NOVA IMS Information Management School

MEGI

Mestrado em Estatística e Gestão de Informação
Master Program in Statistics and Information Management

Development of the BIRD: A metadata modelling approach for the purpose of harmonising supervisory reporting at the European Central Bank

Directorate of General Statistics: Master and Metadata

Mark Peter Connell

Internship report presented as partial requirement for obtaining the Master's degree in Statistics and Information Management

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

DEVELOPMENT OF THE BIRD: A METADATA MODELLING APPROACH FOR THE PURPOSE OF HARMONISING SUPERVISORY REPORTING AT THE EUROPEAN CENTRAL BANK

Directorate of General Statistics: Master and Metadata

by

Mark Peter Connell

Internship report presented as partial requirement for obtaining the Master's degree in Statistics and Information Management, with a specialization in Information Analysis and Management

Advisor: *Professor Mauro Castelli*

DEDICATION

I would like to dedicate this work firstly to my parents, Sonia and Peter Connell for supporting my decision to further my studies overseas and have supported me throughout my time in Europe. Also, my girlfirend Fabiola Rothe who has been there for me since beginning my studies and the Rothe family, who welcomed me into their family and supported me during my time in Germany.

I would also like to dedicate this paper to those who have helped me academically, all the professors and administration staff of NOVA IMS, in particular Professor Castelli who supported me and have shown dedication in their teaching.

Lastly, my colleagues at both internships, Luminovo and the AMA division in the European Central Bank for granting me these opportunities to facilitate my learning and allowing me to complete this report.

ABSTRACT

The work presented is a report documenting the work completed during an Internship at the European Central Bank (ECB), located in Frankfurt Germany from the 15th March 2019 – 15th March 2020. The internship took place in the Directorate of General Statistics (DG-S), specifically in the Master and Metadata section of the Analytical Credit and Master data division (MAM). It will be a continuation of the ECB Internal Banks' Integrated Reporting Dictionary (BIRD) project as well as management of ECB's centralised metadata repository, known as the Single Data Dictionary (SDD). The purpose of the dictionary and BIRD Project is to provide the banks with a harmonized data model that describes precisely the data that should be extracted from the banks' internal IT systems to derive reports demanded by supervisory authorities, like the ECB.

In this report, I will provide a basis for understanding the work undertaken in the team, focussing of the technical aspect of relational database modelling and metadata repositories and their role in big data analytical processing systems, current reporting requirements and methods used by the central banking institutions, which coincide with the processes set out by the European Banking Authority (EBA). This report will also provide an in-depth look into the structure of the database, as well as the principles followed to create the data model. It will also document the process of how the SDD is maintained and updated to meet changing needs. The report also includes the process undertaken by the BIRD team and supporting members on the banking community to introduce new reporting frameworks into the data model. During this period, the framework for the Financial Reporting (FinRep) standards was included, through a collaborative effort between banking representatives and the master and metadata team.

Keywords

Statistics, Supervisory reporting, metadata, data dictionary, data modelling, dimensional modelling.

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ECB	European Central Bank
SDD	Single Data Dictionary
BIRD	Banks’ Integrated Reporting Dictionary
EBA	European Banking Authority
ITS	Implementing Technical Standards
IFRS	International Financial Reporting Standards
NGAAP	National Generally Accepted Accounting Principles
DBMS	Database Management System
SDMX	Statistical Data and Metadata exchange
BSI	Balance Sheet Items
DG-S	Directorate of General Statistics
MAM	Master and Metadata

1 Introduction

The European Central Bank was founded on June 1st, 1998, as an effect of the Treaty of Amsterdam which served as an update to the Treaty of the European Union, in 1993. The European Central Bank replaced the European Monetary Institute (EMI) which was formed towards the end of the Economic and Monetary Union (EMU), used to manage the adoption of the Euro as a currency. The EMI prepared for the creation of ECB and European System of Central Banks (ESCB). The ESCB includes the ECB and the national central banks of all the EU member states, including those who have not yet adopted the Euro as their currency.

The primary responsibilities of the ECB are linked to its main goal of price stability is formulating monetary policy. This involves making decisions about monetary objectives, key interest rates, the supply of reserves in the Eurosystem and establishing guidelines for implementing those decisions. Decisions taken in these regards are supported by research and data provided by the different directorates, within the ECB, such as the directorates in General Statistics.

The Directorate General Statistics (DG-S) is responsible for developing, compiling and disseminating the statistics, the statistical indicators and the related metadata as well as supervisory data and indicators and the related statistical indicators required for that are needed for the ECB's monetary policy, banking supervision and the other ECB/Eurosystem tasks.

The main areas of the ECB’s statistical activity are:

1. Monetary, financial institutions and financial markets statistics, including interest rates and securities statistics
2. Micro prudential data related to banking supervision
3. External statistics covering balance of payments of the euro area, international investment position of the euro area, Eurosystem's international reserves and effective exchange rates of the euro; the Centralised Securities Database (CSDB) and securities settlements statistics
4. Financial accounts for the euro area (including government finance statistics); integrated quarterly non-financial accounts for institutional sectors and survey data of households and enterprises.

DG-S is also responsible for the application of tools and procedures to aid the ECB's collection, management and dissemination of data. It develops, manages and applies the data collection tools, databases and quantitative risk reports for supervisory purposes.

Supervisory responsibilities of the ECB are undertaken by the Supervisory Board, to ensure the robustness of the banking system. The European Central Bank carries out this function through the Single Supervisory Mechanism (SSM) that comprises the ECB and competent national authorities in the member countries. As of the 1st January 2019. The ECB directly supervises 119 significant banks in the Euro area[1].

European banking supervision aims to contribute to the safety and soundness of credit institutions and the stability of the financial system by ensuring that banking supervision across the euro area is of a high standard and is consistently applied to all banks.

Since the occurrence of the financial crisis in 2008, national and international supervisory authorities have implemented an expanding amount of laws and regulations which have in turn increased the reporting obligations of financial institutions considerably. This included some entirely new provisions such as the Liquidity Coverage Ratio and the Net Stable Funding Ratio, which aim to determine the short-term resilience of banks' liquidity profiles and ensuring the stability of funding for the banks' assets and off-balance sheet items (BIS, 2018).

The provisions of the EU frameworks are supplemented in some cases by separate national reporting requirements. The vast amount of current mandatory reports creates a considerable challenge to both banks producing the reports, as well as the supervisory authorities both in terms of consistency, scope and quality of the collated data as well as the transmission and assessment of that data. (EY, 2019).

These reporting obligations are specified in different legal frameworks including the ECB statistical regulations on balance sheet items (BSI) and interest rates (MIR) of monetary financial institutions (MFIs), the sectoral module of Securities Holdings Statistics (SHS-S), granular credit and credit risk (AnaCredit), as well as the European Banking Authority's (EBA's) Implementing Technical Standards (ITS) reports (FINREP and COREP) and the ECB Regulation on reporting of supervisory financial information.

1.1 ECB Response to Lack of Data Harmonisation

One of the main responses to the latest financial crisis was the establishment of a single rulebook in Europe aimed at ensuring a robust and uniform regulatory framework to facilitate the functioning of the internal market and prevent regulatory arbitrage opportunities. The main legislative act to implement the single rulebook is the Capital Requirements Regulation (CRR) which sets out prudential requirements for institutions. The CRR includes a number of articles with specific mandates for the EBA to develop draft Implementing Technical Standards (ITS) relating to supervisory reporting requirements. These ITS are part of the single rulebook enhancing regulatory harmonisation in Europe, with the particular aim of specifying uniform formats, frequencies, dates of reporting, definitions and the IT solutions to be applied by credit institutions and investment firms in Europe. Uniform reporting requirements are necessary to ensure fair conditions of competition between comparable groups of credit institutions and investment firms and will lead to greater efficiency for institutions and greater convergence of supervisory practices and allow supervisors to assess risks consistently across the EU, thus enabling them to compare banks effectively and identify emerging systemic risks.

[1] Link

In 2013 the Statistics Committee[2] (STC) of the European System of Central Banks has investigated the possibility to promote an integrated approach to supervisory and statistical data.

In 2014 an STC Groupe de Réflexion (GRISS) recommended the development, in close collaboration with the banking industry, of a European "input approach" model which would provide a standardised design of the banks' internal data warehouses to support the data reporting to the authorities.

The report found that the establishment of the Single Supervisory Mechanism (SSM) in 2014 entailed greater responsibility than in the past for the provision of sound granular banking data. Therefore, a long term objective was established, proposed by GRISS in which the existing national statistical and supervisory information systems (mainly related to banking data) evolve into a set of coordinated systems, possibly acting as a joint European Information System, with the aim of governing, organising, managing and using information through common practices, methodologies, infrastructures and tools.

The group considered the benefits of a European Information System, which included

(a) increasing the quality of information (greater consistency of data, less necessity for ex post data reconciliations and development of highly accurate systems for cross-checking data quality);

(b) avoiding duplications in data requests and keeping the reporting burden of reporting agents manageable (less information to produce, fewer NCB/NCA points to contacts, higher consistency in the level of data quality requested, better planning of new statistical and supervisory reporting obligations, high comprehensive awareness by the authorities about the wide-ranging reporting obligations for reporting agents);

(c) providing more powerful services to users (faster reply to new data requests, richer information and improved opportunities for cross analysis, higher data sharing and easier use of different data sources); and

(d) reaping economies of scale in data management and in developing IT solutions supporting the statistical activities.

The goals outlined advocated for parallel integration efforts in two dimensions, cross-country (horizontal) and cross-domain (vertical) integration. Horizontal integration requires the long-term harmonisation, to the extent possible, of methodologies, processes and practices followed at the national level for data production. Vertical integration means that harmonisation cannot proceed by way of "stovepipes", but with the aim of managing different areas of statistical and supervision information as parts of a single system, paying attention to the overall process ranging from the identification of possible data sources to final uses, considering each piece of data as a multipurpose item of information (i.e. useable for different tasks and functions of the ESCB/SSM/NCBs/NCAs) and avoiding as far as possible any redundancy of data and indicators and any overlaps in requests to reporting agents.

---

[2] The STC established in 2013 a Groupe de Réflexion on the integration of statistical and supervisory data, with the mandate to develop a set of recommendations and proposals on how to best promote an integrated approach to supervisory and statistical data (Cf. Seventh ECB Conference on Statistics).

The ultimate goal established in the report was a comprehensive and harmonised common reporting framework for regular data transmission by banks to European NCBs/NCAs. This final goal should be planned on the basis of a step-by-step approach, which envisages, as a first step, the definition of an initial version of a harmonised reporting framework for NCBs' data collection from banks ("primary" reporting), covering BSI, MIR, SHS, AnaCredit, the needs of other statistics such as BOP and national accounts and the additional statistical requirements of the SSM, but not yet covering the EBA's ITS. Thereafter and on the basis of a feasibility and cost-effectiveness analysis, a roadmap can be drawn up for the adoption of the ECB's reporting framework for banks by all Eurosystem/SSM countries (which will in any case continue to be able to add specific requirements for national information needs). In parallel, the EBA and the ESCB/SSM should cooperate with the aim of moving, in a second step, towards a single and integrated European Reporting Framework (ERF), incorporating both the EBA's ITS and the ECB's reporting framework on the basis of an appropriate roadmap.

The report also stated that a prerequisite for pursuing effective data integration would be the availability of a complete, up-to-date description of the overall content of what is meant to be integrated, i.e. a reference catalogue of the Information System equipped with a very rich set of metadata. GRISS recommended as a medium-term objective the development of a common Statistical Data Dictionary (SDD) describing the data managed within the ESCB/SSM information system related to banks (including the evolving EBA's ITS), both at the individual and at the aggregated level, as well as, in the longer run, covering other relevant statistical domains.

As an extension onto this prerequisite, it was stated that the quality and timeliness of all data produced for the benefit of end-users strictly depend on the quality and timeliness of the input data received by NCBs and NCAs from the banks. The authorities have every interest therefore in acting as catalysts to bring about improvements in the banks' internal processes for reporting to them, while respecting the fact that such processes remain the banks' own responsibility. GRISS recommended that, in close collaboration with the banking industry and along the lines of the results already provided to the STC by the Joint Expert Group on Reconciliation of Credit Institution's Statistical and Supervisory Reporting Requirements (JEGR), a European "input approach" model should be developed for possibly organising in an integrated way the banks' internal processes for reporting to the authorities. Such a model should be based on a proper data dictionary (Banking Integrated Reporting Dictionary - BIRD), containing a logical description of the data and the transformation rules a bank might use to fulfil the authorities' reporting requirements, which needs to be set up and maintained over time.

1.2 Structure of report

This report aims to provide an overview of the current work undertaken by the ECB internal team managing the SDD and BIRD project. The work undergone as described in this report focuses on two of the major tasks:

1. Maintenance of the elements stored in the SDD and process of updating elements whilst maintaining integrity in the data model.
2. Continuing development of the BIRD metadata content, in particular the addition of the contents required to produce FinRep template 5.01

The report is broken down into the following sections:

Literature review

- Provides a basic overview of database systems, the role of metadata in managing databases and models for managing metadata

Supervisory Reporting

- Overview of the current supervisory reporting climate and requirements.
- Data models used for metadata in supervisory reporting
- Details the structure of the multidimensional data models used in the current production of supervisory reports

The Single Data Dictionary (SDD)

- Explains the ECB answer to data integration in supervisory reporting
- Outlines the SMCube data model, utilised by the SDD
- Describes the contents and data definition package in the SDD

Maintenance and Historisation in the SDD

- Principles and examples of how the SDD in managed
- Examples of maintenance and historisation tasks undertaken by the SDD team

Banks'Integrated Reporting Dictionary (BIRD)

- Introduction to the reporting dictionary and the objectives of the project
- Explains the BIRD model and the process of developing content in the dictionary

Project work: Adding FinRep template 5.01 into the BIRD

- Process of determining the metadata concepts to be added.
- Defining the transformation rules to convert data to produce template
- Application of mappings procedure and publication of the BIRD database on EC website

Conclusion and Future Work

2 Literature review

This section aims will provide a basic understanding about the information management concepts addressed in this report.

2.1 Database Systems

Information management systems have been documented to have been introduced around the 1960's, when file processing systems were used as information management systems. File processing systems usually consisted of a set of files and a collection of application programs. Permanent records are stored in the files, and application programs are used to update and query the files. Such application programs are generally developed individually to meet the needs of different groups of users.

File processing based information systems were discovered to contain a number of disadvantages. These included decreased efficiency in data storage as data was constantly duplicated among the files of different users. The lack of coordination between files belonging to different users often leads to a lack of data consistency (Abdelguerfi &Eskicioglu, 1997). File processing systems also lack the ability to share data, as well as reduced programming productivity, and increased program maintenance. File processing techniques, due to their inherent difficulties and lack of flexibility, have lost a great deal of their popularity and are being replaced by database management Systems (DBMSs).

A database management system is a software system that allows users to access and interact to data contained within a database. The objective of the DBMS is to provide a convenient and effective method of defining, storing and retrieving the information contained in the database (Gunjal, 2003, Prakash, 1991). The DBMS interfaces with the application programs, so that the data contained in the database can be used by multiple applications and users.

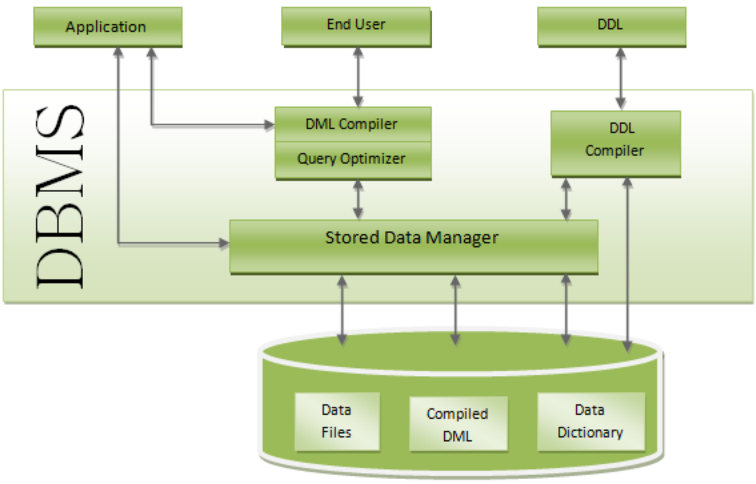


Figure 1 - Database Management System (Raza, 2018)

2.1.1 Characteristics of a Database

A number of characteristics distinguish the database approach from the much older approach of programming with files. Some of the main characteristics include:

Concurrent Use

A database system allows several users to access the database concurrently. Answering different questions from different users with the same data is a central aspect of an information system.

Structured and Described Data

A fundamental feature of the database approach is that the database systems does not only contain the data but also the complete definition and description of these data (Nebiker & Bliesch, 2005). These descriptions are basically details about the extent, the structure, the type and the format of all data and, additionally, the relationship between the data. The metadata is used to provide information about the data regardless of the application in which the data is being used.

Separation of Data and Applications

As just mentioned, the data within the database is described by metadata, also stored within the database. This means that the software applications related to the database do not need any knowledge about the physical data storage like encoding, format, storage place, etc. Applications therefore communicate with the DBMS by issuing commands and queries using a standardised language like SQL (Gitta, 2006). The access to the data and the metadata is entirely done by the DBMS. In this way all the applications can be totally separated from the data.

Data Integration

Instead of operating and querying databases in separate 'silos', database management systems allow for single user interfaces to manage databases that have physical or logical relationships.

Data Independence

One of the main purposes of a DBMS is to provide data independence. This means that user requests are made at a logical level without any need for knowledge as to how the data is stored in actual files in the database. Data independence implies that the internal file structure could be modified without any change to the users perception of the database. (Rasmussen et al, 2003) found that DBMS around 2003 were not optimised in the way to provide users with data independence, requiring first to have knowledge of the physical data structures in the database.

An idea developed by (Rasmussen et al, 2003) which built on the original proposed idea by (Dorf, 1999) was to design database models with three levels of abstraction, in order to ensure data independence. Each level aims to serve a different purpose in its approach, with the levels consisting of:

The internal level

Database viewed as a collection of files organised according to a type of internal data organisation, including types such as tree or relational data organisation. The internal level is the one closest to physical storage i.e. one concerned with the way in which the data is actually stored. It is the lowest level of abstraction describes how the data are actually stored (Coronel & Rob, 2000). At the physical level, complex low-level data structures are described in detail.

Conceptual/Logical level

An abstract view of the database, where the user is unaware of the internal storage details of the database seen in the internal level. This level acts to describe the contents stored in the database, and the relationships that exist between the data.

External Level

The level connected to the end users, where each user group has their own unique view of the database, tending to their specific needs. Due to users only requiring specific data, many will not be concerned with the majority of data that lies at this level.

The mapping between the three levels of database abstraction is the task of the DBMS. Changes in the internal level are caused when there are changes in the files themselves, or the overall structure of the data within the database is changed. When changes to the internal level do not affect the conceptual level and external level, the DBMS is said to provide for physical data independence. When changes to the conceptual level do not affect the external level, the DBMS is said to provide for logical data independence.

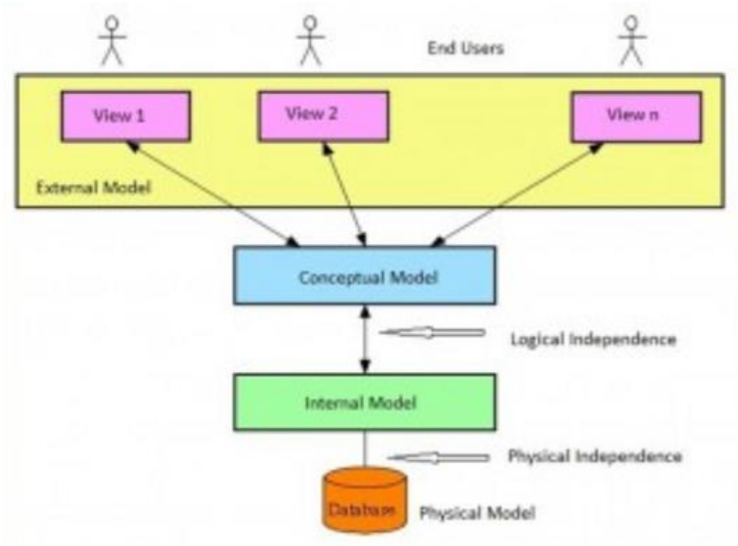


Figure 2 - Levels of Abstraction (Watt, 2004)

As previously mentioned, main part of the overall dbms that allows for this level of abstraction and allows for data to be used in different applications with overall high quality and easibility is the application of metadata.

2.2 Metadata

Metadata is defined simply as data about data. It means it is a description and context of the data. It helps to organize, find and understand data.

Since the beginning of information management systems, metadata has played an important role in managing and organising information. Even during the pre-technology days when data was transferred though punch cards and disk storage, metadata was used to describe the fields in which the punch cards were split into. Punch card systems soon evolved into system database management systems (dbms). Soon to follow were online database applications, such as early accounting systems and airline reservations software (McKenny, 1995). But in these days, systems were built for specific requirements, which meant that only the specific users were able to effectively use the data stored in systems. This was due to the lack of business metadata. (Inmon et al, 2008)

These days, systems such as data warehouses store metadata in many places within the overall system, whether it be the warehouse itself, ETL tools or singular data marts (Inmon et al, 2008). It's this metadata that allows the business users to understand the reports produced by the data warehouses and interpret the numbers or string fields in each row. The figure below displays how metadata is displayed in even simple business reports.

(Cabibbo and Torleone, 2001) were one of the first to introduce a data warehouse modelling approach which also included a new layer to the existing warehouse model, the logical layer. The inclusion of this layer allowed for the metadata to be managed independently form the applications, therefore improving how the data was maintained

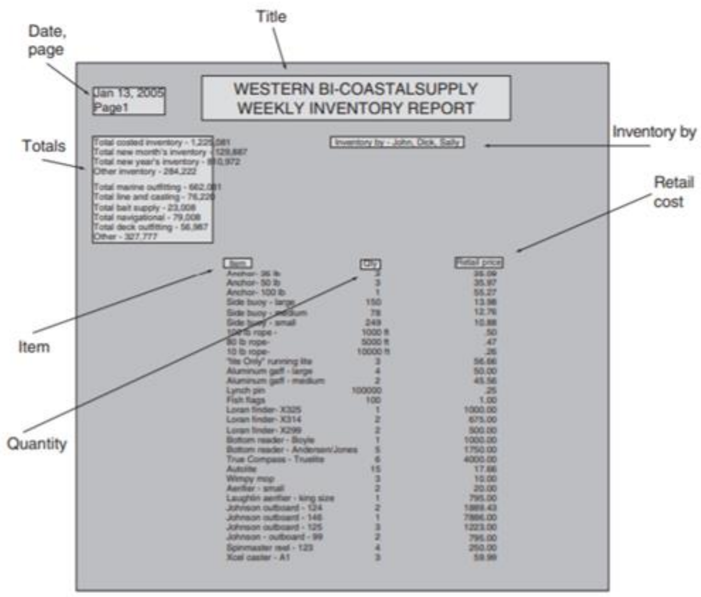


Figure 3 - Business report metadata

The metadata displayed in the report above is described as structured metadata (Posch et al, 2016). Structured metadata is defined by its consistency, always appearing in the expected positions to provide context. As reports are produced periodically, the metadata will always remain in the same position, with headers at the top of columns and summaries provided. Unstructured metadata refers to metadata stored incidentally, like a referencing in a piece of text.

2.2.1 Importance of Metadata

The importance of metadata in database systems has been well documented for some time. (Kimball et al, 1998 & Marco, 2000) were some of the preliminary advocates for metadata being an integral part of the overall success of decision environments, such as data warehouses. Metadata helps to enhance the quality of the data within the warehouse (Jarke et al, 2000), and benefits mostly the business users, who interpret the data to make decisions (Fisher et al, 2003). From this we can see that there are two major areas where metadata can improve systems, on the back-end systems technology and the front-end user defined reporting. (Kimball, 1998)

Integrating metadata into a dbms can have significant benefits. As (Shankaranarayanan and Even,2004) explain, these benefits can be divided into three main areas: Data management, Knowledge Management and Business Intelligence.

Data Management

Metadata helps in managing the data and making better use of it. As already mentioned, metadata allows for the use of the data stored in different applications for different purposes. The metadata also acts as an intermediate layer between business users and the information systems and applications that they have access to (Shankaranarayanan and Even,2004). This layer permits increased flexibility in managing customers, their changing requirements, and their use of resources within the organization. The metadata repository may offer insights to “often-used” versus “dormant” data by tracking usage patterns. Using this information, it becomes easier to anticipate changes in user requirements and take proactive measures when identifying new information product offerings.

Knowledge Management

By providing an abstraction of business and technical information, a metadata repository promotes sharing and reuse of knowledge, expertise, and application components. The metadata repository makes it easier to setup, maintain, and grow systems by explicitly documenting the indepth knowledge about systems and all their components in a format that can be easily retrieved and interpreted.

Data Quality Management

Metadata can help to solve data quality issues such as accuracy, consistency, completeness, and timeliness (Shankaranarayanan and Even,2004). Metadata also helps to improve how users interact with the database management system, which improves efficiency in processes. With metadata implemented, users only need to focus on what data they need, rather than to also consider how they need to go about locating and extracting the data. The complexity of information systems is reduced by the use of metadata and it makes the structure of the database transparent to the users (Elmasri and Navathe, 2003)

2.2.2 Where is Metadata Stored

In order for the metadata to be present, it must be able to be incorporated in the datasets or reports in some way. The same way that physical, printed reports store the metadata on the paper itself, in the form of column headers and titles, metadata in database systems must be able to do the same. Database systems contain tables that innately include features and specifications about the data, such as data types (integer, string etc), indexes and table keys, both primary and foreign. The tables that store this information are kept separate from the data, whilst inside the DBMS.

This information is usually stored in what is known as a metadata repository. A repository serves a similar purpose as the tables mentioned above but they contain information pertaining to data in different systems and different databases, whereas tables containing specs on the specific table are limited to refencing that one type.

2.2.2.1 Data Dictionary

A data dictionary as a DBMS component which acts as a centralised metadata repository, within a DBMS, that stores the definition of data characteristics and relationships. The DBMS data dictionary provides the DBMS with its self-describing characteristic. (Vetter, 2019).

The data dictionary's main function is to store the description of all objects that interact with the database. The information that is commonly seen inside a data dictionary include the following (Bourne, 2014):

• The names of fields contained in all of the organization's databases

• What table(s) each field exists in

• What database(s) each field exists in

• The data types, e.g., integer, real, character, and image of all fields in the organization's

databases

• The sizes, e.g., LONG INT, DOUBLE, and CHAR(64), of all fields in the organization's databases

• An explanation of what each database field means

• The source of the data for each database field

The two main types of data dictionary exist, integrated and stand alone. An integrated data dictionary is included with the DBMS. For example, all relational DBMSs include a built in data dictionary or system catalogue that is frequently accessed and updated by the relational DBMS. In other cases, some DBMS will not include their own built in data dictionary but instead may reference a third party stand alone data dictionary system. Data dictionary information is normally used by the DBMS to improve database query operations.

Integrated data dictionaries tend to limit their metadata to the data managed by the DBMS. Stand alone data dictionary systems are more usually more flexible and allow the DBA to describe and manage all the organization's data, whether or not they are computerized.

The metadata stored in the data dictionary is often the bases for monitoring the database use and assignment of access rights to the database users. The information stored in the database is usually based on the relational table format, thus enabling the DBA to query the database with SQL command (Coronel & Morris, 2014) For example, SQL queries can be used to extract information about the users of the specific table or about the access rights of a specific set of users.

The metadata stored in a data dictionary can be separated into business metadata and technical metadata. Business metadata refers to the definitions of the data used in the end user reports, whereas technical metadata represents information that describes how to access the data in its original native data storage. Technical metadata includes things such as datatype and the name of the data in the enterprise information system.

The following table displays how both business and technical metadata is used in the front end applications and back-end management systems:

Table 1 - Business and Technical Metadata

	Business	Technical
Front-End	• Contents of generated reports • Generated layer of definitions and naming conventions used for data items in business language.	• Queries or syntax for integrating data sources • Mapping of elements to different users, in the form of tables, fields or files.
Back-End	• Business interpretation of data items.	• Data structure • Elements from source material mapped into dbms

2.3 Metadata Management

In today's age of big data analytics, in order to properly integrate various sources of data into any information management system, the relevant metadata should be correctly defined. Previously, the majority of data that business users were encountering was structured data, however, this is rapidly changing due to the increase in popularity of big data systems (Vermuganti, 2003).

Over the last few years, some of the largest producers and users of big data have developed metadata style systems for collection and dissemination of big data. Google has introduced 'Protocol

Buffers'[3], their system for serialising structured data, allowing this data to be stored and interchanged between user teams. Other similar systems include the Apache designed Avro schema[4], used at LinkedIn and the Hive metacatalog[5]

2.3.1 Data Modelling

Typical DBMSs use a data model to describe the data and its structure, data relationships, and data constraints in the database. Some data models provide a set of operators that are used to update and query the database. DBMSs may be classified as either record based systems or object based systems. Both types of DBMSs use a data model to describe databases at the conceptual level and external level. Data models may also be called metadata models if they store metadata.

Two of the main existing data models used in record based systems are the relational model and the network model or entity relationship model.

2.3.1.1 Relational Model

In the relational model, data is represented as a collection of relations. To a large extent, each relation can be thought of as a table. Each table has columns, keys and indexes. A key is a set of columns whose composite value is distinct for all rows. No proper Subset of the key is allowed to have this property. A table may have several possible keys. In relational models, data at the conceptual level is represented as a collection of interrelated tables. The tables are normalized So as to minimize data redundancy and update anomalies. The relational model is a logical data structure based on a set of tables having common keys that allow the relationships between data items to be defined without considering the physical data base organization.

2.3.1.2 Entity Relationship Model

A known high level conceptual data model is the Entity Relationship (ER) model. The entity relationship model was designed by Peter Chen in 1976 to try an incorporate the best parts of the existing relational and network models (Chen, 1976). An ER Model is used to model the logical view of the system from data perspective which consists of these components: In an ER model, data is described as entities, attributes and relationships.

<u>Entities</u>

An entity can be thought of as anything in which data can be stored about, for example a physical object such as a person, a house or a car, or even a concept such as a company or a job. Each entity with similar characteristics is said to belong to a specific entity type and a group of entities is known as an entity set. An example is a banana, which has the entity type 'fruit', where the set of all fruits is

[3] Link
[4] Link
[5] Link

the entity set. IN an entity relationship diagram, an entity type is derived from the relevant attributes of that entity.

<u>Attributes</u>

Each entity has a set of properties, called attributes that describe the entity. Attributes can take the form

- Key Attribute – The attribute which uniquely identifies each entity in the entity set is called key attribute. For example, in figure 4 the identification number (ID) will be unique for each expert.
- Composite Attribute – An attribute composed of many other attribute is called as composite attribute. For example, Address attribute of an institution should consist of Street, City, State, and Country.
- Derived Attribute – Attributes that can be derived from the application of other attributes. The age of an expert can be derived if the date of birth is present as an attribute.

<u>Relationships</u>

A relationship is an association between entities. For example, in figure 4, an expert 'entity' may be described by its name, email, and phone number and can be associated with an institution entity by the relationship "works for". Relationship types can be binary or can be characterised by its cardinality.

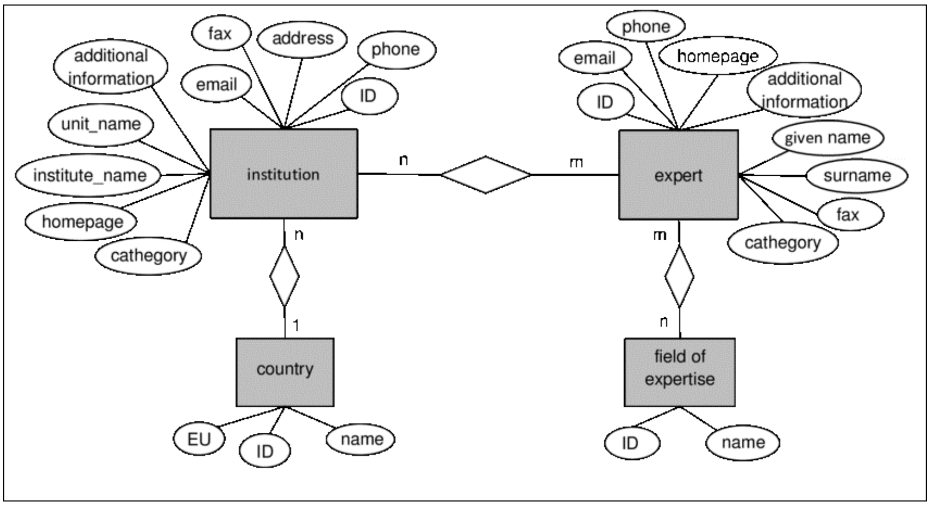


Figure 4 - Entity relationship model

Cardinality refers to the minimum or maximum number of entities can are to be involved in each relationship. In terms of minimum cardinality, there can exist two scenarios:

0: Where an entity can be stored without being connected to another entity, which is called partial participation, or;

1: where an entity must be connected to at least one other entity, or total participation.

In regards to maximum cardinality, we also have two scenarios:

1: where an entity can be connected to at most 1 other entity, or;

N: where an entity can be connected to at most N other entities.

The entity relationship model is well suited to being applied as the logical model in the production of reports, due to the nature of model describing values, therefore being able to aptly describe the meaning behind the values represented in a report, as seen in Figure 5 below.

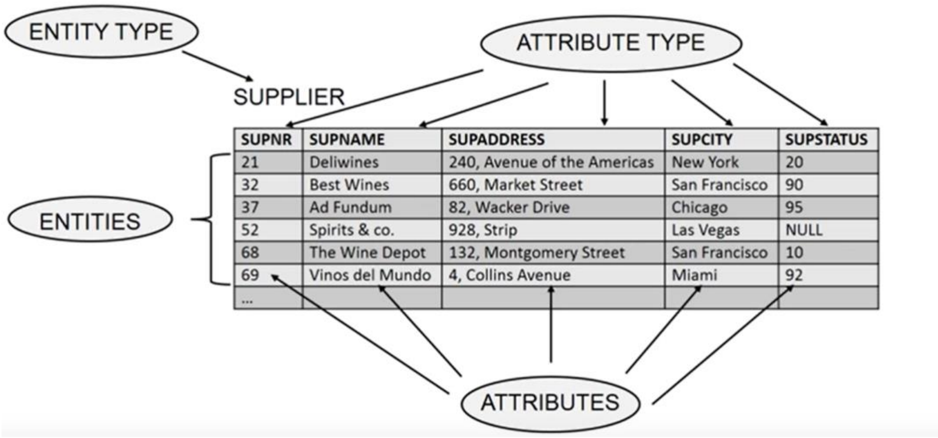


Figure 5 - Entities represented in a report

2.3.1.3 Graph Databases

Graph databases are databases that use the entity relationship model as the base model. A graph database is essentially a collection of nodes and edges. Each node represents an entity (such as a person or business) and each edge represents a connection or relationship between two nodes. Every node in a graph database is defined by a unique identifier, a set of outgoing edges and/or incoming edges and a set of properties expressed as key/value pairs. (Bajpayee et al, 2015)

An example could be fruit and banana is two entities, with a relationship stating “is a type of” from banana to fruit. In this regard they differ from relational databases, where relationships are stored at the level of table definition, relationships here exist on the level of individual rows within a table.

Graph databases are a part of the NoSQL realm of databases, which stands for ‘not an SQL’ database. NoSQL databases provide an alternative to existing SQL using relational databases, allowing for more possibilities in how the database is designed.

As a system, it stores data in a graph structure and allows the execution of more semantic queries, directly retrieving related data. Graph databases enhance the representation of relationships, but also allow for more complex data analysis techniques, such as community detection, pattern recognition and centrality measures[6]. Another advantage of Graph Databases is the flexibility in the data schema. While Relational Databases requires new tables or alterations in the existing ones to add new types of data, in a Graph Database we can add new type of vertices and edges without alterations in the previously stored data

[6] Link

Some of the main advantages in using graph databases ahead of relational databases due to three main reasons. These are:

Data volume

The volume refers to the size of the graph, determined by the number of nodes and edges. As the volume of the graph increases, analysis of such graphs becomes more difficult to process (Strohbach, 2016)

Large datasets simply become too large to be processed efficiently when stored in relational databases. In particular, query execution times increase as the size of tables and the number of JOINs grow. (Zaitev, 2006)

Data Velocity

The velocity is the rate at which the graphs increase in size and complexity. As the size and complexity increase, it becomes difficult to process queries or commands such as finding the distance between two nodes (Kumar, 2016)

Due to high volume as well as changes in data velocity, databases are required to not only handle high levels of edits but also deal with surging peaks of database activity. Relational databases are not modelled in a way to handle a sustained level of write loads and can crash during peak activity if not properly tuned (Robinson, Webber and Eifrem, 2015). NoSQL databases address both data velocity challenges by optimizing for high write loads and by having more flexible data models.

Data Variety

Graphs today are formed from different sources and types of data, and often created through integration of XML/JSON, documents, and graph-structured data (Strohbach, 2016). Most relational databases today were not designed to integrate data this varied, and many relational dmbs encounter null values in tables.

On the other hand, NoSQL databases are designed from the bottom up to adjust for a wide diversity of data and flexibly address future data needs, each adopting their own strategy to how to handle the variety of data. (Sasaki, 2018)

Data Valence

The valence of a dataset refers to the degree of connectedness or interdependence. If data valence is high, then strong relationships exist between the data elements.

Over time, highly dense and lumpy data networks tend to develop, in effect growing both your big data and its complexity (Sasaki, 2018). This is significant because densely yet unevenly connected data is difficult to unpack and explore with traditional analytics, such as those based on RDBMS data stores. Thus, the need for NoSQL technologies where relational databases aren’t enough.

3 Current Supervisory Reporting Requirements

The predecessor of the European Banking Authority (EBA), the Committee of European Banking Supervisors (CEBS), was an independent body in charge of advising and coordinating on banking supervision and regulation in the EU. While one of its main tasks was to contribute to developing high-quality common reporting standards, in practice it had much more limited powers and, in particular, could not issue any directly applicable or binding regulations. Nevertheless, CEBS issued guidelines on reporting on three different areas: the reporting of own funds and own funds requirements; the reporting of large exposures covered by the framework on Common Reporting (COREP) and the reporting of financial information (FINREP). These guidelines were implemented by competent authorities through their national reporting framework.

The FINREP core templates for balance sheets and income statements were implemented by 18 EU supervisors, although only three countries implemented the whole FINREP framework. COREP guidelines were implemented by all EU supervisors, but only seven of them implemented them fully. Out of all the EU supervisors, only two required their banks to report both FINREP and COREP in full (ECB, CEBS 2010).

Following the financial crisis, a G20 meeting in 2009 called upon the Financial Stability Board and the International Monetary Fund to identify and combat the gaps in reporting information. (FSB & IMF, 2009). It was identified in the current status where supervisory reporting was based on requirements set at a national level, with IT systems to match resulted in increasing costs of operating across a single market, and that the absence of a unified supervisory rule book , made it difficult to organise supervisory activities across countries in a coordinated way.

This resulted in the development and implementation of a harmonised EU wide supervisory framework, named the Implementing Technical Standards (ITS). The ITS imposed a large amount of restructuring and effort on behalf of the reporting nations, as it accounted for a major expansion in the scope of reporting requirements from previous national frameworks, but the benefits of the harmonised framework have since been realised (Quagliariello and Rimmanen, 2012).

The crisis also led to a way of creating stronger prudential requirements for banks and creation of the Single Rulebook with exactly the same rules across the EU single market, with no room for regulatory competition between member states.

The CRD IV package consists of the Capital Requirements Regulation[7] (CRR) and the CRD[8], and is the backbone of the Single Rulebook. The CRD IV implements capital requirements issued in 2013 by the Basel Committee on Banking Supervision (BCBS), called Basel III [9]. Basel III requires banks to hold more and better quality capital and introduced prudential requirements for liquidity and leverage. The CRD IV package is complemented by delegated and implementing acts by the European Commission, and by the EBA technical standards, guidelines and recommendations.

[7] Link
[8] Link
[9] Link

3.1 ITS on supervisory reporting

The ITS on supervisory reporting consist of several legal and technical parts, and major advances compared to the previous guidelines that focused only on providing sets of reporting templates. The main objective is not only to achieve a truly uniform reporting framework, but also to help institutions interpret and implement this extensive framework. The ITS bring together several reporting areas and uses common structure, conventions, concepts and definitions across different reporting areas.

The ITS on supervisory reporting include the main standard setting out the reporting requirements and defining the scope of institutions, frequency of reporting and reference and remittance dates and annexes specifying the reporting requirements in the form of templates and instructions. The ITS also include a data point model (DPM) and validation rules to ensure consistent application of the requirements that are maintained by the EBA on the website. The EBA also developed eXtensible Business Reporting Language (XBRL) taxonomies for the ITS as a technical format for data exchange.

The DPM is a technical translation of the requirements included in the reporting templates. The templates and instructions include data definitions and references to legal texts and standards, but fully harmonised data and more-effective data production is ensured by the provision of uniform specifications on a granular level. The DPM describes business concepts and their relations in detail, and provides necessary technical specifications for developing IT reporting formats and common dictionaries of terms that can be used in data collection. By setting out the developments in a DPM, this can contribute to a deeper understanding of the requirements by institutions and therefore ensure harmonised and consistent data.

Financial institutions could report regulatory returns by filling in templates, but the ITS reporting requirements include significantly higher volumes of data and more detailed and complex data items making it very challenging to report them without standardised IT solutions. The DPM helps reporting institutions and supervisors receiving data to develop and automate their reporting processes. The DPM is used by the EBA to create XBRL taxonomies as a technical format for data exchange for each reporting area of the ITS.

4 Data Models used for Supervisory Reporting

This section of the report describes the current data models used to integrate metadata into the production of supervisory reports. We explain two of the major modelling approaches used:

- The XBRL/DPM, used for the production of FINREP and COREP reports collected by the EBA to , based on the ITS mentioned in the previous section; and
- Statistical metadata exchange (SDMX) used in the production of BSI, MIR and SHS datasets.

4.1 EBA XBRL-Data Point Model (DPM)

4.1.1 XBRL

EXtensible Business Reporting Language (XBRL) is a technical standard for description of metadata and exchange of data. It enables defining information requirements in so-called taxonomies and further exchange of data (referring to definitions from taxonomies) in reports called instance documents.

The underlying technology is XML (eXtensible Mark-up Language) and derived specifications, in particular XLink which enables replacing of nested structures (as commonly applied in classic XML schemas and instances) with more flexible linking mechanism that is also responsible for conveying various semantics.

Application of XBRL is very wide and its scope is not explicitly defined hence it can be used for remittance of various types of data. Moreover, taxonomy authors as well are creators of instance documents are equipped with a large number of customization options defined as part of the extensibility of the XBRL.

There are however a few fundamental constructs of XBRL applied in any reporting scenario[10]:

- XBRL taxonomy contains definitions of concepts that describe the requested data. Definition of a concept must at least contain its unique identifier (name), determine the expected type/format of value (data type) and provide the time context for which the value of a concept is expressed or measured (either at a point of time or in duration of time). For documentation purposes, as well as to bring forward more semantics, concepts may be associated with human readable labels, reference to source materials (e.g. legal acts or standards) or be linked with one another in various types of relations.

- XBRL instance document contain facts. Fact carries a value for a concept defined in a taxonomy. It refers to a context identifying an entity and period for which it is reported. Numeric facts refer also to a unit of measure and contain information on precision of value (data accuracy).

[10] Link

XBRL instance document may contain many facts for a single taxonomy concept (for example values for different periods or in different units such as currencies, etc.). Also not all taxonomy concepts need to be represented by facts in an instance document (companies may not conduct all business activities that are addressed by taxonomies).

Implementations of XBRL may also include the Dimensions add-on. Is such a case an XBRL taxonomy apart from concepts (that are later represented by facts in instance documents as described above) may also define artefacts serving as properties detailing or further describing the concepts. These take form of dimensions which may be of two kinds: explicit and typed. Explicit dimensions have their values (so called domain members) defined in the taxonomy while typed dimensions are restricted by expected format of allowed values. Taxonomy may clearly define which concepts are associated with dimensions and their values using cubes. In XBRL instance documents, dimensional properties (i.e. dimensions and their values, that are either domain members or following the defined format) describing a fact are included in the definition of a context to which it refers.

4.1.2 DPM

The DPM methodology has emerged in evolution process of application of XBRL for financial and prudential reporting in the European banking sector. Data Point Modeling is a methodology for the development of financial data models that describe characteristics of the information exchanged in the context of supervisory reporting processes. These data models are often referred to as Data Point Models or as meta-data models. Data Point Models are formally represented by XBRL taxonomy files[11].

In the beginning and for the first few years COREP and FINREP taxonomies have been developed by the IT who basically translated the tabular representation of information requirements to the technical format of XBRL. At some point though the maintenance and updates started to require increasing business input and the financial domain experts had been more and more involved in the process. This caused the need for definition of a formal model for description of requested data which could be provided by the business and translated to technical format by the IT without any loss of information or space for interpretation.

The resulting methodology has been called the Data Point Modelling to emphasise the shift in the approach from the form centric representation of information requirements (based on tabular views) to the data centric definitions (detailing properties of each exchanged piece of information).

[11] Link

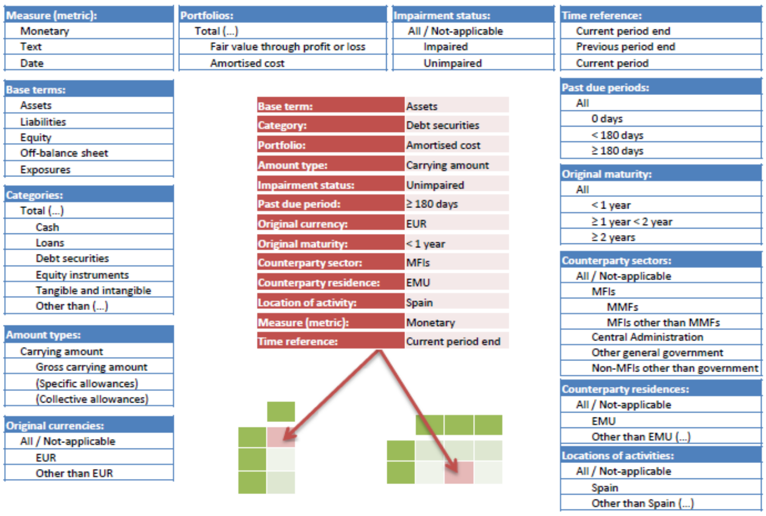


Figure 6 - Data Point

A Data Point represents an individual data requirement. Data points are expressed as a composition of features that univocally identify the financial concept to be measured:

- A metric: characteristic that defines the nature of the measure to be performed.
- A set of dimensional characteristics that qualify and complement the metric and provide the proper context to understand the financial phenomenon represented.
- A time reference that helps to determine the specific instant or interval of time in the context of a given reporting period.

Example:

- Taxonomy: Financial Reporting 2016-A
    - Author: EBA (DPM 2.8)
    - Table: Balance Sheet Statement: Assets (F_01.01)
        - Row: Total assets
        - Column: Carrying amount
- Data Point:
    - Metric: eba_mi53 - Carrying amount
    - Dimension 1: BAS - Base
    - Dimension 1 Value: x6 - Assets
    - Dimension 2: MCY - Main Category
    - Dimension 2 Value: x25 - All assets

<u>Data Point representation in an XBRL Instance</u>

```
<xbrli:context id="c1">
  <xbrli: entity>
    <xbrli:identifier scheme="http://standards.iso.org/iso/17442">DUMMY_ENTITY</xbrli:identifier>
  </xbrli:entity>
  <xbrli:period>
    <xbrli:instant>yyyy-mm-dd</xbrli:instant>
  </xbrli:period>
  <xbrli:scenario>
    <xbrldi:explicitMember dimension="eba_dim:BAS">eba_BA:x6</xbrldi:explicitMember>
    <xbrldi:explicitMember dimension="eba_dim:MCY">eba_MC:x25</xbrldi:explicitMember>
  </xbrli:scenario>
</xbrli:context>
<eba_met:mi53 unitRef="uEUR" decimals="-3" contextRef="c1">FACT_VALUE</eba_met:mi53>
```

4.1.2.1 Dictionary

The basic elements required to develop data models are grouped into a single point called Dictionary, which is common to all taxonomies. A taxonomy developed with DPM methodology will include:
(a) The data model in which all the data required are defined exhaustively, (b) the information necessary to allow graphical representation and (c) validation rules that data must satisfy.
The dictionary contains the following elements:

4.1.2.2 Metrics

Metrics define the nature of the measure to be performed. Each metric determines a data type, a period type and additional semantics of their corresponding data points.

- The data type corresponds to the nature of the values of the facts to be reported (monetary information, percentages, dates, string text etc.)
- The period type establishes whether the property or measure is performed at a specific instant of time (stocks) or during an interval of time (flows).

4.1.2.3 Dimensions, Domains and Domain Members

A Dimension is a characteristic that qualifies and complements the metric and provides the proper context to understand the financial concept represented by a Data Point. Each dimensional characteristic is composed of a specific dimension and a domain member. The dimension represents an identifying property and the member is the value given to that property in the context of the data point.
Domains are sets of members that share a certain semantic identity.
Domain members are the different values of a domain that are given to dimensions.

Example:
Based on the EBA Dictionary, the image below shows a Domain called “Geographical area”, and two Dimensions (“Country of Birth”, and “Country of Residence”). Each dimension takes a different value from members of the Domain.

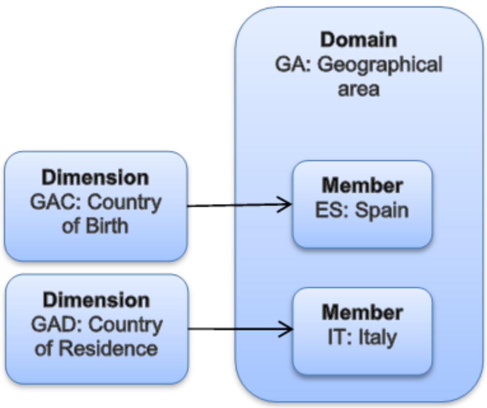


Figure 7 - Dimensions in the DPM

4.1.2.4 Hierarchies

Hierarchies are sets of members of an explicit domain arranged in a hierarchical disposition. A node of a hierarchy can define basic arithmetical relationships (=, <= or >=) in relation to its child nodes.

4.1.2.5 Taxonomies and Frameworks

A taxonomy represents a set of reporting requirements enforced by a certain legal document or set of documents. A taxonomy is defined in terms of modules, tables and table groups. In addition, taxonomies include attributes to identify the legal rules, their version date and an optional currency period.

Frameworks are groups of taxonomies following some functional requirements.

4.1.2.6 Modules

Modules are pre-defined sets of tables and table groups that are used in a certain process. For instance, in reporting processes, a module defines sets of information that must be reported together.

4.1.3 SDMX

SDMX[12] (Statistical Data and Metadata eXchange) is a set of standards and guidelines aimed at facilitating the production, exchange, dissemination, retrieval and processing of statistical data and metadata. SDMX is sponsored by a wide range of public institutions including the UN, the IMF, the Worldbank, BIS, ILO, FAO, the OECD, the ECB, Eurostat, and a number of national statistics offices and national banking authorities.

The SDMX Information Model[13] provides a broad set of formal objects and their relationships to represent statistical data and metadata, actors, processes, and resources within statistical exchanges.

These objects or artefacts are defined as follows:

4.1.3.1 Concept and Concept Scheme

Concepts play an important role in the SDMX Information Model as they are used to describe the structure of a multidimensional statistical table or the structure of a metadata report. In the SDMX Information Model, Concepts can have a specific value representation (coded value, numeric format, date format, string, etc.) that can be defined in the Concept Scheme.

[12] Official SDMX site
[13] Link

4.1.3.2 Code Lists

In the SDMX Information Model, a Code List is an object containing a list of codes and maintained by an agency. A Code List is simply a set of values to be used in the representation of a Concept (Dimension or Attribute) in Data Structure Definitions.

The model allows a code list to have hierarchy of codes. In that case, the hierarchy is made by defining any relationship parent-child among the codes of one code list.

4.1.3.3 Data Structure Definition (DSD)

The Data Structure Definition is a description of all the structural metadata needed to understand the structure of the Data set. In fact, the Data Structure Definition links the statistical data to its structural metadata by assigning descriptor Concepts to the elements of the statistical data.

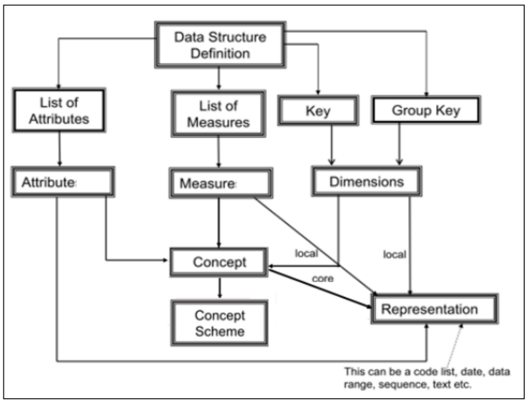


Figure 8 - Data Structure Definition

As shown in the above figure, the Data Structure Definition is formed by three sets of Concepts:

- Dimensions comprising Non-Key Dimensions and the Key lists:
- Key descriptor;
- Group Key descriptor(s) - there may be more than one Group Key descriptor;
- List of Measures;
- List of Attributes.

Example: DSD: ECB_IBSI: Individual Balance Sheet Items.

- Dimensions: FREQ, REF_AREA, ADJUSTMENT, BS_REP_SECTOR, MFI_LIST_IND, BS_ITEM, MATURITY_ORIG, DATA_TYPE, COUNT_AREA, BS_COUNT_SECTOR, CURRENCY_TRANS, BS_SUFFIX
- Mandatory attributes: TITLE, UNIT, UNIT_MULT, DECIMALS, COLLECTION, TITLE_COMPL
- Time series key: M.ES.N.A.ESXXXX.A60.X.1.Z5.0000.Z01.E.
    - FREQ: Monthly (M);
    - REF_AREA: Spain (ES);
    - ADJUSTMENT: Neither seasonally adjusted nor calendar adjusted data (N);
    - BS_REP_SECTOR: MFIs excluding ESCB (A);
    - MFI_LIST_IND: Spanish MFI XXXX (ESXXXX);

- BS_ITEM: Non-financial assets (including fixed assets) (A60);
  - MATURITY_ORIG: Not applicable (X);
  - DATA_TYPE: Outstanding amounts at the end of the period (stocks) (1);
  - COUNT_AREA: Counterpart area: World not allocated (geographically) (Z5);
  - BS_COUNT_SECTOR: Unspecified counterpart sector (0000);
  - CURRENCY_TRANS: All currencies combined (Z01);
  - BS_SUFFIX: Euro (E).
- Time series key attributes: UNIT: Euro; UNIT_MULT: 6; DECIMALS: 2; COLLECTION: End of period.
- Time series key list of measures: OBS_VALUE (observation value); OBS_CONF (observation confidentiality); OBS_STATUS (observation status)
- Time series observations: 145.02

SDMX-Meta language of this series:


```
<Series FREQ="M" REF_AREA="ES" ADJUSTMENT="N" BS_REP_SECTOR="A" MFI_LIST_IND="ESXXXX" BS_ITEM="A60"
        MATURITY_ORIG="X" DATA_TYPE="1" COUNT_AREA="Z5" BS_COUNT_SECTOR="0000" CURRENCY_TRANS="Z01"
        BS_SUFFIX="E" COLLECTION="E" DECIMALS="2" TITLE="IBSI Example"
        TITLE_COMPL="IBSI Example. Complementary Title" UNIT="EUR" UNIT_MULT="6">
  <Obs TIME_PERIOD="2017-05" OBS_VALUE="145.02" OBS_STATUS="A" OBS_CONF="C"/>
</Series>
```

Single Data Dictionary

4.2 Introduction

The ECB Directorate General (DG) Statistics established the Data Intelligence Service Centre in March 2016 to provide a single shared platform for:

- business centric analytical capabilities,
- data integration and ETL services, and
- metadata management based on the Single Data Dictionary (SDD) and Data Inventory

The DISC data platform is the central secure place for organising, storing, analysing data and related collaboration within the ECB. It aims to make large data volumes and the variety of data structures usable by employing the latest big data technologies available.

Whilst DISC provides a common platform, data inefficiencies and inconsistencies continue to be caused by a lack of harmonisation and standardisation. This lack of standardisation stems from collecting data from two different modelling approaches, which define concepts in a similar yet different way, leading to added confusion in the collection process.

In order to combat this, the ECB set out to have one single set of reporting metadata that can be used for all reports produced. It was decided that neither the SDMX nor DPM approach could be used and applied to all reports, as it would result in an innate loss of information. Therefore, the ECB decided to establish the Single data dictionary, as a centralised reporting dictionary, integrating all reporting metadata in one place, as to standardise all reporting models.

4.3 Single Data Dictionary

The Single Data Dictionary (SDD) is a data dictionary aimed to describe the datasets used in the ECB. A data dictionary is a repository of metadata that describe datasets; it provides meaning and relevant information about the organisation and meaning of the data thereby enabling analysis of the data.

The description provided by a formal data dictionary serves both to the users of the information, which can use the dictionary to explore, understand and extract information from the data, and to IT systems, using the dictionary as a reference catalogue to perform tasks.

The ECB manages several different datasets, which are defined by different stakeholders, follow heterogeneous definitions and modelled according to various methodologies. This situation narrows down integration possibilities, limits the comprehension of data definition, collection, production, and dissemination processes, as well as reduces the data usability for the users.

The lack of integration is twofold: dataset description methodology and semantics.

1. Different reporting frameworks are modelled in different ways. For instance, some data frameworks for monetary policy purposes are modelled according to the SDMX standard, while others for supervisory purposes are modelled according to the Data Point Model

(DPM) methodology. This situation makes it difficult to understand the structure of the datasets and their joint use becomes highly challenging, time consuming data manipulation is needed to have the same structures. An additional problem associated with the use of different modelling methodologies is that the metadata are usually documented in very different ways. As a result, users first need to understand how a framework is documented, which can be a very demanding task, and only after they can actually start understanding and using the framework. This calls for an integrated methodology for describing datasets.

2. The codes used in different frameworks to describe the concepts (aspects of reality) might have the same or different definitions. Figuring out whether the same code used in two different reporting frameworks indeed describes the same aspect of reality and has the same underlying definition is not always straightforward. This gives rise to the need for semantic integration.

The objective of the Single Data Dictionary is to describe ECB datasets in a common, standardised dataset modelling approach, which is a prerequisite for effective and efficient data integration. The SDD can be thought of as the reference library of an ECB-wide information system comprising all kinds of datasets.

The SDD addresses the issues of (i) dispersed knowledge about the methodology and semantics of various data dictionaries; (ii) duplicated and/or fragmented metadata in the common information system (if at all); and (iii) inconsistencies in defining reference metadata across different dictionaries. The SDD helps preventing the proliferation of data silos or stovepipes, allow users to make use of the data contained in an information system with greater effectiveness and fewer resources.

The SDD provides a unique dataset description methodology (methodological integration of dataset modelling) and, when necessary, unique codification (semantic integration) for managing data in an integrated manner.

4.3.1 The methodological integration

In the ECB, different data frameworks are described in different ways, with different methodologies. At present, there are mainly two standardised dataset description methodologies in use: the Data Point Model (DPM) and Statistical Data and Metadata Exchange (SDMX) methodologies.

A precondition for effective and efficient data integration is to describe all the datasets with the same dataset modelling framework.

The dictionary methodology is the technique used to provide a formal definition of datasets. A formal data dictionary is a necessary piece for IT systems to work. If the dictionaries of two systems are defined with a different dataset modelling methodology, the systems will not be able to directly understand each other. Taking as an example the DPM and SDMX the same framework may be defined under both methodologies, but at the moment there is no possibility to translate one to the other. So a system based on the DPM, like the Supervisory Banking data system (SUBA) for supervisory data, is not able to read or process information described with SDMX. Vice-versa, a system based on SDMX, like the ECB's Financial Analysis and modelling environment (FAME), is not able to read or process information described with the DPM.

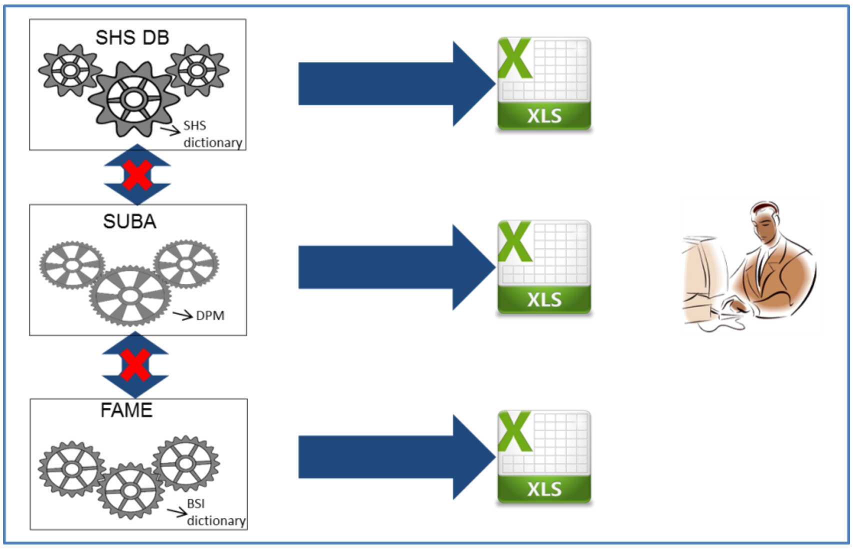


Figure 9: The lack of methodological integration problem.

The picture shows the case in which a user or producer is interested in using together data from SHS, FINREP and BSI. Each of these datasets is in a different system, using different methodologies. Communication between systems is not possible, because they use different dictionaries, so users will need to work outside the systems, for instance exporting the information to different Excel files.

It may be tempting simply to take one of the existing modelling methodologies and describe all the datasets. However, the translation could imply losing information (for instance, not all aspects of DPM fit into SDMX and vice versa) furthermore, it may be difficult to automate this process.

However, there is a more feasible approach, i.e. to create a new methodology that:

1. contains all the substantive details present in the existing modelling methodologies such as SDMX DSDs and DPM (so that no information is lost during the translation);
2. is compatible with the existing methodologies so that the translation process can be automated for the most part.

The SDD uses the SMCube methodology, which fulfils the above conditions and serves to describe all existing reporting frameworks. The benefit of having a common information model, compatible with all other models, is that different kinds of datasets can be stored, managed and retrieved in a common way regardless of the model/standard used for the actual data exchange.

The SMCube methodology is a dataset modelling methodology, which aims to describe the datasets by setting out the structure for data and metadata. It is an abstract model that can be used to model various statistical and supervisory reporting frameworks coming from different standards.

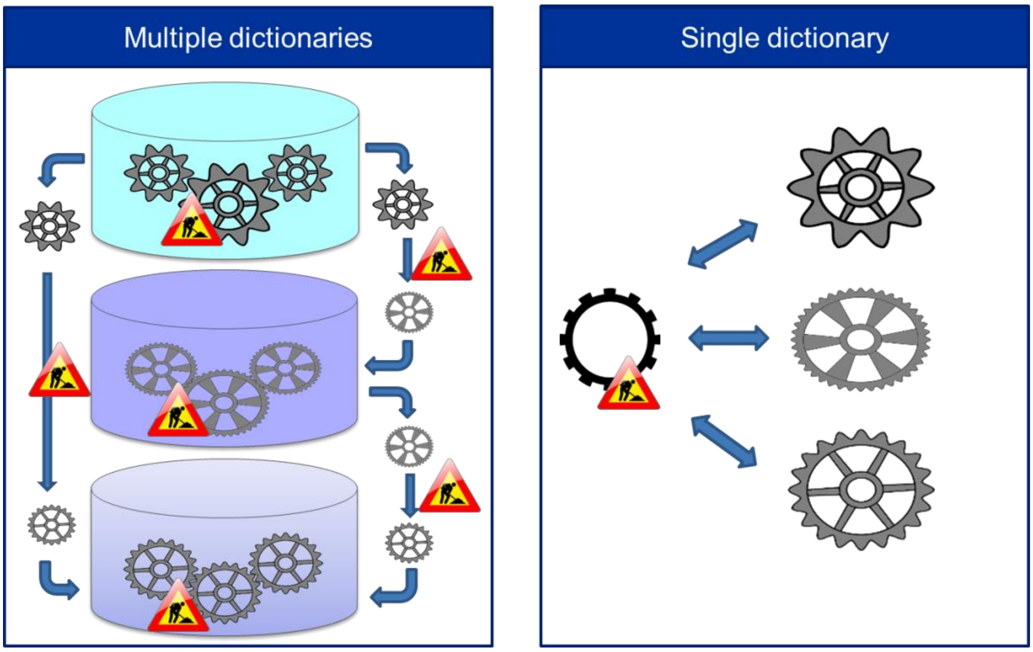


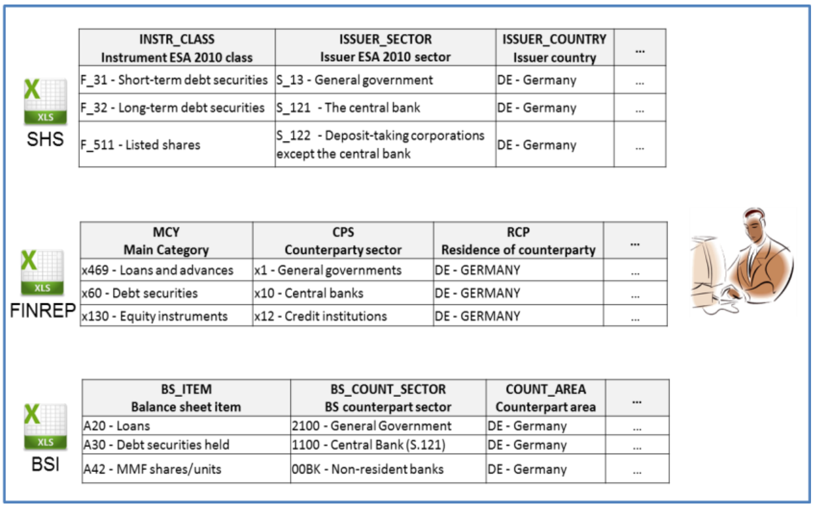
Figure 10: Integrated methodological approach.

The SDD approach for achieving methodological integration consists in building a methodology that is fully compatible with existing methodologies, rather that mapping them individually.

4.3.2 The semantic integration

Once the issue of integrated modelling methodology is solved, the problem of semantical integration still persists.

Let's take as an example of three existing data frameworks containing information related to the balance sheets of the banks: Securities Holding Statistics (SHS), FinRep and Balance Sheet Items (BSI). Figure 3 shows that the datasets, even if related, contain different codes and names for the concepts they use, thus complicating the use of data.



INSTR_CLASS Instrument ESA 2010 class	ISSUER_SECTOR Issuer ESA 2010 sector	ISSUER_COUNTRY Issuer country	...
F_31 - Short-term debt securities	S_13 - General government	DE - Germany	...
F_32 - Long-term debt securities	S_121 - The central bank	DE - Germany	...
F_511 - Listed shares	S_122 - Deposit-taking corporations except the central bank	DE - Germany	...

MCY Main Category	CPS Counterparty sector	RCP Residence of counterparty	...
x469 - Loans and advances	x1 - General governments	DE - GERMANY	...
x60 - Debt securities	x10 - Central banks	DE - GERMANY	...
x130 - Equity instruments	x12 - Credit institutions	DE - GERMANY	...

BS_ITEM Balance sheet item	BS_COUNT_SECTOR BS counterpart sector	COUNT_AREA Counterpart area	...
A20 - Loans	2100 - General Government	DE - Germany	...
A30 - Debt securities held	1100 - Central Bank (S.121)	DE - Germany	...
A42 - MMF shares/units	00BK - Non-resident banks	DE - Germany	...

Figure 11: The lack of semantical integration problem.

The picture shows an extract of SHS, FINREP and BSI with the real codes and names that a user would find after exporting the frameworks to Excel. It seems clear to users that they are all related (not to the IT systems, though), but the extent to which they are the same is not. Can we say that Debt securities held is the same as Debt securities? What is the relation between Credit institutions, Deposit taking corporations except the central bank and Non-resident banks? To understand to what extent this may be problematic, it might be interesting to note that DE – Germany- may not always be the same, since, for instance, the ECB may be aggregated under Germany or not. A thorough analysis of the regulations and technical documentation of the datasets is the only way to get a full understanding of the datasets

The semantic integration means that all concepts for describing statistical and supervisory data should have a single and unambiguous codification. In the followings, we refer to the integrated dictionary as the reference dictionary. This single unambiguous codification should be suited for use in all data models.

Single means that if two categories in two different non-reference dictionaries are the same, a single code is used in the reference dictionary.

Unambiguous means that all definitions in the integrated dictionary should be clear, carrying a univocal meaning.

The ECB currently deals with many non-integrated dictionaries and will keep using these dictionaries at least in the medium term. While it may not be practical to revamp each of them; one can create a container where all the dictionaries that are already in place will be reconciled, integrated and streamlined. To be more precise, the solution to the semantic challenge is to build one integrated and centralised dictionary on top of all the non-integrated dictionaries that have different ownerships, and to map the non-reference and reference contents. The reference dictionary will contain the contents of all the non-reference dictionaries abolishing the duplications and ambiguities.

If two categories in two different non-reference dictionaries are not the same, each of them gets assigned a different code in the reference dictionary. In other words, different concepts will be defined with different codes. For instance, if the definition of “central banks” in dictionary A is the same as the definition of “the central bank” in dictionary B, then these two categories from two distinct non-reference dictionaries are mapped to one single code in the reference dictionary. If the definitions differ, the category “central banks” from dictionary A is mapped to its own category in the reference dictionary, and the category “the central bank” from dictionary B is mapped to a different code in the reference dictionary.

To recapitulate, the SDD approach or solution to the semantic challenge comprises three steps:

first, importing (uploading) all the non-reference dictionaries to the SDD;

second, analysing the definitions in the non-reference dictionaries and creating the unique reference dictionary;

third, mapping the non-reference dictionaries to the harmonized reference dictionary by following certain guiding principles.

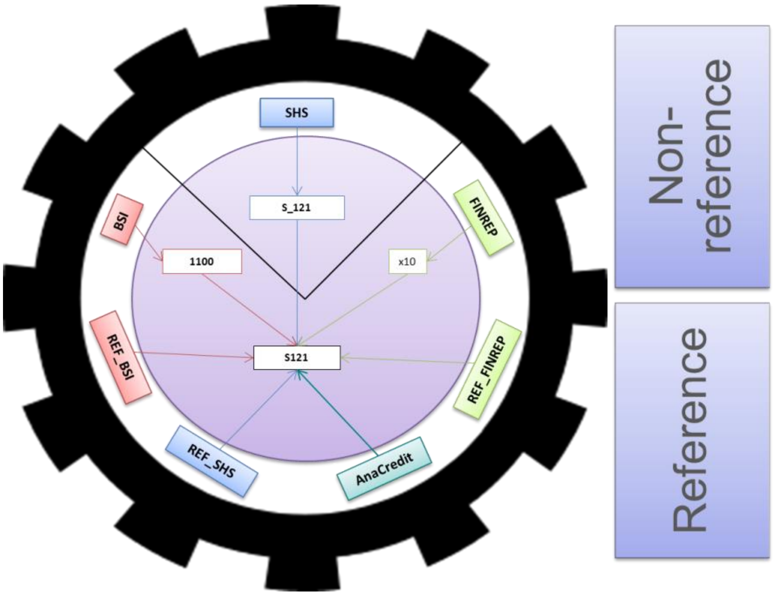


Figure 12: Semantic integration.

It is relevant noting that semantic integration is an option that the SDD offers, but it is not essential to the SDD. This means that it may make perfectly sense that one data framework is methodologically integrated. In reality, the SDD offers three models regarding integration:

Methodological integration

If a dataset is not normally going to be used together with other datasets costs of semantic integration may exceed the benefits, the non-reference dictionary will be imported in the SDD but not mapped in the reference part while still benefitting from the SDD methodology and codification.

Indirect sematic integration.

This model corresponds to the example shown so far, where there are concepts originally defined, and mappings to the reference dictionary. This means in practice that for one dataset there are two definitions: One original, and one with the reference codes. This model will be necessarily used when the original definition of a dataset comes from an external authority or group, but it may also be advisable in other cases, like for already existing datasets, where the cost of changing codifications may be high.

Direct semantic integration

Direct integration means that a framework is directly defined using the reference dictionary, thus avoiding the need for mappings.

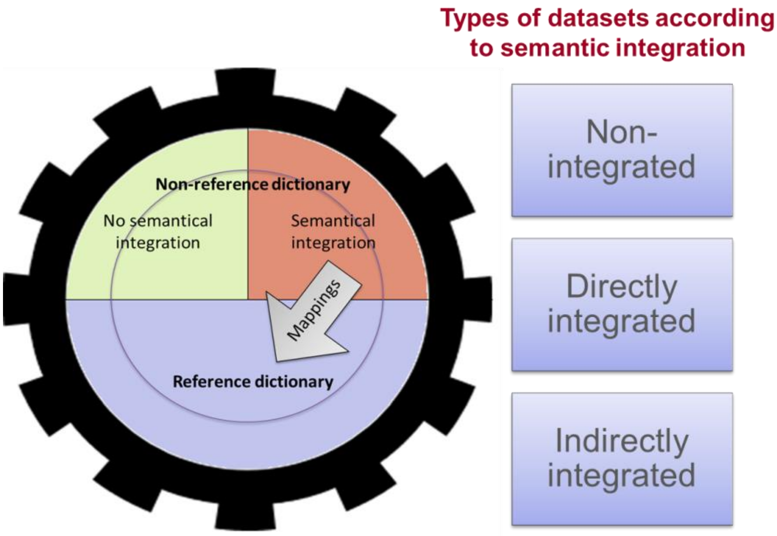


Figure 5: Different options regarding semantic integration

4.3.3 Maintenance agencies in the SDD

The concept of maintenance agencies is essential for understanding the SDD. As seen in the previous section, the SDD is divided in several parts: A number of non-reference dictionaries and one reference dictionary. Each of this parts corresponds to a maintenance agency, which is a group of people in charge of defining a maintaining a dictionary. Each maintenance agency is free to create all the concepts and datasets that it wants, following an approval process within the maintenance agency.

The SDD reference dictionary is the part of the SDD maintained by one specific maintenance agency, called ECB. Besides the reference dictionary, the SDD currently holds maintenance agencies for other owners, like RIAD, SHS, SDW team or EBA for the import of the ITS on supervisory reporting.

The below table outlines the reporting frameworks already covered in the SDD:

Table 2 - Frameworks covered in the SDD

Reporting Framework	Primary reporting	Secondary reporting	Non-reference dictionary	Maintenance agency	Reference dictionary
COREP (*)	harmonised	harmonised	DPM	EBA	SDD reference
FINREP (*)	harmonised	harmonised	DPM	EBA	SDD reference
Other EBA-ITS on SR	harmonised	harmonised	DPM	EBA	SDD reference
EBA-ITS resolution	harmonised	harmonised	DPM	SRB	SDD reference
BSI	non-harmonised	harmonised	SDMX DSD	ECB	SDD reference
MIR	non-harmonised	harmonised	SDMX DSD	ECB	SDD reference
SHSG	non-harmonised	harmonised	SDMX DSD	ECB	SDD reference
AnaCredit	non-harmonised	harmonised	-	-	SDD reference
IReF	standardised	standardised	-	-	SDD reference
RIAD	-	harmonised	-	-	SDD reference

5 Smcube Methodology - Technical Specifications and Guidelines

This section of the report describes the Information Model of the Single Multidimensional Metadata Model (SMCube), which was developed by the European Central Bank to produce its Single Data Dictionary and the Banks' Integrated Reporting Dictionary. The objective of this information model is to define an abstract model that is able to describe the structure of any type of dataset and some of the attached characteristics (such as dataset exchanges or transformation), regardless of the purpose of the collection. The SMCube methodology serves as the basis for the construction of metadata and provides the structure for metadata-driven systems. The metadata, constructed in line with the SMCube methodology, can serve as parameters for the system, so that the definition and management of new datasets is as parametrised as possible from the start, resulting in enhanced collaboration between datasets and in the minimisation of new IT developments.

Currently, many different modelling methodologies are used for defining datasets (SDMX, DPM/XBRL, etc.), this situation is not expected to change in the medium term. Given that the methodologies provide the basis for describing the datasets, different datasets described on the basis of different methodologies is a clear obstacle to data integration. SMCube establishes a new level of abstraction in addition to these approaches in order to facilitate the joint use of different datasets.

The SMCube Information Model accounts for requirements on both a technical and business basis. The following requirements have been identified to satisfy the use case described in the previous paragraph:

• Use as metadata layer for metadata-driven systems: provides system-compatible structural information, assisting information managers and end users

• Compatibility with other standards, such as DPM and SDMX: covers the greatest array of datasets and supports industry sponsored models

• Business users-driven: reflects the business needs of end users

• Historisation: follows the changes in time of defined datasets

• Complex mappings: integrate existing dictionaries and create links between similar information

• Extensibility: gives the possibility of other organisations implementing the SMCube methodology and making use of the definitions provided by the ECB in the SDD

Its pivotal role is in creating cubes, which provide the definition of datasets, in the form of the structure of the tables that shall contain the data. An example of a dataset used is shown by the following table:

Table 3 - Example Dataset

Contract identifier	ISIN	Type of instrument	Amortisation type	Inception date	Legal final maturity date	Subordinated debt	Syndicated loan identifier	Accrued interest	Currency
ABC	DE7365923618	Loan	Bullet	01/01/2015	31/12/2015	FALSE	-	100	EUR
ABC	ES3941829354	Derivative	Not applicable	01/01/2015	31/12/2015				USD

To help in explaining the structure of the dataset, we can include the following details.

<u>What are the fields (columns) of the dataset?</u>

The fields of the datasets are variables (in Table 1, the variables are the titles of the columns). Referring to Table 1, the variables are 'Contract identifier', 'ISIN', 'Type of instrument', 'Amortisation type', 'Currency', 'Inception date', 'Legal final maturity date', 'Subordinated debt', 'Syndicated loan identifier', and 'Accrued interest'.

<u>What are the allowed values for each field?</u>

The allowed values for each field are given by the domain on which a variable is defined. In other words, a domain provides the allowed values for a variable. A domain can be either enumerated, or non-enumerated.

An enumerated domain is composed of a closed list of members (values) a variable can take, whereas a non-enumerated domain does not have a list of members but acts more as a sort of data type. An example of an enumerated domain is 'Type of instrument '. Examples of members include 'Loan', 'Derivative', 'Equity' or 'Deposit'.

Non-enumerated domains are domains where the members cannot be reduced down to a finite list, but rather specify a specific data type. Examples of non-enumerated domains include 'Monetary' for currency specific numeric values, 'integer' for regular, whole numeric values or 'string' for alpha numeric values.

Sometimes further restrictions on what values are valid are put in place within domains, creating a smaller subset of that domain, which is referred to as a subdomain. For instance, if we required information on accounting portfolio in the derivatives cube, not all the values of the accounting portfolio domain are applicable, as members such as 'Cash and cash balances at central banks' are not allowed values in the derivatives cube. In the case of non-enumerated domains, the subdomains will take the form of restrictions to the allowed values. Examples of non-enumerated subdomains are 'positive monetary' (within the monetary domain) and 'string with two letters and ten numbers' (within the string domain). Subdomains can also exist when a single domain can be broken down into different levels. An example of a domain with different possible levels of breakdown is 'GeoArea', which can contain subdomains including 'Continents', 'Countries', 'NUTS regions' and 'SSM countries'.

It is important to note that at cube level, variables are always associated to a concrete subdomain. If a variable can take all members of a domain, then the subdomain that the variable is defined in the context of a cube will comprise of the whole domain.

5.1.1 What is the role of one field within a cube?

Each variable can assume one of the following three roles in the context of a cube: dimension, observation value, or attribute. Dimensions serve to identify a unique record; they are equivalent to the primary keys of a database table. Observation values are variables which provide information referring to the combination of dimensions. Attributes are variables which provide additional information that further clarifies the meaning of a single observation value.

In table 1, our dimensions include 'contract identifier' and 'instrument identifier', as the combination of the two is used to uniquely identify each record. All of the remaining variables in this table are observation values.

As we can see the table 1, it is possible to have the same dimension value for different records, given there is more than one dimension value present which makes up the primary key. As a rule we can state that when there are n dimensions present in a cube, then at most n-1 dimensions can have the same value.

Which value does currency refer to within the table?

In Table 1, 'Currency' is an observation value, meaning that it provides information onto the combination of values of 'Contract identifier' and 'Instrument identifier' or, in other words, on the cube as a whole. Therefore, 'Currency' refers to the instrument. It is important to realize that the location order of the variable 'Currency' can be arbitrary and by no means indicates what 'Currency' applies to and what its role is as the order of variables in a cube can be arbitrary. In Table 1, 'Currency' is listed immediately after the column 'Accrued interest', which might give the false impression that 'Currency' is an attribute, applying to the observation value 'Accrued interest'. But if currency has the observation value rule, that will not be the case.

If 'Currency' were an attribute, it would refer to one of the observation values in Table 1 (it would be necessary to specify to which observation value it applies). In the example, the only observation value 'Currency' could apply to would be 'Accrued interest'. In this case, the attribute 'Currency' would explain in which currency the monetary amount of the accrued interest is reported.

How to deal with non-applicable fields?

Not all cells (fields) of the table are applicable in all cases. The allowed values are specified with combinations, which serve to restrict possible values. As we can see in table 2, with the instrument type 'Cash on hands', values for accounting portfolio and residence of the counterparty are non-applicable, so no values are to be inputted.

Table 4 - Example table with non-applicable fields

Type of instrument	Accounting portfolio	Residence of the counterparty	Currency	Nominal amount
Cash on hands	x	x	EUR	50.000
Loan	Available for sale	Slovenia	USD	30.000
Deposit	Available for trading	Slovakia	GBP	80.000

For this example, there are two methods we can use to define them:

1. Create separate cubes for each instrument, or;

2. Create a single cube and use combinations to define disallowed values.

When creating combinations (of a variable and a subdomain), it is possible to:

-Define all combinations that are allowed/not allowed;
-Define single data points that are allowed/not allowed or to define regions that are allowed/not allowed.

5.1.2 Definition of a cube

A cube is defined in terms of the following elements:

- variables;
- the roles of the variables;
- the subdomains, on which the variables are defined.

For attributes it is necessary to specify the observation values to which they attributes apply. An example of the definition of a cube is presented in Table 3. The non-enumerated domains are coloured in blue.

Table 5 - Definition of a cube

Variable ID	Role	Domain	Subdomain
Contract identifier	Dimension	String	Alphanumeric up to 15 characters
Instrument identifier	Dimension	String	Alphanumeric up to 15 characters
Type of instrument	Observation	Type of	Type of instruments BIRD

	value	instrument	
Amortisation type	Observation value	Amortisation type	Amortisation type full subdomain
Currency	Observation value	Currencies	ISO 4217 standard
Inception date	Observation value	Date	DD/MM/YYYY
Legal final maturity date	Observation value	Date	DD/MM/YYYY
Subordinated debt	Observation value	Boolean	True or false
Syndicated loan identifier	Observation value	String	Alphanumeric up to 15 characters
Accrued interest	Observation value	Monetary	Amount in euros

6 The contents of the SDD

This section will document the different elements of the SDD that allow the dictionary to describe the metadata, datasets and mappings.

6.1 Core package

The core package contains the building blocks that construct the information model. The aim of the core package is to provide the elements that describe the reality of data. The core package is made up of the following elements.

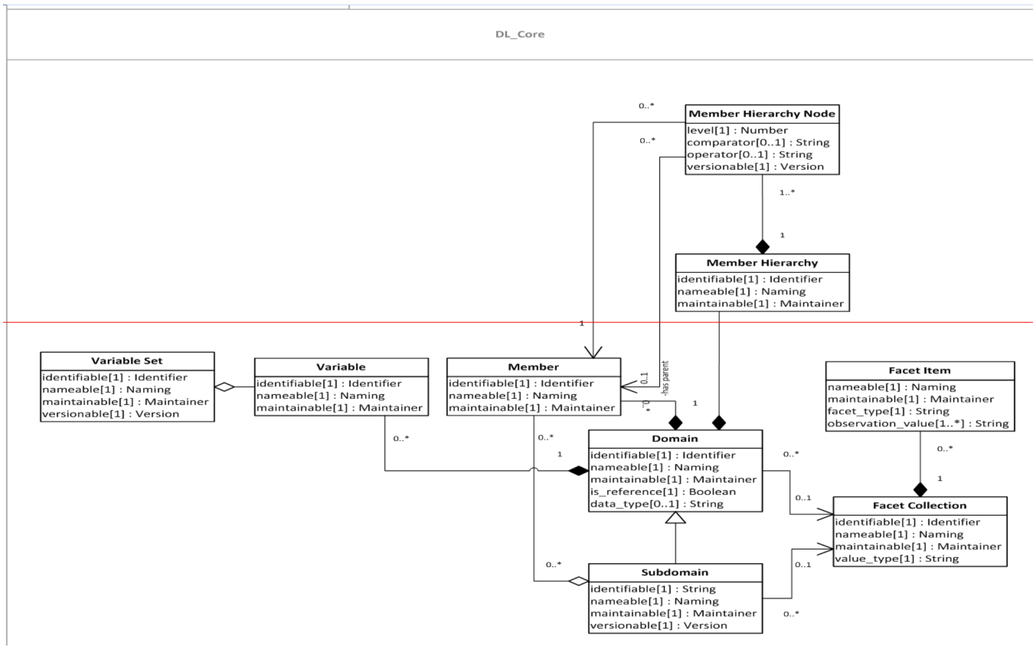


Figure 13 - The Core Package

<u>Domain</u>

Domains represent the categories in which the metadata is organised., such as geographical areas or currencies.

<u>Subdomain</u>

Subdomains are restrictions of domain, that limits possible members through the use of facets or lists of appropriate members. An example would be if on the domain ‘geographical area’ a subdomain was applied that limited the members to countries that currently use the Euro as their official currency.

<u>Member</u>

Members represent a single meaning or value that belongs within a domain. For example, for the variable ‘country’, a possible member of the domain ‘Geographical area’ would be ‘Spain’. Members can also have default values, such as -1 for non-reporting.

Variables

Variables help to add meaning to a domain. For example, the member 'Spain' may be used in different contexts such as in variables 'country or residence of the debtor' or 'country of location of activities'. Variables are represented in datasets as columns.

Variable sets and Variable set enumerations

A variable set is a superset of variables, grouping together variables, when necessary. In some cases, a dataset is described in a way that there is only one field for observation value. This is the case for the DPM, where the members of the variable metric (or amount type) specify the meaning of the observation value. The scenario is demonstrated in the following example, which depicts a FinRep-like table where the variables 'carrying amount' and 'fair value' are displayed in separate columns, as well as both included in a variable set, under the title 'Metric'.

Option 1 (Data Point Model)

Main Category	Counterparty sector	Metric	Value
Loan	Financial corporations	Carrying amount	100
Loan	Non-financial corporations	Fair value	120
Security	Financial corporations	Carrying amount	300
Security	Non-financial corporations	Fair value	290
...	...	...	...

Option 2 (without a Variable set,SmCube structure)

Main Category	Counterparty sector	Carrying amount	Fair value
Loan	Financial corporations	100	120
Security	Non-financial corporations	300	290
...	...	...	...

In both the tables above, the data displayed is equal as the variable set just allows for another representation of the data. In the SMCube methodology, there is no preferred option, and depends on the individual situations when each approach is applied. The decision how to structure the dataset does not affect the core package as the variables in both cases represent the same concepts.

As shown in the DPM example above, 'carrying amount' and ; fair value' are represented as members, not variables. In order to describe the dataset in this case, the variable 'Metric' needs to be associated with a set of variables, instead of being associated to a subdomain, as is the case with variables. This implies that when importing datasets from the DPM, metrics are imported as variables, not as members.

Member Hierarchy

Member hierarchies serve to establish a hierarchical relationship between the members of a domain. These relationships are independent of datasets, and therefore they are part of the core package. Hierarchies provide a very powerful tool for data reconciliation and production. Some use cases for hierarchies are:

- To document relations between members
- Run validations on the data based on the comparators/aggregators defined in the hierarchies
- Calculate aggregated cubes using the relations between elementary-members and aggregated-members
- Aggregated-member reconciliation between data flows (using the elementary-members involved)
- Identify the elementary-members to calculate the 'input subdomain'

Member Hierarchy Node

A member hierarchy node is a member within the context of a hierarchy. Each node belongs to a specific hierarch level, and has a parent member, except for members at the top of the hierarchy. For each node, it is possible to define the operator.

For each node, it is possible to define the operator to be used to perform operations amongst the siblings and the comparator operator to be used on the parent member to compare with its children's operation result. For example, it is possible to define a parent member in the hierarchy with the '>=' comparator for a list of children, whereby the operator attribute is equal to '+'; this expresses a rule that the parent member has to be greater or equal to the sum of the children.

Facet Collection and Facet Item

A facet enables the restriction of non-enumerated fields and the casting of variables when attributed to a subdomain or domain. A facet can contain zero or more facet items. Each facet item contains the attributes *facet_type* (valid format of the facet) and *observation_value* (valid content for the defined format – one or more).

6.2 Data Definition Package

The data definition package defines the structure of the cubes (datasets) described, and groups them into frameworks. The relationships between the tables in the package are shown in the following diagram:

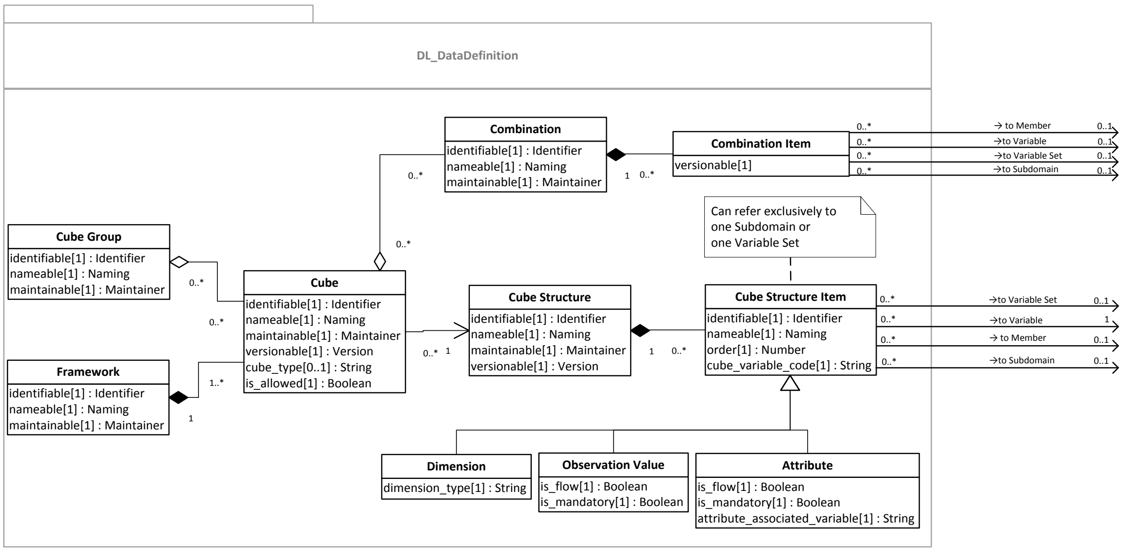


Figure 14 - The Core Package

<u>Framework</u>

The framework is a regulation or mandate for which the datasets are collected, such as AnaCredit or SHS. A framework contains cubes describing the structure of the different datasets which belong to the regulation.

<u>Cube</u>

A cube is the description of a certain dataset, and the central element of the SMCube. Cubes are typically implemented as tables in a database or as a data exchange artefact. It is important to note that cubes must be version able and form a part of a framework.

Each cube is associated to a particular type. Types include collection, production, dissemination and staging. Cubes can be restricted by combinations and can contain one or more combination items. Cubes also contained the attribute ‘is allowed’ which defines whether the combinations associated with the cube represent the allowed values (regions or data points) for the cubes, or ‘not allowed’ values.

<u>Cube Group</u>

A cube group is a collection of cubes defined by the user to facilitate the navigation of cubes. Cube groups do not reflect the grouping associated with the framework, such as counterparties or instruments.

<u>Cube Structure</u>

A cube structure is a collection of structural elements (cube structure items) which depict the multidimensional structure of a cube. Different cubes can be based on different subsets of elements of the same cube structure. This table serves to ensure compatibility with the SDMX standard, where more than one cube can be defined based on the same cube structure. In practice, in most cases (including the BIRD, AnaCredit, FinRep and SHS) there is a 1 to 1 equivalence between cube and cube structure.

Cube Structure Item

Each cube structure item represents a variable and has a specific role (observation value, dimension or attribute), and is associated with a list of possible elements via a subdomain, an implicit member and only in case of measure dimension, a variable set or an implicit variable.

There are three types of cube structure items according to their role:

Dimension

Dimensions represent identifiers of the cube, similarly to primary keys if the cube is represented in a database table. If the cube is conceptualised as a mathematical function, dimensions are the independent variable.

Observation value

Observation values are items that provide information on the full set of dimensions. In mathematical terms, they are the dependent variable, which adopt a value for each combination of values for the dimension of the cube.

Attribute

Attributes provide additional information on a single dimension or observation value. Attributes are dependent variables, but they depend on a single element (that can be dimension or variable), while the observation values depend on the combination of all values for the dimension.

Additionally, a dimension will be one of four types:

- Time dimension: Time dimensions provide the information about the time in the cube. Typically, it is a reference period for dynamic data, or valid from and valid to for registries or static data.
- Unit dimension: Unit dimensions represent the statistical unit being analysed, or the subject of the information. In cubes that represent information about banks, the unit is the dimension that specifies the bank to which the information refers.
- Measure dimension: in some cubes, the characteristics of the dimensions, normally represented by the observation values, are folded into a single variable, normally called the observation value or fact. In such cases, there is a dimension that specifies the meaning of the observation value, and that is the measure dimension. Measure dimensions are the only cube structure items that are associated with a variable set (or an implicit variable) instead of a subdomain (or implicit member).

- Breakdown dimension: All other dimensions that are not one of the types above.

Within the cube structure item table, the attribute order is mandatory and defines the order of elements within a cube structure. The attribute *cube_variable_code* represents the code of the variable in the implementation of the cube. It is linked to the distribution of the dataset. For observation values and attributes, the is_mandatory field indicates whether the element must appear in the cube. The is_flow field indicates whether a variable contains 'stock' or 'flow' values.

Combination and combination item

Combinations restrict the multidimensional space within a cube, by specifying the combinations that are allowed/not allowed. When combinations list the 'allowed' combinations in the form of pairs, such as dimension/member, they are comparable to 'time series' in SDMX and 'data point' in the DPM.

Combination must contain one or more combination items. Each combination item represents a combination of a variable, a member and a subdomain, or another member.

Cube to Combination

6.3 Mapping Package

The mapping package is responsible for linking different codification systems. The links are realised at the levels of variables, cubes and member in a variable. Figure two displays how the mapping package is structured and its relation to the core and data definition packages.

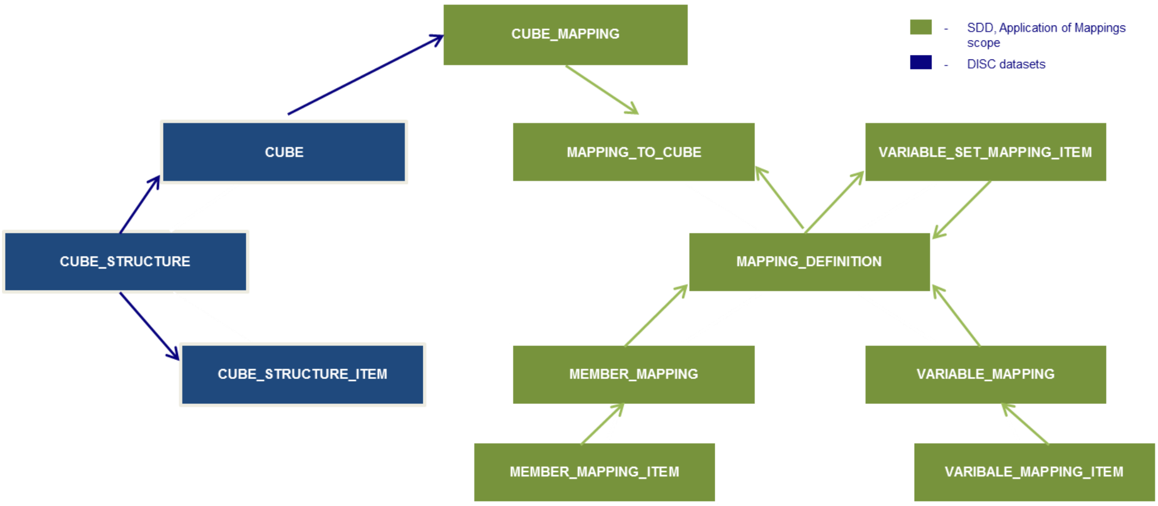


Figure 15 - The Mapping Package

Mapping Definition

Each mapping definition describes the details of mapping processes from variables in a source to variables in a destination cube. Mapping definitions contain one or more variable mapping items and must refer to one cube mapping (combination of source and destination cubes). There are four types of mappings: D (Deletion), A (Algorithm), E (Equivalence), and O (Observation value). They are subtypes of the mapping definition.

- Deletion mapping serves to eliminate one variable from the source cube in the destination one. Yet the destination structure is consistent with the source one.
- Algorithm mappings can be used to map non-listed variables. The algorithm needs to be defined within the mapping.
- Equivalence mappings map listed variables, according to the content of an equivalence table. Equivalence tables provide the mapping from source to destination members.
- Observation value mappings are used to map measure dimensions. As explained in the data definition package, the possible values associated to measure dimensions are sets of variables instead of members. Therefore, equivalence tables are not applicable in this case, since they map members.

Variable Mapping Item

Each variable mapping item defines which variables are involved in a mapping (source and destination). This element is used for all types of mapping definition.

6.4 Code Conventions

The SDD aims to be a data dictionary that can be applied for creating metadata-driven systems independently from the technology chosen to implement it. In order to facilitate that goal, the SDD identifiers should be created in a way that is compatible with most of the existing technologies.

In order to create these conventions, the following technologies have been analysed:

- Oracle
- XML
- SDMX
- XBRL

Metadata-driven systems imply normally that identifiers in the dictionary are used as identifiers in the involved physical artefacts. Therefore, the identifiers in the SDD should be as much as possible compatible with the technical constraints of the involved physical artefacts.
Additionally, some identifiers are intended to be human-readable, since it is often the case that human users use and therefore have to know the identifiers. For that reason, some conventions are added in order to facilitate this readability.

6.4.1 General conventions

6.4.1.1 Business conventions

General Business conventions
The identifiers aimed to be human readable should:
• Use the most relevant words of the concept to be identified (avoid articles, prepositions and similar)
• For each word, vowels will be dropped, unless they are the first letter of the word
• Words will be separated by underscores
• Plurals shall not be used to identify concepts (for instance, for sector use SCTR instead of SCTRS). This should also apply to the labels of the concepts
• Concepts should be as general as possible, in order to allow reusability. Therefore, the IDs should only take the essential content of the labels.

Order of some commonly used concepts
Some concepts are widely used in the SDD. In order to facilitate human readability, the concepts should preferably be placed in the same order. Ideally, SDD identifiers should follow a pattern consisting of three elements:

- Prefix modifier (optional) – gives additional information to the prime concept
- Prime concept (mandatory) – identifies the type of information being defined
- Suffix modifier (optional) – gives additional information to the prime concept.

- The prefix and suffix modifiers allow to harmonise the relative position of the additional information being provided in order to allow easier filtering of SDD identifiers. See below general guidelines to be followed whenever possible:

Prefix modifier (First place):
- Type (TYP)
- Address (ADDRSS)
- Date (DT)
- Class (CLSS)
- Name (NM)
- Country (CNTRY)
- Boolean variable (IS)
- Flow type (Accumulated – ACCMLTD; Cumulative – CMLTV)
- Amount type (Gross – GRSS, Net – NT, Total – TTL, Original – ORGNL; Residual – RSDL)
- Currency of monetary variable (CRRNCY)

Suffix modifier (Last place):
- Amount (AMNT)
- Identifier (ID)
- Code (CD)
- Status (STTS)
- Days (DYS)
- Monetary variable in currency different than Euro (CRRNCY)

6.4.1.2 Codification

All elements in the model require a code. Codes can be provided automatically, for instance using correlative numbers, or following a convention. Although identifiers are supposed to be used by systems (while users are supposed to use only descriptions) it is usual that users need to deal with codes. Therefore, many statisticians advocate for having meaningful codes. Defining conventions for codification is useful for helping the people creating new domains to find a proper code. It might be also convenient for people using the codes to make it possible to realise how the code they want to use could look like.
Codes need to be unique at a certain level, which depends on the element that is codified:

<u>Domains codification</u>
The level of uniqueness of domains applies at the level of defining authority. This means that each defining authority needs to make sure that each domain they define has a different code, although it is possible that two different institutions use the same code.

<u>Members' codification</u>
Members need to be unique at the domain level. This means that two members belonging to different domains can have the same code, which is not possible for members belonging to the same domain. It is important to note that the number of members can be quite high, and therefore

assigning a meaningful code to all of them can be quite challenging. Therefore we propose to use the following convention. In case a standard codification exists, we should use it (note that in SDMX this is not always the case, for instance, ESA institutional sectors have a code, which is not used for codifying the members). In case no standard codification exists, SDMX cross-domain codifications, maintained by SDMX, organisation should be used. If that is neither available, ECB/SDMX codes should be used. If that is neither available, Eurostat/SDMX codes should be used. Last, if none of the previous options is available, a new code should be invented, trying to follow the convention used for other members in the same domain.

Subdomains codification

Subdomains need to be unique at the domain level. Note that some subdomains can be very specific to existing datasets, with very low probability of reusing them. Therefore, in some situations it might be convenient to be able to have a standardised way to create the codes.

Variables codification

From a technical perspective, variables need to be unique at the domain level. Nevertheless, it is a common practice to have unique codes at the level of defining authority. For the reference dictionary, we propose to use the same convention as for domains: i.e. we propose to keep uniqueness at defining authority level, and to use camel cases.

Hierarchies codification

Hierarchies need to be unique at the domain level. For the SMCube, we propose to use the same convention as for subdomains.

6.5 Maintenance and Historisation in the SDD

6.5.1 Historisation process and versioning

The SDD provides metadata to describe datasets and other aspects related to them, like relations between elements of datasets. But datasets evolve over time, as well as the relations between the elements of the datasets. Keeping precise track of these changes is essential for both systems and users to be able to know exactly the metadata applicable at any given point in time. Historisation refers to the ability of one dictionary to keep track of the exact picture of the dictionary at any point in time.

The SDD provides the possibility to provide historisation in different ways, but the most relevant aspect for historisation is not related to the methodology, but to business processes. It is essential first to set up a procedure on how the SDD should be modified in order to satisfy the use cases for historisation in the most efficient way.

We can define historisation as the functionality that provides the ability to understand, for any given date, the exact definition of a dataset in terms of the allowed components for the dataset.

The SDD uses two complementary techniques for providing historisation: through versions or through validity range for individual items.

The main purposes for keeping an account of historised dataset descriptions can be summarized by the following reasons:

1. The need to know at any point in time what is the exact definition of cubes (which implies knowing the core elements that are being used at a certain point in time)

2. The need to be able to generate consistent time series, even if there are changes in the cubes

3. The need to be able to track the evolution of datasets within reporting frameworks (i.e. versioning)

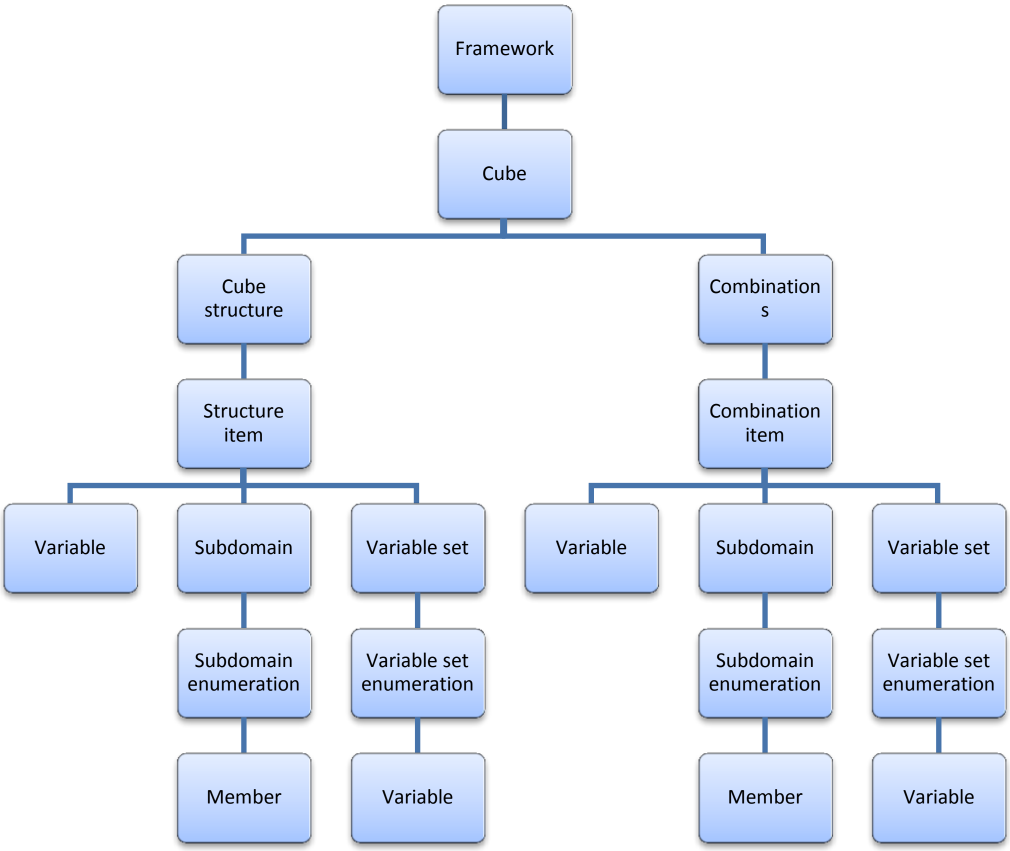
The SDD provides additional artefacts that do not serve to describe the datasets but to describe aspects related to the datasets, like hierarchies, transformations or mappings. Historisation in these cases is required in order for the systems and users to understand when these artefacts are applicable.

6.5.2 Process for Historisation of dataset descriptions

Changes in the data sets may arise for different reasons, which to the effects of setting procedures for historisation can be summarised in natural reasons (one set of elements for which information is required naturally changes) or structural reasons (the data set changes because there is a structural change in the information collected, produced or disseminated). An example of a natural change would be the creation of a new currency or country. An example of structural changes would be the addition of two types of instrument, or a new variable, to a data request. This difference is essential, since it affects the business process.

The full definition of a dataset implies the use of several artefacts, which are linked by the references among them. The following scheme shows all the artefacts involved in the definition of cubes and the references among them

Figure 16 - Artefacts involved in the definition of a cube



6.5.2.1 Natural changes in a dataset

Natural changes are comprised only by the addition or deletion of members within a dataset. These instances are related to datasets that utilise a standardised list of members as a subdomain. Examples are listed currencies, geographical regions or NACE codes. Natural changes affect all cubes which the relevant classification is used, but applies no change in the underlying regulation or in the basic normative definition of the legal basis which the cube is created.

The business process to modify a dataset due to a natural change might imply:

1. Addition of new members

    a. New member is added (in the members artefact) and associated with a specific domain

b. The new member is associated to the subdomain using the subdomain enumeration artefact. The ValidFrom date of the association is the date from which the new member may be included in the dataset

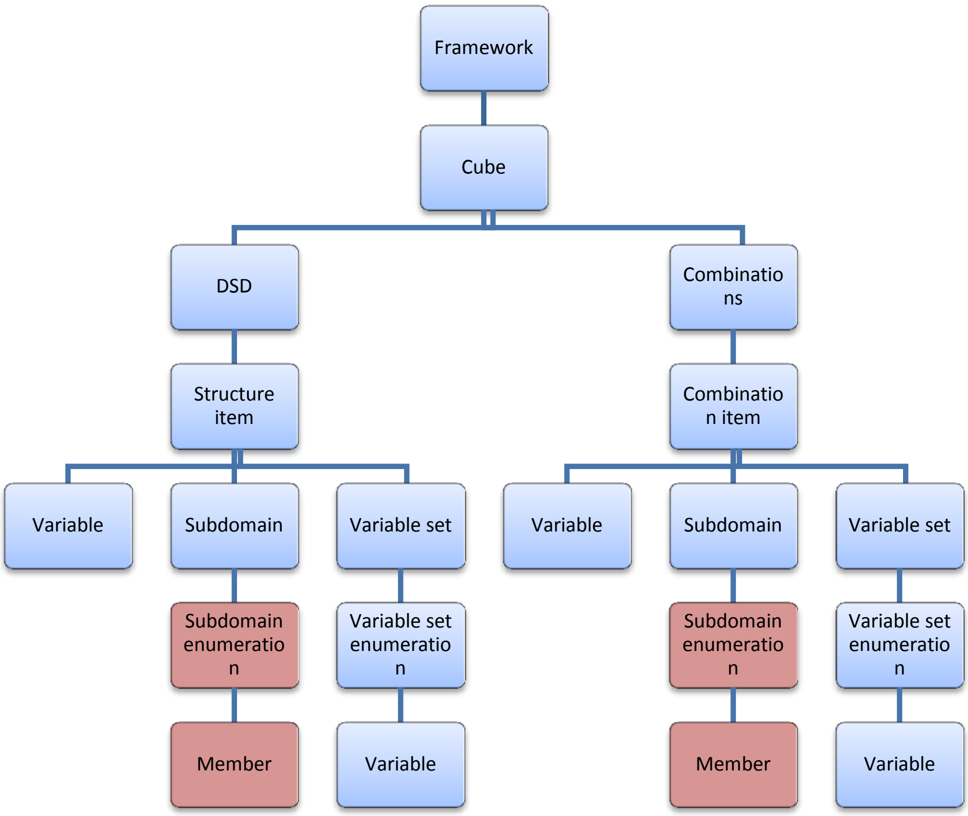
2. Removal of an existing member

a. The member is not removed from the members artefact

b. The ValidTo date of the association with the subdomain is set to the last date in which the member belongs to the dataset

In this case, a change in the subdomain enumeration, by adding/deleting the involved member to the enumeration of the subdomain from a certain date is enough. The two use cases for historisation are met, and from a business perspective the change reflects the reality: There is no real change in the definition of the cube or its structure, the simple fact is that circumstances exogenous to the cube oblige to change the set of possible members, nothing else. Therefore such a natural change does not imply an increase in the version of the cube(s) that are affected by the change.

Figure 17 - Artefacts affected by natural changes in a dataset



6.5.2.2 6.2.2 Structural changes in a dataset

Structural changes in a dataset may imply (i) the addition or removal of variables; or (ii) the addition or removal of members. Another example of pure structural change is the change in the role of a cube structure item from an observation to a dimension.

An example of the first case would be if AnaCredit starts collecting one additional attribute for each instrument. An example of the second case would be the addition of types of instruments to the AnaCredit scope. Note that, from a technical perspective, the second case is equivalent to a natural change in the dataset, so it could be processed similarly. Yet, from a business perspective, the difference is substantial as in the case of a structural change in a reporting dataset, there is a need to amend a regulatory act, while for natural changes there is not.

In the case of structural changes, creating a new version of a cube that is affected by the change is necessary to satisfy our need to be able to track the evolution of datasets. For instance, if the AnaCredit regulation is modified to add new attributes to the instruments dataset, then the version of this cube needs to be increased.

The historisation processes of other SDD artefacts are detailed as the following:

Member hierarchies nodes:

Changes in member hierarchy nodes are always considered as structural changes and therefore require new versions of the related cube(s). However an ‘alert system’ to inform interested parties about changes could be envisaged.

Variable sets:

Changes to variable sets are treated similar to changes in subdomain enumeration.

Cube, Cube structure:

Changes to cubes or related cube structures are always considered structural changes.

6.5.3 Example on treatment of historisation in the SDD

The purpose of this example is to illustrate the different ways how the same business case can be stored in the SDD differently.

6.5.3.1 Business Case Description

The case in this example is a structural change given by an additional member, a new financial instrument (TYP_INSTRMNT_3331), used in the test cube (TST_CB) as a member of related variables.

General Setup

We will consider a cube Test cube (TST_CB):

CUBE_ID	NAME	CODE	FRAMEWORK_ID	CUBE_STRUCTURE_ID	CUBE_TYPE	VALID_FROM	VALID_TO	VERSION	DESCRIPTION
TST_CB_1	Test cube	TST_CB	TST_FRMWRK	CB_STRCTR_1	C	t0	31/12/9999	1	

Figure 18 - Content of Table CUBE

Related to a specific cube structure (i.e. *CB_STRCTR_1*):

CUBE_STRUCTURE_ID	NAME	CODE	VALID_FROM	VALID_TO	VERSION
CB_STRCTR_1	Cube structure for test cube	CB_STRCTR	t0	31/12/9999	1

Figure 19 - Content of Table CUBE_STRUCTURE

This cube structure associates the cube with its related cube structure items:

CUBE_STRUCTURE_ID	CUBE_VARIABLE_CODE	VARIABLE_ID	ROLE	ORDER	SUBDOMAIN_ID
CB_STRCTR_1	TYP_INSTRMNT	TYP_INSTRMNT	D	1	RLVNT_TYP_INSTRMNT
CB_STRCTR_1	CRRNCY_DNMNTN	CRRNCY_DNMNTN	O	2	RLVNT_CRRNCS
CB_STRCTR_1	CRRYNG_AMNT	CRRYNG_AMNT	O	3	MNTRY_ALL_2D

Figure 20 - Content of Table CUBE_STRUCTURE_ITEM

Each cube structure item connects the cube structure with a variable and a subdomain. These objects are stored in the table VARIABLE and SUBDOMAIN respectively. Both, the variables and the subdomain again refer to their related domain (stored in the table DOMAIN).

VARIABLE_ID	CODE	NAME	DOMAIN_ID	DESCRIPTION
TYP_INSTRMNT	TYP_INSTRMNT	Type of instrument	TYP_INSTRMNT	Classification of the instrument according to the type of contractual terms agreed between the parties.
CRRNCY_DNMNTN	CRRNCY_DNMNTN	Currency denomination of instruments	CRRNCY	Currency denomination of instruments, in accordance with the ISO's 4217 standard
CRRYNG_AMNT	CRRYNG_AMNT	Carrying amount	MNTRY	The carrying amount in accordance with Annex V to Implementing Regulation (EU) No 680/2014.

Figure 21 - Content of Table VARIABLE

SUBDOMAIN_ID	NAME	DOMAIN_ID	IS_LISTED	IS_NATURAL	CODE
RLVNT_CRRNCS	All relevant currencies	CRRNCY	TRUE	FALSE	RLVNT_CRRNCS
RLVNT_TYP_INSTRMNT	All members of TypIns	TYP_INSTRMNT	TRUE	FALSE	RLVNT_TYP_INSTRMNT
MNTRY_ALL_2D	Positive and negative monetary amounts with 2 decimals	MNTRY	FALSE	FALSE	MNTRY_ALL_2D

Figure 22 - Content of Table SUBDOMAIN

DOMAIN_ID	NAME	CODE	DATA_TYPE	IS_ENUMERATED
CRRNCY	Currencies	CRRNCY	string(20)	TRUE
TYP_INSTRMNT	Type of instrument	TYP_INSTRMNT	integer(6)	TRUE
MNTRY	Monetary	MNTRY	number	FALSE

Figure 23 - Content of Table DOMAIN

Finally the members each domain (in case of an enumerated domain) are stored in the table MEMBER.

MEMBER_ID	CODE	NAME	DOMAIN_ID	DESCRIPTION
CRRNCY_AUD	AUD	Australian Dollar	CRRNCY	
CRRNCY_CHF	CHF	Swiss franc	CRRNCY	
CRRNCY_EUR	EUR	Euro	CRRNCY	
CRRNCY_GBP	GBP	UK pound sterling	CRRNCY	
CRRNCY_JPY	JPY	Japanese yen	CRRNCY	
CRRNCY_USD	USD	US dollar	CRRNCY	
TYP_INSTRMNT_1000	1000	Deposits other than reverse repurchase agreements	TYP_INSTRMNT	Deposits as defined in paragraph 5.79 of Annex A to Regulation (EU) No 549/2013 other than reverse repurchase agreements
TYP_INSTRMNT_1001	1001	Revolving credit other than overdrafts and credit card debt	TYP_INSTRMNT	Credit that has the following features: (i) the debtor may use or withdraw funds up to a pre-approved credit limit without giving prior notice to the creditor; (ii) the amount of available credit can increase and decrease as funds are borrowed and repa
TYP_INSTRMNT_1002	1002	Credit lines other than revolving credit	TYP_INSTRMNT	Credit that has the following features:(i) the debtor may use or withdraw funds up to a pre-approved credit limit without giving prior notice to the creditor;(ii) the credit may be used repeatedly; and(iii) it is not revolving credit,
TYP_INSTRMNT_1003	1003	Reverse repurchase agreements	TYP_INSTRMNT	Reverse repurchase agreements as defined Part 2.14 of Annex V to Implementing Regulation (EU) No 680/2014.
TYP_INSTRMNT_1004	1004	Loans other than overdrafts, convenience credit, extended credit, credit card credit, revolving credit other than credit card credit, reverse repurchase agreements, trade receivables and financial leases	TYP_INSTRMNT	Loans, as defined in paragraphs 5.112, 5.113 and 5.114 of Annex A to Regulation (EU) No 549/2013, except overdrafts, convenience credit, extended credit, credit card credit, revolving credit other than credit card credit, reverse repurchase agreements, tr
TYP_INSTRMNT_20	20	Overdrafts	TYP_INSTRMNT	
TYP_INSTRMNT_51	51	Credit card debt	TYP_INSTRMNT	
TYP_INSTRMNT_71	71	Trade receivables	TYP_INSTRMNT	
TYP_INSTRMNT_80	80	Finance leases	TYP_INSTRMNT	

Figure 24 - Content of Table MEMBER

On the other hand, the members associated with a particular subdomain are stored in the table SUBDOMAIN_ENUMERATION.

MEMBER_ID	SUBDOMAIN_ID	VALID_FROM	VALID_TO
CRRNCY_AUD	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_CHF	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_EUR	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_GBP	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_JPY	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_USD	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_Z05	RLVNT_CRRNCS	02/01/1900	31/12/9999
TYP_INSTRMNT_1000	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1001	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1002	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1003	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1004	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_20	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_51	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_71	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_80	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999

Figure 25 - Content of Table SUBDOMAIN_ENUMERATION

6.5.3.2 Treatment of Historisation

In this section we will illustrate the treatment of historisation given the described initial setup. The changes will be applicable at time t1 > t0 (= initial setup time).

First of all the new member needs to be defined in the table MEMBER as illustrated in the next figure.

MEMBER_ID	CODE	NAME	DOMAIN_ID	DESCRIPTION
CRRNCY_AUD	AUD	Australian Dollar	CRRNCY	
CRRNCY_CHF	CHF	Swiss franc	CRRNCY	
CRRNCY_EUR	EUR	Euro	CRRNCY	
CRRNCY_GBP	GBP	UK pound sterling	CRRNCY	
CRRNCY_JPY	JPY	Japanese yen	CRRNCY	
CRRNCY_USD	USD	US dollar	CRRNCY	
TYP_INSTRMNT_1000	1000	Deposits other than reverse repurchase agreements	TYP_INSTRMNT	Deposits as defined in paragraph 5.79 of Annex A to Regulation (EU) No 549/2013 other than reverse repurchase agreements
TYP_INSTRMNT_1001	1001	Revolving credit other than overdrafts and credit card debt	TYP_INSTRMNT	Credit that has the following features: (i) the debtor may use or withdraw funds up to a pre-approved credit limit without giving prior notice to the creditor; (ii) the amount of available credit can increase and decrease as funds are borrowed and repa
TYP_INSTRMNT_1002	1002	Credit lines other than revolving credit	TYP_INSTRMNT	Credit that has the following features:(i) the debtor may use or withdraw funds up to a pre-approved credit limit without giving prior notice to the creditor;(ii) the credit may be used repeatedly; and(iii) it is not revolving credit,
TYP_INSTRMNT_1003	1003	Reverse repurchase agreements	TYP_INSTRMNT	Reverse repurchase agreements as defined Part 2.14 of Annex V to Implementing Regulation (EU) No 680/2014.
TYP_INSTRMNT_1004	1004	Loans other than overdrafts, convenience credit, extended credit, credit card credit, revolving credit other than credit card credit, reverse repurchase agreements, trade receivables and financial leases	TYP_INSTRMNT	Loans, as defined in paragraphs 5.112, 5.113 and 5.114 of Annex A to Regulation (EU) No 549/2013, except overdrafts, convenience credit, extended credit, credit card credit, revolving credit other than credit card credit, reverse repurchase agreements, tr
TYP_INSTRMNT_20	20	Overdrafts	TYP_INSTRMNT	
TYP_INSTRMNT_51	51	Credit card debt	TYP_INSTRMNT	
TYP_INSTRMNT_71	71	Trade receivables	TYP_INSTRMNT	
TYP_INSTRMNT_80	80	Finance leases	TYP_INSTRMNT	
TYP_INSTRMNT_3331	3331	Newest financial instrument	TYP_INSTRMNT	

Figure 26 - Content of Table MEMBER after adding the new member

<u>Evolution of the original subdomain:</u>

Given this approach the subdomain is extended such that it also comprises the new member using the relation between members and subdomains in the table SUBDOMAIN_ENUMERATION. Please note that this procedure is exactly the one that would be carried out in the case of a natural change.

MEMBER_ID	SUBDOMAIN_ID	VALID_FROM	VALID_TO
CRRNCY_AUD	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_CHF	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_EUR	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_GBP	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_JPY	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_USD	RLVNT_CRRNCS	02/01/1900	31/12/9999
CRRNCY_Z05	RLVNT_CRRNCS	02/01/1900	31/12/9999
TYP_INSTRMNT_1000	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1001	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1002	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1003	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_1004	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_20	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_51	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_71	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_80	RLVNT_TYP_INSTRMNT	02/01/1900	31/12/9999
TYP_INSTRMNT_3331	RLVNT_TYP_INSTRMNT	t1	31/12/9999

Figure 27 - Content of Table SUBDOMAIN_ENUMERATION after adding new member

Please note the value t1 in the valid from date of the new instance. This allows us to extract the configuration of the subdomain before, as well as after the change.

Due to the fact that this is a structural change the person manipulating the SDD needs enforce a new version of the affected cube which is applied in the table CUBE.

CUBE_ID	NAME	CODE	FRAMEWORK_ID	CUBE_STRUCTURE_ID	CUBE_TYPE	VALID_FROM	VALID_TO	VERSION	DESCRIPTION
TST_CB_1	Test cube	TST_CB	TST_FRMWRK	CB_STRCTR_1	C	t0	31/12/9999	1	
TST_CB_2	Test cube	TST_CB	TST_FRMWRK	CB_STRCTR_2	C	t1	t0	2	

Figure 28 - Content of Table CUBE after providing a new version number

Please note that – in principle – we could also enforce a new version in all artefacts (that are versionable) 'between' the cube and the member / subdomain. In this case the instance of the table CUBE_STRUCTURE would also be affected by this change.

6.6 Process of maintaining and updating the SDD

Following on from the previous section on historisation, one of the main tasks involved in this traineeship was the maintenance and governance of the metadata stored in the SDD. With the dictionary reprising information for a multitude of reporting frameworks, it also handles the requests for updating the databases that have been directly integrated, being the AnaCredit and RIAD frameworks.

The SDD team sits within a division in DG-S, called analytical credit and master data (AMA). The division manages different areas of statistical and supervision information as parts of a single system. Along with the SDD, the other teams that make up the rest of this division include

6.6.1 Register of institutions and affiliates data (RIAD)

RIAD is a multipurpose ESCB-wide dataset of reference data of legal entities and other organisational units. RIAD maintains information on more than 15 million entities including monetary financial institutions such as banks, investment funds and insurance corporations. The database consists of a comprehensive data model, which includes reference data on units plus relationships among them

- Identification – such as identifiers, address, etc.
- Stratification – such as industrial activity, geographical location, etc.
- Demographic developments – such as birth/closure date, split, merger, etc.
- Relationships between units – such as ownership, control, management,

The RIAD database was integrated into the SDD, originally being converted from SDMX. In this system, each entity is represented as one row and attributes of the institution are converted to the fields, as shown in the following figure:

RIAD_ENTTY_ID	CNTRY	DT_BRTH	DT_CLS	NM_ENTTY	LEI	ENTRPRS_SZ	IS_LSTD	INSTTTNL_SCTR	VLD_FRM	VLD_T
111234	ES	17/05/2017	31/12/9999	ABCDE GmBH	1234789678afadsvfja	4		S11	17/05/2017	19/07/2017
111234	ES	17/05/2017	31/12/9999	ABCDE GmBH	1234789678afadsvfja	4	TRUE	S11	20/07/2017	31/12/9999

Figure 29 - SDD compliant RIAD dataset

6.6.2 AnaCredit

The analytical credit dataset (Anacredit) is a dataset with detailed information on individual bank loans in the euro area. It uses data and national credit registers to achieve a harmonised database that supports several central banking functions, such as decision-making in monetary policy and macroprudential supervision. The regulation includes 88 attributes surrounding credit, credit risk and accounting data and is reported on a loan-by-loan basis.

6.6.3 SDD daily maintenance of RIAD & AnaCredit

The SDD team received on average 90-100 e-mail monthly from the AnaCredit and RIAD teams, requesting changes and updates to the respective databases. From these changes, six SDD extractions are provided to AnaCredit and RIAD, in order to update their transactional systems. The requests vary in scope and complexity.

The Anacredit databases remain fairly static in terms of elements stored. Updates to already implemented elements are advised against as it can have an unwanted effect on the collection process of real data, causing errors in the transactional systems.

6.6.3.1 Example of AnaCredit change:

A request was made from the Central Bank of Denmark, to remove two members from the list of legal forms, these being:

- DK10: Sole proprietorship
- DK15: Personal Small Business

Resulting changes:

Member			
CODE	**NAME**	**DOMAIN_ID**	**DESCRIPTION**
DK10	Enkeltmandsvirksomhed	LGL_FRM	Sole proprietorship
DK15	Personlig Mindre Virksomhed	LGL_FRM	Personal small business

Subdomain Enumeration			
MEMBER_ID	**SUBDOMAIN_ID**	**VALID_FROM**	**VALID_TO**
LGL_FRM_DK10	LGL_FRM_RIAD_ANCRDT	1900-01-01 0:00:00	9999-12-31 0:00:00
LGL_FRM_DK15	LGL_FRM_RIAD_ANCRDT	1900-01-01 0:00:00	9999-12-31 0:00:00

6.6.3.2 Example of RIAD change

A request was made by the RIAD team, on behalf of colleagues in DG-M (market operations) and DG-Rm (risk management) to include new variables into the database of institutions aiming to identify entities that are a regional government or local authority, which are treated as exposures to central government in accordance with Article 115(2) of Regulation (EU) No 575/2013 (CRR).

The variables requested included:

- Addition of an attribute to flag the 'CRR public sector entity status' of entities relevant for credit claim management - in accordance with the Credit quality assessment criteria for marketable assets in the absence of a credit assessment provided by an accepted ECAI as established in Article 87 of the General Documentation (Guideline ECB/2014/60)
  --> Articles 115(2) of Regulation (EU) No 575/2013 (CRR)

- Addition of an attribute to flag the 'CRR public sector entity status' of entities relevant for credit claim management - in accordance with the Credit quality assessment criteria for marketable assets in the absence of a credit assessment provided by an accepted ECAI as established in Article 87 of the General Documentation (Guideline ECB/2014/60)
  --> Article 116(4) of Regulation (EU) No 575/2013 (CRR)R)

In this case, we found that the concepts were already used in the counterparties cube, as a variable used in the calculation Exposure class and Credit quality status for Common reporting requirements. The integration of many reporting frameworks into the SDD allowed for the concepts to be re-used in other areas of reporting, thus displaying the advantage of a single dictionary.

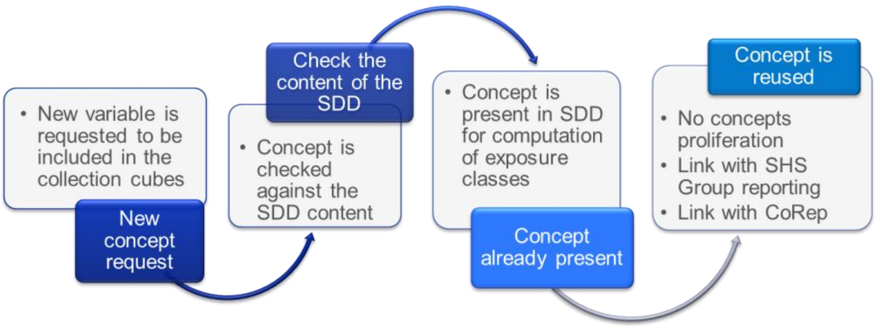


Figure 30- Validation process for new concept

We decided to re-use existing properties rather than create new ones, as it was assessed that the definitions coincided. It was proposed, however, that the variables be altered from a flag (indicating true/false) to enumerated subdomains displaying the values identified in the following table:

Table 6 - Variables to be added in SDD

SUBDOMAIN_ID	Name	Members	Description
IS_CNTRL_G	Is	**0** = The entity is not a regional government or	Indicates whether a

VRNMNT_TRTD_LG	regional government or local authority treated as central government	local authority **1** = The entity is a regional government or local authority which is treated as exposure to its central government in accordance with Article 115(2) of Regulation (EU) No 575/2013 (CRR) **2** = The entity is a regional government or local authority which is not treated as exposure to its central government in accordance with Article 115(2) of Regulation (EU) No 575/2013 (CRR)	regional government or local authority is treated for capital requirements purposes as exposure to its central government in accordance with Articles 115(2) of Regulation (EU) No 575/2013 (CRR)
IS_LCL_GVRNMNT_TRTD_PS	Is public sector entity treated as regional government or local authority	**0** = The entity is not a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) **1** = The entity is a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) which is treated as exposure to the regional government or local authority in accordance with Article 116(4) of Regulation (EU) No 575/2013 (CRR) **2** = The entity is a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) which is not treated as exposure to the regional government or local authority in accordance with Article 116(4) of Regulation (EU) No 575/2013 (CRR)	Indicates whether a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) is treated for capital requirements purposes as exposure to the regional government or local authority in accordance with Article 116(4) of Regulation (EU) No 575/2013 (CRR)
IS_CNTRL_GVRNMNT_TRTD_PS	Is public sector entity treated as central government	**0** = The entity is not a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) **1** = The entity is a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) which is treated as exposure to the central government in accordance withArticle 116(4) of Regulation (EU) No 575/2013 (CRR) **2** = The entity is a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) which is not treated as exposure to the central government in accordance with Article 116(4) of Regulation (EU)	Indicates whether a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) is treated for capital requirements purposes as exposure to the central government in accordance with Article 116(4) of Regulation (EU) No 575/2013 (CRR)

		No 575/2013 (CRR)		

Creating new subdomains in the dictionary required changes to the core package, as well as the related cubes and variables. All changes made in this case can be found in Annex I.

6.6.3.3 Procedures for the SDD maintenance using the SDD IT tool

The process of making the changes in the SDD are done through an IT application, which allows users to view the tables, as they are stored in DISC. In the SDD IT tool work is organised by projects. Projects are mainly related to data frameworks, and act as a container for all the tasks related to the creation or modification of a data framework. Examples of a project could be: creating AnaCredit, or future modifications of BSI. Projects are created in the context of a maintenance agency, by the owner of the maintenance agency. Projects themselves have their own owner, who is responsible (validates and approves) for the activity in the project, which is to be carried out by the dictionary administrators (create, update, delete elements). In case of supervisory data the SDD team will be in charge of importing and maintaining the XBRL taxonomy into the SDD model.

The SDD IT tool also defines specific roles relevant for the maintenance process:

- Maintenance agency owner (MAO) – assigned with the creation of the relevant maintenance agency and has the final say in approving changes to dictionaries;
- Project owner (PO) – assigned by the MAO when a project is initiated and specific to the project. Responsible for overseeing, implementing and approving Create, Read, Update and Delete (CRUD) elements before they reach the MAO for final approval;
- Dictionary administrator (DA) – assigned by the PO. Responsible for validating and implementing the CRUD elements. Project owners are always also Dictionary administrators (but Dictionary administrators are not always Project owners).

The core of the procedure to modify the SDD using the tool is as follows:

1. The dictionary administrators (and the project owner in his role of a DA) are the ones to implement CRUD elements, representing the changes and structure of the actual dataset underlying the dictionary. DAs also run validations on proposed changes to be made to the dictionary, before submitting them to the PO.
2. All of the CRUD elements then have to be approved by the PO, who also validates them. These validations will be automated with the launch of the SDD IT Tool. If the validations do not find valid content – the PO sends the project back to the DAs for revision.

3. If the validations find valid content, the next step is to check whether the project affects the core content of a dictionary, and if it does whether changes to the core content have already been approved by the MAO. If changes have been already approved or the core content is not affected by the changes made by the project – it is moved to production. Otherwise, a notification of unapproved content is sent to the MAO.

4. At this stage the MAO reviews the contents of the project and can either approve it, and thus move it to production, or not approve it and send it back to the DAs for revision.

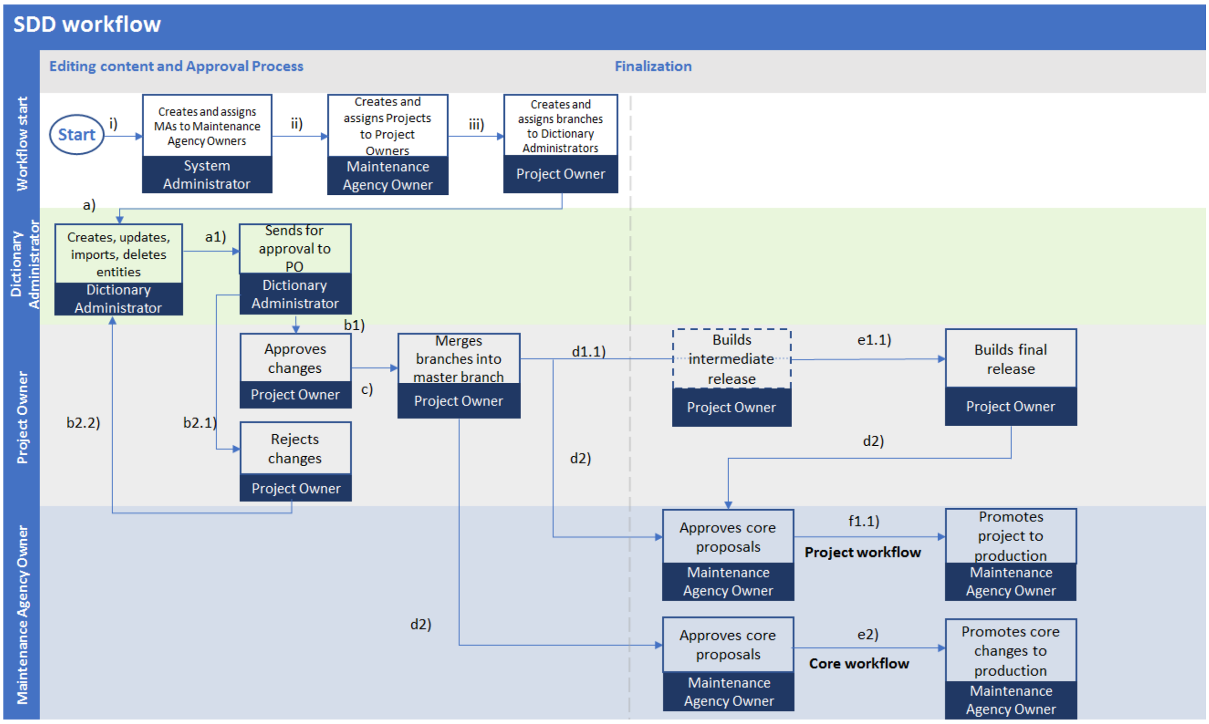


Figure 31 - SDD IT Tool workflow

7 Banks Integrated Reporting Dictionary

The other major task undertaken by the SDD/BIRD team is the development of the Banks Integrated Reporting Dictionary. As mentioned in the introduction, this project is part of the long-term objective of creating an integrated reporting framework for all supervisory reporting undertaken by reporting institutions.

7.1 Introduction

The BIRD initiative is a key component of the Statistics Committee's strategy to streamline the overall process of reporting from banks to national and European authorities, to facilitate the integration of the data management systems of NCBs, NCAs, and the ECB and to improve the consistency and quality of the information reported by banks for various purposes.

The aim is better quality data at source together with more efficient and, in the longer run, less costly report production by the banks. The authorities can rely on more consistent and harmonized data . In its concrete implementation, the BIRD will provide recommendation to banks describing the information they need to extract from their operational systems and the transformation rules applied to these data in order to fulfil their reporting requirements. This multi-purpose "input layer", which can be implemented in the data warehouses of the banks, harmonises the structure, classification, and format of the data sourced from the banks' own operational systems.

Having an input approach is therefore a common goal for both authorities and the banking industry. Such input approach could ideally be developed by the industry alone (the banks), as it has the required know-how and because banks are the users of the BIRD.

The objective of the BIRD initiative is to support commercial banks in integrating their own data management systems. As the consistency and quality of data produced for end-users in the Eurosystem and SSM depends on the quality of the data received from the reporting banks, the BIRD initiative is a crucial step for the overall strategy of the STC.

The aim is for a number of statistical and supervisory reporting requirements to be covered by the BIRD, including:

- the ECB's collection of granular credit and credit risk data (AnaCredit);
- the ECB's Securities Holdings Statistics (SHS);
- the ECB's Monetary Financial Institutions' Balance Sheet Items (BSI) Statistics;
- the ECB's Monetary Financial Institutions' Interest Rate (MIR) Statistics;
- the balance of payments and national accounts;
- the additional requirements under the Single Supervisory Mechanism;
- the EBA's Implementing Technical Standards (ITS), including Common Reporting (COREP) and Financial Reporting (FINREP).

7.1.1 How the BIRD can help banks

Some of the advantages for reporting banks include:

Different reports can be produced from a single input layer.

This reduces the reporting burden for the banks, improves the consistency and quality of their data and removes the need for them to manage each mandatory data collection separately.

Well-defined transformation rules

The transformation rules applied include the calculations used to obtain certain regulatory figures. These provide a univocal interpretation of regulations and greater clarity, while also increasing compliance with the regulatory requirements.

Less time and effort is needed to analyse and comply with new reporting requirements

Whenever a new statistical or supervisory regulatory framework is introduced or an existing one is updated, banks are expected to independently interpret the framework, extract data from their internal systems and then transform the data to obtain the final figure required by the regulation. However, it is not always easy for banks to determine which source data to use and how to correctly process it.

Differences in how banks interpret specific regulations can have a negative impact on the quality of their output data and make it more difficult to compare different banks' data. It is also costly and time-consuming for each bank to study new or revised legal acts separately, therefore cooperation between banks can bring significant efficiency gains.

7.1.2 High level architecture

The target BIRD-based system is aiming to provide a sort of intelligent ETL (Extract-Transform-Load) process, driven by the metadata provided by the BIRD.
The system itself shall be able to get data from external systems (operational systems), transform the data, and produce the information required in the final reports. This information needs to be somehow transmitted to the regulators. In order for the system to be automatized, it also needs to connect to the BIRD repository, which contains the BIRD documentation, and shall allow access by human users. Therefore, the main components of the BIRD-based system can be summarised as:

- Database: Containing the information layers
- Roles: Human users accessing and interacting with the system
- Interfaces with other systems: To collect data and metadata, as well as to transmit the final reports
- Processes: Data collection and transformation processes

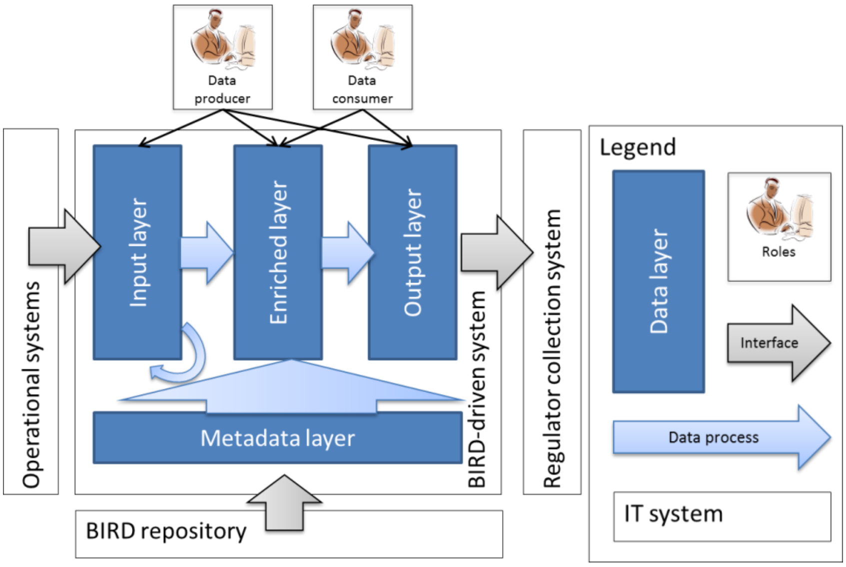


Figure 32 - BIRD Process

7.2 BIRD Model

The BIRD documents a dynamic data process in which data are inputted, validated and subsequently transformed in order to generate the final datasets. These datasets reflect the primary reporting requirements, i.e. the information that banks need to provide to the national central bank. However, when primary reporting requirements are not harmonized across Europe (e.g. AnaCredit or BSI) the datasets reflect the secondary reporting requirements, i.e. the information that national central banks need to provide to the authorities (e.g. the ECB).

The BIRD provides a formal description of the datasets (the input layer and the output layer) and the validation and transformation rules.

The BIRD data model is made up of four distinct layers, with three phases taking place between the layers. The four layers are:

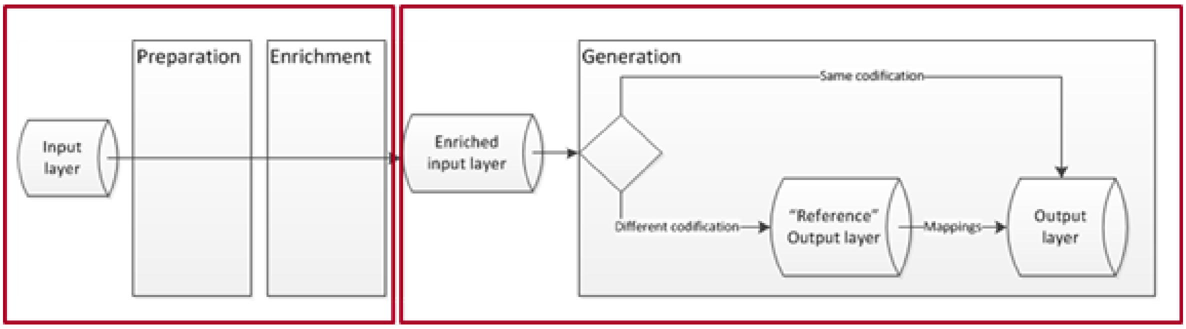


Figure 33 - BIRD Model

7.2.1 The Input Layer (IL)

This layer is a harmonised entity relationship model (ERM) that contains the information required from a bank's internal systems in order to generate the output layer(s). Among other things, this layer contains information about entities (e.g. debtors and protection providers), commitments (e.g. credit facilities), instruments (e.g. loans, securities and deposits) and protections (e.g. financial protection or physical protection such as real estate protection) as well as aggregated. The below figure displays the main categories of the cubes contained.

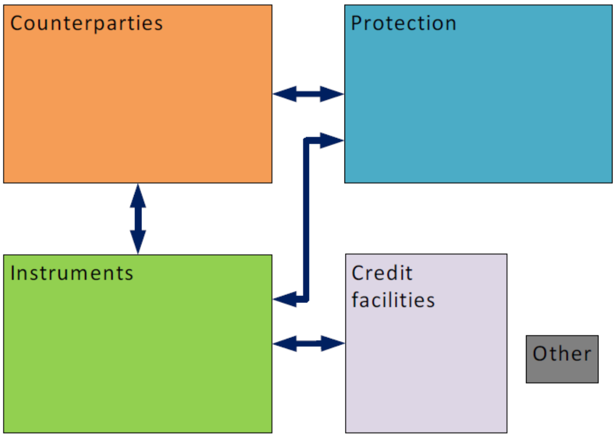


Figure 34 - BIRD Input Layer

<u>Counterparties</u>

The central entity of this block is the Counterparties entity (CNTRPRTS). Counterparties represent the minimum unit to which an instrument can be associated, and therefore the level of granularity of the counterparties shall be the minimum unit for which reporting requirement exists. The counterparties cube contains variables containing information such as Country (CNTRY), Institutional Sector (INSTTTNL_SCTR), Economic activity (ECNMC_ACTVY) and annual turnover (ANNL_TRNVR).

Counterparties are uniquely identified by an internal identifier, i.e. the *Counterparty identifier (CNTRPRTY_ID)*. Head office undertakings, ultimate parent undertakings and immediate parent undertakings have to be represented also as counterparties.

The probabilities of default associated to one counterparty may differ depending on the observed agent, and therefore this information is provided in a separate entity (CNTRPRTS_PD)

For each counterparty, any number of codes can be attached by using the *Counterparty codes (CNTRPRTY_CDS)* entity.

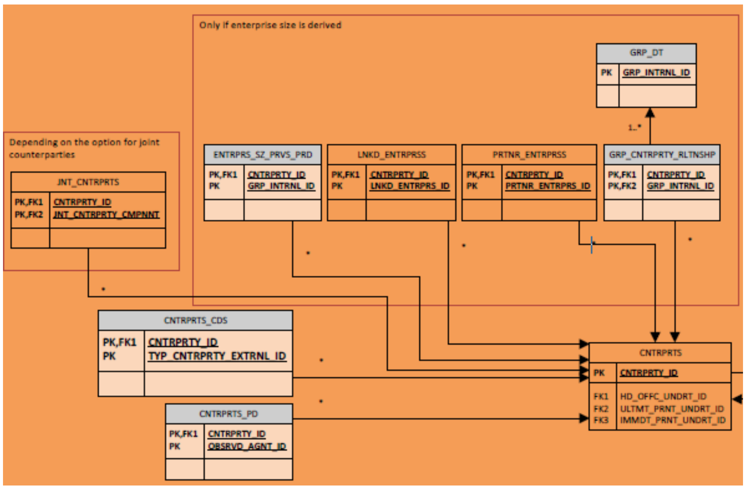


Figure 35 - Counterparties block

Instruments

The instruments block contains information about financial instruments, including reference data about securities and about securitisations and other credit transfers.

The block contains three main entities with the information about financial instruments:

• Loans (LN) includes information on the loans granted by the observed agent. The loans are separated in different input cubes according to the type of instrument.

• Positive current accounts (PSTV_CRRNT_ACCNT) includes information on the credit balances on current accounts.

• Owned securities (OWND_SCRTY) contains the information on securities that are held by the observed agent.

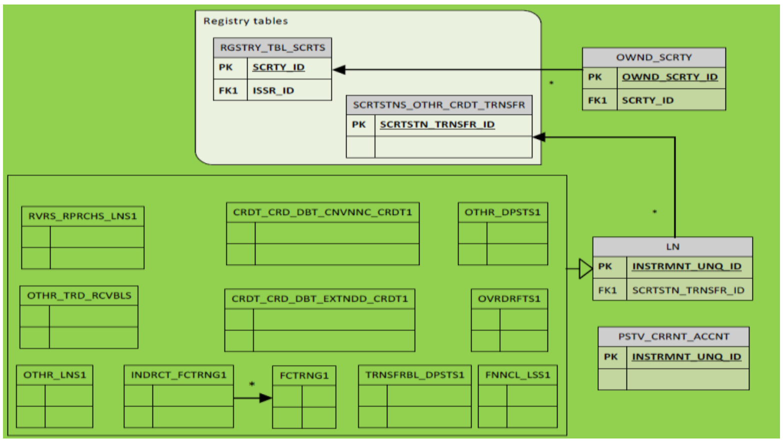


Figure 36 - Instruments

Protections

This block includes all the items that may serve as protection received by the entity for risk mitigation purposes. The level of granularity is the protection item. Protections are represented in five cubes, according to the nature of the protection. For real estate collateral, the actual protection is the claim, which represents a pool of real estate items. The different protection cubes all contain similar variables relating to the type of protection (TYP_PRTCTN), type of transaction (TYP_TRNSCTN) and the

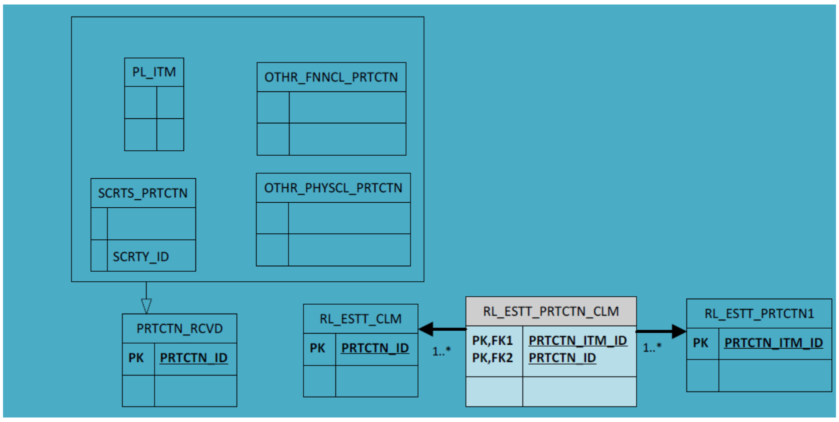


Figure 37 - Protections

7.2.2 The Enriched Input Layer

The Enriched Input Layer is another ERM and acts as an intermediate layer between the Input Layer and the Reference Output Layer. It consists of the information from the Input Layer and some additional information (such as the results of particular derivations where the input parameters of the derivation may no longer be needed and are therefore not present in the Enriched Input Layer).

The following picture shows the ERM for the enriched input layer, reached after the preparation and enrichment phases.

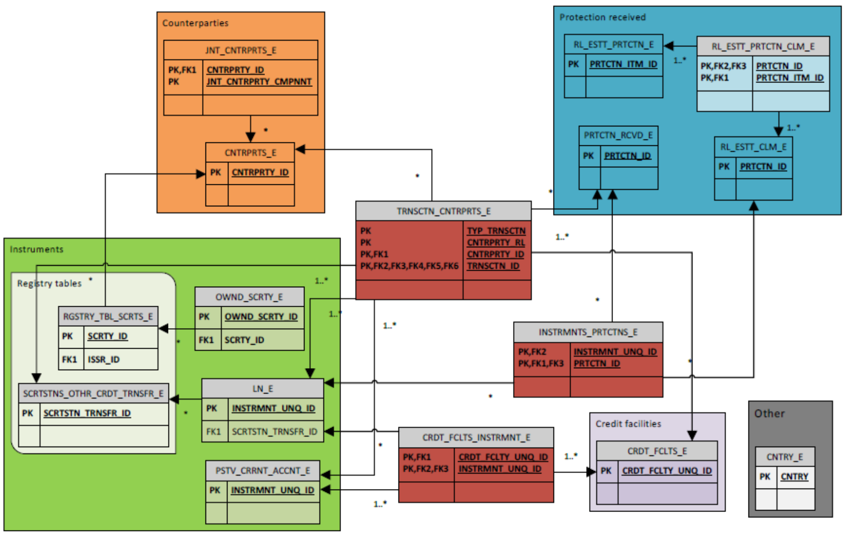


Figure 38 - The Enriched Input Layer

7.2.3 The Reference Output Layer

The Reference Output Layer describes the Non-reference Output Layer using reference codes (i.e. the codes used in the Input Layer). The relationship between the Reference Output Layer and the Non-reference Output Layer is represented by mappings, as described by the process outlined in phase 3 below.

7.2.4 The Non-reference Output Layer

The Non-reference Output Layer describes the reporting requirements as defined in the related documents (e.g. regulations, directives, guidelines and manuals), sometimes using different codification systems (e.g. EBA ITS and Statistical Data and Metadata eXchange SDMX).

Although the output layer is part of the BIRD process, defining the output requirements (e.g. regulations) does not fall within the scope of the BIRD. Instead, the reporting requirements are translated into the language of the BIRD (SMCube methodology) and are therefore represented as cubes (e.g. the EBA's Implementing Technical Standards related to FINREP, which are described in the data point model (DPM)/XBRL, are extracted and included in the dictionary.

The three phases are:

7.2.5 Phase 1 (IL → EIL)

The first two phases focus on the application of transformation rules to the cubes and resulting data. These transformation rules are written in validation and transformation language (VTL). VTL is a standard language for defining validation and transformation rules (set of operators, their syntax and semantics) for any kind of statistical data. VTL is a formal programming language documented by the SDMX community, in line with their reference manual.[14] VTL is used to describe the necessary manipulations of data sets in order to transform the Input Layer into the Enriched Input Layer, and The Enriched Input Layer into the Reference Output Layer.

All transformations are stored in the BIRD database in three tables: Transformation scheme, transformations and transformation node.

<u>Transformation scheme</u>

Transformations are grouped into transformation schemes, which are sets of transformations designed to provide meaningful results for the user. Transformation schemes group transformations that share some functional or business purpose. An example of a transformation scheme is the derivation of the enterprise size, which involves several different transformations that provide the user with the final enterprise size.

<u>Transformation</u>

[14] Link

Transformations are the most basic element of the calculations, and consist of a statement assigning the outcome of an expression to a VTL element. Transformations (except for procedure calls) therefore have the following structure: transformationResult := expression

Transformation node

Each element of a Transformation is a Transformation node. Each Transformation is decomposed into its Transformation nodes and related 'tree structure'.

7.2.5.1 Phase 1 Transformations

Transformations are applied to the Input Layer in order to generate the Enriched Input Layer. These transformations are used to prepare and enrich the data in a structure that can be easily used to generate specific reporting requirements. The types of transformations included in this phase aim to compile all the relevant variables and members required to generate the output layer. Therefore, transformations usually consist of derivations of specific variables or amounts, using other variables as input to classify, as well as join operations, combining cubes to retrieve all relevant variables.

Derivations

The purpose of derivations is to extract logic that may be reused in multiple transformations. Example below shows the derivation of the Carrying amount that may be applied to loans but also to securities:

```
define operator D_CRRYNG_AMNT(ACCNTNG_CLSSFCTN integer, FV integer, GRSS_CRRYNG_AMNT_E_INTRST integer,
ACCRD_INTRST integer, FV_CHNG_HDG_ACCNTNG integer, ACCMLTD_IMPRMNT integer, CRRYNG_AMNT_INPT integer,
IS_CRRYNG_AMNT_DRVD boolean)

  returns integer is

    if (IS_CRRYNG_AMNT_DRVD = "T") then

      if (ACCNTNG_CLSSFCTN in FV_ACCNTNG_CLSSFCTNS) then FV

      else (GRSS_CRRYNG_AMNT_E_INTRST + ACCRD_INTRST - ACCMLTD_IMPRMNT + FV_CHNG_HDG_ACCNTNG)

    else CRRYNG_AMNT_INPT

end operator
```

LNS_E										
[D] INSTRMNT_UNQ_ID	...	IS_CRRYING_AMNT_DRVD	ACCNTNG_CLSSFCTN	FV	GRSS_CRRYNG_AMNT_E_INTRST	ACCRD_INTRST	ACCMLTD_IMPRMNT	FV_CHNG_HDG_ACCNTNG	CRRYNG_AMNT_INPT	CRRYNG_AMNT
#1	...	T	IFRS: Financial assets held for trading (2)	11	NULL	NULL	NULL	NULL	NULL	11
#2	...	T	IFRS: Financial assets at amortised cost (6)	NULL	17	1	3	0	NULL	15
#3	...	T	IFRS: Financial assets designated at fair value through profit or loss (4)	13	NULL	NULL	NULL	NULL	NULL	13
...	...	...	...	...	...	...	...	...	...	...

7.2.6 Phase 2 (EIL → ROL)

Transformations generate the Reference Output Layer based on the Enriched Input Layer. This phase may also involve technical adjustments necessary in order to comply with the Non-reference Output Layer. The types of transformations seen here include filtering the variables contained in the cube, to keep only those variables required to produce the required output, as well as join/union operations.

Filters

The filter operator extracts a subset of records from a data set based on a Boolean condition. The example below displays a filter based on the variable, Accounting Classification.

/*Extraction of a subset where the Accounting classification (ACCNTNG_CLSSFCTN) is equal to IFRS: Financial assets held for trading (2)*/

LNS_E										
INSTRMNT_UNQ_ID	...	IS_CRRYING_AMNT_DRVD	ACCNTNG_CLSSFCTN	FV	GRSS_CRRYNG_AMNT_E_INTRST	ACCRD_INTRST	ACCMLTD_IMPRMNT	FV_CHNG_HDG_ACCNTNG	CRRYNG_AMNT_INPT	CRRYNG_AMNT
#1	...	T	IFRS: Financial assets held for trading (2)	11	NULL	NULL	NULL	NULL	NULL	11
#2	...	T	IFRS: Financial assets at amortised cost (6)	NULL	17	1	3	0	NULL	15
#3	...	T	IFRS: Financial assets designated at fair value through profit or loss (4)	13	NULL	NULL	NULL	NULL	NULL	13
...	...	...	...	...	...	...	...	...	...	...

Joins

The join operator allows us to combine two data sets (i.e. adding columns to a data set). The example below shows the joining of the cubes loans and debtors (LNS_DBTRS) and counterparties (CNTRPRTS).

/* Joining the data set LNS_DBTRS with CNTRPRTS*/

LNS_DBTRS_E := inner_join (LNS_DBTRS, CNTRPRTS on CNTRPRTY_ID) ;

LNS_DBTRS			
INSTRMNT_UNQ_ID	CNTRPRTY_ID	...	CRRYNG_AMNT
#1	ABC_12	...	11
#2	BBA_134	...	15
#3	NVA_456	...	13
...		...	...

CNTRPTS		
CNTRPRTY_ID	CNTRY	...
NVA_456	ES	...
ABC_12	AT	...
BBA_134	IT	...
...	...	...

LNS_DBTRS_E := inner_join(LNS_DBTRS, CNTRPRTS on CNTRPRTY_ID)					
INSTRMNT_UNQ_ID	CNTRPRTY_ID	...	CRRYNG_AMNT	CNTRY	...
#1	ABC_12	...	11	AT	...
#2	BBA_134	...	15	IT	...
#3	NVA_456	...	13	ES	...
...		...	...	...	...

7.2.7 Phase 3 (ROL → NROL)

The third phase involves applying mappings that describe the relationship between the Reference Output Layer and the Non-reference Output Layer.

Mappings provide a way to establish that two concepts created by different maintenance agencies are equivalent. The concepts we are interested in are the variables and the members, which are the building blocks for datasets.

In an ideal world mappings would be very simple: One table with two columns could suffice to express mappings.

Table 7 - Ideal scenario mappings

SOURCE MEMBER	DESTINATION MEMBER
a	1
b	2
c	3

SOURCE VARIABLE	DESTINATION VARIABLE
x	7
y	8
x	9

But, unfortunately, the reality is much more complex, and this implies the need for more complex mappings. There are two main sources of complexity: (i) Use of different classification systems, and (ii) errors.

As an illustrative example, let's take the European System of Accounts (ESA) classification of financial instruments. This classification is done with a specific purpose, and mixes different concepts within the same classification. For instance, in the ESA classification of instruments there are two values for *log-term debt securities* and *short-term debt securities*. So the data frameworks that follow ESA classifications tend to have one variable where two possible values are *long-term debt securities* and *short-term debt securities*. But in other frameworks, like FinRep, this classification is not followed, and therefore there are two separate variables: The *type of instrument* and the *original maturity*.

Table 8 - Need for complex Mappings

ESA_ INSTR_CLASS		TYP_INSTRMNT	ORGNL_MTRTY
F32 - Long-term debt security	➔	1 - Debt security	1 - Long-term
F31 - Short-term debt security	➔	1 - Debt security	2 - Short-term

The SMCube methodology provides a model able to address complex (n to m) mappings. In the SMCube, one full mapping points to one mapping of variables and, eventually, one mapping of members.

One full mapping points only to a variable when the variable is not enumerated. For example, if we want to map the variable *Carrying amount*, with code mi53 in the DPM, to the same concept with code CRRYNG_AMNT in the reference dictionary.

If the mapping is for enumerated variables, then it needs also to point to the member mappings. The MAPPING_DEFINITION table contains the full mappings. It includes one field with the mapping type. The most relevant types of mappings have the value 'E' and 'A'. 'E' mappings imply that a member mapping is required, while 'A' mappings imply that an algorithm is required. The algorithm in the latter case serves to add operations, if needed to the values in the non-enumerated variables. In most cases it will not have any value, meaning that no operation has to be done.

VARIABLE_MAPPING and VARIABLE_MAPPING_ITEM tables provide the variable mappings, while MEMBER_MAPPING and MEMBER_MAPPING_ITEM provide the member mappings. In the MEMBER_MAPPING_ITEM table, related pairs of mappings are found with corresponding MEMBER_MAPPING_ROW values

The two previous examples would be described (for illustrative purposes, the tables are simplified and only the enumeration tables are shown):

Table 9 - Mapping Definition

MAPPING_ID	MAPPING_TYPE	ALGORITHM	VARIABLE_MAPPING_ID	MEMBER_MAPPING_ID
ie1	E		vm1	mm1
ie2	A		vm2	

Table 10 - Variable_Mapping_Item

VARIABLE_MAPPING_ID	VARIABLE_ID	IS_SOURCE
vm1	ESA_ INSTR_CLASS	TRUE
vm1	TYP_INSTRMNT	FALSE
vm1	ORGNL_MTRTY	FALSE
vm2	mi53	TRUE
vm2	CRRYNG_AMNT	FALSE

Table 11 - Member_Mapping_Item

MEMBER_MAPPING_ID	MEMBER_MAPPING_ROW	VARIABLE_ID	IS_SOURCE	MEMBER_ID
mm1	1		TRUE	F32
mm1	1	TYP_INSTRMNT	FALSE	1
mm1	1	ORGNL_MTRTY	FALSE	1
mm1	2		TRUE	F31
mm1	2	TYP_INSTRMNT	FALSE	1
mm1	2	ORGNL_MTRTY	FALSE	2

8 BIRD Expert Group work – Introduction of new frameworks

The BIRD has two main processes involved in the development of the dictionary. These include:

1. Addition of reporting frameworks and their related templates into the dictionary;
2. Ongoing maintenance of frameworks and elements added to the dictionary, to adhere to any and all updates to reporting requirements.

The BIRD aims at covering all statistical and supervisory reporting frameworks that banks need to generate in order to satisfy the requirements of the ECB and the European Banking Authority (EBA). Specifically, it should comprise BSI, MIR, SHS, AnaCredit, the needs of other statistics such as balance of payments and national accounts, additional SSM's statistical requirements and EBA's ITS.

The process of adding new frameworks and reporting templates into the SDD and ultimately the BIRD, starts with the designation and scheduling by the BIRD Steering Group. The BIRD steering group is comprised of representatives from the associated European System of Central Banks (ESCB), and participating commercial banks who aim to outline the objectives for the BIRD project, the timeframe as to which the objectives should be aimed to be met, and the delegation of resources towards these objectives.

Currently, the way in which this is organised is through a work plan, devised by and agreed upon by the steering group. This work plan is split up into sub-groups to each focus on specific objectives. The objectives set out in the work plan for 2019-2020 include:

- the work stream on data modelling;
- the work stream on testing;
- work streams for each new reporting framework to be covered.

The work streams for the new reporting frameworks to be covered focus on the process for adding these frameworks into the dictionary begins. With the plan and announcement being made that a work stream for a particular framework will commence in the future, calls for nominations from professionals with working knowledge of the reporting requirements and systems are called upon to help define the required metadata to be included.

Members of the BIRD Expert Group are allocated to sub-groups and work streams, in accordance with the work plan established by the steering group. Each sub-group and work stream has a clear mandate and helps to expand the content of the BIRD documentation.

The work of the sub-groups culminates in a new release of the BIRD which incorporates a new reporting framework into the database. The activities of the work streams are reflected in quality assurance reports, guidelines, recommendations and consultations with the sub-groups on technical aspects.

Work is carried out in subgroups of roughly 6 to 12 participants (from ECB, NCBs and commercial banks)

Collaboration between NCBs, Authorities and commercial banks is key, as various perspectives on the same topic are considered:

- NCBs and NCAs: bring policy use and meaning behind regulations
- Commercial banks: expertise in data processes and business information systems

This collaboration fosters sharing of knowledge, and more accurate implementation of reporting requirements.

Sources

To implement regulations, the Expert Group relies on public information Public sources consist of:

- EU Regulations[15], Directives, Decisions;
- International or (as applicable) national accounting standards; and
- Published Q&A and Manuals by the relevant authorities[16] (EBA, ECB, SRB, etc.)

Results

The analysis of reporting requirements leads to produce BIRD documentation that is composed by:

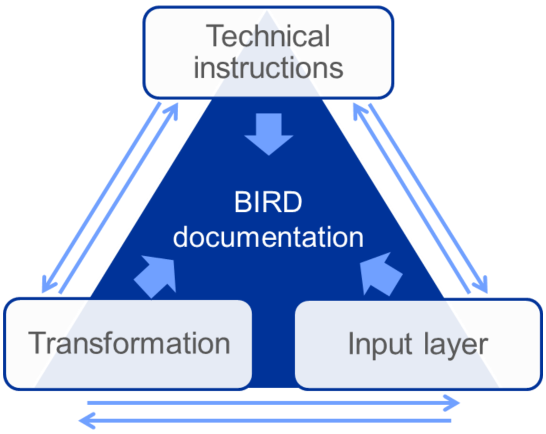


Figure 39 - BIRD documentation

- Technical instructions and illustrative examples,
- Input layer metadata
- Transformation rules in business and formal (VTL) language

[15] Link
[16] Link

Project: Adding FinRep Template 5.01 into the BIRD

One of the major projects undertaken during the traineeship was supporting a BIRD expert group in defining the requirements of the Financial Reporting (FinRep) template 5.01.

FINREP in the EU supervisory reporting framework is based on the International Financial Reporting Standards[17] (IFRS) as endorsed in the EU. FINREP's main objective is to provide supervisors with a comprehensive set of financial information as well as the information necessary to have a comprehensive view of the risk profile of the systemic risks posed by institutions to the financial sector and the real economy. FINREP follows as closely as possible the IFRS but has a clear focus on data needed for supervision of credit institutions and investment firms to provide supervisors with a key set of fully harmonised information on an institution's financial position.

The template 5.01 focuses on loans and advances other than held for trading assets, broken down by product.

8.1 Output Analysis

To begin integrating the report, an initial inspection of the template is done and the expert group discusses several topics to determine what output is required to be generated. In the case of the Finrep reporting template 5.01, displayed below, the main topics to be discussed were broken into the different element types:

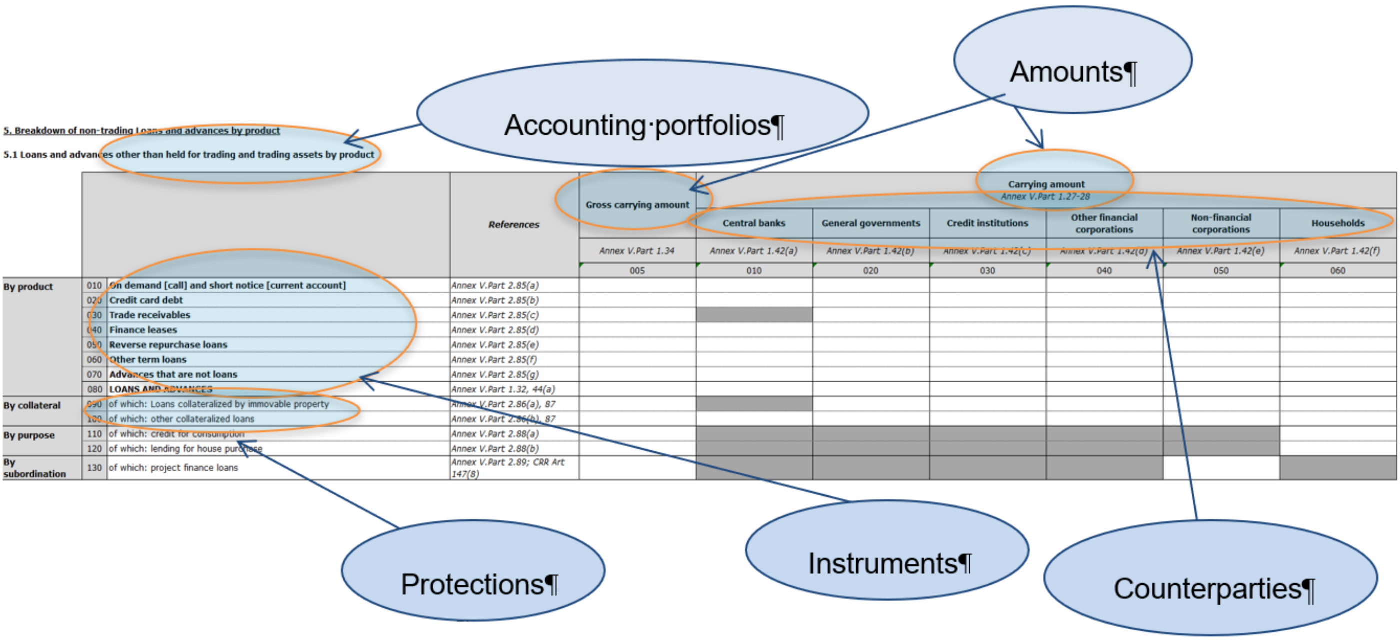


Figure 40 - Financial Reporting Template F 5.01

[17] Link

A general overview of the template showed how the information can be broken down into the different variables and members:

8.1.1 Accounting portfolios

Not all accounting portfolios are requested in template 5. We exclude instruments classified as:

- held for trading (for IFRS),
- trading (for nGAAP), and
- "non-current assets and disposal group classified as held for sale" (IFRS 5)

8.1.2 Amounts

Carrying amount (CA)
The evaluation of Carrying amount depends on the accounting classification. In order to evaluate the carrying amount, we can distinguish between:

a) Fair valued assets

b) Assets at amortised cost

c) Assets at cost (or LOCOM – Lower Of Cost Or Market)

Gross carrying amount (GCA)
Value also depends on accounting classification. Two broad categories:

a) Accounting portfolios subject to impairment (i.e. amortised cost):accumulated impairment added to CA

b) Accounting portfolios not subject to impairment

i.e. fair value through profit and loss: accumulated negative changes in fair value due to credit risk for non performing exposures added to Carrying Amount

8.1.3 Counterparties

Counterparty classification is almost compatible with ESA 2010 classification already included in the BIRD. Main differences are:

a) multilateral development banks reclassified as "*Credit Institutions*"

b) international organisations classified as "*General Government*"

c) establish the *immediate counterparty* and relevant entities for *joint counterparties*

These adjustments will have to be made in order to correctly identify the reclassification of the above members.

8.1.4 Instruments

According to the regulation, loans mean the same as ESA 2010 plus advances that are not loans

N.B.: add assets classified in *cash and cash balances at central banks and other demand deposits* to the instrument category *On demand [call] and short notice*

8.1.5 Protections

Two categories of protections requested in the template:

a) Immovable property
b) Protections other than immovable property

According to the regulation, all loans secured by both a) and b) Regardless ratio loan/collateral (LTV), shall be reported in secured by immovable property only

8.1.6 Input elements required

Through the analysis, all DPM elements required in the production were summarised in the below table:

Table 12 - DPM input variables

Role	*Variable*	*Subdomain / Member / Variable set*
D	Main category (MCY)	Loans and advances (x469)
D	Instrument (MCB)	Total/Not applicable (x0), Project finance loans (x360), Advances that are not loans (x823), On demand [call] and short notice [current account] (x824), Term loans. Credit card debt (x825), Term loans. Finance leases (x826), Term loans. Other than Trade receivables, Credit card debt, Finance leases, Reverse repurchase loans (x827), Term loans. Reverse repurchase loans (x828),

		Term loans. Trade receivables (x829)
D	Accounting portfolio (APL)	Financial assets other than Held for trading and Trading Financial Assets (x77)
D	Counterparty sector (CPS)	Not applicable/ All counterparties (x0), General governments (x1), Central banks (x10), Credit institutions (x12), Financial corporations other than credit institutions (x18), Non-financial corporations (x20), Households (x5)
D	Main category of collateral or guarantee given (MCC)	Total/Not applicable (x0), Other than Real estate (x262), Real estate (x292)
D	Purpose (PUR)	Not applicable/All purposes (x0), Credit for consumption (x1), Lending for house purchase (x3)
D	Artificial variable for metrics (ATY)	Gross carrying amount (mi136), Carrying amount (mi53)
O	Observation_Value (Observation_Value)	Non enumerated subdomain for domain String (355)

8.2 Gap Analysis

Once the group assessed the output requirements, we analysed the available input and identified the gaps between the two. Prior to adding FinRep, the BIRD already contained the necessary information for SHS and AnaCredit derivation.

The following diagram describes the decision process of determining whether new elements are required to be added to the dictionary:

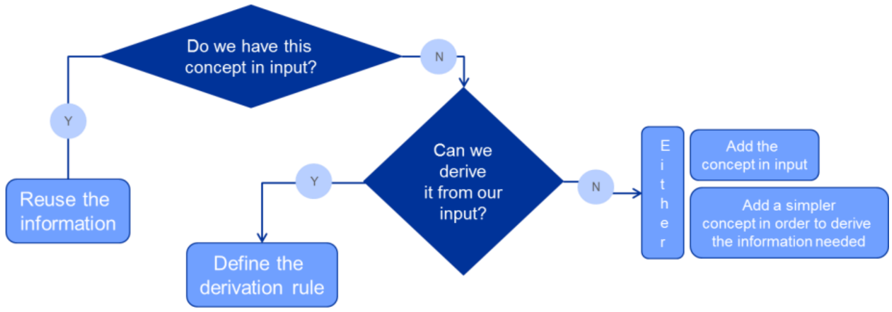


Figure 41 - Gap analysis

8.2.1 Accounting portfolios

All relevant accounting portfolios and information were already present in AnaCredit and SHS. However, was missing the concept of held for sale[18]

a) Introduced an input Boolean variable

b) Hierarchy to establish relationships within the accounting portfolios in input

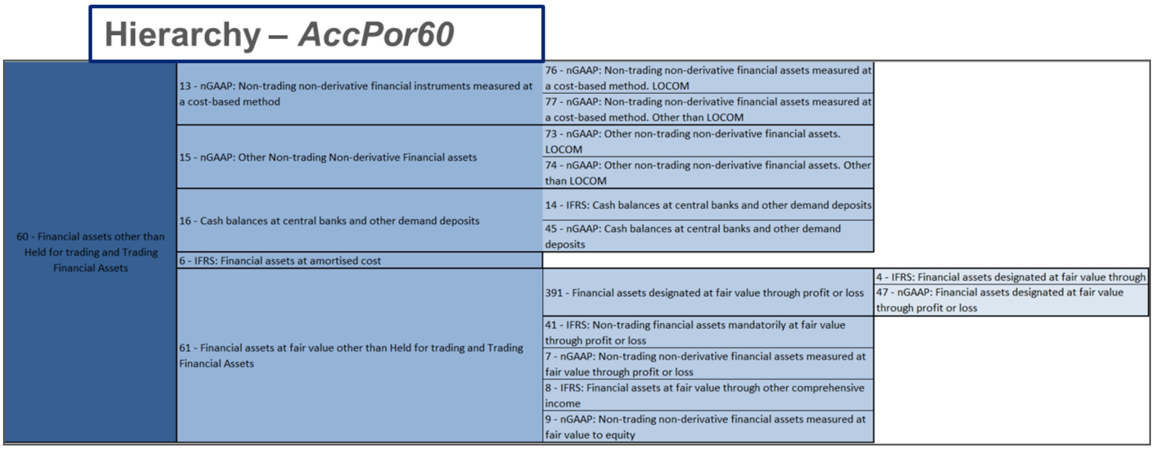


Figure 42 - Member Hierarchy AccPor60

8.2.2 Amounts

<u>Carrying amount</u>

The carrying amount to be reported differs depending on the type of accounting portfolio the transaction falls under, either the IFRS or NGAAP. In regard to the carrying amount, BIRD treats IFRS and nGAAP differently:

a) For IFRS – reporters can either calculate the carrying amount by themselves or use a built-in derivation rule, originally created for AnaCredit purposes

[18] A boolean that identifies the Non-current assets and instruments in disposal groups classified as held for sale in accordance with IFRS5. Is not applicable under national GAAP based on BAD.

b) For nGAAP – reporters have to feed the carrying amount for each asset in input (due to heterogeneity of national accounting standards)

Gross carrying amount

Gross carrying amount is derived from other existing input information. The derivation rule, which will be explained in the later section outlining the transformation process, takes into account input information accounting portfolio, carrying amount, accumulated impairment (also for nGAAP), accumulated changes in fair value due to credit risk, credit quality status (performing/non-perf.), and impairment status (for nGAAP portfolio subject to impairment, or not)

8.2.3 Counterparties

Information already exists in the counterparties cube. Transactions-counterparties cube links instruments and counterparties. Counterparty institutional sector already described in input layer where reconciliation with FinRep is done via transformation rules and hierarchies.

Immediate counterparty is identified by role in transaction-counterparties cube, whereas, for joint counterparties we will require to:

a) Describe both counterparties in a separated cube identifying the main one

b) Create additional counterparty with the features of the main one

8.2.4 Instruments

In regards to the types of instruments to be included, all loans according to BSI regulation are already covered. The one type of instrument not already included was 'Advances that are not loans', which was added in input as a specific cube. The list of related instrument cubes therefore used in the production of the template information were:

Table 13 - Instrument cubes F 5.01

Product category template 5	Instrument cubes in BIRD input
On demand [call] and short notice [current account]	CRRNT_ACCNT (IS_ASST=TRUE) or OTHR_DPSTS (RPYMNT_RGHT=1)
Credit card debt	CRDT_CRD_DBT
Trade receivables	FCTRNG or OTHR_TRD_RCVBLS
Finance leases	FNNCL_LSS
Reverse repurchase loans	RVRS_RPRCHS_LNS

Other term loans	OTHR_LNS or OTHR_DPSTS (RPYMNT_RGHT=2)
Advances that are not loans	ADVNCS_NT_LNS

8.2.5 Protections

In BIRD the connection between *protections received* and *instruments* was already modeled linked to AnaCredit implementation, as displayed in the figure below.

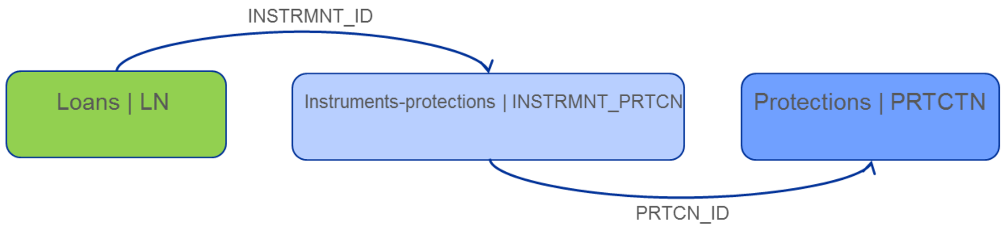


Figure 43 - Instruments-protections connection

In was concurred that a transformation was to be designed that aims to differentiate between protections collateralised by ‘immovable property’ and ‘other than immovable property’ protections, which would give priority to immovable property protections.

8.3 Defining Transformation Rules

After the input elements have been decided on, we developed the set of transformation rules required to generate the reference output layer from the elements received in the input layer.

All formal descriptions of the transformations and the VTL expressions can be found in the Annex, under section "VTL TRANSFORMATIONS FOR FINREP TEMPLATE 5.01". The structure of these transformations are based on the VTL reference manual.[19]

8.3.1 Phase 1 Input Layer → Enriched Input Layer

8.3.1.1 Derivation Rules

The derivation rule functions which are included in the transformations are used to derive and calculate different amounts and elements used in the production of reports, using existing information included in the input layer. In our case, the derivations that are required include:

- Derivation of Carrying Amount
- Derivation of Gross Carrying amount
- Derivation of Type of instrument
- Derivation of Performing status

These functions are stored separately in the transformation scheme and are called as part of other operations relating to the generation of specific outputs.

<u>Derivation of Carrying Amount</u>

For the derivation of carrying amount, an extra transformation was applied after the original derivation, in order to calculate the carrying amount specifically for loans and advances. The calculation reflects the initial calculation, but with the added parameter 'Is carrying amount derived' (IS_CRRYNG_AMNT_DRVD). This is a boolean variable included that specifies if a bank intends to have the carrying amount derived, or if they included the amount in input themselves.

<u>Institutional Sector/Country list for FinRep</u>

This transformation involved creating a function to enhance the list of members of institutional sector classifications that are related to FinRep, adding in classifications for multilateral development banks and international organisations. The function itself runs a check to see if a value is present for the variable International Organisation code (INTRNTNL_ORGNSTN_CD) in the cubes multilateral development banks FinRep and international organisation FinRep. If a value is found present in these fields, the classification is assigned accordingly.

[19] Link

8.3.1.2 Preparation of Enriched Input Layer Cubes

The first step undertaken in the transformation rules is first to fetch all of the data from the related cubes, in order to generate the enriched cubes for use in the production of the reference output. The transformation used gathers the cube associated with the code that is valid at a specific point in time. The rule run for cubes follows the following example, taken for the generation of the Credit Card Debt cube: CRDT_CRD_DBT := GT_CB_VLD_AT("CRDT_CRD_DBT", DT_RFRNC).

Once of the cubes have been fetched from the dictionary, the relevant aggregated enriched input layer cubes can be formed. In our case, the first two enriched input layer cubes to be generated are for the instruments cubes to be added, which are aggregated into Loans enriched (LNS_E) and Advances that are not loans (ADVNCS_NT_LNS_E).

<u>Loans enriched</u>

In creating the loans enriched cube, the transformation that must take place is a union type function of all the relevant input layer cubes. All the cubes mentioned in the Gap Analysis (Table 5) are joined to form the loans cube.

<u>Advances that are not loans enriched</u>

For the advances that are not loans cube, all relevant information for other advances are stored in this cube, therefore no union function is required to be performed. Instead, after the data is fetched from the cube, a function is written to instruct the database to form an enriched input layer cube, comprised on the entire input layer cube

<u>Loans and Advances enriched</u>

Once both enriched cubes have been formed, these two cubes and joined via another union function, to create the cube Loans and Advances enriched (LNS_ADVNCS_E). Once joined, the next operations performed were to firstly, remove the variable Carrying amount from the list of included variables, and replace with the derived carrying amount, which as stated in the Gap Analysis can be either input generated or calculated through the derivation rule, for IFRS. In this case, we performed an inner join between the Loans and Advances Cube and the derivation function, using the INSTRMNT_UNQ_ID to separate the amount for every unique ID given as input.

<u>Counterparties Enriched</u>

The preparation of the enriched counterparties cube involves enriching the cube with the derivations previously defined for variables Country and Institutional Sector, to satisfy the reclassification of multilateral development banks as credit institutions.

8.3.2 Phase 2 Enriched Input Layer→ Reference Output Layer

8.3.2.1 Counterparties

Preparing the counterparties cube information to be included in the reporting information.

<u>Counterparties FinRep</u>

The next step is creating the counterparties FinRep cube (CNTRPRTS_FINREP) through joining the Counterparties enriched cube with the updated institutional sector list created with the function D_INSTTTNL_SCTR_FINREP_1_0, in order to enrich the counterparties cube with a complete list of counterparties.

<u>Loans Advances Debtors enriched</u>

The next step involved enriching the loans and advances cube with information pertaining to the debtors, or counterparties, in order to derive the performing status of the counterparty. The processes that take place at this step involve firstly joining the loans and advances cube with the instruments customers cube (INSTRMNTS_CSTMRS_E) using the instrument unique ID for each unique instrument, then also joining to the cube counterparties (CNTRPRTS_FINREP) using the counterparty identifier as the primary key (CNTRPRTY_ID). The last step is deriving the performing status, calling the function previously created. The variables noted in this operation define the parameters of the function.

<u>Protections received FinRep</u>

This step involved preparing the protections received cube, before it is joined into the enriched input layer. The functions performed were to first retrieve all the relevant variables form the cubes Protections real estate enriched (PRTCTNS_RL_ESTT_FINREP) and protections not real estate, which included the protection ID and type of protection variables from each.

We next needed to derive the information regarding to variables Protection identifier, Is residential collateral and financial guarantee, Is commercial collateral and financial guarantee. In order to do so, we joined the real estate and not real estate cubes, and created a sub-set of both cubes to then evaluate on all protections, whether the collateral used was residential collateral or commercial.

Finally, the sub-set cube and the protections received cube are joined, in order to keep all relevant variables in a single cube, to be added to the enriched input layer.

<u>Loans/advances-debtors-protections enriched</u>

In this step, the loans advances debtors cube is enriched with the information from the instruments and protections cube (INSTRMNTS_PRTCTNS_E), as well as the cube protections received (PRTCTNS_RCVD_FINREP).

The next step taken is to split the monetary values by the number of protections. This is to remove the possibility of creating a duplication of records with regards to protections, as one instrument may be related to many protections. The values that are divided include:

- Accumulated impairment, accumulated changes in fair value due to credit risk – negative
- Carrying Amount
- Gross Carrying Amount
- Outstanding Nominal amount
- Accumulated change in fair value due to credit risk – negative

- Accumulated Impairment

In order to conduct these operations the number of protections must be included as a variable. In order to achieve this, the protections instruments cube (PRTCTNS_RP_INSTRMNTS) is joined using the instrument unique ID as the primary key.

8.3.3 Generation of the reference output layer

8.3.3.1 Generation of unfolded cube

At this stage, all of the required variables and cubes have been assembled, and we were able to begin to keep only the relevant variables used to generate our template. Firstly, in order to apply the template specific transformations, we create a new version of the LNS_ADVNCS_DBTRS_PRTCNS cube called F_05_01_REF_UNFLDD. As the LNS_ADVNCS_DBTRS_PRTCNS cube may be used in the production of other templates, creating a new version allows us to make changes in a sub-space, without effecting other procedures.

The first function included was the aggregation function. This function was used to sum the carrying amount by variables such as type of accounting item or institutional sector. This allows for analysis to be performed not only on a transactional basis but also on a more aggregated level, as requested by analysts.

The next function performed was to assign the roles to each variable within the cube. The variables that are to be used as dimensions are assigned the term ‘identifier’ and the observation values to be reported are assigned as ‘measures’.

Following the assigning of roles, transformations were used to apply restrictions to only allow variables with relevant members to be reported in the template. This filter transformation allows us to ensure that the values reported in the template reflect the requirements on that template. The restrictions placed on these variables are detailed in Table 14 below.

8.3.3.2 Folding the cube

Folding the cube is the final transformation scheme in the process of creating the reference output cube. The first step in this process was to apply the ‘unpivot’ transformation. The unpivot operator basically folds the Observations of a data set into the combination of a Metric (MTRCS) and Observation value (OBSERVATION_VALUE)[20]

The last transformation to be applied is the application of member hierarchies and aggregation. An example of this process is documented below. For the sake of simplicity we only consider the combinations present in the figure 44.

[20] Link, line 7158

			Columns									
			Gross carrying	Carrying amount								
				Central banks	General governments	Credit institutions	Other financial corporations	Non-financial corporations	Households	(MCY:MC) Main category	(APL:PL) Accounting portfolio	(MCB:MC) Instrument
			005	010	020	030	040	050	060			
Rows	009 By product											
	Reverse repurchase loans	050	152590 I€$	152418 I€$	152435 I€$	152425 I€$	152468 I€$	152458 I€$	152445 I€$	(MC:x469) Loans and advances	(PL:x77) Financial assets other than Held for trading and Trading Financial Assets	(MC:x828) Term loans. Reverse repurchase loans
	LOANS AND ADVANCES	080	152598 I€$	152419 I€$	152439 I€$	152429 I€$	152472 I€$	152462 I€$	152449 I€$	(MC:x469) Loans and advances	(PL:x77) Financial assets other than Held for trading and Trading Financial Assets	
	Metric		(mi136) Gross carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]			
	(BAS:BA) Base		(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets			
	(CPS:CT) Counterparty sector			(CT:x10) Central banks	(CT:x1) General governments	(CT:x12) Credit institutions	(CT:x18) Financial corporations other than credit institutions	(CT:x20) Non-financial corporations	(CT:x5) Households			

Figure 44- F.01 Reverse repurchase loans

The above table shows the non-reference cube EBA_FINREP_EBA_F_05.01_FINREP_2.8 while we are describing the reference cube FINREP_REF_F_05.01_REF_16.

The (unfolded) reference representation of the cube EBA_FINREP_EBA_F_05.01_FINREP_2.8: F_05_01 is generated by the transformation scheme (TRANSFORMATION_SCHEME_ID=G_F_05_01_REF_UNFLDD_FINREP_1:G_F_05_01_REF_UNFLDD_FINREP). It takes into account Loans and advances per debtor, applies member hierarchies on the variables Type of instrument (TYP_INSTRMNT) and Institutional sector (INSTTTNL_SCTR) in order to comply with the level of granularity with regards to those variables and applies filters regarding the variables Institutional sector (INSTTTNL_SCTR), Is held for sale (IS_HFS), Type of accounting item (TYP_ACCNTNG_ITM) and Type of instrument (TYP_INSTRMNT). Finally it produces a cube having the following structure:

Table 14 - Restricted Variables

Role	Name	Members/Subdomains
Dimension	Accounting Classification	Financial assets other than Held for trading and Trading Financial Assets (60)
Dimension	Reference Date	Generic date value
Dimension	Institutional Sector	Central Banks (S121), General Governments (S13), Credit Institutions (S122), Other financial corporations (S12), Non-financial corporations (S11), Households (S14)
Dimension	Is held for Sale	False (F)
Dimension	Observed agent internal identifier	String value
Dimension	Project finance loan	Boolean variable to specify if amount reported needs to be reported in row 130

Dimension	Purpose	Boolean Variable to specify whether amount is to be reported in row 110 or 120
Dimension	Repayment rights	Either on demand or short notice (1) or other than on demand or short notice (2)
Dimension	Type of Accounting item	Creditor (40)
Dimension	Type of collateral or guarantee given	No Restriction
Dimension	Type of Instrument	On demand [call] and short notice [current account] (031)', 'Reverse repurchase agreements (1003)', 'Term loans Other than Trade receivables, Credit card balance, Finance leases, Reverse repurchase agreements (114)', 'Advances that are not loans (141)', 'Loans and advances (149)', 'Credit card debt (51)', 'Trade receivables (71)', 'Finance leases (80)
Observation value	Carrying Amount	Monetary amount
Observation Value	Gross carrying amount	Monetary Amount

To reduce the number of redundant variables we will remove the all variables that are not related to members: Observed agent internal identifier (OBSRVD_AGNT_INTRNL_ID), reference date (DT_RFRNC)

Records of the data, described as a cube, may look like the following table:

Table 15 - Example Dataset

Accounting Classification	Institutional Sector	Type of Instrument	Carrying amount
IFRS: Financial assets at amortised cost (6)	Credit Institutions (S122)	Reverse repurchase agreements (114)	6
IFRS: Financial assets at fair value through other comprehensive income (8)	Credit Institutions (S122)	Reverse repurchase agreements (114)	11
IFRS: Non-trading	Credit Institutions	Reverse repurchase	9

financial assets mandatorily at fair value through profit or loss (41)	(S122)	agreements (114)	

We can see from the above example that the values of variable accounting classification are not in line with our required output value (ACCNTNG_CLSSFCTN_60) due to the fact that they reflect the values that are allowed in the input layer. Consequently we need to apply a member hierarchy on this variable such that we may derive the required output values.

Application of member hierarchy

We apply the member hierarchy for 'Financial assets other than Held for trading and Trading Financial Assets (ACCPOR60)': resulting in the following view:

Table 16 - With Member Hierarchy applied

Accounting Classification	Institutional Sector	Type of Instrument	Carrying amount
Financial assets other than Held for trading and Trading Financial Assets (60)	Credit Institutions (S122)	Reverse repurchase agreements (114)	6
Financial assets other than Held for trading and Trading Financial Assets (60)	Credit Institutions (S122)	Reverse repurchase agreements (114)	11
Financial assets other than Held for trading and Trading Financial Assets (60)	General Government (S13)	Reverse repurchase agreements (114)	9

This allows us, via aggregation, to generate the combinations EBA_152435_REF and EBA_152425_REF, resulting in the following:

	Columns					
	Gross carrying	Carrying amount				
		Central banks	General governments	Credit institutions	Other financial corporations	Non-financial corporations
	005	010	020	030	040	050
050	152530 I€$	152418 I€$	9	17	152468 I€$	152458 I€$
080	152538 I€$	152419 I€$	152439 I€$	152429 I€$	152472 I€$	152462 I€$
	(mi136) Gross carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]	(mi53) Carrying amount [mi]
	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets	(BA:x6) Assets
		(CT:x10) Central banks	(CT:x1) General governments	(CT:x12) Credit institutions	(CT:x18) Financial corporations other than credit institutions	(CT:x20) Non-financial corporations

<u>VTL</u>

The associated VTL statements in order to properly generate the combinations were the following:

```
/*Apply filter criteria in order to identify the records contributing to the combination
EBA_152435_REF*/EBA_152435_REF := F_06_01_REF [filter(ACCNTNG_CLSSFCTN = "60" and
INSTTTNL_SCTR = "S13" and IS_HFS = "F" and TYP_ACCNTNG_ITM = "40" and TYP_INSTRMNT =
"114")]; /*Keep only relevant observation*/ EBA_152499_REF := EBA_152499_REF
[keep(CRRYNG_AMNT)];
```

```
/*Apply filter criteria in order to identify the records contributing to the combination
EBA_152435_REF*/EBA_152435_REF := F_06_01_REF [filter(ACCNTNG_CLSSFCTN = "60" and
INSTTTNL_SCTR = "S122" and IS_HFS = "F" and TYP_ACCNTNG_ITM = "40" and TYP_INSTRMNT =
"114")]; /*Keep only relevant observation*/ EBA_152499_REF := EBA_152499_REF
[keep(CRRYNG_AMNT)];
```

8.4 Application of Mappings

The final step in integrating a new report into the BIRD is the final phase, which calls for the application of mappings. In the case of FinRep templates, DPM / XBRL content is translated into the dictionary without any changes to the content itself (i.e. non-reference description, using DPM codification), meaning one annotated template is translated into one cube. In our case, FinRep templates are mapped based on their data points, so each individual data point will be mapped to a combination in the SMCube structure.

Mappings for XRBL/DPM frameworks are handled by an external team, using a purpose developed application. Figure 45 describes the tasks that are necessary in order to apply the mappings, describing the association between a non-reference cube and a reference cube, onto the related non-reference data set in order to generate a reference data set comprising the same information (but using the reference codification system). In general we always map variables into variables and members into members, consequently the application of mappings onto a data set may be considered as an ETL process based on specific rules (i.e. the mappings).

The application developed as an ETL process in Java/Python (or similar), running as a batch job in the DISC environment with this functionality:

- The application reads the non-reference dataset
- It reads the mappings from SDD
- The mappings are applied to the non-reference-dataset
- The output is a reference dataset containing the mapped structure and members.

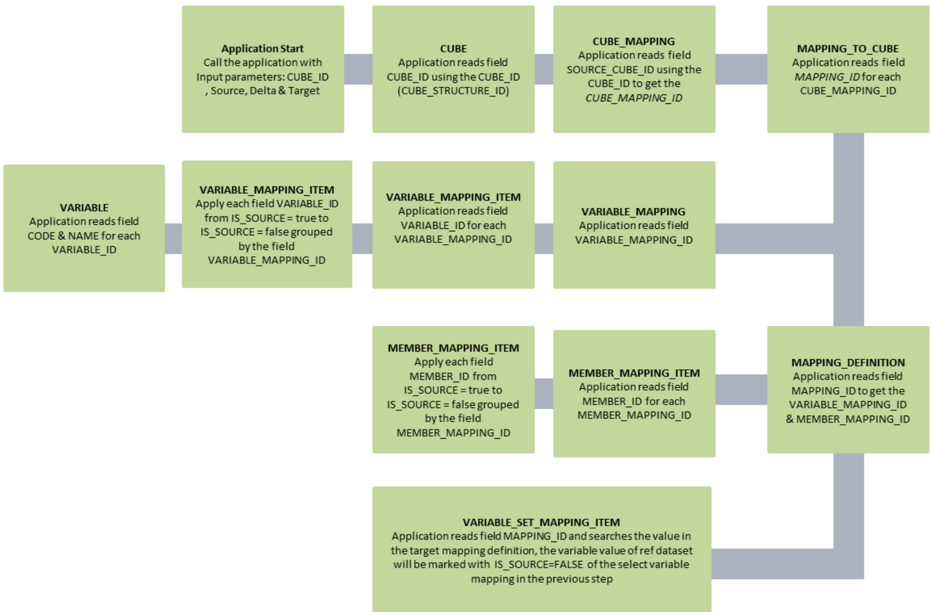


Figure 45 - Mapping Process

After the application has successfully completed developing the above process, the mappings are stored in the relevant tables in the DISC platform, under the SDD.

An example to show how the mappings between data points and combinations are stored in the database is as the following:

		Columns						
		Gross carrying amount	Carrying amount					
			Central banks	General governments	Credit institutions	Other financial corporations	Non-financial corporations	Households
		005	010	020	030	040	050	060
009 By product								
On demand [call] and short notice [current account]	010	152586 €£$	152414 €£$	152431 €£$	152421 €£$	152464 €£$	152454 €£$	152441 €£$
Credit card debt	020	152587 €£$	152415 €£$	152432 €£$	152422 €£$	152465 €£$	152455 €£$	152442 €£$
Trade receivables	030	152591 €£$	€£$	152436 €£$	152426 €£$	152469 €£$	152459 €£$	152446 €£$
Finance leases	040	152588 €£$	152416 €£$	152433 €£$	152423 €£$	152466 €£$	152456 €£$	152443 €£$
Reverse repurchase loans	050	152590 €£$	152418 €£$	152435 €£$	152425 €£$	152468 €£$	152458 €£$	152445 €£$
Other term loans	060	152589 €£$	152417 €£$	152434 €£$	152424 €£$	152467 €£$	152457 €£$	152444 €£$

The variables and members associated with this data point contain the following:

Role	*Variable*	*Subdomain / Member / Variable set*
D	Main category (MCY)	Loans and advances (x469)
D	Instrument (MCB)	Term loans. Reverse repurchase loans (x828)
D	Accounting portfolio (APL)	Financial assets other than Held for trading and Trading Financial Assets (x77)
D	Counterparty sector (CPS)	Credit institutions (x12)
D	Main category of collateral or guarantee given (MCC)	Total/Not applicable (x0)
D	Purpose (PUR)	Not applicable/All purposes (x0)
D	Artificial variable for metrics (ATY)	Carrying amount (mi53)
O	Observation_Value (Observation_Value)	Non enumerated subdomain for domain String (355)

From this data point, the following mappings can be found in the database:

Cube Mapping

MAINTENANCE_AGENCY_ID	CUBE_MAPPING_ID	NAME	CODE	SOURCE_CUBE_ID	DESTINATION_CUBE_ID	DESCRIPTION
ECB	M_F_05.01_REF	F_05.01-->F_05.01_REF	M_F_05.01_REF	EBA_FINREP_EBA_F_05.01_FINREP_2.8	FINREP_REF_F_05.01_REF_16	F_05.01-->F_05.01_REF

Mapping to cube

CUBE_MAPPING_ID	MAPPING_ID	VALID_FROM	VALID_TO
M_F_05.01_REF	DPM_Entity	01.01.1999	31.12.9999
M_F_05.01_REF	DPM_BAS_APL_MCY_MCB_MCC_MCG_SUB_MCP	01.01.1999	31.12.9999
M_F_05.01_REF	DPM_ObservationValue	01.01.1999	31.12.9999
M_F_05.01_REF	VS_F_05_01	01.01.1999	31.12.9999
M_F_05.01_REF	DPM_CPS_CPZ	01.01.1999	31.12.9999
M_F_05.01_REF	DPM_PUR	01.01.1999	31.12.9999
M_F_05.01_REF	DPM_Period	01.01.1999	31.12.9999

Mapping Definition. In this example we will display the mappings for the variable counterparty sector and counterparty size (Mapping_ID: DPM_CPS_CPZ)

MAINTENANCE_AGENCY_ID	MAPPING_ID	NAME	MAPPING_TYPE	CODE	ALGORITHM	MEMBER_MAPPING_ID	VARIABLE_MAPPING_ID
ECB	DPM_CPS_CPZ	Counterparty sector, size of the counterparty mapping	E	DPM_CPS_CPZ		DPM_CT1	DPM_CPS_CPZ

Variable Mapping Item:

VARIABLE_MAPPING_ID	VARIABLE_ID	IS_SOURCE	VALID_FROM	VALID_TO
DPM_CPS_CPZ	EBA_CPS	true	01.01.1900	31.12.9999
DPM_CPS_CPZ	EBA_CPZ	true	01.01.1900	31.12.9999
DPM_CPS_CPZ	ENTRPRS_SZ	false	01.01.1900	31.12.9999
DPM_CPS_CPZ	INSTTTNL_SCTR	false	01.01.1900	31.12.9999

Member mapping item:

MEMBER_MAPPING_ID	MEMBER_MAPPING_ROW	VARIABLE_ID	IS_SOURCE	MEMBER_ID	VALID_FROM	VALID_TO
DPM_CT1	3	ENTRPRS_SZ	false	SZ_0	01.01.1900	31.12.9999
DPM_CT1	3	INSTTTNL_SCTR	false	INSTTTNL_SCTR_S1_MDB	01.01.1900	31.12.9999
DPM_CT1	3	EBA_CPS	true	EBA_CT_EBA_x12	01.01.1900	31.12.9999
DPM_CT1	3	EBA_CPZ	true	EBA_CT_EBA_x0	01.01.1900	31.12.9999

8.5 Publishing a new BIRD release

With the introduction of every new reporting framework into the BIRD, a new version of the BIRD database is published, to be made open to be public

The first step in creating a new release is including all new elements and concepts into the database. This is done through the same process to how the changes are made in the SDD. The BIRD is represented as a framework within the SDD and therefore can be changed accordingly, following the same rules.

Once the processed has been completed of adding all new input elements, transformations and mappings, the entire content of the BIRD framework is exported from the SDD data environment, and converted into a Microsoft access database. From there scripts are run of the database, first to validate the content of the database, and then to compare the new version of the database with the current version. This script prepares an excel file displaying the changes made.

8.5.1 Technical guidelines

Every new release of the database is accompanied by a revision of a Technical Guidelines document. The document is also made available to the public and provides technical information as to the content of BIRD database, as well as instructional information to members of reporting banks as to best fill the cubes related to each framework.

Once all deliverables are prepared, the content of the database and the technical guidelines is made available online. All materials can be found on the ECB website[21]

[21] Link

9 Conclusion

European banking supervision has since reached its five year life span and the effect has already become well documented within the European banking community. Supervision efforts greatly helped to speed up the post-crisis repair of the banks' balance sheets, reducing the amount of non-performing or defaulted loans by half since its inception.

The enhanced supervision requirements obviously come at a cost. Banks constantly complain about the burden that the new regulatory and supervisory frameworks have brought. It is important in this regard to continue development of systems that allow for the transaction of large amounts of data in ways that help to ease this burden. The SDD continues to plays an integral part in ensuring that supervisory reports produced continue to provide users and authorities with higher quality data, which leads to a better understanding about the health of individual banks and the banking sector as a whole.

In regards to the BIRD it will maintain its status as a public good, and open for voluntary adoption by banks, but is starting to garner increasing levels of interest and enthusiasm from members of the banking community. Reporting agents aim to take advantage of the BIRD and eventual integrated reporting framework (IReF), due to the harmonisation of concepts and methodologies underlying statistical reports, as well as reducing the amount of overlapping in statistical data reported. The project will continue until the implementation of the IRef, expected around 2027.

10 Limitations and future works

10.1 Limitations

The major limitation placed on this report is the restriction of access to resources of the different reporting areas that make use of the SDD. Due to the nature of security in the ECB, information regarding other teams or divisions is able to be accessed only by those within the division. This limited the amount of information I was able to obtain about how the different applications and systems which are responsible for collection and production of supervisory reports interact with the SDD. Therefore, I was unable to provide a detailed practical look into the overall systems approach to production of supervisory reports.

10.2 Future Works

The upcoming work aimed to improve the overall BIRD model and process will revolve around two major projects, a work stream on data modelling, which will focus on improving the overall structure of the BIRD logical model, increasing harmonisation and usability, as well as the work stream on testing, which aims to provide the first round of scrutiny and validation of the current input layer, using test scenarios to identify if the correct output can already be produced.

10.2.1 Work-stream on data modelling

So far, and especially due to the origin of the BIRD initiative as a pilot, the various components of the IL were designed by different people based on different user needs. Therefore the structure of the IL lacks a certain level of harmonization and could be improved by generally accepted “design principles” and / or the application of patterns.

Therefore, a work group for data modelling was created, in collaboration with the involved banking institutions, similar to the sub-groups, in order to re-define the structure of the input layer

The main purpose of the Work stream on data modelling is to deliver

- A stable, harmonized, extensible BIRD input layer Stable in the sense that the implementation of new frameworks shouldn’t affect the core structure of the input layer
- Harmonized so that all aspects of the input layer are modelled in a similar way
- National Extensibility, i.e. the utility of the BIRD by an NCB / commercial banks in a specific country
- Ensuring consistency regarding modelling activities in the subgroups
- Modelling guidelines / principles / best practices
- And finally an adoption of the current input layer including transformation rules

A stable, harmonized, extensible BIRD input layer may be achieved via Normalization of the current input layer. Having non-normalised inputs may lead to the input layer data non properly reflecting the banks' internal systems. Other reasons supporting this direction are that normalisation of the input layer cubes would bring other significant advantages, such as:

- Easier interpretability in inputting data into the input cubes
- Provide referential integrity, without the need for increased validation rules.
- Dense Cubes

Upcoming Timeline

- Refactoring / review of Input Layer to define the Logical Model [in-progress]
- Version 1.0 to be published (Instruments, Credit facilities, Securities, Protections, Parties, Roles, Securitisation, Derivatives, Master netting agreements, Balance sheet aspects, Aggregates)
- Derivation of Input Layer based on the Logical Model and Consolidation of (Logical Model) Entities into Cubes
- Amendment of Transformation rules

10.2.2 Work-stream on testing

One of the other major tasks to be undergone is the testing of the input data added thus far, in order to determine its reliability and to identify any errors. The work stream will act a verification process, to ensure that the BIRD process produces reliable data. This will be based on three deliverables:

- Deliverable 1: "Provide / generate test data"
- Deliverable 2: "Based on the test data, verify (in a reproducible way) if the BIRD transformation rules generate correct results"
- Deliverable 3: "Provide suggestions for improvement regarding errors / issues spotted in Deliverable 2"

11 Bibliography

Abdelguerfi, M & Eskicioglu, R (1997). The Electrical Engineering Handbook (Knowledge Engineering, p 2183)

Bajpayee, R, Sonali P.S, & Kumar,V (2015). Big Data: A Brief investigation on NoSQL Databases. (2015)

Bank for International Settlements (BSI), CGFS - Structural changes in banking after the crisis, (p 9). Retrieved from https://www.bis.org/publ/cgfs60.pdf

Bhojaraju.G (2003). Database Management: Concepts and Design. Proceedings of 24th IASLIC–SIG-2003, December, Dehradun: Survey of India.Raza, Muhammad, (2018), What are Database Management Systems? DBMS Explained. Retrieved from https://www.bmc.com/blogs/dbms-database-management-systems/

Bourne, K. C. (2014).Your Organization. Application Administrators Handbook, 329–343.

Chen, P (1976): The Entity-Relationship Model - Toward a Unified View of Data. ACM Trans. Database Syst. 1(1): Retrieved from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.526.369&rep=rep1&type=pdf

Collins, J. (2013), Comparison of Relational and Multi-Dimensional Database Structures, http://www.alphadevx.com/a/36-Comparison-of-Relational-and-Multi-Dimensional-Database-Structures,

Coronel, C. and Morris, S. (2014) Database Systems: Design, Implementation, & Management. Published by Cenage Learning.

Date, C.J., Introduction to Database Systems (7th Edition) Addison Wesley, 2000

Dorf, R. C, (1997) The Electrical Engineering Handbook, Second Edition.

Eckerstorfer, M (2010). Defining Environment Risk Assessment Criteria for Genetically Modified Insects to be placed on the EU Market. EFSA Supporting Publications. 7. 71E. 10.2903/sp.efsa.2010.EN-71.

Elamasri R . and Navathe, S., Fundamentals of Database Systems (3rd Edition), Pearsson Education, 2000

Enrst & Young, (2019) Integrated Reporting Framework (IReF) & Banks' Integrated Reporting Dictionary (BIRD). Retrieved from https://www.ey.com/Publication/vwLUAssets/EY-FSO-IReF-BIRD-01-2019-en/$FILE/EY-FSO-IReF-BIRD-01-2019-en.pdf

Geographic Information Technology Training Alliance (2005), Separation of Data and Applications, retrieved from http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html

Gorman, K & Choobineh, J (1990) The Object-Oriented Entity-Relationship Model (Ooerm), Journal of Management Information Systems, 7:3, 41-65, DOI: 10.1080/07421222.1990.11517896

Kumar, R. (2016). Graph Analytics. Retrieved from http://www.ranjankumar.in/graph-analytics/

Nebiker, S & Bleisch, S (2005) Introduction to Database Systems. Retrieved from http://www.gitta.info/IntroToDBS/en/text/IntroToDBS.pdf

Pedersen T. (2009) Multidimensional Modeling. In: LIU L., ÖZSU M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA

Piechocki, M. (2012), Supervising Models: XBRL and Data Point modelling, in: iBR interactive business reporting, Vol. 02, No. 3, p. 26-29

Prakash, Naveen: Introduction to Database Management New Delhi: Tata MacGraw Hill, 1991 (p 15-16)

Quagliariello, M & Rimmanen, M (2015). Europe's New Supervisory Toolkit: Data, Benchmarking and Stress Testing for Banks and their Regulators.

Rasmussen, G & Cazemier, H 2003 (2003). Metadata Model. Retrieved from https://patentimages.storage.googleapis.com/be/2d/59/78d4d76892b9e6/US6662188.pdf

Rob, P & Coronel, C (2000) Database Systems: Design, implementation and management - 4th ed. Cambridge, Course Technology, (p 1-55,286-321)

Robinson, I, Webber, J & Eifrem, E (2015). Graph Databases: New opportunities for connected data. O'reilly media, Incorporated.

Sakr, S (2013) Processing Large-scale Graph Data: A guide to Current Technology. Retrieved from :http://www.ibm.com/developerworks/library/os-giraph/

Sapia, C./ et al (1999), Extending the E/R Model for the Multidimensional Paradigm, http://link.springer.com/chapter/10.1007%2F978-3-540-49121-7_9?LI=true

Sasaki, B.M. (2018). Graph Databases for Beginners: Why we need NoSQL Databases. Retrieved from https://neo4j.com/blog/why-nosql-databases/

Shankaranarayanan, G. and Even, Adir (2004) "Managing Metadata in Data Warehouses: Pitfalls and Possibilities," The Communications of the Association for Information Systems: Vol. 14, Article 47. Available at: http://aisel.aisnet.org/cais/vol14/iss1/47

Strohbach M., Daubert J., Ravkin H., Lischka M. (2016) Big Data Storage. In: Cavanillas J., Curry E., Wahlster W. (eds) New Horizons for a Data-Driven Economy. Springer, Cham

Uhrowczik, P (1973) Data Dictionary/Directorie. *IBM Systems Journal*, vol. 12, no. 4, pp. 332-350.doi: 10.1147/sj.124.0332. Retrieved from: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5388203&isnumber=5388202

Vemuganti, G. (2013). Metadata management in big data. Big data: Countering tomorrow's challenges, 3

Watt, A. and N. Eng. (2014). Database Design – 2nd Edition. Victoria, B.C.: BCcampus. Retrieved from https://opentextbc.ca/dbdesign01/.

Weber, A. (2013) Data Point Methodology - Guidance for the preparation of data point models based on European supervisory reporting frameworks, Bachelor thesis of the BW Cooperative State University. May, 2013

Zaitev, P. (2006). Why MySQL Could Be Slow With Large Tables. Retrieved from https://www.percona.com/blog/2006/06/09/why-mysql-could-be-slow-with-large-tables

12 Annexes

12.1 Annex I: Changes to SDD elements for example 2

Domain

MAINTENANCE_AGENCY_ID	DOMAIN_ID	NAME	CODE	DATA_TYPE
ECB	IS_LCL_GVRNMNT_TRTD_PS	Is public sector entity treated as regional government or local authority	IS_LCL_GVRNMNT_TRTD_PS	integer(6)
ECB	IS_CNTRL_GVRNMNT_TRTD_LG	Is regional government or local authority treated as central government	IS_CNTRL_GVRNMNT_TRTD_LG	integer(6)
ECB	IS_CNTRL_GVRNMNT_TRTD_PS	Is public sector entity treated as central government	IS_CNTRL_GVRNMNT_TRTD_PS	integer(6)

Subdomain

MAINTENANCE_AGENCY_ID	SUBDOMAIN_ID	NAME	DOMAIN_ID	IS_LISTED	IS_NATURAL	CODE
ECB	IS_LCL_GVRNMNT_TRTD_PS	Is public sector entity treated as regional government or local authority	IS_LCL_GVRNMNT_TRTD_PS	TRUE	FALSE	IS_LCL_GVRNMNT_TRTD_PS
ECB	IS_CNTRL_GVRNMNT_TRTD_LG	Is regional government or local authority treated as central government	IS_CNTRL_GVRNMNT_TRTD_LG	TRUE	FALSE	IS_CNTRL_GVRNMNT_TRTD_LG
ECB	IS_CNTRL_GVRNMNT_TRTD_PS	Is public sector entity treated as central government	IS_CNTRL_GVRNMNT_TRTD_PS	TRUE	FALSE	IS_CNTRL_GVRNMNT_TRTD_PS

Subdomain enumeration

MEMBER_ID	SUBDOMAIN_ID	VALID_FROM	VALID_TO
IS_CNTRL_GVRNMNT_TRTD_LG_0	IS_CNTRL_GVRNMNT_TRTD_LG	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_CNTRL_GVRNMNT_TRTD_LG_1	IS_CNTRL_GVRNMNT_TRTD_LG	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_CNTRL_GVRNMNT_TRTD_LG_2	IS_CNTRL_GVRNMNT_TRTD_LG	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_CNTRL_GVRNMNT_TRTD_PS_0	IS_CNTRL_GVRNMNT_TRTD_PS	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_CNTRL_GVRNMNT_TRTD_PS_1	IS_CNTRL_GVRNMNT_TRTD_PS	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_CNTRL_GVRNMNT_TRTD_PS_2	IS_CNTRL_GVRNMNT_TRTD_PS	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_LCL_GVRNMNT_TRTD_PS_0	IS_LCL_GVRNMNT_TRTD_PS	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_LCL_GVRNMNT_TRTD_PS_1	IS_LCL_GVRNMNT_TRTD_PS	1900-01-00 0:00:00	9999-12-31 0:00:00
IS_LCL_GVRNMNT_TRTD_PS_2	IS_LCL_GVRNMNT_TRTD_PS	1900-01-00 0:00:00	9999-12-31 0:00:00

Member

MAINTENANCE_AGENCY_ID	MEMBER_ID	CODE	NAME	DOMAIN_ID

ECB	IS_LCL_GVRNMNT_TRTD_PS_1	1	The entity is a public sector entity which is treated as exposure to the regional government or local authority	IS_LCL_GVRNMNT_TRTD_PS
ECB	IS_LCL_GVRNMNT_TRTD_PS_0	0	The entity is not a public sector entity	IS_LCL_GVRNMNT_TRTD_PS
ECB	IS_LCL_GVRNMNT_TRTD_PS_2	2	The entity is a public sector entity which is not treated as exposure to the regional government or local authority	IS_LCL_GVRNMNT_TRTD_PS
ECB	IS_CNTRL_GVRNMNT_TRTD_LG_0	0	The entity is not a regional government or local authority	IS_CNTRL_GVRNMNT_TRTD_LG
ECB	IS_CNTRL_GVRNMNT_TRTD_LG_2	2	The entity is a regional government or local authority which is not treated as exposure to its central government	IS_CNTRL_GVRNMNT_TRTD_LG
ECB	IS_CNTRL_GVRNMNT_TRTD_LG_1	1	The entity is a regional government or local authority which is treated as exposure to its central government	IS_CNTRL_GVRNMNT_TRTD_LG
ECB	IS_CNTRL_GVRNMNT_TRTD_PS_2	2	The entity is a public sector entity which is not treated as exposure to the central government	IS_CNTRL_GVRNMNT_TRTD_PS
ECB	IS_CNTRL_GVRNMNT_TRTD_PS_1	1	The entity is a public sector entity which is treated as exposure to the central government	IS_CNTRL_GVRNMNT_TRTD_PS
ECB	IS_CNTRL_GVRNMNT_TRTD_PS_0	0	The entity is not a public sector entity	IS_CNTRL_GVRNMNT_TRTD_PS

Variable

MAINTENANCE_AGENCY_ID	VARIABLE_ID	CODE	NAME	DOMAIN_ID	DESCRIPTION
ECB	IS_LCL_GVRNMNT_TRTD_PS	IS_LCL_GVRNMNT_TRTD_PS	Is public sector entity treated as regional government or local authority	IS_LCL_GVRNMNT_TRTD_PS	Indicates whether a public sector entity in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) is treated for capital requirements purposes as exposure to the regional government or local authority in accordance with Article 116(4) of Regulation (EU) No 575/2013 (CRR)
ECB	IS_CNTRL_GVRNMNT_TRTD_LG	IS_CNTRL_GVRNMNT_TRTD_LG	Is regional government or local authority treated as central government	IS_CNTRL_GVRNMNT_TRTD_LG	Indicates whether a regional government or local authority is treated for capital requirements purposes as exposure to its central government in accordance with Articles 115(2) of Regulation (EU) No 575/2013 (CRR)
ECB	IS_CNTRL_GVRN	IS_CNTRL_GVRN	Is public	IS_CNTRL_GVRN	Indicates whether a public sector entity

	MNT_TRTD_PS	MNT_TRTD_PS	sector entity treated as central governme nt	MNT_TRTD_PS	in accordance with Article 4(1) of Regulation (EU) No 575/2013 (CRR) is treated for capital requirements purposes as exposure to the central government in accordance with Article 116(4) of Regulation (EU) No 575/2013 (CRR)

Cube

NAME	CODE	FRAMEWO RK_ID	CUBE_STRUCTURE_ID	CUBE_T YPE	IS_ALLO WED	VALID_F ROM	VALID _TO
Public sector entity treated as regional governme nt or local authority	RIAD_IS_LCL_GVRNMNT_TR TD_PS_C	RIAD	RIAD_IS_LCL_GVRNMNT_TRT D_PS_C_1	C	FALSE	1900-01-00 0:00:00	##### ###
Regional governme nt or local authority treated as central governme nt	RIAD_IS_CNTRL_GVRNMNT _TRTD_LG_C	RIAD	RIAD_IS_CNTRL_GVRNMNT_T RTD_LG_C_1	C	FALSE	1900-01-00 0:00:00	##### ###
Public sector entity treated as central governme nt	RIAD_IS_CNTRL_GVRNMNT _TRTD_PS_C	RIAD	RIAD_IS_CNTRL_GVRNMNT_T RTD_PS_C_1	C	FALSE	1900-01-00 0:00:00	##### ###
Public sector entity treated as regional governme nt or local authority	RIAD_IS_LCL_GVRNMNT_TR TD_PS_S	RIAD	RIAD_IS_LCL_GVRNMNT_TRT D_PS_S_1	S	FALSE	1900-01-00 0:00:00	##### ###
Regional governme nt or local authority treated as central governme nt	RIAD_IS_CNTRL_GVRNMNT _TRTD_LG_S	RIAD	RIAD_IS_CNTRL_GVRNMNT_T RTD_LG_S_1	S	FALSE	1900-01-00 0:00:00	##### ###
Public sector entity treated as central governme nt	RIAD_IS_CNTRL_GVRNMNT _TRTD_PS_S	RIAD	RIAD_IS_CNTRL_GVRNMNT_T RTD_PS_S_1	S	FALSE	1900-01-00 0:00:00	##### ###

Cube structure

CUBE_STRUCTURE_ID	NAME	CODE	Descriptio n	VALID_FROM	VALID_T O
RIAD_IS_LCL_GVRNMNT_TRTD_PS_C_ 1	Public sector entity	RIAD_IS_LCL_GVRNMNT_TRTD_PS_C		########### ##	9999-12-31 0:00:00

	treated as regional government or local authority				
RIAD_IS_CNTRL_GVRNMNT_TRTD_LG_C_1	Regional government or local authority treated as central government	RIAD_IS_CNTRL_GVRNMNT_TRTD_LG_C		#############	9999-12-31 0:00:00
RIAD_IS_CNTRL_GVRNMNT_TRTD_PS_C_1	Public sector entity treated as central government	RIAD_IS_CNTRL_GVRNMNT_TRTD_PS_C		#############	9999-12-31 0:00:00
RIAD_IS_LCL_GVRNMNT_TRTD_PS_S_1	Public sector entity treated as regional government or local authority	RIAD_IS_LCL_GVRNMNT_TRTD_PS_S		#############	9999-12-31 0:00:00
RIAD_IS_CNTRL_GVRNMNT_TRTD_LG_S_1	Regional government or local authority treated as central government	RIAD_IS_CNTRL_GVRNMNT_TRTD_LG_S		#############	9999-12-31 0:00:00
RIAD_IS_CNTRL_GVRNMNT_TRTD_PS_S_1	Public sector entity treated as central government	RIAD_IS_CNTRL_GVRNMNT_TRTD_PS_S		#############	9999-12-31 0:00:00

Cube structure item

CUBE_STRUCTURE_ID	CUBE_VARIABLE_CODE	VARIABLE_ID	ROLE	ORDER	SUBDOMAIN_ID
RIAD_IS_LCL_GVRNMNT_TRTD_PS_C_1	OBS_VALUE	IS_LCL_GVRNMNT_TRTD_PS	O	5	IS_LCL_GVRNMNT_TRTD_PS
RIAD_IS_CNTRL_GVRNMNT_TRTD_LG_C_1	OBS_VALUE	IS_CNTRL_GVRNMNT_TRTD_LG	O	5	IS_CNTRL_GVRNMNT_TRTD_LG
RIAD_IS_CNTRL_GVRNMNT_TRTD_PS_C_1	OBS_VALUE	IS_CNTRL_GVRNMNT_TRTD_PS	O	5	IS_CNTRL_GVRNMNT_TRTD_PS
RIAD_IS_LCL_GVRNMNT_TRTD_PS_S_1	OBS_VALUE	IS_LCL_GVRNMNT_TRTD_PS	O	5	IS_LCL_GVRNMNT_TRTD_PS
RIAD_IS_CNTRL_GVRNMNT_TRTD_LG_S_1	OBS_VALUE	IS_CNTRL_GVRNMNT_TRTD_LG	O	5	IS_CNTRL_GVRNMNT_TRTD_LG
RIAD_IS_CNTRL_GVRNMNT_TRTD_PS_S_1	OBS_VALUE	IS_CNTRL_GVRNMNT_TRTD_PS	O	5	IS_CNTRL_GVRNMNT_TRTD_PS

1.1 Annex II: Relationship between DPM/XBRL and SMCube/SDD model components

DPM/XBRL component	SMCube/SDD component	Comment
Dictionary/Framework element	IdentifiableArtifact	DPM Dictionary or Framework element is in general represented by SDD foundation concept named IdentifiableArtifact. It is used to uniquely identify an element by a code/name and carries information about historicity, maintaining authority and furthermore through SDD Annotation package provides additional documentation and descriptions. As the scope of the SDD packages is larger comparing to the DPM/XBRL, there are some SDD IdentifiableArtifacts that are not represented in the DPM/XBRL. Also DPM/XBRL reference model does not in general define individual data points or dimensional combinations as identifiable concepts which is the case of SDD (and some of the DPM/XBRL specific implementations, e.g. the EBA DPM database defines codes for data points). Additionally some XBRL intrinsic attributes (other fact aspects) are predefined properties of a report not identifiable in the model.
Dictionary/Framework element code (name)	IdentifiableArtifact uniqueID	SDD codification may differ from the codification used in the DPM (e.g. use of two or more capital letters for domains and dimensions, first letter of a metric code identifying its type, etc.). Translation between the two is done in the Mapping package when migrating metadata from/to the DPM. Uniqueness must be assured on certain level (e.g. for dimensions, domains, members of a domain or relationship sets for a given owner). External codification may be used if applicable (e.g. ISO codes).
Dictionary/Framework element creation/modification dates; Dictionary element currency (fromDate/toDate)	IdentifiableArtifact validFrom/validTo	Declaration of DPM Dictionary elements: dimensions, domains and members as well as DPM Framework elements: frameworks, modules, taxonomies and table groups may include creation and last modification dates. Although technically not implemented in XBRL, these attributes could be also applied on DPM Dictionary relationship sets, and other DPM Framework elements such as tables, axes and nodes. DPM Dictionary element may contain fromDate and toDate attributes that establish its currency period. They are optional so element may have only fromDate, only toDate, both or none. Elements can be closed and subsequently reopened - only their current status is represented. These attributes don’t have any impact on the reporting

DPM/XBRL component	SMCube/SDD component	Comment
		process itself - they are meant to facilitate the management of the elements of the dictionary. SDD IdentifiableArtifact has validFrom and validTo dates. validFrom is the counterpart of the DPM Dictionary/Framework element creation date. validTo represents the DPM Dictionary element toDate (however it may be applied also on the counterparts of DPM Framework elements and is obligatory therefore for elements in use the assigned value is 31 December 9999). SMCube reference dictionary content guidelines aims to address in one of its chapters the topic of historicity which is not explicitly regulated in any of the DPM reference materials or implementations. For more detailed description see the “Maintenance: versioning and extensions” chapter of this report.
Metric	StructureItems: ObservationValue, Attribute and Dimension	SDD does not have a specific dictionary elements to represent metrics. Instead DPM Metric equivalent information is specified as a combination of the StructureItems: ObservationValue, Attribute and Dimension that by reference to certain subdomains inform respectively about the period type (ObvservationValue IsFlow attribute), data type and sematic information included in the DPM definition of a metric. This allows for more flexibility when constructing data definitions but makes a model more complex in definition, maintenance and application. Also to enable consumption of XBRL necessary data it is necessary to define a ComplementaryStructure with properties included in other fact aspects (as described later in this table).
Domain	Domain	SDD Domain is much broader than its DPM counterpart and includes DPM Domains and moreover also DPM Metrics, data types and other fact aspects.
Typed domain	Domain: isEnumerated = false (open domain)	In general DPM Typed domain can be represented by SMCube Domain where isEnumerated attribute is equal “false”. Some DPM Typed domains may be also represented by SDD Explicit domains (e.g. DPM Typed domain of boolean type is represented by SDD explicit domain with

DPM/XBRL component	SMCube/SDD component	Comment
		members “True” and “False” or data type integer from range 0 to 1 inclusive).
Explicit domain	Domain: isEnumerated = true	DPM Explicit domain can be represented by SDD Domain where isEnumerated attribute is equal “true”.
Dimension (explicit or typed)	Variable (used in Structure as Dimension)	DPM Dimension is represented as SDD Variable whose role in a structure is Dimension.
Member	Member	In general DPM Member is equivalent to SDD Member (which additionally may represent also DPM Metric, data type enumerated value or other fact aspect value). SDD Member contains isAggregated attribute that identifies if a member has children in any hierarchy in the dictionary. This attribute value does not have any direct DPM counterpart but could be computed based on information in hierarchies (therefore its explicit declaration is questionable especially that a change of its value does not seem to impact the semantics and therefore shall be addressed specifically in the management procedure from the versioning standpoint).
Default member	Default member; isImplicit Dimension StructureItem	DPM Domains contain default members applicable to all dimensions referring to a domain. In most of the cases they represent “Total/Not-applicable” or “Not-applicable” member but sometimes they may take a specific value. SDD Core package seems not to contain any property marking a particular member as a default value. Moreover, SDD proposal for “Not relevant” as the default member (described in one of the chapters of SMCube reference dictionary content guidelines) may need to be considered in detailed especially in Mapping package and for further use of DPM/XBRL data. Additionally it is not clear if isImplicit attribute applied for a Dimension in a DataStructureDefinition could potentially override the globally defined default member.
Relationship set	Hierarchy, Pattern	DPM Relationship set represents SDD Hierarchy and may be used to represent SDD enumerated domain pattern (by listing as all members fulfilling the patterns filtering criterion).

DPM/XBRL component	SMCube/SDD component	Comment
Relationship node and arc	Hierarchy Node	SDD Hierarchy node contains information carried by DPM Relationship node and arc (similarly to specific DPM implementations in database of EBA and EIOPA these two XBRL artefacts are merged in one). Relation is expressed by parentNodeId attribute and additional (redundant) level attribute. SDD Hierarchy node can only be used once in a hierarchy while in DPM a metric or member could theoretically be used many times (under different parents). DPM Relationship arc order attribute seem to be missing in the SDD representation and shall be potentially added.
Relationship set	Subdomain	SDD Subdomain is a list of values applicable in a particular subset of information requirements (without hierarchical relationships as these could be exclusive). DPM dictionary is defined independently from any particular information requirements therefore subdomains in the SDD sense do not exist in any of the current DPMs but can be computed if needed and included in the DPM models by means of relationship sets (apart from Subdomains for SDD Domains that are not DPM Domains or metrics).
Metric or typed domain data type	Domain data type and format, Subdomain	As described in the SMCube reference dictionary content guidelines, the basic SDD data types are string, monetary, integer, real and percentage. DPM has a set of data types (monetary, decimal, percentage, integer, boolean, date and URI) that can be further extended or restricted (in the model or with validation rules). In particular the DPM boolean type is planned to be represented in the SDD as a domain with two members representing true/1 and false/0. Enumerated data types are expressed in the DPM by a reference from a metric to a relationship set of members representing the allowed values while in the SDD as an observation value referring to a subdomain. Therefore the hierarchical information is not resembled in the SDD comparing to the DPM counterpart.
Framework	Framework	DPM and SDD Framework seem to be equivalent concepts. However their structure and content is different. SDD Framework relates to DataFlows and RenderingTables while DPM Frameworks are versioned as taxonomies that are sets of tables (defining graphical representation as well as

DPM/XBRL component	SMCube/SDD component	Comment
		valid combinations in hypercubes) gathered in modules.
Taxonomy	Framework regulation attribute	DPM Taxonomy not to be present in the SDD model. It can be represented by multiple Frameworks or by means of versioning of Data Flows.
Module	Data definition package provisioning information; CategoryScheme Category and DataFlowPackage	There is no direct counterpart of the DPM Module in the SDD. It could be represented by Data definition package Provisioning information and CategoryScheme Category and DataFlowPackage.
Axis	Axis	DPM and SDD Axis seem to be equivalent concepts. SDD axis orientation is defined as a string but the set of values is limited to “x”, “y” and “z”. DPM Table must contain at least one “x” and one “y” axis. SDD Table also requires at least two axes but the analysed materials do not prescribe their orientation (potentially allowing “x” and “z” which may be impossible to render). Also it is not defined if SDD table may contain multiple axis of the same orientation (which if not allowed could be overcome by merging in a single SDD axis multiple DPM axis as a Cartesian product of their nodes or values).
Data point	Combination	The DPM refers to single cells within the

1.2 Annex III: VTL Transformations for FinRep template 5.01

Transformation ID	Description	Expression
Derivations		
D_INSTTTNL_SCTR_FINREP_1_0	Returns the institutional sector classification used in FinRep taking into account multilateral development banks and international organisations.	define operator D_INSTTTNL_SCTR_FINREP(INSTTTNL_SCTR scalar, INTRNTNL_ORGNSTN_CD scalar) returns string is if (INSTTTNL_SCTR in {"S122_A"} or INTRNTNL_ORGNSTN_CD in MLTLTRL_DVLPMNT_BNKS_FINREP) then "S1_MDB" else if (INSTTTNL_SCTR in Sectors04 or INTRNTNL_TRGNSTN_CD in INTRNTNL_ORGNSTNS_FINREP) then "S13_IO" else INSTTTNL_SCTR end operator
D_TYP_INSTRMNT_ASSTS_FINREP_1_0	Derivation of Type of Instrument	D_TYP_INSTRMNT_ASSTS_FINREP(TYP_INSTRMNT scalar, RPYMNT_RGHTS scalar) returns string is if (RPYMNT_RGHTS in {"1"}) then "031" else if (TYP_INSTRMNT in {"090"}) then "1003" else TYP_INSTRMNT end operator
D_CRRYNG_AMNT_1_0	Derivation of carrying amount	define operator D_CRRYNG_AMNT(ACCNTNG_CLSSFCTN scalar, FV scalar, GRSS_CRRYNG_AMNT_E_INTRST scalar, ACCRD_INTRST scalar, FV_CHNG_HDG_ACCNTNG scalar, ACCMLTD_IMPRMNT scalar, CRRYNG_AMNT scalar, IS_CRRYNG_AMNT_DRVD scalar) returns integer is if (IS_CRRYNG_AMNT_DRVD = "T") then if (ACCNTNG_CLSSFCTN in FV_ACCNTNG_CLSSFCTNS) then FV else GRSS_CRRYNG_AMNT_E_INTRST + ACCRD_INTRST else CRRYNG_AMNT end operator
D_CRRYNG_AMNT_LNS_ADVNCS_1_0	Apply the derivation for the Carrying amount (CRRYNG_AMNT) for Loans and advances - enriched (LNS_ADVNCS_E)	D_CRRYNG_AMNT_LNS_ADVNCS := LNS_ADVNCS [calc measure CRRYNG_AMNT := D_CRRYNG_AMNT(ACCNTNG_CLSSFCTN, FV, GRSS_CRRYNG_AMNT_E_INTRST, ACCRD_INTRST, FV_CHNG_HDG_ACCNTNG, ACCMLTD_IMPRMNT, CRRYNG_AMNT, CNSTNT_IS_CRRYNG_AMNT_DRVD)]
D_CRRYNG_AMNT_LNS	Keep only relevant variables	D_CRRYNG_AMNT_LNS_ADVNCS := D_CRRYNG_AMNT_LNS_ADVNCS [keep INSTRMNT_UNQ_ID,

_ADVNCS_1_1		DT_RFRNC, PRSPCTV_ID, CRRYNG_AMNT];
D_GRSS_CRRYNG_AMNT_1_0	Derivation of gross carrying amount	define operator D_GRSS_CRRYNG_AMNT(ACCNTNG_CLSSFCTN scalar, CRRYNG_AMNT scalar, ACCMLTD_IMPRMNT scalar, ACCMLTD_CHNGS_FV_CR scalar, CRDT_QLTY_STTS scalar, IMPRMNT_STTS scalar, ACCMLTD_IMPRMNT_GAAP1 scalar, ACCMLTD_IMPRMNT_GAAP2 scalar) returns integer is if (ACCNTNG_CLSSFCTN in {"8", "6"}) then CRRYNG_AMNT + ACCMLTD_IMPRMNT else if (ACCNTNG_CLSSFCTN in {"41", "4"} and CRDT_QLTY_STTS in {"18", "19", "2", "20"}) then if (ACCMLTD_CHNGS_FV_CR < 0) then CRRYNG_AMNT + ACCMLTD_CHNGS_FV_CR else CRRYNG_AMNT else if (ACCNTNG_CLSSFCTN in {"41", "4"} and CRDT_QLTY_STTS = "11") then CRRYNG_AMNT else if (ACCNTNG_CLSSFCTN = "14" and ACCMLTD_IMPRMNT > 0) then CRRYNG_AMNT + ACCMLTD_IMPRMNT else if (ACCNTNG_CLSSFCTN = "14" and CRDT_QLTY_STTS in {"18", "19", "2", "20"}) then CRRYNG_AMNT + ACCMLTD_CHNGS_FV_CR else if (ACCNTNG_CLSSFCTN = "14" and CRDT_QLTY_STTS = "11") then CRRYNG_AMNT else if (ACCNTNG_CLSSFCTN = "7" and CRDT_QLTY_STTS in {"18", "19", "2", "20"}) then if (ACCMLTD_CHNGS_FV_CR < 0) then CRRYNG_AMNT + ACCMLTD_CHNGS_FV_CR else CRRYNG_AMNT else if (ACCNTNG_CLSSFCTN = "7" and CRDT_QLTY_STTS = "11") then CRRYNG_AMNT else if (ACCNTNG_CLSSFCTN = "9" and IMPRMNT_STTS = "0" or ACCNTNG_CLSSFCTN = "82" and CRDT_QLTY_STTS in {"18", "19", "2", "20"}) then if (ACCMLTD_CHNGS_FV_CR < 0) then CRRYNG_AMNT + ACCMLTD_CHNGS_FV_CR else CRRYNG_AMNT else if (ACCNTNG_CLSSFCTN = "9" and IMPRMNT_STTS = "0" or ACCNTNG_CLSSFCTN = "82" and CRDT_QLTY_STTS = "11") then CRRYNG_AMNT else if (ACCNTNG_CLSSFCTN = "9" and IMPRMNT_STTS = "26" or ACCNTNG_CLSSFCTN = "81" and IMPRMNT_STTS = "26") then CRRYNG_AMNT + ACCMLTD_IMPRMNT_GAAP1 else if (ACCNTNG_CLSSFCTN = "9" and IMPRMNT_STTS = "21" or ACCNTNG_CLSSFCTN = "81" and IMPRMNT_STTS = "21") then CRRYNG_AMNT + ACCMLTD_IMPRMNT_GAAP1 else if (ACCNTNG_CLSSFCTN in {"73", "74", "76", "77"}) then CRRYNG_AMNT + ACCMLTD_IMPRMNT_GAAP1 else null end operator
D_PRFRMNG_STTS_1_0	Derivation of performing status of an instrument taking into account the	define operator D_PRFRMNG_STTS(APPRCH_CRDT_QLTY_STTS scalar, CRDT_QLTY_STTS scalar, CRDT_QLTY_STTS_CNTRPRTY scalar, IMPRMNT_STTS scalar) returns string is if (CRDT_QLTY_STTS_CNTRPRTY in {"18", "19", "2", "20"} and APPRCH_CRDT_QLTY_STTS in {"1"} or

	debtor / customer information.	CRDT_QLTY_STTS_INSTRMNT in {"18", "19", "2", "20"} and APPRCH_CRDT_QLTY_STTS in {"2"}) then if (CRDT_QLTY_STTS_CNTRPRTY in {"18", "19", "20"} and APPRCH_CRDT_QLTY_STTS in {"1"} or CRDT_QLTY_STTS_INSTRMNT in {"18", "19", "20"} and APPRCH_CRDT_QLTY_STTS in {"2"}) then "12" else if (IMPRMNT_STTS in {"25"}) then "13" else "1" else if (CRDT_QLTY_STTS_CNTRPRTY in {"11"} and APPRCH_CRDT_QLTY_STTS in {"1"} or CRDT_QLTY_STTS_INSTRMNT in {"11"} and APPRCH_CRDT_QLTY_STTS in {"2"}) then "11" else "0" end operator
D_CNTRY_FINREP_1_0	Returns the country classification used in FinRep taking into account multilateral development banks and international organisations.	define operator D_CNTRY_FINREP(CNTRY scalar, INTRNTNL_ORGNSTN_CD scalar) returns string is if (INTRNTNL_ORGNSTN_CD in MLTLTRL_DVLPMNT_BNKS_FINREP or INTRNTNL_TRGNSTN_CD in INTRNTNL_ORGNSTNS_FINREP) then "_X" else CNTRY end operator
Input Layer → Enriched Input Layer		
Counterparties Enriched		
P_CNTRPRTS_E_2_4	Derivation of the Institutional sector (INSTTTNL_SCTR) w.r.t. FinRep taking into consideration the classification of multilateral development banks	CNTRPRTS_E := CNTRPRTS_E [calc INSTTTNL_SCTR_FINREP := D_INSTTTNL_SCTR_FINREP(INSTTTNL_SCTR, INTRNTNL_ORGNSTN_CD)]
P_CNTRPRTS_E_2_5	Derivation of the Country (CNTRY) w.r.t. FinRep taking into consideration the classification of multilateral	CNTRPRTS_E := CNTRPRTS_E [calc CNTRY_FINREP := D_CNTRY_FINREP(CNTRY, INTRNTNL_ORGNSTN_CD)]

	development banks	
Loans Enriched		
P_LNS_E_1_0	Instruments that are considered loans are put together in the cube Loans - enriched (LNS_E)	LNS_E := union(CRDT_CRD_DBT_E, FCTRNG_E, FNNCL_LSS_E, OTHR_DPSTS_E, OTHR_LNS_E, OTHR_TRD_RCVBLS_E, RVRS_RPRCHS_LNS_E, CRRNT_ACCNT_ASSTS_E)
DB_ADVNCS_NT_LNS_E_1_0	Write the data set 'Advances (that are not loans) - enriched (ADVNCS_NT_LNS_E)' to the data base	ADVNCS_NT_LNS_E <- ADVNCS_NT_LNS_E
Loans and Advances Enriched		
P_LNS_ADVNCS_E_1_0	Loans and advances comprises Loans - enriched (LNS_E) and Advances (that are not loans) - enriched (ADVNCS_E)	LNS_ADVNCS_E := union(LNS_E, ADVNCS_NT_LNS_E)
P_LNS_ADVNCS_E_1_1	Drop the carrying amount	LNS_ADVNCS_E := LNS_ADVNCS_E [drop CRRYNG_AMNT]
P_LNS_ADVNCS_E_1_2	Add the variable Carrying amount (CRRYNG_AMNT) that is either input or derived via the derivation rule	LNS_ADVNCS_E := inner_join(LNS_ADVNCS_E as A, D_CRRYNG_AMNT_LNS_ADVNCS as B using INSTRMNT_UNQ_ID keep A#ALL_VARIABLES, B#CRRYNG_AMNT)

Enriched Input Layer → Reference Output Layer		
Counterparties		
P_CNTRPRTS_FINREP_1_0	Writing the counterparites FinRep cube to include the entire Counterparties Enriched cube	CNTRPRTS_FINREP := CNTRPRTS_E
P_CNTRPRTS_FINREP_1_1	For FinRep purposes the Institutional sector needs to take into account International organisations as well as Multilateral development banks	CNTRPRTS_FINREP := CNTRPRTS_FINREP [calc INSTTTNL_SCTR := INSTTTNL_SCTR_FINREP]
P_CNTRPRTS_FINREP_1_2	For FinRep purposes the Country needs to take into account International organisations as well as Multilateral development banks	CNTRPRTS_FINREP := CNTRPRTS_FINREP [calc CNTRY := CNTRY_FINREP]
Loans Advances Debtors Enriched		
P_LNS_ADVNCS_DBTRS_FINREP_1_0	Adding the Counterparty identifier (CNTRPRTY_ID)	LNS_ADVNCS_DBTRS_FINREP := inner_join(LNS_ADVNCS_FINREP as A, INSTRMNTS_CSTMRS_E as B using INSTRMNT_UNQ_ID)
P_LNS_ADVNCS_DBTRS_FINREP_1_1	Adding information about the counterparty	LNS_ADVNCS_DBTRS_FINREP := inner_join(LNS_ADVNCS_DBTRS_FINREP as A, CNTRPRTS_FINREP as B using CNTRPRTY_ID)

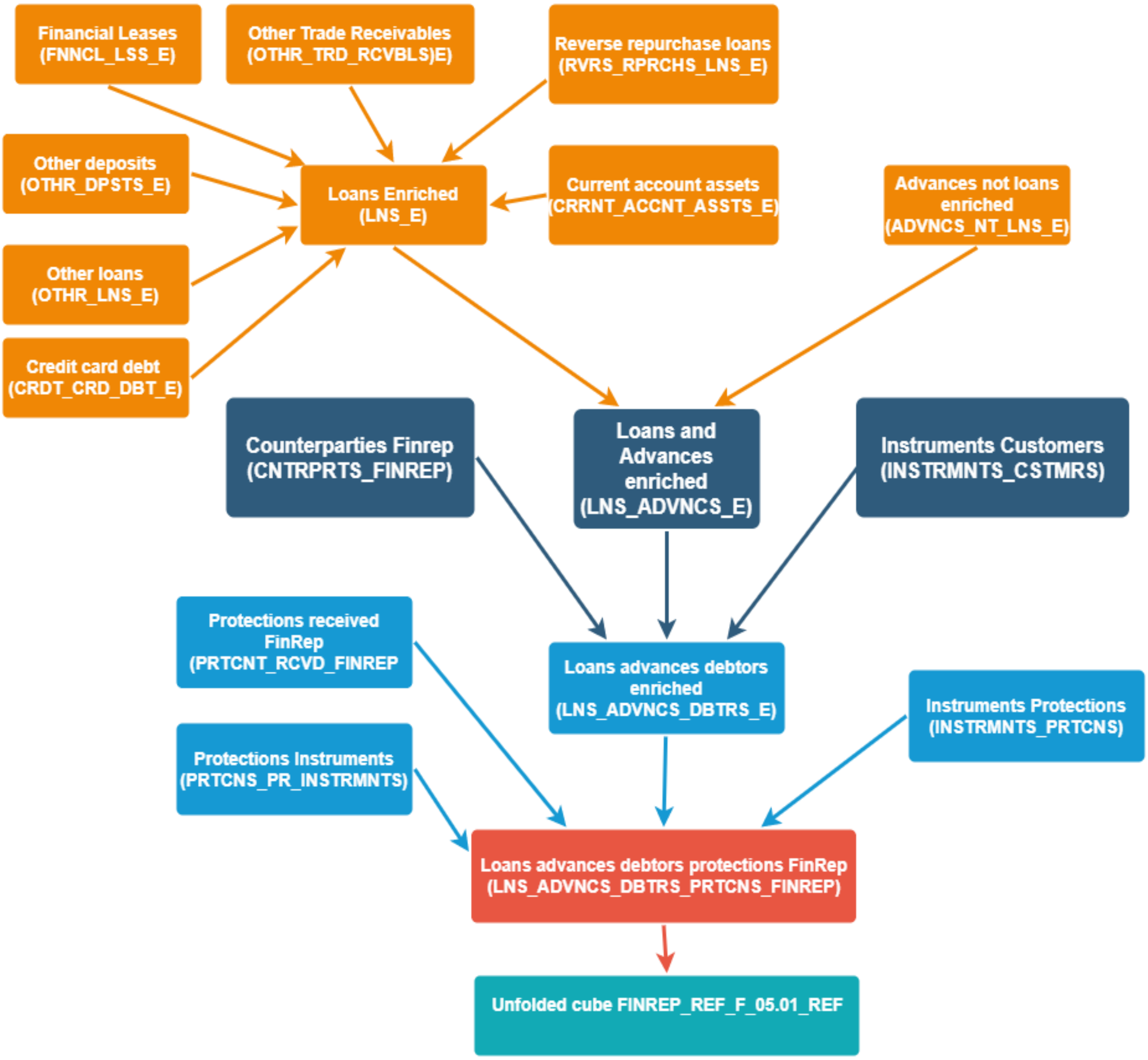
P_LNS_ADVNCS_DBTRS_FINREP_1_3	Derivation of the performing status	LNS_ADVNCS_DBTRS_FINREP := LNS_ADVNCS_DBTRS_FINREP [calc PRFRMNG_STTS := D_PRFRMNG_STTS(APPRCH_CRDT_QLTY_STTS, CRDT_QLTY_STTS, CRDT_QLTY_STTS_CNTRPRTY, IMPRMNT_STTS)]
D_CRRYNG_AMNT_LNS_ADVNCS_1_0	Apply the derivation for the Carrying amount to the Loans and advances enriched	D_CRRYNG_AMNT_LNS_ADVNCS := LNS_ADVNCS [calc measure CRRYNG_AMNT := D_CRRYNG_AMNT(ACCNTNG_CLSSFCTN, FV, GRSS_CRRYNG_AMNT_E_INTRST, ACCRD_INTRST, FV_CHNG_HDG_ACCNTNG, ACCMLTD_IMPRMNT, CRRYNG_AMNT, CNSTNT_IS_CRRYNG_AMNT_DRVD)]
D_CRRYNG_AMNT_LNS_ADVNCS_1_1	Keep only relevant variables	D_CRRYNG_AMNT_LNS_ADVNCS := D_CRRYNG_AMNT_LNS_ADVNCS [keep INSTRMNT_UNQ_ID, DT_RFRNC, PRSPCTV_ID, CRRYNG_AMNT];
Protections Received FinRep		
P_PRTCTNS_RCVD_FINREP_1_0	Keep relevant variables from not real estate cube	PRTCTNS_NT_RL_ESTT_FINREP := PRTCTNS_NT_RL_ESTT_E [keep PRTCTN_ID, TYP_PRTCTN, FNNCL_GRNT_CNSDRD_MX_AMNT];
P_PRTCTNS_RCVD_FINREP_1_1	Keep relevant variables from real estate cube	PRTCTNS_RL_ESTT_FINREP := PRTCTNS_RL_ESTT_E [keep PRTCTN_ID, TYP_PRTCTN];
P_PRTCTNS_RCVD_FINREP_1_2	Calculate the Maximum amount of guarantee that can be considered for financial guarantees	PRTCTNS_RL_ESTT_FINREP := PRTCTNS_RL_ESTT_FINREP [calc FNNCL_GRNT_CNSDRD_MX_AMNT := null]
P_PRTCTNS_RCVD_FINREP_1_3	Join the real estate and not real estate cubes	PRTCTNS_RCVD_FINREP := union(PRTCTNS_NT_RL_ESTT_FINREP, PRTCTNS_RL_ESTT_FINREP)
P_PRTCTNS_RCVD_FIN	Create a subset of structure {Protection identifier, Type	PRTCTNS_RCVD_FINREP_SUB := PRTCTNS_RCVD_FINREP [keep PRTCTN_ID, TYP_PRTCTN];

REP_1_4	of protection}	
P_PRTCTNS_RCVD_FINREP_1_5	Evaluate each protection in order to derive a structure like {Protection identifier, Is residential collateral and financial guarantee, Is commercial collateral and financial guarantee}	PRTCTNS_RCVD_FINREP_SUB := EVLT_TYP_CLLTRL_GRNT(PRTCTNS_RCVD_FINREP_SUB, PRTCTN_ID)
P_PRTCTNS_RCVD_FINREP_1_6	Consolidate the Protections Received cube, keeping all relevant variables.	PRTCTNS_RCVD_FINREP := inner_join(PRTCTNS_RCVD_FINREP as A, PRTCTNS_RCVD_FINREP_SUB as B using PRTCTN_ID keep A#PRTCTN_ID, A#FNNCL_GRNT_CNSDRD_MX_AMNT, A#TYP_PRTCTN, B#IS_RSTNTL_CLLTRL_FNNCL_GRNT, B#IS_CMMRCL_CLLTRL_FNNCL_GRNT)
Loans Advances Debtors Protections Enriched		
P_LNS_ADVNCS_DBTRS_PRTCTNS_FINREP_1_0	Join with Instruments Protections Cube	LNS_ADVNCS_DBTRS_PRTCTNS_FINREP := specialLeftJoin(LNS_ADVNCS_DBTRS_FINREP, INSTRMNTS_PRTCTNS_E)
P_LNS_ADVNCS_DBTRS_PRTCTNS_FINREP_1_1	Join with Protections Received	LNS_ADVNCS_DBTRS_PRTCTNS_FINREP := inner_join(LNS_ADVNCS_DBTRS_PRTCTNS_FINREP as A, PRTCTNS_RCVD_FINREP as B using PRTCTN_ID)
P_LNS_ADVNCS_DBTRS_PRTCTNS_FINREP_1_10	Division of monetary value due to (possible) duplication of records w.r.t. protections (i.e. one instrument may be related to many protections)	LNS_ADVNCS_DBTRS_PRTCTNS_FINREP := LNS_ADVNCS_DBTRS_PRTCTNS_FINREP [calc ACCMLTD_IMPRMN_NGTV_CHNG_FV_CR := ACCMLTD_IMPRMN_NGTV_CHNG_FV_CR / NMBR_OF_PRTCTNS]
P_LNS_ADVNCS_DBTRS_PRTCTNS_FINREP_1_7	Add the variable Number of protections	LNS_ADVNCS_DBTRS_PRTCTNS_FINREP := inner_join(LNS_ADVNCS_DBTRS_PRTCTNS_FINREP as A, PRTCTNS_PR_INSTRMNT as B using INSTRMNT_UNQ_ID)

	(NMBR_OF_PRTCTNS)	
Generation of the unfolded version of the cube		
G_F_05_01_REF_UNFLDD_FINREP_1_2	Write function to specify that unfolded cube is made up of current enriched cube.	F_05_01_REF_UNFLDD := LNS_ADVNCS_DBTRS_PRTCTNS_FINREP
G_F_05_01_REF_UNFLDD_FINREP_1_0	Variables restricted to those relevant to the reference output layer. Sum the amounts reported and group by the different rows to be divided by rows of the template.	LNS_ADVNCS_DBTRS_PRTCTNS_FINREP := LNS_ADVNCS_DBTRS_PRTCTNS_FINREP [aggr CRRYNG_AMNT := sum(CRRYNG_AMNT), GRSS_CRRYNG_AMNT := sum(GRSS_CRRYNG_AMNT) group by ACCNTNG_CLSSFCTN, DT_RFRNC, INSTTTNL_SCTR, IS_HFS, OBSRVD_AGNT_INTRNL_ID, PRJCT_FNNC_LN, PRPS, PRSPCTV_ID, RPYMNT_RGHTS, TYP_ACCNTNG_ITM, TYP_CLLTRL_GRNT_GVN, TYP_INSTRMNT];
G_F_05_01_REF_UNFLDD_FINREP_1_1	Assign roles to each variable where identifier assigns the role 'Dimension' and measure assigns the role 'Observation Value'	LNS_ADVNCS_DBTRS_PRTCTNS_FINREP := LNS_ADVNCS_DBTRS_PRTCTNS_FINREP [calc identifier ACCNTNG_CLSSFCTN := ACCNTNG_CLSSFCTN, identifier DT_RFRNC := DT_RFRNC, identifier INSTTTNL_SCTR := INSTTTNL_SCTR, identifier IS_HFS := IS_HFS, identifier OBSRVD_AGNT_INTRNL_ID := OBSRVD_AGNT_INTRNL_ID, identifier PRJCT_FNNC_LN := PRJCT_FNNC_LN, identifier PRPS := PRPS, identifier PRSPCTV_ID := PRSPCTV_ID, identifier RPYMNT_RGHTS := RPYMNT_RGHTS, identifier TYP_ACCNTNG_ITM := TYP_ACCNTNG_ITM, identifier TYP_CLLTRL_GRNT_GVN := TYP_CLLTRL_GRNT_GVN, identifier TYP_INSTRMNT := TYP_INSTRMNT, measure CRRYNG_AMNT := CRRYNG_AMNT, measure GRSS_CRRYNG_AMNT := GRSS_CRRYNG_AMNT];
G_F_05_01_REF_UNFLDD_FINREP_1_3	Transcoding the members of the variable Institutional sector (INSTTTNL_SCTR) w.r.t. the granularity expected in FinRep	F_05_01_REF_UNFLDD := hierarchy(F_05_01_REF_UNFLDD, INSTTTNL_SCTR_HIER_FINREP rule INSTTTNL_SCTR)

G_F_05_01_REF_UNFLDD_FINREP_1_4	Apply filter to restrict accounting classification to only allow member (60) 'Financial assets other than Held for trading and Trading Financial Assets (60)'	F_05_01_REF_UNFLDD := F_05_01_REF_UNFLDD [filter ACCNTNG_CLSSFCTN in {"60"}]
G_F_05_01_REF_UNFLDD_FINREP_1_5	Apply filter to restrict the variable 'Is held for sale' (IS_HFS) to members: False (F)	F_05_01_REF_UNFLDD := F_05_01_REF_UNFLDD [filter IS_HFS in {"F"}]
G_F_05_01_REF_UNFLDD_FINREP_1_6	Apply filter to restrict variable Type of accounting item to member 'Financial instruments. Creditor (40)'	F_05_01_REF_UNFLDD := F_05_01_REF_UNFLDD [filter TYP_ACCNTNG_ITM in {"40"}]
Folding of the Cube		
FOLD_F_05_01_REF_UNFLDD_F_05_01_REF_1_0	Folds the Observations of a data set into the combination of a Metric (MTRCS) and Observation value (OBSERVATION_VALUE	F_05_01_REF := F_05_01_REF_UNFLDD [unpivot OBSERVATION_VALUE, MTRCS]
FOLD_F_05_01_REF_UNFLDD_F_05_01_REF_1_2	Application of member hierarchies and aggregation in order to derive combinations of the cube F_05_01_REF	F_05_01_REF := APPLY_HIERARCHIES_AND_AGGREGATION(F_05_01_REF)

1.3 Annex IV: Transformation process



12.2 Annex V: Glossary

	Information model	Data Dictionary	Dataset	Validation and transformation language	Harmonised Primary reporting	Harmonised Secondary reporting	Data exchange format	Data platform
SMCube	X							
SDMX IM	X							
SDD		X						
DPM		X						
SDMX DSD		X						
BIRD		X						
BIRD input			X					
VTL				X				
COREP (*)			X		X (DPM)			
FINREP (*)			X		X (DPM)			
BSI			X			X		
MIR			X			X		
AnaCredit			X			X (SDD)		
SHSG			X			X (SDMX)		
IReF			X		X (SDD)			
RIAD			X			X (SDD)		
CSDB			X			X		
XBRL file							X	
SDMX file							X	
Other XML files							X	
DISC								X