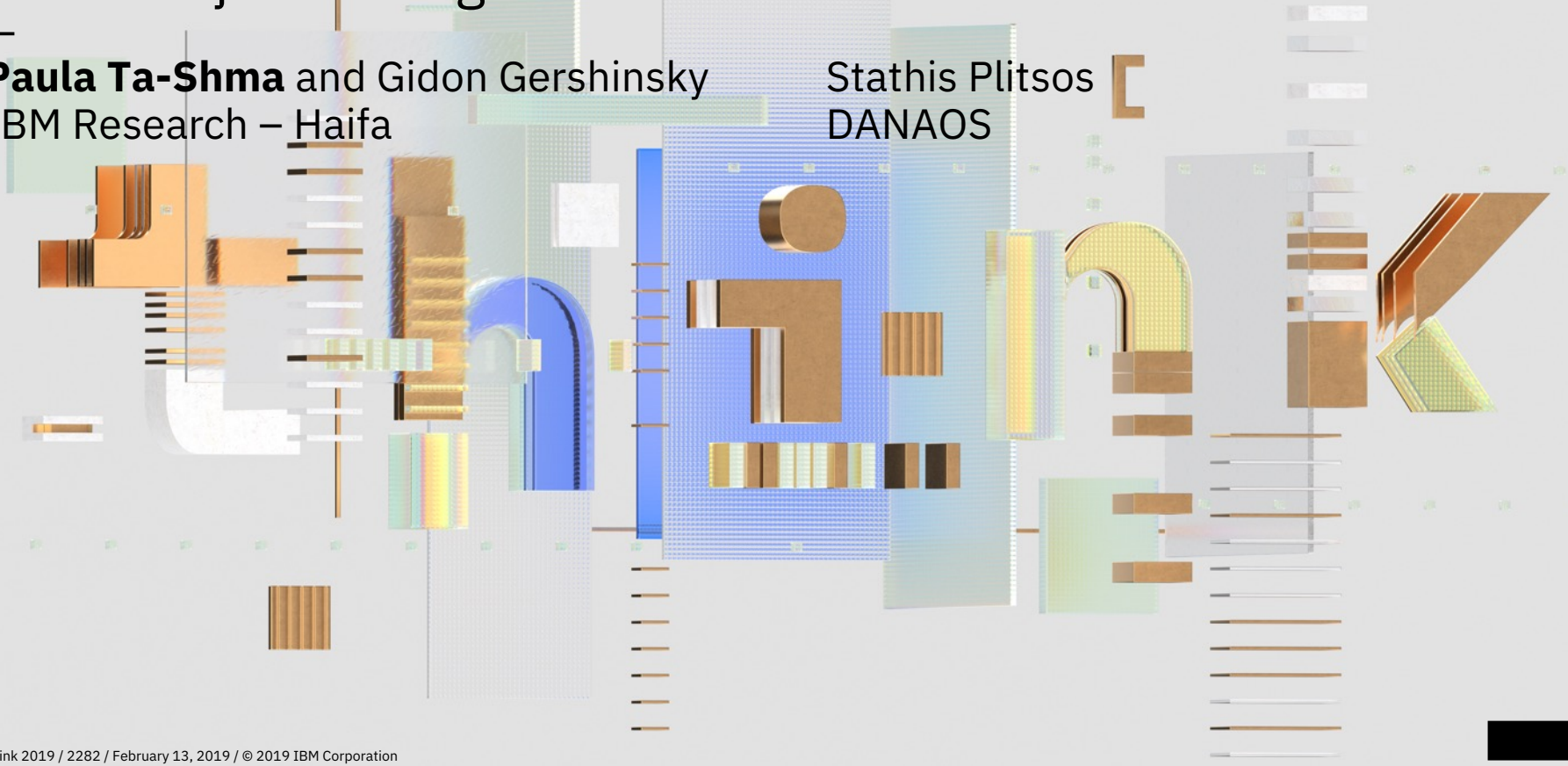
Enterprise-Scale Analytics Performance with Cloud Object Storage

think 2019

Paula Ta-Shma and Gidon Gershinsky
IBM Research – Haifa

Stathis Plitsos
DANAOS



Please note

IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice and at IBM’s sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

danaos

one of the **largest independent owners of modern, large-size containerships**

Established in 1972

59 Container Vessels

Range from 2,200 TEU to 13,100 TEU

BigDataStack

Holistic stack for big data applications and operations

- charter containerships on long-term contracts at fixed rates to many of the world's largest liner companies
- distinct edge in **advanced shipping technology** and long track record of safety, efficiency, and environmental responsibility
- Danaos charters vessels to a diverse group of liner companies including many of the largest

DANAOS Use Case

Efficient fuel consumption is a high priority for DANAOS

- while meeting environmental constraints

DANAOS **data scientists** need to compare the fuel consumption of the various vessels

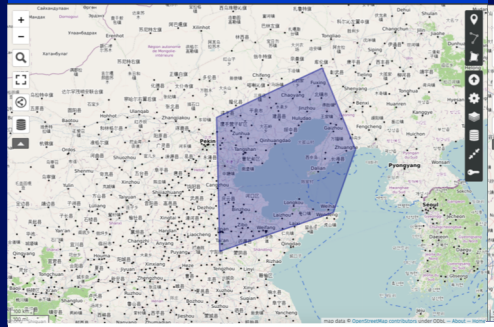
– For similar vessel conditions:

  - **Weather**, speed through water, draft

– Take particular interest in **Sulfur Emission Control Areas** (SECAs) where low sulfur fuel is mandatory

  - More environmentally friendly
  - More expensive

Bohai Rim

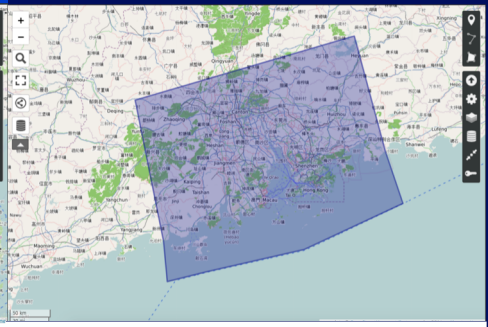


2 columns | 2 rows

WEATHERCONDITION	RELATIVEFUELCONSUMPTION
Good	1.0
Meduim	1.0256715493509159

weather	relativefuel
good	1.0
medium	1.0257

Pearl River Delta



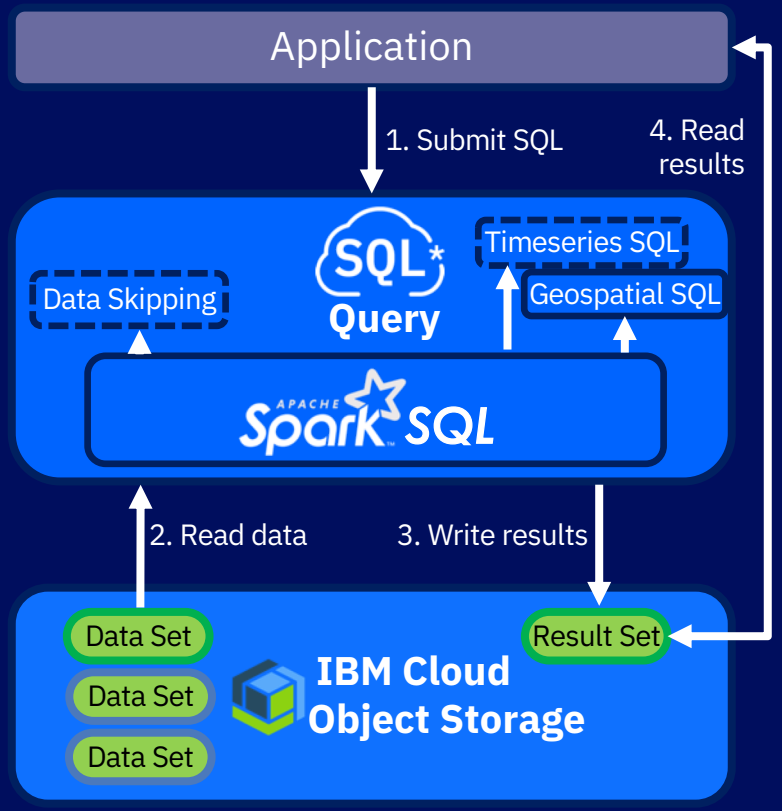
2 columns | 2 rows

WEATHERCONDITION	RELATIVEFUELCONSUMPTION
Good	1.0
Meduim	1.6633965972823266

weather	relativefuel
good	1.0
medium	1.6634

IBM Cloud SQL Query

- **Serverless Analytics**
- **On Structured Data in IBM Cloud Object Storage**
- **Pay per query**
- **$5/TB scanned**
- **Supports JSON, CSV, Parquet, ORC, Avro**



What is Object Storage ?

Storage of choice for big datasets in the cloud

- High capacity, low cost

Objects are like files but:

- Written once and cannot be updated
- No rename operation

Accessed through REST API

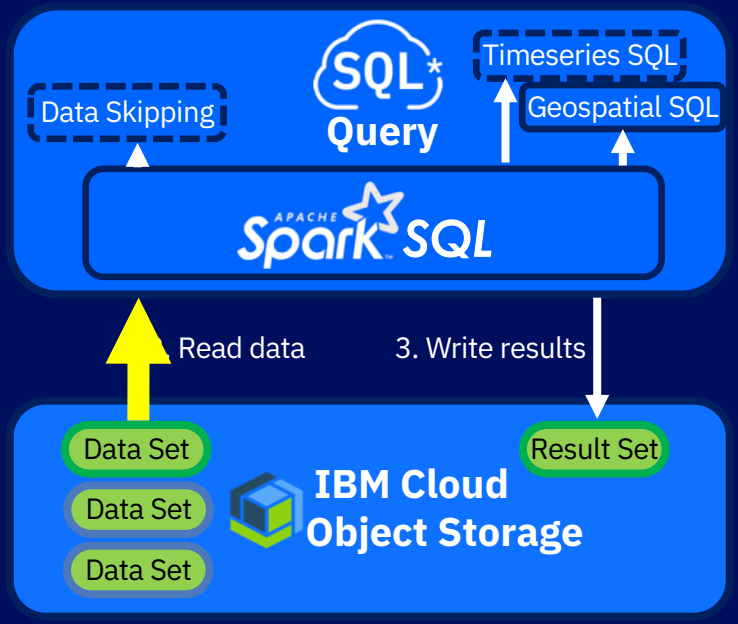
- PUT/GET/POST/DELETE object/bucket
- Flat namespace

Analytics works best on equally sized objects

Examples: IBM COS, Amazon S3, Google Cloud Storage, OpenStack Swift

Cost and Performance Depend on #Bytes Scanned

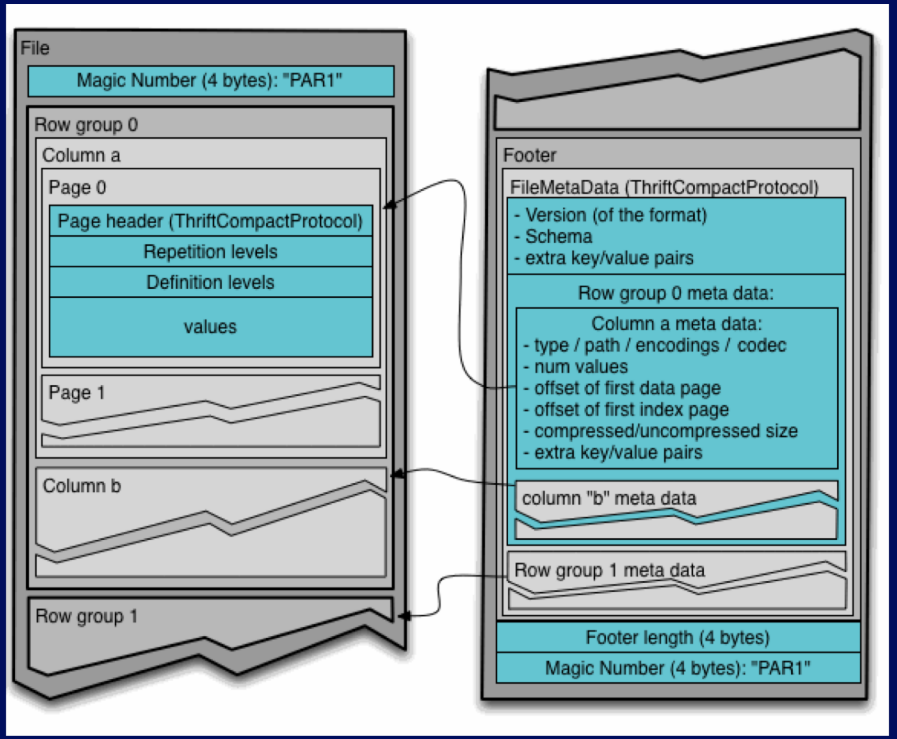
$5/TB scanned



Minimize bytes scanned : Best Practice 1

Use Parquet

- Column based
  - Only read the columns you need
  - Column wise compression
  - Schema and other metadata stored in object footer



Minimize bytes scanned : Best Practice2

Use Hive style partitioning

GPMeterStream/**dt=2017-08-17**/part-00085.csv

GPMeterStream/dt=2017-08-17/part-00086.csv

GPMeterStream/dt=2017-08-17/part-00087.csv

GPMeterStream/dt=2017-08-17/part-00088.csv

GPMeterStream/dt=2017-08-17/part-00089.csv

GPMeterStream/dt=2017-08-18/part-00001.csv

GPMeterStream/dt=2017-08-18/part-00002.csv

GPMeterStream/dt=2017-08-18/part-00003.csv

- **Avoid reading unnecessary objects altogether**
- **Technique has limitations**

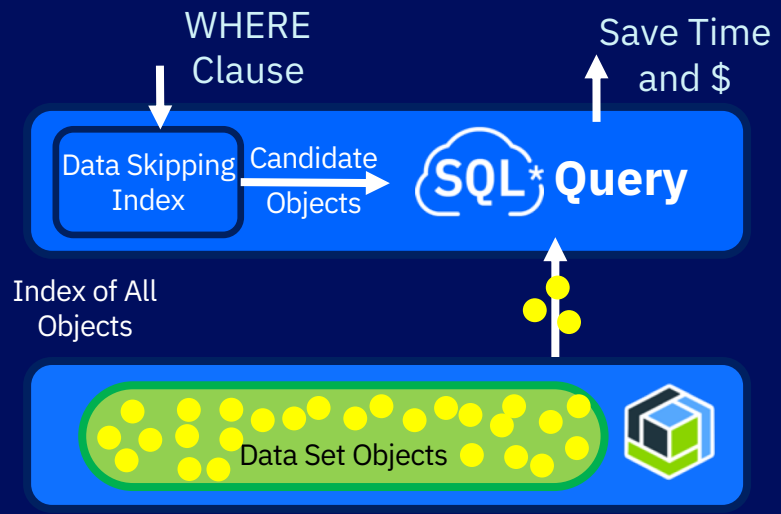
Data Skipping

Determine which objects are NOT relevant to a SQL query using a *data skipping index*

Stores and indexes summary metadata for each object

Skip over irrelevant objects to reduce bytes scanned

Saves time and $



Example: Look for data in violent storm conditions

```
SELECT vessel_code, datetime, longitude, latitude,
wind_speed
FROM cos://us-south/.../danaos stored  as parquet
WHERE wind_speed > 30
```

Data Skipping Indexes

- Store summary metadata per object column and index it
  - Much smaller than data (unlike DB indexes)
- **Various index types :**
  - Min/max, value lists, geospatial
  - Users choose columns to index and index types
- Filter entire objects according to this metadata (without accessing COS)
- **Applies to all SQL Query supported formats**
  - e.g. JSON, CSV, Parquet, ORC etc.

Data

(COS object listing)

GPMeterStream/dt=2017-08-17/part-00085.csv

GPMeterStream/dt=2017-08-17/part-00086.csv

GPMeterStream/dt=2017-08-17/part-00087.csv

GPMeterStream/dt=2017-08-17/part-00088.csv

GPMeterStream/dt=2017-08-17/part-00089.csv

GPMeterStream/dt=2017-08-18/part-00001.csv

GPMeterStream/dt=2017-08-18/part-00002.csv

GPMeterStream/dt=2017-08-18/part-00003.csv

Metadata

(summary metadata per object)

```
{
  "name": "GPMeterStream/dt=2017-08-17/part-00088.csv",
  "metadata": {
    "location": [
      {
        "lat": 47.5,
        "lon": 4.2
      },
      ...,
      {
        "lat": 47.6,
        "lon": 3.4
      }
    ],
    "city": {
      "set": [
        "Kilstett",
        ...,
        "Haussignémont"
      ]
    },
    "temp": {
      "min": 7.97,
      "max": 26.77
    }
  }
}
```

Geospatial

Value list

Min/max

Indexing DANAOS Data

- Demo uses sample containing 6 vessels only
- Materialized view contains vessel data, main engine data and weather data
- Sample is ~1 GB in Parquet format
- 134 columns, ~7 million rows, 64 objects
- Layout in object storage according to geospatial coordinates
- **Build a data skipping index**

Creating a data skipping index for DANAOS

```
CREATE METAINDEX
MINMAX FOR wind_speed,
VALUELIST FOR vessel_code,
GEOSPATIAL FOR latitude, longitude
ON cos://us-south/../danaos STORED AS PARQUET
```

Query 1

SELECT vessel_code, datetime, longitude, latitude, wind_speed

FROM cos://us-south/.../danaos stored as parquet

WHERE wind_speed > 30

Skips 63 out of 64 objects

Retrieves vessel information when the wind speed is over 30 m/s.

This indicates **violent storm** conditions.

Creating a data skipping index for DANAOS

CREATE METAINDEX

MINMAX FOR wind_speed,

VALUELIST FOR vessel_code,

GEOSPATIAL FOR latitude, longitude

ON cos://us-south/../danaos STORED AS PARQUET

Query 1

SELECT vessel_code, datetime, longitude, latitude, wind_speed

FROM cos://us-south/.../danaos stored as parquet

WHERE wind_speed > 30

Skips 63 out of 64 objects

~ 1/64 of the cost!

Retrieves vessel information when the wind speed is over 30 m/s.

This indicates **violent storm** conditions.

Creating a data skipping index for DANAOS

CREATE METAINDEX

MINMAX FOR wind_speed,

VALUELIST FOR vessel_code,

GEOSPATIAL FOR latitude, longitude

ON cos://us-south/../danaos STORED AS PARQUET

Query 2

SELECT latitude, longitude

FROM cos://us-south/.../danaos stored as parquet

WHERE vessel_code="7" **AND**

datetime between to_date('2017-07-01') AND to_date('2017-07-02')

ORDER BY datetime

Skips 18 out of 64 objects

Creating a data skipping index for DANAOS

CREATE METAINDEX

MINMAX FOR wind_speed,

VALUELIST FOR vessel_code,

GEOSPATIAL FOR latitude, longitude,

ON cos://us-south/../danaos STORED AS PARQUET

Geospatial Data Skipping

- Run geospatial queries on CSV, Parquet, JSON etc. data in COS
- SQL Query is integrated with IBM's geospatial toolkit
- Includes functions for distance, area, intersections, bounding boxes etc.
- Boost performance and lower cost with data skipping indexes
- Reduce the number of function invocations as well as the bytes scanned

Example Geospatial Toolkit Functions

- **ST_Contains**
- **ST_Distance**
- **ST_Point -** Returns the point with the specified longitude and latitude values in degrees
- **ST_WKTToSQL** - Constructs geometry objects from an input character string that contains well-known text (WKT) representations of geometries

Query 3

SELECT distinct vessel_code

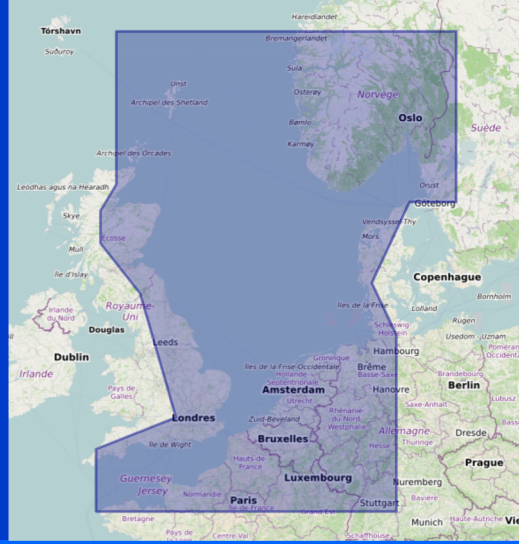
FROM cos://us-south/.../danaos stored as parquet

WHERE
ST_Contains(**ST_WKTToSQL**('POLYGON((-4 62, 13 62, 13 57.735556, 10.65747 57.735556, 8.8 55.5, 10 54, 10 48.5, -5 48.5, -5 50.5, -1 51.5, -2.8 55.2, -4.8 56.6, -4.8 57.5, -4 58.2))'), **ST_Point**(longitude, latitude))

Skips 50 out of 64 objects

Which vessels crossed the North Sea Sulfur Emission Control Area?

Here ships should not use fuel with a sulfur content in excess 1.5% m/m



Creating a data skipping index for DANAOS

CREATE METAINDEX

MINMAX FOR wind_speed,

VALUELIST FOR vessel_code,

GEOSPATIAL FOR latitude, longitude

ON cos://us-south/../danaos STORED AS PARQUET

Query 4 (simplified version)

SELECT weatherCondition, relativeFuelConsumption()

FROM cos://us-south/.../danaos stored as parquet **JOIN** weatherConditionsTable

WHERE **ST_Contains**(**ST_WKTToSQL**('POLYGON((114.624327 23.952784, 114.999264

...

))') , **ST_Point**(longitude, latitude))

GROUP BY weatherCondition

ORDER BY relativeFuelConsumption()

Skips 58 out of 64 objects

Compute the relative fuel consumption in the Pearl River Delta Sulfur Emission Control Area under different weather conditions

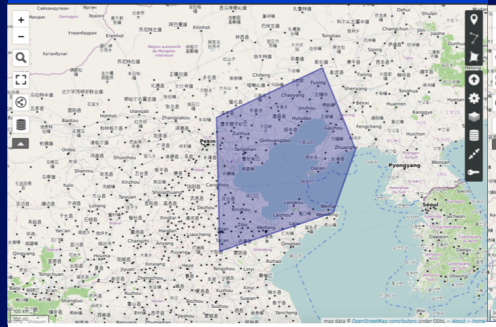
Relative Fuel Consumption() calculates the ratio of the fuel consumption to that for 'Good' weather conditions

weatherConditionsTable

weatherCondition	Wind Speed
Good	<10
Medium	[10,17)
Bad	[17,24)
Storm	>=24

DANAOS Query Results

Bohai Rim

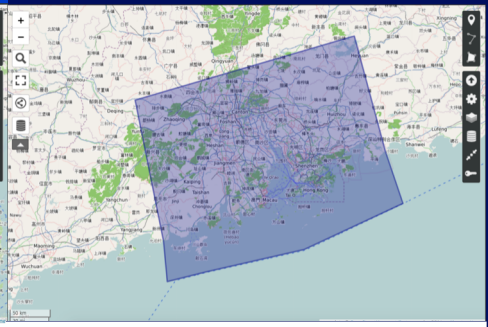


2 columns | 2 rows

WEATHERCONDITION	RELATIVEFUELCONSUMPTION
Good	1.0
Meduim	1.0256715493509159

weather	relativefuel
good	1.0
medium	1.0257

Pearl River Delta



2 columns | 2 rows

WEATHERCONDITION	RELATIVEFUELCONSUMPTION
Good	1.0
Meduim	1.6633965972823266

weather	relativefuel
good	1.0
medium	1.6634

Refresh

If data is added to a dataset after an index is created, the new data will not be skipped

Periodically refresh the index:

```
REFRESH METAINDEX
ON cos://us-south/.../danaos STORED AS parquet
```

Only updates the index for objects which changed since the last CREATE/REFRESH

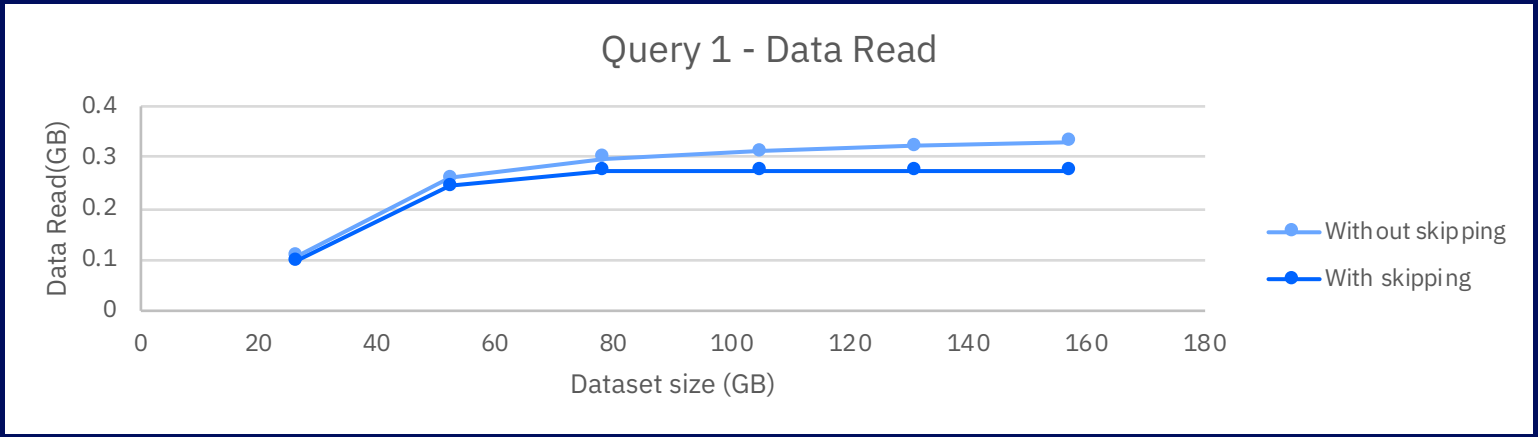
Delete

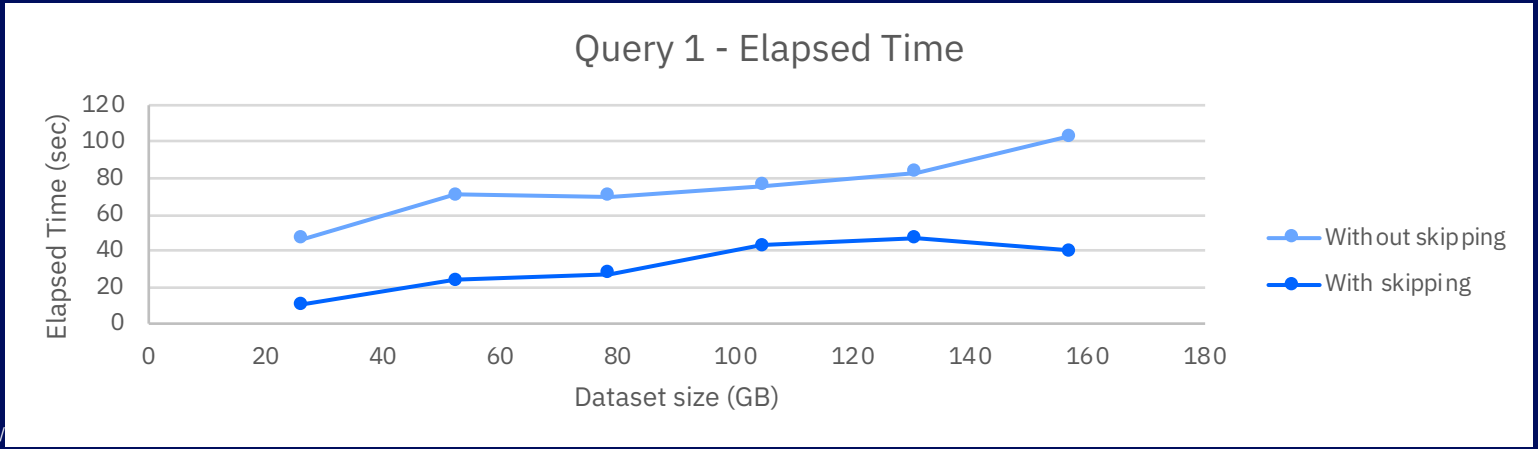
```
DROP METAINDEX
ON cos://us-south/.../danaos STORED AS parquet
```

Data Skipping Performance in SQL Query

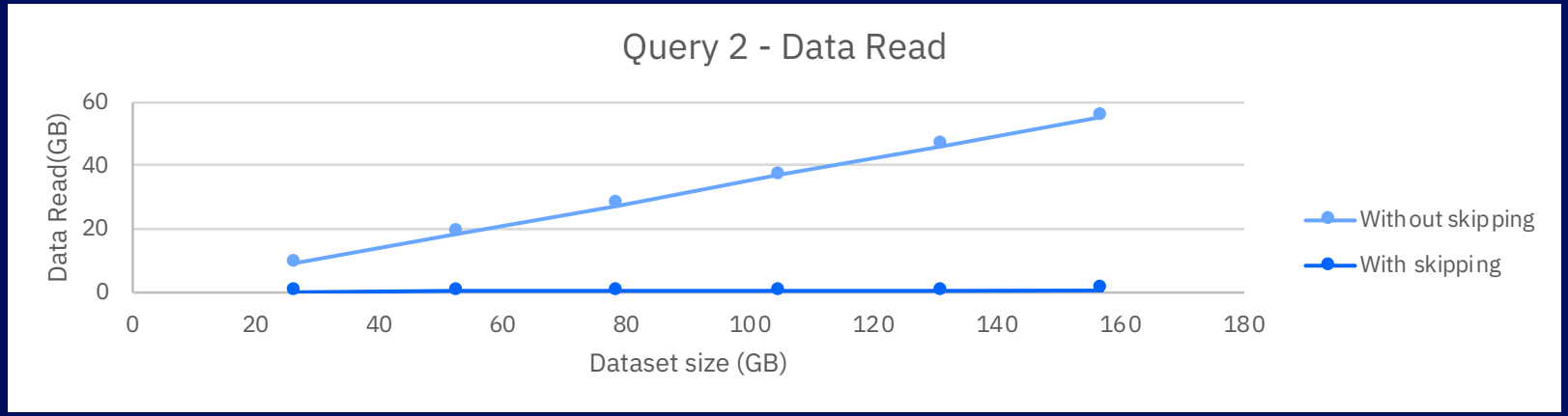
- To test larger datasets we used the metergen generator developed by Gridpocket
- Dataset sizes range from 26 GB to 157 GB in Parquet format
- We tested 3 queries similar to those from the Danaos use case
  - Query 1: exploits a minmax index (temp > 30)
  - Query 2: exploits a value list index (city = ‘Vidauban’)
  - Query 3: exploits a geospatial index (neighbors within 1km)
- We ran queries 10 times and took the average running time
  - With and without data skipping

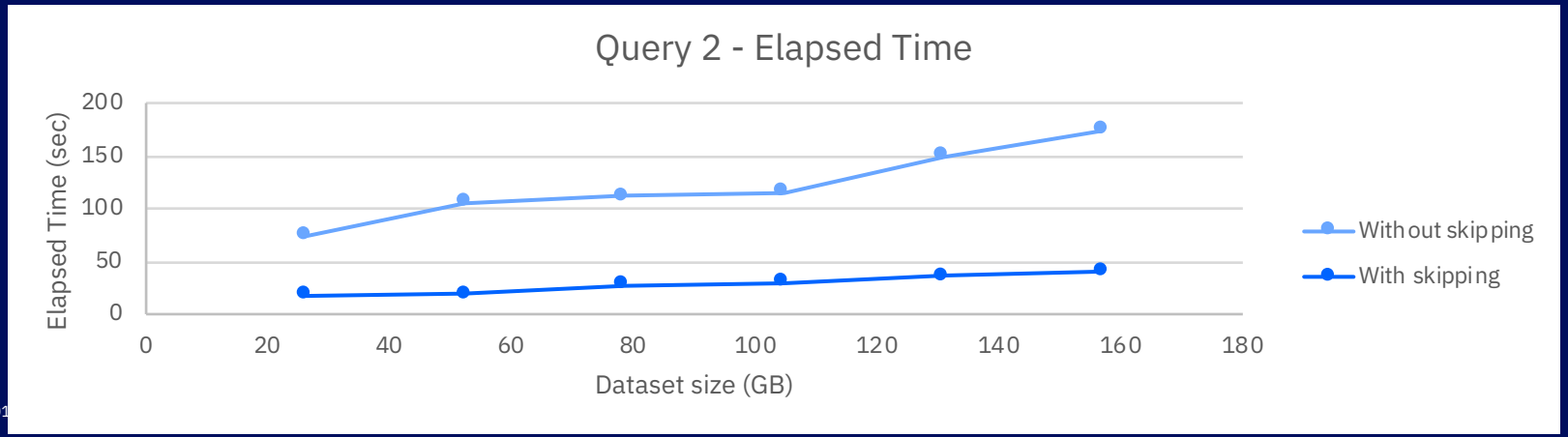
Data Skipping Performance in SQL Query : Query 1



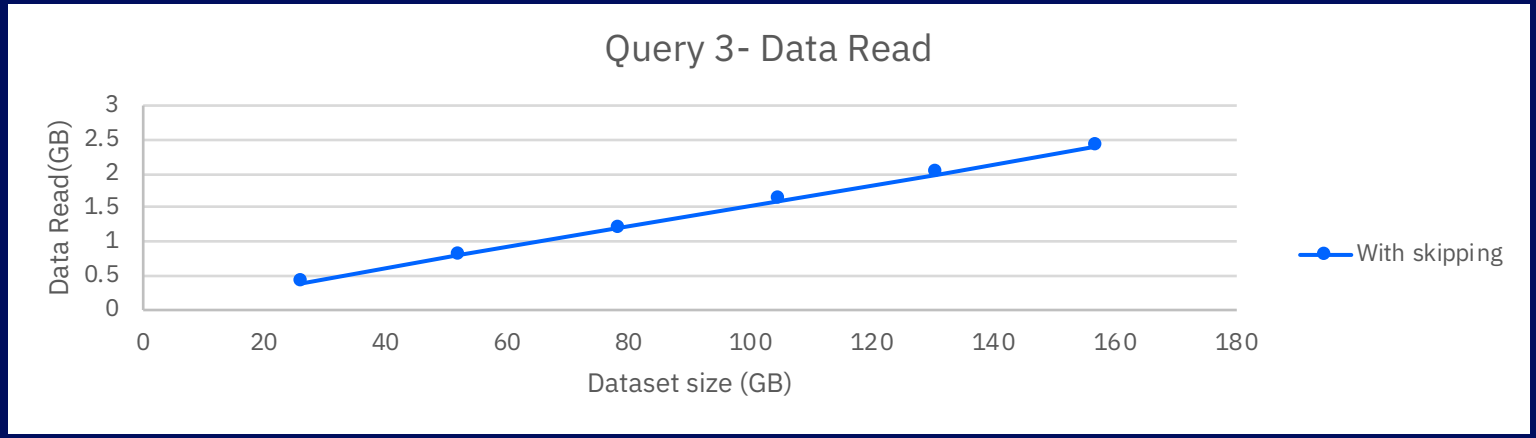


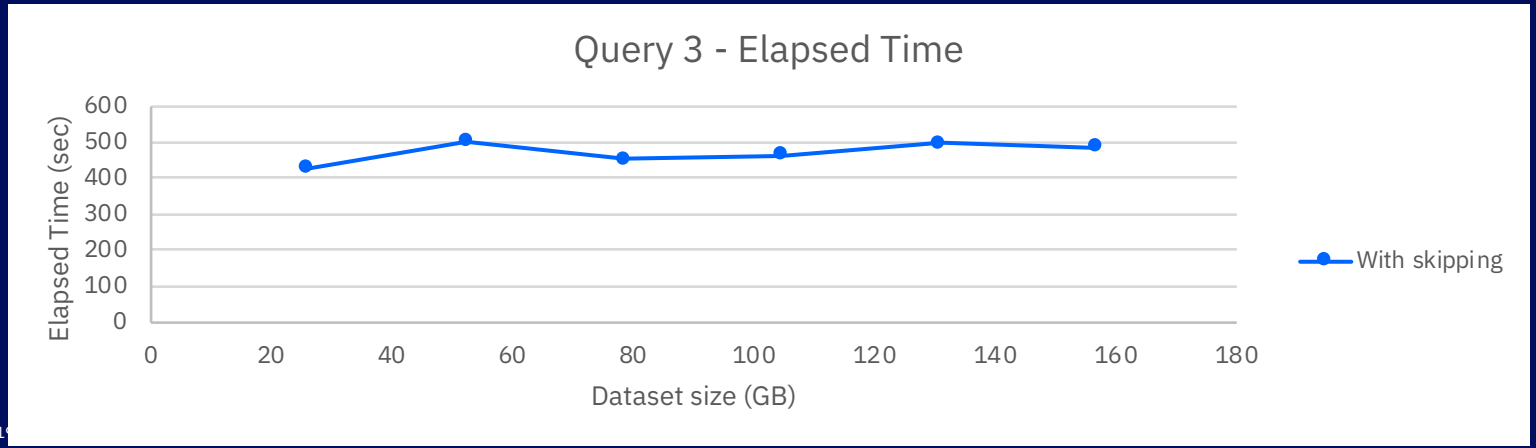
Data Skipping Performance in SQL Query : Query 2





Data Skipping Performance in SQL Query : Query 3





IBM Cloud SQL Query with Data Skipping

- SQL Query is available on IBM Cloud
- Data Skipping is now available for SQL Query users as a closed beta
- Contact Chris Glew cglew@us.ibm.com for beta participation

Data Protection - Research

Fuel consumption is sensitive data for DANAOS

Potential scenario at DANAOS:

DANAOS **fleet manager**

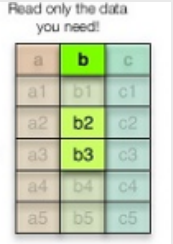
- needs to compare the fuel consumption of the various vessels

DANAOS **crew department operator**

- needs to perform crew scheduling
- should **not** have access to fuel consumption

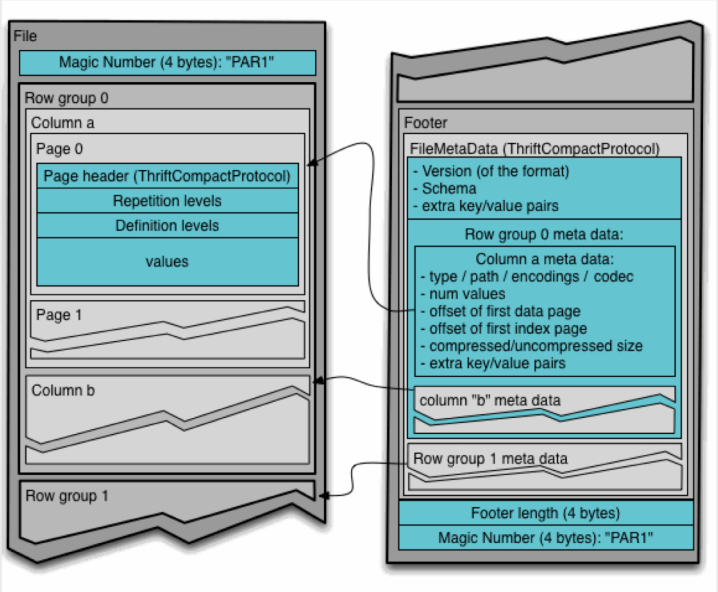
Parquet Encryption: What Problem Are We Solving?

- Protect sensitive data-at-rest
  - data confidentiality: encryption
  - data integrity
  - in any storage - untrusted, cloud or private, file system, object store, archives
- Preserve performance of Spark analytics
  - advanced data filtering (projection, predicate) with encrypted data
- Leverage encryption for fine-grained access control
  - per-column encryption keys
  - key-based access in any storage: private -> cloud -> archive



Parquet Encryption

- **Apache Parquet community work**
- Full encryption: data and metadata
- Enables columnar projection and predicate pushdown
- Storage never sees keys or plain text data
- Works in any storage
- Multiple encryption algorithms
- Data integrity verification
- Column access control
  - encryption with column-specific keys



Policy Management

Fuel consumption is sensitive data for DANAOS

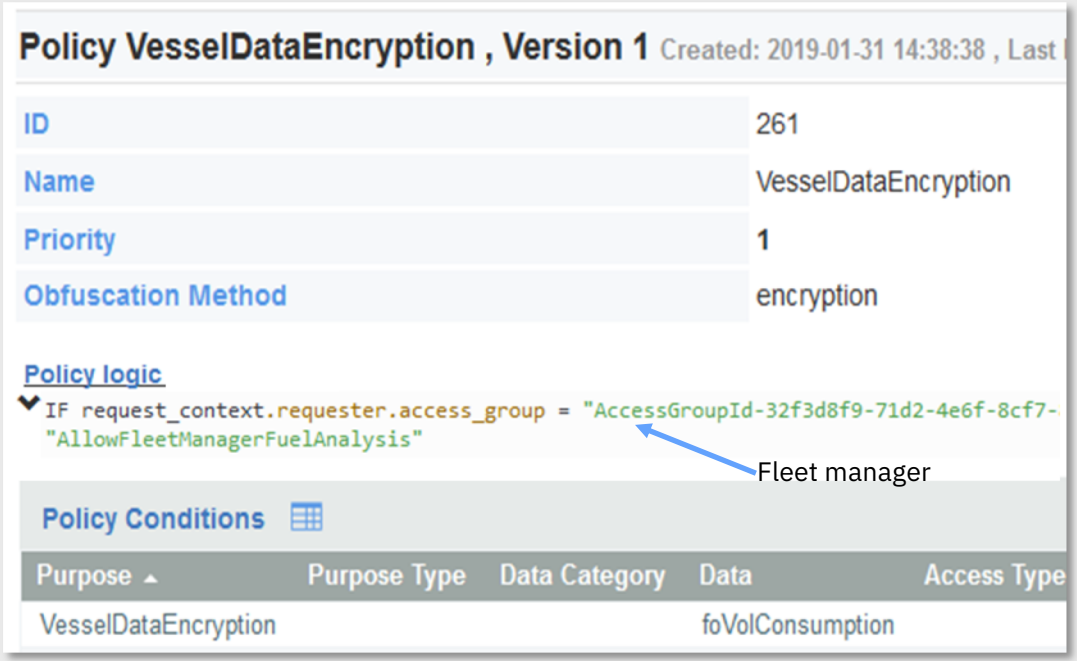
Potential scenario at DANAOS:

DANAOS **fleet manager**

- needs to compare the fuel consumption of the various vessels

DANAOS **crew department operator**

- needs to perform crew scheduling
- should **not** have access to fuel consumption



Spark Integration Prototype

Fleet manager has access to the encrypted fuel consumption column (foVolConsumption)

```
scala> val vessels = spark.read.parquet("vessels.parquet.encrypted")
vessels: org.apache.spark.sql.DataFrame = [vessel_code: string, timestamp: bigint ... 4 more fields]

scala> vessels.createOrReplaceTempView("vessels")

scala> spark.sql("SELECT CASE  WHEN wind_speed < 10 THEN 'Good' WHEN wind_speed < 17 THEN 'Meduim' WHEN wind_speed <
24 THEN 'Bad' ELSE 'Storm' END AS weatherCondition, AVG(foVolConsumption) AS fuelConsumption FROM vessels GROUP BY we
atherCondition ORDER BY fuelConsumption").show()
+----------------+---------------+
|weatherCondition|fuelConsumption|
+----------------+---------------+
|            Good|          314.5|
|          Meduim|          339.5|
|             Bad|          359.5|
|           Storm|          385.0|
+----------------+---------------+
```

Spark Integration Prototype

Crew Department Operator does **not** have access to fuel consumption (foVolConsumption)

The query fails

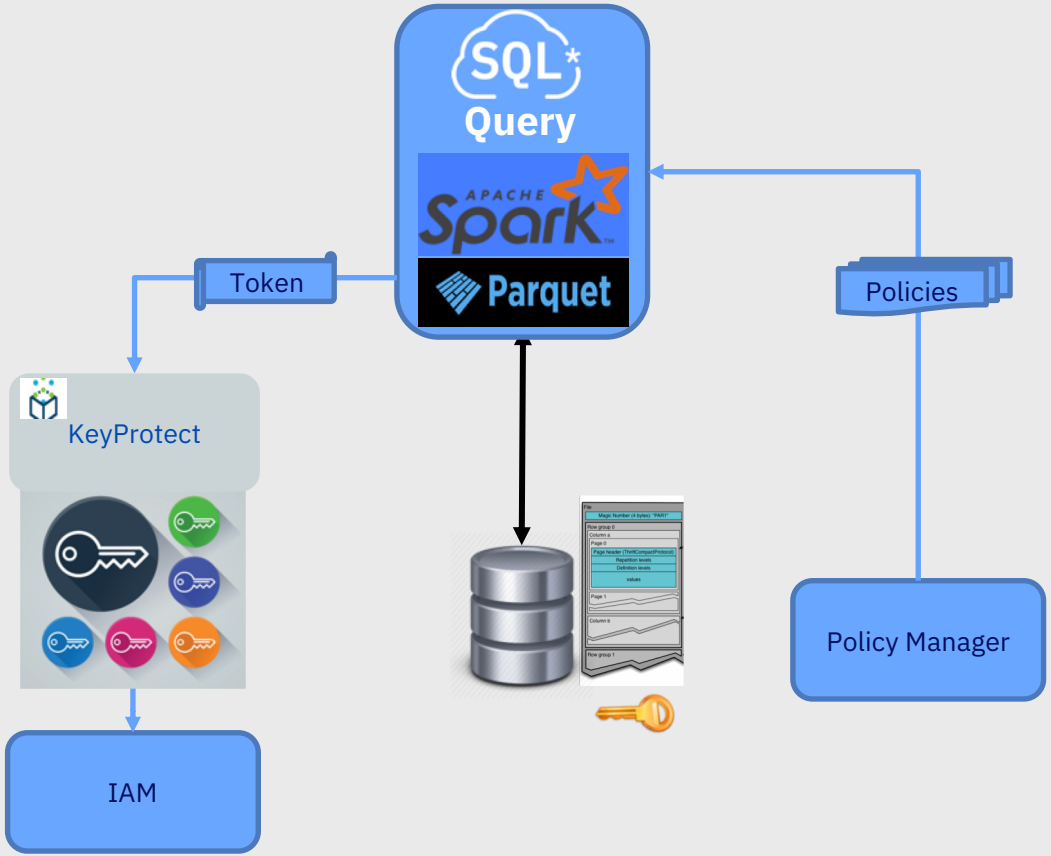
```
scala> val vessels = spark.read.parquet("vessels.parquet.encrypted")
vessels: org.apache.spark.sql.DataFrame = [vessel_code: string, timestamp: bigint ... 4 more fields]

scala> vessels.createOrReplaceTempView("vessels")

scala> spark.sql("SELECT vessel_code, timestamp, wind_speed, foVolConsumption, latitude, longtitude FROM vessels ORDER
 BY foVolConsumption ").show()
[Stage 1:>                                                          (0 + 2) / 2]org.apache.spark.SparkException
borted due to stage failure: Task 1 in stage 1.0 failed 1 times, most recent failure: Lost task 1.0 in stage 1.
2, localhost, executor driver): org.apache.parquet.crypto.HiddenColumnException: [foVolConsumption]
```

Parquet Encryption: Status

- PARQUET-1178
  - format specification approved by Apache Parquet
  - implementation underway
- Community effort, led by IBM
  - many companies taking part
  - even more expressed interest in using in production
- Next: leveraging in Apache Spark
  - strong support in Spark community

A Possible Data Encryption Architecture

- Parquet modular encryption of sensitive columns
- IBM KeyProtect stores Master Keys
- Policy Manager:
  - Data encryption policies – which columns to protect with which Master Key
  - Data access policies
- Data encryption keys:
  - encrypted with Master Keys
  - stored near data
- Identity and Access Management (IAM):
  - Authentication and authorization based on data access policies



Further Resources

Getting started: https://www.ibm.com/cloud/sql-query

SQL Query Intro Video: https://youtu.be/s-FznfHJpoU

SQL Query Starter Notebook in Watson Studio: https://ibm.biz/BdYNrN

SQL Reference: https://ibm.biz/Bd2jF7

SQL Query API doc: https://cloud.ibm.com/apidocs/sql-query

Big Data Layout Best Practices for COS: https://ibm.biz/Bd2jRg

Serverless Data & Analytics: https://ibm.biz/Bd2jF5

SQL Query @ IBM THINK 2019

11-Feb 10 AM:

2263 – The Future of SQL in IBM Cloud (Inner Circle)

12-Feb 9:30 AM:

2238 – What? I Don't Need a Database to Do All That with SQL?

13-Feb 10:30 AM:

2155 – Cloud-Native Clickstream Analysis in IBM Cloud

13-Feb 4:30 PM:

2282 – Enterprise-Scale Analytics Performance with Cloud Object Storage

14-Feb 2:30 PM:

2166 – Self-Service Cloud Data Management with SQL

15-Feb 8:30 AM:

2162 – A Sharing Economy for Analytics: SQL Query in IBM Cloud

Notices and disclaimers

U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. **This document is distributed "as is" without any warranty, either express or implied. In no event, shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.** IBM products and services are warranted per the terms and conditions of the agreements under which they are provided.

IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply."

Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer follows any law.

Notices and disclaimers continued

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products about this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a purpose.**

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

Thank you

Paula Ta-Shma
IBM Research - Haifa
—
paula@il.ibm.com

Gidon Gershinsky
IBM Research - Haifa
—
gidon@il.ibm.com

Stathis Plitsos
DANAOS
—
splitsos@danaos.gr