Data Architect - Data Hub

Ramprasad Tak	1.0	Initial Version	6/02/2023
Ramprasad TaK	1.1	Version 1	20 Jun 2023
Ramprasad TaK	1.2	Version 1	20 Jul 2023

Summary

1- Introduction

1.1- Purpose of the document

This document describes how the datahub process works for all the GCP environments.

1.2- Intended Audience

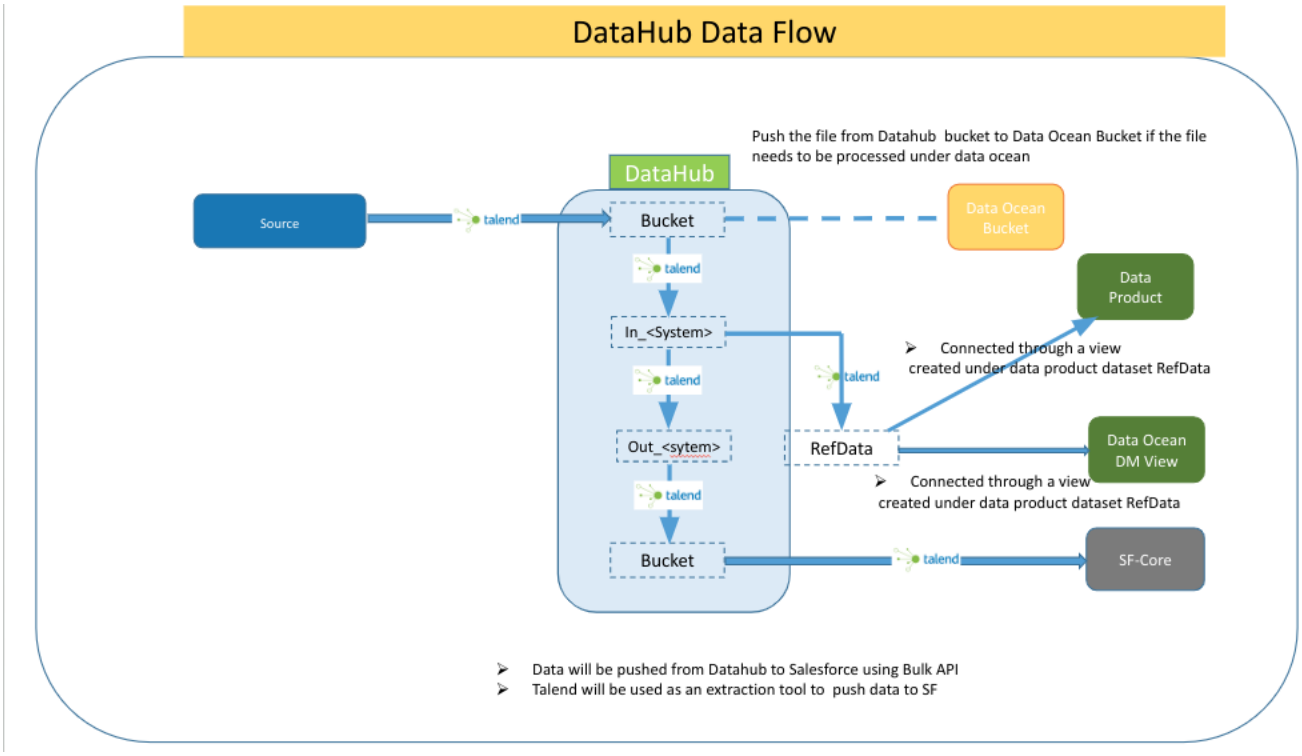
This document is intended for the Data Architectures, Data Engineering and operational team.
It will be used as reference for any project or domain for the developments of the models.

2- How it works

2.1- Description

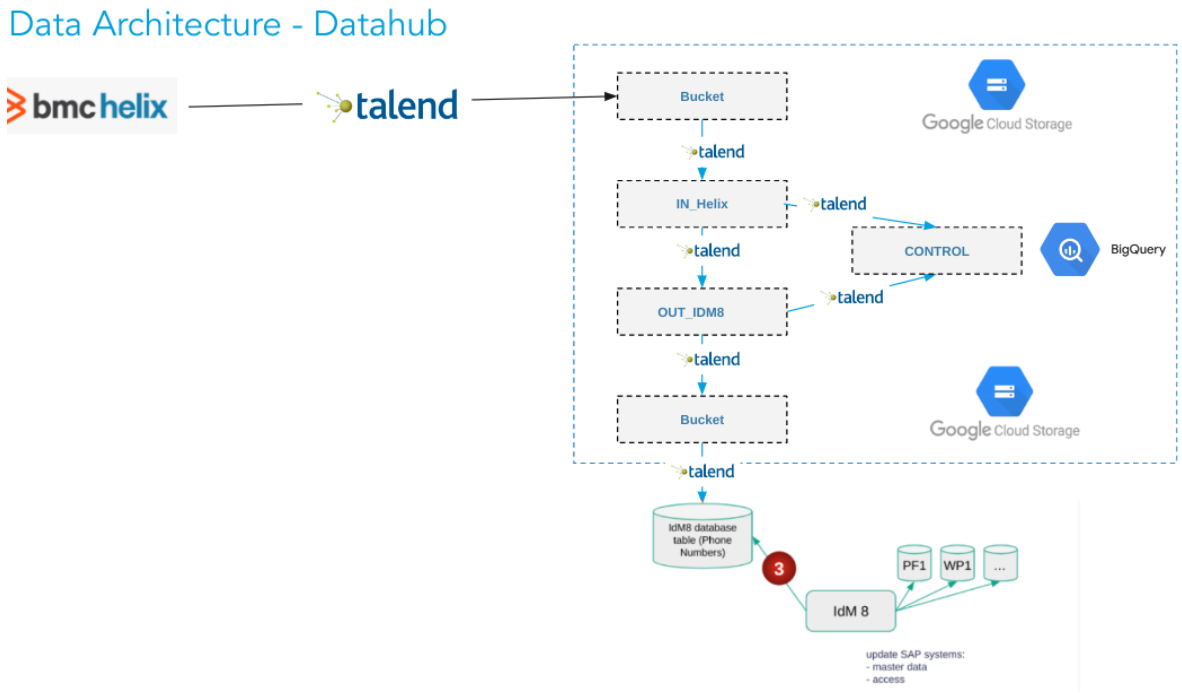
The objective of this datahub system is to guarantee that we are able to store data sets (for example reference data) at one common place and utilize it with different projects.
Using a data hub we are able to track all the runs in GCP, if the files were well integrated, if there is an abort where it happens, when and what do we need to do to fix the issues.
This will help to support the operational teams to visualize all the file integration in the system and have the models and reports well loaded.

2.2- Datahub Process



Gudsis:

Gudsis is example of project using the datahub, as it is sending telephone numbers from the helix to GCP and it will get transformed and then send it to other application



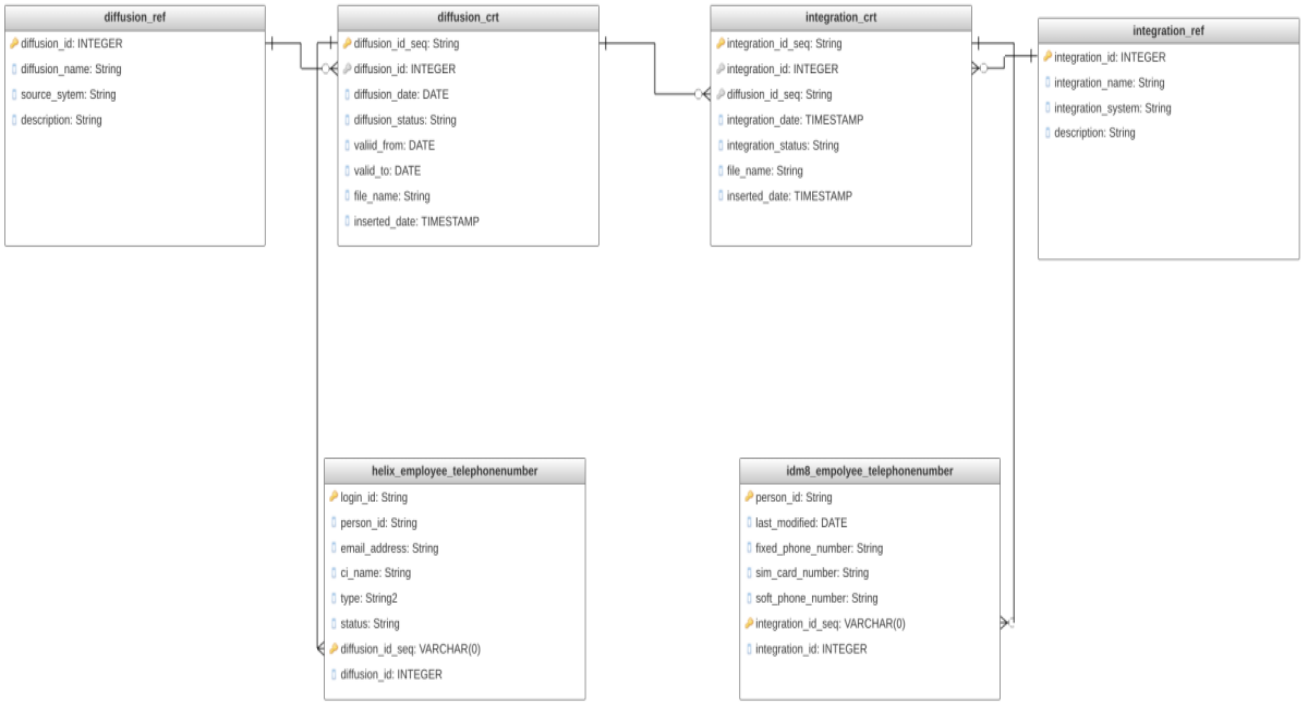
1. First Step is the bucket area. The process needs to load all the files that arrive in GCP; the information will be provided if they are integrated or not.
2. The second step is the run between bucket and reference data set where common data is stored
3. Third step is load data logs in control data set which can track the logs of file
4. The fourth step is the run between source data set to destination data set here for example IN_helix to OUT_IDM8 table
5. The fifth step is send data from

3- Datahub Model

3.1- Description

Describe all the tables that are needed for a full logging process.

3.2- Model



- **diffusion_ref** - This table will track all the sources and its table , with the end to end process in the GCP . When a job starts he needs to fill this table with a record that includes the diffusion id and source system. When the job finish this record needs to be update with the diffusion_name , source system.

- **diffusion_crt**- This table will contain all the information for all the files the job will process,it has unique key diffusion_d_seq (GUID) keeps track of every file loaded in the in_helix table, when the job starts it needs to fill the Diffusion_id from diffusion_ref table , and also unique GUID to diffusion_id_seq , by the end of job it should record the valid_from date for the file and valid_to to 9999, it also has inserted date

- **integration_crt** - This table will contain all the information for all the tables the job will load. On the output table , that has been transformed, it has information such as integration_id_seq GUID which refers to unique file which has been transformed.
- **INtegrtion_ref** : This table represent the information on the destination system and has reference to integration_re