A Hybrid On-premises and Public Cloud Attention Clustering Workflow

James McCombs, Alan Walsh, Takuya Noguchi
Preetesh Kantak †

Research Technologies and Kelley School of Business †
Indiana University

Presentation Outline

1. Motivation
2. Research Overview and the Attention Clustering Problem
3. The Data Management Problem
4. On-Premises Solution
5. Drawbacks of the On-Premises Solution
6. Viable Google Cloud Options
7. Hybrid Cloud and On-Premises Workflow
8. Funding of Cloud Resources
9. Conclusion

Motivation

- Research IT professionals are increasingly confronted with how to provide storage and analysis of very large data sets for researchers
- With a myriad of on-premises and cloud options available, it is not always clear which is most technically feasible or cost effective
  - Development of on-premises solutions can be very labor intensive
  - Shared resources can reduce cost but affect availability/performance
  - Cloud resources can be expensive or restrictive
- We wanted to examine these factors in the context of a challenging use case to better inform research IT professionals who will face similar challenges

Research Overview

- The data to be analyzed is derived from on-line business and financial articles and which companies have accessed those articles
  - associates a company to various topics it has taken an interest in
  - provides a score gauging company's level of interest in a topic
- The research goal is to predict asset price fluctuations based in part on this data
- The company-topic associations are generated on a weekly basis
- 25 TB backlog of data to store and analyze
- More data generated each week

Attention Clustering Problem

- Topics are categorized as general interest and industry specific
- We want the top N topics that cover a certain percentage of industry-specific topics
- We need to compute the following statistics to perform the ranking:

$$F_j = \left[\sum_{k=1}^{K} \mathbb{I}_{(x_{k,j} > x_m)}\right] \qquad F_j^S = \left[\sum_{k=1}^{K_S} \mathbb{I}_{(x_{k,j} > x_m)}\right]$$

F_j - Number of times topic *j* occurs with score $x_{k,j} > x_m$ over all companies

F_j^S - Number of times topic *j* occurs with score $x_{k,j} > x_m$ within an industry or subindustry

The Data Management Problem

- Substantial storage and compute resources needed for the analysis
- New data needs to be ingested weekly and in a timely manner to keep up
- The data needs to be formatted and organized for efficient storage and analysis

Raw Data Schema

Field	Type	Description
Date	INTEGER	Date (same for entire week)
Company	STRING	Name of company
Domain	STRING	Domain name
Size	STRING	Size category of company
IndustrySubindustry	STRING	Industry & Subindustry of company
Topic	STRING	Topic name
Category	STRING	General category of topic
City	STRING	City/Metro area of company
Score	INTEGER	Composite score for company-topic

On-Premises Solution

- Bound by the following constraints:
  1. Researcher has limited funding resources
  2. Limited to community-available systems and virtual enterprise platforms
     - No dedicated, scalable analytics platform
  3. Limited time to build a robust workflow

- Considered the following software systems for implementing the workflow
  - Spark for ingestion and data preparation
  - MySQL RDBMS
  - Apache HIVE or other horizontally scalable NoSQL system

On-Premises Resources

Cost-Free Resources			
System	CPU	RAM	# Nodes
Karst	2 8-core Xeon E5-2650	32GB / 64 GB	256 / 16
Carbonate	2 12-core Xeon E5-2680	256GB / 512GB	72 / 8
Research Data Complex	4 vCPU	96GB	1

Paid Resources	
System	Cost per year (minus storage costs)
Carbonate condominium nodes	$8160 (10 hr limit per month)
Intelligent Infrastructure	$25,000 +

Intelligent Infrastructure Estimates

Mid-Size Option		
	Config	Cost
vCPU	12	$1,140
RAM	256 GB	$3,200
Storage	35,840 GB	$26,880
Backup	53,760 GB	$21,504
Setup	1	$952
ELA	1	$1,428
Total		**$55,104**

Large Option		
	Config	Cost
vCPU	24	$2,280
RAM	256 GB	$6,400
Storage	35,840 GB	$26,880
Backup	53,760 GB	$21,504
Setup	1	$952
ELA	1	$1,428
Total		**$59,444**

On-Premises Solution

- We selected Carbonate for ETL and the RDC for analysis and queries
- We used Spark to perform the ETL
  - apply “snowflake” schema to reduce redundancy of data in RDBMS
- Spark workflow is a multiphase process coordinated with the RDBMS

RDBMS Schema

Industry dimension table	
Field	Specification
industry_id	INT UNSIGNED AUTO_INCREMENT PRIMARY KEY
industry	VARCHAR(255)
sub_industry	VARCHAR(255)
UNIQUE KEY(industry, sub_industry)	

Topic-category dimension table	
Field	Specification
topic_id	SMALLINT UNSIGNED AUTO_INCREMENT PRIMARY KEY
topic_category	VARCHAR(255)
topic_name	VARCHAR(255)
UNIQUE KEY(topic_category, topic_name)	

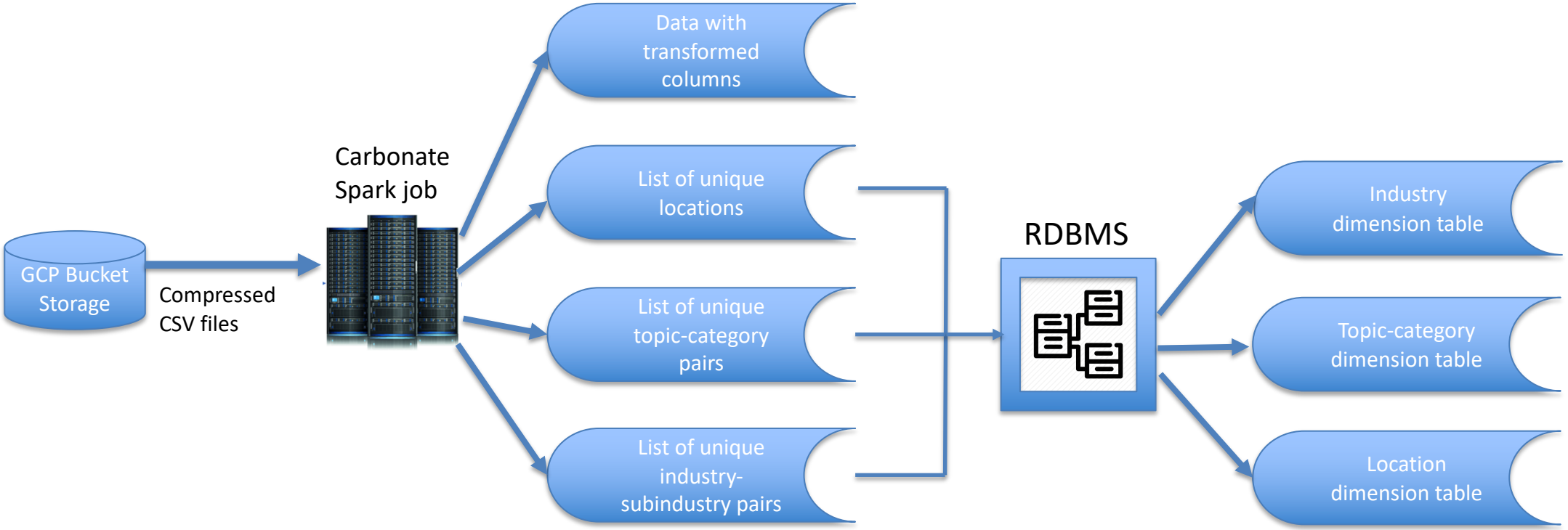
Location dimension table	
Field	Specification
location_id	SMALLINT UNSIGNED AUTO_INCREMENT PRIMARY KEY
location_city	VARCHAR(255)
sub_industry	VARCHAR(255)
Unique Key(location_city)	

RDBMS Schema

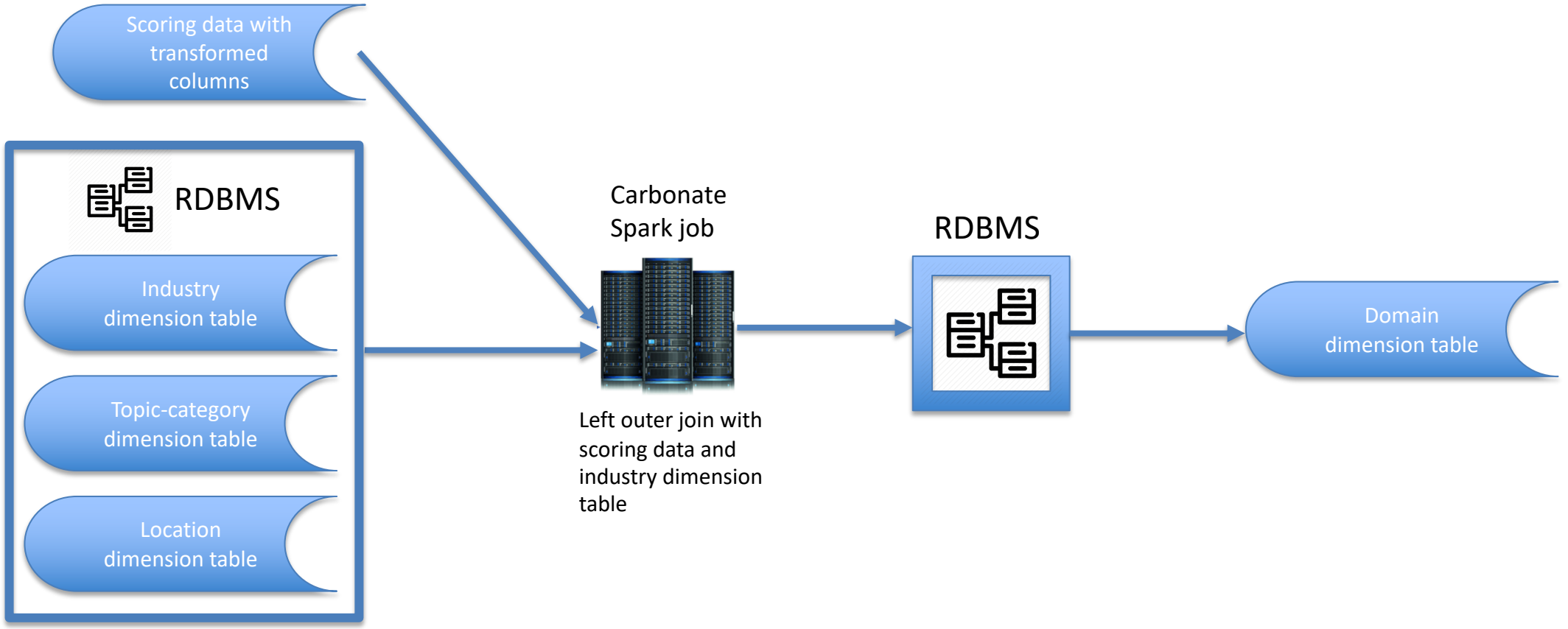
Domain dimension table	
Field	Specification
domain_id	INT UNSIGNED AUTO_INCREMENT PRIMARY KEY
domain	VARCHAR(255)
industry_id	VARCHAR(255)
size_id	TINYINT UNSIGNED
Unique Key(domain)	

Composite data table	
Field	Specification
date_id	INT UNSIGNED
domain_id	INT UNSIGNED
topic_id	SMALLINT UNSIGNED
composite_score	TINYINT UNSIGNED
PRIMARY KEY(date_id, domain_id, topic_id)	

Spark-RDBMS ETL Workflow (Phase 1)



Spark-RDBMS ETL Workflow (Phase 2)



Drawbacks of On-Premises Solution

- The Spark-RDBMS workflow had some significant drawbacks
  - Needed at least 9 Carbonate nodes to process an entire week
    - Queue wait times were as much as three days
  - More work needed to be done to perfect and fully automate
    - Checkpointing
    - Improving how data exchanged between Spark and RDBMS
- More storage would need to be purchased for the RDC
  - One week of data was 197GB under the schema

We began investigating Google's cloud research credits program as an alternative

Viable Google Cloud Options

- $20K in credits awarded

Cloud-based Spark ETL and SQL

Google Cloud Dataproc

# nodes	17
Storage rate (GB month)	$0.040
Disk capacity per node	256 GB
Storage cost per year	$2,089
Estimated hours (ETL only)	600
On-demand rate for 16 vCPU and 128GB RAM node	$0.72312
Hardware costs	$7,375
Service charge	$1,632
Total cost	$11,096

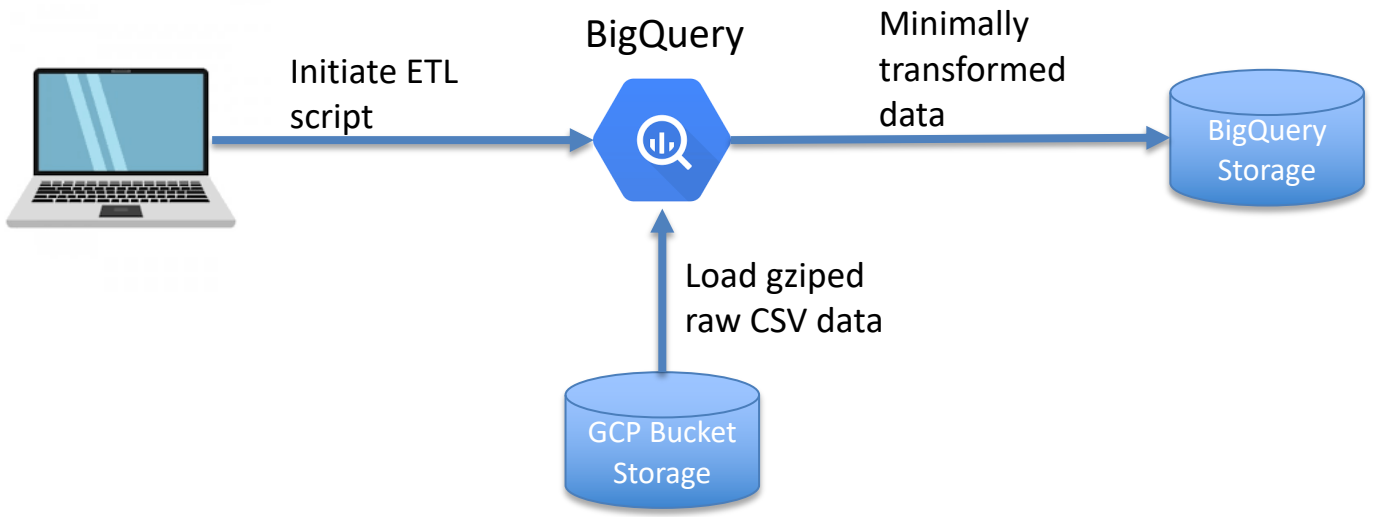
Google Cloud SQL

Storage	30 TB max
# vCPUs	16
Estimated hours (5 days/wk 12hr/day)	3120
Total cost per year	$74,040

Google BigQuery

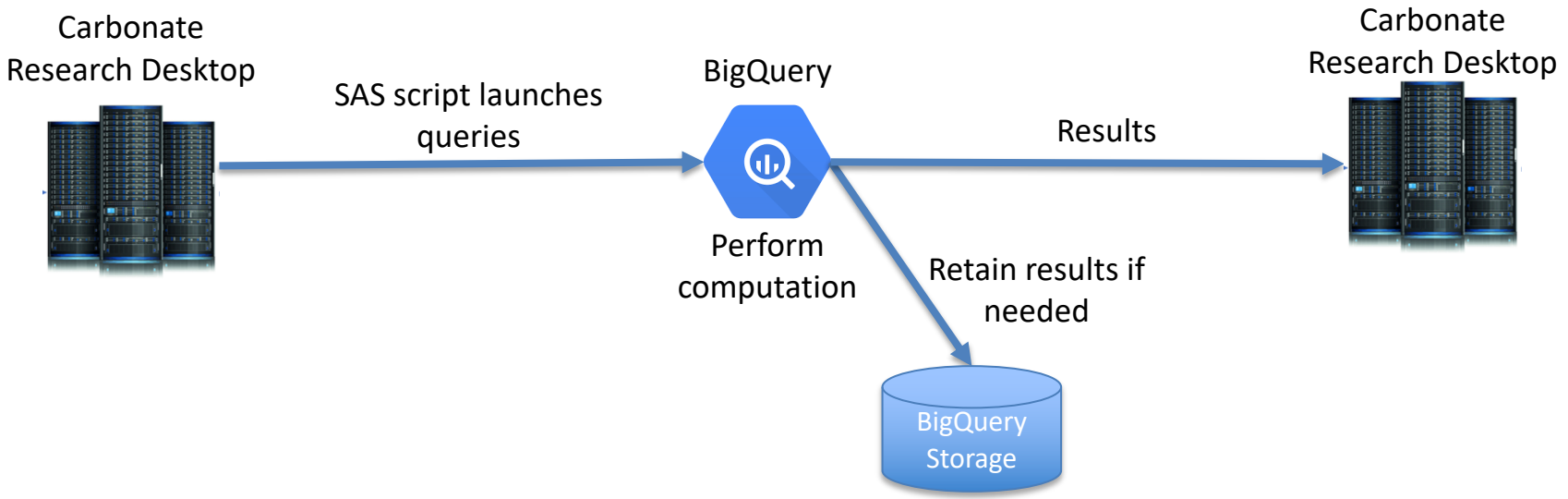
Storage cost (GB month)	$0.02
Cost for 35 TB Per year	$8,602
Query costs	$5.00 / TB queried
Budget available for queries	$13,398

BigQuery Data Ingestion Workflow



BigQuery Analysis Workflow

- Researcher has a preference for using SAS software for analysis
- Carbonate Research Desktop graphical environment runs SAS on-premises
- Installed BigQuery SAS plugin for direct access to BigQuery



Funding of Cloud Resources

- The GCP academic credits made the research possible on GCP
- BigQuery pricing made for efficient use of credits
  - Queries are charged based on amount of data touched, not computation
  - Queries can be crafted to reuse cached results
  - Computation on BigQuery near data reduces need for data transfer
- Still have $8,000 in research credits, research almost complete
  - Enough to produce a solid publication

Conclusions

- Research IT professionals must consider cloud and on-premises resources
  - Limited funding is a good reason to stay on-premises
  - Time constraints and the cost of developing a robust on-premises workflow can make the cloud more appealing
- Google BigQuery provided a technically robust and cost-efficient solution
  - Workflow development was rapid and easy
  - Efficient use of credits has enabled excellent progress
- Google research credits enabled the researcher to nearly complete his work
  - Results are being used to apply to for external funding