SSD designed for Power BI and Tableau

[v0.7 | Draft | 14/11/23]

The SSD definition and design has worked towards minimised storage requirements, but where needed has applied limited de-normalisation in order to simplify some queries and improve performance; specifically using flattened tables with pre-calculated fields to maintain/reduce reporting overheads. PowerBI is one of several powerful tools that local authorities increasingly look to for actionable insights; one that we therefore anticipate being harnessed in maximising potential of SSD structured data. This document contains a summarised overview towards relevant discussion points or concerns regarding the powerful combination of SSD and such as PowerBI and Tableau.

Contents

Indexing:

Indexing on FK and fields expected to underpin key use cases, e.g. date fields.

Data Types:

Used appropriate data types to reduce storage and improve performance. Avoided use of string type for dates given that a great deal of the SSD is based around date types.

Aggregated Tables:

Pre-aggregate data at various levels (daily, monthly, etc.) to reduce the amount of data processed during analysis.

Partitioning:

Though not yet implemented, we're looking to options regarding partitioning the ssd_person and other larger tables based on frequently queried columns (e.g. into date ranges, 1yr, etc) to improve query efficiency.

Specific to Power BI

Star Schema:

We know that Power BI can perform better/well with a star schema design. This involves having a central fact table linked to dimension tables, and we have achieved this around the ssd_person object.

Avoid Complex Relationships:

We have designed around the avoidance of many to many relations and in combination with the star schema, tried to maintain simple relationships to aid performance.

Specific to Tableau

Extracts:

The use of Tableau created data extracts, optimised for speed is not required as the SSD is it's own non-live extract.

Hierarchies and Joins:

We have actively avoided unnecessary joins; in some cases creating an un-normalised structure to better achieve this.

Calculated Fields:

Where stakeholders considered calculated fields relevant, the processing is already completed within the extract structure to avoid performance hits within such as Tableau.

General Best Practices

Data Cleaning:

An underlying principal/aim of the SSD was to avoid less relevant data points from the core reporting CMS, and to a lesser extent to preprocess data. Within the initial SSD deployment, we have applied only limited cleaning (Date definitions, field size restrictions and restructuring (e.g into JSON objects). As the pilot group expands, we will be in a better position to assess the need for further underlying standardisation in data quality. Where data cleaning is considered of benefit to the wider community, it will be considered and introduced on a case-by-case basis.

Incremental Refreshes| Data Refresh Periods:

We are looking towards the use of incremental refreshes rather than full refreshes to update the data within the SSD. Some local authorities have requested the SSD as a set of live views, we are currently testing this and welcome further conversations around this direction.

Documentation:

Proper documentation of the data model, relationships, and transformations is crucial for maintenance and future optimisation. Our front-end page(s) are maintained in-line with any data or structural changes, being concurrently refreshed.

data-to-insight.github.io/ssd-data-model