CIS 9440 - Data Warehousing and Analytics

Class #12

Class note: Multifact Star Models in Tableau

- link

Class note: Milestone #3

- Impressive work! Saw many creative solutions
- How to include in your Final Project presentation?

Class note: Final Project Presentations (Milestone #5, due 12/6/22)

- Milestone #5 due on Blackboard on 12/6/22
- All Final Projects will be presented on 12/7/22
  - Sign up link here: link
  - You may present in-person or via Zoom
  - Maximum length of presentation is **9** minutes
  - Each presentation will have 2 minutes for questions
  - Your presentation may start earlier than the sign up time
  - **Attendance required** for entire class

Class note: Final Exam

The Final Exam will be hosted on 12/21/22 via Blackboard at 6:00pm - 8:00pm.

Like the Midterm Exam, I will leave a Zoom chat open for questions.

Week 12 Class Overview:

1. BI/Analytics Workshop #3 - yfinance
    a. Descriptive vs Prescriptive Analytics
2. Data Warehouse Technical Architecture
3. Distributed Data Processing Architectures

Week 12 Class Overview:

1. **BI/Analytics Workshop #3 - yfinance**
    a. **Descriptive vs Prescriptive Analytics**
2. Data Warehouse Technical Architecture
3. Distributed Data Processing Architectures

Acquiring Yahoo Finance data

- Oftentimes, Data Warehouses contain stock price information
- For our purposes, one easy way to extract this information is through the yfinance python package
- We will acquire ~3 years of company and stock information for companies in the Dow Jones Industrial Average (DJIA)

Our Goal

1. Scrape Dow Jones data with python
2. (download dimensionally modelled Dow Jones company data)
3. (download dimensionally modelled Dow Jones price data)
4. Create descriptive analytics dashboard in Tableau
5. Brainstorm steps to get to prescriptive analytics

KPI’s to create in Tableau

DJIA Company data

- Average Full Time Employees by Sector
- Total Revenue by State (contribution analysis)
- Average 52 Week Change by Industry

DJIA Price data

- Close Price by Month (by Sector)
- Close Price by Month (by Company)
- Average High minus Low Price by Day of Week

Examples for further analytics

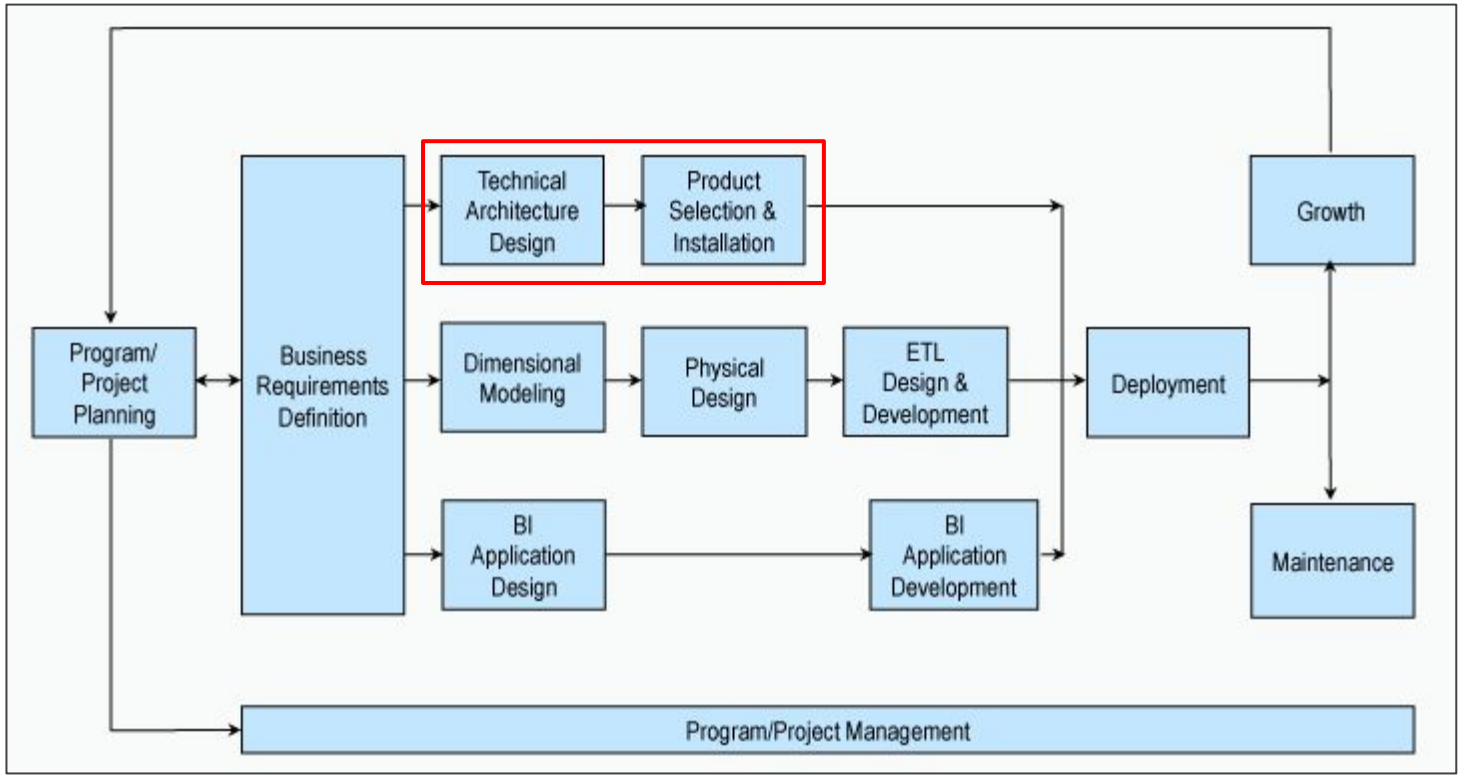
Find an example for each of the following about the DJIA companies and stock prices:

- Diagnostic Analytics
- Predictive Analytics
- Prescriptive Analytics

Week 12 Class Overview:

1. BI/Analytics Workshop #3 - yfinance
    a. Descriptive vs Prescriptive Analytics
2. **Data Warehouse Technical Architecture**
3. Distributed Data Processing Architectures

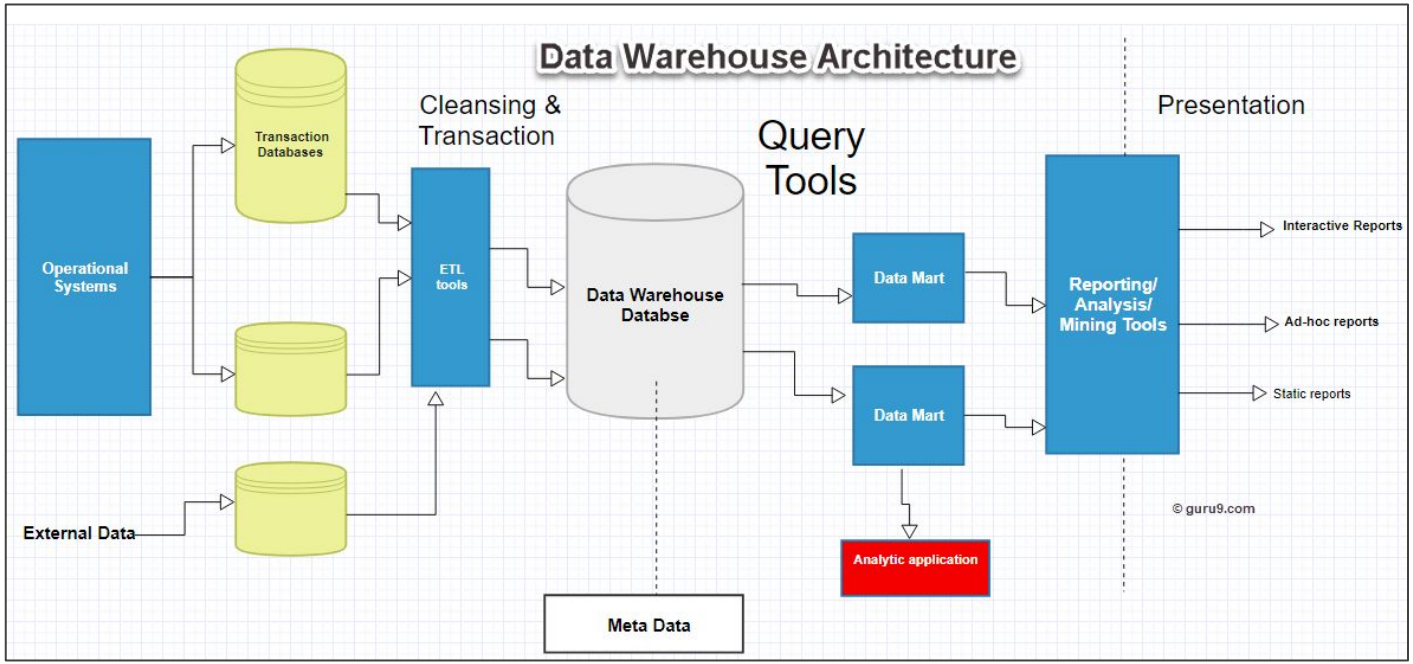
Kimball Lifecycle - Technical Architecture



Why can this be done in parallel to the other two tracks?

What is DW Technical Architecture?

"The DW Technical Architecture is the blueprint for the DW/BI environment's **technical services** and **infrastructure**" - *DWTK Ch 17*



What is DW Technical Architecture?

For example, think about building a house in terms of the Kimball Lifecycle:

Project Planning: You start with the purpose of building the house

KPI's: You define KPI's that will measure the success of the house

Dimensional Modeling: you map out how the rooms will connect

BI Design and Development: you build a system to view the KPI's

Technical Architecture: Who will build the house? What materials will you use to build it/which make sense? Do these materials work together? Are these materials best suited for the climate of the location? Are these materials going to make the house too expensive?

What is DW Technical Architecture blueprint components?

What does the blueprint consist of?

All technologies and infrastructure used in the Data Warehouse:

1. Data Sources
2. Data Integration tools
3. Data Warehouse tools/dbms
4. BI platform(s)

What are DW Technical Architecture blueprint components of your Final Project? (First, without vendor names)

All technologies and infrastructure used in the Data Warehouse:

1. Data Sources?
2. Data Integration tools?
3. Data Warehouse tools/dbms?
4. BI platform(s)?

What are DW Technical Architecture blueprint components of your Final Project? (Now, with vendor names)

All technologies and infrastructure used in the Data Warehouse:

1. Data Sources?
2. Data Integration tools?
3. Data Warehouse tools/dbms?
4. BI platform(s)?

What’s the process for Technical Architecture?

According to the Kimball Lifecycle best practices, the design process is robust (since this is for both small and large companies). The **two main components** are:

- Technical Architecture Design - (*Technical Requirements*)
- Product Selection and Installation - (*Specific Products and Details*)
    - (Similar to the thought exercise we just did with your Final Project)

What's the process for Technical Architecture Design?

1. **Technical Architecture Design** (pg 148)
    a. Collect and document Architecture-related requirements
        i. What are must-have capabilities?
        ii. What are nice-to-have additions?
    b. Create the architecture model - as specific as possible
    c. Determine Architecture implementation phases
        i. What will be the order of events/installations/connections?

Technical Architecture Design example

Technical Architecture Design (pg 148)

1. Collect and document Architecture-related requirements
    a. What are must-have capabilities?
        i. Data will be coming from 3 sources, 2 operational systems and 1 external social media feed.
            1. Will need ETL process to extract and transform data from all 3 sources on an hourly basis
        ii. We must have all data from 2010-present, which will be >20 TBs

Technical Architecture Design example

Technical Architecture Design (pg 148)

1. Collect and document Architecture-related requirements
    a. What are must-have capabilities?
        i. Analysts must be able to view Dashboard style reports from their mobile devices
        ii. All employees must see daily updated Scorecard reports
        iii. Store employees must be able to scan barcodes to get weekly product selling reports

Technical Architecture Design example

Technical Architecture Design (pg 148)

1. done.
2. Create the architecture model - as specific as possible

Technical Architecture Design example

Technical Architecture Design (pg 148)

1. done.
2. done.
3. Determine Architecture implementation phases

What's the process for Product Selection and Installation?

Now, you already have your technical needs from the Technical Architecture Design - fulfill those needs with actual products

2. Product Selection and Installation
    a. Market research and Develop a product evaluation matrix
        i. https://library.educause.edu/-/media/files/library/2013/1/acti1302-xls.xls
    b. Evaluation a short list of options
    c. Select product, install on trial, negotiate
        i. *no price, contact sales (https://www.getdbt.com/pricing/)

DW Technical Architecture - Example

Imagine we are buildpular library books in the Library system and to identifying a Data Warehouse for the NYC Public Library System. The goal of this Data Warehouse is to identify popular books the Library system does not carry. How would we build the DW Technical Architecture?

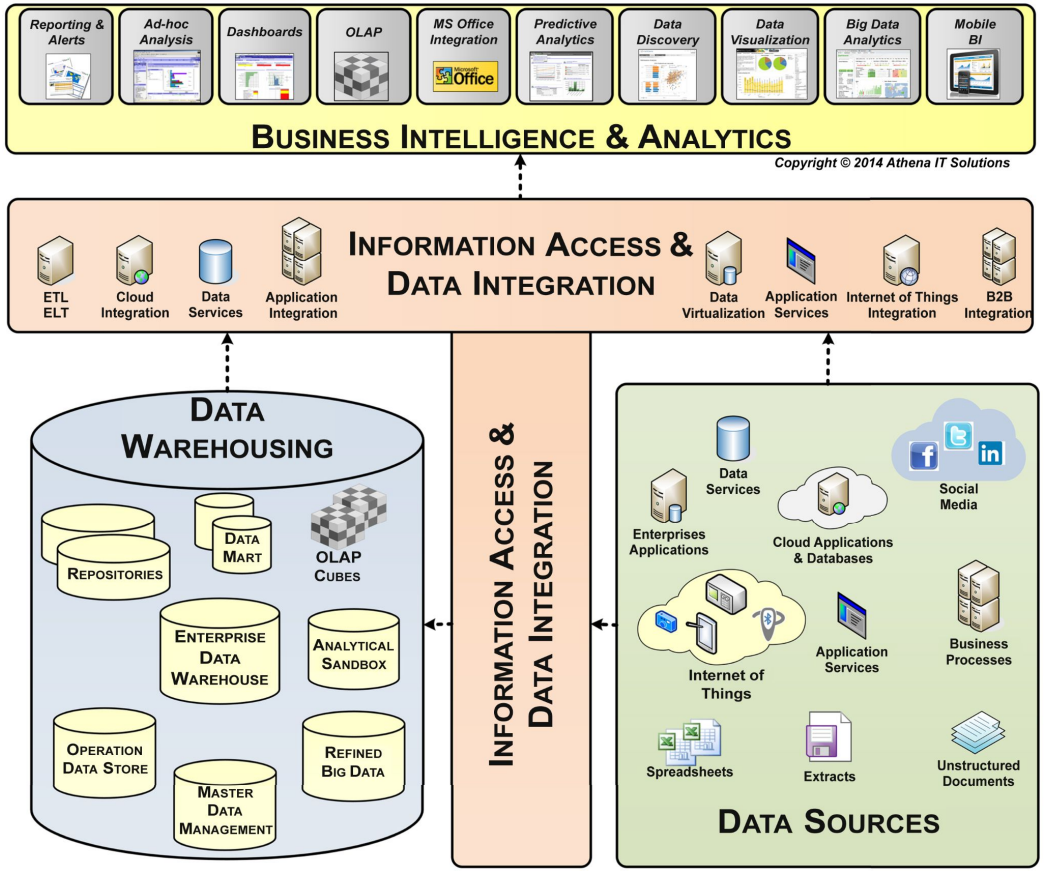
DW Technical Architecture - Example

Step 1 - Collect and document Architecture-related requirements

- Must handle data from in-house OLTP source systems
- Must be able to aggregate data from online customer reviews
- Must be able to connect to Social data APIs
- Must store over 500 terabytes of data about inventory, check-outs, and library members
- Must be able to send library staff weekly reports about new trending books
- Must maintain a near real-time library dashboard for member activity that can drill-in on key metrics

DW Technical Architecture - Example

Step 2 - Create Technical Architecture Model



DW Technical Architecture - Example

Step 3 - Determine Architecture implementation phases

Phase 1

1. Stand up Data Integration tools to connect to 2 in-house OLTP systems
   a. Validate output
2. Set up Data Integration from Social data

Phase 2

3. Stand up Enterprise Data Warehouse and Analytical Sandbox
   a. Connect Data Integration tools from steps (1) and (2) to Enterprise Data Warehouse
   b. Setup ETL process to collect EDW metadata

Phase 3

4. Stand up BI Application
   a. Connect Enterprise Data Warehouse to BI Application
5. Test reporting and dashboarding functionality from BI Application

DW Technical Architecture - Example

Step 4 - Market research and Develop a product evaluation matrix (BI Application Example)

- Fill out this Matrix: https://docs.google.com/spreadsheets/d/1p7xIDlFPJ7rad6_wB02DDhdvmXXGlhfFqdc1HwRBSTc/edit?usp=sharing

DW Technical Architecture - Example

Step 5 - Select product, install on trial, negotiate

- Install a trial version of BI Application
- Connect BI Application to Data Warehouse
- Test functionality
- Negotiate price per user and length of contract

DW Technical Architecture End Results

At the end of the Data Warehousing Technical Architecture track in the Kimball Lifecycle are **two completed tasks**:

1. Detailed Technical Architecture diagram with specific products shown for each component of Data Warehousing
2. Each application from the Architecture diagram installed and ready for use

10 Minute Break

Week 12 Class Overview:

1. BI/Analytics Workshop #3 - yfinance
    a. Descriptive vs Prescriptive Analytics
2. Data Warehouse Technical Architecture
3. **Distributed Data Processing Architectures**

What is Data Processing?

- **Data Processing** is simply the computer operations performed on data to organize, transform, or retrieve data.
  - Examples:
    - Querying a large dataset to retrieve summary statistics
    - Moving data from one table to another
    - Converting many data points into a scatter plot visualization
    - Running data through a machine learning algorithm

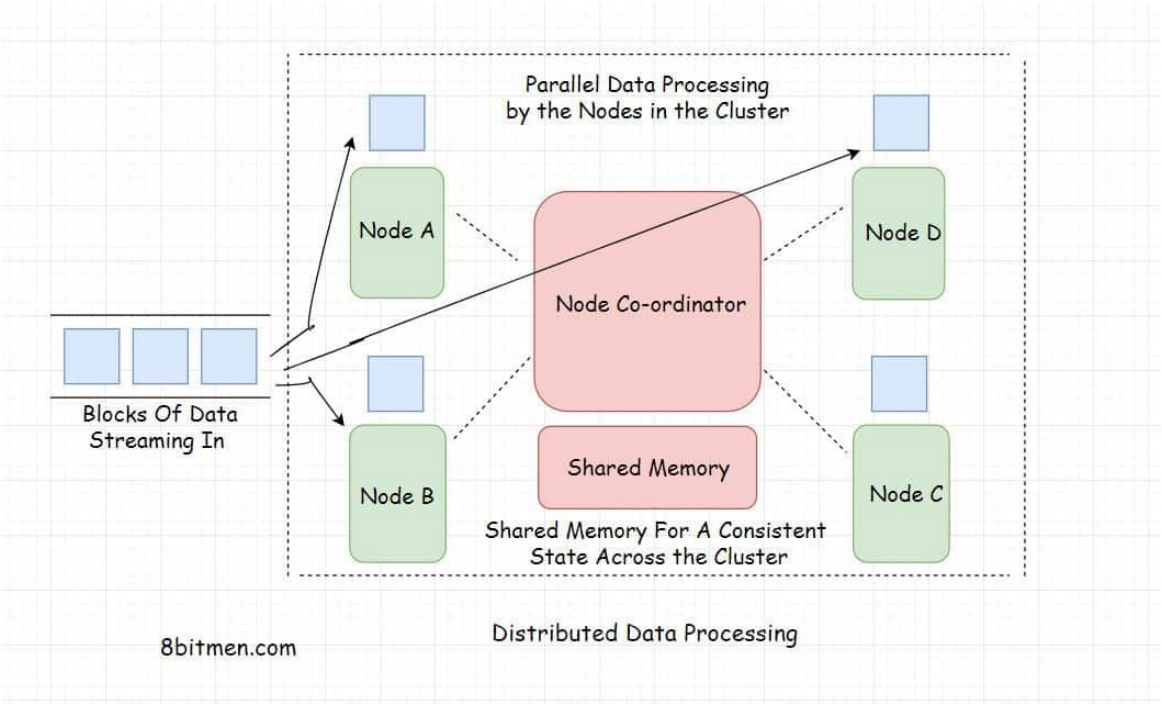
What is Centralized Data Processing?

- **Centralized Data Processing** is performing data processing operations on a single computer, or cluster of computers in a single location.
    - Example: right now, if you wanted to run more applications locally (more data processing) you could buy a computer with more processing power and RAM. But, there is a limit to how powerful a computer you could possible buy.
        - What if you could tie the power of multiple computers together?!

What is Distributed Data Processing?

Distributed Data Processing is performing data processing operations across several nodes running in a cluster at one time. This allows each node to execute tasks in **parallel**; all nodes are connected by a network.

Example: running a query to aggregate time of all clicks on facebook over a month

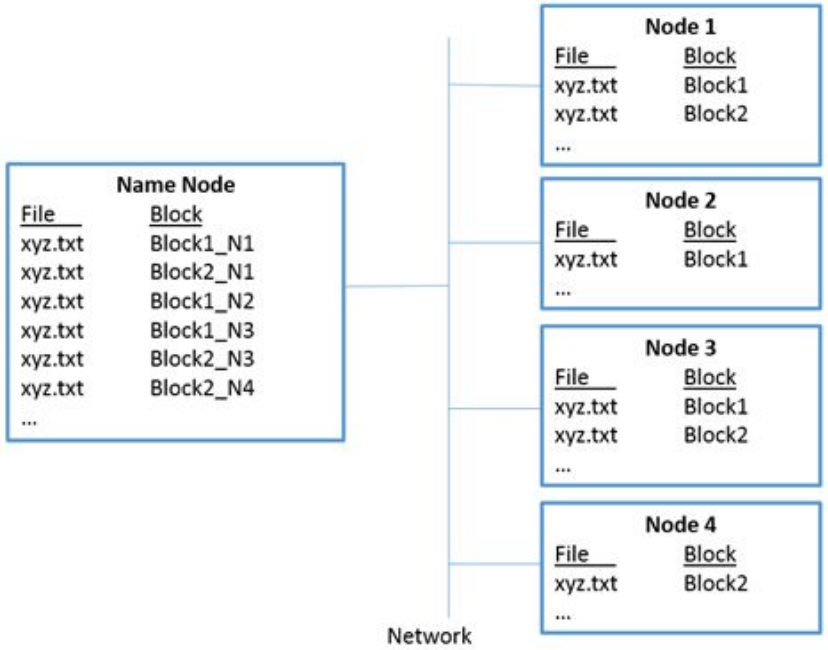


Why is Distributed Data Processing important (especially to Data Warehousing)?

- **Scalable** - Far cheaper than Centralized Data Processing for Big Data
  - 10 machines (nodes) each with 8GB RAM is much cheaper than one machine with 80 GB RAM
    - This is exponential as data gets larger
- **Reliable** - More reliable than Centralized Data Processing
  - Each machine has a chance of failure, especially when the machine is running near computing capacity
    - In Distributed Data Processing, one node can fail but the network can carry-on successfully
- **Fault Tolerant** - data is replicated and processing jobs are re-startable

What is a Distributed Data Processing Architecture?

- Now that you understand Distributed Processing, let's look at the Architecture of Storing Data across multiple nodes (or Distributed Data Processing Architecture)
    - Data can be *partitioned* or *replicated* among nodes
        - If data is **partitioned** it is split among multiple nodes
        - If it is **replicated**, each block exists on multiple nodes
            - The standard is **triple replication**



Distributed Data Processing Architecture

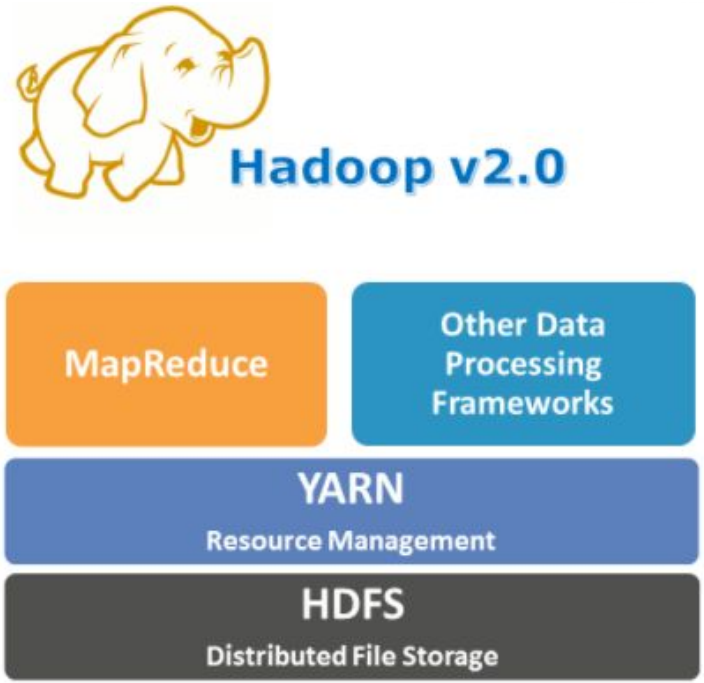
1. If data is replicated, how redundant do you want each block of data?
2. If data is replicated: will you commit *synchronously* or *asynchronously*?
    a. **Synchronous commit**: 2 phases, phase (1) sends a message to all other nodes, “can you commit transaction x?” phase (2) if all nodes reply “yes” then the transaction is committed
    b. **Asynchronous commit**: commits happen on each node when needed, and a snapshot of the master database is propagated to other nodes on a periodic basis
3. Distributed Database systems are more flexible and have higher performance than centralized systems, but this comes with a setup cost to the administrator(s)

Hadoop, Spark, and Hive (poll!)

What is Hadoop?

Hadoop is a collection of *software* and services that support highly scalable distributed processing. The software and services include:

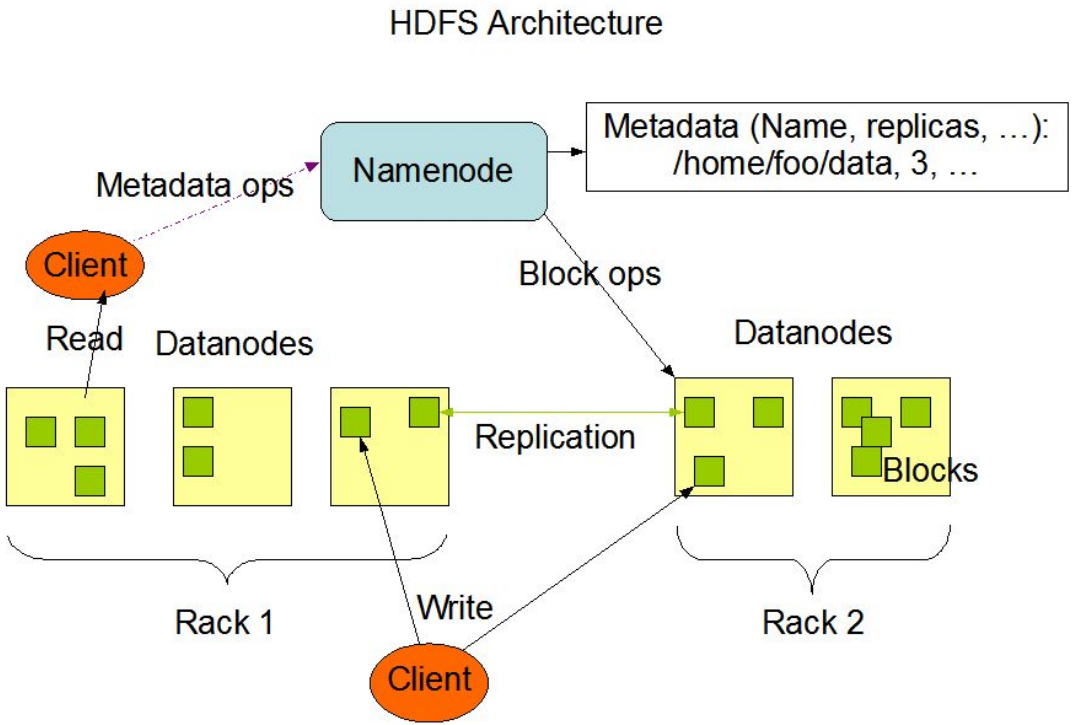
- **Storage Layer**: Hadoop Distributed File System (HDFS)
- **Scheduling Layer**: Hadoop YARN (Yet Another Resource Negotiator)
- **Execution Layer**: Hadoop MapReduce (or Spark)
- **Applications**: Hive (SQL), Flink, Storm



Hadoop's Storage Layer

Hadoop Storage Layer is called "**Hadoop Distributed File System (HDFS)**". Important notes about HDFS:

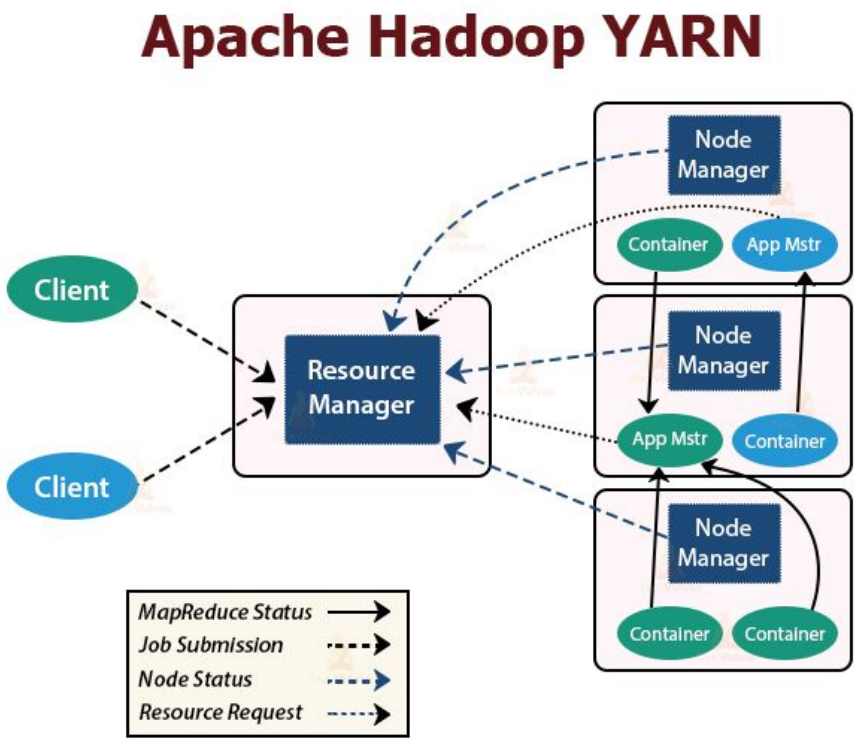
- The Replication Factor is default set to 3, that means 3 copies of each block of data
- Namenode manages the the file system overall



Hadoop's Scheduling Layer

Hadoop Scheduling Layer is called "**Hadoop YARN (Yet Another Resource Negotiator)**". Important notes about YARN:

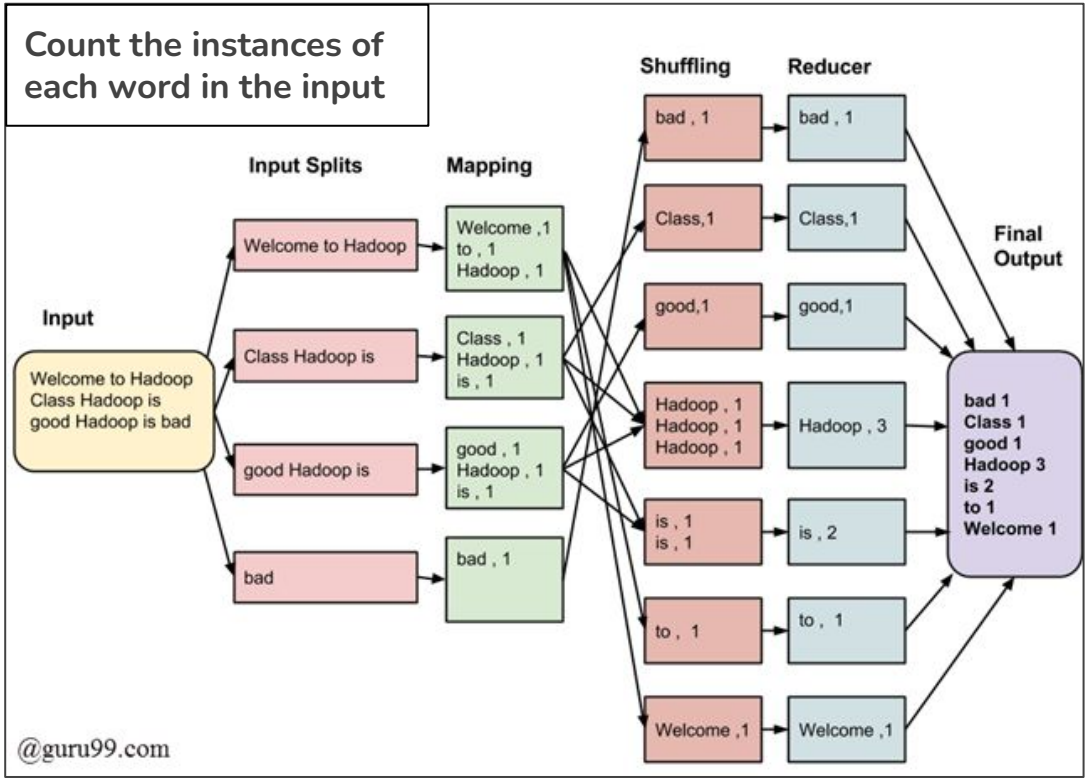
- Container: resources to process job
- Application Master: requests container from Node Manager
- Node Manager: handle resources within a node
- Resource Manager: assigns resources



Hadoop's Execution Layer

Hadoop Execution Layer is called "**Hadoop MapReduce**". Important notes about MapReduce:

- MapReduce is a program that does 2 things:
  - First, allocates work (maps) to each node
  - Second, organizes (reduces) the work of each node into a single answer



Hadoop's Applications

Hadoop has many applications built onto it's platform, some are listed below:

- **Apache Hive**: a data warehouse software project built on top of Apache Hadoop for providing data query and analysis. Hive gives an **SQL-like interface** to query data stored in various databases and file systems that integrate with Hadoop
    - Practice Hive: https://demo.gethue.com/
    - Differences from SQL to Hive: link
- **Apache Pig**: a high-level platform for creating programs that run on Apache Hadoop. The language for this platform is called Pig Latin. Pig can execute its Hadoop jobs in MapReduce, Apache Tez, or Apache Spark
- **Apache Zookeeper**: an open-source server for highly reliable distributed coordination of cloud applications. It is a project of the Apache Software Foundation

What is Apache Spark?

Apache Spark: an open-source unified analytics engine for large-scale data processing. Spark provides an interface for programming entire clusters with implicit data parallelism and fault tolerance.

- **More simply**, you can write applications in Java, Scala, Python, R, or SQL right on top of the Hadoop HDFS

```
import os
import sys

# Path for spark source folder
os.environ['SPARK_HOME']="/Users/renienj/Kohls/Kohls_Research/apache-spark/spark-1.1.0"

# Append pyspark  to Python Path
sys.path.append("/Users/renienj/Kohls/Kohls_Research/apache-spark/spark-1.1.0/python/")


# Now we are ready to import Spark Modules
try:
    from pyspark import SparkContext
    from pyspark import SparkConf

    print ("Successfully imported Spark Modules")

except ImportError as e:
    print ("Error importing Spark Modules", e)
    sys.exit(1)
```

```
Run  test
/usr/local/bin/python2.7 /Users/renienj/Kohls/Kohls_Research/spark-python/test.py
Successfully imported Spark Modules

Process finished with exit code 0
```

Homework:

1. Final Project Milestone #5 on Blackboard, due Friday, 12/6/22 at 6:00pm ET