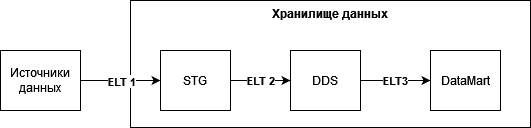
# Тестовое задание

Необходимо реализовать загрузку данных в корпоративное хранилище в слой DDS, для моделирования слоя DDS использовать методологию Data Vault 2.0. Скрипт загрузки разработать на python. Данные загружать в БД PostgreSQL. (Если нет своей тестовой БД, инструкция по получению БД в Приложении 1).

Слои хранилища представлены на схеме:



Задача:

1. Спроектировать два слоя: STG и DDS;
2. Разработать потоки перекладки и трансформации данных: ELT 1 и ELT 2.

# Тестовый источник

Как источник данных необходимо использовать REST-сервис <https://jsonplaceholder.typicode.com/posts/>. В качестве бизнес-ключей использовать поля Id и userId.

# Результат выполнения задания:

1. DDL скрипты по созданию таблиц в слоях STG и DDS;
2. python скрипты для потоков ELT 1 и ELT 2.

Решение

1. Создал диаграмму для визуального представления

<https://drawsql.app/teams/home-175/diagrams/post-diagram>

1. Создал базу данных в PostgreSQL

DDL:

**CREATE DATABASE posts**

**WITH**

**OWNER = postgres**

**ENCODING = 'UTF8'**

**CONNECTION LIMIT = -1**

**IS\_TEMPLATE = False;**

1. Создал схемы для двух слоев STG и DDS

DDL:

**CREATE SCHEMA stg**

**AUTHORIZATION postgres;**

**CREATE SCHEMA dds**

**AUTHORIZATION postgres;**

1. Создал таблицы с необходимыми сущностями

DDL STG:

**CREATE TABLE stg.stg\_posts**

**(**

**user\_id bigint NOT NULL,**

**id bigint NOT NULL,**

**title text,**

**body text,**

**PRIMARY KEY (user\_id)**

**);**

**ALTER TABLE IF EXISTS stg.stg\_posts**

**OWNER to postgres;**

DDL DDS:

**CREATE TABLE dds.dds\_posts\_hub**

**(**

**post\_hash\_key text NOT NULL,**

**PRIMARY KEY (post\_hash\_key)**

**);**

**ALTER TABLE IF EXISTS dds.dds\_posts\_hub**

**OWNER to postgres;**

**CREATE TABLE dds.dds\_posts\_satellite**

**(**

**post\_hash\_key text NOT NULL,**

**user\_id bigint NOT NULL,**

**title text,**

**body text**

**);**

**ALTER TABLE IF EXISTS dds.dds\_posts\_satellite**

**OWNER to postgres;**

**ALTER TABLE IF EXISTS dds.dds\_posts\_satellite**

**ADD CONSTRAINT post\_hash\_key FOREIGN KEY (post\_hash\_key)**

**REFERENCES dds.dds\_posts\_hub (post\_hash\_key) MATCH SIMPLE**

**ON UPDATE NO ACTION**

**ON DELETE NO ACTION**

**NOT VALID;**

1. Создал PYTHON скрипы для STG и DDS