable cloud-based data warehouse to QlikView's data analysis capabilities in order to provide world-class solutions.

The diagram below provides an overview of how QlikView works with Amazon's web services.



Amazon released Redshift in 2013, adding the flexibility of a massively scalable cloud-based database to QlikView's data analysis capabilities.

Why use QlikView and Amazon Redshift together?

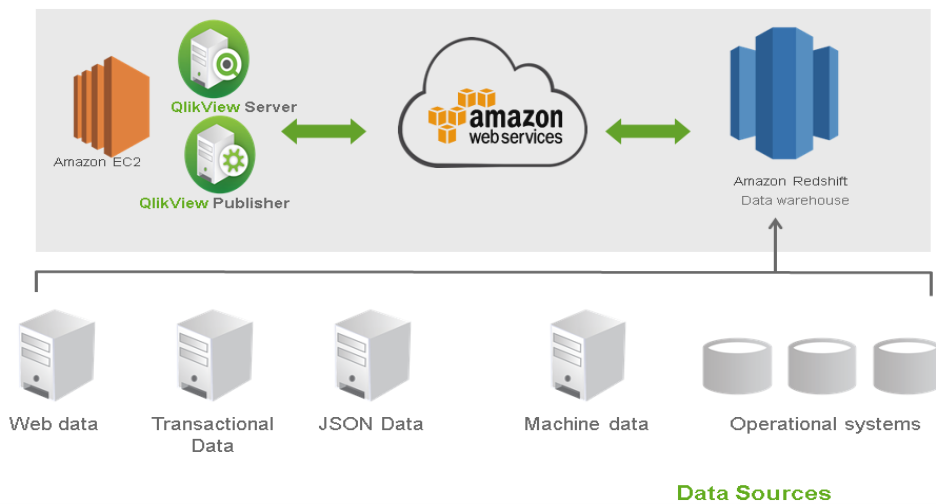
- Redshift is certified for QlikView 11.20 SR5 and subsequent versions.
  - Redshift was certified by the Qlik Partner Engineering team in the 2nd half of 2013
- QlikView Server has been certified for a few years in a row to run AWS EC2 servers
  - Since 2011, all instances of EC2 running Microsoft Windows Server have been tested with QlikView
- Redshift is a preferred Big Data Platform for QlikView Direct Discovery (in-database processing)
  - QlikView 11.20 SR5 has been tested by extracting 100 million rows into QlikView's associative in-memory data store in the cloud
  - Using QlikView's Direct Discovery platform with data sourced from Redshift, QlikView 11.20 SR5 has been tested with 1 billion rows of data
  - QlikView has shown consistent performance in running inside AWS Environment
- Many System Integrator Partners are experienced in deploying QlikView and AWS.

## Recommended architecture

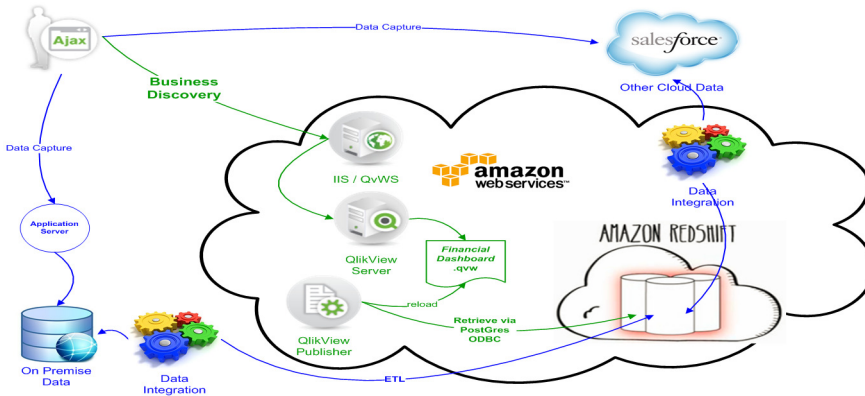
The following diagrams depict the certified and recommended Redshift/QlikView architecture (QlikView Server and QlikView Publisher running within an Amazon EC2 instance).

### QlikView and Redshift in AWS based BI Architecture

## The Qlikview+Reshift+AWS BI architecture

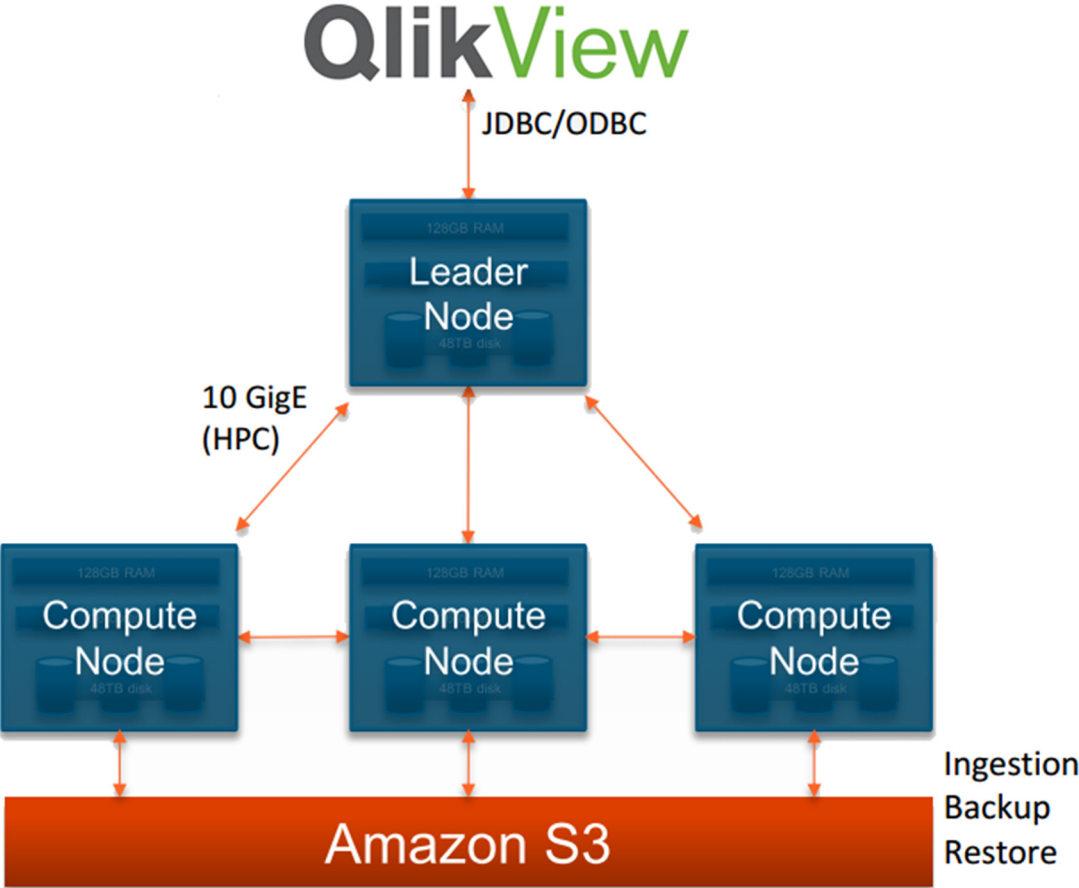


### QlikView Data Integration Workflow with Redshift



QlikView running in a different location than Redshift is not recommended. This is because of potential bandwidth variability issues that can degrade performance. However, if such configuration cannot be avoided then it is recommended to use AWS Direct Connect. AWS Direct Connect provides dedicated bandwidth that removes variability to ensure a positive end user experience.

QlikView accesses data through the Redshift leader node via ODBC data connectors (see the figure below).



Due to distribution of data inside AWS Redshift, users should follow Redshift best practices for loading data to achieve optimal performance.

# Getting Started with Amazon Redshift

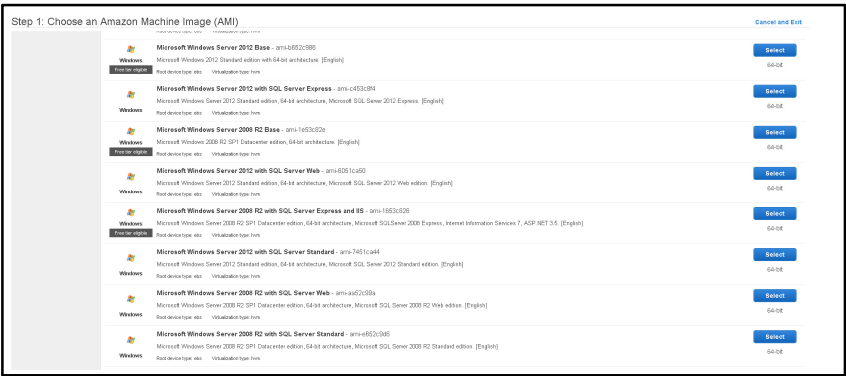
QlikView is a fully supported platform on the AWS platform.

The following steps describe how to get started.

1. Creation of a Microsoft Windows AMI (Amazon Machine Image) on an Elastic Compute Cloud (EC2) instance. To minimize latency, you should choose the region closest to you. The Redshift Cluster and QlikView Server running in the cloud should reside in the same region.

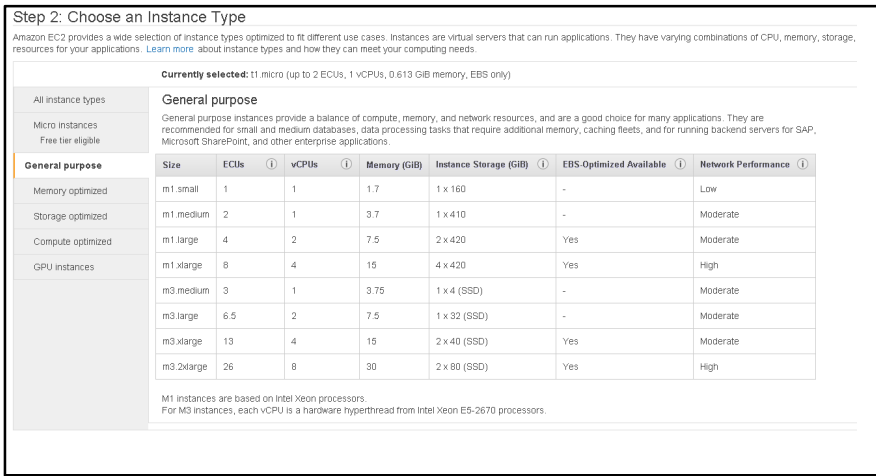
\*QlikView Server requires the Microsoft Windows Server 2008 AMI with IIS

Figure 1. Amazon Machine Image Selection Window



To handle large user bases, we recommend you choose general purpose machines for QlikView. For example, we suggest the m1.large and the m1.xlarge instances for single server and for cluster machines.

Figure 2. Amazon Instance Type Selection Window



## Connecting QlikView and Redshift

For demonstration purposes, and for the sake of brevity, the paper covers the ODBC connectors to access Redshift. Nevertheless, the JDBC connection process is very similar.

### Which drivers?

AWS recommends Postgres SQL drivers. Full installation instructions for these drivers are in the AWS Redshift documentation. Both the ANSI and Unicode drivers have been tested to work with QlikView.

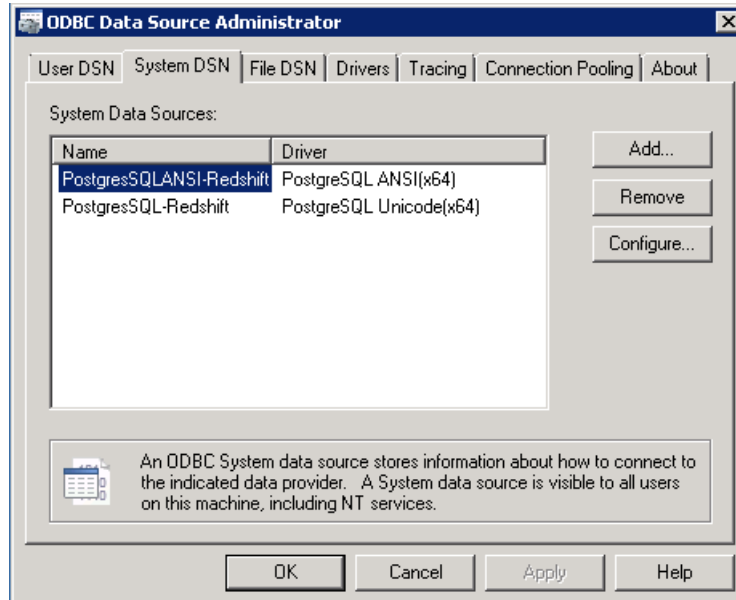
Amazon Documentation:

- [Redshift Drivers](#)
- ODBC Driver: [Postgres ODBC Download](#)

### Configuring the connector

Start the “Data Sources (ODBC)” program in Windows (notice that both 32bit and 64bit have been tested but this paper only covers steps for 64bit architecture, the 32bit are similar).

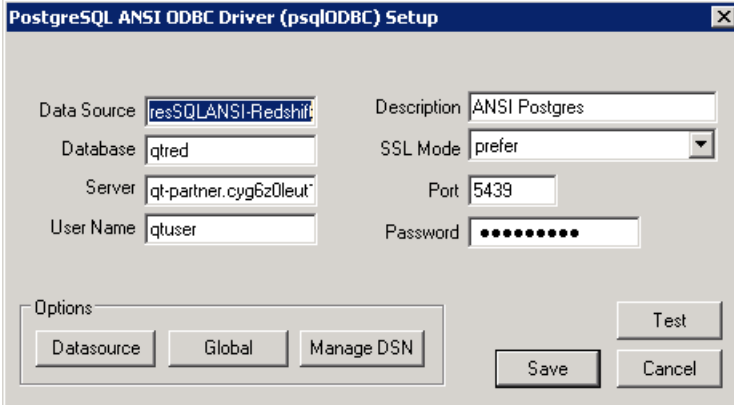
The following window should appear:



Highlight the PostgreSQL ANSI-Redshift data source and click on “Configure”.



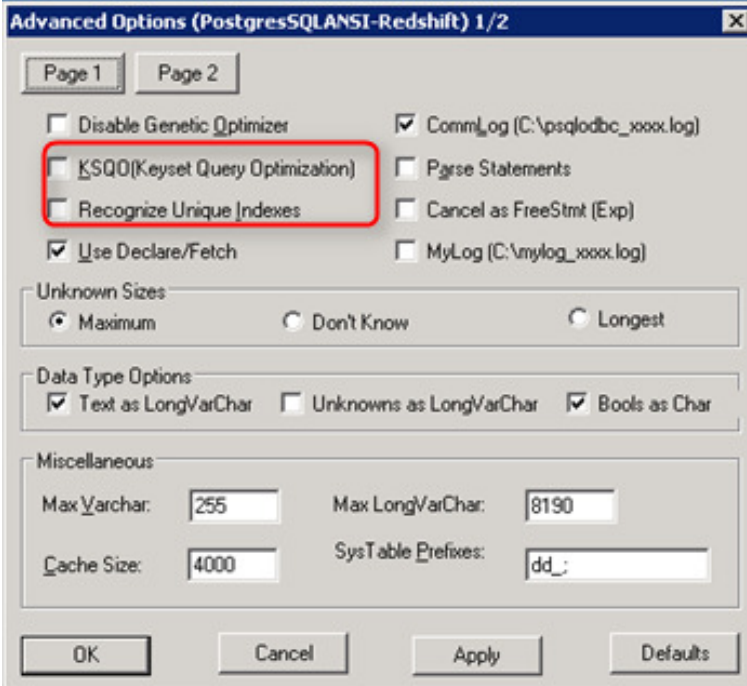
In the next window, click DataSources button to view the configuration.



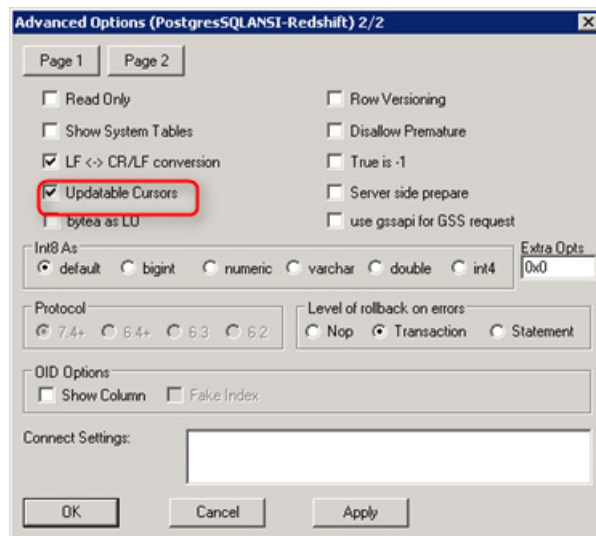
In the next window noticed the following changes:

- a) Uncheck the KSQO checkbox
- b) Uncheck the Recognize Unique Indexes Checkbox
- c) Enter a value in the Cache Size Adjustment box in order to find the optimal cache size

*\*Please note Maximum cache side size for a single Amazon Redshift node is 100*



In Step 2/2, check the Updateable Cursors checkbox to enable the ODBC driver to use cursors.



## Tuning QlikView and Redshift for performance

For performance reasons, it is recommended that complex SQL queries (such as multiple sub-selects and complex joins) are not executed from QlikView. A Best Practice would be to perform these types of queries within Redshift and send the resulting data set to QlikView via extraction through ODBC or Direct Discovery.

Below are some reference documents on how to design an Amazon Redshift Data Warehouse to work well with Qlikview.

- Design Tables for fast read
  - Be able to understand and analyze explain plans
    - Link <http://docs.aws.amazon.com/redshift/latest/dg/c-optimizing-query-performance.html>
  - Selection of correct sort keys
    - Link-[http://docs.aws.amazon.com/redshift/latest/dg/c\\_best-practices-sort-key.html](http://docs.aws.amazon.com/redshift/latest/dg/c_best-practices-sort-key.html)
  - Selection of best distribution keys
    - Link-[http://docs.aws.amazon.com/redshift/latest/dg/c\\_best-practices-best-dist-key.html](http://docs.aws.amazon.com/redshift/latest/dg/c_best-practices-best-dist-key.html)
  - Smallest column size and data set.
    - Link - [http://docs.aws.amazon.com/redshift/latest/dg/c\\_best-practices-smallest-column-size.html](http://docs.aws.amazon.com/redshift/latest/dg/c_best-practices-smallest-column-size.html)
  - Compression
    - Link[http://docs.aws.amazon.com/redshift/latest/dg/t\\_Compressing\\_data\\_on\\_disk.html](http://docs.aws.amazon.com/redshift/latest/dg/t_Compressing_data_on_disk.html)
  - Be able to understand data distribution
    - Link [http://docs.aws.amazon.com/redshift/latest/dg/t\\_Distributing\\_data.html](http://docs.aws.amazon.com/redshift/latest/dg/t_Distributing_data.html)

## Big Data, QlikView, and Redshift

---

So far, we have focused on data sets that are small enough to be analyzed in-memory. For data sets that are too large to be held in-memory, QlikView's Direct Discovery technology provides data analysis capabilities. Direct Discovery, the hybrid approach allows QlikView to access data residing in-database. The architecture of Direct Discovery places small reference data in memory and access large fact data in-database. Amazon Redshift has been tested with Direct Discovery and is known to perform well with millions of rows of data.

Keep in mind the following key points in order to make sure Direct Discovery performs well.

- Redshift Cluster and QlikView components are in same AWS Zone
- Redshift data uses correct column types and sizes
- Redshift data is sorted during inserts depending on query pattern
- If multiple clusters are used, take advantage of zone maps so tables scans are more efficient
- Ensure cursors and fetch sizes are set correctly

Note: All tests have been performed with high performance EC2 (m3.large and m3.xlarge) instances in same AWS zone to Redshift cluster.

In conclusion, Amazon Redshift and Qlik provide Keep in mind the following key points in order to make sure Direct Discovery performs well.

- Redshift Cluster and QlikView components are in same AWS Zone
- Redshift data uses correct column types and sizes
- Redshift data is sorted during inserts depending on query pattern
- If multiple clusters are used, take advantage of zone maps so tables scans are more efficient
- Ensure cursors and fetch sizes are set correctly

Note: All tests have been performed with high performance EC2 (m3.large and m3.xlarge) instances in same AWS zone to Redshift cluster.

In conclusion, Amazon Redshift and Qlik provide organizations the following new capability: ***to quickly create the right infrastructure to host big data environments, perform a multitude of discoveries within all of their data assets and quickly obtain valuable insights to better manage their businesses.***