Justification

Over the last decade, data has seen an exponential growth, which according to forecasts will continue in the foreseeable future as more devices are connecting to internet, generating events that describe the consumer’s behavior on real-time. This has been possible thanks to the costs drop in storage and CPUs for processing.

Businesses around the world are focusing on leveraging data, as it provides a means to discover potential customers, keep current ones interested and swiftly react to market trends. In consequence, many industries are starting to embrace fact-based data-centric decisions to skyrocket their profit.

Provided that more precise and trustworthy predictions demand more information, it becomes crucial to ingest and analyze more data. However, this entails a whole new problem once the data reaches a volume where it is unmanageable by a single machine, since a new level of abstraction is needed: distributed systems.

The problem with distributed systems is that most current open-source technologies require complex tuning and are difficult to integrate with other technologies, in order to create a complete Extraction Transformation and Load (ETL)\* pipeline. While for large organizations this may be a no-brainer, for small to mid-size ones this becomes extremely difficult to handle, as the expertise required is considerable, experts are sparse, and the implementation cost grows outside the affordable range of the companies, forcing them to utilize inefficient and eventually unmaintainable systems thus limiting their ability to react opportunely.

This work proposes a system that allows users to simply: migrate already-existing data from a local machine to a cloud data warehouse; create customizable big data pipelines, adapted to the data size of the company; provide a way to graphically visualize aggregates of the data, such as plots and charts.

\*Introducir termino ELT

\*Metadata management (Apache Atlas)

\*Paper mas significativo

\*Frases al inicio de cada capitulo

Relevant concepts

First, let’s define what Big Data is about, why it is so important to modern industries and how it differs from traditional data scale mindset. To put it into perspective, it is important to briefly describe the history to understand the major industry shifts that led to this concept.

# History of data before 2000

Prior to 2000, data was exclusively produced and consumed within individual machines. Software installed on PCs read user behavior data to understand what actions to take next or how to improve user experience, although this was very rarely used outside of the software itself.

Between the late 90s and early 2000s, internet became accessible to mid-class users which translated to millions of computers getting access to internet on a regular basis. At the same time, companies started investing on web sites to lure new customers in by providing new ways of acquiring products on internet. One notorious case is Amazon. Amazon started selling books online on the late 90s, but benefited a lot from the internet growth, as many customers started getting used to this new business model.

# New era of connectivity

However, it was not until mid-2000s that connectivity took off with the official presentation of iPhone on 2006. This ushered in a new era: the smartphones era. Smartphones were now available to the big masses and the cellular network was advanced enough to handle several thousands of concurrent connections which contributed to increase the internet traffic and therefore the data production.

# Data Analytics

Companies then realized that this trend could result in a higher potential customer acquisition by leveraging the data footprint users generate on their websites. In addition, social media was born, and it started gaining momentum as more users found it useful. Businesses commenced to interest more and more on the data as a means to target new audiences. However, in order to effectively understand the users, they needed a way to extract value from the data they were able to collect. Enter Data Analytics.

Data Analytics refers to a series of techniques to examine a data set in order to extract value out of it. This translates into predictions that allow organizations to take fact-based decisions and visualize potential risks in the near future, ensuring that resources are correctly allocated and that the company will be able to react in case of a market shift.

# SQL and its limitations

Data Analysts rely on a number of tools to dissect, massage and visualize data; the most prominent ones being SQL and Tableau. SQL stands for Structured Query Language, which has been the de facto domain language to query and aggregate from databases for about 4 decades. This language allows analysts to slice and dice a database to only get the information required for a specific analysis without having to worry about the internals of the database engine. On the other hand, Tableau is a data visualization tool that provides several functions to compare and present data on an appealing form. It is well-known for its smooth integration with many databases and its flexibility.

SQL was originally designed with individual machines in mind, having the underlying data structures optimized for handling concurrency in a multi-threaded environment and being well suited to retrieve data rapidly. As data grew, companies pushed databases to the limit; queries that normally took couple seconds now take several hours.

# Vertical and horizontal scaling

A way to circumvent this issue was to scale up the database, also known as vertical scaling. Vertical scaling implies increasing the computer’s capacity (either the memory size or the storage). This was possible thanks to the evolution of CPUs and the steady transistor size reduction. The trend was coming to an end, though, as researchers found out that higher frequencies on CPU clock entails more heat radiation, and the smaller the transistors, the more fragile they became. On top of that, the price of more sophisticated CPUs and RAM circuitry was too high at the time. Therefore, it became clear that vertical scaling was not feasible for the long term.

In response to this limitation, companies such as Intel and AMD came up with CPUs featuring multiple cores within a single chip, as well as optimized CPU cache. This allowed programs to perform operations in true parallelism and effectively fetch frequently used data.

In spite of these improvements, data was still forcing system administrators to look for new long-term solutions to process data efficiently without having to spend millions on it. This is when distributed systems came into play. Distributed system is a generic term that describes many commodity computers interconnected through a network to perform a specific task, divided into smaller pieces in parallel. So instead of trying to get a computer with the best specs available in the market, people could simply buy a handful of PCs and connect them. This is known as horizontal scaling or scaling out.

# MapReduce

Distributed systems seemed promising, but they added a plethora of new complexities: concurrency, network traffic, data consistency, machine outages, etc. As a result of internal research at Google, a new model emerged to tackle all these nuances: MapReduce. MapReduce is a programming model for processing and generating large data sets in a reliable and scalable manner. Users only have to specify a function (called mapper) to process the key value pair, and then a reducer to collect and aggregate that data. This proved to be very scalable and work very effectively across Google products.

Just around a year later, Google also released the Google File System (GFS) paper which illustrated a fault-tolerant distributed file system for data-intensive applications. MapReduce made extensive use of it to store results of its computations reliably.

# Hadoop

Despite MapReduce was quite innovative by the time, it remained almost unheard for 2 years more, until a Yahoo employee decided to homologate an implementation of both GFS and MapReduce on a single technology. On 2006, Hadoop was born as an attempt to replicate the MapReduce engine and it quickly became relevant for multiple industries. Hadoop additionally provided the Hadoop Distributed File System (HDFS) which is based off GFS.

One crucial factor that contributed to the success and quick adoption of Hadoop was the rich ecosystem around it. Particularly, it offers a wide variety of methods to interact with HDFS programmatically, such as a native Java API as well as a Thrift API available on multiple languages. This flexibility was key to the adoption of the technology as well as its

As more industries were prototyping with Hadoop and the complexities of their jobs increased, engineers found it hard to translate already existing and complicated SQL statements used for their analytics to MapReduce, because of the difference…

# Hive

# Spark

# NoSQL

# Datawarehouse

# Data Governance

# Data wizards: analysts, engineers and scientists

# Cloud Computing

/\* Spare ideas \*/

In 2005, Roger Mougalas from O'Reilly Media coined the term Big Data for the first time. This intended to illustrate humongous amounts of data being accumulated. Little that he knew how important that would become over the next decade.

Big Data, as the name implies, refers to vast amounts of data. Data that often comes in either structured or unstructured formats. Before cloud computing, all data had to be structured before being used. As such many organizations designed systems to retrieve data from a given location, transform it into the desired schema and then use them for the business needs.

State of the Art

MapReduce paper

Big Table paper

Dremel paper

Spark paper