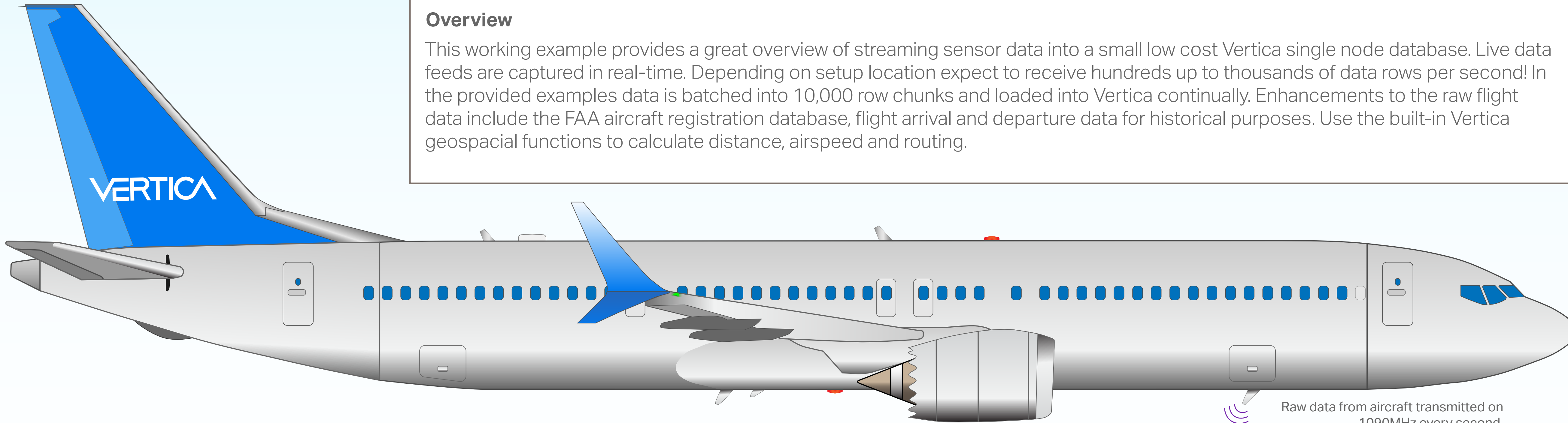# VERTICA

# Visual Guide–IoT Flight Demo

## Overview

This working example provides a great overview of streaming sensor data into a small low cost Vertica single node database. Live data feeds are captured in real-time. Depending on setup location expect to receive hundreds up to thousands of data rows per second! In the provided examples data is batched into 10,000 row chunks and loaded into Vertica continually. Enhancements to the raw flight data include the FAA aircraft registration database, flight arrival and departure data for historical purposes. Use the built-in Vertica geospacial functions to calculate distance, airspeed and routing.

Raw data from aircraft transmitted on 1090MHz every second.

## Kafka Integration

Kafka is designed for a streaming use case (high volumes of data with low latency). Using the Kafka integration feature, data enters Kafka as a message. A feed of messages in a common category come together to form topics. Kafka divides the topics up into partitions that it can be fed in parallel to Vertica target tables and analyzed by you.

Job scheduler

COPY

UDL

Producer: data stream

Kafka cluster

Consumer: Vertica

## Aircraft Registration Data

Downloadable from the FAA website: www.faa.gov
Search for "aircraft regirstration database"

**Dumpfix REGEX for Data Strings**
`[0-9]{4}\/[0-9]{2}\/[0-9]{2}\,`
**Before**
`2017/04/24,22:03:33.105`
**After**
`2017-04-24 22:03:33.105`

## Bill of Materials

| | |
|---|---|
| (1) 1090MHz ADS-B Antenna - 66cm / 26in | $39.00 |
| (2) RF pigtail cable SMA male to N male RG58 5M | $18.00 |
| (3) FlightAware Pro Stick ADS-B USB Receiver | $18.00 |
| (4) USB 3.0 Extension Cable -1 Foot | $5.59 |
| (5) Raspberry Pi 3 Kit, with case and power supply | $49.00 |
| (6) SanDisk 32GB microSDHC UHS-3 Card | $18.75 |
| **Total** | **$148.34** |

## ADS-B RAW Data Stream

```
*8DAD18CE58C3873D9F8F8420BCC1;
*02E19716E1A61C;
*8DACD14358AF973CB35031FDA24C;
*8DAD18CEF83300020049B8A769AA;
*8DA7C3C958B97334CE42920E8171;
*5DAD4171106B48;
*5DABD69D0A7626;
*5DA7C3C9863D24;
*5DA7133FE6FDF1;
*5DA7C3C9863D24;
*20001717126DEA;
*02E195B9052F1B;
*8DAC45AC58A586A30CEE2BF81664;
*02E19716E1A61C;
*00E616906B27E0;
*8DA7133F990A2B91180887DF2374;
*00E616906B27E0;
*8DA7C3C9990CC2A5E01407B151F7;
*8DAC45ACEA3E9866F33C082A7F0D;
*8DAC741960C38665FEEF4D075F01;
*5DAC45ACDE5024;
*02461998CE5C5D;
*02E19498F583CD;
*8DA7133F58BF03D1851F0FB28043;
*5DACD143054A5E;
*8DA7133FEA485865213E88486130;
```

## Vertica SQL

```sql
create table dump1090 (
    record_type         char(3),
    record_type_number  integer,
    sg_session_id       integer,
    sg_aircraft_id      integer,
    hex_ident           varchar(10),
    sg_flight_id        integer,
    msg_gen_ts          timestamp,
    msg_log_ts          timestamp,
    call_sign           varchar(12),
    altitude            integer,
    ground_speed        integer,
    track               integer,
    latitude            decimal(8,5),
    longitude           decimal(8,5),
    vertical_rate       integer,
    squawk              varchar(12),
    alert               integer,
    emergency           integer,
    spi                 integer,
    is_on_ground        integer
);
```
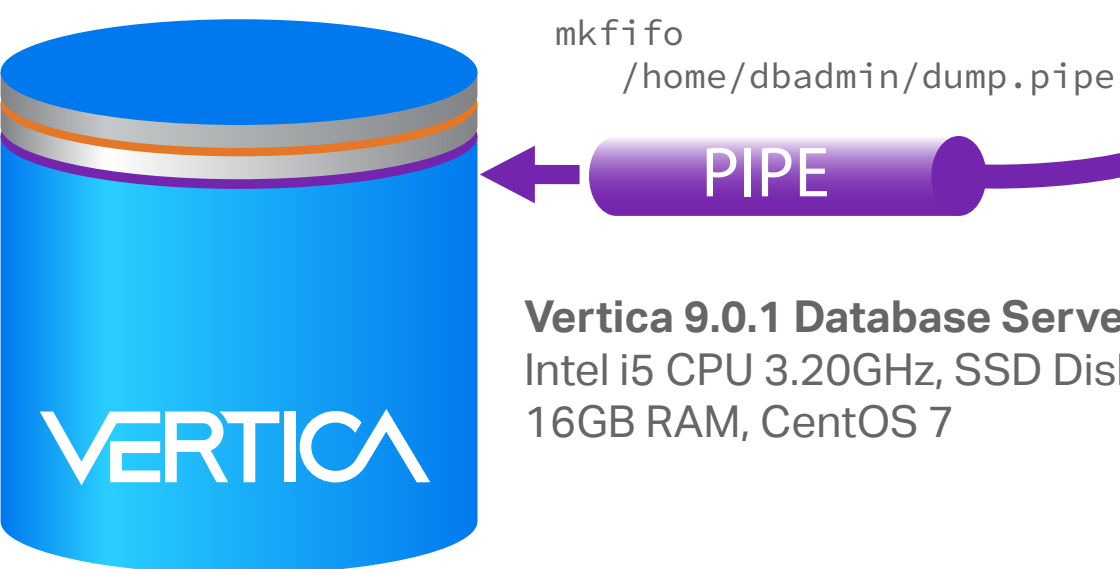
## What is ADS-B?

Automatic Dependent Surveillance-Broadcast
ADS-B is different from radar and does not depend on centralized controllers watching radar scopes. Instead, aircraft self-report GPS positions in a networked environment allowing pilots to see the entire air traffic picture around them in real-time.
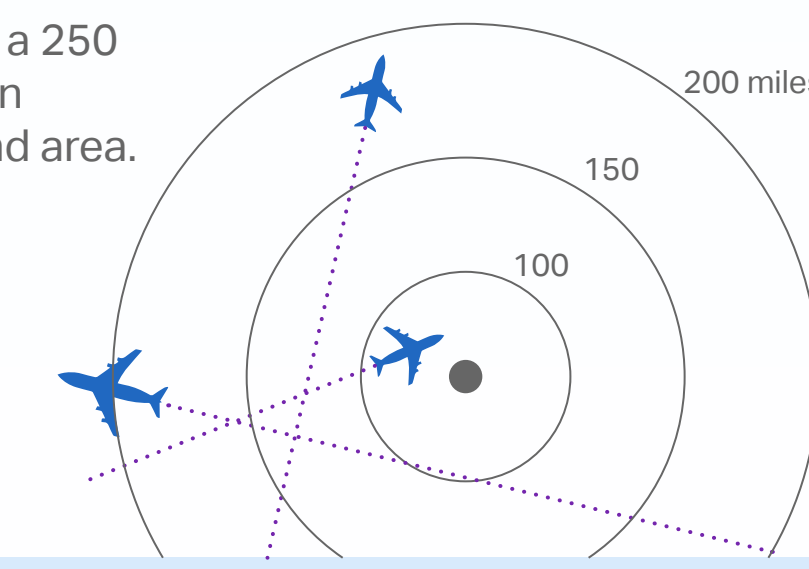
Receive data within a 250 mile radius based on antenna location and area.

200 miles
150
100

26"

1090MHz antenna

① 

Download dump1090 from github:
`https://github.com/MalcolmRobb/dump1090`

## IoT Device (edge)

② ③ ④ ⑤ ⑥

USB Receiver RTLSDR device

**Raspberry Pi**

Raspberry pi Model 3 • CPU Broadcom BCM2837 • 1.2 GHz Quad-Core ARM Cortex A53 • 1 GB LPDDR2-900 SDRAM

Example decoded message
```
MSG,3,1,1,A260BC,1,2017/04
/24,22:03:33.105,2017/04/2
4,22:03:33.121,,36000,,,34
.65033,-92.28835,,,,,,0
```

Dump1090 is a simple Mode S decoder for RTLSDR devices

Data batch sent from raspberry pi into the pipe Vertica COPY statement loads each batch

Google Cloud Platform
Microsoft Azure
amazon web services

## Edge Analytics

mkfifo
/home/dbadmin/dump.pipe

PIPE

**Vertica 9.0.1 Database Server**
Intel i5 CPU 3.20GHz, SSD Disk, 16GB RAM, CentOS 7

Collect data from multiple Vertica edge devices into a central location cloud or private data center to provide broad historical coverage, predictave analytics.

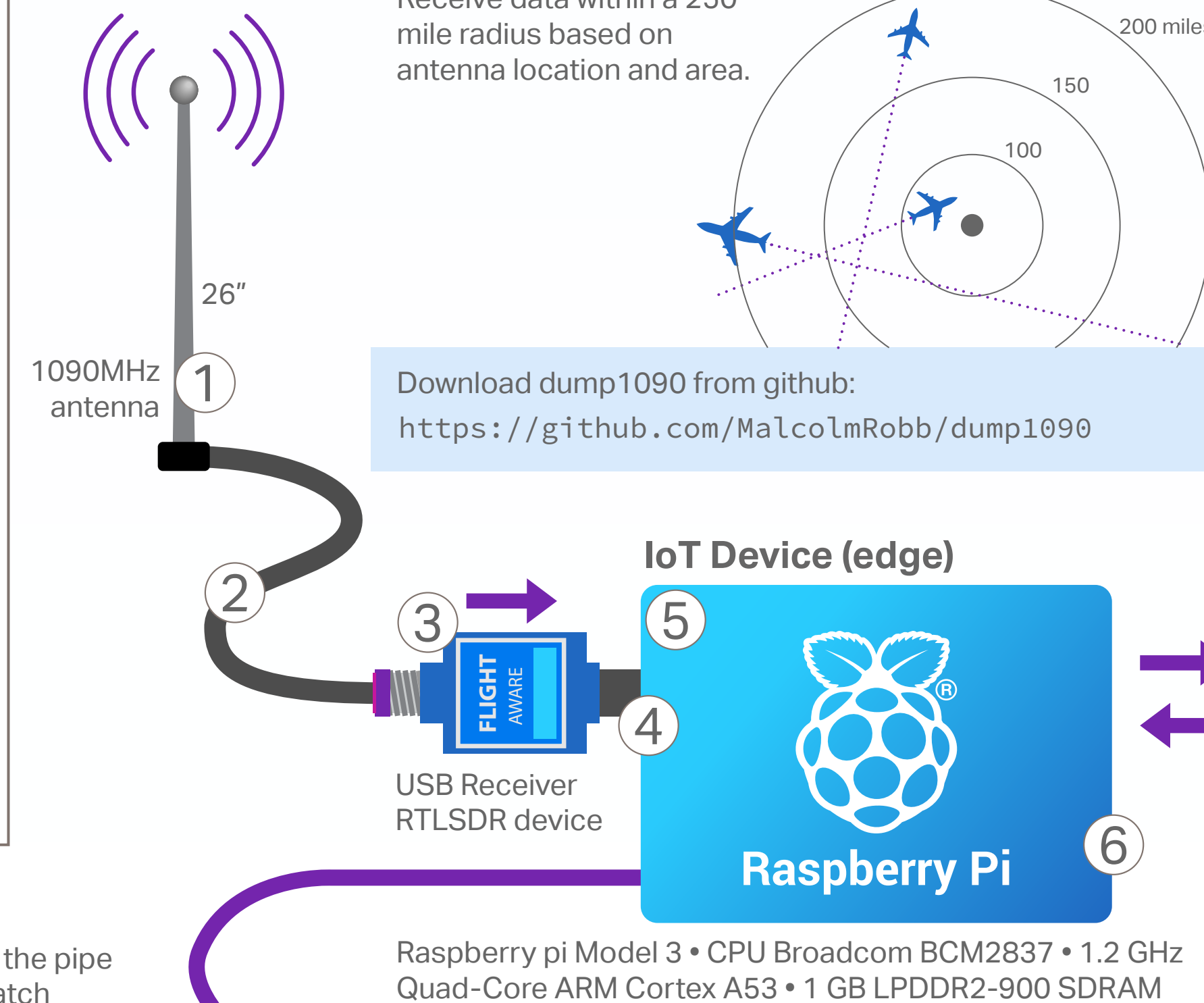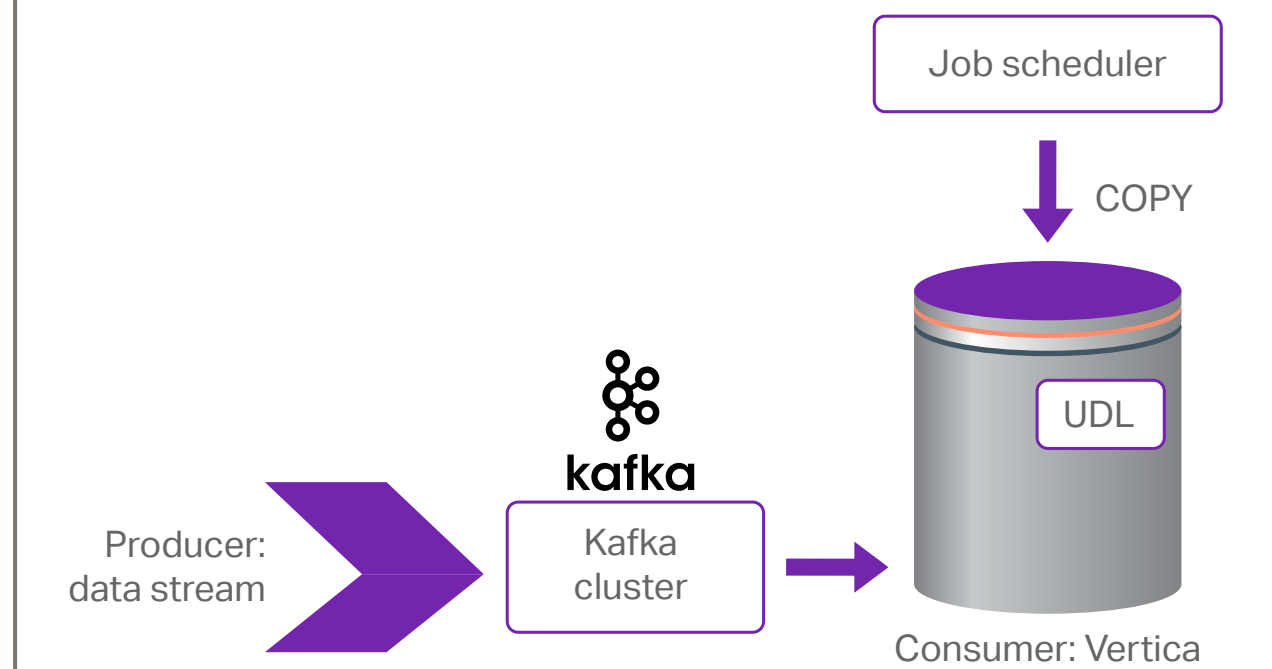## Commands to collect dump1090 data and load into Vertica

netcat command retreives a 10,000 row batch
```
nc localhost 30003 | head -10000 > batch.txt
```
From raspberry pi initiate vsql COPY with named pipe
```
ssh dbadmin@192.168.1.8 "/opt/vertica/bin/vsql -U dbadmin -c \"COPY DUMP1090 FROM LOCAL '/home/dbadmin/dump.pipe' DELIMITER ',' NULL '' DIRECT ;\"" >> load.log &
```
Cat newest batch of rows pipe to ssh command to Vertica server, dumpfix reformats date into standard TIMESTAMP format for Vertica COPY.
```
cat batch.txt | ssh dbadmin@192.168.1.8 "cat - | /home/dbadmin/dumpfix > \
    /home/dbadmin/dump.pipe" >> load.log
```

For more information **www.vertica.com/iot**