# Semantic and ontologies of data sets along a data production process

**Michele K. Riccio[1], Mauro Scanu[2]**

[1]*Italian National Institute of Statistics – Istat, Rome, Italy*

[2] *Italian National Institute of Statistics – Istat, Rome, Italy*

## Abstract

National Statistical Institutes (NSI) organize data production processes according to models, as GSBPM, that represent all the relevant steps. Some of these steps are characterized by the presence of data and their transformation. To achieve the purposes of each data production step, NSIs need different concepts and semantic data structures. Hence, representation of data transformations for each step seems necessary to achieve lineage. Our goal is to represent these concepts and data structures by a meta ontology for each data production step. To describe data transformation we combine meta ontologies, SparQL queries and data access based on ontologies. By this way we get a formal representation - machine readable - of semantic structures and data transformations. The purpose of this paper is to investigate the corresponding semantic structures that characterize each milestone data set in the data production process and to represent it by means of ontologies.

**Keywords:** Meta Ontology, GSIM, Lineage, metadata, aggregated data

## 1. Introduction

Nowadays, National Statistical Institutes (NSI) gather many sources of information (traditional surveys, possibly based on samples, administrative archives, etc). The appropriate selection of (possibly integrated) data from the different sources is the starting place of many statistical productions. The activity of a statistician in an NSI can be organized in different process steps, from data collection up to the production and dissemination of statistical data. Here we identify the main steps that seem to be extremely important for metadata management. More precisely, we investigate those process steps where metadata can be formalized according to specific semantic structures that make use of metadata in previous steps: these process steps and the management of the relative metadata are consequently extremely important in order to preserve the meaning of data throughout a statistical program.

We propose to organize metadata in these steps in appropriate ontologies that make use as much as possible of the conceptualization available in the Generic Statistical Information Model (UNECE 2023). This formalization was firstly proposed in Scanu *et al* (2022) for data production processes on registers. Here, we give more details on the semantic structure of some of the main concepts to be used in the different process steps.

## 2. The main steps of transformation along a statistical process in a NSI

The organization of the activities of a statistical program follows traditional phases as described in the Generic Statistical Business Process Model (GSBPM). Along the process we identify some “milestone” data sets, where metadata can be described by specific semantic structures that make use of metadata of data sets in previous process phases.

1. A statistician starts its process with some input data (it can be collected by a traditional survey, or an administrative archive, or a register, or an integration of already available data sources). From now on, we consider as input a register, given that in Istat all registers are designed by ontologies. Its first activity is the selection from the input data of the features of the data set that the statistician needs to analyse in order to get the desired output. In order to do this, statisticians create a “design matrix” from its input, i.e. the typical starting point of any statistical analysis consisting of a rectangle of microdata with units as rows, variables as columns, and the set of rows that is either a population or representative of a population. In our approach the Design Matrix concepts will be selected from a set of concepts represented by an ontology, so that there are two milestone to consider: the Input data (not necessarily a data set, here we focus on a register), and the Design Matrix, representing the micro data set to be worked by the statistician in order to get its output.
2. Apart possible adjustments of the initial design matrix (in order to check and impute data, by adding transformed variables and so on) that still has the form of a “unit data set”, the data structure does not change its main organization. A milestone where a metadata structure substantially changes happens when the first statistical products (in terms of aggregates) are computed directly from the validated data set. Hence, the third milestone is a dimensional data set containing statistical aggregates. The measure associated with this data set can be of two types: *i)* parameters or characteristics of the distribution of a univariate or multivariate distribution of variables computable from the unit data set: totals, percentages, conditional percentages, means, medians and quartiles, interquartile ranges, Gini indices, are typical examples of this activity; *ii)* indicators that compare statistics along time or space, between populations or between variables (as ratios, percentage changes, index numbers). What is actually measured in both these aggregates has a semantic structure that makes advantage of metadata available from the data sets created in the previous process steps. These elements characterize mainly the “measure component” of a dimensional data structure and that need specific semantic structures that, again, affect the “measure component” of a dimensional data structure.

In the next sections we investigate the corresponding semantic structures that characterize each milestone.

### 2.1. Design matrix

Statisticians usually start their statistical activities on data with a very simplified representation. They use to refer to units in a population while observing variables on the units themselves. The result of this starting point is a rectangular matrix of data, usually named Design Matrix, with as many rows as the units in the population and as many columns as the variables observed on the units. The generic element in the i-th row and j-th column is the datum corresponding to the observation of variable j on unit i. Design matrices can be built selecting the necessary concepts from a conceptual framework of an integrated register on which statistics are computed: each register conceptual framework in Istat is designed by means of an ontology. Hence, the design matrix is built by selecting the necessary concepts in the Register Ontology. These are the main aspects to consider for setting up a design matrix:

1. *Unit Type* is selected from one of the concepts in the register ontology. For instance, persons or enterprises.
2. *Population*: a population follows by filtering some concepts on the units: time, territory and possibly other characteristics. For instance, from the unit type persons we end up with the population of “residents in Italy on the 1 of January 2022” by fixing residence and country and time of residence. The single elements in the register filtered by the query are the *units* of the population: each unit refers to a row in the design matrix.
3. *Variables*: On the ontology, it is possible to find variables among the attributes of the class chosen as Unit Type or among attributes of classes linked with the Unit Type class following a directional path along roles with cardinality One-to-Many or One-to-One (but not Many-to-One or Many-to-Many). For instance, sex can be an attribute of person and consequently a variable. Another variable on person can be derived from the One-to-Many relationship of person with household, deriving the variable “number of household components”.

### 2.2 Aggregate data

Statisticians analyse data that refer to units in order to compute information on groups of units appropriately identified by some common characteristics: populations. The statistical action consists in summarizing data from unit data sets into tables of aggregates that describe how variables are distributed in a population. NSIs use a set of specific summaries of the variable distribution: as already said, we identify these statistical products as “aggregated values”, because they are computed aggregating what has been observed on sets of units. The

aggregated values represent the main features of how the variables distribute over all the population of interest. The way to derive these aggregated values depends on the nature of the variables themselves. Usually, NSIs do not treat the case of more than one numeric variable, anyway it is not difficult to extend the same approach also to the numeric multivariate case.

In this context, what is measured has a specific structure whose concepts are derived from the validated unit data set and the desirable synthesis of what observed in the unit data set in terms of aggregated data of the population. Hence, what is measured should be explained by filling in the following information in GSIM terms:

- *Process method*: this concepts describe which synthesys of the microdata has been taken into account (mean, median, variance, percentage, total, …)
- *Universe*: once a specific territory and time is selected, this concept reveals over which set of unit the measure refers to;
- *Variable*(s): this concept describes on which variable(s) the aggregation method is applied. If the value domain is described and numeric, usually NSIs work on a single variable and other variables are only with an enumerated value domain and specify better subsets of populations over which the computation is done (statistician call these categorical variables as conditional variables); if the variables specified are only categorical, they can be more than one (e.g. this happens with percentages).

These concepts are mandatory for the measure of aggregate values that are synthetic values of a single distribution: if any one of the previous elements is missing, the measure is not well represented. As an example, the measure “Household average income” is represented by a process method (average), a universe consisting of resident household (populations are then derived by selecting territory can be Italy, the North west of Italy, Lombardy,…, and reference times can be the year 2020, 2021, …), and a single variable with described numeric value domain (income).In the case of aggregate data in terms of comparison indicators, apart the typical scientific names given to these indicators, it is important to show their input, i.e. which aggregates are compared, and by means of which comparison tool. This becomes a precious source of meta-information in order to give the meaning of these indicators. Hence, the main GSIM concepts to consider are:

- *Process method*: in this case ratio, index number, differences, densities, etc
- *Core Process input*: in this case, which aggregates are considered for the comparison.

## 3. Ontologies

To model statistical requirements and data transformations, in line with what represented in Section 2, we can distinguish three main Data Description Layers representing respectively data available as input (Input Data Layer, e.g. obtained in the Collect Phase), microdata to be worked by statistician (Design Matrix Layer, e.g. a validated dataset) and macrodata (Macrodata Layer, i.e. desired output of statistician). In our approach, each Layer is represented by an ontology:

1) **Input Data Layer**: For this layer, we describe by a **Domain Ontology** the concepts as available at the beginning of the process (as already said, we consider for simplicity a register because it is already designed as an ontology, so that we can name this layer as a Register data layer; otherwise, the concepts in the input data should be organized as an ontology). This Ontology is a conceptual model of data used as input for the Process Phase. A Domain Ontology does not describe only data actually collected or available, but it describes concepts and the expected relations: for instance, we will define a self-relation parent-of for class Person independently if data contains records with parent relation. Therefore, a Domain Ontology describes concepts and rules among concepts that we believe valid for that domain. In Figure 1 an example of Domain Ontology to analyse smoker habits of persons, distinguished for educational level and date of birth.
2) **Design Matrix Layer**: In this layer we have to describe statistical choices for specific statistical activities and, meanwhile, we have to define the structure of one Design Matrix: which is the Unit Type to be analysed (or which Universe or Population), what are Variables and so on. For this purpose, we will use a meta ontology called Design Matrix Ontology, where many concepts are inherited from GSIM model. To achieve this goal, for a specific Design Matrix, we instantiate the Design Matrix Ontology with concepts of Domain Ontology. Design Matrix meta ontology is general and the same for every Design Matrix, its instantiation is specific for only one Design Matrix. In Figure 2 an example of Design Matrix Ontology.
3) **Macrodata Layer:** In this layer we have to describe transformations and aggregation of data, measures and indicators created to summarize microdata. Similarly with Design Matrix Layer we can introduce a meta ontology to represent these concepts starting from Design Matrix Ontology and Domain ontologies. About Aggregate Data representation refer to: Lembo, D. et Al. (2023).

To understand how to realize Data Description Layers above we have to focus on some technological aspects of ontologies.

Figure 1: Very simple example of Domain Ontology

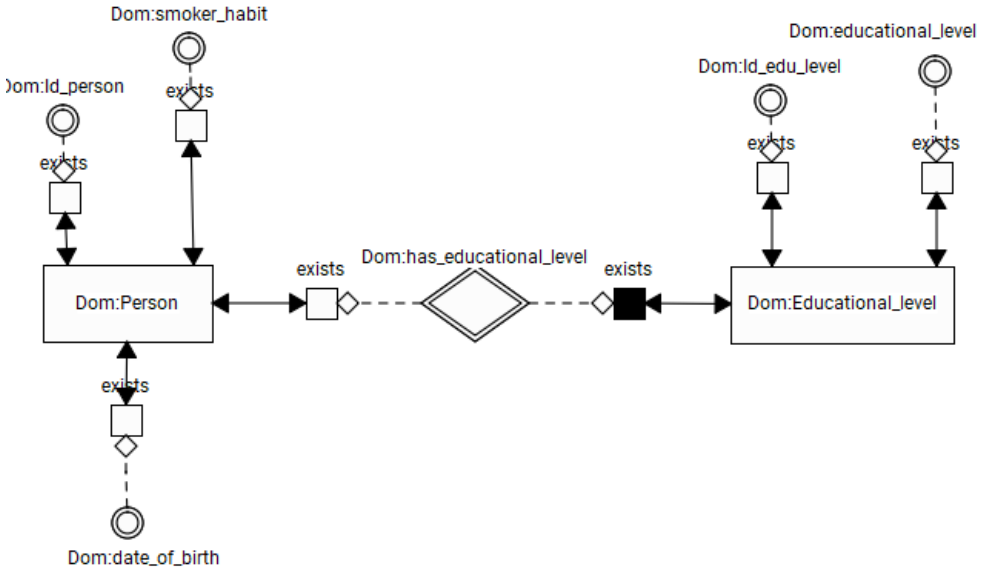
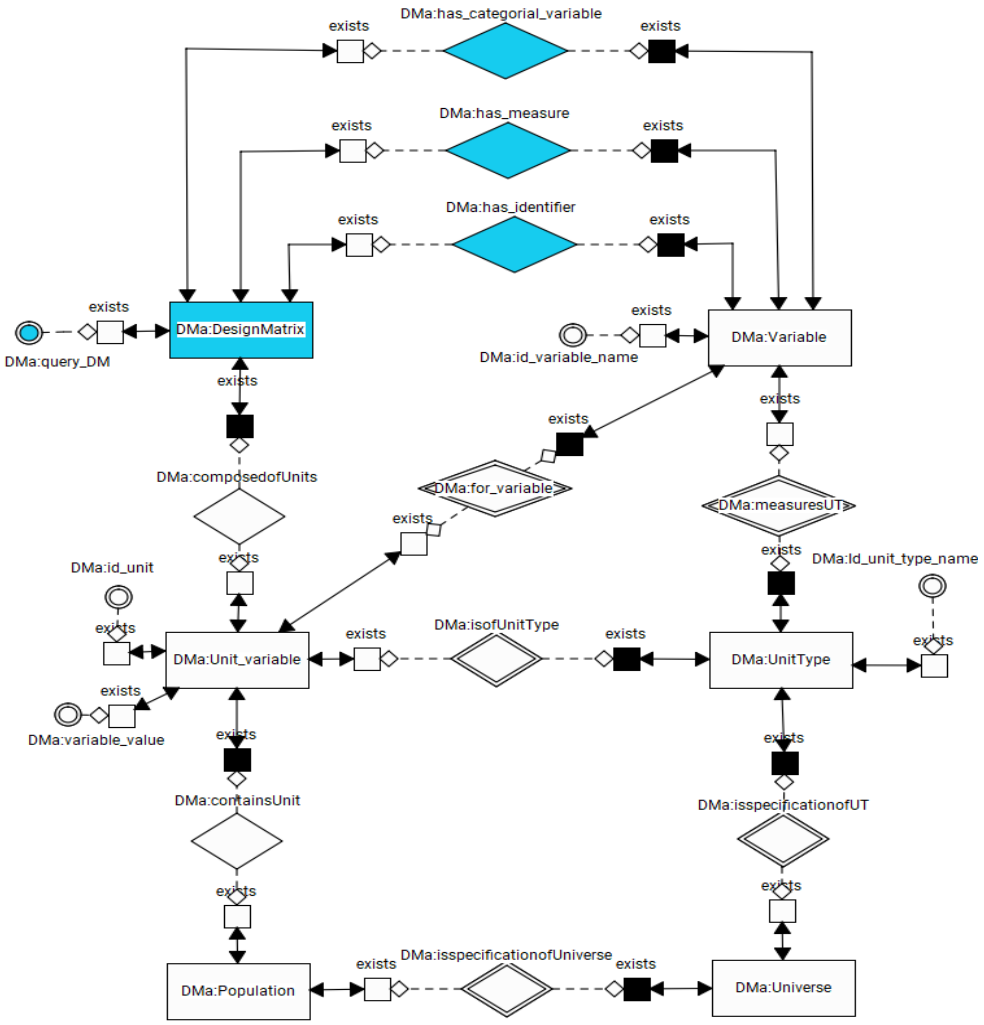



Figure 2: minimal example of a Design Matrix meta ontology:



### 3.1 Accessing Input Data trough Domain Ontology by OBDA approach

Ontology Based Data Access (OBDA) approach allows to access data through ontologies. By this way it is possible to access and to get out instances of Input Data simply querying the Domain Ontology with SparQL. We suppose Input Data are stored in a relational database (for instance data of a statistical registry), on the other side we have defined the Domain Ontology of Input Data (a Registry Ontology). By OBDA technology we can link objects in the ontology (classes, attributes, roles) to data using sql queries. On OBDA refer to: Calvanese, D et Al. (2011). For each ontology entity (classes and roles) we define a sql query to map entity to data, by this way we can consider every ontology entity as a view over relational data. In this way, when we query the ontology using SparQL language we can get back records of data coming from relational data base. It is not necessary to duplicate Input Data in a RDF triple store.

### 3.2 Populating Design Matrix by OBDA approach

The Design Matrix Ontology is not only a tool to describe and document assumptions on a statistical process.

When we combine this ontology with the OBDA approach, then it is possible to populate data table of Design Matrix with data stored in Registers (or any input data stored in a relational database).

To achieve our goals we have extend ontology constructs. We introduce Ontological Views written in SparQL: An Ontological View is a special ontology class, extended with a special attribute Query. Each attribute of View has to correspond, one to one, to a target variable of the Query. Each instance of an Ontological View contains a different SparQL query and any instance represents the set of records in output from this query.

Roles linked with Ontological Views are not actually roles, but they represent joins of the SparQL query with other classes. To distinguish these new kind of objects, they are blue coloured in Figure 2.

Let us suppose metadata catalogue is stored in a relational database with tables (in brackets field names): *Unit_variable*(Id_unit, id_variable, variable_value, Id_unit_type_name), *UnitType*(id_unit_type_name), *Variable*(id_variable_name, id_unit_type_name ). Fields beginning with Id_ are identifiers for entities.

The Unit_variable class represents values assumed by a specific variable for a specific unit therefore, it has to be mapped with the Union of values for couples [Id_unit, id_variable_name] extracted from any table in data domain containing units (Person, Enterprise, Family, …).

Other tables have been mapped with normal classes and roles of Design Matrix Ontology by simple SQL queries as Select * from .

Instead Query_DM of DesignMatrix View has to define statistical requirements of our design matrix. To explain how to do it, we use our example: we have to analyse smoker habits of persons, distinguished for educational level and date of birth.

Design matrix columns have to be: Id_person, date_of_birth, smoker_habit, educational_level. SparQL query for this design matrix has to filter by:

- Id_unit_type_name =“Person”
- has_categorial_variable.id_variable_name IN (“date_of_birth”, “smoker_habit”, “educational_level”)
- has_identifier_variable.id_variable_name=“Id_person”.

Query has to join classes: Unit_variable and UnitType by rule isofUnitType; Variable and UnitType by rule measuresUT; Unit_variable and Variable by rule for_variable. Special rules: has_categorial_variable, has_identifier are defining the meaning of variables exposed by the DesignMatrix view, these variables will be columns of output of DesignMatrix view.

Hence, all statistical requirements to build the Design Matrix are contained in the SparQL query. Requirements have been represented by a formal and machine readable way. Now it is possible to extract data records of Design Matrix executing queries in the views, because involved concepts are mapped to Domain ontology and this one is mapped to data records.

## References


Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A., Rodriguez-Muro, M., Rosati, R., Ruzzi, M. & Savo, D. F. (2011) The Mastro system for ontology-based data access, *Semantic Web J.* 2 (1) 43–53.

Lembo, D., Santarelli, V., Savo, D. F. & De Giacomo, G. (2022) Graphol: A graphical language for ontology modeling equivalent to OWL 2*, Future Internet 14.*

Lembo, D., Poggi, A., Radini, R., Riccio, M. K. & Santarelli, V. (2023). A Knowledge Representation Approach for Modeling Aggregates: A case study at ISTAT. Available at: https://www.ital-ia2023.it/submission/61/paper

Scanu, M., Scannapieco, M., Tosco, L., Bianco, A. M., & Riccio, M. K. (2022). Metadata for statistical processes on registers: how to organize facts with GSIM, in: *Proceedings of the First Workshop on Methodologies for Official Statistics*, ISTAT, Roma, Italy. Avaliable at: https://www.istat.it/it/archivio/277812

United Nations Economic Commission for Europe – UNECE (2023). *Generic Statistical Information Model (GSIM): Specification*, Geneva, Switzerland: UN-ECE. URL: https://statswiki.unece.org/display/gsim/GSIM+Specification