2016

Real-time Visualization On Hadoop using D3

Vinnas Kuttery

11/1/2016

Contents

Objective

Being in Data Industry, We all know the importance of visualization in any data analytics projects. It doesn't matter how complex or efficient analysis you have done until and unless you present and convince your clients with your thoughts . Yes, it's all about how do you prove your idea with others, and I strongly believe that an efficient visualization is the best medium to inject your thoughts into others mind.

Hadoop , a trending term in the area of data science. Apache Hadoop is an open-source software framework for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware. Hadoop is a framework which has wide varieties of components to manage your data. Components like Hive and Hbase are the most used data management (storage/access) components in the current developments.

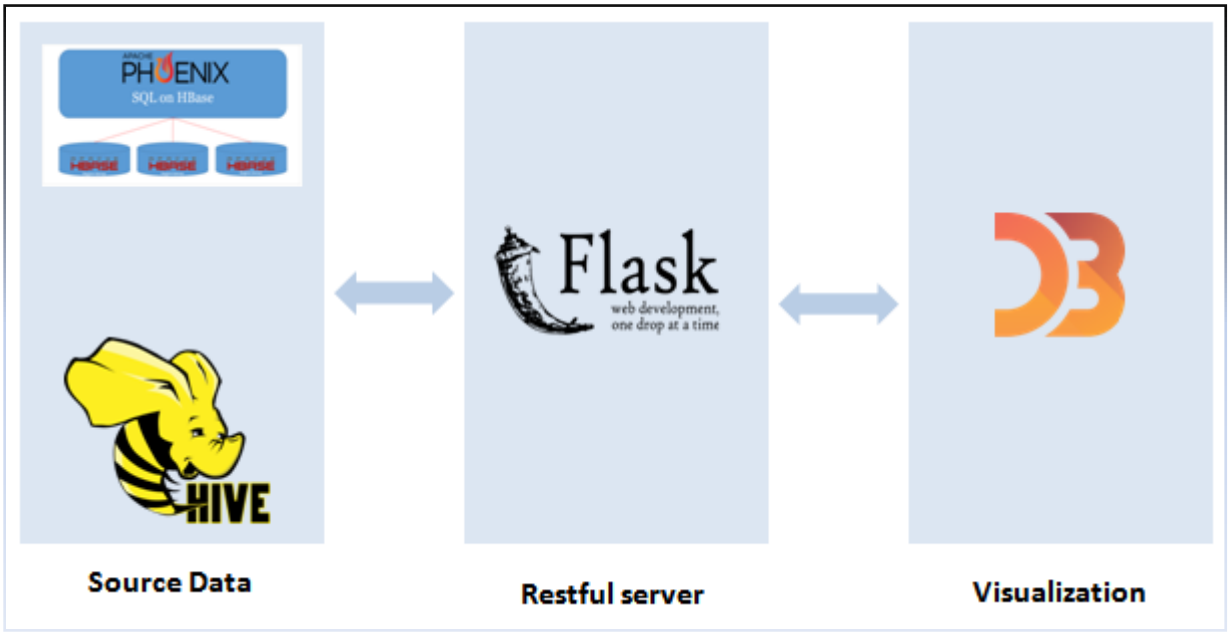
Objective of this project is to develop few real time interactive visualizations on Hadoop (Hive or Hbase) data using an open source platform (Will be using D3 with Flask).

Architecture

When we talk about architecture of any visualization tool, logically it can be divided into three layers as follows,

- **Data Source (Back end) layer :** where the actual data stored (Hive/Hbase)
- **Intermediate Web server layer :** it act as a modem for Visualization Layer (Flask Server) to fetch data from the back end.
- **Visualization Layer :** GUI where end user interact with Visualizations(Java Script and D3).

The below image gives a brief about technologies in each specified levels in this project.



Img1. Overall Architecture

1. Source Data

for the knowledge sharing purpose ,we will be exploring how to manage Hive and Hbase as backend data sources. but, using Hbase though phoenix will provide the far better performance than using hive.

Hbase (Apache Phoenix)

Apache Hbase is an open source NoSQL database that provides real-time read/write access to those large datasets. Hbase is natively integrated with Hadoop and works seamlessly alongside other data access engines through YARN.

In this project, the tweets are saved in Hbase through phoenix.

Apache Phoenix

Apache Phoenix is an open source, massively parallel, relational database engine supporting OLTP for Hadoop using Apache Hbase as its backing store. Phoenix will allow any application to access the Hbase data using SQL queries.

In this project, access to the Hbase table (both read and write) has been implemented with Phoenix.

Hive

Apache Hive is a data warehouse infrastructure built on top of Hadoop for providing data summarization, query, and analysis.[2]Hive gives an SQL-like interface to query data stored in various databases and file systems that integrate with Hadoop. The traditional SQL queries must be implemented in the MapReduce Java API to execute SQL applications and queries over a distributed data

2. Web Server

A web server is a computer system that processes requests via HTTP, the basic network protocol used to distribute information on the World Wide Web. The term can refer to the entire system, or specifically to the software that accepts and supervises the HTTP requests.

Flask

Flask is a micro web framework written in Python and based on the Werkzeug toolkit and Jinja2 template engine.

In this project, the web server is developed using flask.

Gunicorn

Gunicorn 'Green Unicorn' is a Python WSGI HTTP Server for UNIX. It's a pre-fork worker model ported from Ruby's Unicorn project. The Gunicorn server is broadly compatible with various web frameworks, simply implemented, light on server resources, and fairly speedy.

Here, Gunicorn is used to enable the user concurrency in flask application.

3. Front End

The visualization side is implemented using JavaScript along with its various important components like D3 and Ajax.

JavaScript

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

In the project, most of the visualization objects are dynamic. So, all visual objects are created using java script.

D3

D3.js is a JavaScript library for manipulating documents based on data. **D3** helps you bring data to life using HTML, SVG, and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

Objects like pie chart, bar chart and word cloud in this project are created in JavaScript by invoking libraries from D3.

Ajax

Ajax is a set of web development techniques using many web technologies on the client-side to create asynchronous Web applications.

JavaScript request/response to flask for getting data is implemented using Ajax calls.

Platform Requirement

Operating System	Linux CentOS 6
Hadoop Version	HDP 2
Hadoop Components	Hive, Phoenix, Hbase
Software/Languages	Python

Download Working Project

you may download or clone the entire working project from the below github repository,

https://github.com/Vinnas-Kuttery/Real-time-Visualization-On-Hadoop-using-D3

Demo

About Data

Here, we are using the famous IMDB data for the demo purpose.

Table Schema

following are the table schema for the same.

Column Name	Data Type
movie_title	text
color	varchar(10)
director_name	varchar(100)
actor_2_name	varchar(100)
gross	Double
genres	text
actor_1_name	varchar(100)
movie_imdb_link	text
language	varchar(100)
country	varchar(100)
budget	Double
title_year	int
movie_facebook_likes	Bigint

Sample Data

Make sure we upload the dataset in both Hive and Hbase with a table name ' imdb_data '

Visualization

There a two simple and interactive word cloud visualization which uses Hbase and Hive as their data source. Performance for both the views are completely depended on their own process engines.

Hbase Visualization

This visualization uses Phoenix as the processing engine for Hbase Data. This will give you the best performance than using Hive.

Hive Visualization

This uses hive as processing engine. in this particular use case ,it is comparatively slower than Phoenix(Hbase) processing.

Use Cases

This application can be used in any Hadoop based real time analysis applications. few of the sample use case are below,

- Social Media Analytics
- Log Analytics
- IOT

Limitations

Since we are using flask based web Framework, even if we use Gunicorn , it is not recommended for improving user concurrency with large scale application.

We can upgrade the flask application to Django or Pyramid Framework to improve the user concurrency.

Conclusion

I'm concluding by saying this is completely a free solution which can be implemented with Any Hadoop based visualization projects. as I mentioned in earlier section, the limitation of user concurrency can be resolved by replacing Flask with Django or Pyramid, rest of the components are going to be same in the architecture.

Note: from data source perspective , using Hbase through phoenix as data source will provide the best performance than using Hive.