

# The DPM metamodel detail

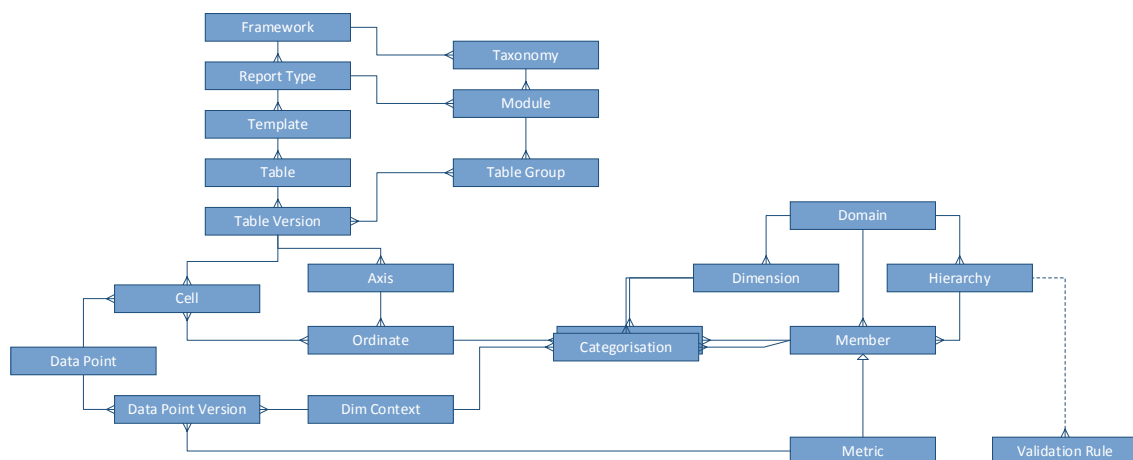
The EBA process for developing the DPM is supported by interacting tools that are used by policy experts to manage the database data dictionary.

The DPM database is designed as a metamodel, whose objects represent the data point modelling concepts (e.g. *Module*, *Template*, *Data point*, *Dimension*, *Member*,...) and relationships. Thus, the concrete DPM elements are stored as metadata content describing the concrete instances of these objects (e.g. Module: 'FinRep', Template: 'F01.00', cell: 'F01r30c10', Dimension: 'Base', Member: 'Assets') and their relationships.

New EBA Technical Standards and Guidelines usually add new content and introduce changes as well. This gives rise to new versions of templates with a different structure, and new versions of data points with a different categorisation. Other technical amendments resulting from the Single Rulebook Q&A process lead to dimensional modelling corrections and enhancements.

The database fully supports the DPM change management process, keeping track of the history of templates and data point versions across successive ITS releases. This is a fundamental feature of a data dictionary supporting data warehousing and time series analysis.

The complete structure of the DPM database includes a high number of tables and relationships. While it can be difficult to understand the whole structure at first, the initial approach to the DPM can be facilitated by focusing just on the core parts of the meta-model, which include only a relatively small number of concepts:



In the following sections, the core meta-model will be broken down into different subject areas for a more detailed presentation.

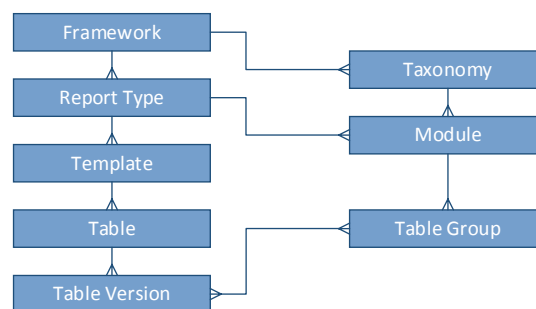
## Report structures

The reporting requirements are organised in Frameworks according to different regulatory areas (e.g. COREP for own funds, FINREP for financial reporting). Frameworks are further broken down into Report Types (*a.k.a.* Conceptual Modules), covering more specific regulatory subjects (e.g. Own Funds, Large Exposures, Liquidity), which define the comprehensive set of data to be submitted in each report file instance.

Each report type consists of a set of Templates that are designed by regulation experts to define the concrete reporting requirements (*i.e.* *what* should be reported). The Templates are also used as the unit for defining the reporting obligations (*i.e.* *who* should report *what*), and, therefore, are a very central element of the Technical Standards.

While in most cases Templates can be modelled without a problem, sometimes, they may have to be split into different Tables due to a more complicated structure. A Table is either directly equivalent to a Template (e.g. Template C 01.00 becoming Table C 01.00), or is a part of it (e.g. Template C 09.01 originating Tables C 09.01.a and C 09.01.b).

Across the successive ITS releases, templates might remain unchanged or they may have to undergo alterations due to restructuring or different reporting requirements. The history of different versions of a template, or part of a template, is kept in the DPM as Table Versions, with an associated validity period (*from date, to date*).



Each Taxonomy represents a version of a Framework for a certain DPM release. Similarly, each Module (*a.k.a.* “entry point” in XBRL terminology) represents a version of a Report Type (*a.k.a.* “Conceptual Module”) in a DPM release. Therefore, both Taxonomies and Modules have also a validity period, and since a report instance refers to a specific point in time (reference date), it should be produced according to the correct Module, or otherwise it will be invalid.

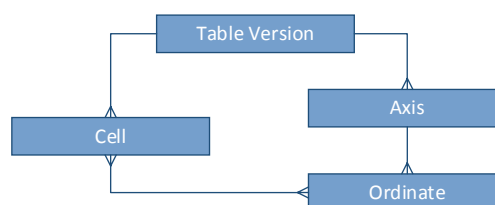
Within each report type, Templates are organised in intermediate groups for different subject areas (e.g. Capital Adequacy, Credit Risk, and Market Risk). Both Report Types and Tables are standalone versioned, and, therefore, the resulting many-to-many relationship was established in the DPM between Table Versions and Modules, via Table Groups.

## Templates layout

Templates consist of a grid of Cells organised in rows, columns, and sheets. Their layout is defined in the DPM by means of an Axis and Axis Ordinate. Each Axis represent one of three possible directions (“X” for columns, “Y” for rows, and “Z” for sheets), and Ordinates represent the individual rows, columns, or sheets.

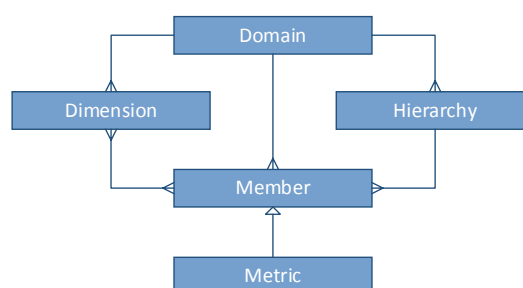
There are Cells at the intersection of rows, columns, and sheets. Therefore, the position of each Cell can be defined by three ordinates on different axes — or two ordinates only, for single-sheet templates that do not require a z-axis.

Because of the aforementioned complicated and changing layout of templates, the DPM defines Axis and Ordinates for each Table Version and Cells are generated in the database by crossing the Ordinates of the axes of Table Versions.



## Core vocabulary

The core vocabulary contains all the metadata elements used in the data point modelling to give a precise meaning to each individual cell in the framework. These are classified as Domains, Dimensions, Members, Hierarchies, and Metrics.



Domains and Dimensions are similar concepts representing categories of data of the same nature (e.g. “Type of Risk”), while Members are the actual instances of those categories (e.g. “Credit Risk”). The difference between Domains and Dimensions is that the first is just used to group Members of the same semantic nature, while the later takes into account the role played by the members play in the model, which means that multiple Dimensions (e.g. “Currency of the collateral”, “Currency of the exposure”) can take their Members from the same Domain (“Currency”).

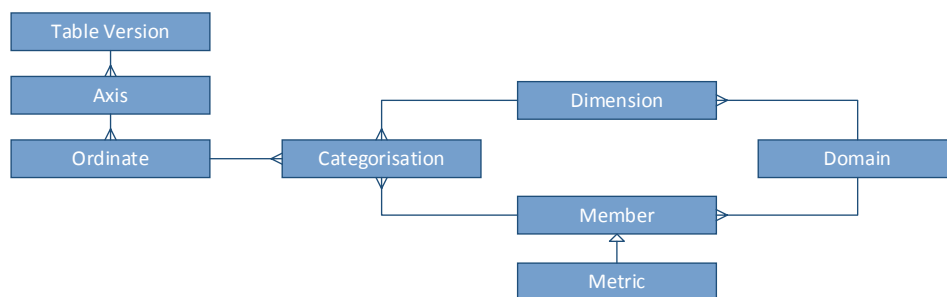
Some Domains in the DPM are "open", i.e. they do not have a predefined list of Members because it is not possible to enumerate all their possible instances (e.g. “Legal entity”).

Hierarchies are used to organise Members of a Domain and indicate how Members relate to each other. They can also define the aggregations from lower to upper levels in the Hierarchy. Hierarchies are sometimes simply used to define subsets of Members (subdomains).

Metrics are the Members of a special Dimension that must be applied to all Data Points and also provide additional meaning to the Data Points by being associated to a Data Type, and having a “stock”/“flow” property.

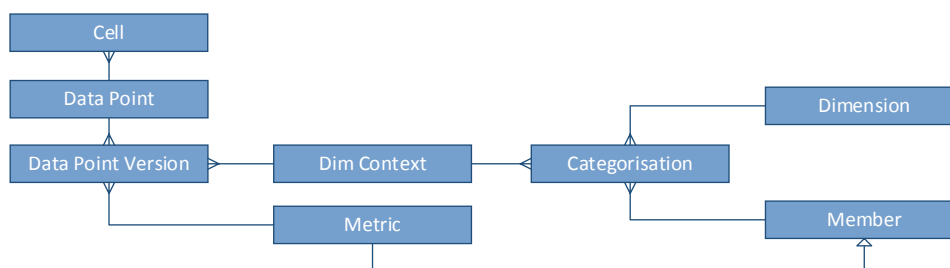
## Dimensional modeling

The regulation experts model each Template by assigning a set of Dimension/Member pairs to each Ordinate (row, column, or sheet) that fully categorises it.



The manual dimensional categorisation of Ordinates is the input to an automated process that generates the full categorisation of each individual Cell, starting from the categorisation of the Ordinates of the Cell in the table Axis. Multiple validations apply to check the quality model, and the process might undergo a number of iterations before the model is considered correct from a formal technical perspective.

Each table Cell (not considering the grey-shaded ones) corresponds to a single Data Point. There are, however, some Data Points represented in multiple table Cells. In the latter case, the twin Cells refer to the same data, and should, therefore, share the same categorisation in the DPM.



A Data Point represents an elementary reporting requirement item. Because the categorisation of a Data Point might change from one DPM version to another (due to improved data modelling), new Data Point Versions are created as a result of a different categorisation. However, in order to preserve the traceability of data across time (and enable time series analysis), all versions of the

same data point, which are valid for consecutive periods of time, are linked to the single Data Point to indicate that all relate to the same business concept whose meaning has not changed.

Because of its special role, Metrics are separated from the full categorisation of data point versions. Some Data Points might have the same categorisation except of the Metric (equivalent to the XBRL “primary item”). In the latter case, they share the same Dimensional Context (equivalent to the XBRL “scenario”).

## Validation rules

Data Validation Rules are also represented in the Data Point Model database in a semi-structured format able to support automatic processing. Some Validation Rules are explicitly (manually) defined by business experts, while others are first derived (automatically) from implicit rules in members’ Hierarchies. Business experts should then whether these rules are applicable or not.

## “Open” tables representation

The model contains two major types of Tables:

- The “closed” tables – where each Axis of the tables has all its required values explicitly listed, and therefore the precise size of the reported table is known, and
- The “open” tables - where one or more Axes are “open” (allowing a variable number of entries, chosen either from a restricted list (e.g. “Counterparty sectors”), or of a particular type (e.g. “Integer”).

Closed Tables fall into two main types: those with X-axes and Y-axes only (which are simply plain Tables), and those that also specify a Z-axis. Such tables are also made up of multiple sheets, each with a complete copy of the Table, one for each Ordinate on the Z-axis.

Open Tables also fall into two types: list tables and tables with optional sheets. Tables with optional sheets are simply normal Tables with an open z-axis, for which a copy of the table should be completed for each applicable value. The open Z-axis for these Tables typically allows values chosen from an explicit Domain (e.g. a sheet per currency or per country where significant exposures are present etc.).

List tables are slightly more complex as they represent a Table where a series of rows with identical columns must be entered, one for each reported item of a particular kind (e.g. a row per entity in a banking group, a row per security held, a row per transaction etc.). These Tables have an open Y-axis.

For both open Y-axis and open Y-axis tables, the unknown number of entries on the Axis is represented in the DPM database by an Axis Ordinate with an ordinate code of “999”.

For open Y-axis tables, the anticipated rendering involves a special column, into which the identifier of each particular row is entered. In many cases, this identifier will simply be a meaningless number (e.g. a line number). In the database, this is represented by the row Axis Ordinate (“999”) associated with an open dimension, which describes the nature of this identifier (such as the code of the security or obligor grade), and by a member with ID 999 (to indicate that the entire row is associated with a unknown value from the dimension).