Paper 4631-2020

Best Practices for Enabling SAS® Analytics in the Cloud—at Scale

Heather Burnette, Teradata Corporation

ABSTRACT

We know cloud is the future, but it is quickly becoming our present! An increasing number of companies, large and small, are turning to the cloud to deliver advanced analytic solutions that will scale. In fact, the cloud can be the perfect environment for evolving your SAS® analytics to solve real-world problems quickly, easily, and affordably. But to do that, you need to understand how to make it work for your business.

Whether you're an administrator, business user, architect, or engineer, this presentation has something for you. We discuss everything from broad best practices to specific technical tips and tricks to show how to optimize your cloud solution. We illustrate how companies are using SAS®9, SAS® Viya® and Teradata Vantage in public and private clouds to achieve the following: reduce and avoid cost; better manage and prepare data; increase operational efficiency; leverage cloud marketplaces to streamline setup and configuration; and quickly scale SAS solutions to production.

INTRODUCTION

There are many reasons companies are moving to the cloud. Some of the benefits include more agile and scalable development and simplified administration/management of environments with reduced operational overhead. However, before you can take advantage of the benefits of cloud, you must successfully move your analytic ecosystem there.

The purpose of this paper is to provide you with best practice tips and resources to help you get the most out of your analytic ecosystem in the cloud. We'll do this in 3 ways:

1) By pointing out common pitfalls we've seen organizations make when moving to the cloud, and how to avoid them.
2) Providing specific tips to help you optimize the performance of analytic jobs in your new cloud environment.
3) Show an example of how customers are starting to combine traditional data with data in cloud storage to get the best possible answers to their business questions.

To help with this paper, if you're not familiar with cloud architecture or terminology, please start by reading the "CLOUD TERMINOLOGY" section at the end of this document.

Also note that for simplicity, many of the Cloud terms that vary by vendor are used interchangeably in this document. For example, VPC/VNet.

OK, Let's get started…

SO, YOU'RE MOVING YOUR ANALYTICS TO THE CL**OUD…**

Many of the common pitfalls of moving to the cloud can be avoided by following some basic principles. Here are our tips to help you along the way:

PLANNING TIP #1: MAKE A PLAN FOR YOUR WHOLE ECOSYSTEM – AND NOT IN A VACUUM

The end goal of this step is to derive a plan for every part of your analytic ecosystem. You should determine what components will move to the cloud vs what is staying on-premises and how and when you propose to migrate them. We see two main pitfalls during this stage.

Common Pitfall #1:

Only planning for only parts of your analytic ecosystem without considering impacts to other components (i.e. missing dependencies between key pieces)

How to Avoid it:

To avoid missing key components or dependencies when moving your analytic ecosystem to the cloud, it's best to start by taking a complete inventory of your systems. This includes, but is not limited to:

- Applications
- Authentication
- Workloads (SAS jobs, Analytic scripts…etc)
- Data Sources (Databases, Data Sets…etc)
- Connectivity
- Security
- Home-grown processes

This may seem like a lot of work, but it is necessary to create a successful plan. And you don't need to perform the inventory or make all the decisions alone. That leads us to the next common pitfall…

Common Pitfall #2:

Doing everything yourself, and thus making decisions in a vacuum.

How to Avoid it:

To avoid doing everything yourself and making decisions in a vacuum, it's best to form a team of people from multiple areas of your business, such as:

- IT
- Security
- Developers
- Data Owners
- Business Analysts

The purpose is to cover all the equity stake holders affected by moving your analytic ecosystem to the cloud. Feel free to include representatives from other departments in your company you feel would be helpful.

You should also include experienced cloud implementation specialists. These specialists may come from within your organization (if other projects have already moved to the cloud), or they may need to come from outside if this is your first major cloud project.

As you gather information and discuss cloud requirements with the different groups, you may discover "shadow" data and analytic applications that should be included in your cloud migration plan.

The newly formed team should perform the following tasks during this planning phase:

1) Help complete a comprehensive inventory
2) Perform a detailed analysis of requirements for each component. Make sure you determine all links/dependencies between each part of your analytic ecosystem.
3) Create a plan for every component in the analytic ecosystem. Be specific about:
    a. What will to move to the cloud?
    b. What will continue to stay on-premises?
    c. How and when you will migrate each piece?
4) Communicate this plan to all affected groups

> Expert Tip – To Increase Operational Efficiency:
>
> When making your plan, move components with shared dependencies to the cloud at the same time.
>
> For example, when moving analytics to the cloud, consider moving all the data that is used for the analytics to the cloud as well.
>
> Moving dependent components to the cloud simultaneously will result in faster speeds for your analytic jobs, which increases your operational efficiency.

Note that having a cross-functional team will continue to be useful at all stages of this process. Have the team meet regularly, even after the planning phase, to track implementation progress and troubleshoot any issues that arise.

PLANNING TIP #2: CO-LOCATE EVERYTHING IN THE CLOUD AS MUCH AS POSSIBLE

Now that you know what is moving to the cloud and when, you need to determine where to put it in the cloud. There is one common pitfall we see at this stage.

Common Pitfall:

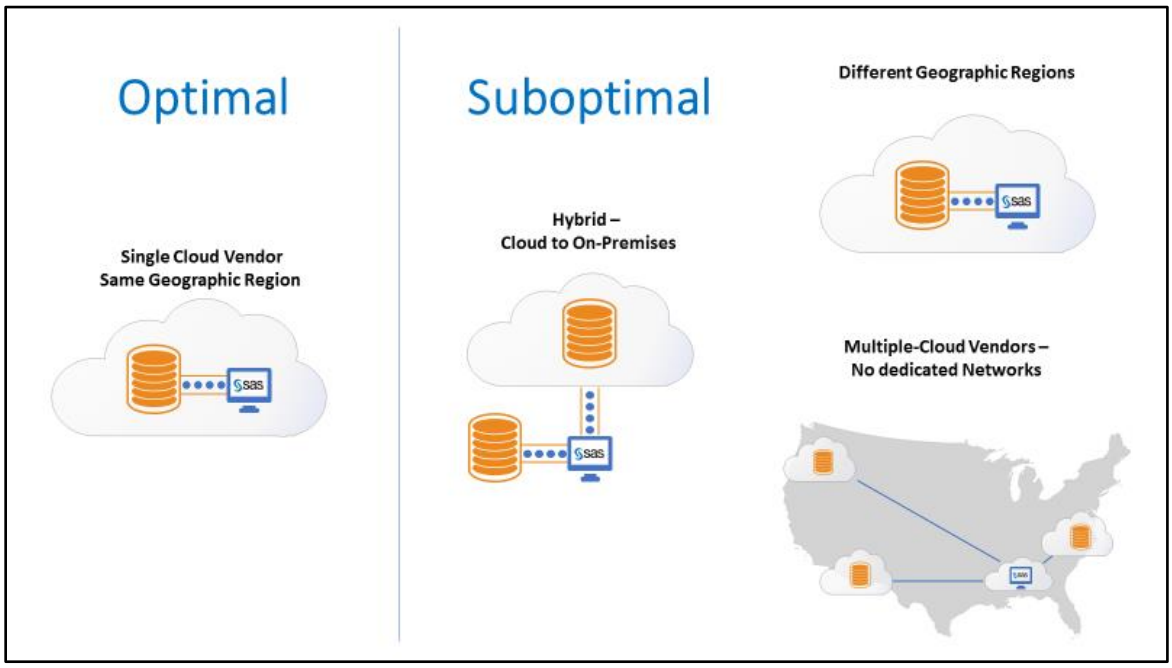
Not moving all dependent Applications, Processes or Data Sources to the Cloud OR
Not placing all Instances in the cloud as close together as possible.

How to Avoid it:

One of biggest causes of performance issues when moving to the cloud is increased latency. The further apart the individual components of your analytic ecosystem are physically located, the longer it will take them to communicate. To counteract this, you must get every part of your analytic ecosystem as close together as physically possible.

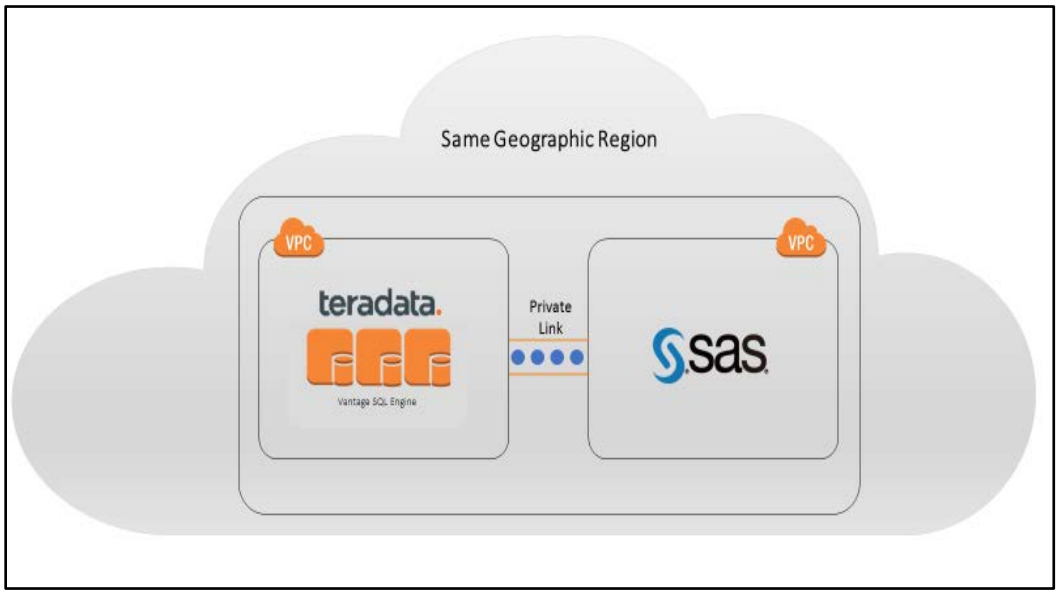
At a minimum, we recommend putting everything in the same Geographic Region of a single cloud vendor. This will improve performance, reduce data transfer costs and increase operational efficiency.

Figure 1. Possible Cloud Configurations



Notice: The further apart the key components are located, the worse your performance will be.

Figure 2. Best Configuration for SAS and Teradata Vantage in the Cloud - Detail



For its architecture in the cloud, Teradata Vantage assigns all of its components to a single Availability Zone. Then for security reasons, Teradata Vantage is placed into a single tenant Virtual Private Cloud (VPC). SAS recommends you assign all your SAS servers to a single Availability Zone and place them in one VPC. While the Teradata and SAS VPCs are separate, to achieve the best performance the SAS and Teradata VPCs should exist in the same Geographic Region.

If you are concerned about high availability, backup and restore capabilities or redundancy, we recommend setting up a secondary system in a different region.

PLANNING TIP #3: INVEST IN HIGHSPEED CONNECTIVITY OPTIONS

Depending on your new configuration in the cloud, the default networking options may not enable the kind of analytic processing speeds, or security, that you are used to with your on-premises network. This leads us to our next pitfall….

Common Pitfall:

Using the default, slower, cloud networking options.

How to Avoid it:

To avoid this pitfall, it's essential to educate yourself on what connectivity options are available from your cloud vendor. Networking speeds, security and cost will vary by cloud vendor, connectivity option and usage. When researching, pay close attention to networking speeds and whether the option is private or shared. SAS recommends using networks connections with minimum speeds of 10 Gigabits per second (Gbps)

There are three kinds of connectivity you want to carefully consider at this stage.

1) Connecting multiple Virtual Private Networks in the Cloud (i.e. VPC to VPC connectivity) if you have more than one.
2) Connecting to Cloud Storage (S3, Azure Blob or Google Cloud Storage)
3) Connecting your cloud environment back to your on-premises network (Hybrid configuration).

For best performance in each of these scenarios, use private, dedicated connections, where available.

You should keep in mind that small delays in time, due to increased latency or slower networking speeds, will add up over time and are multiplied across users. In the end, a lack of network efficiency will end up impacting response times as well as incurring upcharges in networking fees. However, you will need to weigh the cost benefits of each choice to make the best decision for your company.

We have some recommendations and resources to help get your research started.

For VPC to VPC Connectivity in the Cloud:

If you need multiple Virtual Networks in the cloud, you'll want a network connection that is fast and secure. We recommend researching these options from the various cloud providers:

AWS: Private Link

Azure: Private Link

Google Cloud: VPC Network Peering

For Connectivity to Cloud Storage:

For a private, secure connection to your Cloud Storage, we recommend reading about the following options:

AWS: VPC Endpoint

Azure: Private Endpoint; Using Private Endpoints with Cloud Storage

Google Cloud: VPC Service Controls

Connectivity for Hybrid Configurations:

Typically, there are company resources that must remain on-premises for a myriad of reasons. Because hybrid configurations have the furthest distance to travel, they can be the least performant scenario. For hybrid configurations like these, we recommend talking to your cloud vendor about setting up a private, dedicated high speed connection between the cloud and your on-premises environments.

For AWS and Azure, this requires you to work with a networking partner. Google Cloud owns their own worldwide private network and can generally offer better speeds.

Here are links with more information:

AWS: Direct Connect

Azure: ExpressRoute

Google Cloud: Cloud Interconnect

PLANNING TIP #4: APPROPRIATELY SIZE YOUR SYSTEM FOR THE CLOUD

Your analytic ecosystem running in the cloud will look very different from your on-premises configuration. Each Cloud Vendor offers specific types and sizes of instances that you can provision. This leads us to our next common pitfall:

Common Pitfall:

Under-sizing your analytic ecosystem in the cloud.

How to Avoid it:

To determine what resources to provision in the cloud, start by using the detailed assessment you performed during the planning phase. Pay close attention to the workloads that will be moving to the cloud. Some workloads require faster CPU for processing data, others require more IO throughput, and each have memory and/or file storage considerations. This is a good place to start when selecting instance sizes.

Here are some additional resources that can help appropriately size your environment:

For Your SAS Environment:

The SAS Performance Lab has written several technical papers with recommendations for sizing your SAS Systems in the cloud. This is a fantastic resource. It suggests specific kinds and sizes of servers for each of the major cloud vendors.

They give two general guidelines when choosing instance types for SAS Servers:

1) Be aware, CPUs listed on public clouds may be hyperthreaded virtual CPUs. This is important because SAS's requirements are based on Physical cores.
2) To get good network performance from your SAS system, you may need to provision an instance that has more physical cores than your current workload requires. This is because an instance with more cores, often includes a dedicated Network Interface Card (NIC) to maximize IO throughput.

The full paper from the SAS Performance Lab can be found here:

Important Performance Considerations When Moving SAS®to a Public Cloud

NOTE: Check for updates to this information that are planned to be part of the SGF 2020 proceedings.

For Teradata Information:

To build the best performing Teradata Vantage system in the cloud, we strongly recommend reaching out to Teradata directly. For existing customers, you can contact your Teradata Account Team. For new or prospective customers, please contact us at: https://www.teradata.com/About-Us/Contact

Additional resources & Market Place Offerings:

AWS: Teradata Vantage on AWS; Teradata Vantage Market Place

Azure: Teradata Vantage on Azure; Teradata Vantage Market Place

Google Cloud: Teradata Vantage on Google Cloud

For Cloud Storage Information:

You can use these resources to read about Cloud storage options and choose what works best for your company:

AWS: Simple Storage Service (S3)

Azure: Blob Storage

Google Cloud: Cloud Storage (GCS)

Helpful Sizing Tip – To Save Time and Money:

When making sizing decisions, you should add in estimates for any new workloads and/or storage needs you anticipate will be added shortly after you move to the cloud.

One of the benefits of cloud is agility and scalability, but it wastes company resources, time and money to go through a resizing exercise too soon after moving to the cloud.

If all of this seems like a lot to consider, we agree. That's why we have our next tip…

PLANNING TIP #5: SEEK OUT EXPERT ADVICE

Planning a move to the cloud can seem overwhelming, especially if you have a large analytic ecosystem or limited resources. New cloud technologies and concepts have a steep learning curve if this is your first large scale cloud implementation. This leads us to the next common pitfall we've seen.

Common Pitfall:

Trying to move to the cloud on your own.

How to Avoid it:

There's no need to start from scratch or reinvent the wheel. Many experts who have been through this process already are available to help. Leaning on their expertise will ultimately save you time, headaches and money.

Here are some recommendations we've collected from talking to SAS and Teradata Cloud experts:

Resources For SAS:

For your SAS Installation, we recommend reaching out to your SAS Account Team. SAS can help you perform a detailed sizing of your existing SAS system, discuss what you would like to add to it, and help you get it all implemented in the cloud. You can get additional information about SAS Cloud offerings at:
https://www.sas.com/en_us/solutions/cloud-analytics.html

Resources For Teradata Vantage:

For existing Teradata customers, we recommend contacting your Teradata Account Team. Teradata has experience helping many customers move to the cloud. There's even a dedicated team just for Cloud Security. The Teradata Cloud teams can help develop a customized plan to meet your exact performance, capacity and security needs. You can get more information about Teradata in the Cloud at:
https://www.teradata.com/Cloud

Resources for Cloud Vendors:

We also recommend you get your cloud vendor of choice involved. They will be able to work with you on cost estimates and service level agreements, as well as alerting you to any discounts that your organization may qualify for. Here are links to the three major cloud vendors we recommend:

AWS Azure Google Cloud

PLANNING TIP #6: PERFORM YOUR OWN PROOF OF CONCEPT (POC)

After you've designed a solution and worked with the experts to come up with a good plan to implement your analytic ecosystem in the cloud, we strongly recommend performing your own Proof Of Concept (POC) before turning your system over to users. This leads us to our next common pitfall…

Common Pitfall:

Not adequately testing your new system before turning off your old system.

How to Avoid it:

It's important to note, some of your workloads may perform differently in the cloud than they did on-premises. For this reason, it's a good idea to perform a Proof of Concept (POC) on your new cloud system, based on your applications and your data, before fully committing to move off your old systems.

Your POC should include a variety of existing analytic workloads, such as:

- SAS jobs,
- Campaigns,
- Python/R/other Analytic Scripts
- ETL Jobs
- Security Scans

and any other kinds of jobs you typically run in your current environment.

Make sure the POC tests have the same size, scale, and replicate the same number of concurrent users as your current system. Then verify the results still meets your SLA's and performance goals. Based on your testing you may need to tweak some of your system sizing, network or analytic workloads to get the desired performance level.

Extra Tip – Detect Issues Early:

After you've completed your POC and turned on your new system in the cloud, you should set up continuous monitoring. Each cloud vendor has networking and detailed billing tools available for you to use. Many are included with your regular service fees.

You can easily import the data into Teradata Vantage and use SAS to track and quickly identify anomalies and other issues, before they create larger problems. Using these monitoring tools will help increase your proactive security measures and reduce unwanted costs before you incur them.

SO Y**OU'RE** IN THE CLOUD, NOW **WHAT???...**

You may think all the work to optimize your cloud system is done now you've made your transition to the cloud, but that is incorrect. To get the best performance from your new cloud environment, running your analytic programs in the cloud may require a shift in thinking. Here are some tips to help optimize the performance of your analytic jobs in the cloud:

TECH TIP #1: RETHINK HOW YOU RUN YOUR ANALYTIC WORKLOADS

A tip to combat increased latency and lower bandwidths is to rethink how you manage your workloads. Running more analytic jobs at the same time will overcome increased network latency by filling up the available network pipeline with work.

To do this, you should examine the jobs you run regularly. Then educate yourself on the network bandwidth capabilities of your system. See if there are (m)any of your smaller SAS jobs that you can run at the same time, where there are no dependencies between jobs. If needed, you can work with your IT team to help you schedule some of your regular workloads.

Consider moving less important jobs to non-peak hours to help balance the demand for network bandwidth.

Warning: You should be thoughtful when choosing jobs which will run at the same time. Running similar analytic jobs that use the same database tables could introduce locking issues.

Extra Tip – For Jobs That Take a Long Time to Run:

If you have slow running jobs, you can look at the SAS logs to determine which steps in the process take the most time. Then study the steps to see how they can be improved. Turn on additional tracing by adding the following statements to your SAS programs.

For SAS 9.4 – sastrace options:

```
options sastrace=",,d,d" sastraceloc=saslog nostsuffix ;
```

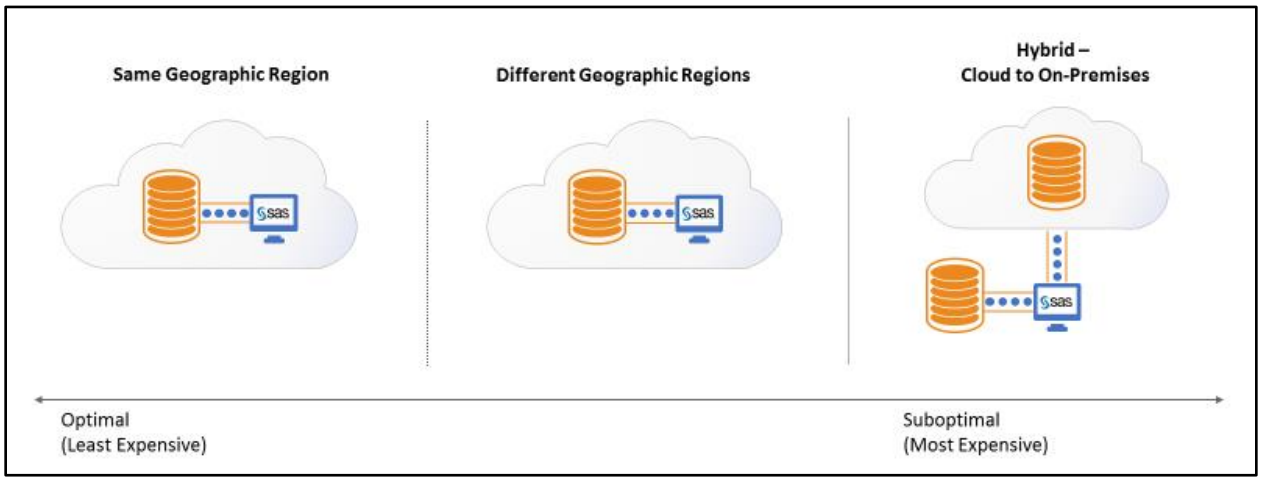
For Viya – caslib options:

```
caslib tdlib datasource=(
       srctype="teradata",
       username="myusr1",
       password="mypwd1",
       server="tdserver",
       database="sas",
       DRIVER_TRACE="SQL",
       DRIVER_TRACEFILE="/tmp/sastrace.log",
       DRIVER_TRACEOPTIONS="TIMESTAMP|APPEND"
);
```

TECH TIP #2: LIMIT YOUR DATA TRANSFER

In the cloud, transferring large amounts of data can cost you in two ways: TIME & MONEY! Being in the cloud requires a new way of thinking. In general, the further the data travels in the cloud, and the more data you move, the longer it will take, and more expensive it will be.

Figure 3. Cost of Data Transfers



Caption: The cost of transferring data in the cloud is determined by how much data is moved and how far.

When writing your analytic programs to run in the cloud, you should ask yourself:

1) How much data am I moving?
2) What is the smallest amount of data I can use and still get the answer I need?
3) Where am I moving the data to/from?

This may require you to do some research and find out where your data & SAS servers are located. They may not be as close to each other as you think they are. And remember, the most expensive kind of data transfer is moving data out of the cloud.

Extra Tip – To Reduce Character Data Expansion in Viya:

For SAS Viya, when data is transferred it may be expanded before moving. This is controlled by the NCHARMULTIPLIER option.This option specifies a multiplication factor that is used to increase the number of bytes for characters. By default character data is transcoded to UTF-8 encoding with three bytes per character when it is sent from Teradata Vantage to the CAS server.

There are 2 settings which control this:

Viya – system level setting:

```
CASNCHARMULTIPLIER=<value between 0 and 4>
```

Viya – caslib option:

```
caslib tdlib datasource=(
       srctype="teradata",
       username="myusr1",
       password="mypwd1",
       server="tdserver",
       database="sas",
       ncharmultiplier=<value between 0 and 4>
);
```

If you run SAS jobs that only use ASCII values, you can set the value of the NCHARMULTIPLIER to 1. This indicates you do not need to expand the data when it's transcoded to UTF-8. However, if most of your data is multi-byte Unicode, you should set the value of the NCHARMULTIPLIER to 2 or 3.

TECH TIP #3: RUN AS MUCH AS POSSIBLE "IN THE DATABASE"

Another way to reduce data transfers is to run SAS programs "in the database". SAS and Teradata have jointly developed a wide range of In-Database capabilities to do just this, including:

- SAS® Scoring Accelerator
- SAS® Code Accelerator
- SAS® In-Database Formats
- SAS® Data Quality Accelerator
- SAS® Base Procedures
- SAS/ACCESS Interface to Teradata
- SAS/ACCESS Interface to Teradata (on SAS® Viya)
- SAS® In-Database technologies for Teradata (on SAS® Viya) (includes Scoring, Data Quality and Data Connect Accelerator)

Please check with your SAS Sales representative about product bundling and licensing.

The key benefits of In-Database processing include:

- Pushing SAS processing down to the Database nodes using the SAS® Embedded Process for Teradata
- Eliminates the lag time and cost of transferring large amounts of data
- Leverages the scalability and efficiency of the Teradata Vantage SQL Engine's optimizer for processing large volumes of data in parallel
- Provides access all the functions in the Teradata Vantage SQL Engine, including 100's of existing functions and many new analytic functions
- Improved data governance and security, due to fewer copies of sensitive data

One of the easiest ways to take advantage of these benefits, is to perform your Data Discovery and Data Preparation steps in the Teradata database. This is a very efficient way to manage and prepare your data without adding the increased time and cost to move it.

NOTE: All the commands in the examples shown below can be run using SAS Studio.

Connect to Teradata:

Set up your LIBNAME statement to point to your data in Teradata:

```
%let idconn = server=<teradata server> user=<user name>
            password=<password> database=<database name> mode=teradata;
libname td teradata &idconn;
```

Use SAS Base Procedures that will be pushed down to the database:

PROC MEANS - Example:

```
proc means
     data=td.some_table;  /* columns a,b,c */
quit;
```

Use SAS PROC SQL to write SQL statements to view and prep your data:

PROC SQL - Implicit Pass-Through Example:

```
proc sql;
     select a,b,c, count(*) as cnt
     from td.some_table
      group by a,b,c;
quit;
```

PROC SQL - Explicit Pass-Through Example:

```
proc sql;
     connect using td as tdx;
     create table work.some_table2 as select * from connection to tdx
          (select a,b,c from sasdemo.some_table where a < 100);
     disconnect from tdx;
quit;
```

Use SAS Data Quality Accelerator (DQA) to cleanse data in the database:

Data Quality Accelerator – Standardize Example:

```
proc sql;
     connect to teradata(&idconn);
     execute (
          call sas_sysfnlib.dq_standardize(
          'State/Province (Abbreviation)',
          'demo_data.master_customer_list',
          'state',
          'cid',
          'demo_data.cust_state_std',
          'ENUSA')
     ) by teradata;
     disconnect from tdx;
quit;
```

NOTE: This uses the SAS Quality Knowledge Base (QKB) with the SAS Data Quality Accelerator (DQA) to standardize the data.

After you have finished your data prep, you can use your favorite SAS modeling product, such as Enterprise Miner or SAS Model Studio, to create your models. Then use SAS Model Manager to Publish models to Teradata Vantage.

Finally, you can use the SAS Scoring Accelerator to efficiently score the model inside the Teradata Vantage Database without moving any of your data.

Use SAS Scoring Accelerator to score models in the database:

For SAS 9.4 – Use PROC SQL to Call SAS_SCORE_EP in Teradata - Example:

```
proc sql;
     connect to teradata as td (&idconn);
     execute (
            call sas_sysfnlib.sas_score_ep (
            'MODELTABLE=tddemo.sas_model_table',
            'MODELNAME=HomeEquity',
            'INQUERY=tddemo.scoredata',
            'OUTTABLE=tddemo.score_out',
            'OUTKEY=id',
            'OPTIONS=VOLATILE=NO;UNIQUE=YES;DIRECT=YES;')
     ) by td;
     disconnect from td;
quit;
```

For Viya – Use PROC SCOREACCEL to score the model in Teradata - Example:

```
proc scoreaccel sessref=session&usernumber.;
   runmodel
              target=teradata
              server="myserver"
              username="myusr1"
              password="mypwd1"
              database="viya"
              modelname='forest'
              modeltable="sas_model_table"
              intable="viya.&tablename._&usernumber"
              outtable="viya.&tablename._&usernumber._s"
              outkey="id_cust"
              verbose
              trace;
        ;
    run;
 quit;
```

While we suggest you do as much processing in the database, where the data lives, we recognize there are times you will have to move data from the database into SAS applications. For these cases, we have another tip…

TECH TIP #4: WHEN YOU MUST TRANSFER DATA, DO IT EFFICIENTLY

SAS and Teradata have jointly developed several methods to transfer data efficiently. These methods take advantage of the Massively Parallel Processing (MPP) architecture built into Teradata Vantage.

Here are some ways to do this:

For SAS 9.4 Applications – Use LIBNAME options:

SAS applications that connect to Teradata Vantage via LIBNAME statements, can use the Teradata Parallel Transporter (TPT) export/loading built into the SAS/ACCESS Interface to Teradata. The LIBNAME statement can be set up once in the SAS Metadata Server and used by all SAS users in other SAS applications, like SAS Customer Intelligence.

For Small Data Transfers – Option #1:

```
LIBNAME tdata1 Teradata server=myserver user=myusr1 pw=mypwd1 TPT=NO;
```

For Large Data Transfers – Option #2:

```
LIBNAME tdata2 Teradata server=myserver user=myusr1 pw=mypwd1 TPT=YES
FASTEXPORT=YES FASTLOAD=YES;
```

NOTE: We recommend setting up both LIBNAME Options shown above and using them where appropriate. If you are limited to using just one LIBNAME statement, use TPT=NO on the LIBNAME statement and add the TPT=YES data set option to the specific steps that transfer a large amount of data.

For Example, use Option #1 for queries that return small amounts of data (like Count(*) – single row responses). Use Option #2 for processes that transfer large amounts of data. "Large amounts of data" refers to queries that return 100K to 1M rows or more. Please note, there is some overhead for TPT. You may need to run a few tests in your system to determine the exact cross-over point, where you gain the most benefit from using TPT

For SAS 9.4 Programs – Use TPT to Export & Load data in SAS Programs:

Example #1: Exporting Data from Teradata Vantage and loading it into a SAS Dataset:

```
LIBNAME td Teradata server=myserver user=myusr1 pw=mypwd1;

data mytest (TPT=YES FASTEXPORT=YES);
     set td.tenmillion;
run;
```

You can confirm TPT was used by looking at the SAS log. You will see something like:

```
NOTE: Teradata connection: TPT FastExport has read 10000000 row(s).
```

Example #2: Saving a SAS Dataset to Teradata:

```
LIBNAME td Teradata server=myserver user=myusr1 pw=mypwd1;

proc append base= td.customers (TPT=YES FASTLOAD=YES)
     data= work.customers
run;
```

You can confirm TPT was used by looking at the SAS log. You will see something like:

```
NOTE: Teradata connection: TPT FastLoad has inserted 10000000 row(s).
```

These are additional good resources in the SAS Documentation about using TPT:

TPT Dataset Option

Maximizing Teradata Load and Read Performance

For SAS Viya – Use the DataTransferMode CASLIB option:

There are three ways you can transfer data in Viya (Serial, Multi-Node, Parallel) – listed from least to best performance. Serial loading is the least performant and Parallel loading performs the best.

SAS/ACCESS Interface to Teradata for Viya – Multi-node Load Example:

NOTE: This method causes each CAS worker to send its own SQL query to the database to read or write a subset of the data. Serial loading (without the numreadnodes, numwritenodes options) is slower because all data is transferred through the CAS controller using a single SQL query.

```
/* Create a default CAS session and create SAS libref */
cas mysess;
libname mycas cas caslib=casuser;

/* Use multi-node loading */
caslib tdlib datasource=(
            srctype="teradata",
            username="myusr1",
            password="mypwd1",
            server="tdserver",
            database="sas",
            datatransfermode="serial",
            numreadnodes=0,
            numwritenodes=0
);
```

```
proc casutil;
  load
      incaslib="tdlib"    casdata="accounts"
      outcaslib="casuser" casout="accounts"
      ;
run;
```

NOTE: We recommend using numreadnodes=0, numwritenodes=0, as this will use all available CAS worker nodes. In the SAS log you can see the Load was performed successfully:

```
NOTE: Performing serial LoadTable action using SAS Data Connector
to Teradata.
```

SAS Data Connect Accelerator for Teradata – Full Parallel Load Example:

NOTE: This method is faster than a Multi-Node Load.

```
/* Create a default CAS session and create SAS libref */
cas mysess;
libname mycas cas caslib=casuser;

/* Use Parallel loading */
caslib tdlib datasource=(
            srctype="teradata",
            username="myusr1",
            password="mypwd1",
            server="tdserver",
            database="sas",
            dataTransferMode="parallel"
);

proc casutil;
  load
    incaslib="tdlib"    casdata="testdata"
    outcaslib="casuser" casout="testdata";
run;
```

In the SAS log you can see the Parallel Load was performed successfully:

```
NOTE: Performing parallel LoadTable action using SAS Data Connect
Accelerator for Teradata.
```

These are additional resources regarding Muti-Node and Parallel Loading in Viya:

Five approaches for High Performance Data Loading

What's New in SAS Data Connectors for SAS Viya

TECH TIP #5: TAKE ADVANTAGE OF THE BENEFITS OF CLOUD AGILITY

One of the benefits of being in the cloud is the ability to scale easily and quickly. Some of the best examples of how to utilize cloud's agility is to quickly spin up environments for:

- Dev / Test
- Personal Development Sandboxes
- Special Projects
- Scaling up systems for additional "Seasonal" use
- Proof Of Concept workspaces

You can leverage products found in the Cloud Market Places to streamline setup and configuration for these kinds of environments. Two great resources to use are the SAS Quick Starts and the Teradata Vantage Developer version.

The SAS Quick Starts are easy to provision and up and running in less than 2 hours. The Teradata Vantage Developer version is also easy to spin up and can be provisioned in about 30 minutes. So, in less than 3 hours, you can have a new analytic environment to use.

For more information or to create an instance, check out the links below:

SAS Quick Starts*:

AWS: SAS 9.4 Grid Manager, SAS Viya Quick Start

Azure: SAS Viya Quick Start

Google Cloud: SAS Viya Quick Start

*NOTE: These require SAS licenses

Teradata Vantage Developer Edition (Free**):

AWS: Teradata Vantage Developer

Azure: Teradata Vantage (Choose "Developer" Tier when prompted)

**NOTE: Teradata does not require a license nor change a fee for these instances. However, AWS & Azure do charge for storage & usage.

HOW TO USE CLOUD STORAGE WITH TRADITIONAL DATA

With more data moving to the cloud, we're seeing the need to combine more data from cloud storage, like S3, with traditional data from a database. Teradata Vantage has a new feature being released Q2 of 2020, that will allow you to combine these data sources seamlessly to perform your analytics.

USING NATIVE OBJECT STORE (NOS) DATA WITH TRADITIONAL DATA

Using Teradata Vantage, you can access JSON, CSV and Parquet files you have stored in AWS S3, and Azure Blob Storage. The way you accomplish this is to define a Foreign Table (FT) object in Teradata Vantage identifying the location of the file. Then you can treat the FT like any other table in the database. This allows you to view the data, join it to other tables, aggregate it, and perform many other SQL operations.

Syntax of a Foreign Table in Cloud Storage:

For JSON:

```
CREATE FOREIGN TABLE <Table_Name> (
      Location VARCHAR(2048) CHARACTER SET UNICODE CASESPECIFIC,
      Payload JSON(8388096) INLINE LENGTH 32000 CHARACTER SET UNICODE
)
USING (
      LOCATION ('<location of your file in cloud storage>'));
```

For CSV:

```
CREATE FOREIGN TABLE <Table_Name> (
      Location VARCHAR(2048) CHARACTER SET UNICODE CASESPECIFIC,
      Payload DATASET INLINE LENGTH 64000 STORAGE FORMAT CSV
)
USING (
      LOCATION ('<location of your file in cloud storage>'));
```

For Parquet:

```
CREATE FOREIGN TABLE <Table_Name>(
      Location VARCHAR(2048) CHARACTER SET UNICODE CASESPECIFIC,
      <<List column names in Parquet file, case specific>>
)
USING (
      LOCATION ('<location of your file in cloud storage>')
      STOREDAS ('PARQUET')
)
NO PRIMARY INDEX
PARTITION BY COLUMN;
```

Sample Use Case:

Background: A large retail company with approximately 1 million customers sells their products through 2 main channels: in their stores and online. During the last few years, the retailer has introduced 2 new offerings when purchasing online via their website or mobile app: 1) In Store Pickup 2) Same Day Delivery. This is in addition to the traditional option of making a purchase online & having it shipped to the customer's home.

Data: The retailer keeps demographic information for all their customers, as well as detailed transactional data, at the item level, for the current year in their Database.

To save money, the company recently moved their historical data for the last 5 years from their data lake to S3 Cloud Storage as Parquet files. The data is rolled up to the transaction level. It contains a customer id, and channel number for each transaction. One Parquet file contains one year of data.

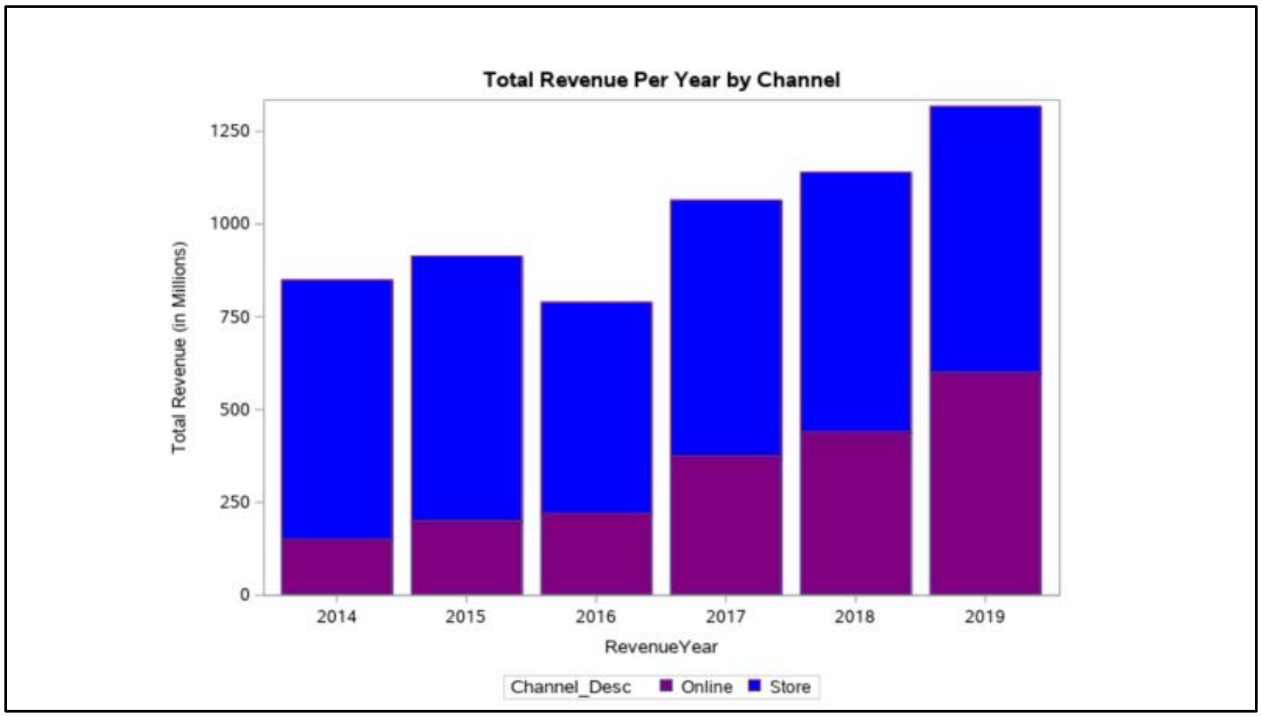
SQL statement to Create a Foreign Table to point to one year of Parquet data in S3:

```
CREATE MULTISET FOREIGN TABLE sas.cust_txn_hist_2014
(
      Location VARCHAR(2048) CHARACTER SET UNICODE CASESPECIFIC,
      custid BIGINT,
        Channel BYTEINT,
      TransactionAmt Integer)
USING (
      LOCATION  ('/S3/s3.amazonaws.com/mydata/cust_txn_2018')
      STOREDAS  ('PARQUET'))
NO PRIMARY INDEX
PARTITION BY COLUMN;

<repeat for 2015-2018>
```

Once you combine five years of historical data (for 2014-18) from cloud storage, together with the last full year of data (for 2019) in the database, you can start to look at the historical trends.

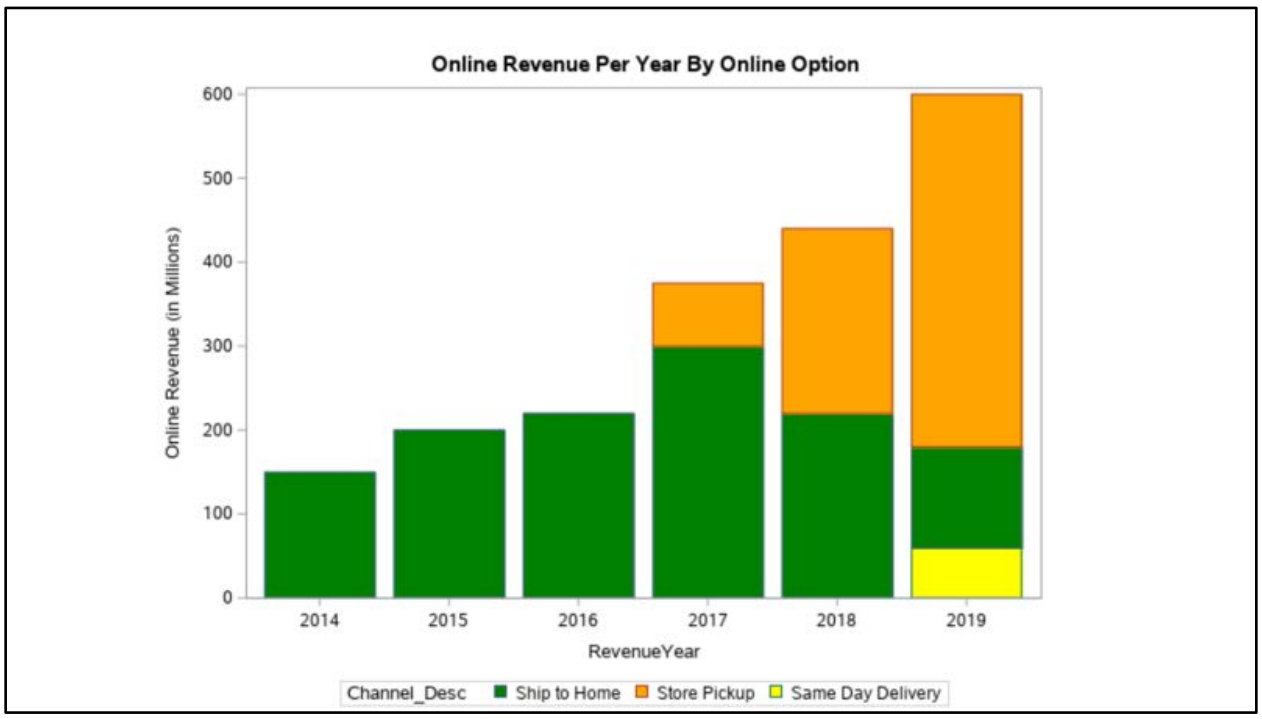
Graph #1 – **The total revenue for the last 6 years (online vs in store purchases)**



Observations: Online purchases have increased year over year; including in 2016 when Total Revenue was down.

Question #2: What trends are we seeing with the new online purchasing options?

Graph #2 – **Revenue for purchases originating online**



Observations: In 2017 customers started making purchases online for pickup in the store. This option has been gaining popularity each year, and now generates the most revenue. The trend to purchase

merchandise online and then have it shipped to a customer's home has been declining since the introduction of new online purchasing options. In 2019 customer started getting Same Day Delivery service.

Next Steps: Now that the retailer can see these trends, they can use the historical data to start doing additional customer targeted analysis (i.e. which customers are making what kinds of online purchases…etc). They can create campaigns for promotions or models to predict future spending or churn.

CONCLUSION

SAS and Teradata have had a strong partnership for over a decade, with focus on optimized analytic solutions leveraging In-Database processing and efficient data interchange. Extending this partnership into the cloud, we continue to capitalize on the strengths of both technologies to provide cutting-edge analytic solutions. We hope you enjoyed the paper and found its recommendations, resources, and tips helpful for your journey to the cloud.

ACKNOWLEDGMENTS

We would like to thank associates from the SAS® Performance Lab, Global Alliances, R&D Project Management, and Teradata® Cloud teams for their support. This paper would not have been possible without their assistance.

CLOUD TERMINLOGY

Virtual Private Cloud – For AWS & GCP, refers to a Virtual Private Cloud network created in their cloud.

Virtual Network – For Microsoft Azure Cloud, refers to a Virtual Network created within their cloud.

Geographic Region -

Availability Zone – For AWS & Azure, refers to a unique physical location within a Geographic Region

Single Tenant – an instance of a software application and supporting infrastructure which serves exactly one customer, not shared between customers.

Multi-Tenant - a single instance of software which runs on a server and serves multiple customers.

On-Premises –on-site at the individual or organization's location

Hybrid Configuration – Part of the system is in the cloud, while another part is on-premises

Multi Cloud – Refers to the distribution of cloud assets (software, storage…etc) across several cloud vendors, like AWS, Azure, Google Cloud.

S3 – This stands for (S3). It is Cloud Storage available in AWS. In this context of this paper, S3 storage can be interchanged with Azure Blob Storage.

CONTACT INFORMATION

Heather Burnette
Teradata Corporation
Heather.Burnette@Teradata.com
www.teradata.com/cloud