

Using Streaming Views for Real Time Analytics

Rafael Borges

Today's Agenda

- Introduction
- What? Why? How?
- OSIsoft and CRISP-DM
- Architecture & Data Flow
- Live Demo

Before we begin...

- This is a Tech Talk, not a traditional lab or lecture
- It's informal and interactive
- There will be a Q&A session at the end so we can discuss the content that was presented.
- It will be very dense!

What, Why and How

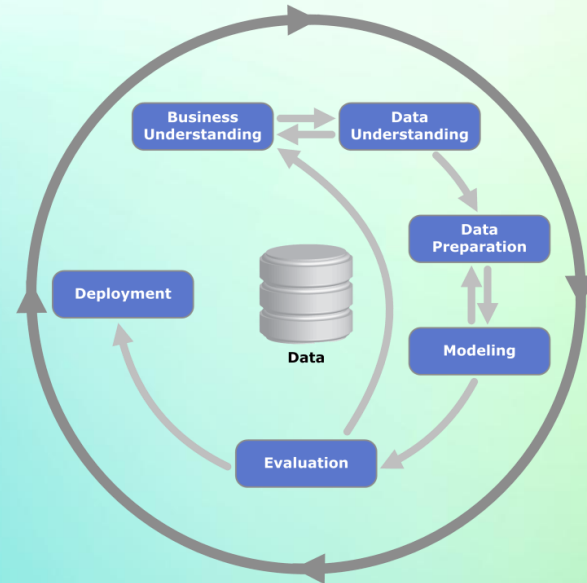
- What?
 - Stream live data to an external server in order to process it with more powerful analytical tools
- Why?
 - Because there are platforms that are more capable for specific tasks and Spark is perfect for HPC
- How?
 - Using Integrators' Streaming View, Kafka and Spark

OSIsoft and CRISP-DM

OSIsoft and CRISP-DM

Cross-industry Standard Process for Data Mining

- Business Understanding
- Data Understanding
- Data Preparation
- Modeling
- Evaluation
- Deployment

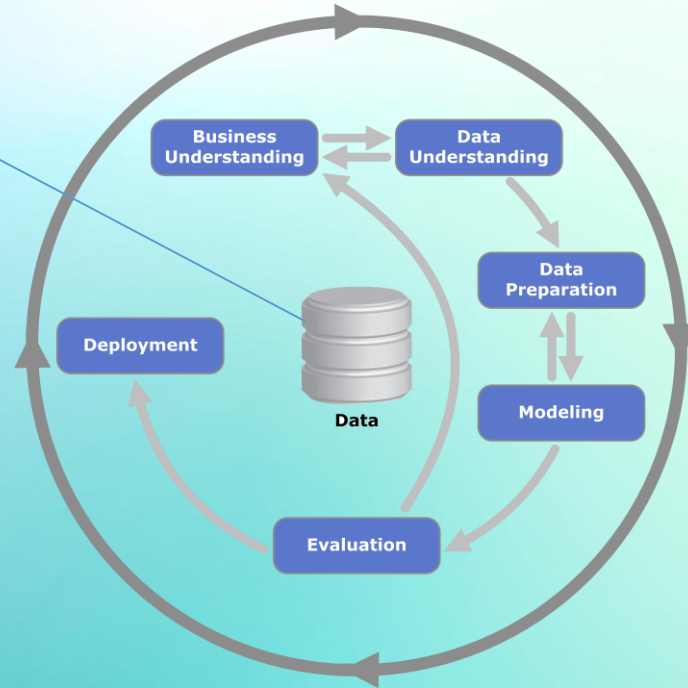


Source: [Kenneth Jensen / Wikipedia](#)

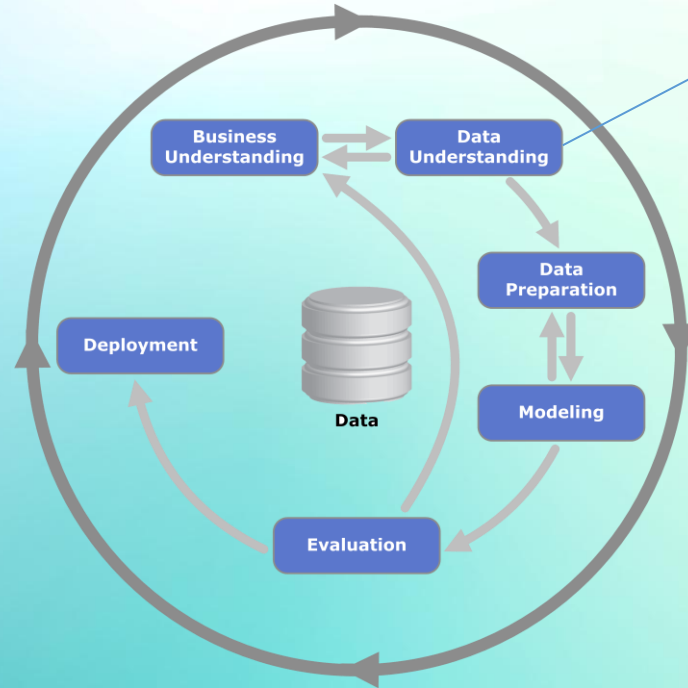
OSIsoft and CRISP-DM

Data Ingress Products

- Interfaces
- Connectors
- Custom Code



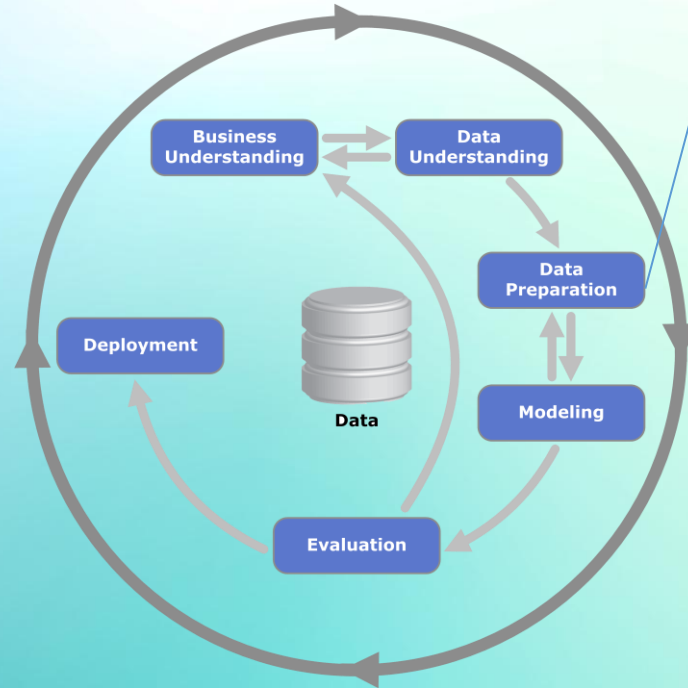
OSIsoft and CRISP-DM



Data Exploration Products

- PI Vision
- PI Datalink
- Power BI
- Python / R Scripts

OSIsoft and CRISP-DM



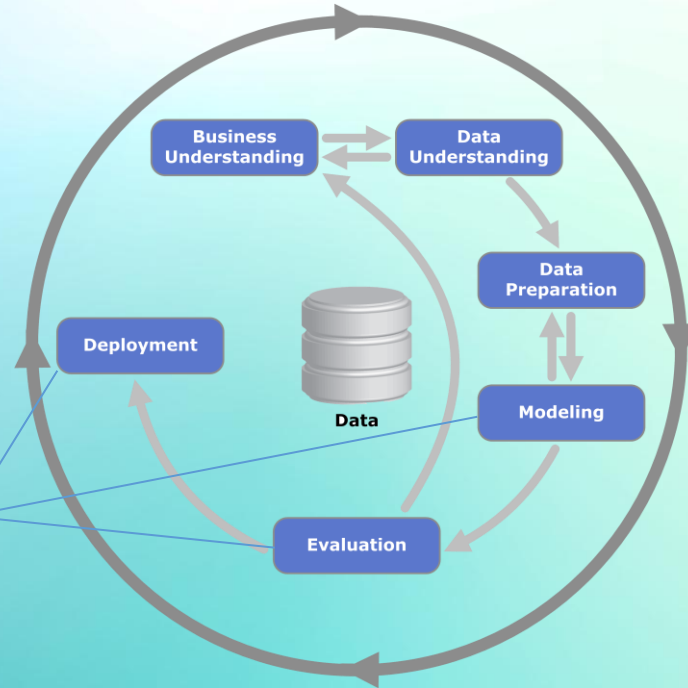
Data Wrangling Products

- PI Datalink
- PI Integrators
- PI Dev Technologies

OSIsoft and CRISP-DM

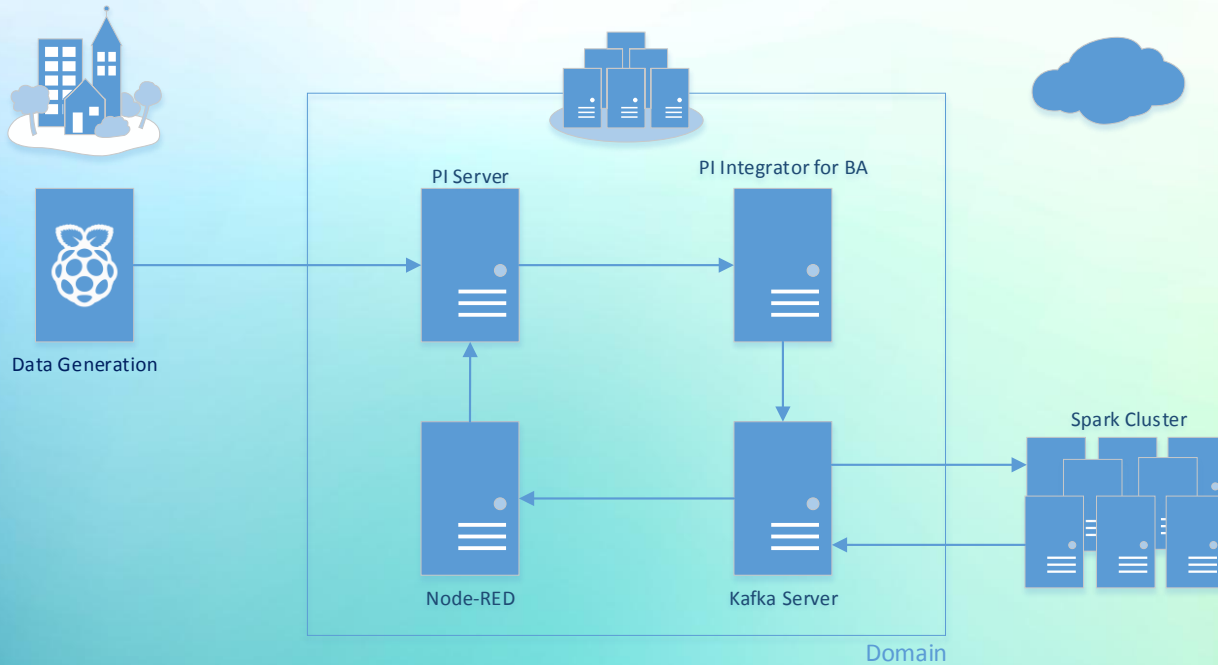
Third-Party Products

- Python (SciPy / NumPy)
- R Scripts
- Spark
- Jupyter
- Flink
- Hadoop
- ...



Architecture & Data Flow

Architecture & Data Flow



Data Generation

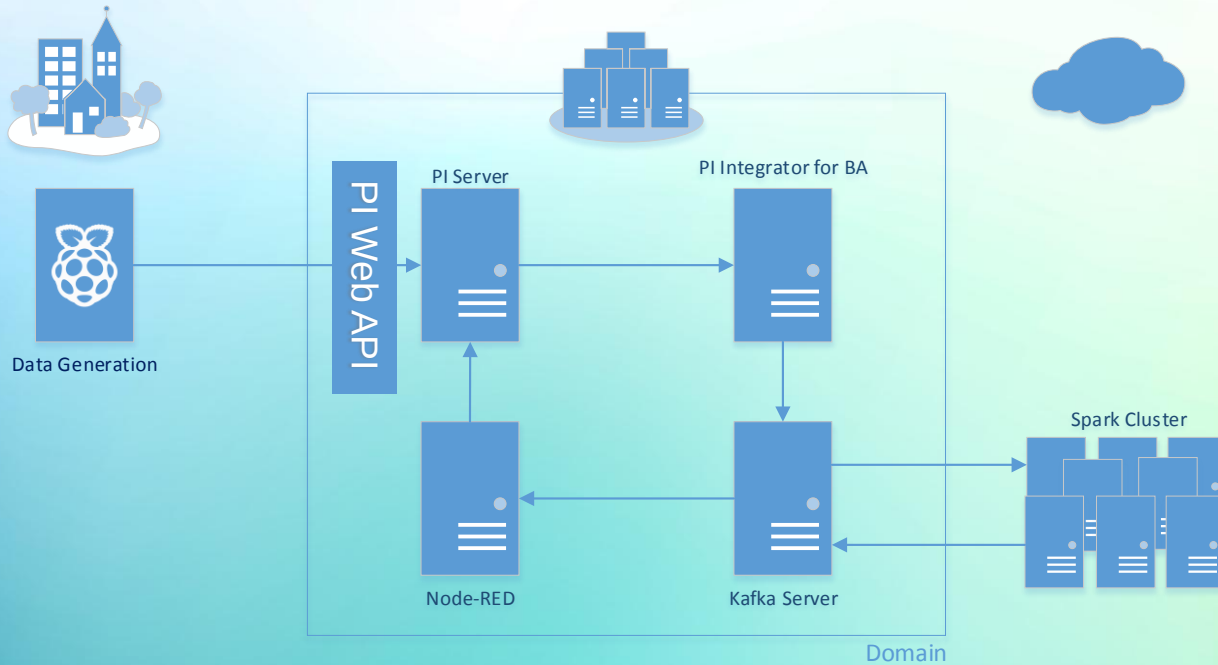
Data Generation



Data Generation



Data Generation



Data Generation – PI Web API

```
def send_data(timestamp, data, variable_info):  
    body = get_body(timestamp, data, variable_info)  
    connection = HTTPSConnection(host, context=ssl._create_unverified_context())  
    connection.request("POST", url, headers=get_header(), body=body)  
    response = connection.getresponse()
```

Data Generation – WebId 2.0 (2017+)

```
def generate_web_id_by_path(path):  
    marker = "Ab" #Because I know it's an Attribute  
    owner_marker = "E" #Because I know it's an Element  
    if path[0:2] == "\\\\":  
        path = path[2:]  
    encoded_path_bytes = path.upper().encode('utf-8')  
    encoded_string = b64encode(encoded_path_bytes).decode()  
    encoded_path = encoded_string.strip('=')\  
        .replace('+', '-').replace('/', '_')  
    return "P1{}{}{}".format(marker, owner_marker, encoded_path)
```

PI Integrator for BA Advanced Edition

PI Integrator for BA Advanced Edition



kafka



amazon
KINESIS

ORACLE®
DATABASE



amazon
REDSHIFT



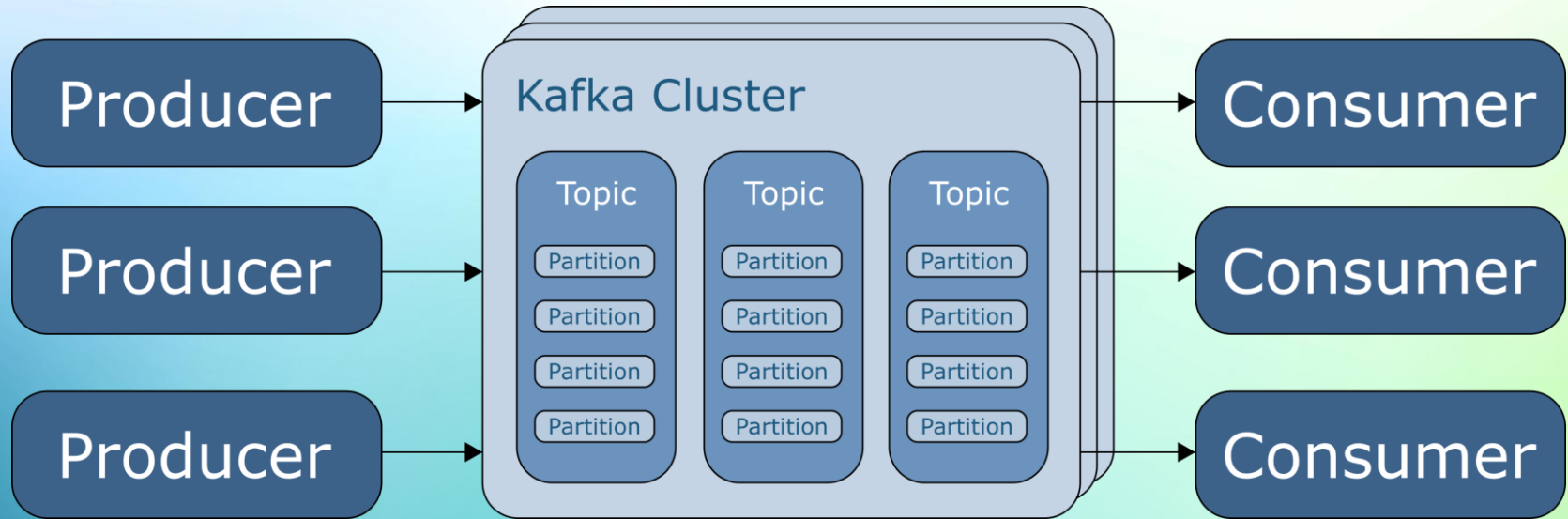
Azure IoT Hub



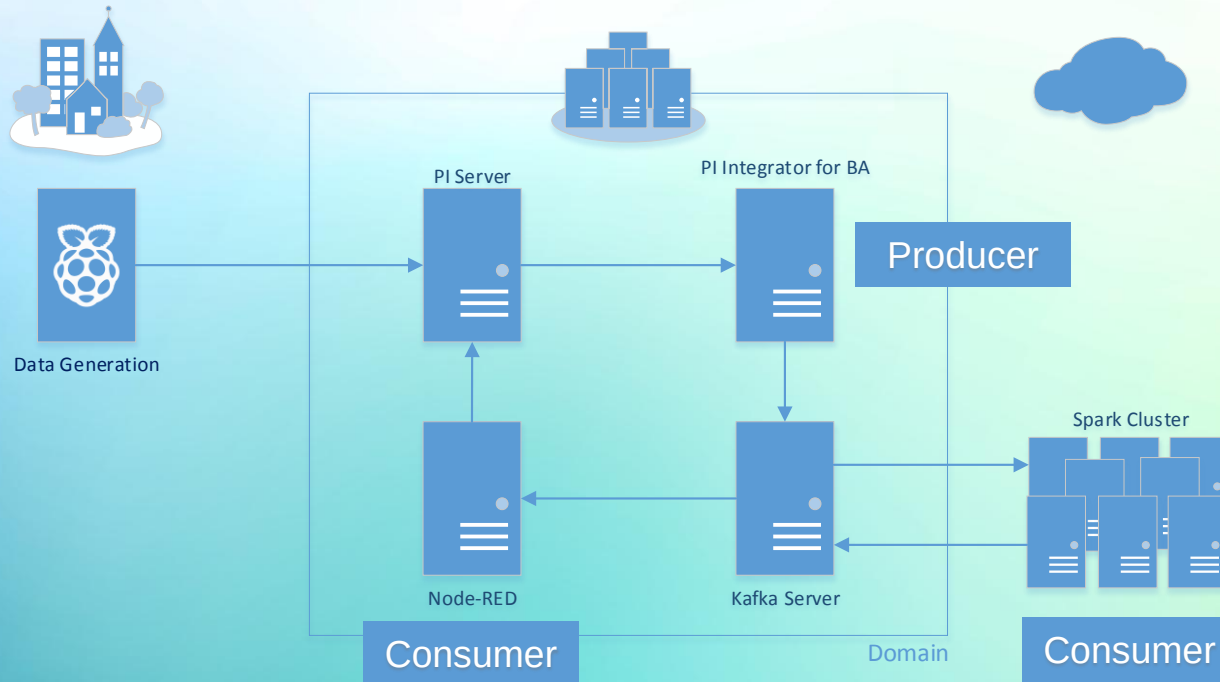
amazon
S3

Apache Kafka

Apache Kafka

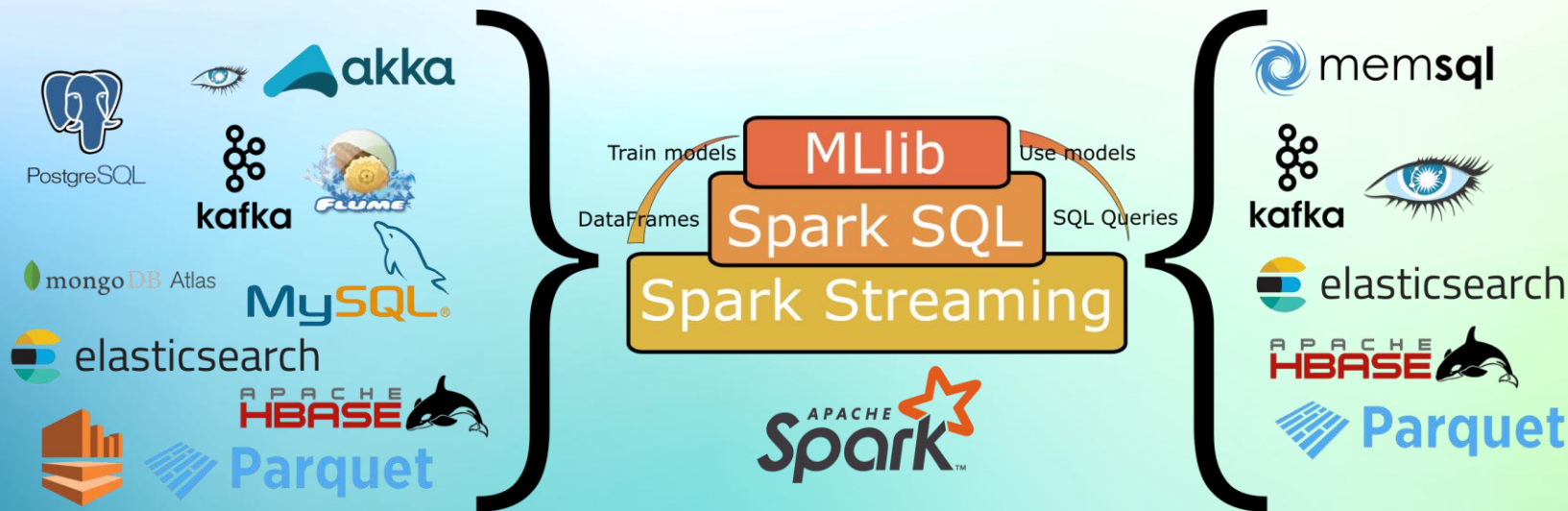


Apache Kafka

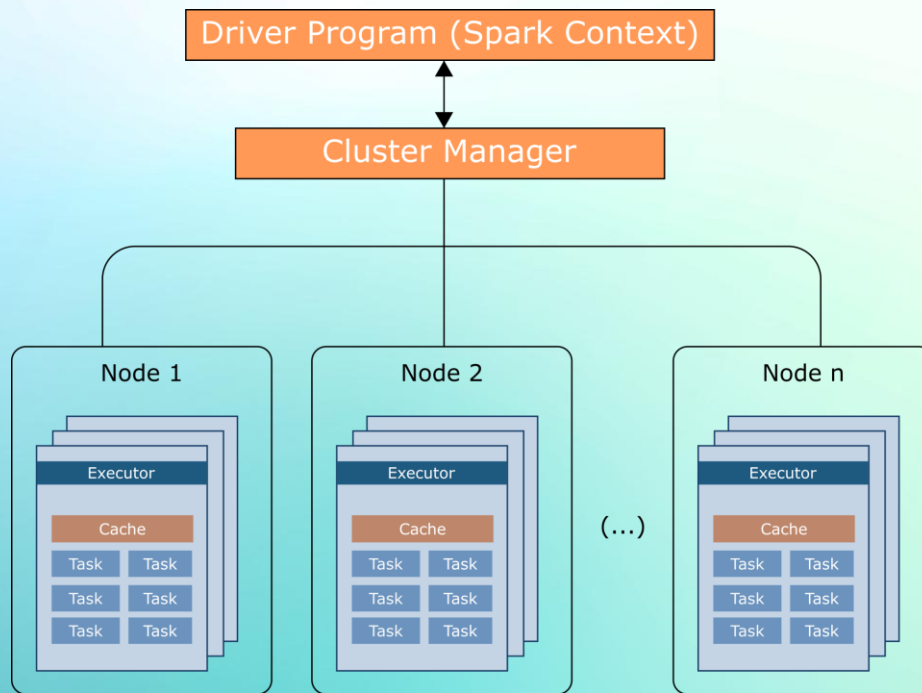


Apache Spark

Apache Spark



Apache Spark



Apache Spark

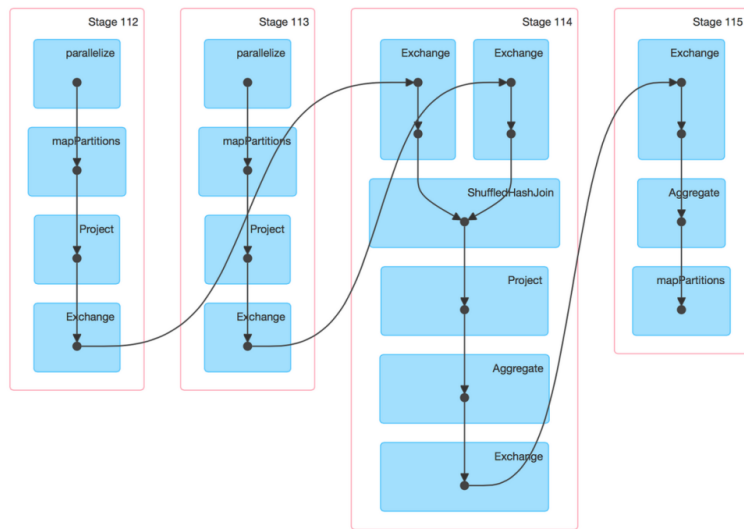
Details for Job 8

Status: SUCCEEDED

Completed Stages: 4

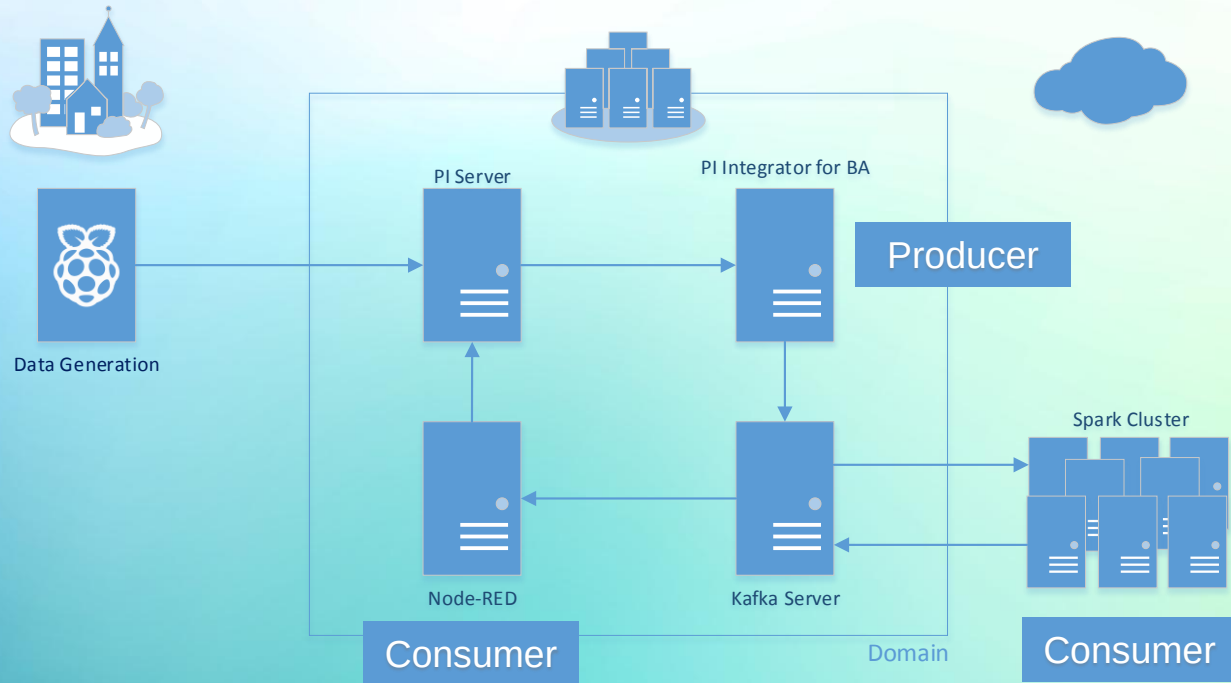
▶ Event Timeline

▼ DAG Visualization

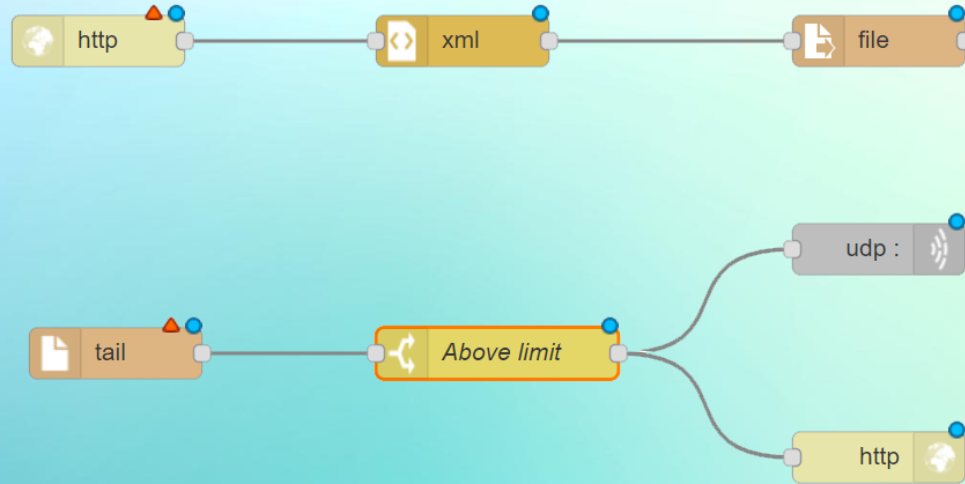


Node-RED

Node-RED



Node-RED



Conclusion

Conclusion

- It's easy to send data out of the PI System
 - PI Web API
 - PI Integrator for BA
 - Not only for Kafka, but any platform!
- It's easy to do real-time analytics with PI data
- It's easy to bring data back to the PI System

Speaker Info



- Rafael Borges
- Technology Enablement Engineer
- OSIsoft
- rborges@osisoft.com

Questions?

Please wait for
the **microphone**

State your
name & company



Please remember

TO DOWNLOAD
APP, SEARCH
OSISOFT



Download on the
App Store



GET IT ON
Google Play

