Welcome to the Case Study!

Today's Lecture

- SAS made a deal with Snowflake early on as a cloud storage partner – but you can use other environments (e.g., MS Azure)

- Using SAS Access to move data from one environment to another is not complex from a programming standpoint…

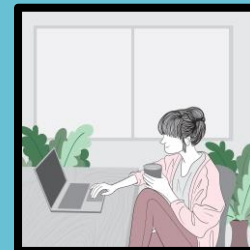- …but it is a **HUGE HEADACHE** from an engineering standpoint!

Try all of our career-building data science resources!

Join our group online data science mentoring program: https://buff.ly/3Sw4Fsa

Take our exclusive online data science courses aimed healthcare analysts: https://monika-s-school-e295.thinkific.com

Watch our tutorials on YouTube: https://buff.ly/2W6ijuc

Take our LinkedIn Learning courses: https://buff.ly/2Ihd4Rq

Why Are We Talking About Snowflake?

- **What if your SAS server gets "full"? What do you do?**
- **US Army: I had a fire-proof safe, and I stored archived data in it.**
- **Today's solution: Cloud storage.**
- **But how with SAS?**

An Insider's Guide to SAS/Access Interface to Snowflake

By Jeff Baily
https://buff.ly/3NYUSKz

SAS Global Forum 2020

# The Main Challenge Moving SAS Server Data to Cloud Storage

- I call it, "Carrying a rock up a ladder".

- SAS data are already heavy – like a rock.

- You need to get the data out of the SAS environment, and into the Snowflake environment

- There are a lot of hand-offs along the way

Internet and

snowflake

SAS Data

SAS Server

White Paper: Bulk Loading

- Data step language is not good for processing that needs to be threaded - like "bulk loading"

- SQL, however, is

- From white paper: "At the time I am writing this (February 2020), the only object store supported for SAS/ACCESS Interface to Snowflake bulk loading is AWS S3"
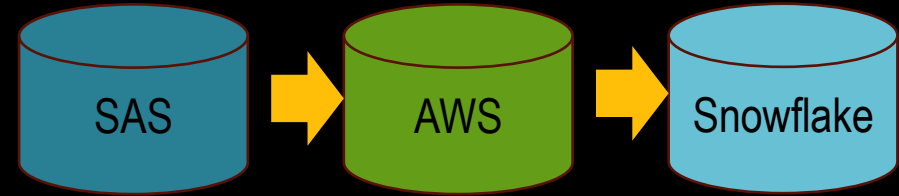
- AWS (Amazon Web Services) is a platform that lets you load big data and run ML algorithms (among other things)

- So the white paper suggests you are passing data from your SAS server through AWS's server to Snowflake. Really?

This might not be ready for prime time...

- From White Paper

- SAS's tools have good I/O in the SAS environment, but terrible I/O outside of it.

- AWS (or the bulk loading function) is basically optimizing I/O (see graph)

- Author compares "bulk load" function to "insert" function

JUST TELL ME WHAT TO DO!

Reading From Snowflake

When reading data from Snowflake using SAS, you should always start by setting the READBUFF= option to a number close to the maximum value (32767). I typically set READBUFF=32000 because it is close to the max and easy to remember. I have not seen a large performance difference between 32000 and 32767. Please understand, you might see a difference – so experiment.

> **Pro Tip:** Always set READBUFF= to a high number when you are starting. If you want to optimize read performance, experiment to find the best value for your situation. I set the READBUFF= option in the LIBNAME statement and use the data set option to override.

INSERT VS. BULK LOAD

With small data sets, there is not a huge benefit to using bulk loading. In fact, with small data sets, bulk loading might be slightly slower. I tend to argue that the difference is so small you might as well use bulk loading because the data set could grow large enough to cause problems with future INSERT statements.

I have experimented with INSERT statements vs bulk load using an on-premises version of SAS Viya 3.5 running on a Linux machine. If you are running SAS in AWS, you can expect different numbers, but the exercise is valid. Use this exercise to determine the point at which bulk loading performs best.

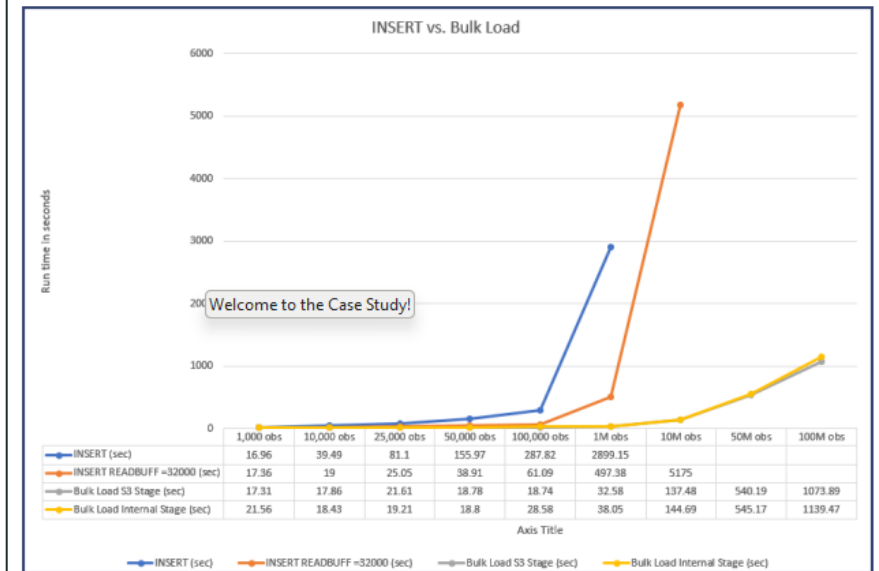Figure 1 shows the relative performance of various writing techniques.

	1,000 obs	10,000 obs	25,000 obs	50,000 obs	100,000 obs	1M obs	10M obs	50M obs	100M obs
INSERT (sec)	16.96	39.49	81.1	155.97	287.82	2899.15			
INSERT READBUFF =32000 (sec)	17.36	19	25.05	38.91	61.09	497.38	5175		
Bulk Load S3 Stage (sec)	17.31	17.86	21.61	18.78	18.74	32.58	137.48	540.19	1073.89
Bulk Load Internal Stage (sec)	21.56	18.43	19.21	18.8	28.58	38.05	144.69	545.17	1139.47

Figure 1. Comparison Graph of INSERT statements versus Bulk Load

HOW DO I MAKE MY QUERIES RUN FASTER?

This paper does not cover query tuning. Other papers (see "The SQL Tuning Checklist: Making Slow Queries a Thing of the Past" in the References section) cover this topic. One of the first steps in tuning a query is to find out what SAS is asking the database to do.

If You Need to Move SAS Server Data to the Cloud, You Will Have I/O Problems

- Technology solutions only go so far – we might need more creative management solutions

- Do we really need all these data anyway for what we are doing? Do we need to move ALL the data into the cloud?

- Can we somehow stay out of the cloud (e.g., preserve our SAS server environment)?

- And what about today? Should a "new SAS shop" just start in the cloud to begin with (and be ultimately compatible with SAS Viya)?
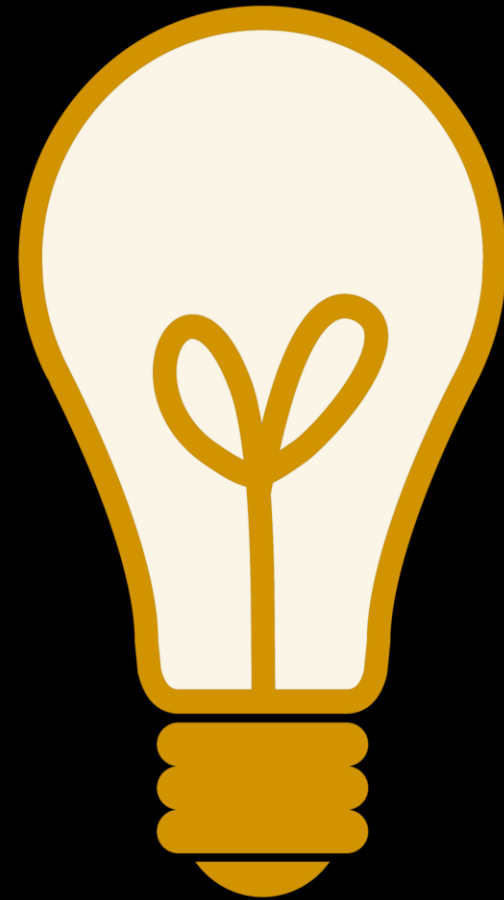
Questions?

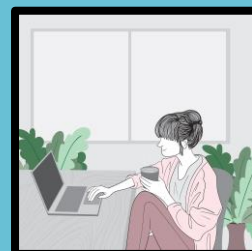**Facilitator:** Monika M. Wahi, MPH, CPH

Email:
dethwench@gmail.com

Blog:
https://dethwench.com/blog/

LinkedIn:
https://www.linkedin.com/in/dethwench/

Try all of our career-building data science resources!

Join our group online data science mentoring program:
https://buff.ly/3Sw4Fsa

Take our exclusive online data science courses aimed healthcare analysts:
https://monika-s-school-e295.thinkific.com

Watch our tutorials on YouTube:
https://buff.ly/2W6ijuc

Take our LinkedIn Learning courses: https://buff.ly/2lhd4Rq