Big Data 230 Week 1

Bio

Mike Roshak PhD

Currently: IoT/Cloud Architect Microsoft

- Big Data
- IoT
- Machine Learning

- I like hiking

Your Bio

- What you do
- Your background as it relates to big data
- What do you do for fun

Goal of this course

- Explore where/how big data is being used
- Cement concepts
- Match teaching with what big data engineers do
- Work with more tools

What do Big data engineers do?

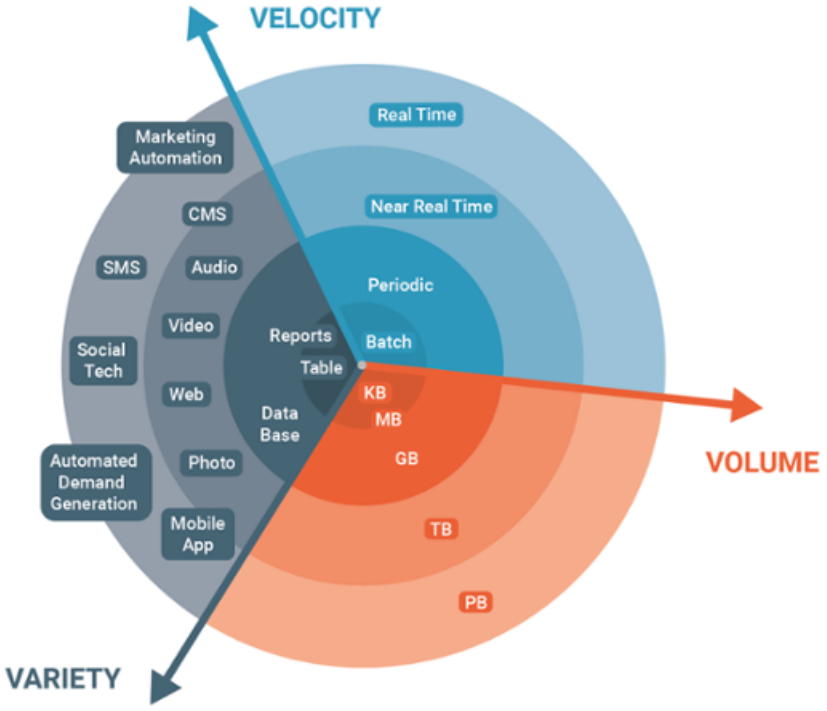
- ETL
- Process Data
- Analyze Data
- Trusted Advisor

What is Big Data

1. Volume
2. Velocity
3. Variety

The 3 V's of Big Data

G2 CROWD



What is Big Data

1. Volume
2. Velocity
3. Variety
4. **Veracity**
5. **Value**

Course Recap

Class 1 - Introduction to Big Data

- Databricks
- Spark
  - Scala
  - Spark Notebooks
  - RDDs
  - DataFrames
- HDFS
- Graph

Class 2 - Building the data pipeline

- Spark
  - Stream Processing Semantics
  - Performance Tuning
- Kafka
- Flink
- Data formats
  - Parquet
  - Avro
- Nifi
- Scalablity
- Architectures
  - Lambda
  - Kappa

Containers

- Docker for Big Data
- Docker Compose
- Kubernetes

Data Analysis

- Data Bricks
- Delta lake
- Cleansing Data
- Data Theory
- Data Analysis

Value from Data

- Sampling
- Anomaly Detection
- Feature Engineering with time series data

Smart Data

- Binary Classifiers
- Model Accuracy
- Artificial Neural Nets

Internet of Things

- IoT Hub
- Streaming Analytics
- Cosmos DB
- Blob Storage
- Serverless

Edge

- GraphFrames in Spark
- IoT Edge
- Fog Computing
- Offloading workloads to Edge

DataOps

- Data Ops
- Collaboration Tools
- Methodologies
- Processes
- ML Flow

Structure of this class

- Different than previous classes
- Hands on
- Assignments
  - Portfolio Project
  - Credits
- Fast talker
- Break up lectures
  - Class Questions
  - Class Excersizes
  - Quiz
  - Demos

Big Data Echosystem

- Linux
- Java
- Collaboration/Ownership
  - Devops
  - Machine Learning
  - Data Product
- Git

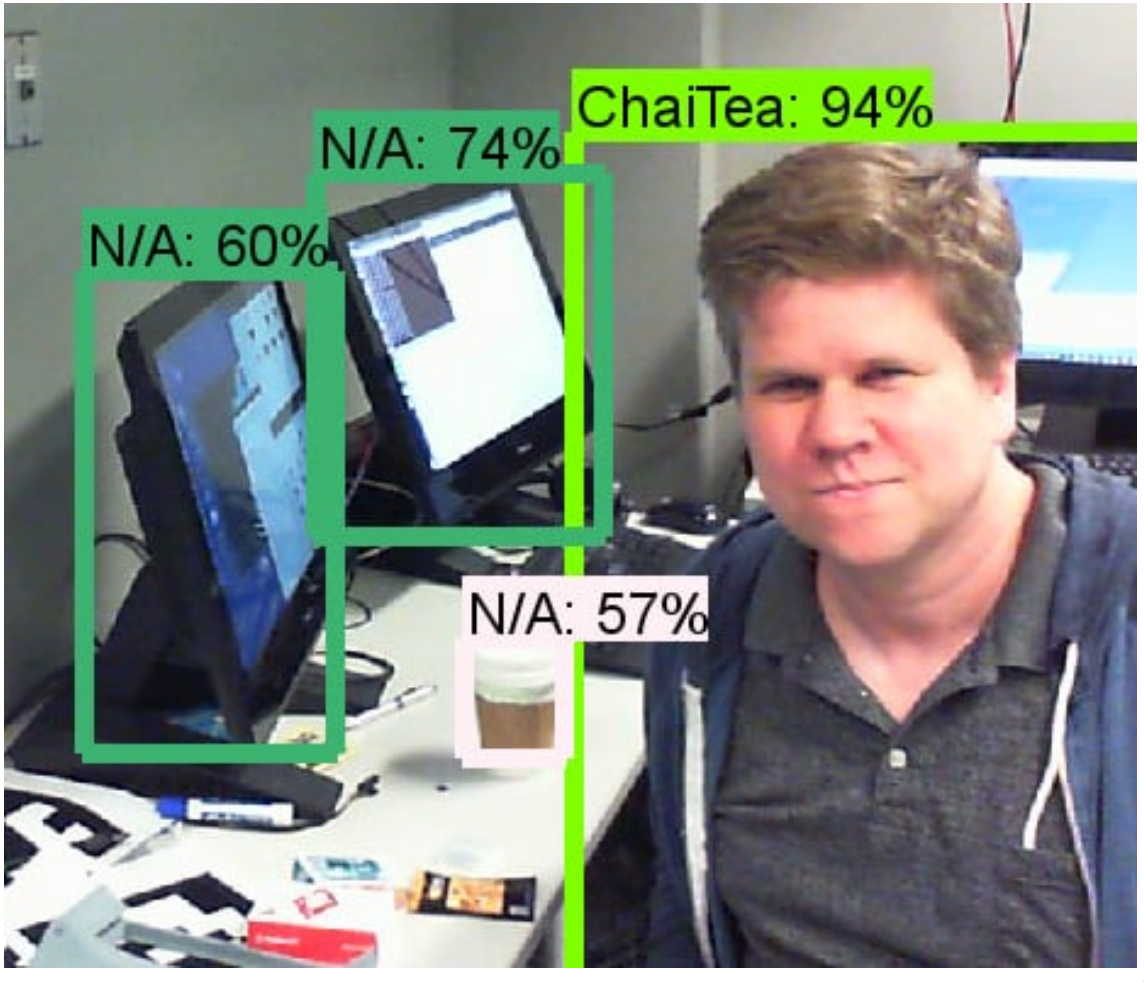
Excersize

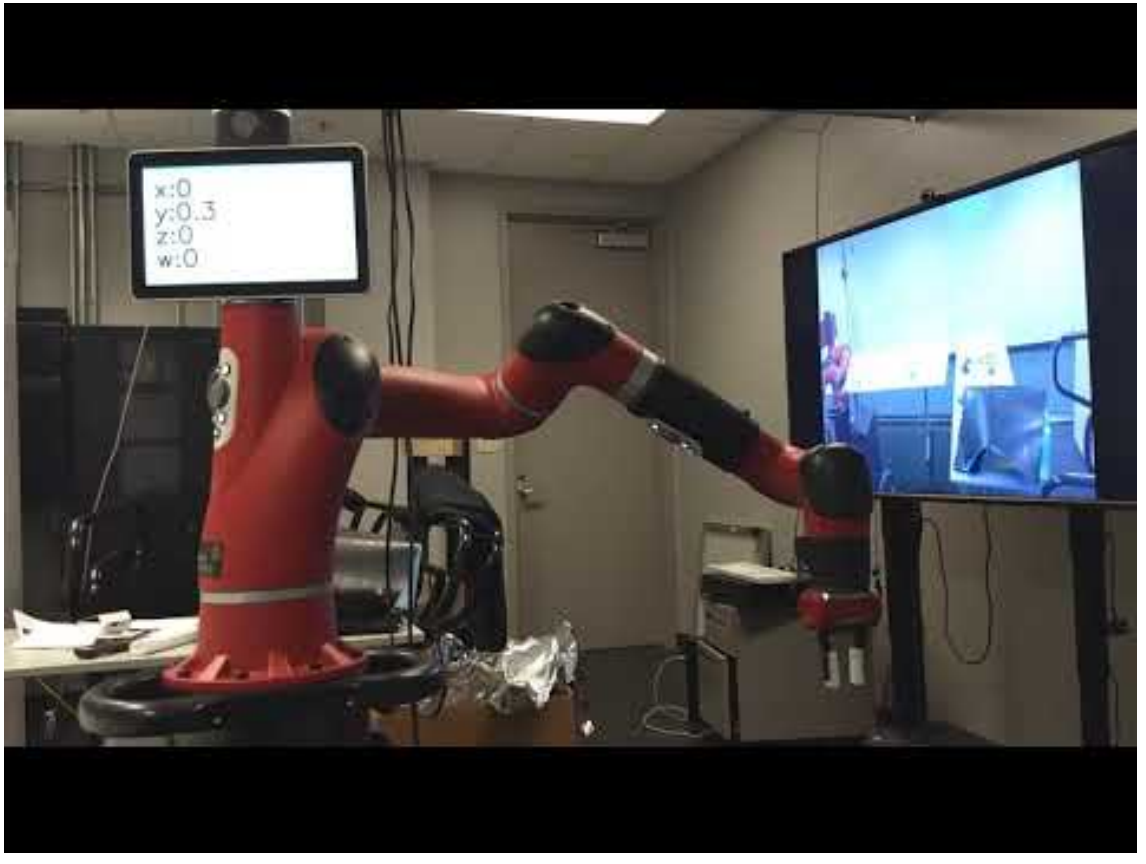
- Multi Node Cluster
- Reduce Memory Kafka
- Monitoring Kafka
- Monitoring Cloud

Portfolio Project

- The final project will be due on the last class
- The project will consist of a proposal, a presentation and a project write up including code
- The project proposal will be due April 24th.
- Everything will be in Github
- Examples

Changes in Big Data


ChaiTea: 94%
N/A: 74%
N/A: 60%
N/A: 57%


x:0
y:0.3
z:0
w:0

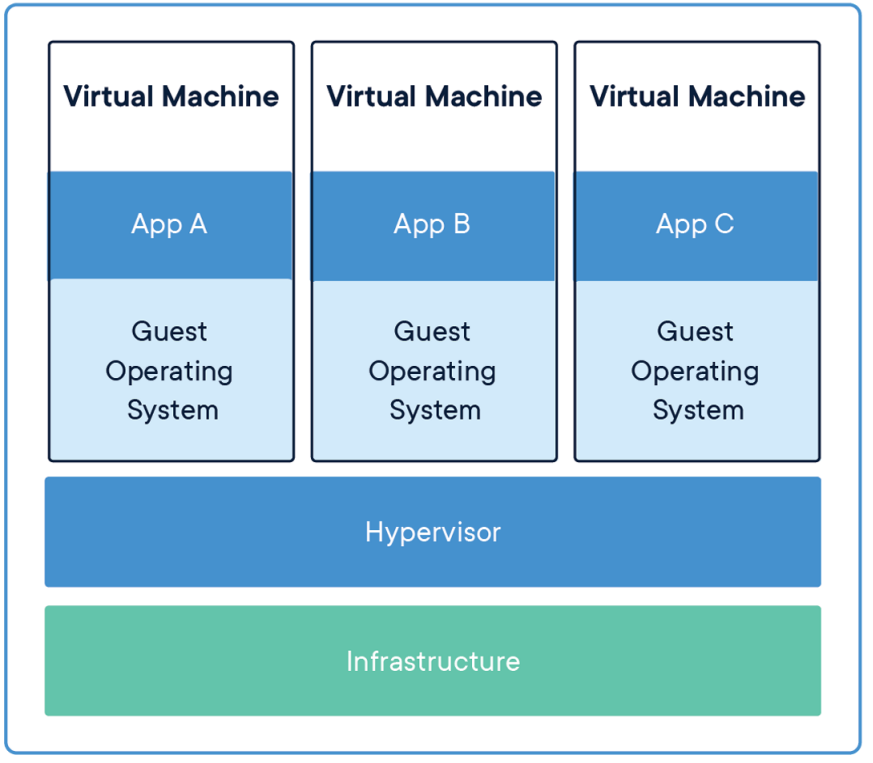
Containers

By the end of this section you should know

- How to use containers
- How to spin up your own containers
- How to test out big data tools

What is a container

Compared to VM



Containers and Orchestrators

- Containers
  - Docker
  - Mobey
- Orchistrators
  - Kubernetties,
  - ~~DC/OS~~
  - ~~Docker Swam~~,
  - ~~Apache Mesos~~,

Why containers,

- You build it you own it
- Cattle vs Pets

Evolution to Containers

- Apps do not play nicely with others
  - DLL Hell
  - 1 app = 1 machine
- VMs
  - Multiple OS on 1 device
  - Time slicing

Evolution of Containers

- Silo Systems
  - ERP
  - CRM
  - MRP
- Scale Out Architecture

Benefits

- Package software and dependencies together
- Isolated
- Runs the same environment
- Sandbox projects
- Quick Start Projects
- Scales Easy
- No OS per instance over head
- Package software and dependencies together
- Distribution mechanism
- Shared image or dockerfile
  - Easy dev environments
  - Easy CI/CD pipelines
- Can have different docker files for production and dev
- Compostable
- Incremental Build
- Map to local directory (persistance)
- Upload to repository
- CI/CD

Defined

- Defined by a Dockerfile

Under the covers

- Images
- Containers
- Daemon (API)

Docker Development

- Multi Stage Builds
  - Increase Efficency
  - Reduce Attach Surface
- Easy to Clean Up
- Volume Map local files
- IDE Support

Docker Containers

- Run (locally or remotely)
- Expose external ports
- Setup a internal network
- SSH Into

Docker Images

Create a flask app.py

```
From flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Flask Dockerized'

if __name__ == '__main__':
    app.run(debug=True,host='0.0.0.0')
```

Create a docker file

```
FROM ubuntu:latest

RUN apt-get update -y
RUN apt-get install -y python-pip python-dev build-essential

COPY . /app
WORKDIR /app
RUN pip install Flask
ENTRYPOINT ["python"]
CMD ["app.py"]
```

Build and Run

```
docker build -t helloflask .
docker run -d -p 5000:5000 helloflask
```

What went wrong

```
docker run -it -p 5000:5000 helloflask
```

Lets rebuild

```
docker build -t helloflask .
docker run -d -p 5000:5000 helloflask
```

Lets make it better

```
docker build -t helloflask .
docker run -d -p 5000:5000 -v $(pwd):/app helloflask
```

Lets modify the container

```
docker exec -it  [☢name of container☢] /bin/bash
apt-get install nano
nano app.py
```

Lets clean up

```
docker kill [☢name of container☢]

docker system prune -a
```

Class Question

Solar Panels as a Service

- Company that works in developing countries has
  - 30,000 devices that have sent 1 billion records (200 gbs a 1 year)
  - Bug in code sent 10 retry messages for every message (duplications)
  - **Can't afford expensive cloud services**
  - They have
  - Local computers (slow and unstable)
  - Power BI
  - Data is stored in Key Value store
  - They need tools to analyze and reporting
  - Aggregegation
  - Deduplication
  - Trends
  - Spikes

Commands to remember

docker images -a # list all images

docker ps -a # list all containers

Images

LIST: sudo docker images -a

Containers

LIST: sudo docker ps -a

Build

```
$ cd folder Dockerfile

# for cpu
$ docker build -f jupyter/tensorflow-notebook -t jupyter/tensorflow-notebook .
```

Run Locally

```
docker build -t myapp .
```

```
docker run --rm -t hey
```

```
docker run -it -t hey bin/bash
```

Special Environemnts

- Machine Learning GPU
- Docker Hub

Docker Hub

- Login
- Certified, Official, Verified
- Upload Docker Builds

3 Things

- Docker allows you to run programs in a:
  - Repeatable
  - Sharable way
- Docker consist of images and containers
- Docker can be build locally shared and run everywhere

Docker Excersize

- Kylin - Distributed Analytics
- Pulsar - Replacement for Kafka?
- Air FLow - Schedualing

Task

Give me a elevator pitch for a big data project you ahve not used

- Apache Big Data