# Data Warehouses

Google BigQuery
AWS Redshift
Azure Data Lake
Apache Hive

Portland State
Computer Science

# Motivation

- What if you want unlimited capacity while supporting fast querying (i.e. like Google Search)?
  - Small transactional in-memory databases support fast queries, but do not scale (SQL, MySQL etc.)
  - Large file systems support scale, but can not (natively) support querying (GCS, S3)
  - NoSQL databases store massive datasets via distributed hash-table, but are difficult to query efficiently (i.e. scan and query, not SQL)

# Data warehouses

- Typically used for On-line Analytical Processing (OLAP) apps
  - e.g. Log processing for site/app analytics
- Store large datasets organized for write once, read/query many access (WORM)
  - Optimized for large reads and writes
- Does not require transactional properties of On-line Transaction Processing (OLTP)
  - e.g. No need for ACID (i.e SQL/Spanner semantics)
- Implemented via cheap disks and slower CPUs

# BigQuery

# BigQuery

- Gartner's Magic Quadrant report on public cloud services

"Google's differentiation factor lies in its deep investments in analytics and ML. Many customers who choose Google for strategic adoption have applications that are anchored by BigQuery."

# BigQuery

- Fully managed, no-ops data warehouse
  - Developed by Google when MapReduce on 24 hours of logs took 24 hours to execute
  - Supports fast, streaming data storage
    - 100k rows and hundreds of TB per second
  - High-performance querying via SQL-like interface
    - Near real-time analysis of massive datasets via replication and parallelism
  - Allows one to bring code to where data is (in the cloud)
- How?

# Initially: Flat-file (row-based) storage for logs

- Log data typically stored in a flat file in file system
- Example: packet traces

```
09:59:27.329166 IP 10.218.128.13.62562 > mashimaro.cs.pdx.edu.http:
Flags [.], ack 419, win 258, length 0
09:59:33.011286 IP 10.218.128.13.62562 > mashimaro.cs.pdx.edu.http:
Flags [F.], seq 373, ack 419, win 258, length 0
09:59:33.011297 IP mashimaro.cs.pdx.edu.http > 10.218.128.13.62562:
Flags [F.], seq 419, ack 374, win 237, length 0
09:59:33.017169 IP 10.218.128.13.62562 > mashimaro.cs.pdx.edu.http:
Flags [.], ack 420, win 258, length 0
```

  - Consider calculating inter-arrival times of packets
    - Stored as a flat file? Reads entire trace
  - Common access pattern for data analytics is to access only one piece of data (i.e. slice or column) from each entry (row)

# Idea #1: Column-oriented storage

- Split columns into separate contiguously stored files for performance
  - Reduces data accesses for column-oriented queries
  - Achieves better compression
    - Grouping of similar data types in columns
- Replicate to support parallel querying
  - Only common columns needed in queries replicated

# Idea #2: Employ serverless querying

- Queries spawn off computing and storage resources to execute
  - Up to 2,000 nodes/shards if available
  - Done over a petabit network in backend data center
- Pay per query with minimal cost to store data

# BigQuery demo

- Copy and paste the query into editor
  - Use settings to specify "Legacy SQL"
  - Note how much data will be accessed

```
SELECT name, sum(number) as name_count
  FROM [bigquery-public-data:usa_names.usa_1910_2013]
  WHERE gender='F'
  GROUP BY name
  ORDER BY name_count DESC
  LIMIT 10

SELECT language, SUM(views) as views
  FROM [bigquery-samples:wikipedia_benchmark.Wiki1B] // 1 b rows
  WHERE regexp_match(title,"Goog.*")
  GROUP BY language
  ORDER BY views DESC
```

- Cached results are free
  - Check timing

# BigQuery demo

- Larger query (Preview only. DO NOT RUN)

```
SELECT language, SUM(views) as views
    FROM [bigquery-samples:wikipedia_benchmark.Wiki100B] // 100 b rows
    WHERE regexp_match(title,"G.*o.*o.*g")
    GROUP BY language
    ORDER BY views DESC
```

- Pricing
  - < $0.02 per GB stored per month (first TB free)
  - But, $5 per TB processed
    - Do NOT do a "SELECT *"
    - Pay attention to dry run cost estimate before execution!

# Public datasets on BigQuery

- QuickDraw with Google
  - 50 million drawings
  - https://quickdraw.withgoogle.com/data
- Github
  - Find out whether programmers prefer tabs or spaces
- NYC public data
  - Find out which neighborhoods have the most car thefts
  - Find out which neighborhoods have issues with rat infestation (311 calls on rats)
- NOAA ICODE ship data from 1662
  - Find ships nearby when Titanic sank
- Bitcoin and Ethereum block-chains

# Data Notebooks

iPython, Jupyter

Google Cloud Datalab

Azure Notebooks

Portland State
Computer Science

# Data notebooks

- Interactive authoring tool
  - Combine program code (Python) with rich document elements (text, figures, equations, links)
    - e.g. Like a Google Doc that can execute code
  - Used to document data exploration, transformation, analysis, and visualization
  - Notebook includes the data products and artifacts along with code that generated them
  - Allows one to disseminate results in a reproducible manner!

# Data notebooks

- Initially iPython (interactive Python) run locally
- Now Jupyter
  - Server-based notebooks
  - Interpreter runs on server, wrapped in HTML and served via web
  - Server installed with all packages and data for producing artifacts within code

# Installing Jupyter locally

```
virtualenv -p python3 env
source env/bin/activate
pip install jupyter
jupyter-notebook
```

- Launches a web server that hosts the interactive notebook as a web app
- Visit URL in browser

# Binder

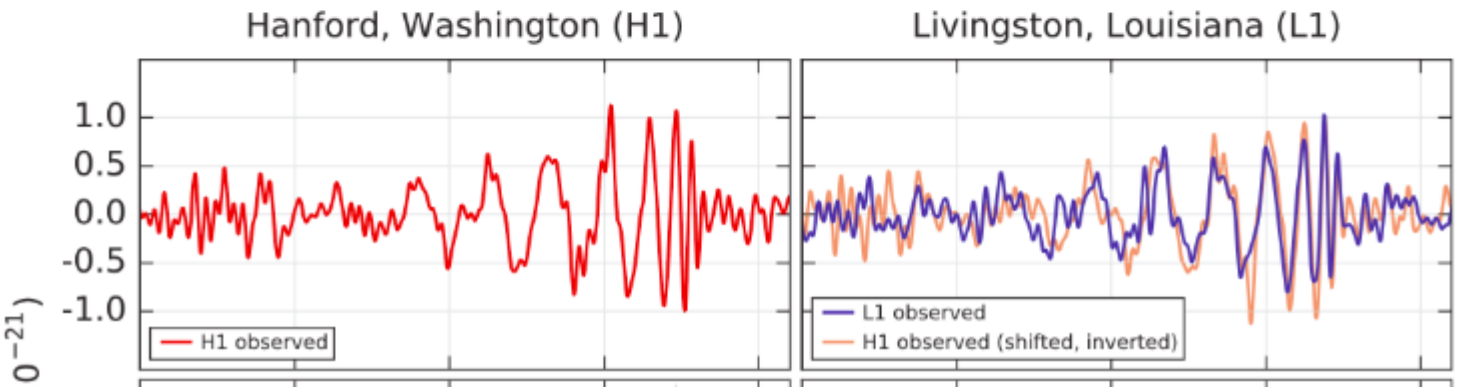
- Combine Github and Docker with Jupyter
- Github repo contains notebook and dependencies for launching it (via `requirements.txt` or `app.yaml`)
- Binder parses dependencies and builds Docker image to run notebook
- Allows you to replicate notebook environment of others

# Used to disseminate experiments

## LIGO, the 2017 Nobel prize in physics, and wrapping up Makefiles

The 2017 Nobel Prize in physics was awarded this week to three leads of the LIGO collaboration for the discovery of gravitational waves.

This is the key figure from the original paper about event GW150914, Observation of Gravitational Waves from a Binary Black Hole Merger:




At the LIGO Open Science Center, the collaboration publishes the actual Jupyter Notebooks necessary to replicate the final steps of the analysis.

# Google Cloud Datalab

- Hosted Juypter instance
  - For analyzing data in the cloud
  - Avoid downloading data
  - Avoid installing all of the GCP libraries
- Service automatically spins up a Jupyter instance on a Compute Engine VM
  - Access to BigQuery and Cloud Storage
  - Access to services such as Machine Learning Engine

# BigQuery, DataNotebooks Labs