

Introduction to presto

Zbigniew Baranowski

Vasileios Dimakopoulos

Motivation

- The need for DB like solutions to manage big data
 - ...that RDBMS cannot cope without significant costs
 - ...Open-source with cheap storage
 - Easy to use -> SQL
- HDFS + Spark we have data lakes
 - People can use notebooks to create visualisation of aggregated data (that are stored in certain directories)
 - Spark is great for ETL and batch analytics
 - Supports multiple data sources
 - Scientific use-cases
 - Flexible in defining data processing flows (Scala, Python, SQL)
- ...but, there are some downsides too
 - Difficult to bind with an existing system/application , lack of JDBC natively supported
 - Significant latency for setting up the engine
 - (typically) client has to declare (and know) the resources needed before starting a session
 - A steep learning curve
- There is a room for omni SQL engine
 - Low latency, multi source with JDBC support – complementary to Spark
 - Single end point for many datastores, for convenience and to allow data store technology evolution

We have evaluated what is available on the market

Open Source

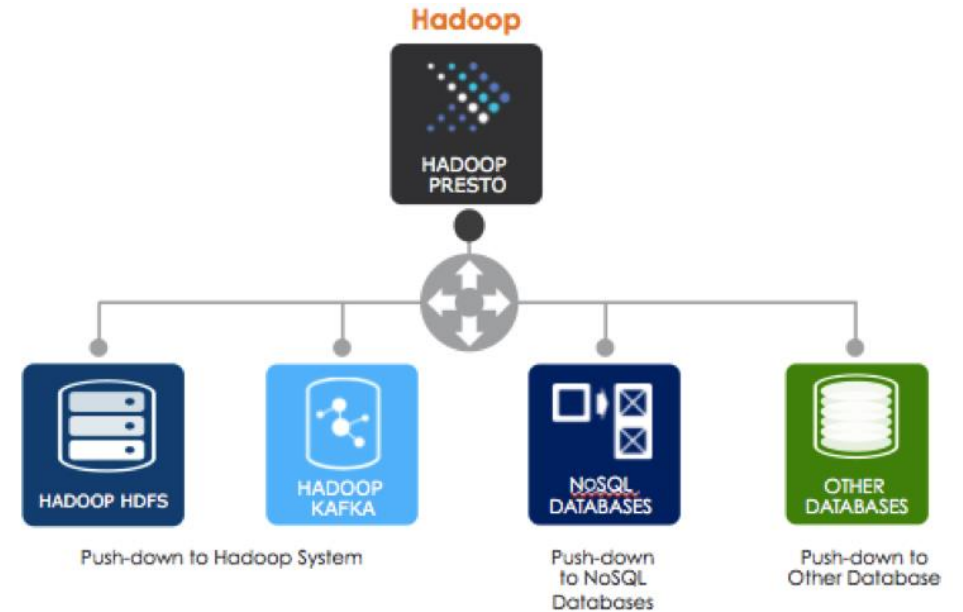


Close Source



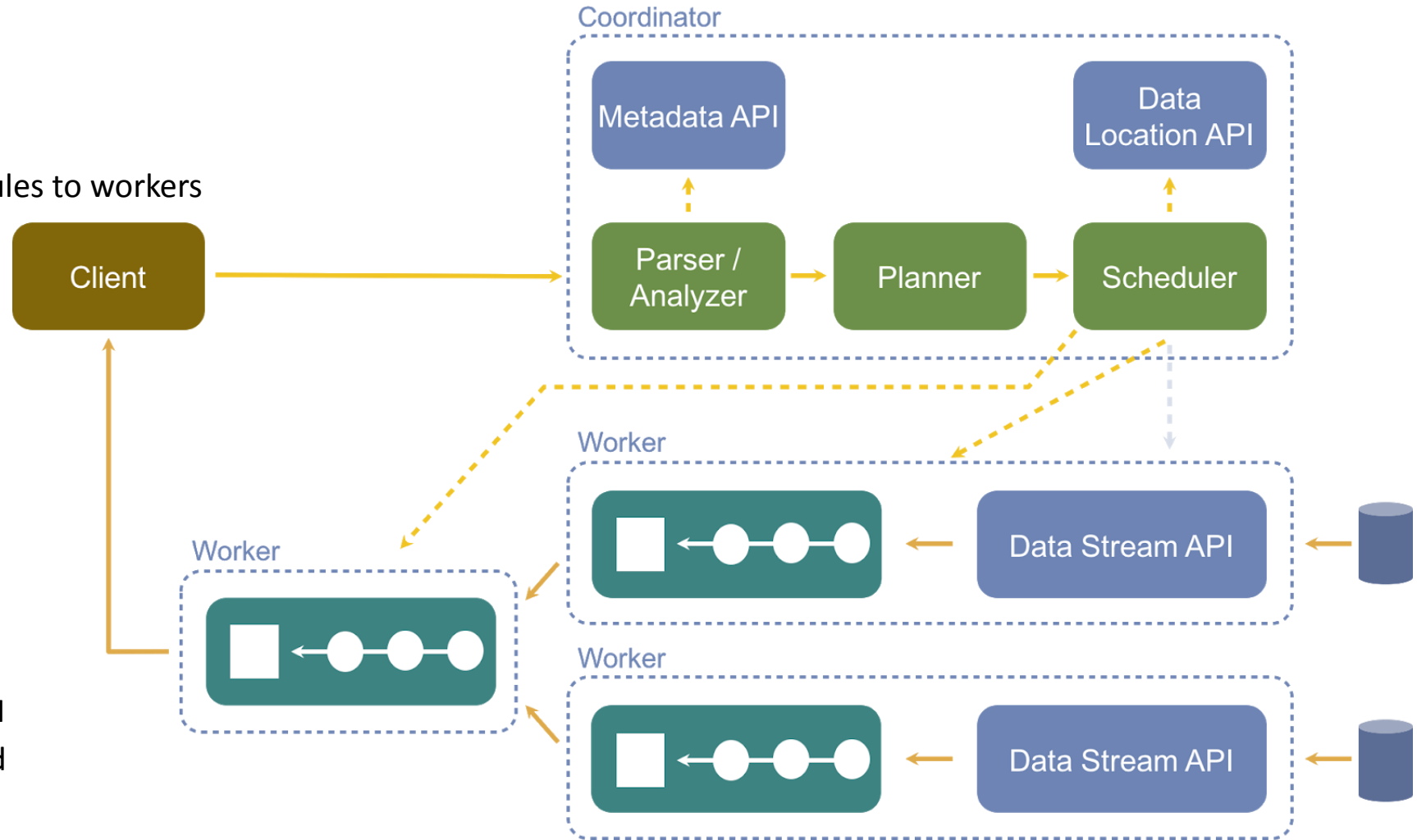
What is Presto?

- A distributed MPP SQL query engine
 - Open-sourced
 - Started by Facebook as a successor of HiveMapReduce
- Typically run on top of a Hadoop cluster
- SQL-on-Anything
 - Hive/HDFS (Parquet, CSV, JSON etc.), S3
 - Relational dbs (MySQL, PostgreS, SQLServer...Oracle*)
 - NoSQL (Cassandra, Redis, Phoenix/HBase)
 - Apache Kafka
 - and more (data sources are pluggable)

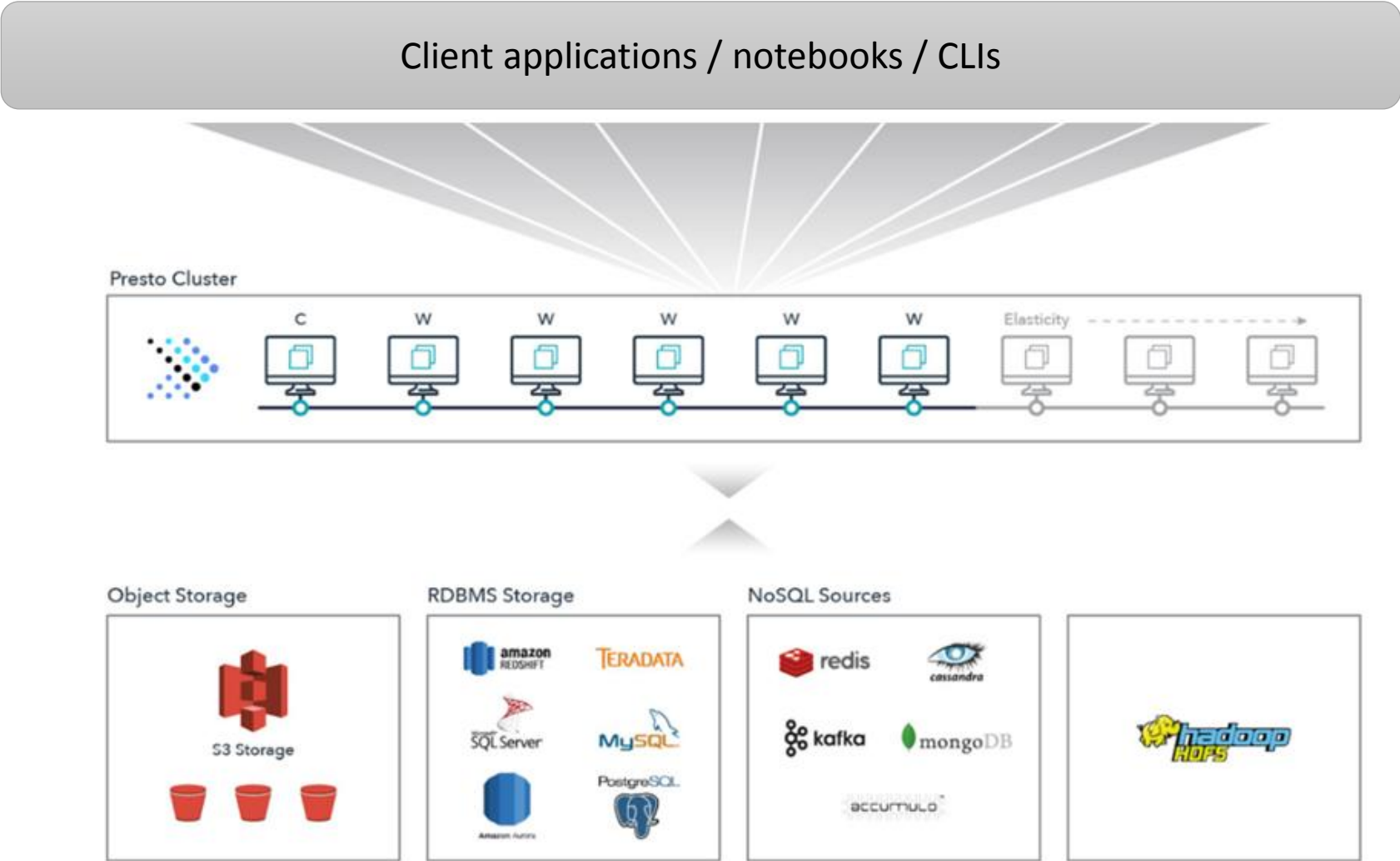


Architecture

- Coordinator
 - Receives a query from the client
 - Analyzes, parses, plans and schedules to workers
- Worker
 - Executes scheduled tasks
 - Reads and writes to/from data sources
 - Shuffles data between other workers
 - Sends the final results to the client
- Client
 - Application
 - Presto console
- Data source plugins
 - Data Location API/ Data Stream API
 - Custom plugins can be easily added



Architecture



What are the differences between Presto and SparkSQL?

- Spark was designed for batch or long running ETL jobs, when Presto targets on interactive queries with low-latency start up time
 - Cluster starts on demand | Presto runs as a shared cluster all the time
 - You have to declare resources needed | Presto gives all that is available at the moment to satisfy the query
- SparkSQL currently JDBC driver available only for Hive tables
- Third party storage plugins in Spark | Centrally managed plugins by the project
 - Eventually will work with a new version | Will work
- In general the purpose of both is different and both plays an important role...

Example - Netflix



ETL

Machine
Learning

Scale



Exploratory

Interactive

Reporting

Audits



Custom Viz

Dashboarding

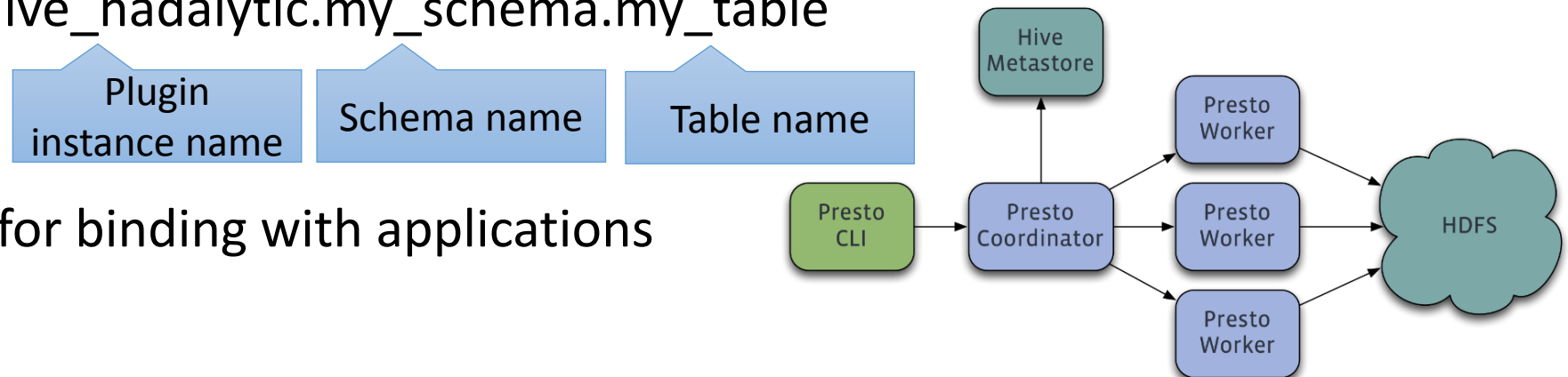
Alerts

Presto for Hadoop in practice

- Has dedicated connector for HDFS via Hive metastore
 - Only data mapped via Hive tables can be accessed
 - Already existing HDFS folders can be easily mapped to Hive (if schema is coherent)
- An alternative connector with Iceberg will be available soon
- Each connector type can have multiple instances (called catalogs)
 - Multiple hives (Hadoop clusters) can be accessed simultaneously
 - `select * from hive_hadalytic.my_schema.my_table`

• Interfaces

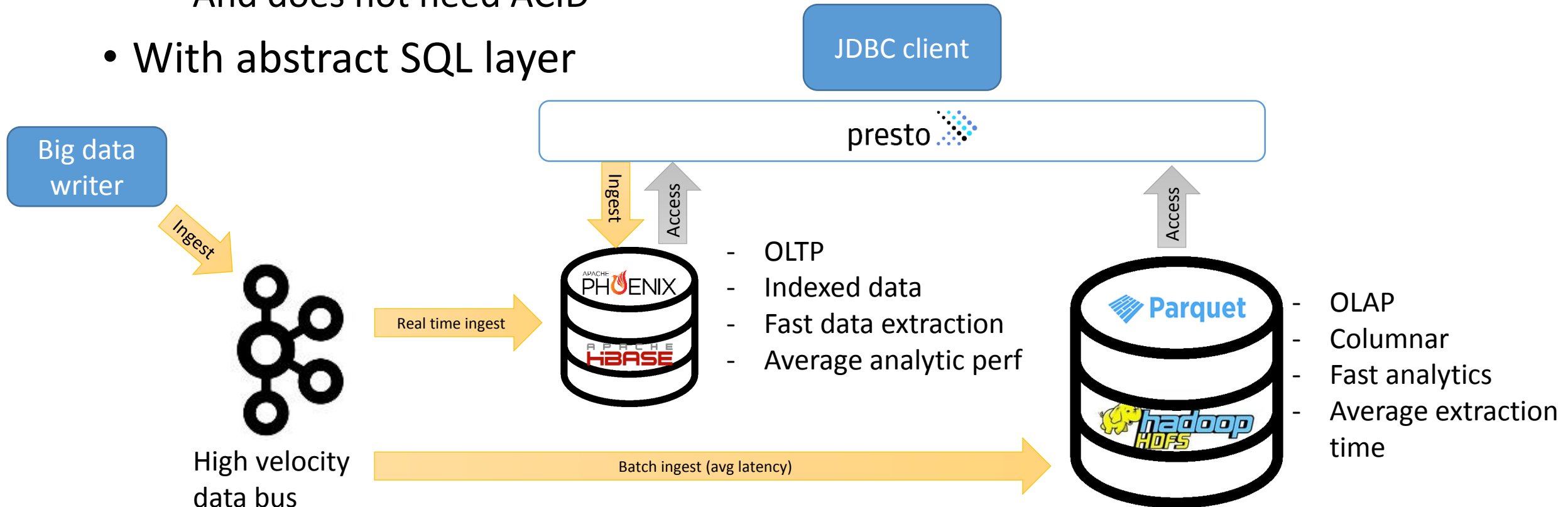
- Presto shell
- JDBC or ODBC for binding with applications



Demo

Towards BigData scale-out database

- SQL scale-out database
 - Alternative RDBMS when data has to be written, stored, accessed at scale
 - And does not need ACID
- With abstract SQL layer



Wrap-up

- Presto – distributed SQL engine on multiple pluggable data stores
- Excels on low latency query executions, not for batch processing
- With single JDBC client you can write and read into big data stores and not only...
- Abstract SQL – not storage specific
 - allows relatively easily replacing underlying data storages
- Can easily glue data stores into a bigger hybrid systems

Presto pilot next year

- Currently we have a test cluster
 - Early testers are welcome to try
- In 2020 we are planning to release a pilot service
 - Fully secured
 - With HA of coordinators
 - Monitoring