

Bluesky Tweets Streaming Pipeline – WSL

Setup (Commands Only)

1. Update System & Install Prerequisites

```
sudo apt update && sudo apt upgrade -y
```

```
sudo apt install -y openjdk-11-jdk scala wget curl tar unzip python3-pip git software-properties-common
```

2. Install Kafka

```
# Download Kafka
```

```
wget https://downloads.apache.org/kafka/3.4.1/kafka_2.13-3.4.1.tgz
```

```
tar -xvzf kafka_2.13-3.4.1.tgz
```

```
cd kafka_2.13-3.4.1
```

```
# Start Zookeeper in background
```

```
bin/zookeeper-server-start.sh -daemon config/zookeeper.properties
```

```
sleep 5
```

```
# Start Kafka broker in background
```

```
bin/kafka-server-start.sh -daemon config/server.properties
```

```
sleep 5
```

```
# Create topic for Bluesky tweets
```

```
bin/kafka-topics.sh --create --topic bluesky-tweets --bootstrap-server localhost:9092 --partitions 1 --replication-factor 1
```

```
# List topics
```

```
bin/kafka-topics.sh --list --bootstrap-server localhost:9092
```

3. Install InfluxDB 2.x

```
# Download InfluxDB  
  
wget https://dl.influxdata.com/influxdb/releases/influxdb2-2.7.0-linux-amd64.tar.gz  
  
tar xvfz influxdb2-2.7.0-linux-amd64.tar.gz  
  
cd influxdb2-2.7.0-linux-amd64  
  
  
# Start InfluxDB  
  
../influxd &  
  
  
# Setup InfluxDB (interactive)  
  
../influx setup  
  
# Enter: Org, Bucket, Username, Password, Token
```

4. Install Apache Spark

```
# Download Spark  
  
wget https://dlcdn.apache.org/spark/spark-3.4.1/spark-3.4.1-bin-hadoop3.tgz  
  
tar xvf spark-3.4.1-bin-hadoop3.tgz  
  
cd spark-3.4.1-bin-hadoop3  
  
  
# Spark-submit example (Python)  
  
.bin/spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.4.1  
path/to/bluesky_stream.py
```

```
# Spark-submit example (Java)
```

```
.bin/spark-submit --class YourMainClass --master local[*] target/your-spark-app.jar
```

5. Install Grafana

```
# Add Grafana repository
```

```
sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main" -y
sudo apt update
sudo apt install grafana -y

# Start Grafana service
sudo systemctl start grafana-server
sudo systemctl enable grafana-server

# Check Grafana status
sudo systemctl status grafana-server | head -n 20

# Access Grafana: http://localhost:3000 (default login: admin/admin)
```

6. Kafka Commands

```
# Produce messages
bin/kafka-console-producer.sh --topic bluesky-tweets --bootstrap-server
localhost:9092

# Consume messages
bin/kafka-console-consumer.sh --topic bluesky-tweets --from-beginning --bootstrap-
server localhost:9092

# List topics
bin/kafka-topics.sh --list --bootstrap-server localhost:9092
```

7. InfluxDB Commands

```
# List buckets
./influx bucket list
```

```
# Query last 1 hour data  
./influx query 'from(bucket:"<bucket>") |> range(start: -1h)'
```

8. Spark Streaming Commands

```
# Python streaming job  
./bin/spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.4.1  
path/to/bluesky_stream.py  
  
# Java streaming job  
./bin/spark-submit --class YourMainClass --master local[*] target/your-spark-app.jar
```

9. Grafana – Data Source Setup

```
# Add InfluxDB 2.x as Data Source in Grafana
```

URL: <http://localhost:8086>

Token: <your-token>

Org: <your-org>

Default Bucket: <your-bucket>

Query Language: Flux

10. Full Verification Commands

```
# Kafka: check messages
```

```
bin/kafka-console-consumer.sh --topic bluesky-tweets --from-beginning --bootstrap-  
server localhost:9092
```

```
# InfluxDB: check buckets & query
```

```
./influx bucket list
```

```
./influx query 'from(bucket:"<bucket>") |> range(start: -1h)'
```

```
# Spark: submit jobs
```

```
./bin/spark-submit --master local[*] path/to/bluesky_stream.py
```

```
# Grafana: open browser http://localhost:3000
```