

An Insider's Guide to SAS/ACCESS Software on SAS 9.4M5 in the Cloud

Jeff Bailey, SAS Institute Inc., Cary, NC

ABSTRACT

You might not have data in the cloud today, but chances are you will in the very near future. Are you ready? Do you have any idea where to start? If "no" is the answer to either of these questions, then this presentation is just what you are looking for. We answer the following questions (and more):

- What are the steps involved in creating a database in a cloud environment?
- What is Database as a Service (DBaaS)?
- How does SAS® define the terms "cloud variant" and "database variant"?
- How do I determine whether my cloud database is supported by SAS?

The presentation would not be complete without discussing the major cloud players. We discuss Amazon Web Services (AWS), Microsoft Azure Cloud, Oracle Cloud, Teradata IntelliCloud, and Google Cloud Platform.

INTRODUCTION

43,000! That was the attendance number being discussed at Amazon AWS re:invent in 2017. To put this into perspective, the town I grew-up in had a population of approximately 18,000. Granted, that was years ago, but the attendance number still amazes me. Another amazing fact is that in 2016 the conference drew approximately 30,000 people.

The takeaway is that cloud computing is here to stay. Moving to the cloud is inevitable, like the sunrise.

Over the course of the last year, I have personally heard the following comments from SAS customers:

- "We do not want to move to the cloud, but we must if we want to remain competitive."
- "Companies are not moving to the cloud to save money. They are doing it for the agility it offers."
- "Moving to the cloud is necessary."

Like a mad scientist's laboratory, cloud environments encourage experimentation. If you have a business idea and little else, the cloud might be just the ticket. With only a credit card, you can test your idea in the real world. Cloud environments encourage this. It is no coincidence that Amazon Web Services (AWS) calls its customers "builders" and "makers." You never hear Amazon refer to their AWS customers as users.

Let's take a brief look at some of the cloud players.

THE CLOUD

This paper focuses heavily on the database aspect of cloud. Keep in mind: this is a tiny portion of the services available on most cloud platforms.

DATABASE AS A SERVICE

Database as a Service (DBaaS) is a cloud computing model that provides users access to a database without the need to set up a machine or install database software. Everything is handled by the cloud provider. The DBaaS model enables us to create a database in a matter of minutes.

Keep in mind: it is entirely possible to create a compute instance in the cloud and then install your own database. The downside is that you are responsible for creating the compute instance, disk storage, backup strategy, and database redundancy. This is a lot of work. DBaaS handles all of that for us. In fact,

many of the cloud providers state that you do not need a database administrator (DBA) when you use DBaaS. As a recovering DBA, I am skeptical.

AMAZON WEB SERVICES (AWS)

AWS is a vast collection of services that can be used to create anything you imagine. Databases play a large role in this environment, and AWS does not disappoint. The following DBaaS products exist on AWS (Note: SAS might not support all of these products. Check the SAS cloud database support page for details.):

- Amazon Redshift
- AWS Relational Database Services (RDS)
 - Amazon Aurora (MySQL and PostgreSQL versions)
 - MariaDB
 - Microsoft SQL Server
 - Oracle
 - PostgreSQL
- AWS Elastic MapReduce (EMR) - Hadoop as a Service
- Teradata (via the Amazon Marketplace)

MICROSOFT AZURE

The following DBaaS products exist on Microsoft Azure (Note: SAS might not support all of these products. Check the SAS cloud database support page for details.):

- Azure Cosmos DB
- Azure Database for MySQL servers (Preview)
- Azure Database for PostgreSQL servers (Preview)
- Azure SQL Database
- Azure SQL Warehouse
- Azure HDInsight – Hadoop as a Service

GOOGLE CLOUD PLATFORM

The following DBaaS products exist on Google Cloud Platform (Note: SAS might not support all of these products. Check the SAS cloud database support page for details.):

- BigQuery
- Cloud SQL
- Cloud Bigtable
- Cloud Spanner
- Cloud Dataproc – Hadoop as a service

ORACLE CLOUD PLATFORM

The following DBaaS products exist on Oracle Cloud Platform (Note: SAS might not support all of these products. Check the SAS cloud database support page for details.):

- Oracle Database Exadata
- Oracle Database Cloud Service

- Oracle MySQL Cloud Service

TERADATA INTELLICLOUD

Teradata IntelliCloud is a subscription service providing Teradata Database in the cloud. This high-performance cloud service is managed by Teradata. It is important to keep in mind that Teradata runs as a service in many cloud environments.

SAS 9.4M5 SAS/ACCESS SUPPORT FOR CLOUD DATABASES

In 2017 SAS decided that to support our customers, SAS/ACCESS® software and SAS In-Database must support cloud databases. We began our cloud work by asking ourselves a very simple question – is database Foo running on-premises the same as database Foo running as a service in the cloud? This question, though interesting, is not really the right question. The right question is how do we tell if a database running on platform A is “materially different” from the same database running on platform B. Said another way, how can we determine if A is “materially different” than B?

After much discussion and even more experimentation, we came up with a validation protocol that enables us to answer this question with confidence. However, another question appeared – how can we determine if two, supposedly compatible, database management systems (DBMS) are the same? While pondering these questions, we discovered two key ideas:

- **Cloud Database variant:** The cloud version of a currently SAS supported database. Examples include: Oracle DBaaS running on Oracle Cloud Platform, Teradata Database running on Azure, and MySQL running on Amazon. Until this work was completed, connecting to one of these databases in the cloud was unsupported.
- **Database variants:** The database that is compatible with a SAS supported database. Examples include: MariaDB (API compatible with MySQL), Azure SQL Database (a variant of Microsoft SQL Server), Amazon Aurora (a variant of MySQL) and Amazon EMR (a variant of Apache Hadoop, which runs in the cloud).

This powerful idea enables SAS to greatly expand support for SAS/ACCESS software and SAS In-Database products.

WHICH DATABASES DOES SAS SUPPORT IN THE CLOUD?

The following information is accurate as I write this, but because we are constantly adding support, this list will be out-of-date as soon as this paper is published. Please check the cloud support page for the most up-to-date information.

<https://support.sas.com/en/documentation/third-party-software-reference/9-4/support-for-database.html>

SAS has extended SAS/ACCESS and In-Database support to the following cloud and database variants:

Support for Hadoop

Starting with SAS 9.4M4, these Hadoop as a Service platforms are supported:

- Amazon Elastic MapReduce (EMR)
 - SAS FILENAME statement for Hadoop
 - Reads and writes using the REST API are not supported. Set environment variable SAS_HADOOP_RESTFUL=1.
- SAS Hadoop procedure
 - Submitting Pig Latin code is not supported due to Java 8 issues.

- Reads and writes using the REST API are not supported. Set environment variable SAS_HADOOP_RESTFUL=1.
- SAS/ACCESS® Interface to Hadoop

Support for MySQL

Starting with SAS 9.4M4, SAS/ACCESS® Interface to MySQL supports these platforms:

- Amazon RDS Aurora (MySQL variant)
- Amazon RDS MariaDB
- Amazon RDS MySQL
- MariaDB

Support for Microsoft SQL Server

Starting with SAS 9.4M4, SAS/ACCESS® Interface to Microsoft SQL Server supports these platforms:

- Amazon RDS Microsoft SQL Server
- Microsoft Azure SQL Database

Support for Oracle

Starting with SAS 9.4M4, these Oracle platforms are supported:

- Amazon RDS Oracle
 - SAS/ACCESS® Interface to Oracle
- Oracle Cloud Platform – Oracle Database Cloud Service
 - SAS/ACCESS Interface to Oracle
 - SAS Scoring Accelerator for Oracle

Support for Teradata

Starting with SAS 9.4M4, these Teradata platforms are supported:

- Teradata IntelliCloud
 - SAS/ACCESS® Interface to Teradata
 - SAS® Scoring Accelerator for Teradata
 - SAS® Code Accelerator for Teradata
- Teradata Database on Amazon Web Services
 - SAS/ACCESS Interface to Teradata
 - SAS Scoring Accelerator for Teradata
 - SAS Code Accelerator for Teradata
- Teradata Database on Microsoft Azure
 - SAS/ACCESS Interface to Teradata
 - SAS Scoring Accelerator for Teradata
 - SAS Code Accelerator for Teradata
- Teradata Database on VMware
 - SAS/ACCESS Interface to Teradata

- SAS Scoring Accelerator for Teradata
- SAS Code Accelerator for Teradata

DATABASES IN THE CLOUD

CREATING A DATABASE IN OLDEN TIMES

Back in the good old days, creating a database was difficult. First, you had to convince someone to buy you a machine. When that failed, you had to beg, borrow, or steal one. Having to buy a machine is the worst possible case because it requires a lot of justification and then there is the parts list. It is not easy to order a server class machine.

Once, I was given the approval to order a very nice Sun server – I named it Shakey.

It took weeks to convince management that we needed a new machine. We proceeded to create a sizing estimate. Naturally, we exaggerated the specifications because being wrong meant having a poorly performing database server. Having too much power was OK; having too little would be bad.

It turns out that the simple task of finding a place to put the machine was difficult because there were “rules” covering where you could place a server. With this in mind, we ordered a model that included a case that could sit on the floor. Shakey was hidden in my office. One week later, Shakey found a new home in a common area because he was intolerably loud and turned my office into a sweat box.

When he first arrived, we powered-on Shakey and installed Solaris. Once the OS was set up properly, we installed and configured Oracle and SAS. This was the easy part.

Keep in mind, this would not be a production server. We were going to use it for development work. A production server would have been much more difficult.

Bottom line: The entire process, from idea to first SQL query, took 2 months.

The key takeaways are:

- It is time consuming and expensive to obtain the machine.
- The cost of a sizing error can be huge. (Therefore, it is so common to err toward the more expensive side.)
- Setting up a development environment is easier than a production environment. Production requires redundancy, monitoring, and backups.
- Putting a UNIX server in your office is a bad idea.
- And most importantly: By the time you go through this process, the business opportunity might have vanished.

THE CLOUD CHANGES EVERYTHING

The pre-cloud world can be summarized as follows:

- Convince management that you need an expensive machine.
- Determine the specifications for the expensive machine.
- Order expensive machine.
- Wait for expensive machine.
- Install and configure database software on expensive machine.
- Be responsible for the expensive machine.

In the cloud world, this process can be reduced to starting up a Database-as-a-Service (DBaaS) instance. If the idea works out, change to a more powerful instance. If it does not work out, delete your DBaaS and move on to something else.

The cloud enables us to be flexible. We are no longer required to make critical decisions at the beginning of our projects – no perfect foresight required. Plus, we are no longer required to make a huge investment up front. We can get started with little more than a credit card and an idea. If we are fortunate enough to need a more robust environment, we can handle that with a few clicks of a mouse. If it doesn't work out, salvation is a Delete button away.

CREATING A DATABASE IN THE CLOUD

There are many approaches to creating databases in the cloud:

- Use the vendor's web UI to create the database.
- Use the vendor's web UI and a configuration file (usually JSON or YAML) to make the process easier and less error prone.
- Create a script and a configuration file that automatically creates the database.
- Use an API (typically Python, Java, Go, and so on) to programmatically to create the database.
- Use REST API calls to create the database.

The specific approach depends on the cloud platform being used.

Check my Github for examples and notes on how to do this.

<https://github.com/Jeff-Bailey>

BONUS: THREE QUICK PERFORMANCE TIPS

To understand cloud database performance, you must understand where processing is taking place and data movement. Data movement is especially important in hybrid environments (SAS on-premises accessing a cloud database system). Not only is moving data between on-premises and the cloud slow, it tends to be expensive.

PERFORMANCE TIP #1

Consider using the READBUFF= and INSERTBUFF= SAS LIBNAME options. These options can significantly improve the performance of data movement. This is especially true of ODBC data sources.

For more information, see my 2014 SAS Global Forum Paper: "An Insider's Guide to SAS/ACCESS Interface to ODBC," which is available at

<https://support.sas.com/resources/papers/proceedings14/SAS039-2014.pdf>.

PERFORMANCE TIP #2

Use the DBIDIRECTEXEC option to ensure that the DBMS does the work when running CREATE TABLE AS (CTAS), INSERT AS SELECT (IAS), UPDATE, and DELETE operations. Once again, this option can significantly reduce the amount of data that must move between SAS and the database.

I have written an article that covers this topic. It is available from the SAS Communities.

<https://communities.sas.com/t5/SAS-Communities-Library/DBIDIRECTEXEC-GOFAST-YES-for-Database-Processing/ta-p/342717/jump-to/first-unread-message>

PERFORMANCE TIP #3

On AWS, bulk load data using Amazon S3 (the cloud object store for Amazon) when possible. For example, SAS/ACCESS® Interface to Amazon Redshift enables us to load from Amazon S3.

This technique should be explored when using ODBC data sources, too. It might be possible to combine PROC S3 with the SAS/ACCESS® Interface to ODBC to mimic the approach taken with SAS/ACCESS to Amazon Redshift. This approach is favored by cloud database vendors such as Snowflake (Snowflake LOAD SQL statement). Snowflake provides a way to write the output of large queries to Amazon S3. Review your DBMS documentation to see if this type of approach can be applied to your situation.

CONCLUSION

SAS 9.4M4 and later supports many new DBaaS environments. The number will continue to grow. Be sure to check the SAS Cloud Database Support page often. It is important to understand that hybrid environments pose a special set of problems with data moves. This is due to the nature of the cloud itself. We must do our part to ensure that we do not move data between cloud and on-premises unless it is necessary. We discussed SAS options and techniques that will help increase the performance of our cloud environment.

The key takeaway is that the cloud is inevitable. Don't fight it because it is a lot of fun!

REFERENCES

Bailey, Jeff. 2010. SAS Institute white paper. "Troubleshooting SAS and Teradata Query Performance Problems." Cary, NC: SAS Institute, Inc. Available at <http://support.sas.com/resources/papers/TroubleshootingSASandTeradataQueryPerformanceProblems.pdf>.

Bailey, Jeff, and Tatyana Petrova. 2013. "The SQL Tuning Checklist: Making Slow Database Queries a Thing of the Past." *Proceedings of the SAS Global Forum 2013 Conference*. Cary, NC: SAS Institute Inc. Available <https://support.sas.com/resources/papers/proceedings13/080-2013.pdf>.

Bailey, Jeff. 2014. "An Insider's Guide to SAS/ACCESS® Interface to ODBC." *Proceedings of the SAS Global Forum 2014 Conference*. Cary, NC: SAS Institute Inc. Available <https://support.sas.com/resources/papers/proceedings14/SAS039-2014.pdf>

ACKNOWLEDGMENTS

A handful of my favorite people were subjected to a litany of questions and asked to review the ideas presented in the paper. Lucky for me they provided more help than I could have dreamed of. Thank you:

Marie Dexter, SAS Institute Inc.

Salman Maher, SAS Institute Inc.

Tatyana Petrova, UltraStrategy, LLC.

David Shamlin, SAS Institute Inc.

Kim Sherrill, SAS Institute Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Jeff Bailey
100 SAS Campus Drive
Cary, NC 27513
SAS Institute Inc.
Jeff.Bailey@sas.com
www.sas.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.