AMUSE DATA PLATFORM

<Name>

<Position>

Affine Analytics

Contents

[Introduction 3](#_Toc74205735)

[AMUSE 3](#_Toc74205736)

[OBJECTIVE: 3](#_Toc74205737)

[ARCHITECTURE 3](#_Toc74205738)

[VISUALISATION 3](#_Toc74205739)

[DESCRIPTION 4](#_Toc74205740)

[Data Sources Involved 4](#_Toc74205741)

[Using Fivetran to address ETL: 4](#_Toc74205742)

[Load and Transformation Part of the ETL 5](#_Toc74205743)

[DATA SOURCES 5](#_Toc74205744)

[Employee Data Source Systems 6](#_Toc74205745)

[**Greenleaf HR**: A HR and Payroll system 6](#_Toc74205746)

[**Humanity**:An hourly employee scheduling 6](#_Toc74205747)

[Inventory Data Source Systems: 6](#_Toc74205748)

[**Magento**: A content management system for the e-commerce 6](#_Toc74205749)

[**Treez**: Enterprise software platform for Cannabis products 6](#_Toc74205750)

[**Google Sheets**: 6](#_Toc74205751)

[Fleet Data Source Systems: 7](#_Toc74205752)

[**OnFleet**: A logistics management system 7](#_Toc74205753)

[**Samsara**: A fleet management platform 7](#_Toc74205754)

[Helpdesk Data Source: 7](#_Toc74205755)

[**Zendesk** (Zendesk Support): A customer service system. 7](#_Toc74205756)

[Email or SMS Marketing platform data sources: 7](#_Toc74205757)

[**Klaviyo**: An email and SMS marketing platform 7](#_Toc74205758)

[**Twilio**: A cloud communications platform to enable sending and receiving text messages. 7](#_Toc74205759)

[TECHNOLOGIES USED 8](#_Toc74205760)

[AWS Lambda 8](#_Toc74205761)

[FIVETRAN 8](#_Toc74205762)

[SNOWFLAKE 8](#_Toc74205763)

[DBT 8](#_Toc74205764)

[Why DBT? 8](#_Toc74205765)

[Final Tables and their Descriptions: 10](#_Toc74205766)

[ Customer Master Table: CUSTOMERS\_MASTER 10](#_Toc74205767)

[ Customers Details Table: CUSTOMERS\_SOURCE\_INFO 11](#_Toc74205768)

[ Delivery Tasks Table: DELIVERY\_TASKS\_MASTER 11](#_Toc74205769)

[ Transactions Items Table: TRANSACTION\_ITEMS\_MASTER 11](#_Toc74205770)

[ Transactions table: TRANSACTIONS\_MASTER 12](#_Toc74205771)

[ Worker Analytics table: ONFLEET\_WORKER\_ANALYTICS 12](#_Toc74205772)

[ Worker Analytics table: DAILY\_METRICS 13](#_Toc74205773)

[Data pipeline 13](#_Toc74205774)

[SETTING UP CONNECTORS 13](#_Toc74205775)

[STORED PROCEDURES 13](#_Toc74205776)

[TRANSFORMATIONS USING DBT 14](#_Toc74205777)

[Need to use DBT: 14](#_Toc74205778)

[MODELS AND TRANSFORMATION DONE: 14](#_Toc74205779)

[IMPORTANT FEATURES INCORPORATED INTO DBT: 16](#_Toc74205780)

# Introduction

## AMUSE

Amuse is an e-commerce cannabis delivery business that is reinventing the way consumers order and consume cannabis.

# OBJECTIVE:

•The company wants to create a data platform for their business which would help them analyse their business at various levels of processes like transaction, deliveries, customers, products etc.

•Affine provides data engineering solutions for their data platform and work on the structuring, managing and automating the various data acquisition and transformation processes for their platform.

# ARCHITECTURE

## VISUALISATION

### DESCRIPTION

* Data are extracted from the sources mentioned below using Fivetran.
* Connections to the database are made through Fivetran connectors or through API call from lambda then connecting lambda to Fivetran.
* The extracted data is loaded into snowflake into the “AMUSE INGESTION” database.
* A third-party tool “DBT” is connected to snowflake to generate queries to do the transformations for the final tables.
* Fivetran runs the DBT models after certain intervals.
* Transformations are done only to the recent data and the transformed data is inserted/merged to the final table.

### Data Sources Involved

1. Treez
2. Magento
3. Onfleet
4. Domo

### Using Fivetran to address ETL:

Extraction of the raw data from the data sources are done using Fivetran **Connectors**.

### Load and Transformation Part of the ETL

Diagram

Description automatically generated

Fig: The Flow chart represents the flow of the processed data to the final tables from ingestion.

All the ETLs follow a similar approach on the plan as shown above.

The description to the data sources, ingestion tables generated by Fivetran, and final tables are given below.

# DATA SOURCES

Following is the list of source systems we are aware of:

## Employee Data Source Systems

### **Greenleaf HR**: A HR and Payroll system

Data Ingestion: No data ingestion from this system.

Process Desc.: Generally used to create Employee profiles (both hourly and corporate employees). To generate employee IDs which are to be used in **Humanity**. Also used to issue pay checks for the employees using the data from **Humanity**.

### **Humanity**:An hourly employee scheduling

and time tracking system

Data Ingestion: AWS Lambda Ingestion (Using Fivetran).

Process Desc.: Separate system used to track, and schedule hourly employees based on their profiles in **Greenleaf HR** manually linked via. EmployeeID from **Greenleaf HR**. Also used for generating CSV data outputs for **Greenleaf HR** to help generate paychecks to the employees.

## Inventory Data Source Systems:

### **Magento**: A content management system for the e-commerce

Data Ingestion: Magento support connection (through Fivetran)

Process Desc: Amuse manages the products in their catalog on Magento. Each product entry has descriptions of the price, features of the product (ingredients, THC content etc). These are manually created by the employees.

### **Treez**: Enterprise software platform for Cannabis products

Data Ingestion: AWS Lambda (through Fivetran)

Process Desc: Treez manages orders and creates corresponding tasks for riders in OnFleet. Also interfaces with Metrc to update inventory to reconcile the data as per the orders and deliveries. It is the source of truth for inventory.

### **Google Sheets**:

Data Ingestion: Google Sheets support connection (through Fivetran)

Process Desc: The delivery areas and their respective delivery cut-off timings are imported through Google Sheets (manually).

## Fleet Data Source Systems:

### **OnFleet**: A logistics management system

Data Ingestion: AWS Lamda Ingestion (Using Fivetran).

Process Desc.: Creates driver tasks. Dispatcher assigns drivers the task. Webhooks used to update the Treez Tickets with delivery progress status.

### **Samsara**: A fleet management platform

(“Not used in a meaningful way”)

Data Ingestion: No data ingestion.

Process Desc.: Used to track the activity status of drivers assigned with the task of delivery to comply with the Cannabis delivery laws. No instance of this service mentioned in Fivetran.

## Helpdesk Data Source:

### **Zendesk** (Zendesk Support): A customer service system.

Data Ingestion: Zendesk Support connection used to ingest.

data via. Fivetran

Process Desc.: Creates Zendesk tickets for customers with ID verification failures at check outs. Also reviewing negative customer ratings for orders (rated 3 or less on the scale of 5).

## Email or SMS Marketing platform data sources:

### **Klaviyo**: An email and SMS marketing platform

Data Ingestion: Klaviyo connection in Fivetran

Process Desc.: Used to manage the email and SMS templates for various campaigns. It uses Twilio to send the SMS at the backend which is a major element to it. Sendgrid to send off emails at the backend.

### **Twilio**: A cloud communications platform to enable sending and receiving text messages.

Data Ingestion: AWS Lambda ingestion via. Fivetran

# TECHNOLOGIES USED

## AWS Lambda

[Lambda](https://docs.aws.amazon.com/lambda/latest/dg/welcome.html) is a serverless compute service that lets you run code without provisioning or managing servers or managing runtimes. With Lambda, you can run code for virtually any type of application or backend service, with zero administration. You can set up your code to automatically trigger from any AWS service or call it directly using web/mobile.

LAMBDA FUNCTIONS:

<getting data from api>

## FIVETRAN

[Fivetran](https://fivetran.com/docs/getting-started) allows for efficient collection of business processes and customer data from related applications, websites, and servers. The data collected is then transferred to other tools for analytics, marketing, and data warehousing purposes.

DESTINATIONS: Warehouses, in simpler terms, are connected to Fivetran where the connectors sync data into it from the sources.Here, our destination is Snowflake for both the ingested tables and tables from dbt transformations.

TRANSFORMATIONS: It allows data teams to organize, analyze, and visualize data without sacrificing data quality or analytical flexibility. These run-in destination after data is loaded, so the raw data is always available alongside your transformed data.

CONNECTORS: A Fivetran connector is a data pipeline that moves data from your source to your destination. Here, we have connectors from AWS Lambda, Redshift, S3, Google Sheets into fivetran for ingested data.

## SNOWFLAKE

[Snowflake](https://docs.snowflake.com/en/) offers a cloud-based data storage and analytics service, generally termed "data warehouse-as-a-service". There's no hardware or software to select, install, configure, or manage, so it’s ideal for organizations that don’t want to dedicate resources for setup, maintenance, and support of in-house servers.

## DBT

1. [DBT](https://docs.getdbt.com/docs/introduction) is a Tool to transform data in their warehouses using only select statements.
2. DBT turns these select statements into tables and views.
3. A model is created with single or multiple such select queries and relationship is established between these models in the form of DAG.

### Why DBT?

* DBT handles boilerplate code to materialize queries as relations.
* DBT determines the order of model execution.
* Can generate documentation for the project.

### Models

### Macros and Templates

### Target

INGESTION DATABASE IN SNOWFLAKE:

The ingestion database (AMUSE\_DEV\_INGESTION) has tables coming from Fivetran connectors to ingest raw data from given data sources.

|  |  |
| --- | --- |
| **TABLES IN INGESTION** | **USAGE** |
| MAGENTO\_PROD\_PROD.SALES\_ORDER | For orders placed in Magento on Amuse using web application. Used to get transactions /tickets to be used in:   * customers\_master * transactions\_master * transactions\_items\_master |
| MAGENTO\_PROD\_PROD.CUSTOMERS\_ENTITY | For customers registered on Magento for Amuse. Used to get customer details to be used in:   * customers\_master * transactions\_master |
| INGESTION\_TREEZ\_LAX1\_CUSTOMERS.CUSTOMERS, INGESTION\_TREEZ\_SFO1\_CUSTOMERS.CUSTOMERS,  INGESTION\_TREEZ\_SJC1\_CUSTOMERS.CUSTOMERS | For customers coming through Treez for Amuse. Used to get customer details to be used in:   * customers\_master * transactions\_master |
| INGESTION\_TREEZ\_LAX1\_TICKETS.TICKETS,  INGESTION\_TREEZ\_SFO1\_TICKETS.TICKETS,  INGESTION\_TREEZ\_SJC1\_TICKETS.TICKETS | All the orders placed. to get transactions /tickets to be used in:   * customers\_master * transactions\_master * Transactions\_items\_master |
| INGESTION\_TREEZ\_LAX1\_PRODUCTS. PRODUCTS,  INGESTION\_TREEZ\_SFO1\_ PRODUCTS. PRODUCTS,  INGESTION\_TREEZ\_SJC1\_ PRODUCTS. PRODUCTS | To get product details in Treez depots to be used in:   * Transactions\_items\_master |
| INGESTION.INGESTION\_TREEZ\_INVENTORY | Analytical Overview of inventory, used for product\_sku and state tracking id, used by:   * transaction\_items\_master |
| INGESTION.INGESTION\_TREEZ\_INVOICES | Analytical Overview of invoices,For cost per unit, used by:   * transaction\_items\_master |
| INGESTION\_ONFLEET\_WORKER\_ANALYTICS.WORKER\_ANALYTICS | For workers information on Onfleet. Used by:   * worker\_analytics |
| INGESTION\_ONFLEET.TASKS | For tasks created on Onfleet. Used by:   * delivery\_tasks\_master * transactions\_master |
| INGESTION\_ONFLEET.HUBS | For hubs information on Onfleet. Used by:   * delivery\_tasks\_master |
| INGESTION\_ONFLEET.WORKERS | For workers’ information on Onfleet. Used by:   * delivery\_tasks\_master |

The list of tables ingested through different sources are as follows:

**Lambda:**

* INGESTION\_TREEZ\_LAX1\_CUSTOMERS.CUSTOMERS
* INGESTION\_TREEZ\_SFO1\_CUSTOMERS.CUSTOMERS
* INGESTION\_TREEZ\_SJC1\_CUSTOMERS.CUSTOMERS
* INGESTION\_TREEZ\_LAX1\_TICKETS.TICKETS
* INGESTION\_TREEZ\_SFO1\_TICKETS.TICKETS
* INGESTION\_TREEZ\_SJC1\_TICKETS.TICKETS
* INGESTION\_TREEZ\_LAX1\_PRODUCTS. PRODUCTS
* INGESTION\_TREEZ\_SFO1\_ PRODUCTS. PRODUCTS
* INGESTION\_TREEZ\_SJC1\_ PRODUCTS. PRODUCTS
* INGESTION\_ONFLEET\_WORKER\_ANALYTICS.WORKER\_ANALYTICS

**S3:**

* INGESTION.INGESTION\_TREEZ\_INVENTORY
* INGESTION.INGESTION\_TREEZ\_INVOICES

**Redshift:**

* MAGENTO\_PROD\_PROD.SALES\_ORDER
* MAGENTO\_PROD\_PROD.CUSTOMERS\_ENTITY

# Final Tables and their Descriptions:

## · Customer Master Table: CUSTOMERS\_MASTER

This table records the customer details like patient type, date of account creation along with the average financial metrics and some basic customer behavioral data like acquisition sources, date of association, latest and first purchase date, sales channel for latest and first purchase, latest zip codes, latest depot used used and a new or a repeating customer for each individual customer. The technical details follow as:

* **Data Sources:**

1. Primary Data Source (Source of Truth): Treez customer table
2. Secondary Data Sources: Magento (CUSTOMER\_ENTITY table, SALES\_ORDER table), Treez Ticket table

* **ETL frequency and timing:**

1. Ingestion or Extraction Frequency:
   1. Treez – Every **60 minutes** (Average time taken to ingest raw data: 90 seconds as per date 6th of June,2021)
   2. Magento – Every **120 minutes** (Average time taken to ingest raw data: 2 minute 30 seconds as per date 6th of June,2021)
2. Transformation and Load Frequency: Every **3 hour** (Average time taken: 18 seconds as per date 6th of June,2021)

## · Customers Details Table: CUSTOMERS\_SOURCE\_INFO

This table records just the basic customer details like name, email, treez ids, phone and date of birth for each individual customer. The technical details follow as:

* **Data Sources:**
* Primary Data Source (Source of Truth): Treez customer table
* Secondary Data Sources: Magento (CUSTOMER\_ENTITY table)
* **ETL frequency and timing:**
* Ingestion or Extraction Frequency:
  + Treez – Every **60 minutes** (Average time taken to ingest raw data: 90 seconds as per date 6th of June,2021)
  + Magento – Every **120 minutes** (Average time taken to ingest raw data: 2 minute 30 seconds as per date 6th of June,2021)
* Transformation and Load Frequency: Every **3 hour** (Average time taken: 8 seconds as per date 6th of June,2021)

## · Delivery Tasks Table: DELIVERY\_TASKS\_MASTER

This table records the delivery tasks under OnFleet and their respective task states as individual records at the table. This tables basically tracks the transition between the various states and the time taken in each state along with some filtered and other derived data fields details of each individual task. The technical details follow:

* **Data Sources:**
* Primary Data Source (Source of Truth): OnFleet Task table (TASKS table)
* Secondary Data Sources: Magento (SALES\_ORDER table), CUSTOMER\_MASTER table, TRANSACTIONS\_MASTER table, Onfleet Workers, Hubs and Teams Tables.
* **ETL frequency and timings:**
* Ingestion or Extraction Frequency:
  + Onfleet – Every **60 minutes** (Average time taken: 5 minutes and 2 seconds as per date 6th of June,2021)
* Transformation and Load Frequency: Every **3 hours** (Average time taken: 12 seconds as per date 6th of June,2021)

## · Transactions Items Table: TRANSACTION\_ITEMS\_MASTER

This table records the product level details of associated with transactions, which basically includes the product name, type, sub-type, price, taxes involved, etc. The technical details follow:

* **Data Sources:**
* Primary Data Source (Source of Truth): Treez Tickets table
* Secondary Data Sources: DOMO data (INVOICE table, INVENTORY table)
* **ETL Frequency and timings:**

1. Ingestion or Extraction Frequency:
   1. Treez – Every **60 minutes**
   2. Domo – Every **6 hours**
2. Transformation and Load Frequency: Every **3 hour** (Average time taken: 7 seconds as per date 6th of June,2021)

## · Transactions table: TRANSACTIONS\_MASTER

This table records the various transaction along with the financial columns, sales channels, and onfleet task id and their status based on orders placed on Treez. The Technical details follows:

* Data Sources:

1. Primary Data Source (Source of Truth): Treez Tickets table
2. Secondary Data Sources: OnFleet (TASKS table), Magento (SALES\_ORDER), AMUSE\_DEPOTS table.

* ETL Frequency and timings:
* Ingestion or Extraction Frequency:
  + Treez – Every **60 minutes**
  + Onfleet – Every **60 minutes**
* Transformation and Load Frequency: Every **3 hours** (Average time taken: 13 seconds as per date 6th of June,2021)

## · Worker Analytics table: ONFLEET\_WORKER\_ANALYTICS

This table records the details of the workers and their basic details like name, phone, id along with task related details like task failed, task succeeded, distance and vehicle in onfleet . The Technical details follows:

* Data Sources:

1. Primary Data Source (Source of Truth): Ingestion onfleet worker analytics (Worker Analytics table).

* ETL Frequency and timings:

1. Ingestion or Extraction Frequency:
   1. Treez – Every **60 minutes**
   2. Onfleet – Every **60 minutes**
2. Transformation and Load Frequency: Every **3 hours** (Average time taken: 13 seconds as per date 6th of June,2021)

## · Worker Analytics table: DAILY\_METRICS

This table is a day-based aggregation of the metrics like customers, discounts, revenue,fees , taxes and net sales all of which are divided by sales channels (Weedmaps, 710 Labs - the list, Concierge, Amuse.com, Houseplant). The Technical details follows:

* Data Sources:
* Data Source: transactions master, customers\_master and delivery\_tasks\_master.
* ETL Frequency and timings:

1. Ingestion or Extraction Frequency:
   1. Treez – Every **60 minutes**
   2. Onfleet – Every **60 minutes**
2. Transformation and Load Frequency: Every **3 hours** (Average time taken: 7 seconds as per date 6th of June,2021)

## Data pipeline

* Amazon Simple Storage Service (Amazon S3) is considered as the primary data store. Given the scalability and high availability of Amazon S3, it is best suited as the single source of data.

# SETTING UP CONNECTORS

# STORED PROCEDURES

* Stored procedures were written using SQL, one file for each master table.
* Each file includes all the transformations for the corresponding master table.
* Stored procedures are available here -

# TRANSFORMATIONS USING DBT

## Need to use DBT:

Stored Procedures have the following disadvantages:

* Query to create final table from the source data is too long and hence readability of the code is poor.
* Difficult to debug\troubleshoot the code later.
* Difficult to comprehend for a person who was not involved in the development.
* Lack of flexibility to incorporate major improvements in the code.
* Major code changes required even for small improvements in the final table.
* Difficult to do unit testing on the code.

To overcome these, it was decided to do transformations in snowflake using DBT.

## MODELS AND TRANSFORMATION DONE:

|  |  |  |
| --- | --- | --- |
| # | MODEL NAME | IMPORTANT TRANSFORMATION DONE |
| CUSTOMERS MASTER | | |
| 1 | customer\_first\_order | Details for customer’s first completed order found. |
| 2 | customer\_last\_order | Details for customer’s last completed order found. |
| 3 | Cohort\_columns | To get cohort date time columns for customer master |
| 4 | Is\_converted | To check whether the customer is converted or not. |
| 5 | Acquisition\_info\_support | Support model for first and last order models. |
| 6 | Customers\_data | Transacted customer details to be used for transformations. |
| 7 | Existing\_customers\_data | Recently transacted customers who exist in the master table |
| 8 | Delivery\_information | Stats of deliveries made to the customers and orders made. |
| 9 | Order\_information | Financial stats for the orders made. |
| 10 | Customers\_revenue | Revenue and sales financial info pertaining to the customer. |
| DAILY METRICS | | |
| 1 | L1\_stg\_metrics\_COM\_data |  |
| 2 | L1\_stg\_metrics\_concierge\_data |  |
| 3 | L1\_stg\_metrics\_houseplant\_data |  |
| 4 | L1\_stg\_metrics\_list\_data |  |
| 5 | L1\_stg\_metric\_WM\_data |  |
| DELIVERY TASKS | | |
| 1 | L1\_stg\_onfleet\_tasks\_completed. |  |
| 2 | L1\_stg\_onfleet\_CM\_phone\_numbers |  |
| 3 | L3\_stg\_onfleet\_customer\_details |  |
| 4 | L2\_stg\_onfleet\_requirement\_details |  |
| 5 | L2\_stg\_onfleet\_tasks\_recepients |  |
| 6 | L2\_stg\_onfleet\_task\_details |  |
| 7 | L2\_stg\_onfleet\_task\_metadata |  |
| 8 | L2\_stg\_onfleet\_timestamps |  |
| 9 | L3\_stg\_onfleet\_notes |  |
| 10 | L1\_stg\_onfleet\_TM\_transaction\_data |  |
| 11 | L2\_stg\_onfleet\_order\_numbers |  |
| 12 | L3\_stg\_onfleet\_transaction\_details |  |
| 13 | L4\_stg\_onfleet\_transaction\_details |  |
| 14 | L5\_stg\_onfleet\_worker\_hub\_details |  |
| TRANSACTIONS | | |
| 1 | stg\_delivery\_information | To check if the task is replacement one. |
| 2 | stg\_existing\_transactions | Transactions from historic data for recent customers |
| 3 | stg\_is\_first\_order | To get flag is\_first\_order , for sales cahnnels adn Amuse. |
| 4 | stg\_items\_flattened | Extraction of item level details for each order |
| 5 | stg\_transaction\_deliveries |  |
| 6 | stg\_treez\_financial\_information | Calculation of financial data per transaction |
| 7 | stg\_treez\_items\_finances | Aggregation of flattened data for financial columns over transaction id |
| 8 | stg\_treez\_order\_information | Extraction of sales channel based on ticket note, inventory location and order source. |
| TRANSACTIONS ITEMS | | |
| 1 | tickets\_flatten |  |
| 2 | items\_inventory |  |
| 3 | margin\_calculation |  |
| 4 | products |  |
| 5 | discount\_calculaton |  |
| 6 | tax\_columns |  |
| 7 | tickets\_aggregation |  |

## IMPORTANT FEATURES INCORPORATED INTO DBT:

* Usage of Jinja templates to automatically change the code and scale up as new depots get added.
* Declaring variables so that any minor change to the code can be done in dbt\_project.yml through jinja templates.
* Incremental update of ingestion data in source models based on the date column in the respective tables, so that transformations done only for recent data.
* Models are separated according to the topic and the level they are in with respect to the final table.
* Tables are not created for intermediate models and are represented as CTEs except staging tables.
* Models are created and named according to the kind of transformation that is done on the data.
* No transformation is done in the final models.
* Aggregation and other such transformations are always performed at the intermediate models.

<flow of code>