**Project 1: Migrate Dynamic Slotting from Snowflake to GCP**

**What problems are we trying to solve ?**

Currently the Dynamic Slotting model is held in Snowflake and managed by the Business Intelligence (BI) team. First we need to migrate data, data pipeline and models from Snowflake to GCP; Second we need to transfer this work from BI to Data Lake + Data Science, so that in the future

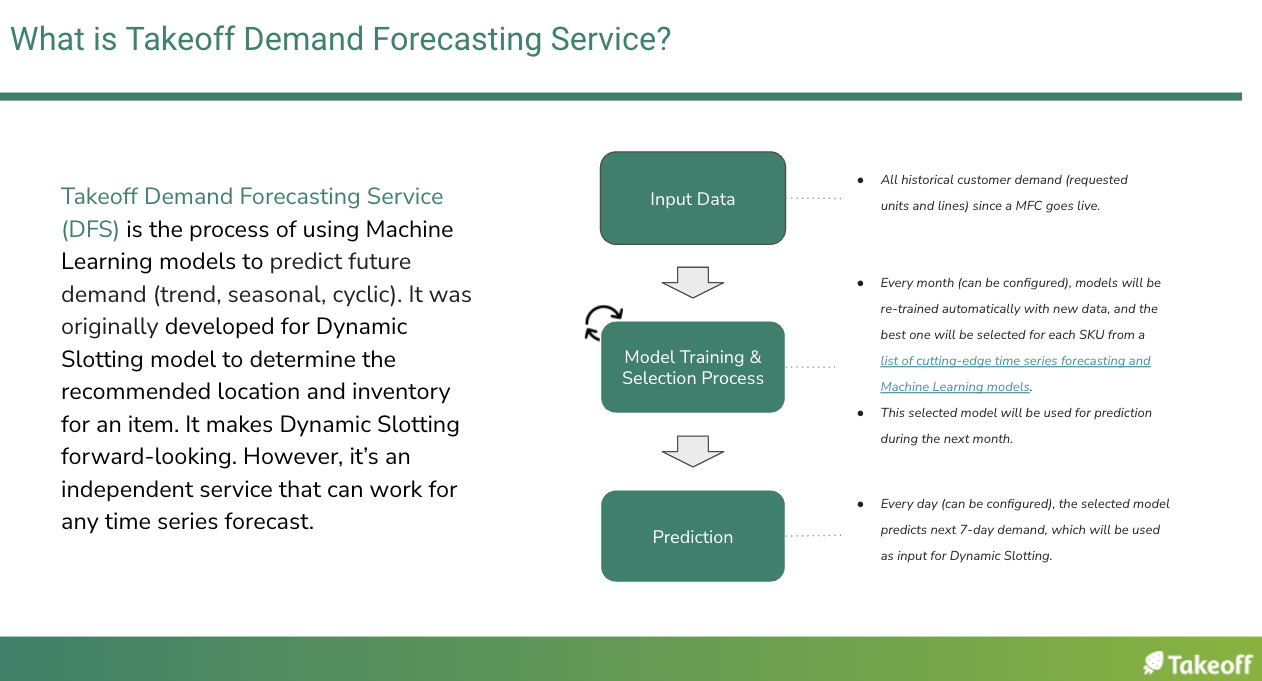
* BI is not going to support any data science models.
* Data Lake will be responsible for ETL any data into the data lake needed by data science projects.
* Data Science will be responsible for building their own data pipelines to load and transform data from data lake to the data science data mart.

Right now Data Lake and BI are still busy with the BI migration. There is no clear timeline when we can get to the data science migration. This becomes a concern because 1) we don’t know when we can retire Snowflake to avoid additional cost; 2) the Assortment Management Platform (a front end UI platform to hold Dynamic Slotting recommendations as well as customer interactions) will be released soon so customer decisions will be sent to GCP BigQuery while Dynamic Slotting model is in Snowflake. Therefore, the data science team won’t be able to incorporate customer decisions into the model to build the feedback loop.

**Potential scope of this project for Quantiphi**

* Set up a data science environment in GCP (datamart - BQ instances, dbt, IAM…)
* Load source data needed by Dynamic Slotting from source to the Data Lake (there could be data already loaded in the Data Lake during the BI migration)
* Migrate Dynamic Slotting data, data pipelines, transformations/procedures from Snowflake to GCP, including all historical input data for model training purpose, and all historical output data
* Make sure all existing issues in the data pipeline are fixed
* Work with data science team to make sure existing services (Demand Forecaster, Model Selector and SDIA) can be reconnected to new input and output tables in GCP
* Train Data Lake and Data Science teams so they can take over Dynamic Slotting model, but also use the same process to build future data science models

****

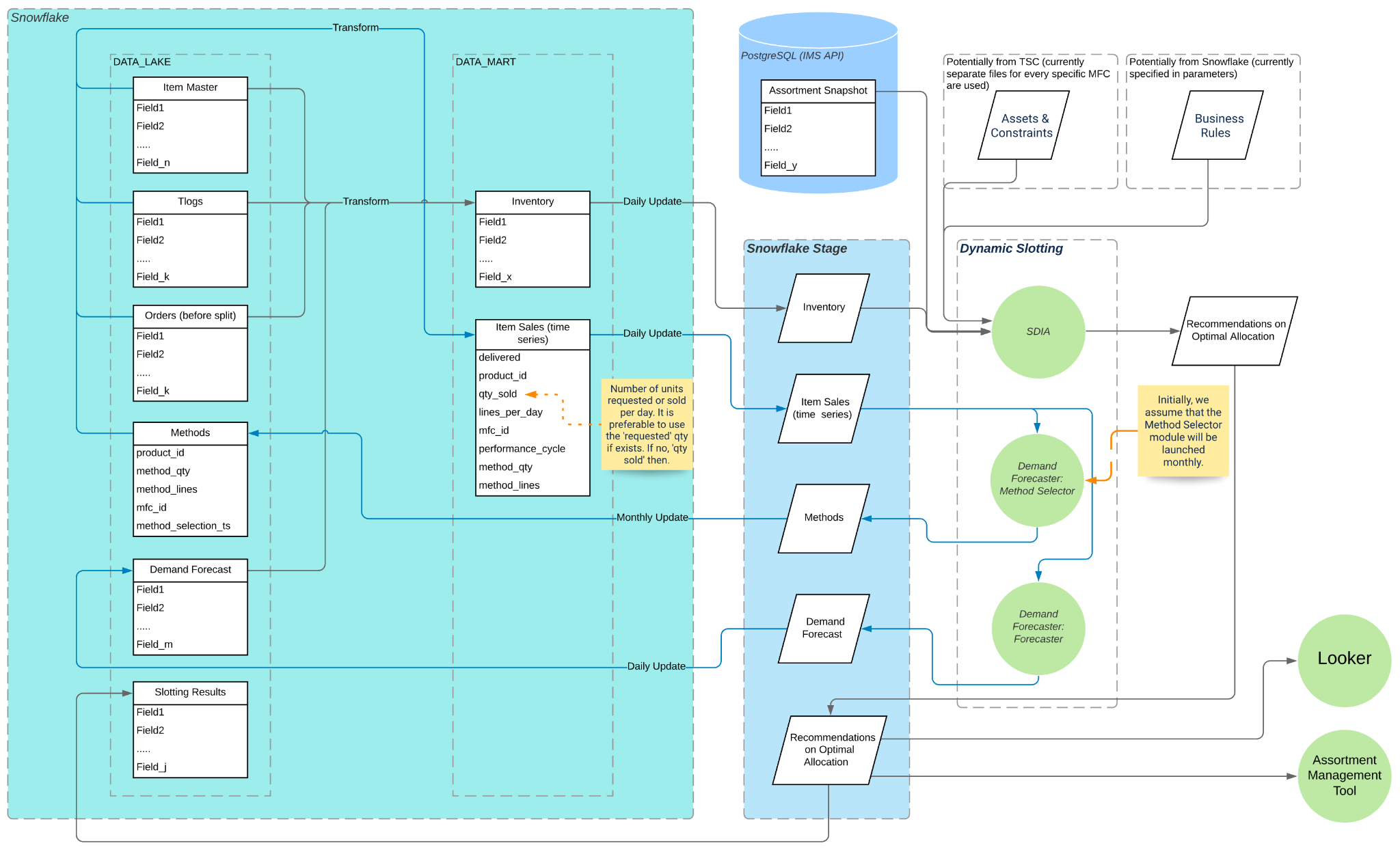
****

**Dynamic Slotting data in Snowflake**

| **Category** | **Table Name** | **Purpose** |
| --- | --- | --- |
| Source- Demand | Order Management System (OMS) | Build raw customer demand time series (requested units, requested lines, OMS created date). |
| Source - Demand | Tlogs (Woolies only) | Get shop floor demand for retailers who don’t send us full demand data in OMS. Use it to replace OMS data. |
| Source Item Master | Product Catalog | Product attributes are used in slotting models or to build business rules. |
| Source - Inventory Snapshot | Inventory Snapshot | For items that don’t have demand but still have inventory in the MFC, we need to include these items so they can be recommended to move out from the MFC. |
| Model Intermediate Table | Item Sales | Use OMS or Tlogs to build demand time series which will be used as input for Model Selector and Demand Forecaster. |
| Model Intermediate Table | MO\_Methods | After the monthly forecasting model training process is done in Method Selector, meta data for selected models will be stored in this table. |
| Model Intermediate Table | MO\_Demand\_Forecast | On a daily basis, demand is forecasted for each item for the next 7-14 days (depending on customer requirements), by using the models selected in MO\_Methods. This table is used to store forecasted demand. |
| Model Intermediate Table | Inventory | Merge Demand Forecast, Product Catalog, statistics from OMS/Tlogs and Inventory Snapshot to create a big-wide table as input for SDIA (Dynamic Slotting model). |
| Model Output Table | Consolidated Inventory | Output from SDIA after assigning product priority and target location for each item along with other important information. |
| Model Output Table | Recommendations on Item Movement | Show daily recommendations only. It is derived from Consolidated Inventory. |

[**Dynamic Slotting Architecture**](https://takeofftech.atlassian.net/wiki/spaces/DS/pages/2790326620/Dynamic+Slotting+MVP+Architecture+Overview)

**Data Flow**

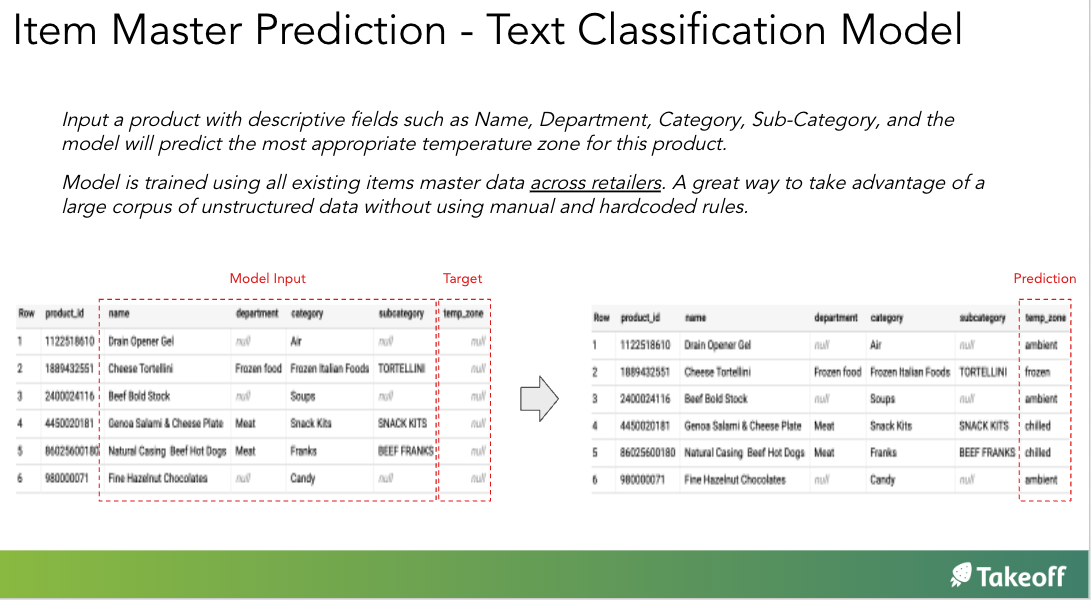
****

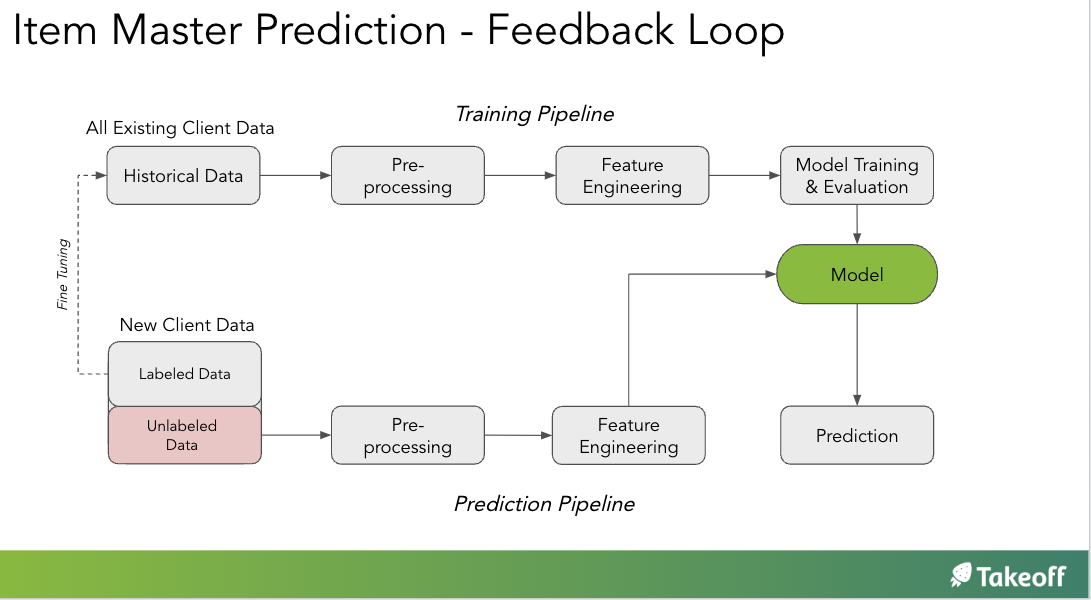
**Project 2: Build Continuous and Automatic Model Training and Monitoring pipelines in GCP (ML Ops) for Item Master Prediction**

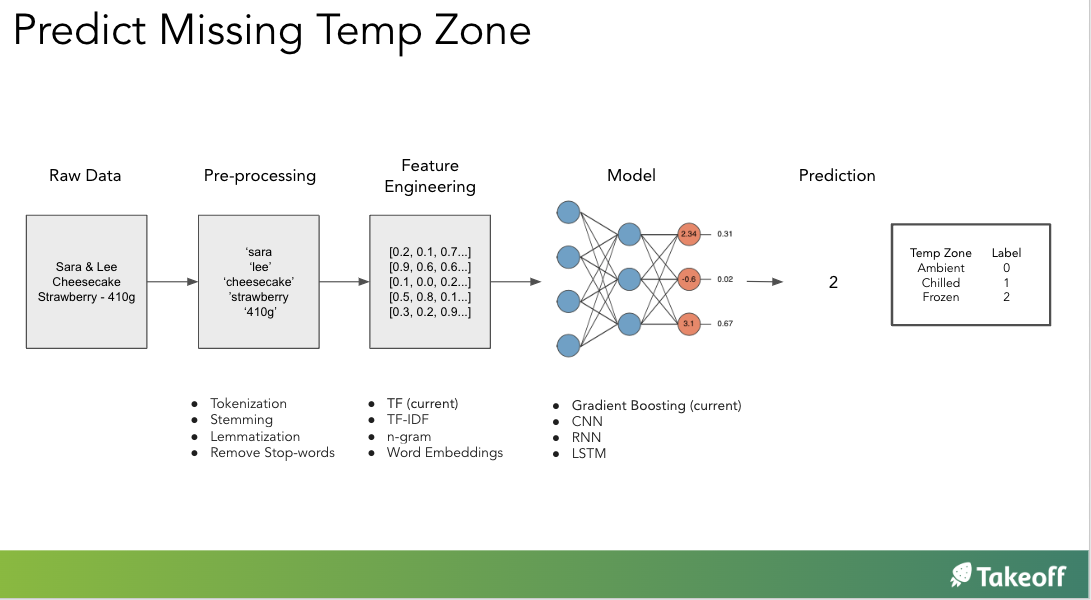
**What problems are we trying to solve (***add Epic Brief link if we have***)?**

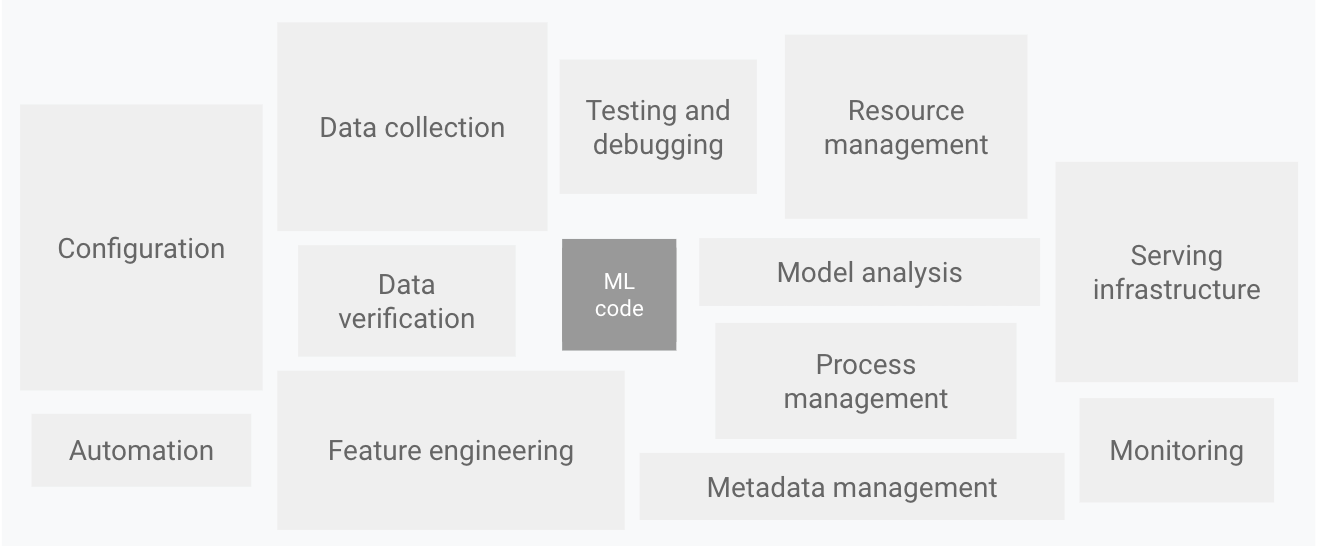
Item Master Prediction is a service to predict missing or suspicious product attributes, such as temp zone, product dimensions. We train models with item master data across retailers. Currently, we build a one-time model and there is no way to continuously train, evaluate and monitor the model. In addition, other than the model itself, we do have various components in the MLOps process, such as CI/CD, but they are in different places.

We want to build a more scalable process and an integrated ML system to continuously operate in production by leveraging end-to-end ML Ops with Vertex AI in GCP.









**Potential scope of this project for Quantiphi**

* Propose and design ML Ops infrastructure for Item Master Prediction model
* Build MLOps for Item Master Prediction model, including but not limited to
  + Tools to easily manage data, data pipelines and models. We currently have a process of doing all these. But may need to refactor so that it can work in Kubeflow or VertexAI.
  + Continuously train and evaluate models with live data
  + Continuously and automatically monitor model performance metrics, input/output distribution, data shift/concept shift, generate alerts and retrain models if required
  + Implement CI/CD pipeline automation
  + Build Model Registry as a central place to track and manage model artifacts
  + Build Feature store as a central place for organizing, storing, and serving ML features. We may want to design a feature store not just for this Item Master Prediction one service, but as a universal component that various services could work with
* Train Data Lake and Data Science teams so they can take over Item Master Prediction model, but also use the same process to build future data science models