bpx energy

CMAPP – Architecture and Design

12 September, 2023

Introduction

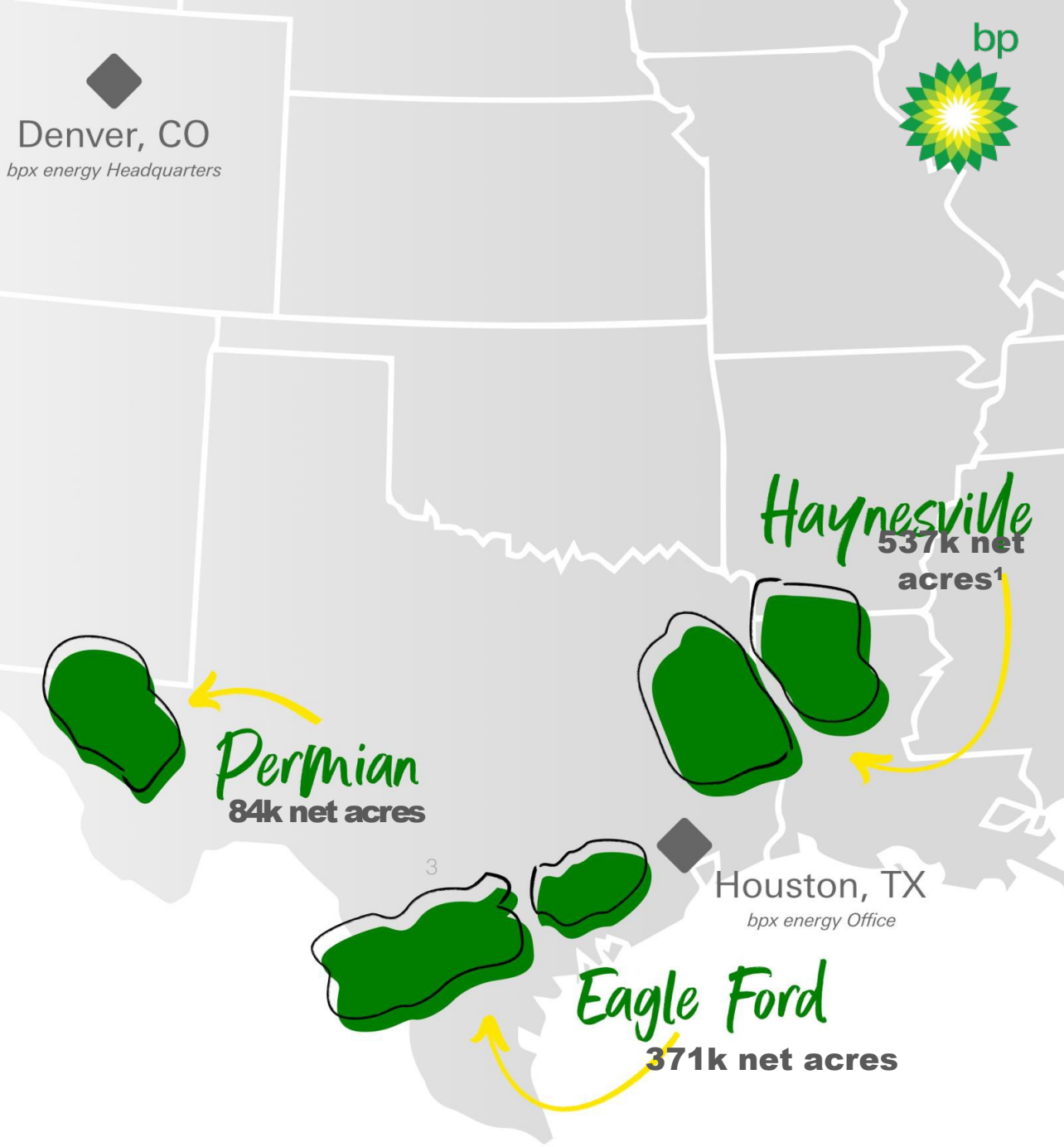
Matthew McElhaney
Data Science and Digital Solutions Manager
10 Years Industry Experience in Operations, Data Science, & Innovation
5 Years United States Army
BS Bioengineering, University of Pittsburgh
MBA, Carnegie Mellon University
MS Petroleum Engineering, Texas A&M
MS Information and Data Science, UC Berkeley.

Mike Buckner
Senior Software Engineering Platform Owner
7 years experience software development in the methane monitoring space.
BS, Chemical Engineering Colorado State University

bpx energy

bpx energy

- High quality US onshore position
- Portfolio positioned in the core of the Permian, Eagle Ford and Haynesville shale plays
- Driving operational excellence through our focus on safety and environmental stewardship



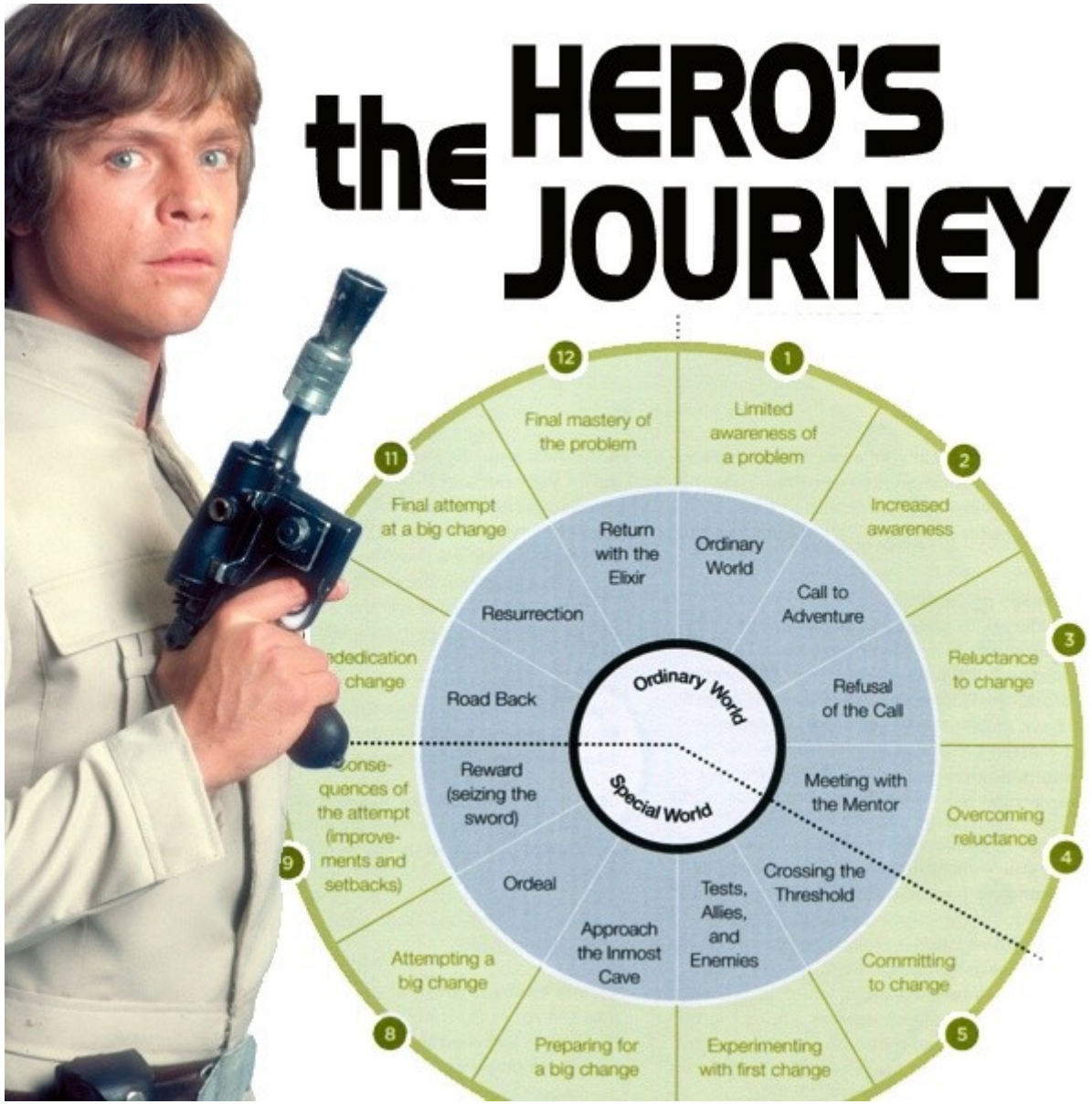
Acreage data as of YE21 [1] Excludes mineral leasehold

Why do people tell stories? The stories that tend to stick to our bones are those that teach us something. This, I believe, is the primary reason we tell stories: To teach. – Brian McDonald, Invisible Ink

The Heroes Journey

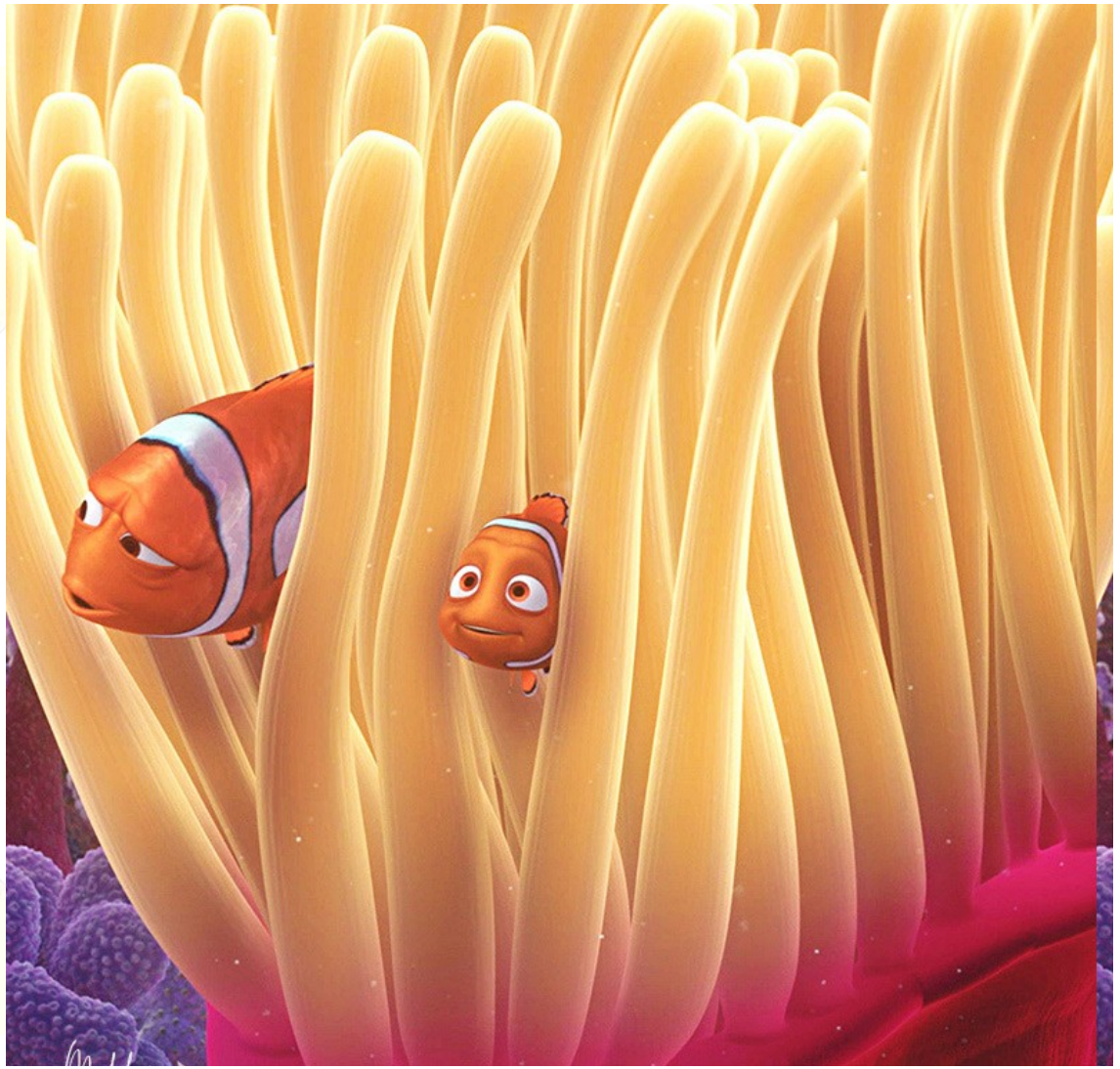
The Heroes Journey

Stage 1: The Ordinary World
Stage 2: Call of Adventure
Stage 3: Refusal of the Call
Stage 4: Meeting the Mentor
Stage 5: Crossing the threshold
Stage 6: Tests, Allies, enemies
Stage 7: The Approach
Stage 8: The Ordeal
Stage 9: Reward
Stage 10: The Road Back
Stage 11: Resurrection
Stage 12: Return with the Elixir



bpx energy

Ordinary World



bpx energy

Ordinary World

HOUSTON – BP has completed the $10.5 billion acquisition of BHP's U.S. unconventional assets in a landmark deal that will significantly upgrade BP's U.S. onshore oil and gas portfolio and help drive long-term growth.

The acquisition – which was announced in July and closed as scheduled on October 31 – adds oil and gas production of 190,000 barrels of oil equivalent per day (boe/d) and 4.6 billion oil equivalent barrels (boe) of discovered resources in the liquids-rich regions of the Permian and Eagle Ford basins in Texas and in the Haynesville natural gas basin in East Texas and Louisiana.

Following integration, the transaction will be accretive to earnings, is estimated to generate more than $350 million of annual pre-tax synergies and is expected to boost Upstream pre-tax free cash flow by $1 billion, to $14-15 billion in 2021.

"By every measure, this is a transformational deal for our Lower 48 business. It is an important step in our strategy of growing value in Upstream and a world-class addition to BP's global portfolio," said Bernard Looney, BP's Upstream chief executive. "We look forward now to safely integrating these great assets into our business and are excited about the potential they have for delivering growth well into the next decade."

Call To Adventure

bpx energy

Call To Adventure

Help!?

Chris Hines
To Matthew McElhaney

Reply | Reply All | Forward

Fri 2/22/2019 8:43 AM

Follow up. Start by Friday, February 22, 2019. Due by Friday, February 22, 2019.
You forwarded this message on 2/22/2019 2:10 PM.

RE: MidCon CapEx Daily Flowback-Database name and location
Outlook item

Howdy,
Got a question for you if you and not sure if you are the man that can help but.....

Challenge:
Flowback reports
We get one flowback report per well (excel version) every 6 hours with data from the previous 6 hours + previous hours and days of flowback operations.
When you have 6 wells in flowback that is 24 emails a day.
The information is in an excel spreadsheet that then needs to be pasted into a database of some sort for the RE's perform analysis.
Once the analysis is complete only that RE has access to their personal database and the data has a tendency to disappear

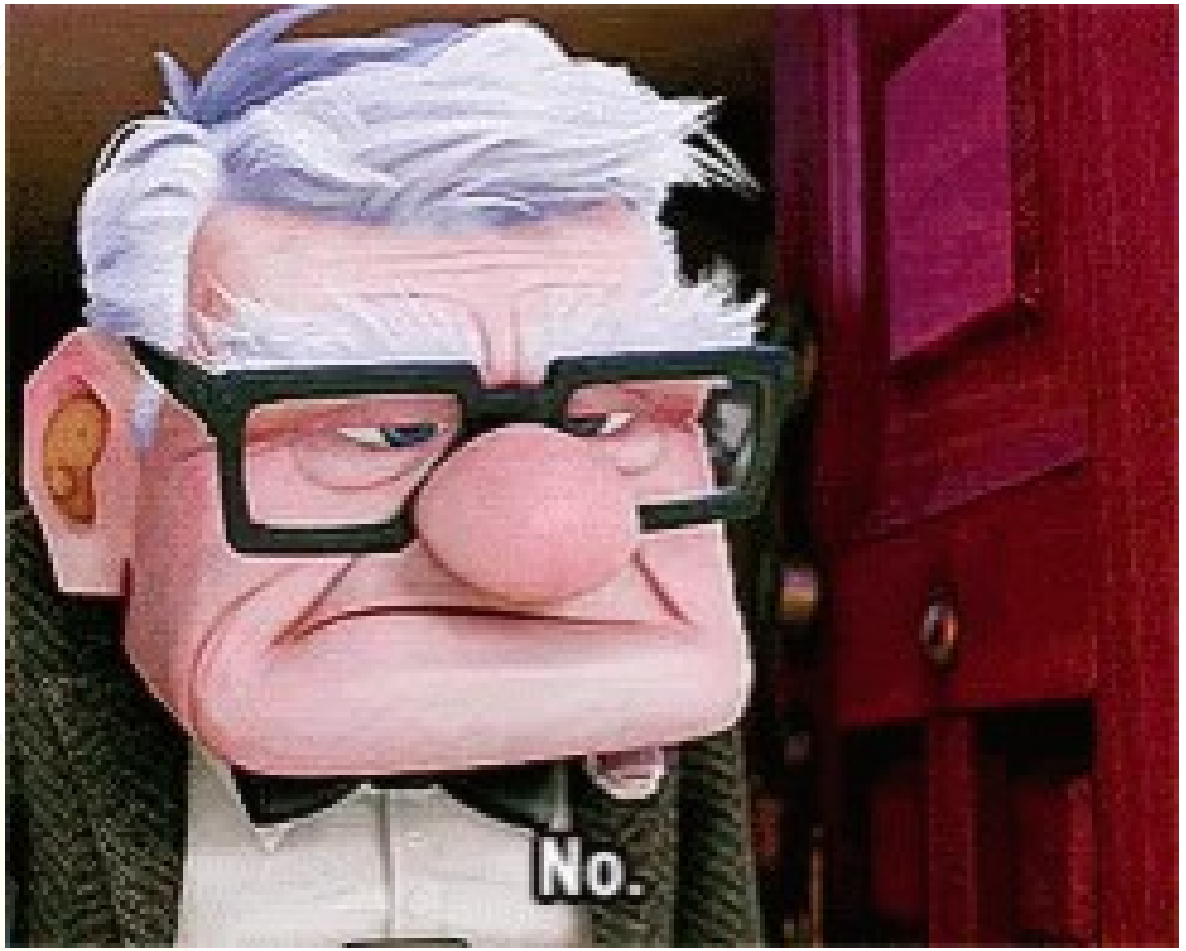
Solution:
A standard BP flowback report that can be sent via email to a central email address where a bot takes the info and places in a centralized database.
Create a PowerBI workfront that:

1. Can be accessed at anytime with all the data and viewed from a phone
2. We can create a report showing all the wells and their performance that is sent via email to a distribution list by BU every 12 hours. (Basically replacing the 24 emails a day with 2 but still has the same information)

bp

bpx energy

Refusal Of The Call



bpx energy

Crossing The Threshold

bpx energy

Crossing The Threshold

bpx energy

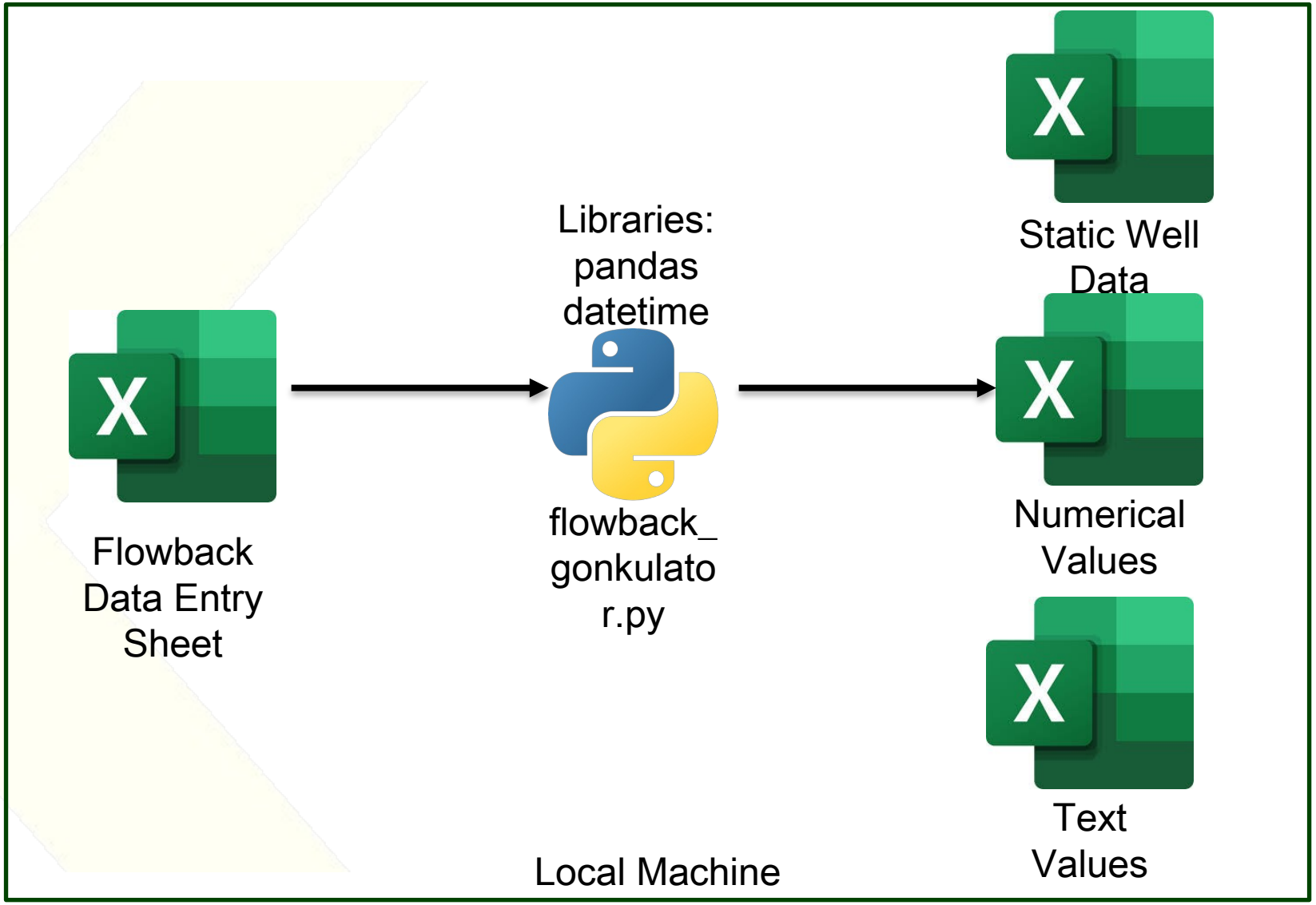
Crossing The Threshold

A	B	C	D	E	F	G	H
Time	Date	Tubing Pressure	Production Casing Pressure	Intermediate Casing Pressure	Surface Casing Pressure	Well Head Choke	Well Head Temp (F)
2:00 PM	6/8/2019	3050	1100			10/64	99
3:00 PM	6/8/2019	3050	1650			10/64	108
4:00 PM	6/8/2019	3050	2400			10/64	109
5:00 PM	6/8/2019	3050	2900			10/64	110
6:00 PM	6/8/2019	3050	3200			10/64	110
7:00 PM	6/8/2019	3850	3300			10/64	128
8:00 PM	6/8/2019	3850	3350			10/64	121
9:00 PM	6/8/2019	3900	3350			10/64	110
10:00 PM	6/8/2019	3900	2600			10/64	93
11:00 PM	6/8/2019	3900	2050			10/64	93
12:00 AM	6/9/2019	3900	2000			10/64	93
1:00 AM	6/9/2019	3900	1500			10/64	0
2:00 AM	6/9/2019	3900	1350			10/64	0
3:00 AM	6/9/2019	3900	1250			10/64	0
4:00 AM	6/9/2019	3900	1200			10/64	0
5:00 AM	6/9/2019	3900	1100			10/64	0
6:00 AM	6/9/2019	3900	1000			10/64	0
7:00 AM	6/9/2019	3900	900			10/64	79
8:00 AM	6/9/2019	3900	900			10/64	81
9:00 AM	6/9/2019	3900	800			10/64	85
10:00 AM	6/9/2019	3900	800			10/64	87
11:00 AM	6/9/2019	3900	700			10/64	90
2:00 PM	6/10/2019	3600				6/64	111
3:00 PM	6/10/2019	3600				6/64	107

Instructions | Static Data | **Hourly Data**

bpx energy

Crossing The Threshold



bpx energy

Crossing The Threshold

SQLQuery5.sql - s...ney@bpx.com (129)) | SQLQuery4.sql - s...ney@bpx.com (144)) | SQLQuery3.sql - s...ney@bpx.com (137)) | SQLQuery2.sql - s...ney@bpx.com (128))*

```sql
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP (1000) [REC_SRC]
      ,[HASH_DIFF]
      ,[SRC_READ_DT]
      ,[WELL_UID]
      ,[Metric_Date]
      ,[Metric_Type]
      ,[Metric_Value]
  FROM [Flowback].[Well_Metric_Hourly]
```

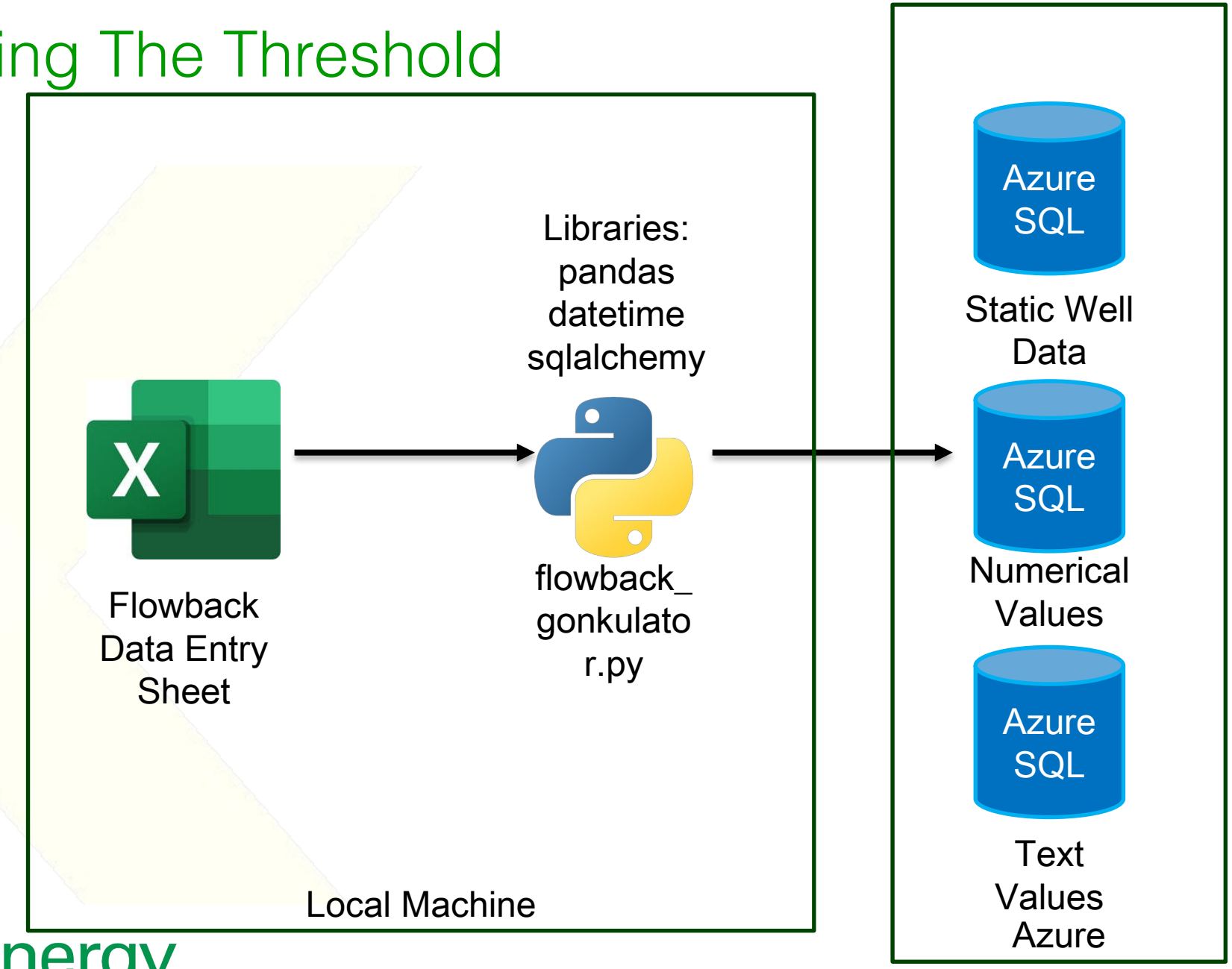
100 %

Results | Messages

	REC_SRC	HASH_DIFF	SRC_READ_DT	WELL_UID	Metric_Date	Metric_Type	Metric_Value
48	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Oil API	37
49	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Oil PL Pressure	3000
50	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Water BWPH	200
51	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Water Chlorides	75
52	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Total Fluid (bph)	8955
53	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Sand Measurement (ml)	60
54	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Flare Gas (mcfd)	60
55	FINAL Validated BP Flowback Report Rev CH 0328...	0	2019-04-08 10:35:12.000	12345678900	2019-04-08 05:00:00.000	Impurities, H2S (ppm)	6789

bpx energy

Crossing The Threshold



bpx energy

Test, Allies, Enemies: Architecture Review Board

bpx energy

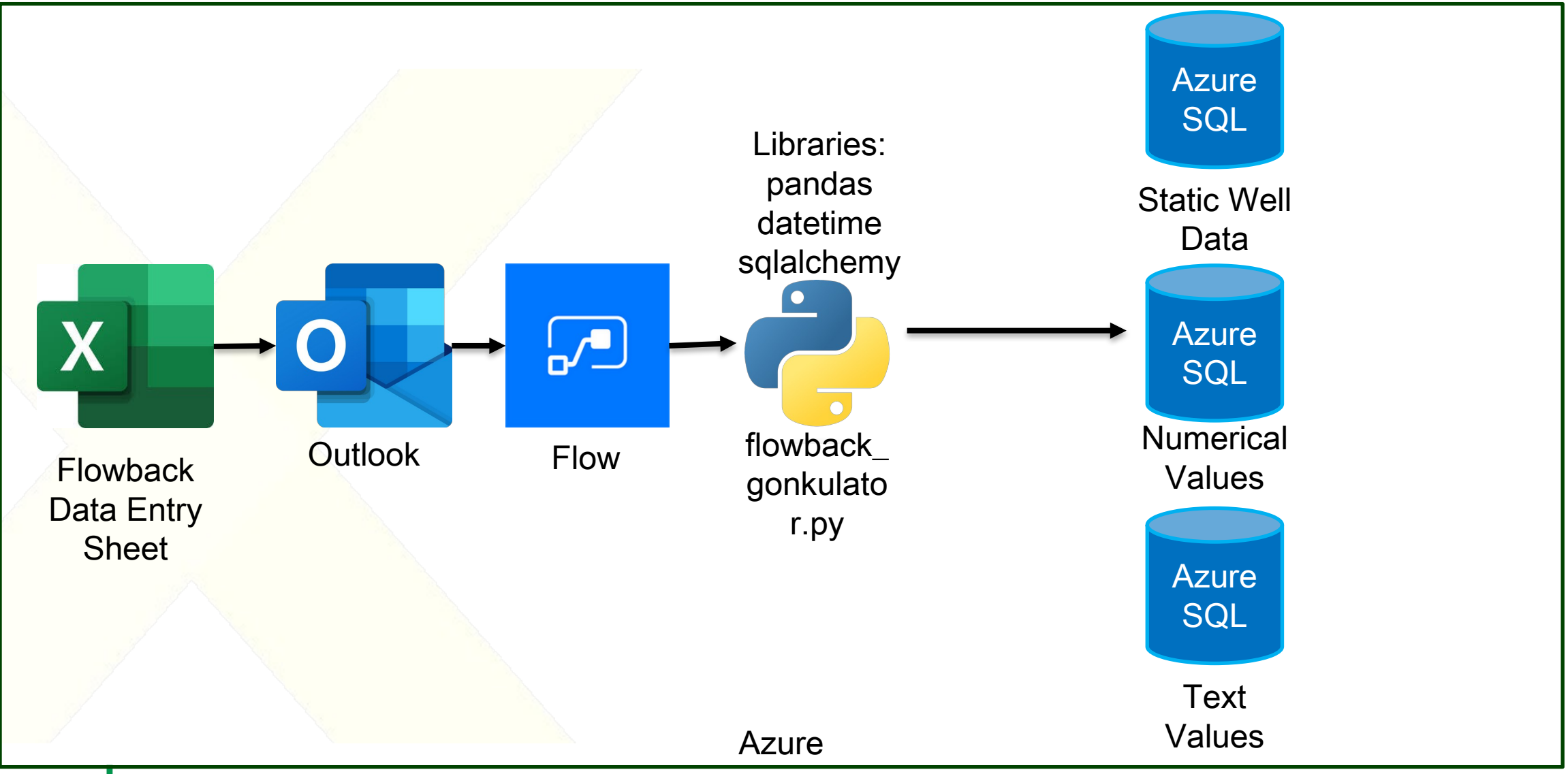
Test, Allies, Enemies: Digital Security

bpx energy

The Approach: Industrializing and Scaling

bpx energy

The Approach: Industrializing and Scaling



bpx energy

The Approach: Industrializing and Scaling

Wells Loaded

MM Matthew McElhaney
To Mike Ward; Chris Hines; Matthew Blose

You replied to this message on 4/25/2019 3:35 PM.

Good afternoon,

Below is a screenshot of wells that have been loaded into the database-if this doesn't look right let me know. Thank you.

WELL_UID	Pad_Name	Well_Name
14380514000	STS C 3 PAD	STS NORTH 7H
14380515000	STS C 3 PAD	STS NORTH 8H
14380516000	STS C 3 PAD	STS NORTH 9H
14384047000	STS C 3 PAD	STS NORTH 10H
14401872000	HSS 113 10 Pad B	HSS-113 10 4H
14401870000	HSS 113 10 Pad B	HSS-113 10 2H
14401667000	State Blake	State Blake 57-T3-46 W103H
14401871000	HSS 113 10 Pad B	HSS-113 10 3H
14401666002	State Blake	State Blake 57-T3-46 W102H
14401506000	State Blake	State Blake 57-T3-46 W104H

bpx energy

The Ordeal: Pilot

bpx energy

The Ordeal: Pilot

Flowback Reports

Matthew McElhaney
To Chris Hines; Matthew Blose (MATTHEW.BLOSE@BPX.COM); Mike Ward

2019-08-05 12-05-31State Roy Kimsey 56-T2-46 W103H_07292019_Flowback Report.xlsx
221 KB

Good morning team,

A few errors on Flowback loading:

State Roy Kimsey-0s in Sales Rate with no dates (attached)
JC Martin- Times without dates (attached)

Edwards 5, 6, 9, 10: not opening in excel, giving an xml error. Were the sheets altered?

RE: Power BI well performance report.

Matthew McElhaney
To Marshall Johnson
Cc Mickey Moulton; Timothy Credeur; Michael Mulder

2019-09-04 12-31-16Gentry32_FB_09419.xls.xlsx
221 KB

Good morning Marshall,

I went through a sample of the files and the errors are below in red. There are two primary issues:

1. There can only be a single value per day per time per well. If there are duplicates the load fails. I
   duplicate row.
2. Some of the files are corrupted and need to get recovered when I open them (example attached

RE: Flowback Errors - Future dates

Matthew McElhaney
To Matt O'Neal

We have Miss Cleo out on flowback sites.

From: Matt O'Neal <Matthew.Oneal@bpx.com>
Sent: Tuesday, September 17, 2019 9:21 AM
To: Matthew McElhaney <MATTHEW.MCELHANEY@BPX.COM>
Subject: Flowback Errors - Future dates

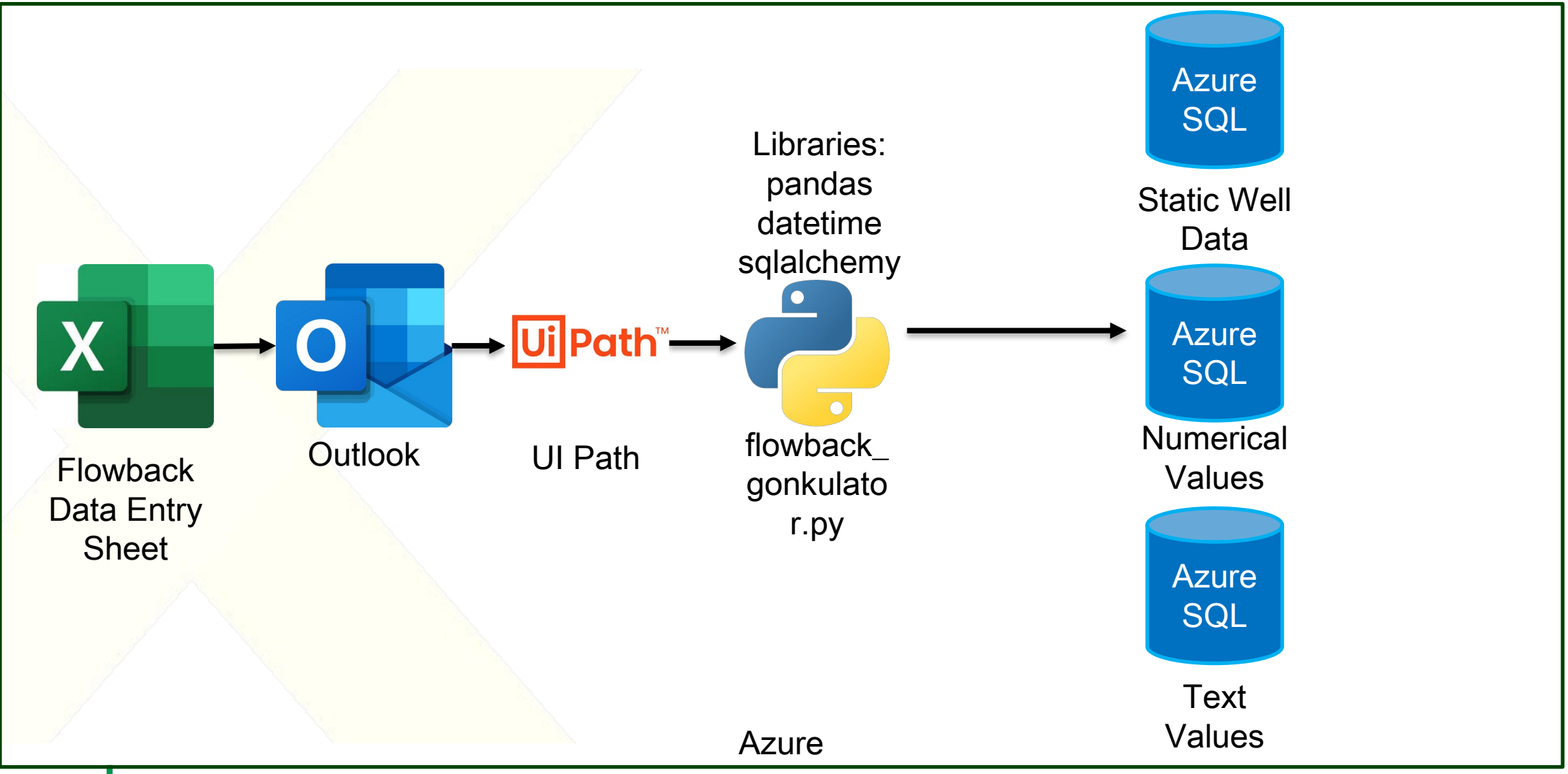
```
SELECT
  f.*
FROM
  [bpx_flowback].[Well_Metric_Hourly] f
WHERE
  f.Metric_Date > GETDATE()
ORDER BY
  WELL_UID, Metric_Date
```

Gives 657 rows with measurements in the future ...

I will hand correct the one in my sample data set but you may wanna add

-matto-

The Ordeal: Pilot



bpx energy

The Reward

bpx energy

The Reward

Re: Flowback Report Ingestion

Chris Hines

To Matthew McElhaney; Mike Ward; Matthew Blose

Start your reply all with: You're welcome. | You're very welcome. | Happy to help. | Feedback

Thank you very much. Really appreciate your support throughout this process

Hines

bpx energy

Resurrection – Design Improvements

bpx energy

Resurrection – Design Improvements

Flowback Tracker Rollout

Reply | Reply All | Forward

Wed 1/27/2021 5:49 PM

Cole Larrabee (Captech)
To Matthew Blose
Cc Matthew McElhaney; Carolyn Pescheret (Captech); Darin Thompson (BPX Energy)

General

Flowback Tracker 01-27-21.xlsx
328 KB

Hey Matt,

As discussed, here's the flowback tracker we'd like you to roll out. It includes all the changes we have discussed throughout our conversations – OOOOa columns, enhanced data validation rules, report compliance section, etc.

I'll be waiting on your email to push the code changes into production so if you could CC me that'd be great. Thanks again for your help with this process.

Regards,
Cole

Cole Larrabee
Data Engineer | CapTech Consulting
M: (804)516-4815
E: cole.larrabee@bpx.com

bpx energy

Resurrection – Design Improvements

Report Compliance

Please Verify All Tests Pass Before Saving and Emailing

Measure *(see note for description)*	**Status**	**Message**
Corporate ID	FAIL	Enter a value for the Corp ID
Flowback Start Time	FAIL	Please enter a value for Flowback Start Time
Time and Date Match	PASS	
First Gas Date and Time	PASS	
First Gas Seperator Date and Time	PASS	
Comments Not Provided	PASS	

Return With The Elixir

bpx energy

Return With The Elixir

Search

```
1   select count(*) from edh.bv_edh_flowback_well_metric_hourlyasis
```

Output | Execution Logs | **Query Results**

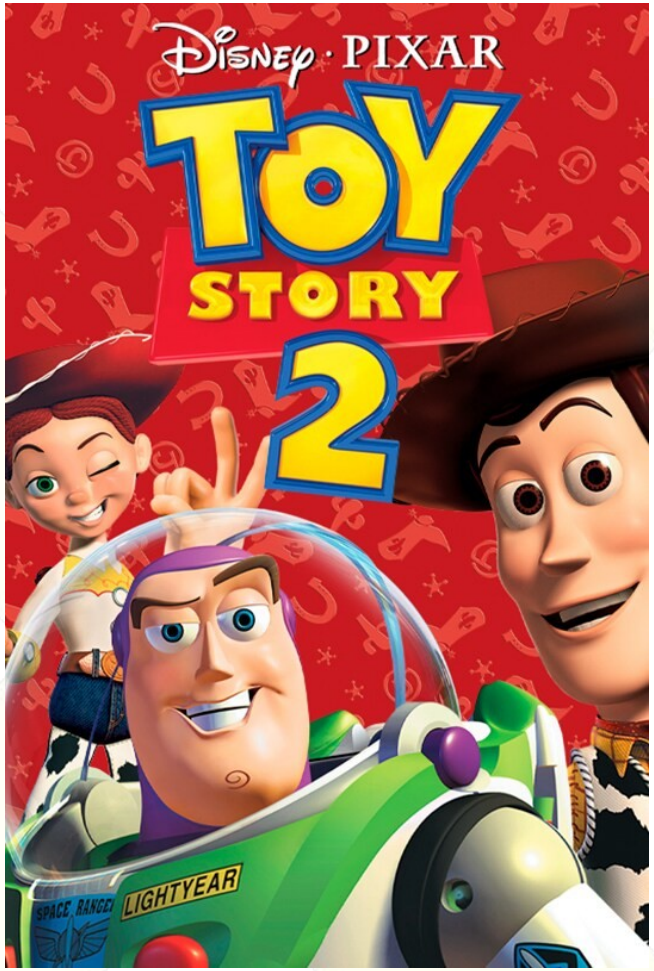
RESULTS **EXECUTION TRACE**

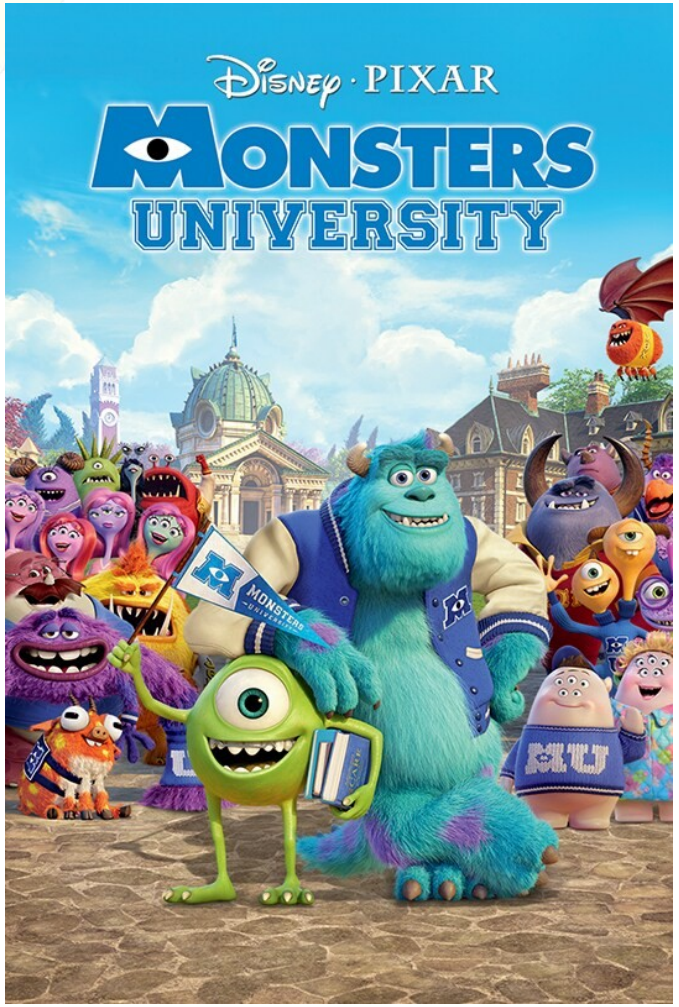
1 rows 1 columns received select count(*) from edh.bv_edh_flowback_well_metric_hourlyasis

count
9025800

bpx energy

The Sequel

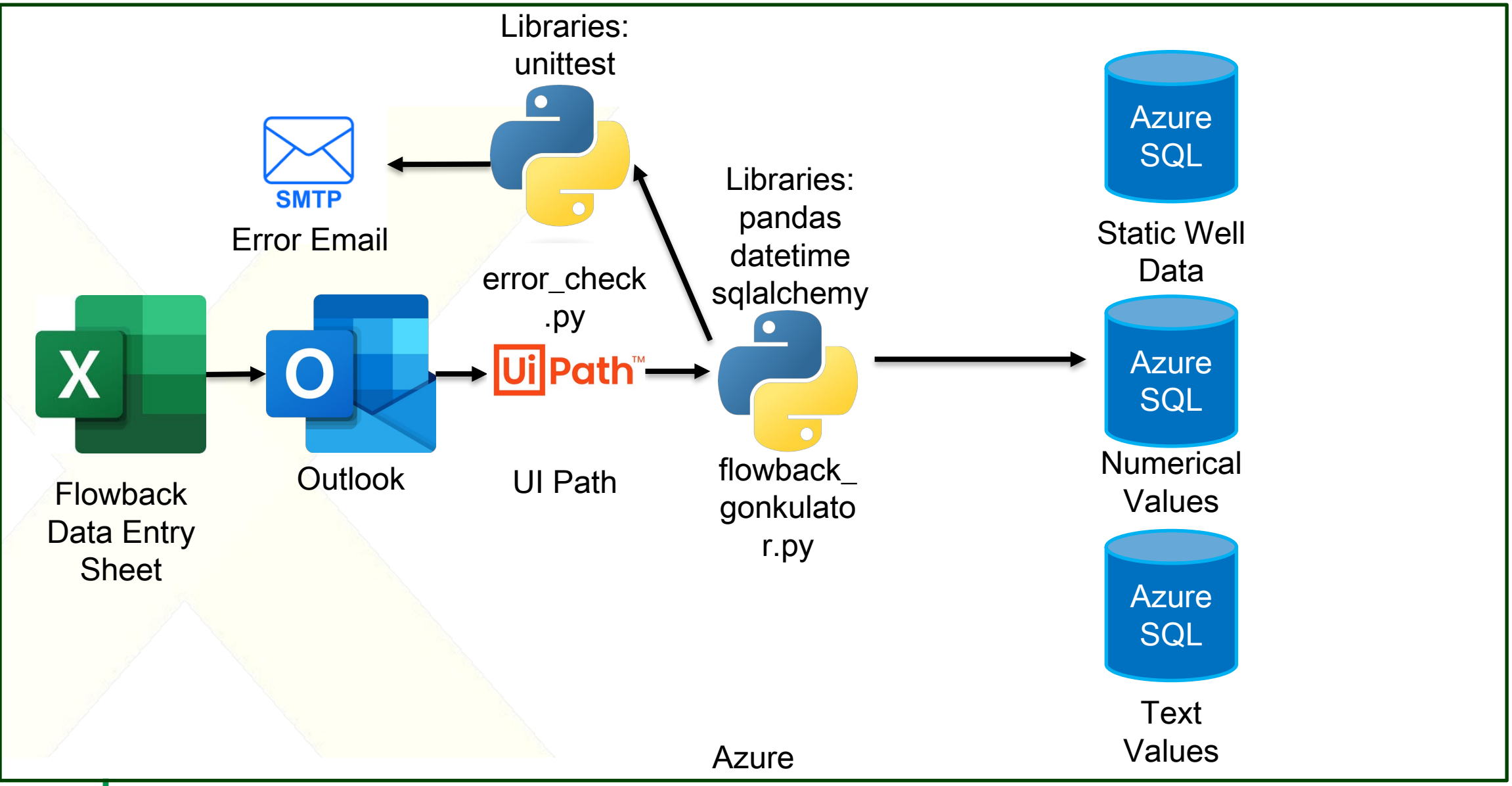




bp

bpx energy

The Sequel



bp

bpx energy

Architecture in Practice

Software Architecture in Energy

- Oil and natural gas exploration presents unique challenges
- Safety and reliability are paramount
- Downtime costs $$$/hour
- Regulatory compliance
- Security against internal/external vectors

- How do we write software that avoids risk?

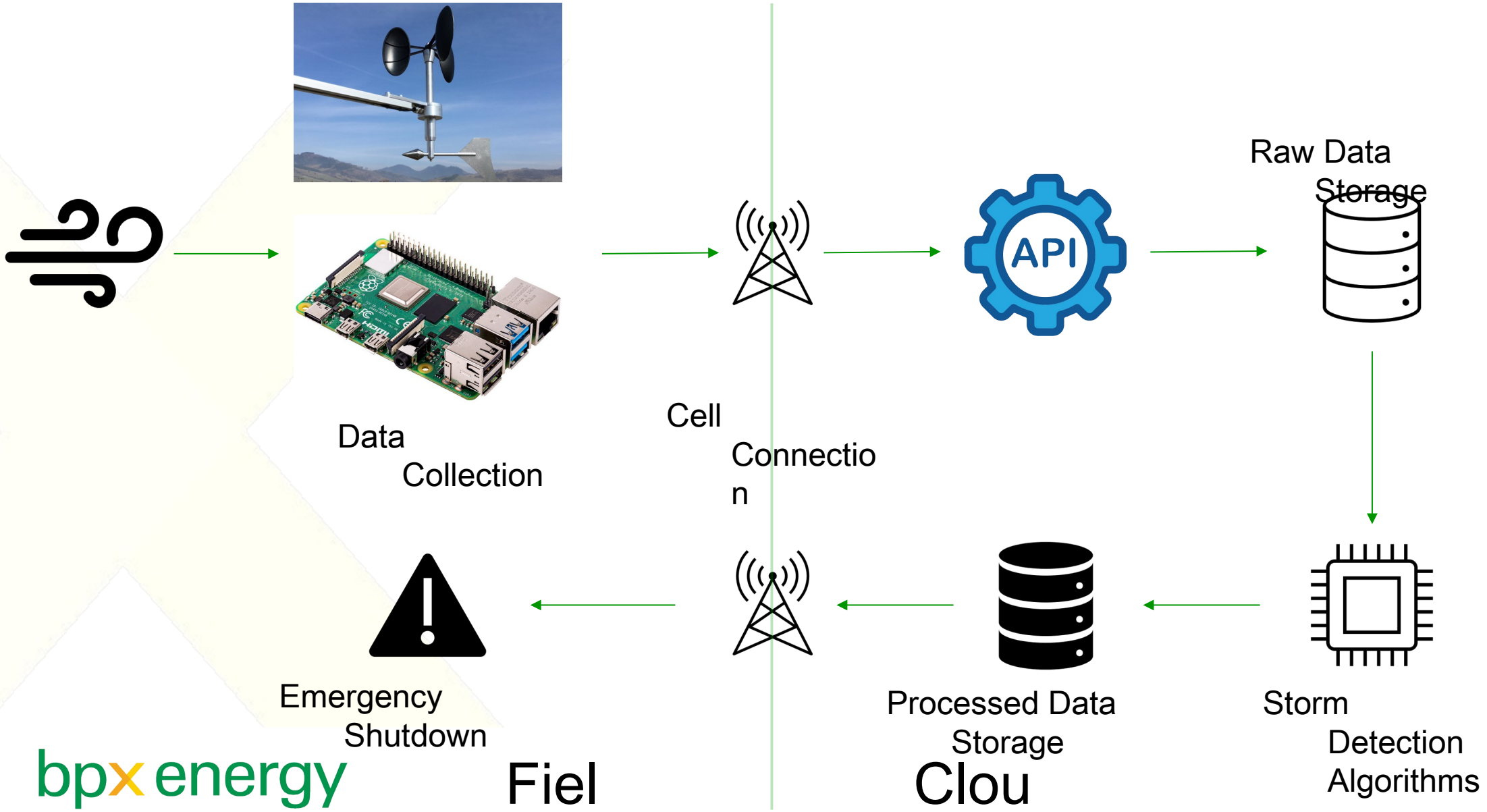
bpx energy

Case Study: Mobile Data Collection & Processing

- Principle: Maintain safe operations
- Objective: Conduct safe shutdowns in the event of severe weather
- Proposal: Deploy an early warning system comprising peripheral sensors to capture wind speeds, humidity readings, and barometric pressure. Avoid unnecessary shutdowns at all cost.

bpx energy

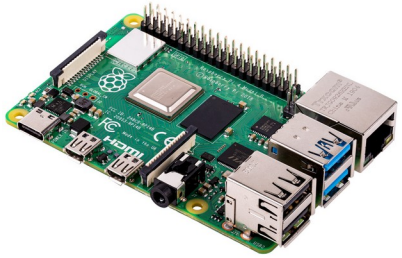
Proposed Data Flow – 30,000 Feet



bpx energy

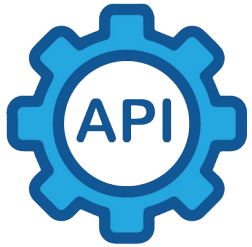
Data Collection/Generation

- Storage (space requirements/format)
- Consistency between device types
- Connection instability
  - Caching
  - Heartbeat
- Security (certificates, disk encryption, etc)
- Updates
  - Firmware vs Higher Level Language updates
  - OS Updates
- NaN != 0

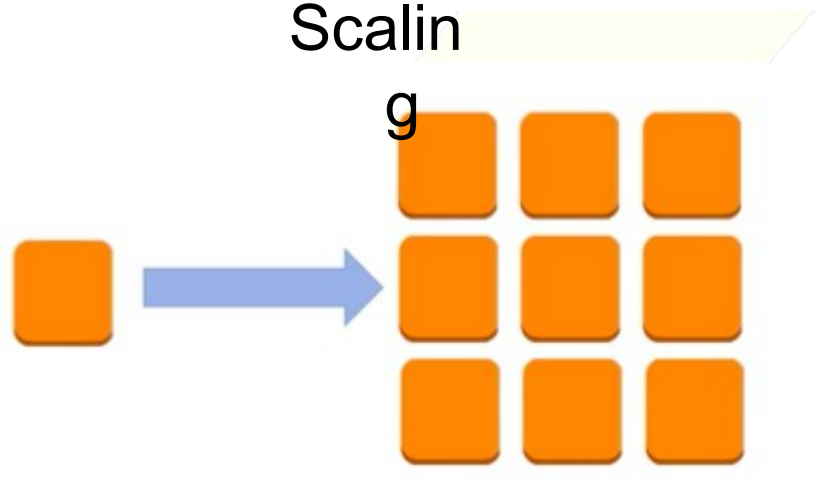


bpx energy

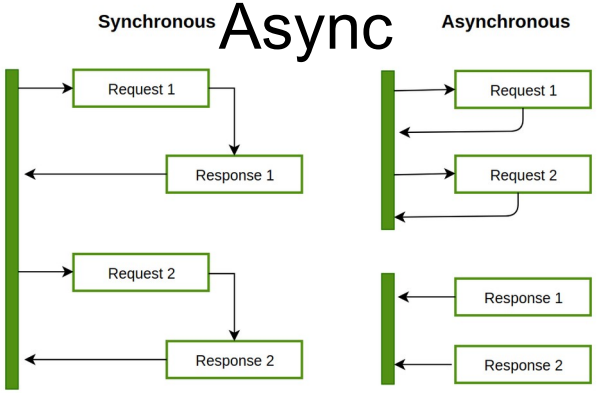
Data Ingestion



Scalin
g



Sync vs
Async



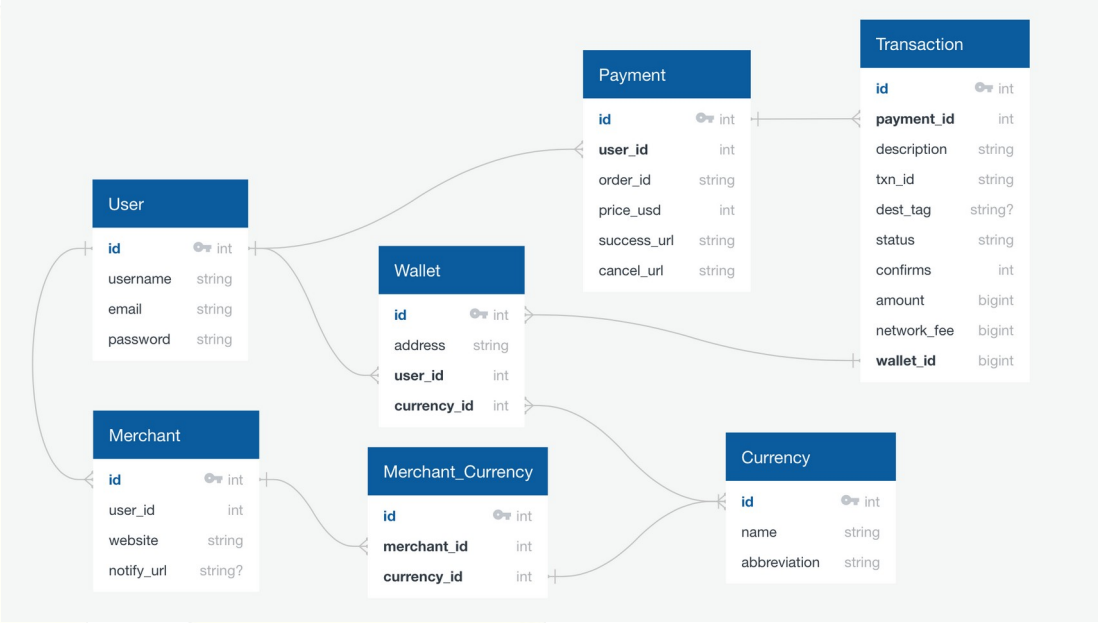
Data
Complexity

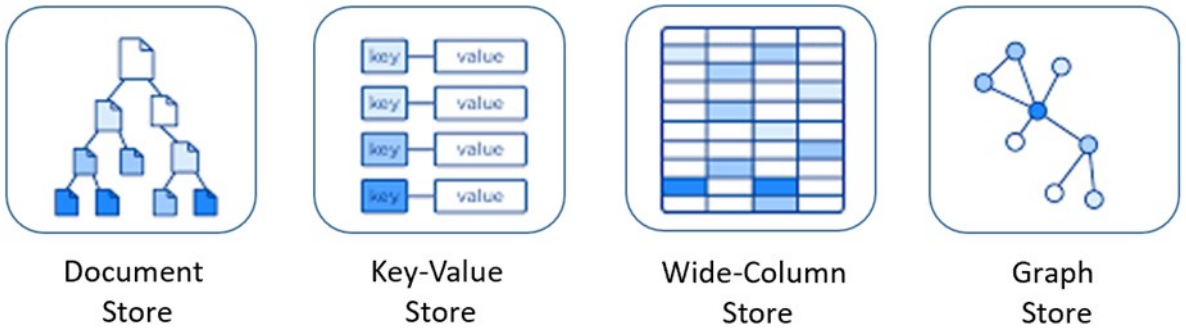
Data
Security

bpx energy

Raw Data Storage

- Storing processed data vs “raw” data





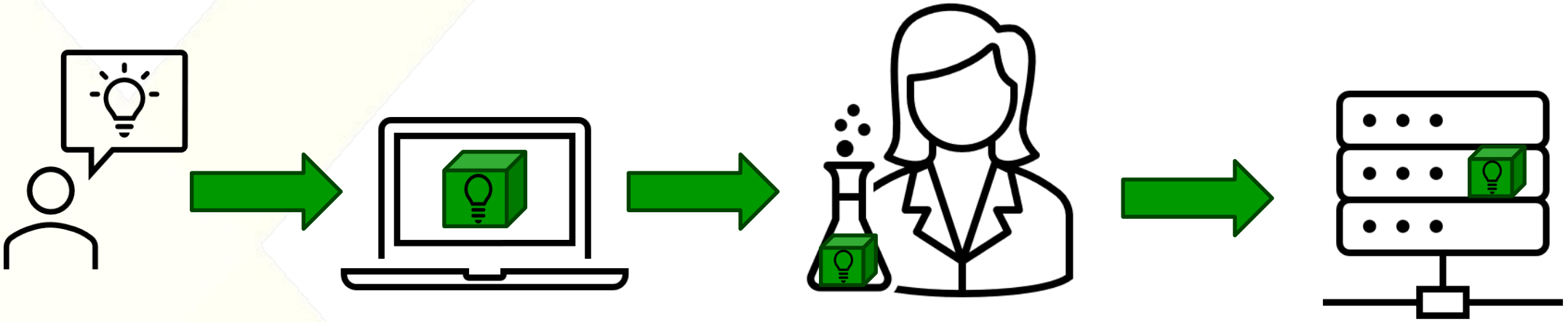
bpx energy

Storm Detection Algorithms

- Leverage 3rd party libraries where possible
- Python and PyPI
  - numpy, scipy, pandas, pydantic, sqlmodel, sketch
- Machine Learning
- Containers
- High availability

bpx energy

Containers



Idea developed in a container

Idea tested in the same container

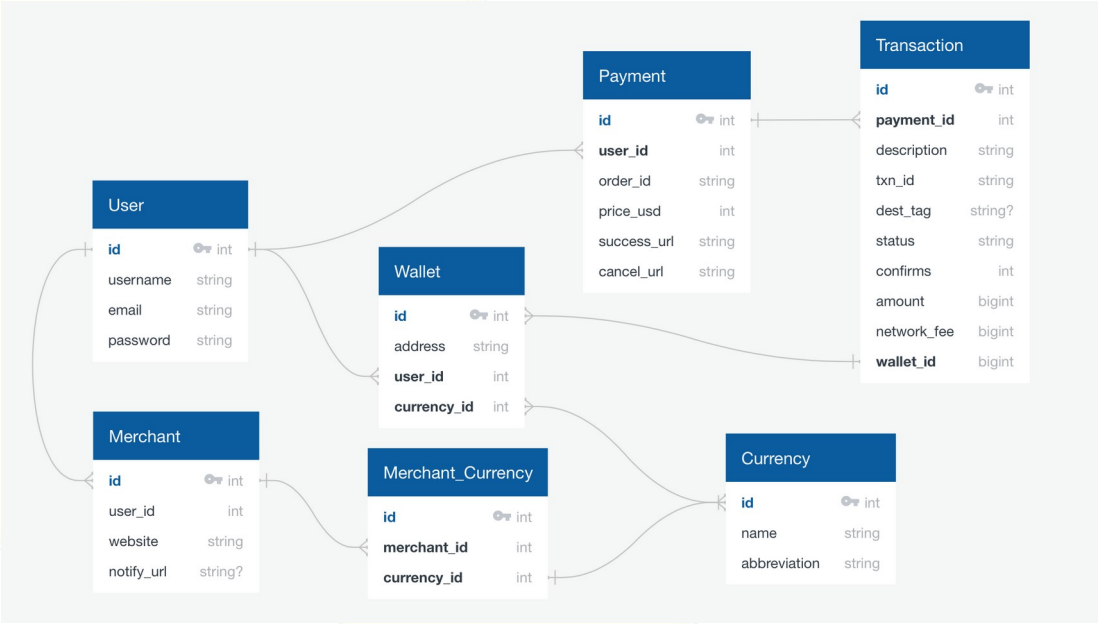
Idea deployed to production in the same container

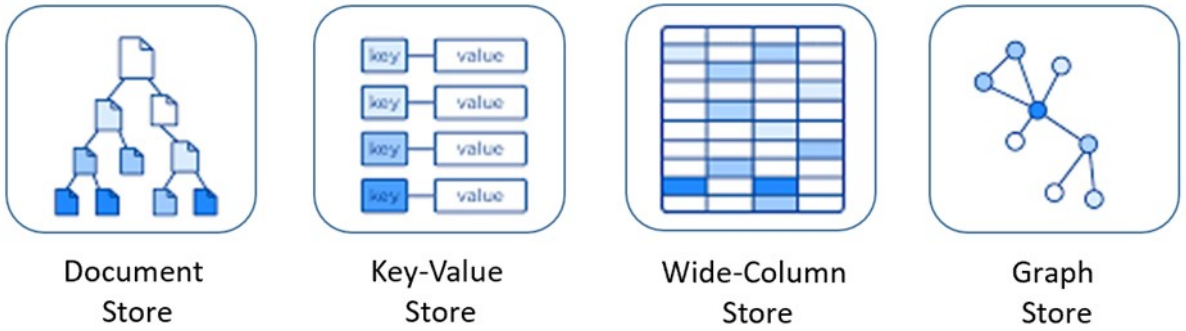
Consistent Environments Promote Consistent Behavior

bpx energy

Processed Data Storage

- What's different about storing processed data?
- High availability – What does that mean for data storage?





bpx energy

Data Layer Abstraction

- Does this work?
  - …sure. ChatGPT strikes again!
- What's wrong with this picture?
- How can we make this better?
- What's the best way to make this future-proof?
- How could we add support for new data stores?

test > bad > humidity.py > insert_humidity_reading

```python
import psycopg2
from datetime import datetime
from math import isnan


def insert_humidity_reading(db_host, db_name, db_user, db_password, sensor_id, humidity_value):

    try:
        # Establish a database connection
        connection = psycopg2.connect(
            host=db_host,
            database=db_name,
            user=db_user,
            password=db_password
        )

        # Create a cursor object
        cursor = connection.cursor()

        # Handle NaN humidity values
        if isnan(humidity_value):
            humidity_value = None

        # Define the data you want to insert
        data_to_insert = {
            "sensor_id": sensor_id,
            "timestamp": datetime.now(),
            "humidity_value": humidity_value
        }

        # SQL query to insert data into the "humidity" table
        insert_query = """
        INSERT INTO humidity (sensor_id, timestamp, humidity_value)
        VALUES (%(sensor_id)s, %(timestamp)s, %(humidity_value)s)
        """

        # Execute the INSERT query with the data
        cursor.execute(insert_query, data_to_insert)

        # Commit the changes to the database
        connection.commit()

        # Close the cursor and the database connection
        cursor.close()
        connection.close()

        print("Record inserted successfully!")

    except (Exception, psycopg2.Error) as error:
        print("Error while inserting data into PostgreSQL:", error)

    finally:
        # Close the database connection (even in case of an exception)
        if connection:
            connection.close()
```

bpx energy

Data Layer Abstraction – Business Logic Separation

test > better > main.py > ...

```python
from datetime import datetime
from math import isnan
from dal import DataAccessLayer


def insert_humidity_reading(dal: DataAccessLayer, sensor_id, humidity_value):

    # Handle NaN humidity values
    if isnan(humidity_value):
        humidity_value = None

    # Define the data you want to insert
    data_to_insert = {
        "sensor_id": sensor_id,
        "timestamp": datetime.now(),
        "humidity_value": humidity_value
    }

    try:
        dal.insert_humidity_reading(data_to_insert)

        print("Record inserted successfully!")

    except Exception as error:
        print("Error while inserting data into data store:", error)

```

test > better > dal > __init__.py > DataAccessLayer

```python
from abc import ABC, abstractmethod

class DataAccessLayer(ABC):
    def __init__(self, *args):
        ...

    @abstractmethod
    def get_connection(self):
        ...

    @abstractmethod
    def insert_humidity_reading(self, data_to_insert: dict):
        ...

```

- We don’t have to know how the data access layer is going to work to use it.
- We can now test business logic separately from connection logic
- Support for multiple data backends

bpx energy

Data Layer Abstraction

- The data access layer doesn't know anything about the business logic
- The implementation of get_connection and insert_humidity_reading can be completely different from other data backends.
- Changes here don't require re-testing the business logic

test > better > dal > postgresql.py > ...

```python
import psycopg2
from dal import DataAccessLayer

class PostgresqlDataAccessLayer(DataAccessLayer):
    def __init__(self, db_host: str, db_name: str, db_user: str, db_password: str):
        self.db_host = db_host
        self.db_name = db_name
        self.db_user = db_user
        self.db_password = db_password

    def get_connection(self):
        return psycopg2.connect(
            host=self.db_host,
            database=self.db_name,
            user=self.db_user,
            password=self.db_password
        )


    def insert_humidity_reading(self, data_to_insert: dict):
        try:
            # Create a cursor object
            with self.get_connection() as connection:
                with connection.cursor() as cursor:

                    # SQL query to insert data into the "humidity" table
                    insert_query = """
                    INSERT INTO humidity (sensor_id, timestamp, humidity_value)
                    VALUES (%(sensor_id)s, %(timestamp)s, %(humidity_value)s)
                    """

                    # Execute the INSERT query with the data
                    cursor.execute(insert_query, data_to_insert)

                    # Commit the changes to the database
                    connection.commit()

        except (Exception, psycopg2.Error) as error:
            print("Error while inserting data into PostgreSQL:", error)
            raise
```

bp

bpx energy

Data Layer Abstraction – Future Proofing

- The data access layer still doesn't know anything about the business logic
- Changes here don't affect the business logic OR any other data access layers

test > better > dal > kafka.py > KafkaDataAccessLayer

```python
from confluent_kafka import Producer
from dal import DataAccessLayer

class KafkaDataAccessLayer(DataAccessLayer):
    def __init__(self, kafka_broker: str = "localhost:9092", kafka_topic: str = "humidity_readings"):
        # Define Kafka broker(s) and topic
        self.kafka_broker = kafka_broker
        self.kafka_topic = kafka_topic

    def get_connection(self):
        # Kafka producer configuration
        producer_config = {
            'bootstrap.servers': self.kafka_broker
        }

        # Create a Kafka producer instance
        return Producer(producer_config)

    def insert_humidity_reading(self, data_to_insert: dict):
        try:
            # Produce the humidity reading as a message to the Kafka topic
            with self.get_connection() as producer:
                producer.produce(self.kafka_topic, key=str(data_to_insert["sensor_id"]), value=str(data_to_insert))
                # Wait for any outstanding messages to be delivered and delivery reports to be received
                producer.flush()

        except Exception as e:
            print(f"Error pushing data to Kafka: {str(e)}")
            raise
```

bpx energy

Safety

The value of what you do is larger than the code you write

bpx energy