**ORE Data Engine v1.0**

**Requirements**

* Lightweight
* Low cost running in cloud
* Easy to use
* Simple
* Multitenant
* Config-driven

**Technology**

**PostgreSQL** 9.5.5on x86\_64-pc-mingw64, compiled by gcc.exe (Rev5, Built by MSYS2 project) 4.9.2, 64-bit

***Why***

Low cost if running in AWS in comparison to MS SQL Server, open source, good support, powerful and has potential for development of quite complex product (common table expressions, analytical functions, merge, etc).

***Design approach***

Data Vault is our approach to building data store for individual clients. It gives a framework for relatively quick build and support process which means it is highly reproducible and automated. Though human effort is still needed for DV modelling and configuration.

**Blocks**

**ORE data store is** config-driven engine. It stores client’s data vault config. See Pic 1 and Pic 2.

*Config engine****(b1)***

* + Configuration database **(b1.1)**
  + Configuration setup **(b1.2)**

*Model implementation engine****(b2)***

* + Data vault setup **(b2.1)**
  + Data vault orchestration **(b2.2)**

*Service engine* ***(b3)***

* + Scheduler **(b3.1)**
  + Logger**(b3.2)**

***Configuration database***

Pretty much all structures for building data vault.

Version 1.0 includes implementation:

* Owner
* Release
* Source system -> Source table
* Stage table -> Stage table columns -> Business logic
* Hub -> hub keys -> hub columns
* Satellite -> satellite columns
* Defaults
* Auditing structures (triggers, constraints, keys, indices).

***Configuration setup***

Scripting logic to add, delete, modify config data, populate source table/satellite/hub config columns using source data.

Example – use case

Procedure to insert, update, delete config data for hub.

***Data Vault setup***

Scripting logic to create data vault objects using data stored in configuration database. Should be as simple as possible – to create objects with minimum effort.

Example – use case

Procedure to create hub.

***Data Vault orchestration***

Scripting logic to perform data vault orchestration tasks, e.g. loading data into hubs, satellites, etc.

Example – use case

Procedure to load satellite.

***Scheduler***

Manages loading tasks.

***Logger***

Contains information about functioning ORE and Data Vaults.

*Service engine*

As amazon rds PostgreSQL does not support event scheduler it should be automated externally.

**Pic 1. Architecture overview**



**Pic 2. Config data store v1.0 structure**

