### 

**SDN PROJECT**

ROUTING AND CLUSTERING METHODOLOGY

CODE AND LOGIC FLOW

Dipersiapkan oleh



**Dattabot - PT Mediatrac Sistem Komunikasi**  
Grha Tirtadi 2nd Floor   |   Jl. Senopati 71-73   |   Jakarta 12190   |   Indonesia   |   **Map** 6°13'57.37"S 106°48'42.29"E 

**P** +62 21 520 2568   |   **F** +62 21 520 4180   |   [**www.ci-mediatrac.com**](http://www.ci-mediatrac.com/)

**Daftar Isi**

[Daftar Isi 1](#_Toc485809532)

[1. Pendahuluan 1](#_Toc485809533)

[1.1. Tujuan dokumen 1](#_Toc485809534)

[1.2. Definisi dan Singkatan 1](#_Toc485809535)

[1.3 Daftar Revisi 1](#_Toc485809536)

[2. Menyiapkan Environment 1](#_Toc485809537)

[2.1. Python 1](#_Toc485809538)

[2.2. Cloning Project dari Gitlab 1](#_Toc485809539)

[2.3. Python Library 1](#_Toc485809540)

[3. Menjalankan Engine 1](#_Toc485809541)

[3.1. Mempersiapkan Input Data 1](#_Toc485809542)

[3.2. Menjalankan Engine 1](#_Toc485809543)

[3.3. Normalisasi Output Engine 1](#_Toc485809544)

[4. Engine Architecture 1](#_Toc485809545)

[4.1. Logic Flow 1](#_Toc485809546)

[4.2. Struktur Code 1](#_Toc485809547)

# 1. Pendahuluan

Dokumen ini berisi informasi mengenai Prosedur Penggunaan Engine Geocoding serta architecture dari engine tersebut sehingga memudahkan proses pemeliharaan atau versioning code. Dokumen ini disiapkan untuk Data dan GIS engineer untuk keperluan pekerjaan Geocoding. User yang dapat menggunakan engine ini ditujukan pada user yang memiliki pengetahuan mengenai basic command line dan pemahaman akan database.

## 1.1. Tujuan dokumen

Tujuan dari pembuatan dokumen ini adalah untuk mendokumentasikan code routing dan clustering untuk Dashboard Sinarmas 2.

## 1.2. Definisi dan Singkatan

|  |  |
| --- | --- |
| **Singkatan** | **Definisi** |
| MSK | PT. Mediatrac Sistem Komunikasi |

## 1.3 Daftar Revisi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Editor** | **Versi** | **Tanggal** | **Deskripsi** |
| 1 | Mia Renauly | 1.0 | 25-09-2017 | Content Writing |

# 2. Menyiapkan Environment

## Python

Engine Geocode menggunakan Python versi 3.6. Disarankan untuk menggunakan versi tersebut atau versi Python yang lebih baru. Untuk file instalasi Python dapat diunduh langsung di website resmi Python: [https://www.python.org/downloads/](https://www.python.org/downloads/%20).

Jika membutuhkan lebih dari satu environment Python, sebaiknya lakukan instalasi Python menggunakan Anaconda dari Continuum Analytics, kunjungi link berikut: <https://www.continuum.io/downloads>. Pilih Python 3.6 Version versi 64 bit.

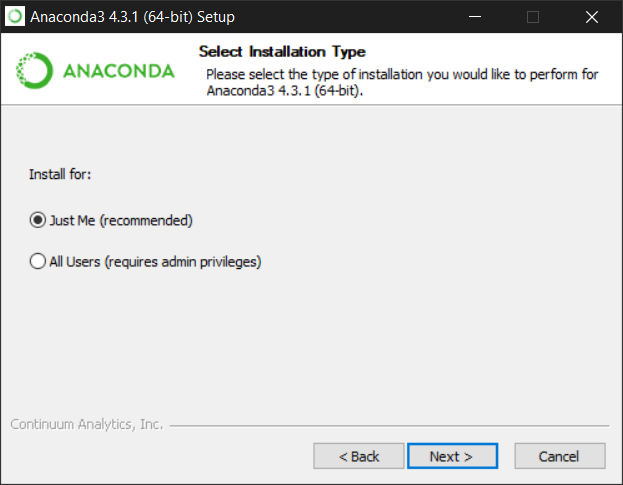
Berikut ini merupakan langkah instalasi dan cara membuat environment pada Anaconda:

1. Buka file Installer Anaconda. Akan muncul Jendela instalasi:



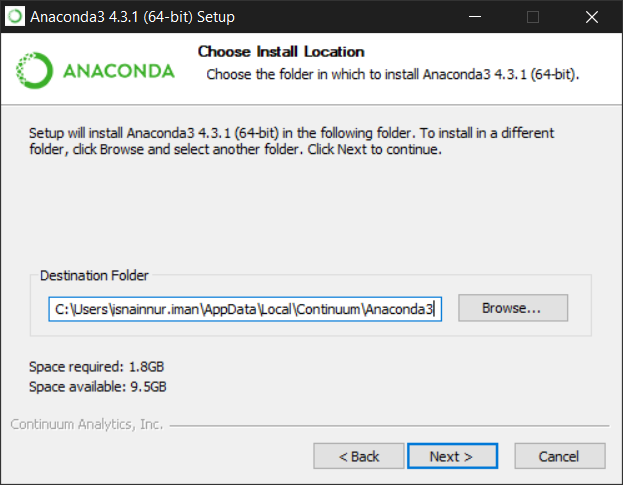
Gambar 1. Jendela Instalasi Anaconda 64-bit

1. Pada License Agreement klik I Agree. Akan muncul jendela Installation Type. Memilih All Users mewajibkan anda untuk menjalankan Anaconda Navigator sebagai Administrator:



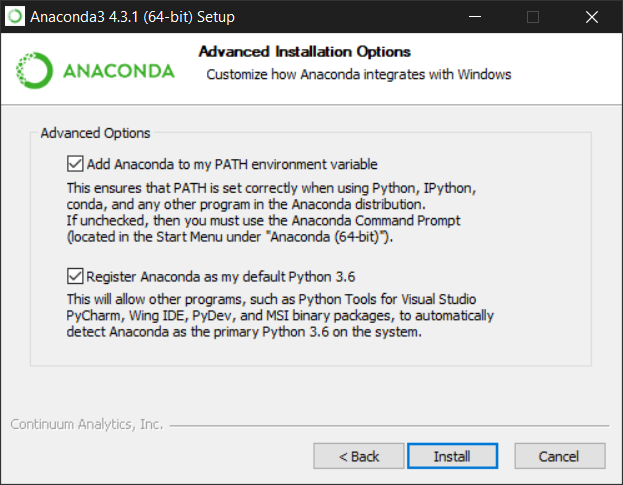
Gambar 2. Jendela Installation Type

1. Klik Next, lalu pilih direktori yang anda inginkan untuk instalasi:



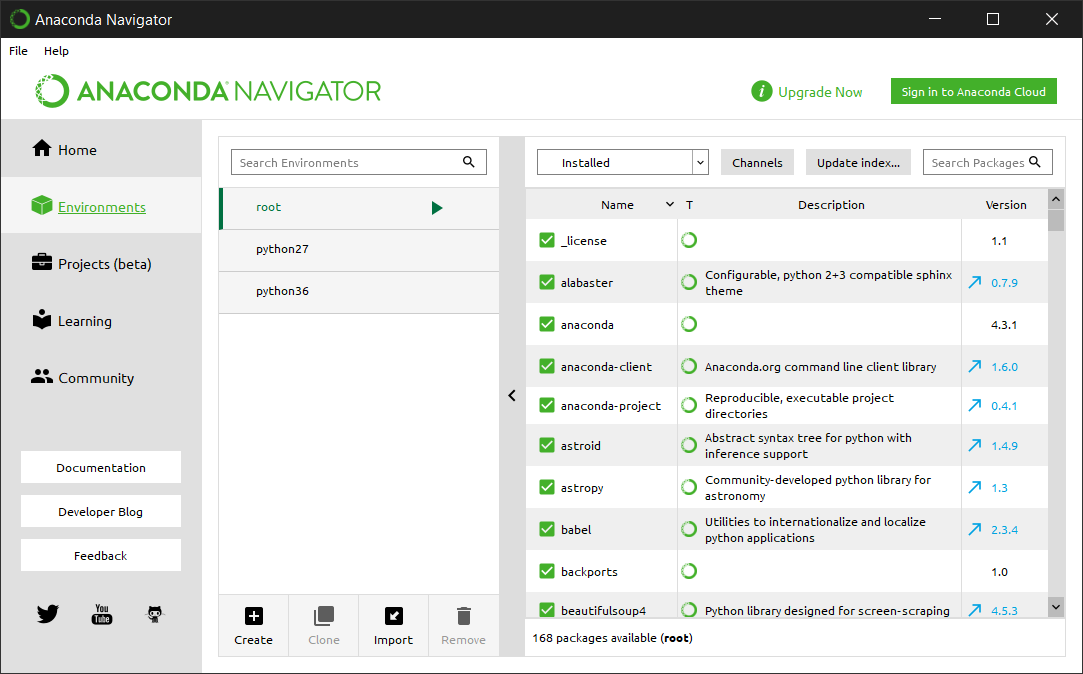
Gambar 3. Jendela Install Location

1. Centang pada kedua Check Box, lalu klik Install:



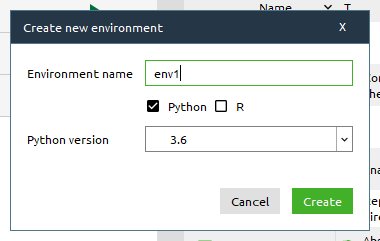
Gambar 4. Jendela Installation Options

1. Setelah instalasi selesai, buka Anaconda Navigator -> Environments. Jangan lupa untuk Run as Administrator jika anda sebelumnya instalasi dilakukan untuk All User.



Gambar 5. Jendela Environment Anaconda Navigator

1. Klik Create pada Jendela Environment, akan muncul jendela Create New Environment:



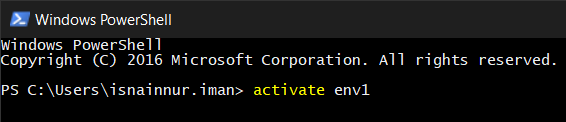
Gambar 6. Jendela Create New Environment

Pada bagian environment name, masukkan nama yang diinginkan. Centang pada Python, lalu pilih versi 3.6. Klik Create.

1. Setelah Environment dibuat. Kita harus mengaktifkan environment yang dibuat tadi, jalankan command berikut pada Command Prompt atau PowerShell:

activate [Environment Name]

Contohnya seperti berikut:



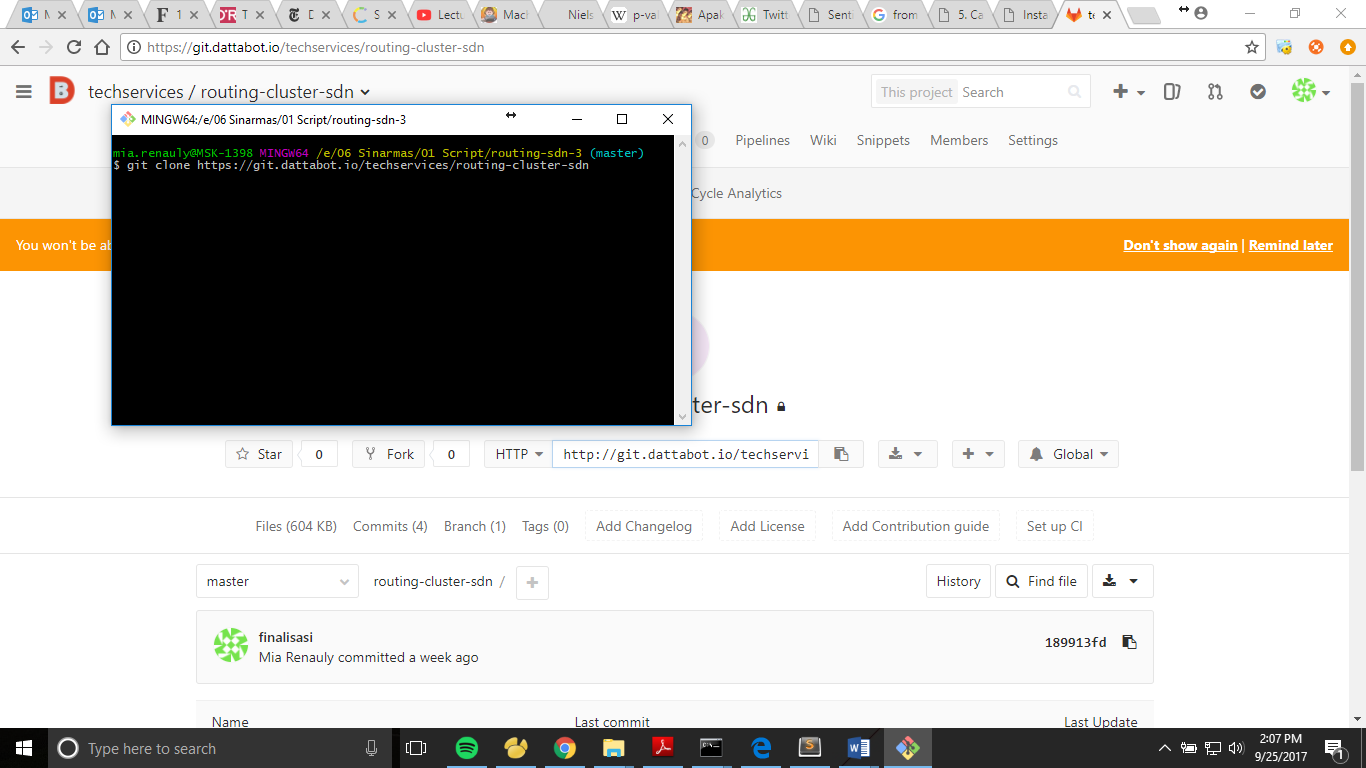
Gambar 7. Mengaktifkan Environment Anaconda

## Cloning Project dari Gitlab

Untuk Cloning Project dapat dilakukan dengan menjalankan command berikut pada jendela git bash:

**git** clone **[**url**]**

**URL: https://git.dattabot.io/techservices/routing-cluster-sdn**



Gambar 8. Cloning Project menggunakan git Bash

Jika terkendala masalah akses atau error, alternative lainnya adalah dengan mengunduh Code secara langsung di halaman git Tech Services: <https://git.dattabot.io/techservices/routing-cluster-sdn>.

## Python Library

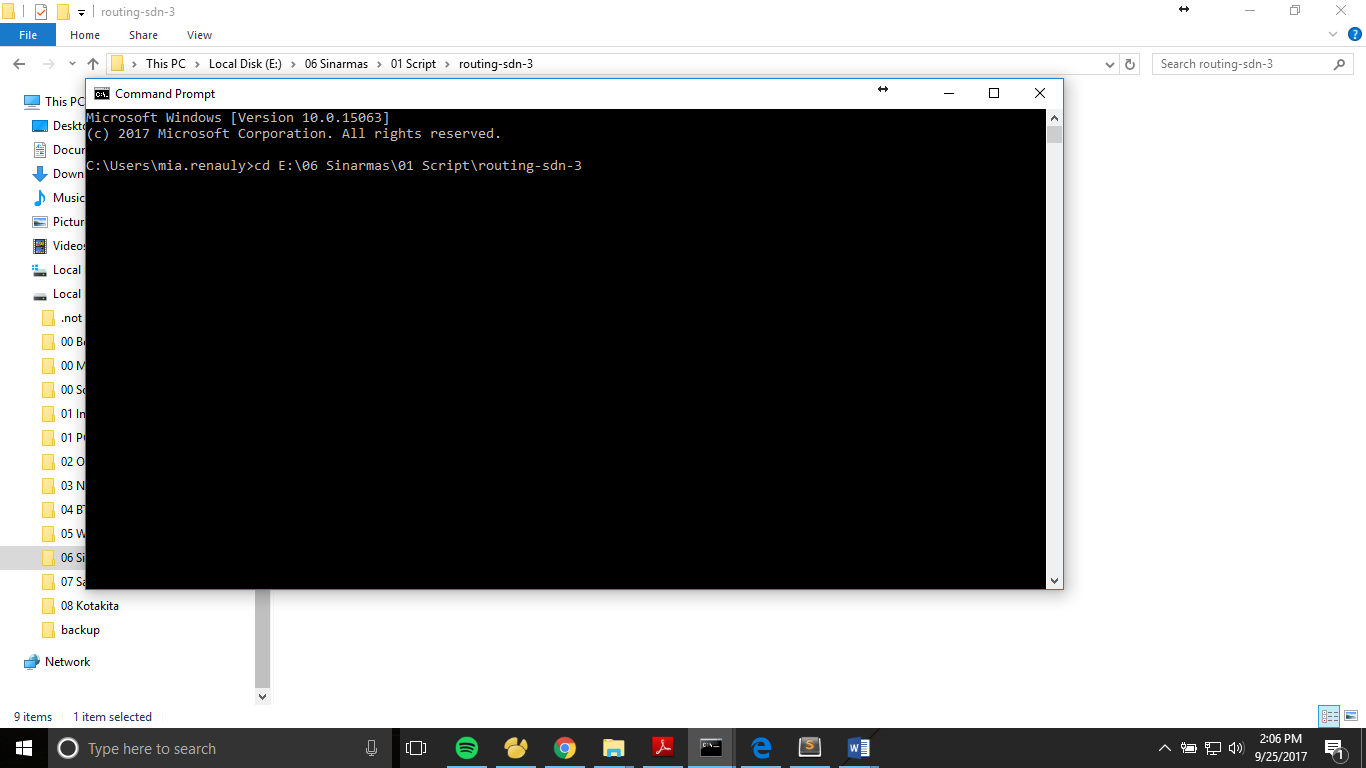
Sebelum melakukan instalasi Library Python, lakukan instalasi pip terlebih dahulu. Pip berfungsi untuk mempermudah instalasi library-library untuk Python. Instalasi pip dapat dilakukan dengan menjalankan command berikut pada Command Prompt / PowerShell:

easy\_install pip

Untuk instalasi library, terlebih dahulu masuk ke direktori utama dari Engine Geocode dengan menjalankan command berikut:

**cd** [Drive:\[directory]]

Direktori pada command diatas merupakan direktori Kode Clustering. Contoh command yang dijalankan adalah sebagai berikut:



Gambar 9. Cara Pindah Direktori menggunakan Command Prompt/PowerShell

Jalankan command berikut untuk Instalasi library menggunakan pip secara langsung dari requirements.txt:

install python3 library (pip install -r requirements.txt)

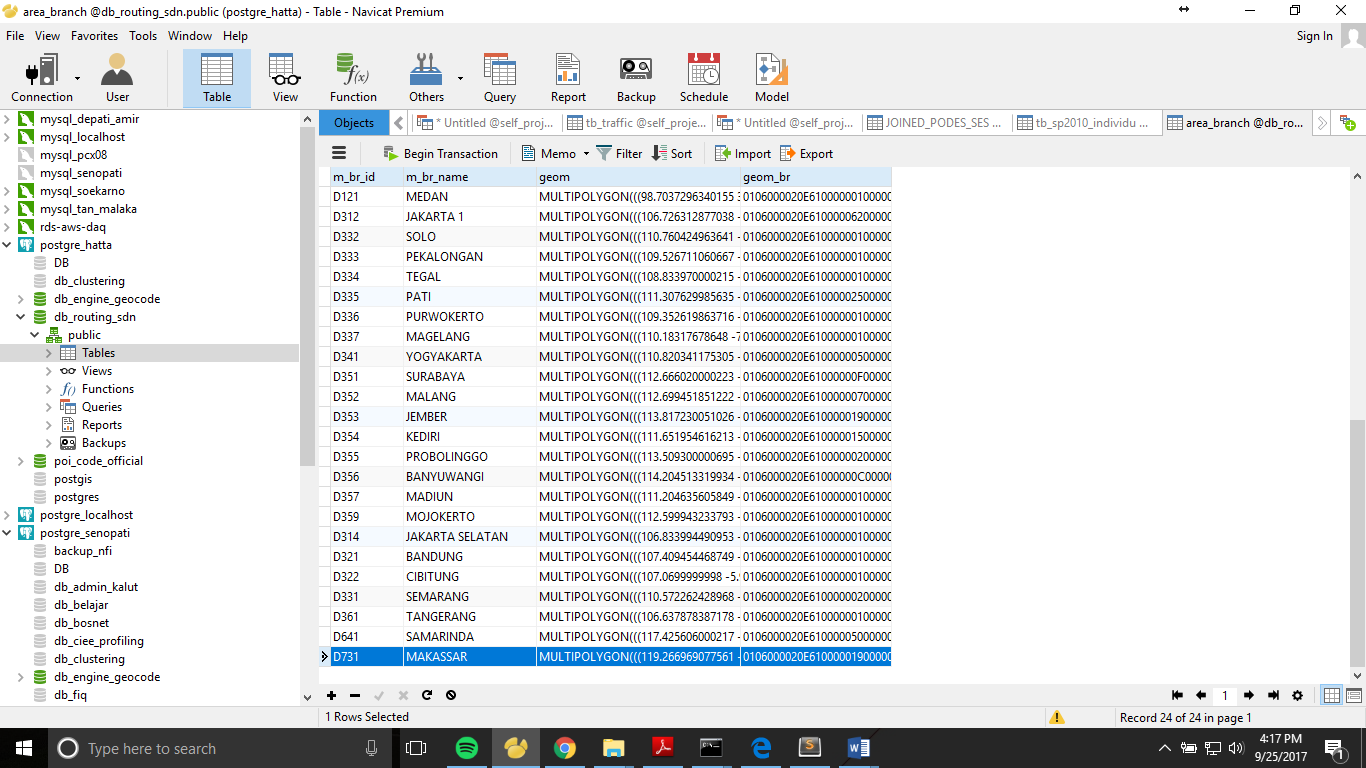
# 3. Menjalankan Engine

## 3.1. Mempersiapkan Input Data

Sebelum menjalankan Engine Geocode ada beberapa master tabel yang perlu dipersiapkan terlebih dahulu:

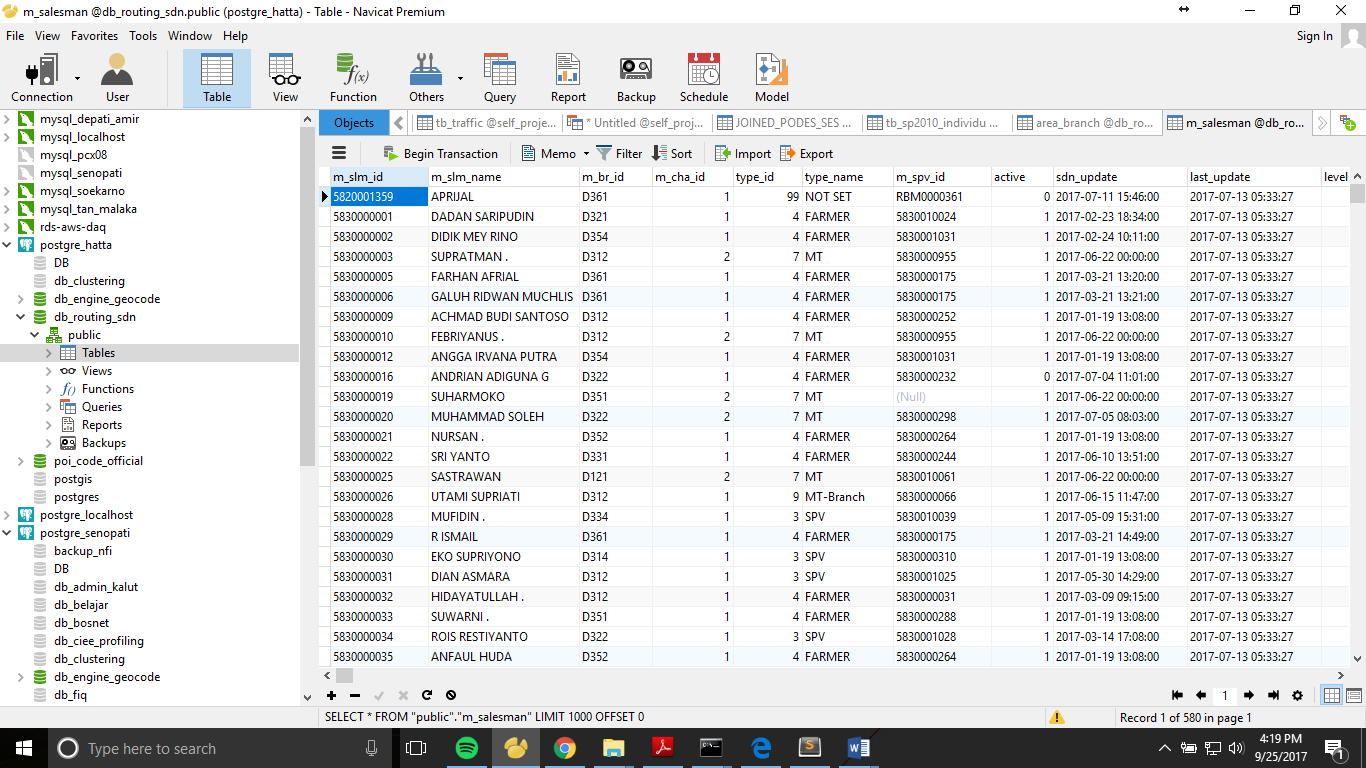
1. area\_branch

Adalah tabel yang berisikan informasi polygon geometri dari masing masing branch area SDN



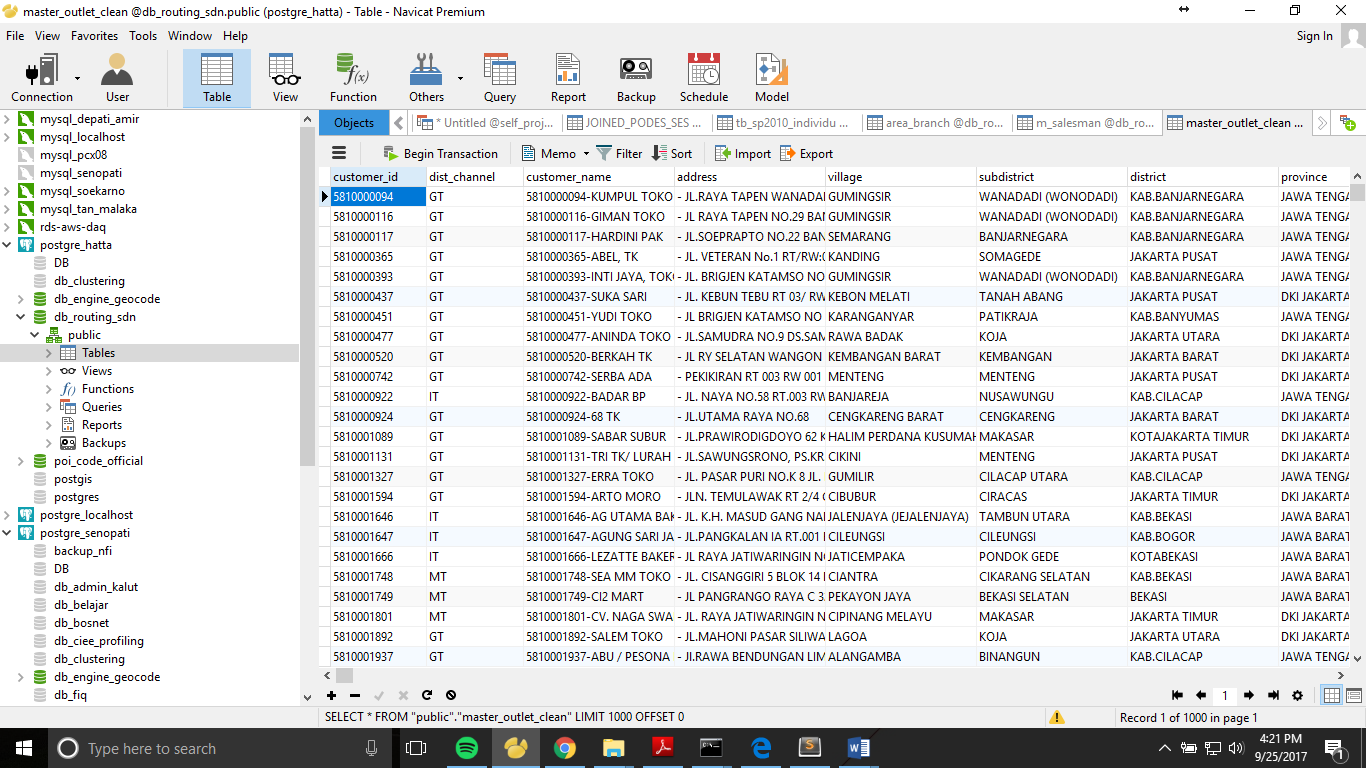
1. m\_salesman

Adalah tabel yang berisikan informasi sales di masing masing branch area SDN



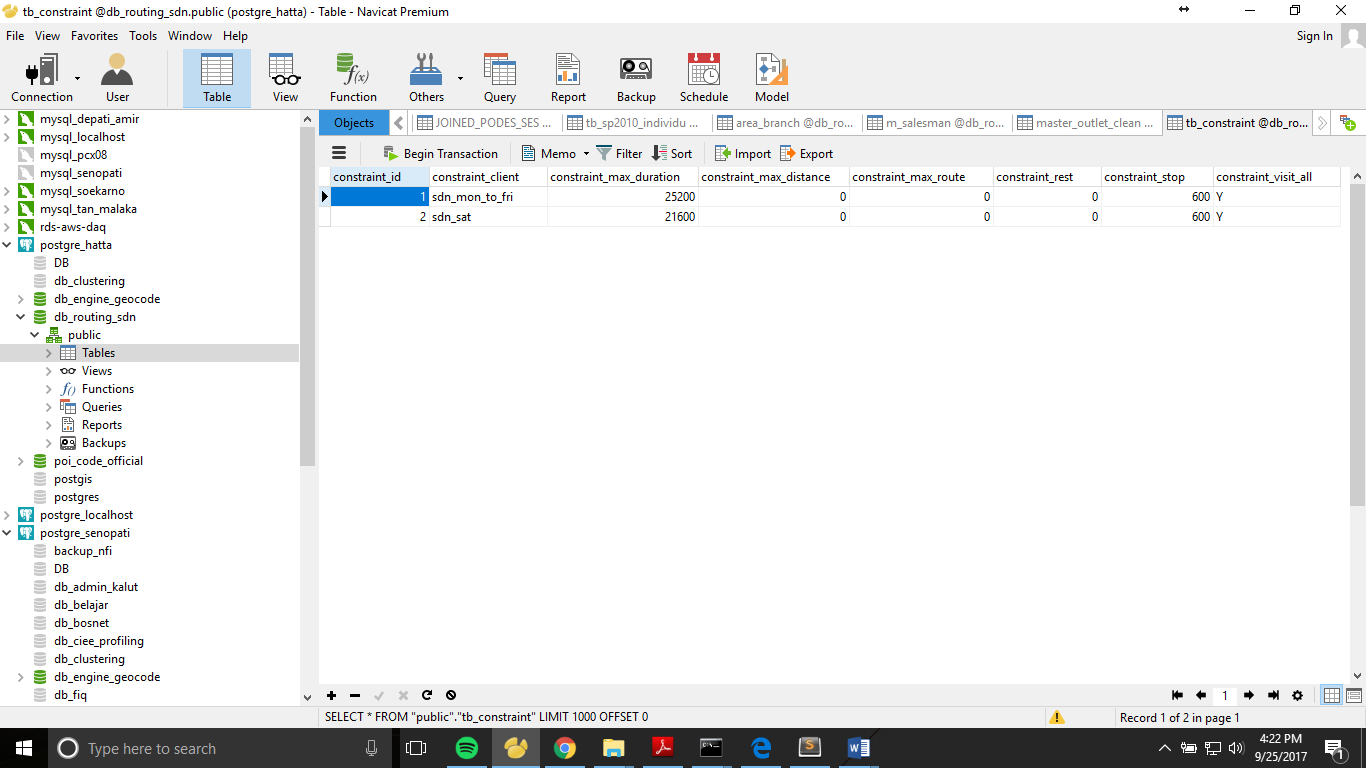
1. master\_outlet\_clean

Adalah tabel yang berisikan informasi outlet. Tabel ini merupakan output dari modul Best Longlat di Sistem Back End SmartDash SDN



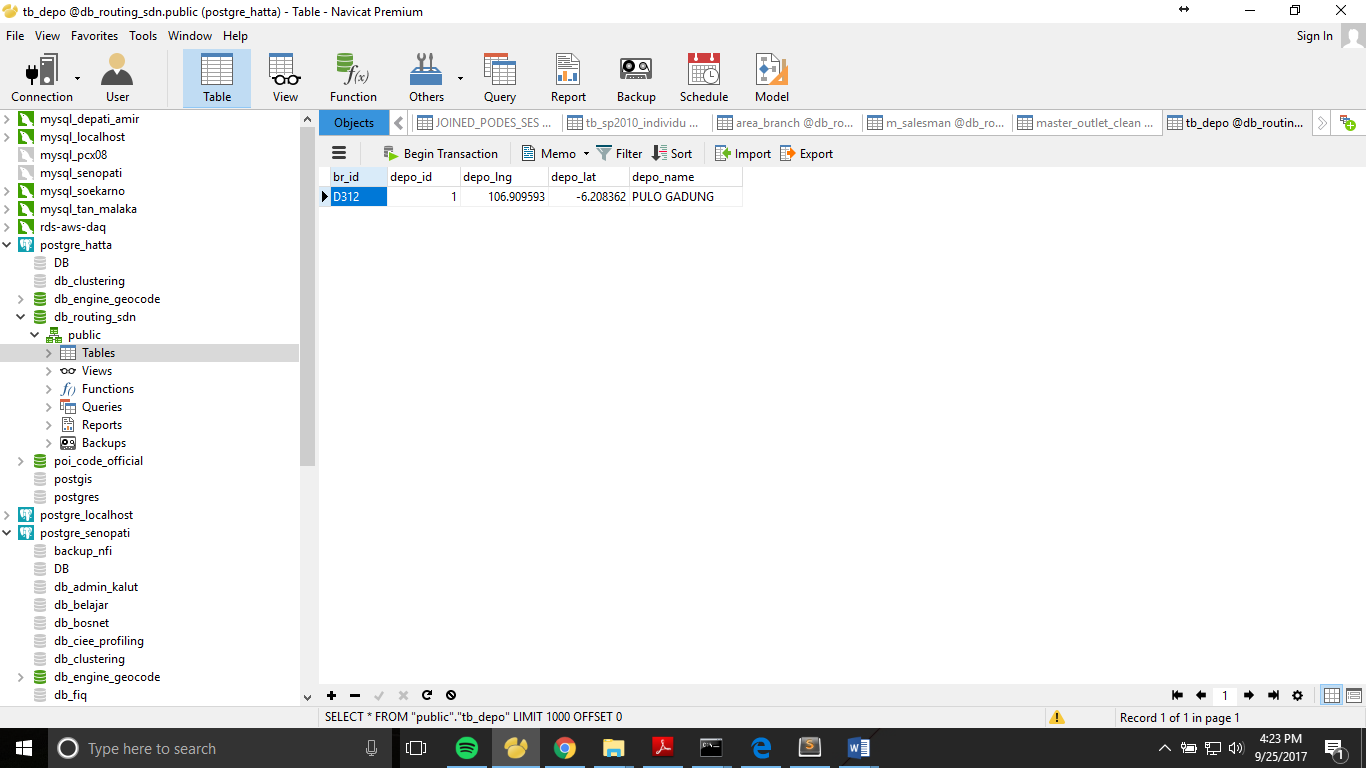
1. tb\_constraint

Adalah tabel yang berisikan informasi constraint (yang dapat diubah isinya sesuai kebutuhan).



1. tb\_depo

Adalah tabel yang berisikan informasi lokasi depo branch area SDN.



Seluruh contoh tabel master diatas dapat diakses di:

**>> Server: PostgreSQL Senopati**

**>> Database: db\_routing\_sdn**

## 3.2. Menjalankan Engine

Untuk menjalankan Engine Routing, masuk terlebih dahulu ke direktori utama Kode Routing di Command Prompt / Powershell. Kemudian jalankan kode dengan urutan seperti berikut:

1. **data\_prep.py**

Proses yang terjadi saat menjalankan kode ini adalah proses data normalisasi dan pembuatan struktur tabel turunan yang dibutuhkan untuk proses selanjutnya, serta melakukan clustering data (sesuai dengan constraint yang sudah ditentukan sebelumnya).

1. **permutation\_route.py**

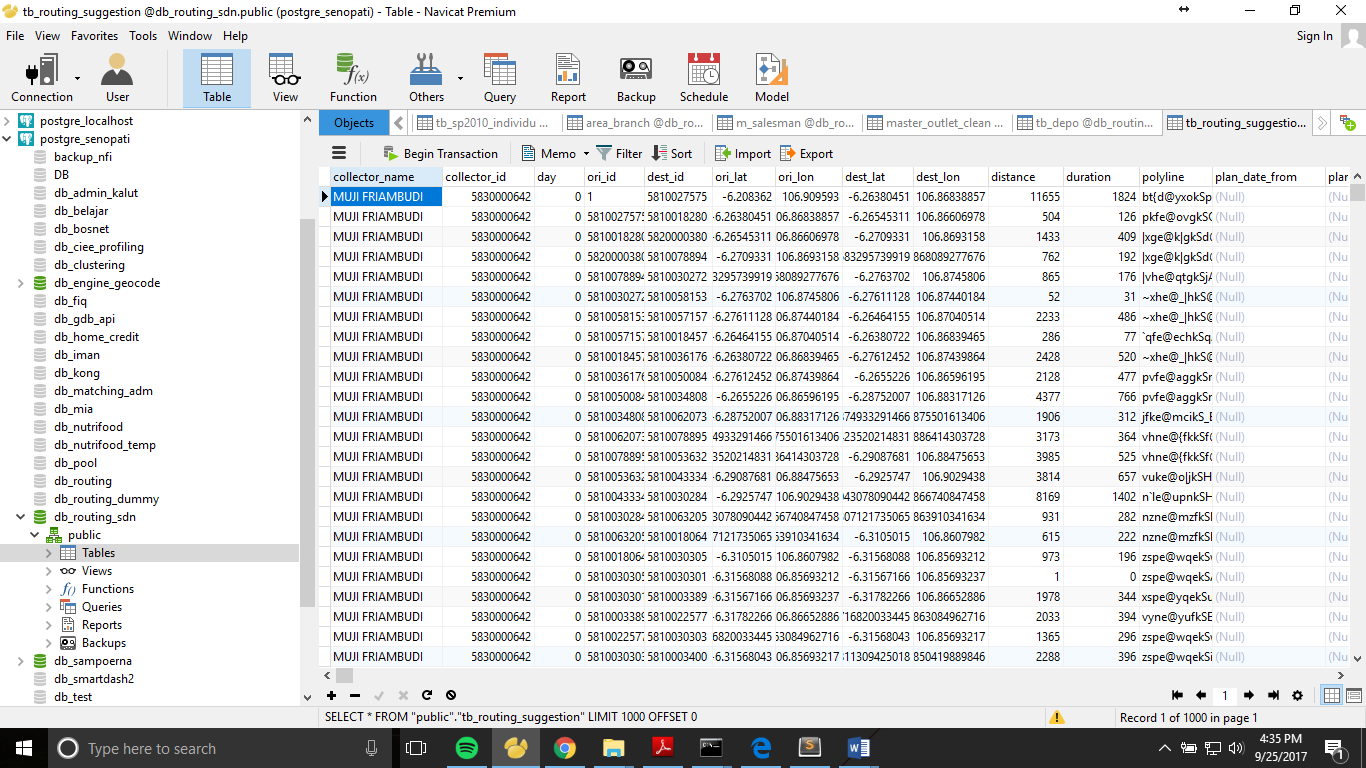
Proses yang terjadi saat menjalankan kode ini adalah melakukan permutasi dari seluruh data outlet yang sudah di buat cluster nya. Selain itu pada kode ini juga dilakukan proses crawling data terkait jarak dan kecepatan mengemudi dari Google Direction API.

1. **route\_build.py**

Proses yang terjadi saat menjalankan kode ini adalah melakukan proses iterasi pencarian outlet terdekat hingga konstrain waktu kerja masing masing sales terpenuhi (waktu tempuh antar outlet dan waktu yang dihabiskan di masing masing outlet)

## 3.3. Normalisasi Output Engine

Tabel final yang dihasilkan dari proses menjalankan rangkaian kode diatas adalah: **tb\_routing\_suggestion**



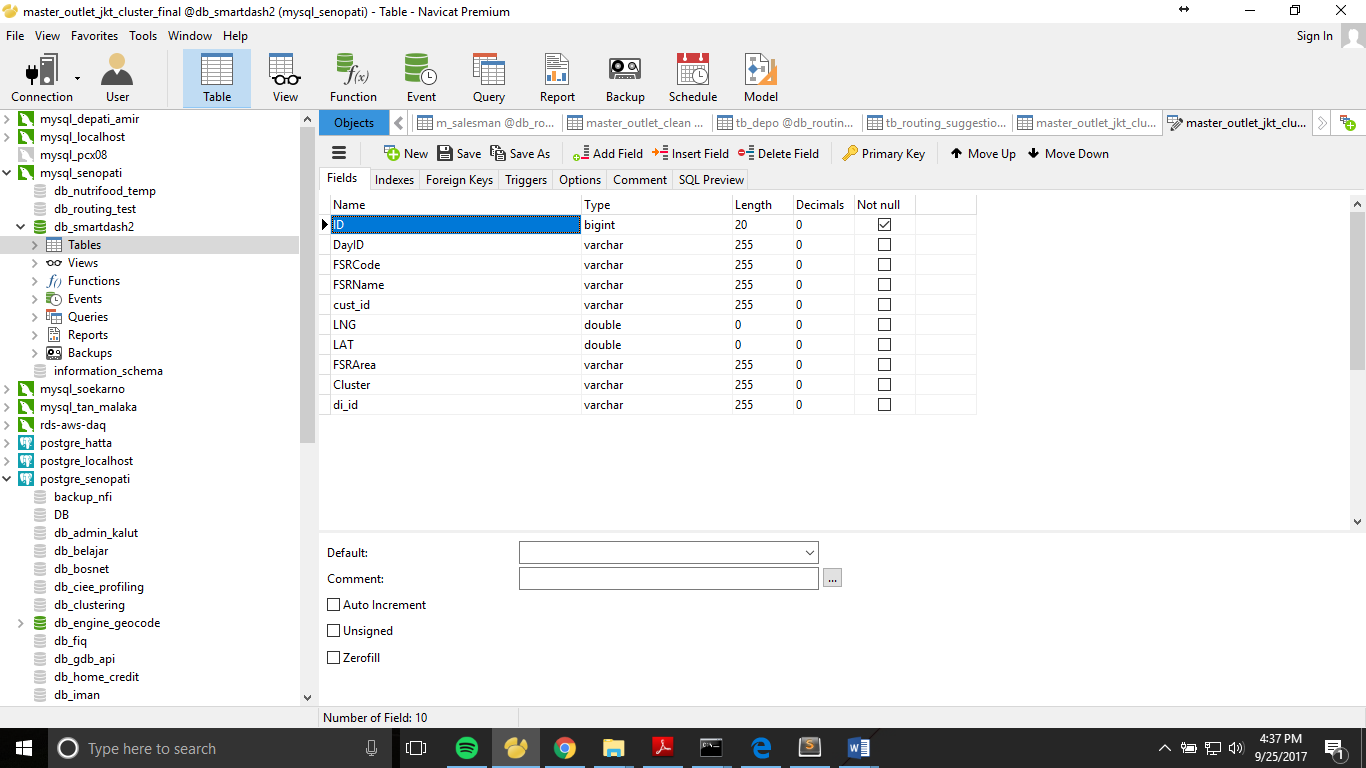
Namun struktur tabel diatas masih harus dinormalisasi sehingga dapat langsung di consume oleh Sistem Back End SmartDash. Struktur tabel final yang dibutuhkan untuk Back End SmartDash terdapat di server production SmartDash2:

**>> Server: MySQL Senopati**

**>> Database: db\_smartdash2**

**>> Tabel: master\_outlet\_jkt\_cluster\_final**

Dengan skema tabel seperti gambar dibawah ini:



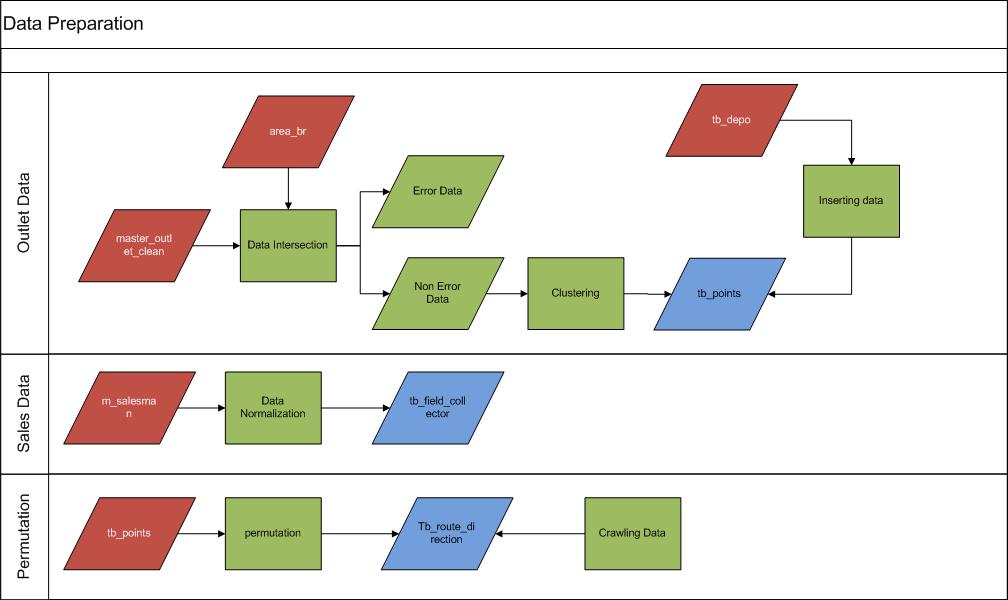
Untuk melakukan proses normalisasi data sehingga sesuai dengan struktur diatas dapat dilakukan dengan menjalankan query yang terdapat pada file:

**normalisasi\_output.sql**

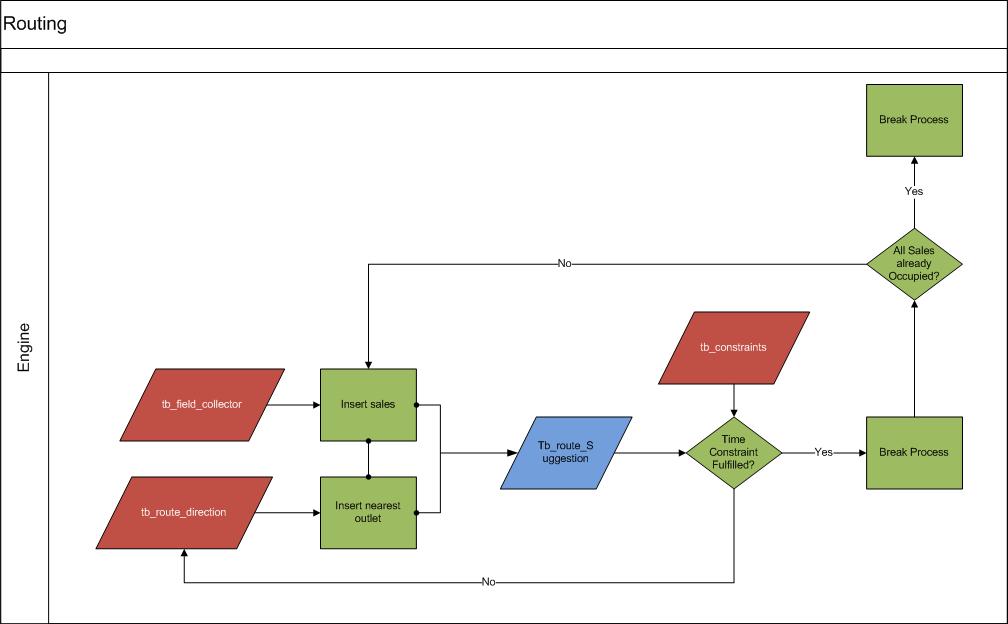
# 4. Engine Architecture

## 4.1. Process Flow

1. Data Preparation and Permutation



1. Routing



Keterangan:

: Input

: Process

: Output