

SDMX USER GUIDE

VERSION 2009.1

January 2009

Foreword

This User Guide aims to provide explanations and guidance to users and potential users of SDMX.

Version 2009.1 is the second release of the User Guide as a complete document. In this release, the material has been separated into two main parts:

Part A - the *core thread* - aims to explain SDMX in a non-technical way, dealing with the use of SDMX at all stages of the production and dissemination of statistics. This part of the User Guide should be accessible to statisticians without any special knowledge or interest in informatics.

Part B - the *tutorial thread* - is aimed at readers who want more in-depth technical information on SDMX. It assumes that the reader wants to work with, or at least to understand, the information technologies which underly SDMX.

Of course, many readers will want to look at both parts of the User Guide, which has been structured to make it easy for readers to work through the core thread while, if they wish, seeking further explanations or examples in specific chapters of the tutorial thread.

The emphasis throughout the User Guide is on practical use of SDMX, supported by examples. No prior knowledge of SDMX is assumed.

The SDMX User Guide is based partly on explanatory documents and tutorials from the SDMX sponsoring organisations, together with new material written for the user Guide. In addition to the examples in the text, some sections of the User Guide refer to example files which are too long to be included in the document; these files will be available for download with the electronic version of the User Guide, which is available from the SDMX website (<http://www.sdmx.org>).

The SDMX User Guide is accompanied by additional *contributed documents* which provide further in-depth tutorials and guidance on SDMX. These documents have been kept separate from the User Guide in order to keep the length of the User Guide within reasonable limits, and also to avoid extensive rewriting of documents to match the style of the User Guide.

It is expected that further revised releases will follow, taking advantage of new examples drawn from real-world usage of SDMX standards and guidelines, and responding to needs expressed by readers.

Comments on the User Guide are welcomed and can be sent to the SDMX Secretariat (secretariat@sdmx.org), indicating the version of the User Guide to which your comments refer.

Structure of the User Guide

Chapter	Title	Scope of the chapter
A. CORE THREAD		
A.1	What is SDMX?	Provides users and potential users of SDMX with the basic facts about the origin and purpose of SDMX, together with the business case for using SDMX.
A.2	How does SDMX fit with statistical work (the statistical value chain)?	Explains how SDMX fits into the statistical business processes of national and international statistical organisations.
A.3	The SDMX information model: Data Structures	Provides a short introduction to the part of the SDMX information model relating to data structures, it introduces the nomenclature and shows the development of a sample SDMX data structure definition.
A.4	The SDMX Information Model: Metadata Structures	This chapter explains reference metadata and metadata reports. It also provides some information on the steps involved in building metadata structures.
A.5	The SDMX Cross-Domain concepts	Explains the SDMX Cross-Domain Concepts as laid out the Content-Oriented Guidelines and their use for Data and Metadata Structures. Their role in the harmonisation of metadata concepts is also described.
A.6	Publishing data and metadata using SDMX	Explains the scenarios for using SDMX to publish statistics on the Internet
A.7	Uses for an SDMX Registry	Explains what an SDMX Registry is and what it does
B. TUTORIAL THREAD		
B.1	SDMX Technical Overview	Provides an overview of SDMX from a technical perspective, as a basis for the following chapters of the tutorial thread.
B.2	Obtaining and using SDMX tools	Explains how the freely available SDMX IT tools can be used in the implementation of SDMX standards in local IT systems.
B.3	XML-based technologies used by SDMX	Provides some very basic information about XML and technologies using XML, which are commonly used in the implementation of SDMX.
B.4	Differences between SDMX-EDI and SDMX-ML	Discusses the technical and functional differences between the two formats provided: SDMX-EDI and SDMX-ML.

Chapter	Title	Scope of the chapter
B.5	SDMX message types for data	Explains the six standard message types for data and structural metadata.
B.6	SDMX message types for reference metadata	Explains the message types for reference metadata.
B.7	SDMX architectures using the pull mode for data sharing	Explains the SDMX architectures based on the pull mode, and provides guidance on how to implement these architectures.
B.8	Building and operating an SDMX Registry	
B.9	Data Structure Definitions: a tutorial	Provides a basic tutorial on data structure definitions, aiming at the basic level of understanding needed to make sense of the SDMX standards.
B.10	Guidance on setting up Data Structure Definitions	Provides some general principles that may guide the development of Data Structure definitions and also a practical example.
C. FAQ	Frequently Asked Questions	Gives short answers to common questions about SDMX.

Contributors

The writing and editing of the SDMX User Guide has been a collaborative exercise involving a number of people from the SDMX secretariat and other people involved in implementing SDMX at national and international level. In many cases, chapters have been developed from earlier documents, and the original authorship is not always known.

The following members of the SDMX secretariat contributed to release 2009.1 of the User Guide:

John Allen	Eurostat
Christos Androvitsaneas	ECB
Gabriele Becker	BIS
Stuart Feder	BIS
August Götzfried	Eurostat
Shahbaz Khan	IMF
Marco Pellegrino	Eurostat
Francesco Rizzo	Eurostat, Istat
Jolanta Stefanska	IMF
Lars Thygesen	OECD

Laura Vignola (Istat) was the co-author of chapter B.6 *SDMX architectures using the pull mode for data sharing*

Throughout the User Guide, the secretariat made extensive use of material provided by Chris Nelson and Arofan Gregory (Metadata Technology).

Contents

A	CORE THREAD	11
A.1	What is SDMX?	11
A.1.1	Scope of this chapter.....	11
A.1.2	Origin and purpose of SDMX	11
A.1.3	History	11
A.1.4	When can SDMX be used?	12
A.1.5	The main elements of SDMX	12
A.1.6	Tools for SDMX.....	15
A.1.7	Costs and benefits for organisations using SDMX	15
A.1.8	The gradual implementation of SDMX	16
A.1.9	Communication and capacity-building.....	17
A.2	How does SDMX fit with statistical work (the statistical value chain)?	18
A.2.1	Scope of this chapter.....	18
A.2.2	Statistics work in national organisations.....	18
A.2.3	Statistics work in international organisations.....	18
A.2.4	The statistical process.....	19
A.2.5	The statistical process and SDMX	20
A.2.6	Where and how to apply the SDMX framework?	21
A.3	The SDMX information model: Data Structures	24
A.3.1	Scope of this chapter.....	24
A.3.2	What is a data structure definition?	24
A.3.3	Deriving a Data Structure for my data	24
A.3.4	The SDMX information model for data in a nutshell.....	26
A.3.5	Further information	27
A.4	The SDMX Information Model: Metadata Structures	28
A.4.1	Scope of this chapter.....	28
A.4.2	Reference metadata.....	28
A.4.3	Metadata Reports.....	28
A.4.4	Metadata Structure definitions.....	30
A.4.5	Building a Metadata Structure Definition	30
A.5	The SDMX Cross-Domain concepts	31
A.5.1	Scope	31
A.5.2	Cross-Domain Concepts	31
A.5.3	The institutional management and harmonisation of metadata.....	32
A.6	Publishing data and metadata using SDMX	34

A.6.1	Scope of this chapter.....	34
A.6.2	Overview of this chapter.....	34
A.6.3	Scenarios	34
A.6.4	SDMX as a standard format for end-users.....	35
A.6.5	SDMX as a format for web presentation	37
A.6.6	SDMX as the basis for web portals	38
A.6.7	SDMX as the basis for services provision	40
A.6.8	Conclusions.....	42
A.7	Uses for an SDMX Registry.....	43
A.7.1	Scope of this chapter.....	43
A.7.2	Introduction.....	43
A.7.3	Functions of an SDMX Registry	43
A.7.4	Architecture of an SDMX Registry.....	44
A.7.5	Examples of working SDMX Registries.....	46
A.7.6	Joint External Debt Hub	46
A.7.7	FAO CountryStat.....	47
A.7.8	Eurostat SDMX Registry	49
A.7.9	References.....	50
B	TUTORIAL THREAD	51
B.1	SDMX technical overview.....	51
B.1.1	Scope of this chapter.....	51
B.1.2	Technical standards: from Version 1.0 to Version 2.0.....	51
B.1.3	Scope of the SDMX standards.....	52
B.2	Obtaining and using SDMX Tools.....	54
B.2.1	Scope of this chapter.....	54
B.2.2	Introduction.....	54
B.2.3	Types of Tools.....	54
B.2.4	Availability of SDMX Tools	55
B.2.5	Open Source and shared development.....	55
B.3	XML-based technologies used by SDMX.....	56
B.3.1	Scope of this chapter.....	56
B.3.2	Introduction.....	56
B.3.3	Why is XML ideal for the purpose of SDMX?	58
B.3.4	Web services.....	59
B.3.5	XML tools	60
B.4	Differences between SDMX-EDI and SDMX-ML.....	61
B.4.1	Scope of this chapter.....	61

B.4.2	SDMX-EDI.....	61
B.4.3	SDMX-ML.....	61
B.4.4	Comparative table: SDMX-EDI and SDMX-ML	62
B.5	SDMX message types for data.....	63
B.5.1	Scope of this chapter.....	63
B.5.2	The different kinds of standard messages	63
B.5.3	Structure definition message.....	63
B.5.4	Generic Data Message.....	64
B.5.5	Compact Data Message.....	64
B.5.6	Utility Data Message	65
B.5.7	Cross-Sectional Data Message.....	65
B.5.8	Query Message	65
B.5.9	Deriving one SDMX-ML message from another.....	65
B.6	SDMX message types for reference metadata	67
B.6.1	Scope of this chapter.....	67
B.6.2	Message types	67
B.6.3	Structure Message	67
B.6.4	Generic Metadata Message	69
B.6.5	Metadata Report message	69
B.7	SDMX architectures using the pull mode for data sharing	70
B.7.1	Scope of this chapter.....	70
B.7.2	Introduction.....	70
B.7.3	The database-driven architecture.....	70
B.7.4	The data hub architecture	71
B.7.5	Data producer architectures	72
B.7.6	The mapping process.....	73
B.7.7	From a data file to an SDMX data file	78
B.7.8	Disseminating SDMX data files starting from a database	78
B.7.9	An SDMX solution valid for both the database-driven and data hub architectures	79
B.8	Building and operating an SDMX Registry	83
B.8.1	Scope of this chapter.....	83
B.9	Data Structure Definitions: a tutorial.....	84
B.9.1	Scope of this chapter.....	84
B.9.2	What is a data structure definition?	84
B.9.3	Grouping Data.....	85
B.9.4	Attachment Levels.....	86
B.9.5	Keys	86

B.9.6	Attributes	87
B.9.7	Code lists and other representations.....	87
B.9.8	Cross-sectional data structures.....	88
B.10	Guidance on setting up Data Structure Definitions	91
B.10.1	Scope of this chapter.....	91
B.10.2	Choice of Dimensions and Attributes	91
B.10.3	Principles for deciding the order of dimensions in the data structure	91
B.10.4	Code lists.....	92
B.10.5	Change management.....	92
B.10.6	DSDs and data life cycle	92
B.10.7	Organisational issues.....	92
C	FREQUENTLY ASKED QUESTIONS	95
C.1	What are the differences between SDMX Version 1.0 and Version 2.0.....	95
C.2	What is a key family?.....	95
C.3	What is a data structure definition?	95
C.4	Are tools available to help build a data structure definition? To do other things with SDMX?.....	95
C.5	What is the difference between structural and reference metadata?	95
C.6	How and when will organizations implement SDMX?.....	96
C.7	What is an SDMX Registry?	96
C.8	Is there a single central SDMX Registry?	96
C.9	What is the Metadata Common Vocabulary?.....	96
C.10	What are the SDMX Content-Oriented Guidelines?	97
C.11	How can I use existing code lists to help me develop a data structure definition?	97
C.12	If more than one code list exists for similar information, how do I find where they overlap?.....	97
C.13	Is SDMX multilingual?	97

A CORE THREAD

A.1 What is SDMX?

A.1.1 Scope of this chapter

This chapter aims to provide users and potential users of SDMX with the basic facts about the origin and purpose of SDMX, together with the business case for using SDMX. Most aspects of SDMX which are mentioned here are explained in greater detail in the later chapters of the User Guide.

A.1.2 Origin and purpose of SDMX

The Statistical Data and Metadata eXchange (SDMX) initiative was launched in 2001 by seven organisations working on statistics at the international level: the Bank for International Settlements (BIS), the European Central Bank (ECB), Eurostat, the International Monetary Fund (IMF), the Organisation for Economic Co-operation and Development (OECD), the United Nations Statistical Division (UNSD) and the World Bank. These seven organisations act as the sponsors of SDMX.

The stated aim of SDMX was to develop and use more efficient processes for exchange and sharing of statistical data and metadata among international organisations and their member countries. To achieve this goal, SDMX provides standard formats for data and metadata, together with content guidelines and an IT architecture for exchange of data and metadata. Organisations are free to make use of whichever elements of SDMX are most appropriate in a given case.

With the Internet and the world-wide web, the electronic exchange and sharing of data has become easier and more common, but the exchange has often taken place in an *ad hoc* manner using all kinds of formats and non-standard concepts. This creates the need for common standards and guidelines to enable more efficient processes for exchange and sharing of statistical data and metadata. As statistical data exchange takes place continuously, the gains to be realised from adopting common approaches are considerable both for data providers and data users.

SDMX aims to ensure that metadata always come along with the data, making the information immediately understandable and useful. For this reason, the SDMX standards and guidelines deal with both data and metadata.

Common standards and guidelines followed by all players not only help to give easy access to statistical data, wherever these data may be and without demanding prior agreement between two partners, but they also facilitate access to metadata that make the data more comparable, more meaningful and generally more usable.

A.1.3 History

The Version 1.0 SDMX standards include the information model as well as the XML-based data format SDMX-ML and the GESMES/TS data format, renamed SDMX-EDI (see chapters B1, B2 and B9 for further explanations).

The Version 1.0 SDMX standards were approved by the sponsors in September 2004 and accepted as an ISO technical specification (ISO/TS 17369:2005) in April 2005.

In November 2005, the sponsors approved Version 2.0 of the SDMX standards, which are fully compatible with Version 1.0 but in addition provide for the exchange of reference (explanatory) metadata, and include the registry interface specification.

SDMX 2.0 standards are being submitted to ISO in 2008, with some adjustments and corrections to take account of comments received since Version 2.0 was released in 2005.

The first draft of the Content-Oriented Guidelines was released for public review in March 2006, and a consolidated version was released for public review in February 2008. The full release of the Content-Oriented Guidelines, which has been extensively revised to take account of comments received from many organisations, took place in January 2009.

In March 2007, the sponsoring institutions signed a Memorandum of Understanding (MoU), which is intended to set out the arrangements for a durable collaboration by the sponsors on all aspects of SDMX. The MoU explicitly excludes the formation of any legal entity or common budget for SDMX; each sponsoring institution and its member countries will continue to use its existing procedures to agree on arrangements for transmission and publication of statistics.

In the conclusions of the 39th Session of the UN Statistical Commission (New York, February 2008), SDMX was recognised and supported as "the preferred standard for exchange and sharing of data and metadata in the global statistical community"¹. This acceptance of SDMX at UN level is a major step forward towards the broader use of SDMX at world-wide level.

A.1.4 When can SDMX be used?

The SDMX standards are designed for exchange or sharing of statistical information between two or more partners. Evidently, the SDMX standards have been developed by the sponsoring organisations in order to accommodate their constituencies, which include national statistical offices, central banks, ministries and other bodies. Within and across these constituencies, the standards are intended for reporting or sharing statistical data and metadata in the most efficient way.

SDMX standards can also be used within a national system for transmitting or sharing statistical data and metadata. This is particularly interesting in countries with a federal structure or a fairly decentralised statistical system. In such cases, a close link can be established between the national system for data sharing and the international ones, allowing for additional efficiency gains for the involved organisations.

The use of SDMX for data exchange can easily evolve towards open SDMX-based dissemination; such dissemination may respond well to user demands for well-structured data and metadata in reusable formats, and should be considered as an option for national authorities as well as international organisations. It is also an interesting option for private data providers, such as re-sellers of statistical databases.

SDMX can also be used for data and metadata management *within* statistical organisations, since its information model is applicable for much of the information stored and processed within statistical organisations, and such organisations can make use of the SDMX IT tools to reduce the costs of developing their data management systems.

A.1.5 The main elements of SDMX

As mentioned above, SDMX consists of an information model, standard formats, an IT architecture for data exchange, and content-oriented guidelines.

¹ See documents E/2008/24, E/CN.3/2008/34 and E/CN.3/2008/13 which are linked from this page:

<http://unstats.un.org/unsd/statcom/sc2008.htm>

SDMX Information Model: SDMX provides a way of modelling statistical data, metadata and data exchange processes. The data (and related metadata) for a particular statistical domain are structured according to a "Data Structure Definition" (DSD, formerly known as a "key family"). The DSD describes the structure of a particular statistical data flow through a list of dimensions (for example: country, variable/topic, year), a list of "attributes" (for example, unit of measure) and their associated code lists. Attributes are metadata about an individual value, a time series or a group of time series.

SDMX also defines a model for additional explanatory metadata, which are often referred to in SDMX as *reference metadata*. Reference metadata are generally in a textual format, using concepts describing the content, methodology and quality of the data. The reference metadata for a particular statistical data flow, statistical domain or – if used homogeneously throughout a statistical institution - a particular statistical institution are structured according to a "Metadata Structure Definition" (MSD).

Figure A.1.1 and Box 1 below indicate the ideas behind structuring data and metadata and the components that would be used to define a DSD or an MSD, respectively.

Metadata to be used in a Data Structure Definition

Country (Dimension)
Stock/Flow (Dimension)
Unit Multiplier (Attribute)
Unit (Attribute)
External Debt Data, from Creditor Sources (Topic)
Time/Frequency (Dimension) (Dimension)


South Africa, Stocks, in Millions of US Dollars

Topic	1998				1999				2000				2001			
	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
Bank Loans	15837	16541	16280	16030	16669	16457	15346	14731	14063	13672	13635	13960	14696	14059	15228	14538
Debt Securities Issued Abroad	4313	4552	4603	4763	4807	5815	5428	5498	6034	6443	6175	6018	5819	6099	7277	6809
Non-Bank Trade Credits	-	-	-	-	1493	942	-	942	-	1060	-	786	-	654	-	641
Multilateral Claims	308	205	105	38	36	48	72	07	83	145	-	-	-	-	-	-
Official Bilateral Loans (DAC Creditors)	-	-	-	404	-	-	-	-	535	-	-	-	413	-	-	380

Notes:

Reference metadata organized via a Metadata Structure Definition

Dissemination Standards Bulletin Board
Special Data Dissemination Standard
(AS PROVIDED TO THE IMF BY THE RESPECTIVE COUNTRY)



Canada National accounts
(National Economic and Financial Accounts)

Last Posted: [Nov-13-2008](#)
Last Certified: [Oct-10-2008](#)
Last Updated: [Jul-23-2003](#)

Contact Person(s)

General Information Officer,
Income and Expenditure Accounts
Statistics Canada,
R.H. Coats Building
Ottawa, Ontario, Canada K1A 0T6
Phone : 1 613 9513640
Fax : 1 613 9513618
Email : iead-info-dcrd@statcan.ca

[Printer Friendly Page](#)

Reference metadata concepts in a hierarchy

1. Integrity

1.1 Professionalism

1.1.1 Impartiality of statistics

Statistics Canada
Statistics Canada acts independently to establish the methods used in data estimation as well as the timing and content of all data releases. Data release dates are publicly announced and adhered to strictly.

1.2 Transparency

1.2.1 Disclosure of terms and conditions for statistical collection, processing, and dissemination

Statistics Canada
The Statistics Act can be accessed in English and in French on the Justice Canada Internet website (<http://laws.justice.gc.ca/en/3-19/index.html>).

1.2.2 Internal governmental access to statistics prior to release

The information in the press release is provided, under strict embargo, to a small number of staff of the Department of Finance, the Office of the Prime Minister, and the Bank of Canada for preparation of ministerial briefings on the afternoon before official release.

Topic code list

Code	Value
A	External Debt, All Maturities, Bank Loans
B	External Debt, All Maturities, Debt Securities Issued Abroad
C	External Debt, All Maturities, Brady Bonds
D	External Debt, All Maturities, Non-Bank Trade Credits
E	External Debt, All Maturities, Multilateral Claims
F	External Debt, All Maturities, Off. Bilateral Loans (DAC Creditors)
G	Debt due within 1 year, Liabilities To Banks

Figure A.1.1: Metadata in SDMX

BOX 1

Metadata in SDMX

Structural metadata are those metadata acting as identifiers and descriptors of the data, such as names of variables or dimensions of statistical cubes. Data must be associated to some structural metadata, otherwise it becomes impossible to properly identify, retrieve and browse the data.

Reference metadata are metadata that describe the contents and the quality of the statistical data (conceptual metadata, describing the concepts used and their practical implementation, methodological metadata, describing methods used for the generation of the data, and quality metadata, describing the different quality dimensions of the resulting statistics, e.g. timeliness, accuracy). While these reference metadata exist and may be exchanged independently of the data (and its underlying structural metadata), they are often linked (“referenced”) to the data.

The idea is that it should be possible, using the SDMX standards, to exchange or share the data and the metadata that will allow a thorough understanding and interpretation of the corresponding statistical data.

The *data exchange process* is represented in SDMX via the definition of “data flows”, “data providers”, and, in particular “provision agreements”. The latter describes the way in which data and metadata are provided by a data provider. Thus, a data provider can express the fact that it provides a particular data flow covering a specific set of countries and topics, with a particular publication schedule.

Content-Oriented Guidelines: these are recommendations for categorising and describing data. The guidelines consist of

- the *Cross-Domain Concepts*, which provide common descriptors for concepts used in DSDs and MSDs for different statistical domains;
- the Cross-Domain Code Lists, which provide a collection of code lists that are used for different statistical domains;
- the *Statistical Subject-Matter Domains* which provides a list of statistical domains based on the UNECE Classification of International Statistical Activities;
- the *Metadata Common Vocabulary (MCV)* of terms used for describing statistics and their compilation processes (across subject-matter domains) by national statistical authorities and international organisations.

The Content-Oriented Guidelines are intended to be generic and not influenced by the specificities of any domain or organisation.

Standard formats for the exchange of data and metadata: Based on the common information model, the SDMX standards include data exchange formats based on XML (SDMX-ML) and EDIFACT (SDMX-EDI, which is identical to GESMES/TS and which has been widely used since the 1990s).

The use of SDMX-EDI is fully SDMX-compliant, since it follows the SDMX 1.0 Information Model. The advantage of the XML syntax is that it is a widely-used open standard, which can be processed with a wide range of IT applications, including free and/or open-source software. The EDIFACT syntax is more specialised (e.g. appropriate for representing large databases, due to its compact format) and is usually processed with proprietary applications. The use of SDMX-ML, which supports the full Information model of SDMX 2.0, provides additional benefits. In the SDMX User Guide, unless there is a specific reference to SDMX-EDI, the use of SDMX-ML is assumed.

Architecture for data exchange: SDMX supports two complementary modes for data exchange and data sharing (see Box 2): the “push” mode (where data are transmitted from one organisation to another) and the “pull” mode (where one organisation retrieves data from another organisation’s server). SDMX also supports the “hub” concept, where users obtain data from a central hub which itself automatically assembles the required dataset by querying other data sources.

The SDMX IT architecture also comprises SDMX registries, implementing the general idea of a *metadata registry* for use with SDMX standards. The idea of a metadata registry is essentially that when a business wants to start a relationship with another business, it queries a registry in order to locate a suitable partner and to find information about requirements for dealing with that partner. SDMX has developed specific registry standards in order to enable statistical organisations to perform efficient data and metadata sharing. In general terms, an SDMX registry is essentially an application which can accept SDMX query messages and return the locations of SDMX-compliant information, which may include data as well as structural and reference metadata. More information on SDMX Registries is provided in chapters A.9 and B.6.

BOX 2

Push and pull

Messages can be exchanged in two different modes, the push mode and the pull mode:

Push mode means that the data provider takes action to send the data to the organisation collecting the data. This can take place using different means, such as e-mail or file transfer. These are the traditional modes of data collection, carried out by international organisations for many years.

Pull mode implies that the data provider makes the data available via Internet technology. The data may be made available for download in an SDMX-conformant file, or they may be retrieved from a database in response to an SDMX-conformant query, via a web service running on the provider’s server. In both cases, the data are made available to any organisation requiring them, in formats which ensure that data are consistently described by appropriate metadata, whose meaning is common to all parties in the exchange.

While all combinations of the modes above are supported by SDMX standards, it is the aim of the SDMX initiative to further promote data sharing exchange using the pull mode.

A.1.6 Tools for SDMX

To support the use of SDMX, many IT tools have been developed by the SDMX sponsoring organisations or by other bodies. These tools can generally be freely downloaded via the SDMX website. The source code is available so that they can be used as components for building IT systems in statistical organisations. Further information is provided in chapter B7.

A.1.7 Costs and benefits for organisations using SDMX

The benefits of standardised data file formats and metadata include:

- automated production and processing of data files;

- enabling consistent use of concepts and code lists.

SDMX adds to the known advantages of standardisation of transmission formats the additional advantages of using XML, as well as improved consistency between data structures used in different domains and for different purposes. It can facilitate data sharing, and will support the production and exchange of reference metadata suitable for all classes of users.

Many national data providers are already using, or are planning to use, XML as the basis for their data management and dissemination systems. This means that SDMX-ML can be generated and manipulated using the NSI's normal IT tools with often only small adjustments. In addition, many IT applications are available to work with XML-based data, and expertise for working with XML is readily available and will often be available in-house.

The SDMX Content-Oriented Guidelines will improve the quality of data and metadata structures developed in future, by promoting the use of consistent statistical concepts and code lists across domains and between organisations. This will make it easier for NSIs to generate data files automatically from their own data warehouses.

DSDs whose use is agreed across different organisations will also reduce the workload of NSIs, by facilitating "data sharing" arrangements.

The application of SDMX standards and Content-Oriented Guidelines can facilitate the routine transmission and dissemination of structured reference metadata. This will help NSIs, who will be able to automate the generation of reference metadata files using standardised MSDs which will be meaningful not only to international organisations, but also to users accessing NSI data directly.

There will be modest IT costs to implement SDMX, particularly where using data sharing using the pull method. However, in many cases the costs will be absorbed within the cost of setting up new data warehouses or maintaining existing IT systems.

A.1.8 The gradual implementation of SDMX

To a large extent, SDMX will come into use gradually, making use of the existing cooperation arrangements between statistical organisations active at international level. In most statistical domains, there exist intersecretariat working groups or other groups which have successfully promoted joint questionnaires, standard classifications, standard reference metadata and other tools.

These groups are in several cases already considering how to make use of SDMX standards to make their data collections more efficient and to reduce the response burden for countries. As these international arrangements on data collection are usually voluntary, changes to such arrangements can only take place by consensus, and subject to the normal processes of consultation with countries.

Any organisation or group of organisations could set up its own DSDs and MSDs. Such local DSDs and MSDs could either conform directly to the Content-Oriented Guidelines, or they could be linked to the Content-Oriented Guidelines, for example through correspondence tables mapping different concepts and code lists to one another. Over time, the SDMX sponsoring organisations would review the conformity of local DSDs and MSDs to the Content-Oriented Guidelines and some DSDs and MSDs would be offered for wider use.

As more DSDs and MSDs become available, users will be able to retrieve them through a network of federated SDMX registries.

A.1.9 Communication and capacity-building

Communication and capacity-building activities on SDMX are a fundamental part of the SDMX initiative. They have been organised in a decentralised manner by all the SDMX sponsoring organisations. Some of the most significant actions are:

- The SDMX website (<http://www.sdmx.org>) which provides a single point of entry for all information on SDMX, ranging from the documentation on standards and guidelines to the downloadable software, together with announcements, events and information on implementation activities and DSDs. The SDMX User Guide and other tutorials are available via the website;
- The SDMX Global Conferences, which have been held in Washington (January 2007) and Paris (January 2009);
- Training courses which have been organised or supported by the sponsoring organisations.

A.2 How does SDMX fit with statistical work (the statistical value chain)?

A.2.1 Scope of this chapter

This chapter explains how SDMX fits into the statistical business processes of national and international statistical organisations.

A.2.2 Statistics work in national organisations

Most countries have one or more national statistical organisations (NSIs), carrying the responsibility of maintaining a national statistical system. Core tasks of NSIs are to collect, process and organise statistical data, and subsequently put them at the disposal of various communities of users, often termed as dissemination of the statistics. Obviously, some of the main obligations of NSIs are to make the necessary strategy decisions on what should be measured and how, and to manage and document the statistical system.

A widespread problem is lack of harmonisation across different fields of statistics in a country, even within the same national organisation. This is often related to the statistics production being organised in so-called stove-pipes, or independent production lines. This makes it difficult to use statistics for different subjects in a coherent way, thus impairing the quality of statistics as seen from the user perspective. It also reduces efficiency in the production process.

To overcome these problems, there has been a strong tendency in NSIs towards standardisation and integration, breaking down stove-pipes. This leads to the creation of corporate statistical data warehouses, bringing together statistics on different subjects under one system. In this endeavour, the creation of statistical metadata plays an important part. The changes required towards such integrated systems are not only technical, but also organisational.

A.2.3 Statistics work in international organisations

The statistical work in international organisations resembles the work in national statistical organisations. It basically consists of the same work phases, but an important characteristic is that data collection most frequently has the national organisations as respondents. And normally the data collected is aggregate data on a national or regional level, as distinct from micro data on individual persons, households or companies.

Stove-pipe organisation is as frequent in international organisations as in national ones. Therefore the organisation of data collection from countries has been little standardised: Different organisations have been using different media and formats for data collection, and the same applies within each international organisation.

A.2.4 The statistical process

In order to explain the use of SDMX standards and guidelines in statistical work, we need to present a simple model of business processes in a statistical organisation² (see Figure A.2.1).

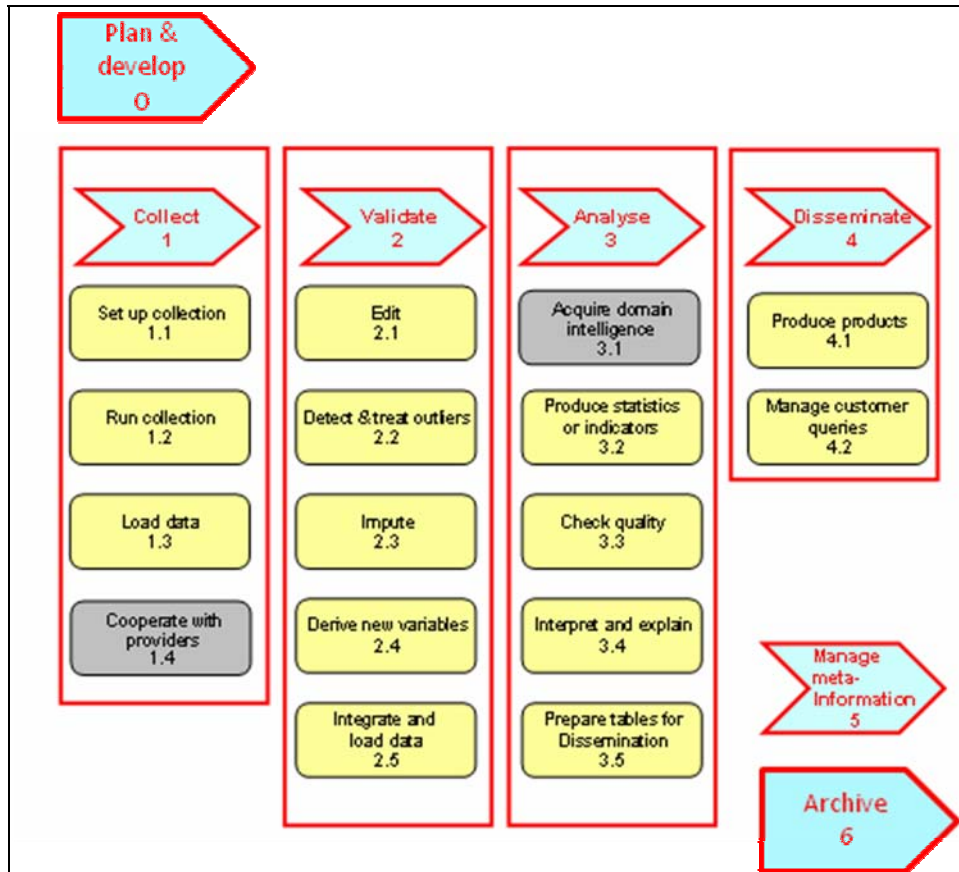


Figure A.2.1: A simple model of business processes in a statistical organisation

For reasons of simplicity and overview, we will mainly stick to the first level of the business processes in this chapter:

0. Plan and develop the statistical task
1. Collect
2. Validate
3. Analyse
4. Disseminate
5. Manage meta information
6. Archive

² There are several such models in the literature. The model presented here represents a kind of common understanding of the statistical processes among the international community of official statistics. This work is ongoing and the issue is open to further inputs from various groups.

A.2.5 The statistical process and SDMX

The SDMX framework consists of the following components:

- Information model for data and metadata (SDMX IM)
- Content-oriented Guidelines covering the SDMX Cross-Domain Concepts and the Metadata Common Vocabulary
- Two types of data exchange formats to exchange data and metadata (SDMX-EDI and SDMX-ML)
- SDMX Registry standards and registry interfaces
- SDMX tools

When *planning and developing the statistical task* the SDMX Information Model forces us to think about the organisation of the data what we want to collect. It gives us a framework for this organisation that can be applied consistently across statistical tasks, thus leading to economies of scale. The Content-oriented Guidelines help us to choose, for example, statistical concepts and code lists that have been used by others, thus potentially increasing the interoperability and statistical harmonisation. (There is no need to reinvent the wheel.) Thinking about data organisation and data (and metadata) structures will also have a learning effect across the statistical organisation.

The *collection and validation* steps build on the data structures developed during the previous step of the statistical process. The SDMX data formats now become important for the actual data and metadata exchange related to the data collection tasks. The use of standards formats facilitates automation of the exchange process and also the technical validation of the exchanged information. SDMX registries can further enhance the data exchange process, e.g. if statistical information has to be provided to more than one recipient.

The SDMX information model brings together data and metadata into a consistent and integrated model and thus supports moving towards metadata driven processing: generic processing (and storage) systems based on the SDMX IM are being developed. They will help to lower the marginal cost for new data collection activities. In addition they have "built-in" features for offering metadata driven navigation and search facilities for statistical analysts and users and thus support the *analysis* and *dissemination* steps. The standard data exchange formats facilitate the dissemination of statistical information to a variety of clients. In a way, dissemination from, for example, a national statistical organisation may at the same time be the reporting to an international statistical body, using the SDMX "pull" mode. The *manage meta information* and *archiving* steps of the statistical process are related to and rely on a consistent treatment of metadata, so they also benefit from the application of the SDMX framework.

In general we can derive the following benefits from using the SDMX framework in the statistical process:

- Speed up movement of data through the statistical process
- Reduced "time to users" for data and metadata
- Increased level of automation, reducing the risk for errors
- Enhanced support for metadata for a better understanding of the data

It is important to note that the implementation of the SDMX framework should not be considered as a "pure IT task". As shown above the key ingredient is the SDMX Information Model and its application and this is mainly a task for statistical experts. They need to take responsibility for modelling data and metadata structures for their

statistical domains and processes before any IT implementation of data exchange formats and processes can even start.

A.2.6 Where and how to apply the SDMX framework?

As shown in Table A.2.1, SDMX can and should play a role in each of these basic processes. However, it can be argued that in most organisations, SDMX has its most important place in the processes *1. Collect*, and *4. Disseminate*, i.e. the processes where data and metadata have to be communicated with the outside world. Here there is considerable scope for standardising, and the result would be that, for instance, the data flowing from national organisations to international ones can be generated in a standardised, orderly and efficient way across different topics and different international organisations, and in particular metadata can be attached and exchanged to a much higher degree than has been the case so far; also international organisations receiving the data can receive them in a standardised way across topics and countries and integrate them into their systems. In these processes, the Content-oriented guidelines play a particularly important role in aligning the concepts used among receivers and providers of data and metadata.

It flows from the use of the SDMX standards in the phases *Collect* and *Disseminate* that they will also be used in *Plan and develop the statistical task*. Here they have the additional strength of helping to identify possible existing sources, which may be registered in SDMX registries.

In order to use the SDMX standards for collection and dissemination, organisations will build gateways into and out of their databases using SDMX. These gateways can also be used for the processes *Validate* and *Analyse*.

The *Archiving* function for statistics has long been in need for a solid, well-documented standard format that is also rich enough to preserve attached metadata and allow for them to be readable and understandable after hundreds of years. The SDMX standards, being official ISO standards and having rich metadata capability, are ideal for this.

Table A.2.1: SDMX components and the phases of the statistical life cycle

Statistical task (business process)	SDMX standard or guideline to help	SDMX tool to use	Key issue
0. Plan and develop the statistical task	SDMX-ML	SDMX Registry	Find possible sources of data that could be used
	SDMX-IM	Data Structure Definition (DSD) and Metadata Structure definition (MSD)	Organise the desired data and metadata as multi-dimensional cubes with attached metadata; agree with respondents on modes of collection
	Content-Oriented Guidelines	SDMX “ Rosetta ” mapping tables linking the concept schemes of different participants in the sharing of data	Decide on concepts, align with internationally agreed concepts; map different concept schemes

Statistical task (business process)	SDMX standard or guideline to help	SDMX tool to use	Key issue
1. Collect	SDMX-ML	SDMX Registry SDMX-ML generic, compact or cross-sectional messages; metadata messages	Collect data and metadata using pull or push technology
	Content-Oriented Guidelines	Cross-domain concepts MCV SDMX "Rosetta" mapping tables linking the concept schemes of different participants in the sharing of data	Use internationally agreed concepts, terms and code lists to ensure mutual understanding and alignment; map different concept schemes
2. Validate	SDMX-IM SDMX-ML	DSD and MSD	Ensure that data and metadata have correct structure and content
3. Analyse	SDMX-IM SDMX-ML	SDMX-ML web services SDMX-ML generic, compact or cross-domain messages; metadata messages	Use SDMX-ML as the generic interface for analysis tools; use the SDMX-IM for a generic processing system to integrate data from many sources (if relevant); pull data and metadata from data warehouse using web services;
4. Disseminate	SDMX-IM SDMX-ML	SDMX-ML Registry SDMX-ML web services SDMX-ML data and metadata messages Query messages	Set up registry; design web services to allow users to query and access data and metadata; offer navigation and search facilities based on SDMX-IM
	Content-Oriented Guidelines	Cross-domain concepts MCV SDMX "Rosetta" mapping tables linking the concept schemes of different participants in the sharing of data	Use internationally agreed concepts, terms and code lists to ensure mutual understanding and alignment; map different concept schemes

Statistical task (business process)	SDMX standard or guideline to help	SDMX tool to use	Key issue
5. Manage metainformation	SDMX-IM SDMX-ML	Metadata Structure definition (MSD) SDMX-ML metadata messages	Receive metadata from external partners; provide metadata to partners; share metadata within the organisation; manage the appropriate attachment levels of metadata
	Content-Oriented Guidelines	SDMX "Rosetta" mapping tables linking the concept schemes of different participants in the sharing of data	Set up metadata management system linked to data, using SDMX attachment levels
6. Archive	SDMX-IM SDMX-ML	SDMX-ML DSD and MSD SDMX-ML data and metadata messages	Archive data in a standardised way to allow for reuse centuries later

It should be added, that as SDMX-ML increasingly becomes the dissemination standard for statistical data and metadata, organisations outside the sphere of "official statistics" can also benefit from using this standard, especially if they are consuming such data in their systems, analysing or transforming them and finally re-distributing them in some form.

A.3 The SDMX information model: Data Structures

A.3.1 Scope of this chapter

This chapter of the User Guide provides a short introduction to the part of the SDMX information model relating to data structures. It introduces the nomenclature and shows the development of a sample SDMX data structure definition.

A.3.2 What is a data structure definition?

In order to answer this question, we need to look at statistical data. Statistical data is represented with numbers, such as:

17369

If you are presented with a number - as above - you will have no idea of what it actually represents. You know that it is a piece of statistical data, and therefore is a measurement of some phenomenon - also known as an "observation" - but you can't tell from the number alone what it is a measurement of. A number of questions come immediately to mind:

- What is the subject of the measurement?
- What units does it measure in?
- What country or geographical region, if any, does it apply to?
- When was the measurement made?

The list of questions is potentially endless. Behind each of these questions is a particular idea, or "concept", which is used to describe the data. In our questions above, these descriptor concepts are Subject, Unit of measure, Country, and Time. If I tell you the answers to these questions, the data will begin to make sense:

- the Subject is "total population"
- the Unit of measure is "thousands of people"
- the Country is "Country ABC"
- the Time is "1 January 2001"

This is a simplified and fictional example, but it does demonstrate how we can begin to make sense of statistical data with a set of descriptor concepts. We now know that our number represents the fact that the total population of Country ABC on 1 January, 2001, was 17,369,000.

The simplest explanation of a data structure definition is that it is a set of descriptor concepts, associated with a set of data, which allow us to understand what that data means. There is more to it, however.

A.3.3 Deriving a Data Structure for my data

Let us assume that you have so far not thought about your data along the lines of the previous paragraphs. However, you have a set of data that you produce or otherwise work with. These can be presented in the form of tables and we can use these to derive an SDMX data structure for our data. Let's assume that you present your data in a set of tables, such as the one below. You may also have a second set of tables showing seasonally adjusted data. And you may have annual and monthly data for several years.

Exports and imports of United Kingdom per quarter (not seasonally adjusted, in millions of Euros)				
	Exports to / imports from Germany		Exports to / imports from United States	
	Exports	Imports	Exports	Imports
	Q:GB:N:2:100:DE	Q:GB:N:3:100:DE	Q:GB:N:2:100:US	Q:GB:N:3:100:US
2008q1	12,450	13,660	14,328	15,156
2008q2	12,480	13,670	14,330	15,150
2008q3	12,520	13,710	14,324	15,163

From reviewing the tables, we can derive the following *statistical (= descriptor) concepts* and their possible values, providing also sample code values:

1. Frequency (M = monthly, Q=quarterly, A = annual [for the “full year” numbers])
2. Reporting Country (GB = United Kingdom, DE = Germany, US = United States)
3. Adjustment (N = not seasonally adjusted, S = seasonally adjusted)
4. Direction of the flow or type of the transaction (2=credit/receipts/goods’ exports, 3=debit/payments/goods’ imports, 4=net).
5. Topic/variable (100= goods, 200=services, 300=income)
6. Vis-à-vis Country(GB = United Kingdom, DE = Germany, US = United States)
7. Unit of measure (EUR = Euros)
8. Unit multiplier (6 = millions)

We note that all time series in our overall data set are expressed in millions of Euros. Unit of measure and unit multiplier do not contribute to the identification of our series, but provide additional information on them. Hence they will be treated as *attributes*, attached at the data set level.

The concepts 1 to 6 above are required to *identify* our time series so we will use them as *dimensions* and will build our series key from them. Assuming that we use the order given above, a possible series key would look like this, taking the colon “:” to separate the dimension values:

Q:GB:N:2:100:DE

This would identify the quarterly (=Q), not seasonally adjusted (=N) exports (2=credit/exports) of goods (=100) from country United Kingdom (= GB) (the reporting country or area) to country Germany (= DE) (vis-à-vis area).

We may also wish to indicate the “compiling agency” for the data and as that may vary between the series, we decide to attach this attribute at the “sibling group level” (so, monthly/quarterly/annual series for the same topic would be provided by the same agency; but series for different topics may be provided by one, two or more different agencies).

A more formal definition of the data structure is provided in Table A.3.1. It should be noted that in this particular example, there are only two domain specific concepts used (balance of payments topic, direction of the flow/type of the transaction). Most of the other concepts are actually SDMX Cross Domain Concepts, taken from the SDMX Content Oriented Guidelines.

Table A.3.1: Sample Data structure definition: my balance of payments data

Position in key	Dimension/attribute name	Identifier	Presentation ³	Attachment level	Code list
1	Frequency	FREQ	A1		CL_FREQ
2	Reporting/reference area	REF_AREA	A2		CL_AREA
3	Adjustment	ADJUSTMENT	A1		CL_ADJUSTMENT
4	Data type for balance of payments statistics	DATA_TYPE_BOP	A1		CL_DATA_TYPE_BOP
5	Balance of payments topic	BOP_ITEM	A3		CL_BOP_ITEM
6	Vis-à-vis area	COUNT_AREA	A2		CL_AREA
	Unit of measure	UNIT_MEASURE	AN3	Data set	CL_UNIT_MEASURE
	Unit multiplier	UNIT_MULT	N1	Data set	CL_UNIT_MULT
	Compiling agency	CL_COMPILING_ORG	AN3	Sibling Group	CL_ORGANISATION

A.3.4 The SDMX information model for data in a nutshell

The SDMX standards are based on the SDMX information model (SDMX-IM) which represents statistical data and metadata.

The list below describes the minimal knowledge needed about the SDMX information model [\[4\]](#) so that we can start developing data structure definitions based on the SDMX standard:

Descriptor concepts: In order to make sense of some statistical data, we need to know the concepts associated to it (for example, the figure 1.2953 alone is pretty meaningless, but if we know that this is an exchange rate for the US dollar against the euro on the 23 November 2006, it starts to make more sense).

Packaging structure: Statistical data can be grouped together. The following levels are defined: the *observation level* (the measurement of some phenomenon), the *series level* (the measurement over time of some phenomenon, usually following a regular interval), the *group level* (group of series. A well-known example is the sibling group which contains a set of series which are identical except that they are measured with different frequencies) and the *data set or data flow level* (made up of several groups, for instance to cover a specific statistical domain). The descriptor concepts mentioned in point 1 can be attached at various levels in this hierarchy.

Dimensions and attributes: There are two types of descriptor concepts: the ones which both identify and describe the data are called *dimensions*, and those which are purely descriptive are called *attributes*.

Keys: Dimensions are grouped into *keys*, which allow the identification of a particular set of data (for example, a series). The key values are attached at the series level, and are given in a fixed sequence. By convention, frequency is the first descriptor concept, and the other concepts are assigned an order for that particular data set. Partial keys can be attached to groups.

Code lists: Each possible value for a dimension is defined in a *code list*. Each value on that list is given a language-independent abbreviation (a *code*) and a language-

³ Presentation: AN1 stands for: Alpha-numeric, exactly 1 position

specific description. Attributes are sometimes represented with codes, but sometimes represented by free-text values. This is fine as the purpose of an attribute is solely to describe and not to identify the data.

Data Structure Definition: A *Data Structure Definition* (key family) specifies a set of *concepts* which describe and identify a set of data. It tells which concepts are *dimensions* (identification and description), and which are *attributes* (just description), and it gives the *attachment level* for each of these concepts, based on the packaging structure (*Data Set, Group, Series, Observation*) as well as their status (mandatory versus conditional). It also specifies which *code lists* provide possible values for the dimensions, as well as the possible values for the attributes, either as code lists or free text fields

A.3.5 Further information

More detailed information on the SDMX information model is available in the following documents in the SDMX Version 2.0 standards package:

SDMX Implementors Guide: especially section 3.2

SDMX Information Model: UML conceptual design

A.4 The SDMX Information Model: Metadata Structures

A.4.1 Scope of this chapter

This chapter explains reference metadata and metadata reports. It also provides some information on the steps involved in building metadata structures.

A.4.2 Reference metadata

Reference metadata are any metadata which are reported not as an integral part of a data set but independent from the statistical data. Another important point is that very often these metadata are associated not with specific observations or series of data, but with entire collections of data or even with the institutions providing the data. Each statistical organisation disposes of a metadata system, and reference metadata are in general treated as a basic component of that system.

From a content point of view, reference metadata can be broken down into:

- conceptual metadata, describing the concepts used and their practical implementation,
- methodological metadata, describing methods used for the generation of the data,
- quality metadata, describing the different quality dimensions of the statistical data.

A.4.3 Metadata Reports

Reference metadata may be organised in metadata Reports, which provide a structured (maybe even hierarchical) presentation of specific reference metadata items, such as "Contact", "Metadata update" or "Classification System", which are used in the sample metadata report below. The example presented in Table A.4.1 refers to the national data transmitted from Bulgaria to Eurostat for the short-term indicator domain "Industrial Production Index".

Table A.4.1: Example of partial metadata report

SDMX Reference Metadata Report (partial)	
Country: Bulgaria	
Domain name: Industrial Production Index	
1. Contact	
1.1 Contact organisation	National Statistical Institute
1.2 Contact organisation unit	Industrial Statistics Division, Department of Business Statistics
1.3 Contact name	Mrs. Guergana Maeva
1.4 Contact person function	State expert

1.5 Contact mail address	2, P. Volov Street, Sofia, Bulgaria 1038
1.6 Contact email address	
1.7 Contact phone number	
1.8 Contact fax number	

2. Metadata update	
2.1 Metadata last certified	2008-04-04
2.2 Metadata last posted	2008-04-04
2.3 Metadata last update	2008-04-04

3. Statistical presentation	
3.1 Data description	
The index measures the monthly change in the value of production, covering mining, manufacturing, electricity, water and gas supply.	
3.2 Classification system	
Classification of Economic Activities (NACE.BG), which complies with NACE Rev.1.1, is used for the classification of statistical units to the 4-digit level.	
3.3 Sector coverage	
The Industrial production index covers the mining and manufacturing industries, the production and distribution of electricity and steam, and natural gas and water supply. All resident market enterprises are in scope for the Production index.	
3.4 Concepts and definitions (main variables)	
The index aims to follow the monthly change in the value of industrial production. The current concepts and framework are those set forth by the EU Regulation 1165/98, concerning short-term statistics amended by Regulation (EC)1158/2005 that established data requirements in relation to coverage, periodicity, and timeliness, as well as the new 2005 Eurostat manual <i>Methodology of Short-term Business Statistics (interpretations and guidelines)</i> . The index measures the monthly change in the value of production, covering mining, manufacturing, electricity, water and gas supply.	
3.5 Statistical unit	
Statistical units are classified according to their principal activity at the class level (4-digit) in accordance to the National Classification of Economic Activities (NACE.BG), which complies with NACE rev.1.1.	
3.6 Statistical population	
There are about 30 000 enterprises in the population and approximately 3 340 units are sampled each month. Enterprises are stratified first by group (3-th digit of NACE.BG) and afterwards in each group they are stratified according the number of persons employed. Enterprises having more than 100 persons employed are surveyed exhaustively. Enterprises with more than 10 employed, but less than 99 are randomly sampled. In this way 97% of the turnover of the industrial enterprises is covered with the survey. Enterprises with less than 9 persons employed are not surveyed monthly.	
3.7 Reference area	
Industrial production index survey covers entire country.	
3.8 Time coverage	
3.9 Base period	
2000 = 100	

A reference metadata report is a set of information regarding almost any object within the formal SDMX view of statistical data and metadata exchange: they may describe the schedule on which data is released; they may describe the flow of a single type of data over time; they may describe the quality of data, etc. With SDMX, the creators of reference metadata may take whatever concepts they are concerned with, or obliged to report, and provide a reference metadata report containing that information.

A.4.4 Metadata Structure definitions

A reference metadata report has a metadata structure definition which describes how it is organized. This is similar to a data structure definition describing how a data set is organised.

A *Metadata Structure Definition (MSD)* identifies what metadata concepts are being reported, how these concepts relate to each other (typically as hierarchies), what their presentational structure is, i.e. how they may be represented (as free text, as coded values, etc.), and to which formal object they are attached.

Reference metadata may be attached to different data objects (for instance to a data set, a time series, or an observation). However, this kind of metadata is usually attached at a high level (data set, data flow or even at agency level). In the above example, the metadata report is attached to the combination of "Reference Area" (= Bulgaria) and the "Statistical Domain" (= Industrial Production Index). There is a certain hierarchical structure relating to the various metadata items describing the "contact", "Metadata update" and "Statistical Presentation". Note that all items listed in the report are part of the SDMX Cross-Domain Concepts, which are explained in the next chapter, or can be mapped to them.

More than one *metadata report* may be attached to the objects identified in a MSD. For instance, a release calendar can be structured and posted separately from the main report on reference metadata, which includes all the usual elements of data content and data quality, together with a link to the external release calendar.

A.4.5 Building a Metadata Structure Definition

The tasks that need to be undertaken in defining the metadata structure definition (MSD) are:

1. Analysis of the metadata in order to identify and document the "concepts" for which metadata are to be reported or disseminated.
2. Determine the structure of the "Metadata Report" in terms of the concepts used, the hierarchy of the concepts when used in the report, and their "representation" (for example: is a code list used, or is the format free text?).
3. Specify the "object type" to which the metadata are to be attached, and how this object type is identified: knowledge of the SDMX Information model is needed here, as the metadata can only be attached to objects that can be identified in terms of the object types that exist in the information model, e.g. an organisation, a data flow, a statistical domain, a code list or a provision agreement.

It does not really matter in which order these tasks are performed. The starting point can be the concept list (*concept scheme, task 1*) or the *metadata report (task 2)* from which the concept scheme (task 1) can be derived, and then the object type where the metadata are to be attached can be identified (task 3).

A.5 The SDMX Cross-Domain concepts

A.5.1 Scope

The chapter explains the SDMX Cross-Domain Concepts as **laid out** in the Content-Oriented Guidelines, and their use for Data and Metadata Structures. Their role in the harmonisation of metadata concepts is also described. The use of SDMX cross-domain concepts ensures a common understanding of the contents of a statistical concept within and across statistical organisations.

A.5.2 Cross-Domain Concepts

The SDMX Content-oriented Guidelines (COG) contain specific guidance (mostly cross-domain oriented) for statistical organisations for setting up SDMX compliant data and metadata structure definitions: the SDMX Cross-domain Concepts. These presently contain 135 statistical concepts or sub-concepts. SDMX Cross-domain Concepts mainly used as structural metadata in Data Structure Definitions comprise, for example, frequency, reference area, possibly currency etc. They also contain statistical concepts which are in general used for the measurement or declaration of data quality. Examples are "accuracy", "comparability" or "timeliness". These quality-related concepts are normally implemented by statistical organisations within their respective Metadata Structure Definitions. This enables the applying statistical organisation to measure and analyse data quality in a structured and harmonised manner across the whole statistical production of this organisation.

The Guidelines provide for each SDMX Cross-domain concept an ID and a detailed description, which may be complemented by additional comments. The table below presents the two Cross-Domain concepts "Accuracy" and "Currency".

Concept ID:	ACCURACY
Description:	Closeness of computations or estimates to the exact or true values that the statistics were intended to measure.
Context:	The accuracy of statistical information is the degree to which the information correctly describes the phenomena. It is usually characterized in terms of error in statistical estimates and is often decomposed into bias (systematic error) and variance (random error) components. Accuracy can contain either measures of accuracy (numerical results of the methods for assessing the accuracy of data) or qualitative assessment indicators. It may also be described in terms of the major sources of error that potentially cause inaccuracy (e.g., coverage, sampling, non response, response error). Accuracy is associated with the "reliability" of the data, which is defined as the closeness of the initial estimated value to the subsequent estimated value. This concept can be broken down into: Accuracy - overall (summary assessment); Accuracy - non-sampling error; Accuracy - sampling error.
Presentation:	Free text
Concept ID:	CURRENCY
Description:	Monetary denomination of the object being measured.
Presentation:	CL_CURRENCY

Figure A.5.1 provides a simplified view of how the cross-domain concepts are used for defining data/metadata structure definitions in the SDMX framework. In the data/metadata structures they are usually combined with domain specific concepts.

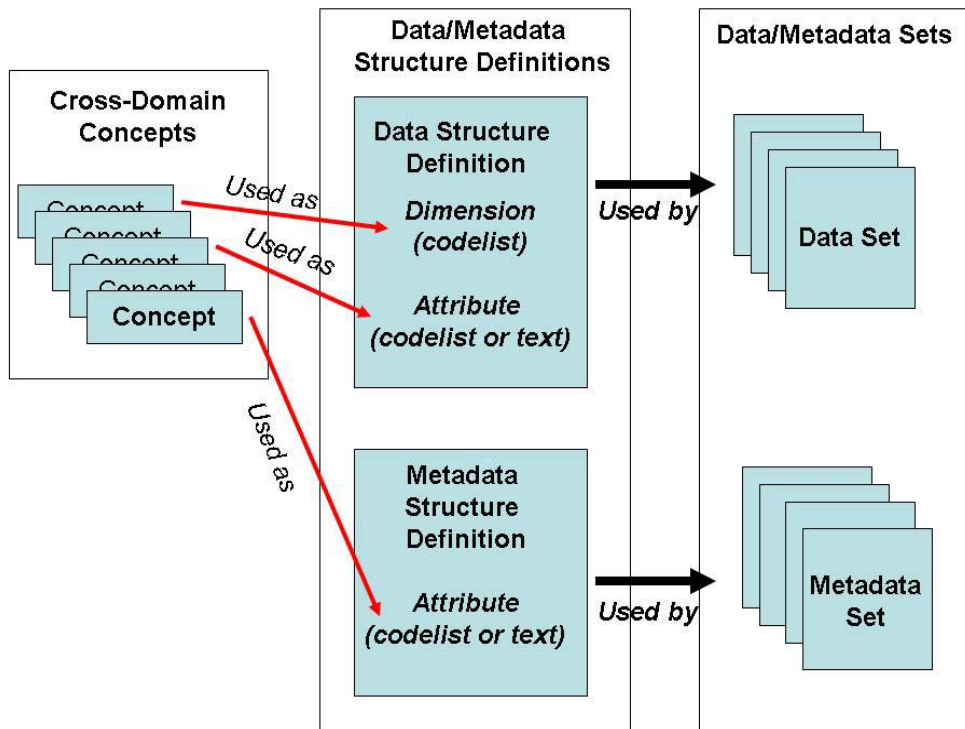


Figure A.5.1: Use of concepts defining data/metadata structure definitions in the SDMX framework

The illustration shows that cross-domain concepts can have two different roles:

- As structural metadata: as dimensions in a data structure definition, to identify each statistical observation. For example, a dimension named “Reference Area” would explain which country or geopolitical aggregate a specific statistical observation refers to.
- As Reference metadata: Attributes in a data structure definition or in a metadata structure definition. Attributes provide information about the data, thus qualifying the data further (example, “unit of measure”) or can be used to report metadata, for example with concepts such as timeliness, reference period, classification system and data compilation. The values of these concepts may be coded, but are more often free text.

In case a concept can be represented as coded, there must be a link to the code list containing valid values that may be reported. If the concept is used as an attribute, the attachment level must be indicated. This means an indication of the data object or structure, e.g. “time series” or “observation”, to which the concept is linked.

A.5.3 The institutional management and harmonisation of metadata

Statistical organisations use different organisational set-ups for the production and management of metadata. Often one central unit in a statistical organisation is responsible for the creation and maintenance of harmonised metadata, co-operating closely with the units being responsible for statistical production in subject matter domains. The harmonised metadata centrally maintained should then be used all across the statistical life cycle, i.e. in data collection, data production and in data dissemination.

SDMX also deals with the harmonisation of metadata at international level. As part of the SDMX Content-oriented Guidelines, a limited number of Cross-domain code lists are published (for example, for Frequency and Confidentiality Status) which show the recommended code values and the textual descriptions. The proposals are based on harmonisation work already ongoing or done in the SDMX sponsoring organisations.

SDMX will pursue this first attempt of harmonising structural metadata by creating and adding more of those SDMX Cross-domain code lists (for example, on geographical areas) and by improving and completing the existing lists already released.

A.6 Publishing data and metadata using SDMX

A.6.1 Scope of this chapter

This chapter of the User Guide looks at the different scenarios for web publishing, focusing on the business case and best practices for each type of scenario.

A.6.2 Overview of this chapter

SDMX standards and guidelines can be usefully applied to the publication of statistical data and reference metadata on websites.

The major aspect of web publishing, which distinguishes it from data reporting or collection over the Internet, is that the counterparties which receive the data are not known ahead-of-time to the publisher. That is, anyone with access to the data may come to the site to browse and view it, even if they only just discovered the existence of the data and metadata. Thus, it cannot be assumed that the user knows anything about the data or metadata other than that which is provided on the site by the publisher, or which is provided for in the SDMX technical Specifications or SDMX Content-Oriented Guidelines, as relevant.

A.6.3 Scenarios

There are several basic scenarios for the use of SDMX in Internet publication of statistics. These are briefly described as they will be discussed in this paper:

- 1) **SDMX as a standard XML format for end users:** this scenario is primarily aimed at providing data and metadata in a more easily used delivery format. Having an XML format for statistical data and metadata means that users can more easily load it into databases and other applications. Because XML has become ubiquitous in databases and development environments, this is generally well-received by users, as seen as very positive.
- 2) **SDMX as a format for web presentation:** if the capability exists to realize SDMX-ML from internal data sources, it makes sense to use this standard XML format as a way of creating other output formats such as CSV, HTML, PDF, etc. Further, it makes the use of web services possible, as web services require an XML format, whether done on a local web-site or a remote one. This is an internal, strategic scenario, because having a single standard output format will lower maintenance costs of web dissemination in the long term, by decoupling data and metadata sources and presentations. This scenario pairs naturally with the distribution of SDMX-ML as an end-user format.
- 3) **SDMX as the basis for web portals:** this scenario uses SDMX version 2.0 to provide a single place on the Internet where users can come to get all of the data and metadata for a particular domain, or for a topic or set of related topics. Very similar to contemporary ideas of a “data bank”, this scenario leverages the SDMX standards to make it easier for people to find and use the data they need within a “data bank”. Typically, there will be the use of SDMX-ML formats (as for scenarios 1 and 2 above) but also the use of a standards-based registry, and a use-specific front-end application, which can be seen as a catalogue tool, allowing users to browse through collections of data effectively to find what they need.
- 4) **SDMX as the basis for services provision:** this scenario is aimed at users who both produce and consume data, but who will benefit from having a single mechanism for sharing this data. In this scenario, the creator of the site is not

responsible for the data or metadata themselves, but provides a portal mechanism which may or may not have a front end “catalogue” application for the use of a community. The business driver behind this scenario is to remove the middle-man from the statistical publication process. By creating a mechanism and policies for use, data producers and consumers are connected by the services operator, but still are responsible for their own data and metadata. This is an extension of the web portal case above, but provides a greater degree of strategic benefit by allowing for a decentralization of responsibility for content, while still offering a single place on the Internet for the acquisition and use of data and metadata. This scenario can be seen as the fullest realization of a “data sharing” process.

A.6.4 SDMX as a standard format for end-users

Statistical websites typically offer data as an HTML presentation or PDF presentation, often with a downloadable format such as CSV or Excel files for those who wish to use the data in other applications. Although useful, there tends to be an informal and non-standard presentation of accompanying metadata, found as column headings or in other parts of the display in all of these formats. In some cases, there is a distinct paucity of metadata.

The advent and adoption of XML as a common processing format provides a way to improve on how data can be made available. Any type of XML tends to be more metadata-rich than other formats such as HTML, PDF, or CSV. SDMX provides such a metadata-rich XML format for statistical data, as well as allowing the structural metadata (DSD) to be expressed in its own XML format. Between these two formats, the data becomes not only available as XML, with much accompanying metadata, but also can be fully understood through the processing of its accompanying Data Structure Definition.

This is a significant improvement from the perspective of the end-user. The reasons are many, and exist both for those who wish to load the data into databases or other applications, and for those who need to integrate the data with data coming from other sources, an activity which requires a thorough understanding of the parameters of a particular data set. Because SDMX formats are standard, it is possible for dedicated tools to fully automate the display and loading of data and metadata into databases and other applications. This degree of automation is not generally possible with CSV, HTML, or PDF formatting of the data.

Some of the major benefits are as follows:

- The data is surrounded by markup (tags) which make it easy to understand what a particular bit of data or metadata is, both to application code and to human developers. This is a generic benefit of XML, but it is significant, because many developers and development tools today are designed to work with XML formats.
- The concepts used to describe and identify the data are made explicit, as used in the data, and as understood by the person who structured the data set. Formal descriptions of the concepts and their descriptions exist in the DSD, and can be used by both human developers and applications receiving the data.
- The code lists used to identify the data and to carry accompanying metadata in attributes are spelled out for the user. This is useful not only in understanding the data, but also for receiving applications wishing to do a default display. Because the codes are themselves associated with descriptions, it is possible to create generic displays which will work with any kind of SDMX-formatted data on the fly.

For those wishing to provide SDMX-ML formatted data as a website deliverable, a few best practices should be kept in mind:

- Like CSV, SDMX-ML provides a useful downloadable format, but may not be sufficient for all users. It does not typically replace HTML and PDF displays, but supplements them. It is primarily a machine-readable format, whereas HTML and PDF are primarily human-readable.
- The DSD should be made available to end users, containing the description of the data structure (that is, which concepts are used as dimensions, attributes, and measures) as well as the codelists used to represent values for these. This should be in the standard SDMX-ML format, and should be easy to find at a fixed URL. Ideally, each DSD is a complete, single file for each data format used on the site, although this may not be ideal for the maintenance of shared resources such as code lists.
- The XML schemas, generated according to the standard bindings from the DSD, should also be made available on the site. There are several different flavours of standard XML in SDMX – with the DSD and schemas available, users will have access to everything they need to transform SDMX-ML data between these standard flavors. Because it cannot be predicted exactly how a given user will process the data, it is best to simply provide users with a complete set of data, schema, and metadata, and let them use the free SDMX tools or other tools to process the data as they need.
- SDMX-ML provides a message structure which has a standard set of header fields, including message identifiers and human-readable names. It is important that SDMX-ML files – whether DSDs or data – have unique identifiers for each message, and also for each data set. Although these identifiers may not be important to the producer of the SDMX-ML data (especially if generated on-demand by a database) they are critical to users who wish to perform and use several queries from a single database, or who repeat a query over time as data is updated. Even the provision of simple generated ID numbers for each data set and each message will allow users to more easily organize the data they have received. Further, it is good practice to use a consistent code to identify the agency involved in producing SDMX data and metadata. While these fields are not required in the version 1.0 formats, they are very useful to users.

Ultimately, the users of data will be pleased to have a standard, metadata-rich XML format containing all data and metadata disseminated by an organisation. With a wealth of generic tools which can use these formats today, and an increasing number of SDMX-specific ones, data will become better understood and more easily manipulated by users.

Example: U.S. Federal Reserve Board Data Download Project

There are many good real-world examples of this use of SDMX today. One which we can highlight is the use of SDMX at the U.S. Federal Reserve Board in the Data Download Project (<http://www.federalreserve.gov/datadownload/>). This is a pilot covering several types of data, including industrial production, flow of funds, selected interest rate data. There is also an ability to configure a persisted query for the specific data which is of interest, and to obtain it in SDMX-ML format. Another example – and perhaps a more straightforward one – is the example from the Federal Reserve Bank of New York, where there are financial data sets such as foreign exchange rates in SDMX format (<http://www.newyorkfed.org/xml/index.html>).

In both cases, the XML schemas and the SDMX structure files are available to users as well as the data itself. The utility of these data downloads is that people who wish to process the data in their own applications now have access to the data and relevant metadata in a standard format which is easy to use.

A.6.5 SDMX as a format for web presentation

When SDMX-ML data files and DSDs are produced for distribution on a website as an end-user deliverable, it is natural to consider the benefits from the use of SDMX-ML in publishing and maintaining a website. Today, there are two common approaches to generate the distribution formats for statistical data. Either a set of equivalent formats (HTML, PDF, CSV, Excel, etc.) are generated from a database, and then published as static files on the server, or they are generated from the database when requested by the user. The latter strategy is easier to maintain, and requires less server space in the long run.

As described above, SDMX-ML formats for data and metadata can be added to this list of distributed formats. While representing a benefit for the end user, the simple generation of SDMX-ML actually adds marginally to the workload of those maintaining the site, since now there is an additional format which must be produced and distributed, whether on-the-fly with a script, or as a set of static files.

Looking at the historical development of XML and related markup languages, however, we will realize that they were first designed to function as production formats to help with exactly this type of problem. If one can generate the XML, then all other formats can be derived from it. Because XML is a metadata-rich format (and especially in the case of SDMX-ML, which has DSDs to supplement the XML tagging), it contains all of the basic information needed to format the data for the automatic creation of human-readable or processible formats.

Further, most development platforms, whether .NET, Java, or XSL-based, have an excellent set of tools for transforming XML into other outputs. Regardless of whether the SDMX-ML formats are generated and stored as static files, or derived automatically from a database at run-time, there is a real benefit to using this format to drive transformations into other output formats. It should be noted that most web browsers today are capable of taking an XML file and an XSLT stylesheet and displaying HTML directly to the user, with the transformation taking place in the client web-browser.

The greatest benefit of using this approach is strategic, rather than tactical. Maintaining a transformation from a known database format or from a standard XML format would seem to be equivalent tasks. However, databases change over time, and sometimes data will be coming from more than one internal source. Further, each internal database format tends to be different, requiring developers to understand each database as they work with it.

If instead all databases produce a standard XML format, and also are capable of expressing the structural metadata as an XML DSD, then maintaining a system over

time as it grows and changes is greatly simplified. Not only is it easier to work with the data in various development tools, but new developers do not need to spend time learning each database – they only need to be familiar with the standard formats. Ultimately, the maintenance of a system, and the integration of its various parts, becomes less resource-intensive.

This approach is known as “loose coupling of applications using document interfaces”, and it is becoming an increasingly common technique. Web services technologies assume this approach. Ultimately, it makes not only the publishing of formats for web-delivery easier, but also the internal integration of any type of statistical application within the organization.

Example: European Central Bank’s Euro Area Indicators

One example of this approach is the European Central Bank’s Euro Area Indicators. These data are disseminated in an SDMX-ML format from the ECB website, and are then used by other European central banks for re-styling into a page which is suitable for use on their own sites (labels in the national language or languages, look and feel the same as the rest of the site...) This transformation is performed in real time using XSLT stylesheets.

Further explanations and links to the relevant web pages of the national central banks are available here:

<http://www.ecb.eu/stats/services/escb/html/index.en.html>

<http://www.ecb.eu/press/pr/date/2005/html/pr051206.en.html>

A.6.6 SDMX as the basis for web portals

Up to this point, the scenarios described will work with either the 1.0 or 2.0 versions of the SDMX Technical Specifications. The web portals scenario relies on the use of an SDMX Registry, which was introduced as standard functionality only within the version 2.0 Technical Specifications. Thus, the newer version of the standard is required for this scenario.

A “web portal” is a generic term that has many equivalents in different domains. In statistics, the term “data bank” is often used, although that arguably has a somewhat narrower definition. Regardless of the term, however, a web portal is a simple concept: instead of making users resort to search engines to locate statistical data and metadata, it should be possible for them to go to a single, known site on the Internet, where they know they can get a complete and up-to-date picture of the available data and metadata. This makes it much easier for users, and allows them to build applications which can rely on the availability of data and metadata through a known process, at a known location.

The business case for web portals is very much focused on the end-user – the next scenario demonstrates how this approach can also be of benefit to those who publish data coming from many sources. The simple type of SDMX-based web portal consists of two parts: an SDMX registry, and a catalogue application. The SDMX Registry is a type of common web-services application – a registry - which supports the SDMX interfaces for registering and querying data.

Each DSD is registered, so that its concepts and codelists can be queried. Each data set is registered, so that it can also be queried. In the SDMX Registry, data sets are organized as “data flows”, which means that as data is updated over time, each subsequent data set in the series can be located and used as part of an organized group of related updates. A “data flow” is like a subscription to a journal – each month there is a new issue of the magazine, but the whole set of issues can be thought of as a single, ongoing whole. Data flows allow this type of grouping with sets of related data.

In the version 2.0 Technical Specifications, there is also provision for non-structural metadata sets which document the data in any useful fashion. This is similar to the publication of data using DSDs. Now, one can also publish metadata sets with their structures (including concepts and code lists) as Metadata Structure Definitions (MSDs). These MSDs and metadata sets can also be registered in the SDMX Registry, and then become available for querying.

The SDMX Registry represents a specialized database which knows exactly what data and metadata are available. It allows fine-grained queries regarding the structure or contents of the data to be made. Further, it has a mechanism for notifying applications or users when something of interest has changed, using either the dedicated SDMX interfaces, or an RSS feed. These capabilities are very useful in creating a web portal, but they represent only half the picture.

The other half is the catalogue application. SDMX does not specify a particular catalogue, because this is very dependent on user requirements. Each catalogue application will be designed and developed to allow users to locate and use data and metadata in a way that is specific to their needs, and may vary depending on the statistical content found within the registry.

It is possible to make some general statements about the types of functionality typically found in a catalogue application. The first of these is the location of data and metadata. Search engines such as Google are typically unaware of the structure of statistical data, and do not allow focused searches to be made on specific concepts. Thus, if one wants to find all data related to Slovenia, one will search and get a hit for every mention of Slovenia on the entire web. This takes a lot of time and effort to sort through, and may not produce the needed data, because it does not understand the coding of the statistical data sets one is looking for.

An SDMX Registry understands the coding, and restricts its searches only to the domain of data and metadata which makes up the web portal. Thus, a search on Slovenia would find all the data sets which share the coding for Slovenia – there would be no random hits from across the web, and no misses generated by the use of foreign languages or contextual inconsistencies. SDMX Registries are designed to support focused queries on a known set of data and/or metadata, utilizing their knowledge of the structural metadata found in DSDs and MSDs. Thus, the reliable location of data becomes possible. This functionality is fundamental to a catalogue application, which must first allow users to find what they are looking for.

It should be noted that catalogue applications often use real-time queries to create specific search screens for data. Thus, if one knows that the DSD has five dimensions, one can create a search screen on the fly with five search parameters, showing values for the codes which exist in the data, displayed in human-readable form. The existence of structural metadata within an SDMX registry makes this type of focused searching possible to implement. If the SDMX Content-Oriented Guidelines are also followed, then registry searches can be even more powerful, as all of a common set of concepts will share a name and a representation.

Another ubiquitous catalogue functionality is data retrieval. An SDMX Registry contains information about the location and formatting of the data and metadata. This makes it possible for the catalogue application to retrieve one or more files at the users request, and pre-process it so that only the data or metadata of interest is retrieved. If the data is coming from a variety of sources – a set of static files, and maybe from some real-time database queries – the catalogue application can hide all of the complexity of assembling the data into a useable set expressed in a standard XML format.

Additionally, the data and/or metadata can then be formatted using the techniques described above in scenarios A and B. Thus, the user could conduct a search and then ask for the data to be delivered as a single HTML page, or as a PDF, or in some other

format. The standard formats and metadata of SDMX allow for generic tools to be able to produce these outputs, regardless of the structure of the inputs. Thus, the maintenance burden for the web portal operator can be reduced.

An SDMX Registry supports multiple classifications of data and metadata sets and structures. Thus, a standard classification such as that offered by the SDMX Content-Oriented Guidelines can be used, as well as domain- or organization-specific classifications which may be more fine-grained. Classifications allow users to more easily find what they are looking for and thus support the location and retrieval of data and metadata sets.

The scenario described here assumes that a publisher of statistics is collecting and processing the data within their organization, and then offering the web portal as a convenience for users. The operator in this scenario is responsible for all of the data and metadata which the portal contains, and uses the SDMX standards as a way of automating this convenient type of web site for the location and use of statistical data and metadata. This approach mostly benefits the user of statistical data, by making it easy for them to find and use the data and metadata.

The next scenario demonstrates the extension of this technique to benefit the collector of the statistical data and metadata as well, and indirectly to facilitate the needs of the entire community of data reporters, collectors, and users.

Example: Joint External Debt Hub

The best example of the portal approach today is the BIS-IMF-OECD-World Bank Joint External Debt Hub (<http://www.jedh.org>). At this site, which is hosted by the World Bank with a supporting registry hosted by OECD, we can see data published on the sites of several institutions acting as a single point of access. At the JEDH, data is published in SDMX-ML format as the hub assembles and caches it for a single, unified presentation to the user. This mechanism uses an SDMX Registry to find and present the data to the end user.

See §A.7.6 Joint External Debt Hub

A.6.7 SDMX as the basis for services provision

The operator of a data bank (or web portal) traditionally acts first as a collector of data, then performs processing and quality-assurance, and then acts as a disseminator of the statistical data and metadata. With the support of the various SDMX standards and guidelines, this function can be distributed among all of the players in this scenario: the producers and reporters of data and metadata, the collector and processor of the data and metadata, and the disseminator of data and metadata. This distributed arrangement maximizes efficiencies for all the counterparties involved.

In essence, the operator of a web portal becomes a service provider to the data producers and reporters. These services are several:

- 1) **As the developer and maintainer of the central web portal (or data bank):** This is simply running an SDMX Registry and catalogue applications which allow reporters to register the data (and metadata) they have published, and to allow users to find it and use it (as described in the scenario above). These services may include the provision of tools for data and metadata reporters to use in their integration with the web portal and on-going registration of data and metadata sets.
- 2) **As the policy provider for governing the community:** In order to guarantee the usefulness and comparability of various data and metadata, there may be standards and guidelines which need to be created, published, and enforced. These functions naturally belong to the service provider operating the web portal on behalf of a community of users. Thus, if a particular DSD or MSD is to be used, the

web portal operator is a natural agency for coordinating the development and use of that structure. If data is to be published on a particular schedule, this publication can be monitored by the operator of the web portal. These types of activities can often be automated, based on the wealth of metadata found in an SDMX Registry.

- 3) **As the quality assurance agency:** Because the web portal itself provides visibility into the entire communities' data and metadata for those within the community, many aspects of quality assurance (such as comparing your own data against everyone else's to check that it is within reasonable variation) can be performed by almost anyone. However, this processing naturally occurs at the center – in the same place where the SDMX Registry and the catalogue application are hosted. Also, there is a governance function associated with meeting minimum guidelines in terms of quality, making the web portal operator the natural monitor of data quality. It is possible to implement automated processes where the registration of a data set or metadata set triggers processes which check the quality of the data, and only if the quality is sufficient will the existence of the data set become visible to others within the community through the catalogue application at the web portal.

These services are beneficial to creators and reporters of data, to the web portal operator, and to users. Creators and reporters of data appreciate having control of their own data. Because they simply register data and metadata sets which live on their own sites, they retain control over the data, and are able to correct and update it more easily than if they have to work through another organization for these functions. Further, because they have visibility into and access to the data and metadata of other reporters for the same type of data, they have much better information on which to base their own processing and quality checks before publishing.

Web portal operators no longer have the responsibility of collecting, processing, and disseminating the entire communities' data. Because the creators of data have first-hand knowledge of their own data sets, they generally have an easier time making sure that it is complete and correct. This function is out-sourced to those who are most able to perform it, rather than being inherited by the collecting agency at the center of the community. This allows the operator of the web portal to focus on those activities for which they are best suited: providing a mechanism for the data creators/reporters to use in disseminating their data to users, and setting standards and monitoring the quality of the data and metadata made available through the portal. Typically, this re-allocation of functions is less resource-intensive for the operator of the web portal, and is better suited to the function of their organization.

Users of data benefit from having a single point on the Internet for obtaining the data and metadata they need, as seen in the preceding scenario. Because many creators and reporters of statistics are also consumers of those reported by others, all the members of the community benefit from this arrangement.

Example: Joint External Debt Hub

The Joint External Debt Hub given as an example in the preceding section is in fact an example of web services provision, although this fact is not visible to the end user. The platform used to develop and operate the site – the World Bank's "Data Development Platform" – is a web-services-based application which provides the sophisticated front end to the user, but relies on web-services connections behind the scenes to provide the data and metadata.

Thus, it serves as an example of both a simple portal, bringing data together from many sites, but also as an example of a single service-driven application which uses an SDMX Registry and data available in SDMX-ML.

See §A.7.6 Joint External Debt Hub

A.6.8 Conclusions

The SDMX standards and guidelines can be used in a number of different ways to improve the quality of statistics publishing, and can help in making the dissemination of statistics a more efficient process. It is important to understand the business reasons for using SDMX, and these have been characterized in the scenarios above. Whether the focus is improving the usability of the published statistics, or increasing the strategic maintainability and efficiency of the dissemination process, SDMX can be used to achieve these goals.

It is evident that the scenarios described here are not mutually exclusive – rather, they build on each other, showing a path from the simple publishing of a standard XML format through the de-centralization of the reporting-collection-dissemination process using a data-sharing model. These steps can be taken in the sequence presented or not, depending on the needs of the organization and community of counterparties.

SDMX offers a variety of tools designed to support these activities, and by considering the business goals of a particular project, it is possible to see which types of implementation are most appropriate.

A.7 Uses for an SDMX Registry

A.7.1 Scope of this chapter

This chapter of the User Guide explains what an SDMX Registry is and what it can be used for.

A.7.2 Introduction

Version 2.0 of the SDMX technical standards introduced a series of enhancements to Version 1.0, in particular for metadata management (with the introduction of the “metadata structure definition” to describe the structure of a reference metadata set) and for the registry architecture, useful for providing visibility to large amounts of data and metadata.

SDMX envisages the promotion of a data-sharing architecture using the pull mode to facilitate low-cost and high-quality statistical data and metadata exchange: a data reporting organization publishes data once, and lets their counterparties “pull” data and related metadata as required. The data-sharing architecture is based on the possibility of discovering easily where data and metadata are available and how to access them.

The SDMX Registry plays an important role in this architecture; in fact it can be seen as a central application which is accessible to other programs over the Internet (or an Intranet or Extranet) to provide information needed to facilitate the reporting, collection and dissemination of statistics.

In its broad terms, the SDMX Registry – as understood in web services terminology – is an application which stores metadata for querying, and which can be used by any other application in the network with sufficient access privileges. It can be seen as the index of a distributed database or metadata repository which is made up of all the data provider’s data sets and reference metadata sets within a statistical community.

It is important to stress that registry services are not concerned with the storage of data or reference metadata sets. Data and metadata sets are stored elsewhere, on the sites of the data providers. The registry is only concerned with providing information needed to access the data and reference metadata sets. An application which wants a particular data or metadata set would then query the registry for the URL, and then go and retrieve the data or metadata set directly from the provider's web server.

This document reports on the implementation and setup of the SDMX Registry at Eurostat, as this is the cornerstone of an architecture for data and metadata exchange aimed at facilitating collection, processing and dissemination of statistics. Moreover it describes the distinct registry modules and its purposes.

A.7.3 Functions of an SDMX Registry

An SDMX Registry performs a number of tasks:

- It provides information about what data sets and metadata sets are available, and where they are located.
- It provides information about how the data sets and metadata sets are provided: how often they are updated, what their contents are, how they can be accessed, and similar questions.

- It provides information about the structure of data sets and metadata sets, answering questions like: What code lists do they use? What concepts are involved?
- It allows applications to sign up (or subscribe) for notifications, so that when a data set or metadata set of interest becomes available, the application will be automatically alerted.

These functions form the basis on which an SDMX Registry is organized. There are three layers, which correspond to the first three bullet points above, while the subscription/notification functionality is available for all of these layers:

- The Data and Metadata Registry
- The Provisioning Metadata Repository
- The Structural Metadata Repository

A.7.4 Architecture of an SDMX Registry

In general terms, an SDMX Registry is based on a structural metadata repository which supports a provisioning metadata repository which supports the registry services, according to a “layered” architecture as represented in Figure A.7.1.

Registration	Discovery	Subscription Notification	Other Services
Provisioning Metadata Repository Provision Agreement, Data Sources, Constraints, etc			
Structural Metadata Repository Data Structure Definition, Metadata Structure Definition, Item Scheme, etc			

Figure A.7.1: Schematic architecture of SDMX Registry/Repository

Structural Metadata Repository Layer: the Structural Metadata Repository Layer contains metadata such as Data Structure Definitions, Metadata Structure Definitions, Maintenance Agencies, etc. This layer must allow structural definitions to be created, modified and removed in a controlled way, also allowing the structural metadata to be queried and retrieved either in part or as a whole. Structural metadata is information about how data sets and metadata sets are structured. This type of information is needed by applications to process the data and metadata sets. Thus, once an application has discovered and retrieved a data set, it can then query for the structural metadata which goes along with that data set. In addition to concepts and code lists, the structural metadata repository contains many other pieces of needed information, including categorization and classification schemes, lists of organizations, and so on.

Provisioning Metadata Repository Layer: provisioning metadata is information about how data and metadata sets are made available by data providers. This is analogous to a “service level agreement” whereby a data provider commits to publishing a dataflow or metadataflow according to an agreed schedule. This layer includes details about the online mechanism for getting data (e.g., a queryable online database or a simple URL) as well as information about the release calendar, sources and contents of the data and metadata sets. This information is stored in the SDMX

Registry, which is why this layer is termed a “repository”. All of its information is accessible over the Internet using SDMX-ML messages, just as for all communications with the SDMX Registry.

Data and Metadata Registry Layer: this portion of the SDMX Registry acts like a catalogue or a phone book, allowing applications to look up and see which data and metadata are available. Data and metadata sets are categorised to facilitate searches. Although there is a recommended high-level categorisation for statistical data in SDMX, each Registry can have a tailored categorisation which matches the statistics within the statistical community that the registry serves.

Subscription/Notification: a user may wish to receive updates regarding a specific part of the contents of any of the layers of the Registry, for instance when a new data set is published or when a list of organizations is updated. There are two ways to receive such updates: the first is the subscription/notification mechanism, using SDMX-ML messages. Another mechanism is the use of RSS feeds which is typically used for updates to data. In either case, the update can serve as a trigger for the receiving application – to go out and get the updated or new data set, or to perform some other automated process.

As the objective of an SDMX Registry is to allow organisations to publish statistical data and metadata in known formats such that interested third parties can discover and interpret them accurately and correctly and within the shortest possible timescale, the setup of structural metadata and the exchange context (referred to as “data provisioning”) is a key issue, which involves a series of steps for maintenance agencies:

- 1) Agreeing and creating a specification of the structure of the data (called “data structure definition, DSD) which defines the dimensions, measures and attributes of a dataset and their valid value set.
- 2) Defining a subset or view of a DSD which allows some restriction of content (called a “dataflow definition”)
- 3) Agreeing and creating a specification of the structure of metadata (metadata structure definition, MSD) which defines the attributes and presentational arrangement of a metadata set and their valid values and content
- 4) Defining a subset or view of an MSD which allows some restriction of content (called a “metadataflow definition”)
- 5) Defining which subject matter domains are related to the dataflow and metadataflow definitions to enable browsing
- 6) Defining one or more lists of data providers (which includes metadata providers)
- 7) Defining which data providers have agreed to publish a given dataflow and/or metadataflow definition - this is called a provision agreement

Publishing the data and metadata involves the following steps for a data provider:

- 1) Making the metadata and data available in SDMX-ML conformant data files or databases (which respond to an SDMX-ML query with SDMX-ML data) - the data and metadata files or databases must be web-accessible, and must conform to an agreed dataflow or metadataflow definition (data structure or metadata structure subset)
- 2) Registering the published metadata and data files or databases with one or more SDMX Registries
- 3) Notifying interested parties of newly published or re-published data, metadata or changes in structural metadata. The Registry can optionally support a subscription-

based notification service which sends an email announcing all published data that meets the criteria contained in the subscription request.

Discovering published data and metadata involves the following steps:

- 1) Optionally browsing a subject matter domain category scheme to find dataflow definitions (and hence DSD) and metadataflows which structure the type of data and/or metadata being sought
- 2) Build a query, in terms of the selected data structure or metadata structure definition, which specifies what data are required
- 3) Submit the query to an SDMX Registry which will return a list of (URLs of) data and metadata files and databases which satisfy the query
- 4) Processing the query result set and retrieving data and/or metadata from the supplied URLs

A.7.5 Examples of working SDMX Registries

Several organisations have set up SDMX Registries. The following sections describe some of these cases, emphasizing how the registries are used to support the requirements of different projects and organisations.

It should be noted that the software for two different SDMX Registry implementations is available via the Tools pages of the SDMX website (see §B.2 Obtaining and using SDMX Tools).

A.7.6 Joint External Debt Hub

The Joint External Debt Hub (JEDH) is a collaborative product of BIS, IMF, OECD and the World Bank within the Inter-Agency Task Force on Finance Statistics. It brings together in one central location comprehensive national and creditor/market data and metadata. JEDH provides instant and easy access to external debt statistics in graphic and numerical form as well as facilitating macroeconomic analysis, and cross-country and data source comparisons.

JEDH uses an agreed DSD for the external debt data, which was jointly defined by the participating agencies. The DSD supports a wide range of external debt data to serve the first production version of the JEDH, as well as possible future versions that would include additional detailed data from the various data providers. The countries covered include industrial as well as developing countries.

The JEDH uses an SDMX Registry which functions as a web "catalogue" or "phone book" where people and computer programs have a single place to go to find out what information is available for some specific purpose. The key point is that the registry does not store the data itself, but only the metadata about data available on data producer's web sites.

The technology behind the JEDH website, including the JEDH SDMX Registry, provides smooth automation of live data supply to the site and its maintenance. Figure A.7.2 gives an overview of the JEDH architecture. The automated process comprises the following steps.

- 1) Participating organizations (data providers) register their SDMX-ML files with the JEDH SDMX Registry.
- 2) The registered SDMX-ML files are retrieved, and then transformed into the World Bank's Development Data Platform (DDP) database.

3) The JEDH uses the DDP technology, and standards for data dissemination to present both national- and creditor/market-based data on external debt statistics and selected foreign assets in a user-friendly environment.

The DDP database is live and instantly displays the refreshed results via web reports, charts and maps, which can be published and shared over the web.

The World Bank was responsible for developing the JEDH website, which is at <http://www.jedh.org>. The JEDH SDMX Registry is hosted by OECD.

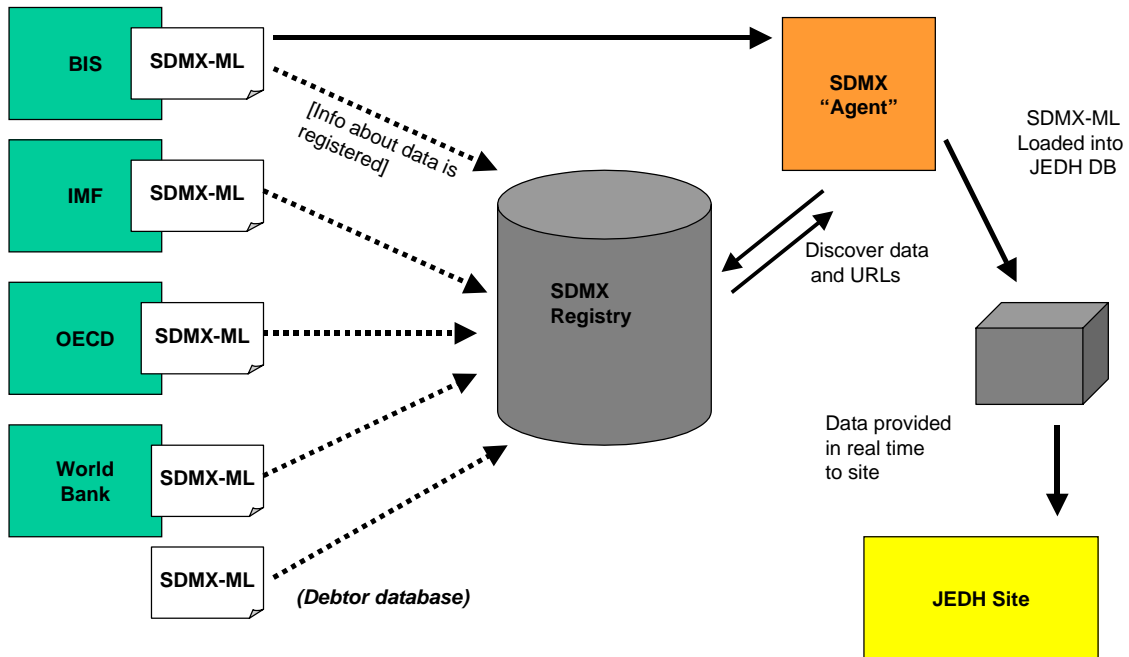


Figure A.7.2: Architecture of the JEDH

A.7.7 FAO CountryStat

CountrySTAT (<http://countrystat.fao.org>) is a statistical framework and applied information system for analysis and policy-making to organise, integrate and disseminate statistical data and metadata on food and agriculture coming from different sources. CountrySTAT gathers and harmonises scattered institutional statistical information so that information tables become compatible with each other at the country level and with data at the international level. CountrySTAT

The CountrySTAT approach is based on the application of data and metadata standards of FAOSTAT, SDMX and GAUL (Global Administrative Unit Layers). The web-based system was developed since May 2004 using PX-Web at FAO Headquarters and successfully tested in the statistical offices of Kyrgyz Republic, Kenya and Ghana during 2005. Many other developing and developed countries have shown interest and are adopting it.

CountrySTAT and RegionSTAT are client programs using the PC Axis engine, running on Windows. The .PX file format is native to these applications. The Publication Servers are Windows servers which form an existing part of the CountrySTAT and RegionSTAT tools - what is proposed here is an extension of their functionality.

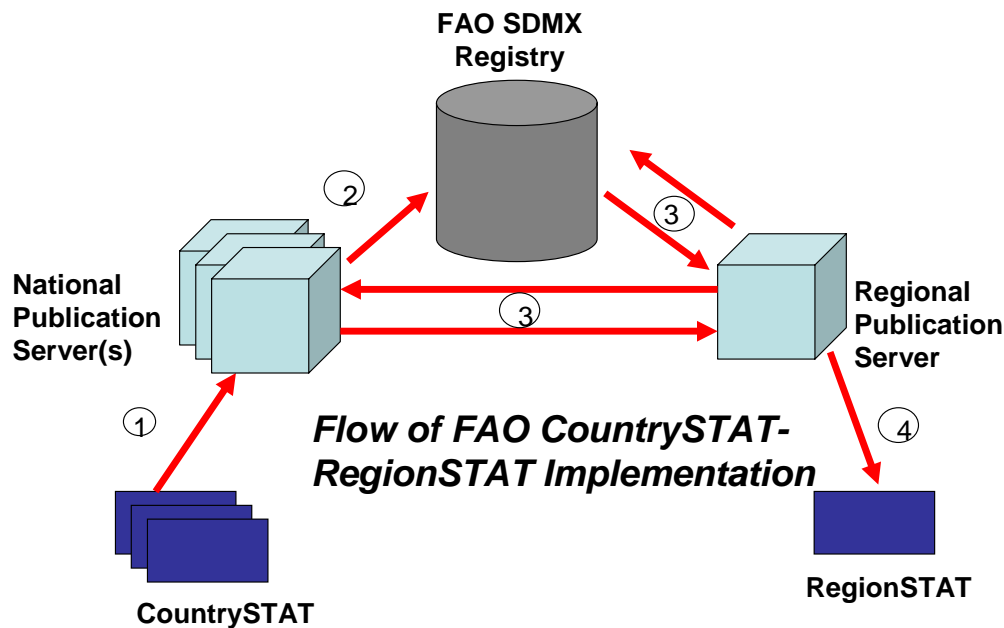


Figure A.7.3 Architecture of the CountrySTAT system

The overall project involves a flow of data between two end-point applications: CountrySTAT, which is the originating application for the reported data, and RegionSTAT, which is the application used by the receiving organization (see Figure A.7.3). Between these two end-point applications are national and regional publication servers, and facilitating the communications between these is an instance of an SDMX Registry, termed the "FAO SDMX Registry".

The FAO SDMX Registry is an implementation of the existing freeware tool from Metadata Technology Ltd (which is available via the Tools pages of the SDMX website, see §B.2 Obtaining and using SDMX Tools). The FAO SDMX Registry is hosted by the the Open Data Foundation⁴.

The four key activities which take place and which are documented below are:

- 1) The structural and provisioning metadata is set up in the FAO SDMX Registry.
- 2) The CountryStat data set file(s) is transformed to SDMX-ML and placed at a web addressable location (i.e. available at a URL)
- 3) The existence of the data set is registered in the FAO SDMX Registry
- 4) The Regional Publication Server periodically queries the registry to determine if any new data sets have been registered (note that the subscription/notification service was not available for this demonstration)
- 5) The newly registered data set(s) are retrieved from the web location and integrated into the database of the Regional Publication Server.

⁴ <http://www.opendatafoundation.org>

A.7.8 Eurostat SDMX Registry

The Eurostat SDMX Registry currently provides a web-based user interface and web services for interacting with the SDMX structural metadata objects in use within Eurostat and with statistical partners (Concept Schemes, Code Lists, Data Structure Definitions, Metadata Structure Definitions, Data Flows, Metadata Flows, Category Schemes, Organization Schemes, Provision Agreements).

Eurostat's SDMX Registry will initially be used as a "back-office application" for internal access to data and metadata structure definitions by eDAMIS (the single entry point), the SODI infrastructure and by other information systems inside Eurostat. The registry will also enable NSIs and other external organisations to obtain DSDs and other structural metadata, such as the MSD for the Euro SDMX Metadata Structure (ESMS).

The Eurostat SDMX Registry comprises three major blocks:

- the Database (DB) which is the storage of all the data maintained within the Registry;
- the Web Service (WS) which exposes the registry interface via Simple Object Access Protocol (SOAP);
- the Graphical User Interface (GUI), a web interface for human interaction with the registry. The GUI offers a user-friendly web interface for adding/deleting/updating structural information, as well as import/export features for interaction with SDMX-ML and GESMES structure definition files.

Eurostat's SDMX Registry was developed during 2006-2008. As of September 2008, the first version of the registry was installed and running in three different environments:

Test environment: accessible only internally at Eurostat and used for test purposes.

Production environment: the GUI is accessible at the following URL:

<https://webgate.ec.europa.eu/sdmxregistry>

Training environment: used as a "sandbox" for training courses and presentations, without the risk of modifying the real Registry. Access to the training environment can be provided on request to Eurostat. The GUI is accessible at the following URL:

<https://webgate.training.ec.europa.eu/sdmxregistry>

The production and training installations are accessible to any external user using the GUI via a CIRCA account⁵ in read-only mode. Web services are accessible, at the moment, only by internal applications; external applications will be able to access the registry web services in during 2009.

The Eurostat SDMX Registry contains all the DSDs used by Eurostat and ECB. It is envisaged to add further further content, including all harmonised structural metadata and the ESMS (Euro SDMX Metadata Structure) MSD.

Alongside the Registry, Eurostat has also developed and deployed the Data Structure Wizard application, which is a desktop application designed to work with any SDMX-compliant registry for editing and viewing SDMX structural metadata objects. The Data Structure Wizard is a Java standalone application that can be used both off-line and on-line, depending on user choice and access rights.

⁵ To register for a CIRCA account, go to: <http://circa.europa.eu/Public/irc/dsis/Home/main> and click on "Sign up".

A.7.9 References

SDMX Standards, Version 2, November 2005 - Registry Specifications: Logical interfaces; Implementor's Guide for SDMX standards. Available via the Standards page of the SDMX website

Eurostat SDMX Registry User Guide: included as part of the Eurostat SDMX Registry implementation: see the Tools pages of the SDMX website.

Lindblad, Bengt-Åke, Marco Pellegrino and Francesco Rizzo (2008) *Registry facilities for supporting the exchange of statistical data and metadata*. METIS April 2008. <http://www.unece.org/stats/documents/ece/ces/ge.40/2008/wp.8.e.pdf>

The FAO CountryStat SDMX project: detailed explanation of the use of SDMX standards and technical constructs. Paper given at SDMX Global Conference, January 2007, Washington DC, USA.

<http://sdmx.org/docs/2007/Conf07/doc%2033%20Capacity%20Building%20Room%20Document%20-%20FAO%20Project%20details.doc>

Caprazli, Kafkas, Arofan Gregory and Chris Nelson (2006) *FAO CountrySTAT SDMX project: A functional overview of a cross country implementation of the Food and Agriculture SDMX Registry Model*. 15th Annual PC-AXIS Reference Group Meeting, Reykavík, Iceland, 21-23 August 2006

<http://www.statice.is/lisalib/getfile.aspx?itemid=4634>

Similar document: <http://unstats.un.org/unsd/acsub/2006docs-8th/SA-2006-13Add1-FAO.pdf>)

B TUTORIAL THREAD

B.1 SDMX technical overview

B.1.1 Scope of this chapter

This chapter of the User Guide provides an overview of SDMX from a technical perspective, as a basis for the following chapters of the tutorial thread.

B.1.2 Technical standards: from Version 1.0 to Version 2.0

Version 1.0 of the SDMX standards (ISO/TS 17369) is concerned mainly with data sets and the structural definitions for data sets. Figure B.1.1 shows the main artefacts supported by the version 1.0 standards.

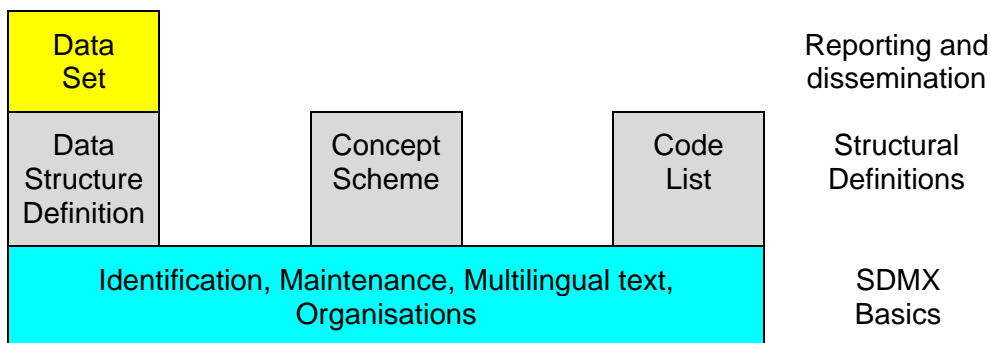


Figure B.1.1: Main artefacts supported by the Version 1.0 standards

Version 2.0 has seen a large increase in the functionality supported by the SDMX standards. The additional functionality has resulted largely from requirements requested, both during the development of the version 1.0 standard and from early adopters of the version 1.0 standard. The version 2.0 standards introduced support for reference metadata (metadata structure definition, metadata set), mapping between structural metadata (e.g. code list maps, data and metadata structure maps, category and concept scheme maps), Category Scheme (e.g. this supports a domain category scheme), hierarchic code lists, transformation metadata (to better support primary reporting), and registry based functionality (data and metadata provisioning, subscription and notification, data and metadata registration, and data and metadata discovery). The diagram below shows the main artefacts supported at version 2.0.

Data Set	Metadata Set	Data Metadata Provisioning	& Subscription & Notification	Registration	Discovery	Reporting and dissemination		
Data Structure Definition	Metadata Structure Definition	Structure Maps	Concept Scheme	Category Scheme	Code List	Hierarchic Code Scheme	Transformations & Expressions	Structural Definitions
Identification, Maintenance, Multilingual text, Organisations								SDMX Basics

Figure B.1.2: Main artefacts supported by the Version 2.0 standards

B.1.3 Scope of the SDMX standards

The scope of the SDMX standards can be described in relation to Figure B.1.3.

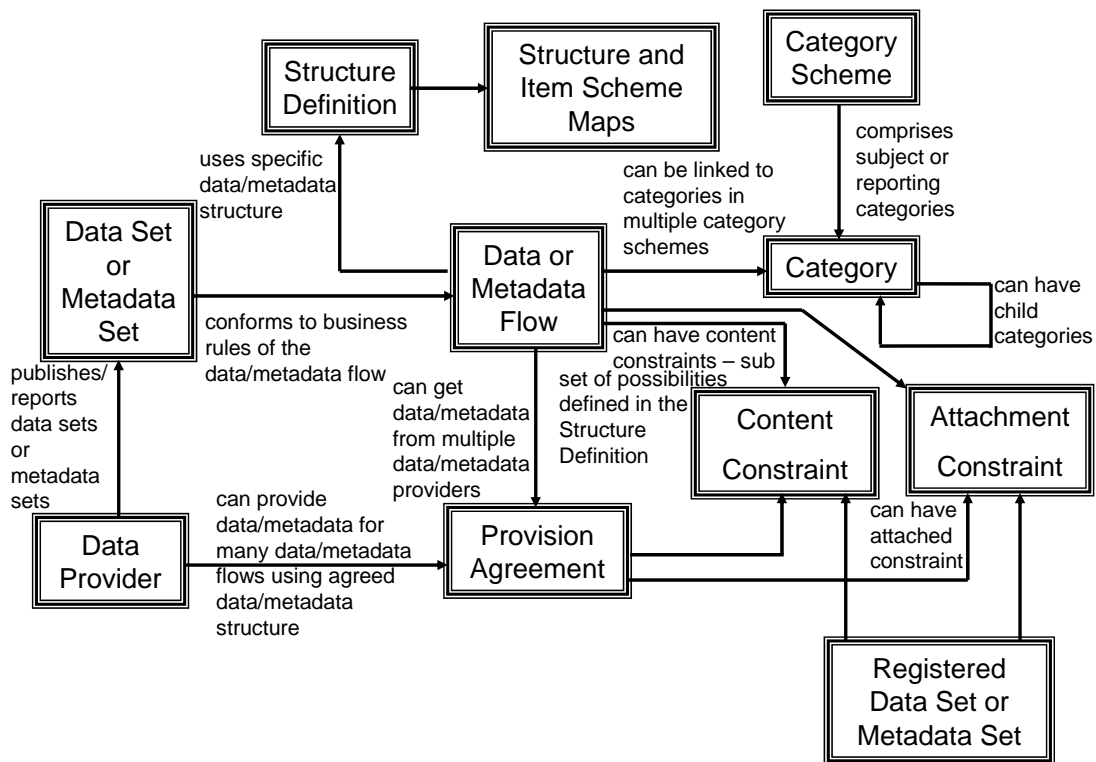


Figure B.1.3: Scope of the SDMX standards

It is important to understand at the outset that SDMX can be used at any level in the data provision or dissemination chain provided that the data are aggregated⁶, and the reference metadata aspects of SDMX can be used for all metadata regardless of what it is or for which object it relates. So, for a National Statistical Institute (NSI), SDMX can be used for upwards reporting at the international level, or the NSI can use SDMX for the reporting of data and metadata from other national authorities. Similarly, an organisation can use SDMX to define the structures that support data and metadata dissemination.

⁶ In certain cases, SDMX has also been used effectively for microdata.

Data Sets and Metadata Sets are reported over time by Data Providers according to certain business rules. These organisations can operate at the national level reporting to international organisations, at the sub-national level reporting to the national level, or at any level to support dissemination. The business rules include the agreed format of the data and metadata and the reporting schedule. In SDMX terms these business rules relate to the Data Flow or Metadata Flow. The Data Flow and Metadata Flow describe the provision of data and metadata over time. They are analogous to the concept of a periodical in magazine publishing: the periodical describes the artefact for which a magazine is published on, say, a monthly basis – at the very least the data held for the periodical would be its publication dates, but would also link to other information such as distribution channels, advertising sponsors etc. The Data Set or Metadata Set is analogous to the specific magazine (e.g. the June 2007 edition). For data and metadata the Data Set/Metadata Set contains the actual data and metadata values for a specific period or periods, whereas the Data Flow/Metadata Flow holds or links to the other information. Importantly, the Data Flow/Metadata Flow can be linked to one or more subject matter domains (Category), and must be linked to exactly one Structure Definition (Data Structure Definition or Metadata Structure Definition). The subject matter domain link allows for a drill down for queries for data or to assemble data sets that are similarly structured.

The Structure Definitions includes Concepts and Code Lists and their link to the Data Structure and Metadata Structure Definitions that use them.

In any reporting scenario (e.g. from national level to international level, or from sub national to national level) one Data Provider can report data or metadata for many Data or Metadata Flows, and for one Data or Metadata Flow there may be many Data Providers. The Provision Agreement is the union of these two – it contains the details of the reporting of data or metadata by a specific Data Provider for a specific Data or Metadata Flow. Note that in the SDMX model the term Data Provider refers to a provider of data or (reference) metadata.

A Constraint may be applied to either the Data or Metadata Flow or the Provision Agreement. The purpose of the Constraint is to define the sub set of the entire universe of data or metadata implied by the Data or Metadata Structure Definition that is applicable to either the Data or Metadata Flow or the provision of data for that Data or Metadata Flow by a specific Data Provider (i.e. for the Provision Agreement). The Constraint is also used in a registry centric scenario to hold the details of the registered Data or Metadata Set.

B.2 Obtaining and using SDMX Tools

B.2.1 Scope of this chapter

This chapter explains how the freely available SDMX IT tools can be used in the implementation of SDMX standards in local IT systems.

B.2.2 Introduction

Adoption of the SDMX technical standards and guidelines by organisations involved in statistical data and metadata exchange will need to result in actual implementation of the standards in the local statistical IT systems. Statistical and IT experts need to understand the basics of SDMX when starting such a project. Along the way it becomes obvious that developing specific SDMX-based functionality is a task, which also other organisations are faced with. As a natural consequence efficiency gains can be achieved by offering to share one's developments with each other.

As part of its advocacy activities SDMX promotes the provision of freely available tools and open source software products that offer functionality needed for this implementation of the SDMX standards. Various types of tools for different purposes and target audiences are available: demonstration tools, production modules and production applications.

B.2.3 Types of Tools

SDMX demonstration tools help statistical as well as IT experts to understand the basic principles of SDMX and allow them to execute (on a small scale) the different types of functionality required for setting up an "SDMX capable" statistical processing system. They are meant for educational and demonstration purposes and are often used in SDMX courses. A "DSD Builder Tool", for example, can be used to take the first steps in understanding DSDs when practicing how to define data structures for data that are to be processed in the institution. Loading freely available public DSDs into the tool will allow one to analyse the relationships between different data structures. Similar examples can be given for other functionalities, such as the conversion between different SDMX technical formats or the creation of actual SDMX data and metadata files.

SDMX production modules provide specific functionality required in SDMX-based statistical processing at an "industrial strength" level. This means that they could be directly integrated into a production application to perform a specific task in the processing workflow. Examples are the "SDMX checker suite" developed by the ECB and the "SDMX converter" offered by Eurostat. The SDMX production modules are targeted to IT experts who need to implement SDMX functionality. The modules can considerably shorten the required development time for an individual organisation implementing SDMX in its local systems.

SDMX production applications are a set of production modules, usually offered by one organisation, that cover a wider range of SDMX statistical processing functionality in an integrated way. They have been developed by that organisation for its own internal use and it can thus be expected that the application is actually used for production purposes in that organisation. The target audience are business and IT experts tasked to implement an SDMX production system in an organisation. They will want to evaluate such applications from both the business and the IT "fit" for their own workflow and application environment.

B.2.4 Availability of SDMX Tools

The existing SDMX tools have been developed (or commissioned) by SDMX sponsors and other organisations actively involved in implementing SDMX. Up-to-date Information about available tools can be found via the Tools page on the SDMX website, from which the user is redirected to a site from which they can actually be downloaded. Information about the specific tools, such as their purpose, the actual functionality implemented, and the licence type under which they are being offered, is also available here. Statistical and IT experts can also scan the available tools for a given functionality that they need to implement in order to find the appropriate tool or modules that could be useful for the particular task they are faced with.

The SDMX Tools Forum⁷ is an informal and volunteer meeting-place on the internet for exchanging views about the use and implementation of SDMX tools and related standards more generally, with a particular focus on how these can be facilitated by emerging tools. Neither the tools nor the Tools Forum have any formal relationship with the SDMX initiative.

B.2.5 Open Source and shared development

The tools found via the SDMX website are in general made available under open source licences. The specific licence type for a given tool will be clearly indicated to the user upon downloading the tool. The growing SDMX user community is encouraged to evaluate and test the already existing tools and modules and to consider offering their own developments that might fill a gap with respect to the functionality required for an efficient SDMX implementation. Individuals or organisations wishing to contribute their developments under an open source licence are invited to contact the SDMX Secretariat⁸.

It is expected that the growing application of SDMX standards will implicitly lead to the harmonisation of not only of the statistical data exchange, but also of the statistical processing applied in different organisations. Starting from the freely available open source tools and modules this could result in cases of even closer cooperation between different organisations in the form of shared development of statistical processing applications.

⁷ <http://www.metadatatechnology.com/userforum/>

⁸ secretariat@sdmx.org

B.3 XML-based technologies used by SDMX

B.3.1 Scope of this chapter

This chapter provides some very basic information about XML and technologies using XML, which are commonly used in the implementation of SDMX. For more in-depth explanations, readers are advised to look at some of the many books and online tutorials on XML.

B.3.2 Introduction

XML, the eXtensible Markup Language, is designed to describe data⁹.

It is extensible in that it can be used to create new languages for describing particular kinds of information. In this sense, SDMX-ML can be understood as an XML-based language for describing statistical data. To understand how SDMX-ML is commonly used, it is important to know what is meant by a *schema*, by *XSLT*, and by *namespaces*.

Schema

An XML schema defines the permitted building blocks of an XML document, using the terms of a schema definition language, which is itself written in XML.

SDMX makes use of the schema definition language known as W3C XML Schema (XSD). There exist other schema definition languages, such as DTD and Relax NG; these are not used by SDMX.

An XML Schema Definition:

- defines elements that can appear in a document
- defines attributes that can appear in a document
- defines which elements are child elements
- defines the order of child elements
- defines the number of child elements
- defines whether an element is empty or can include text
- defines data types for elements and attributes
- defines default and fixed values for elements and attributes

The schema is contained in a separate file, with the extension “.xsd”.

⁹ This description draws on the w3schools tutorials *Introduction to XML*, (http://www.w3schools.com/xml/xml_what_is.asp) and *Introduction to XML Schema* (http://www.w3schools.com/schema/schema_intro.asp)

eXtensible Stylesheet Language Transformations (XSLT)

- XSL = XML Style Sheet Language
- XSL describes how the XML document should be displayed
- XSLT - a language for transforming XML documents

XSLT is used to process an XML document, generating an output document in XML or another format (text, HTML, etc).

Example: An XML file before transformation to HTML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/xsl" href="cdcatalog.xsl"?>
<catalog>
<cd>
  <title>Amandla</title>
  <artist>Miles Davis</artist>
  <country>USA</country>
  <company>Decca</company>
  <price>10.90</price>
  <year>1989</year>
</cd>
. . .
</catalog>
```

Example: XML to HTML via XSLT

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <xsl:template
match="/">
  <html>
    <body>
      <h2>My CD Collection</h2>
      <table border="1">
        <tr bgcolor="#9acd32">
          <th align="left">Title</th>
          <th align="left">Artist</th>
        </tr>
        <xsl:for-each select="catalog/cd">
          <tr>
            <td><xsl:value-of select="title"/></td>
            <td><xsl:value-of select="artist"/></td>
          </tr>
        </xsl:for-each>
      </table>
    </body>
  </html>
</xsl:template>
</xsl:stylesheet>
```

Namespaces

```
<?xml version="1.0"?>
<note
xmlns=http://eurostat.cec.eu.int
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xsi:schemaLocation="http://eurostat.cec.eu.int note.xsd">
```

- Namespaces (xmlns) are used to resolve conflicts, where the same name is used for different elements in different files
- The namespace defines a prefix for any name in the file
- Although the namespace and schema references look like URLs, this is just a naming convention: these lines do not have to point to a real file location.

B.3.3 Why is XML ideal for the purpose of SDMX?

With XML, data can be exchanged between otherwise incompatible systems. Here are some simple examples to explain the way these XML technologies work.

XML carries information

```
<note date="12/05/2005">
<to>Bernhardt</to>
<from>Giuseppe</from>
<heading>Reminder</heading>
<body>Don't forget the slides!</body>
</note>
```

XML can be validated against a schema

```
<?xml version="1.0"?>
<note
xmlns=http://eurostat.cec.eu.int
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xsi:schemaLocation="http://eurostat.cec.eu.int/note.xsd">
...
</note>
```

A violation of the Schema rules will be detected by an XML application and will usually cause further processing of the file to be halted.

Example: XML Schema Definition (note.xsd)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://eurostat.cec.eu.int"
xmlns="http://eurostat.cec.eu.int" elementFormDefault="qualified">

<xs:element name="note">
<xs:complexType>
<xs:sequence>
<xs:element name="to" type="xs:string"/>
<xs:element name="from" type="xs:string"/>
<xs:element name="heading" type="xs:string"/>
<xs:element name="body" type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

B.3.4 Web services

A "web service"¹⁰ is defined by the World Wide Web Consortium (W3C)¹¹ as "a software system designed to support interoperable machine-to-machine interaction over a network".

There are a number of technical approaches to achieve this interaction; SDMX follows the W3C approach where

- a web service has an interface described in a machine-processable format, specifically WSDL¹⁸;
- other systems interact with the web service, in a manner prescribed by its WSDL description;
- communication between systems uses SOAP (Simple Object Access Protocol) messages, typically conveyed using HTTP.

Conventional applications and services traditionally expose their functionality through application programming interfaces (APIs). Web services can be understood as a kind of API; they provide a public version of the function calls which can be accessed over the web, using the standard web protocol (HTTP).

In principle, web services provide a completely platform-independent mode of communication between systems, allowing interaction between organisations using different operating systems and programming languages.

Web services exchange data via SOAP messages, which are constructed using XML: this is how the data passed between web services is formatted. SDMX-ML, as a standard XML for exchanging data and metadata within the statistical realm, provides a useful XML format to support web services for statistics.

The following simplified example shows a SOAP message to a web service which provides prices of stocks and shares¹². The request message carries a query for the price of Motorola shares (MOT). The response message carries the answer "\$14.50".

¹⁰ "Web Service" is often written with initial capital letters to emphasise that this is a specific technical term which is distinct from everyday usage relating generally to information delivered via the web.

¹¹ <http://www.w3.org/TR/ws-gloss>

¹² This example is taken from the O'Reilly article *A Web Services Primer* by Venu Vasudevan
<http://webservices.xml.com/pub/a/ws/2001/04/04/webservices/index.html>

Example: SOAP messages: (a) query

```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml;
charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePrice
      xmlns:m="Some-URI">
      <symbol>MOT</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example: SOAP messages: (b) response

```
HTTP/1.1 200 OK Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePriceResponse
      xmlns:m="Some-URI">
      <Price>14.5</Price>
    </m:GetLastTradePriceResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

B.3.5 XML tools

Software for processing XML files is freely available and can be used on SDMX-ML files, which conform to all the rules and conventions of XML.

B.4 Differences between SDMX-EDI and SDMX-ML

B.4.1 Scope of this chapter

This chapter discusses the technical and functional differences between the two formats provided: SDMX-EDI (based on the EDIFACT¹³ syntax) and SDMX-ML (based on XML). Both formats are directly derived from the SDMX Information Model and are thus equivalent. The different syntaxes place some restrictions on their use. A table at the end of the chapter provides a quick overview. For a detailed technical discussion, please see chapter 3.3 of the SDMX Implementor's Guide.

B.4.2 SDMX-EDI

SDMX-EDI has been in use for data exchange (also under its previous name: GESMES/TS) since the year 2000. It is the basis for the Version 1.0 SDMX standard and fully implements this version, ie Data Structure Definitions and Data (and related metadata) Messages. It does not support Metadata Structure Definitions and metadata messages based on these and also does not support other features of SDMX Version 2.0, e.g. hierarchical code lists. SDMX-EDI has one message type for Data Structure definitions and one message type for data (and related metadata) messages. SDMX-EDI data messages are very compact and about 10% or less of the size of an SDMX-ML message of the same content. SDMX-EDI is mainly used for the batch exchange of large data amounts between systems or organisations.

Any valid SDMX-EDI message can be transformed into an equivalent SDMX-ML message and back without any information loss, for example, using SDMX Tools.

B.4.3 SDMX-ML

The SDMX-ML format was introduced with the Version 1.0 SDMX standard and has been enhanced since then to fully support SDMX Version 2.0. This includes the support for Metadata Structure Definitions and metadata messages based on them. The XML syntax is designed to leverage URIs and other Internet-based referencing systems, and these are used in the SDMX-ML structure messages. These options are not available in SDMX-EDI messages. For the exchange of data (and related metadata) SDMX-ML offers three different message types: "generic" data message, "compact" data message and "utility" data message. SDMX-ML is usually used in web-based systems involving the exchange of limited amounts of data and when SDMX 2.0 features are required.

An SDMX-ML message making use of features added in SDMX Version 2.0 cannot be transformed into an equivalent SDMX-EDI message, as the latter only supports SDMX 1.0. If only SDMX 1.0 features are used, SDMX-ML can be transformed to SDMX-EDI and back with any information loss.

Only SDMX-ML can support interaction with an SDMX registry/repository.

¹³ EDIFACT: Electronic Data Interchange for Administration, Commerce and Transport
(http://www.unece.org/trade/untid/texts/d100_d.htm)

B.4.4 Comparative table: SDMX-EDI and SDMX-ML

Feature	SDMX-EDI	SDMX-ML
Usually used for	Batch exchange of large amounts of data between systems or organisations	Web-based applications dealing with limited amounts of data, taking advantage of XML functionality
SDMX Version 1.0	Fully supported	Fully supported
SDMX Version 2.0	Not supported	Fully supported
Conversion to other format	Fully convertible to SDMX-ML and back	Fully convertible to SDMX-EDI, if only SDMX 1.0 features were used.
Character encoding	ISO 8879-1	UTF-8
Data typing	Different data typing mechanisms may limit interoperability. If SDMX-EDI is to be supported in parallel to SDMX-ML, certain conventions need to be observed.	
Data Structure messages (DSD)	Only Version 1.0 supported	Version 1.0 and 2.0 supported
Data and metadata messages based on DSDs	Data and metadata message	“generic” data and metadata message “compact” data and metadata message “utility” data and metadata message
File sizes (data messages)	Roughly 10 % of SDMX-ML message with same content	Roughly 10 times larger than SDMX-EDI message with same content (ZIP will reduce file size considerably.) “Compact” message results in smallest file sizes.
Data delete messages	Yes	Only “generic” and “compact” message
Metadata Structure Definitions (MSD)	Not supported	Fully supported
Metadata messages based on MSDs	Not supported	Fully supported
Interaction with SDMX Registry	Not supported	Fully supported

B.5 SDMX message types for data

B.5.1 Scope of this chapter

This chapter explains the six standard message types for data and data structure definitions.

B.5.2 The different kinds of standard messages

In order to appreciate the flexibility and elegance of SDMX, it is advisable to look at the whole collection of SDMX-ML messages and to understand how they relate to each other. There are six standard messages for data and DSDs: see Table B.5.1.

Table B.5.1: Standard message types for data and DSDs

	Name of message	Short description	Schema file
1	Structure Definition Message	Contains a data structure definition	Fixed
2	Generic Data Message	Conveys data in a form independent of a data structure definition. It is designed for data provision on websites and in any scenario where applications receiving the data may not have detailed understanding of the data set's structure before they obtain the data set itself.	Fixed
3	Compact Data Message	Exchange of large data sets in a data structure definition-dependent form	Derived from data structure definition message
4	Utility Data Message	For schema-based functions, such as validation, in a data structure definition-dependent form	Derived from data structure definition message
5	Cross-sectional Data Message	Exchange of many observation types in a data structure definition-dependent form	Derived from data structure definition message
6	Query message	To query a database to obtain an SDMX-ML message as the result	Fixed

Any SDMX-ML message is constructed according to an XML schema (contained in a schema definition file with the extension .xsd). In some cases the schema file is already fixed in the SDMX standards. In other cases, the schema is derived from the data structure definition message.

The following sections give more explanations for each message type, with links to examples.

B.5.3 Structure definition message

- Contains a data structure definition: all SDMX-ML message types share this common XML expression of the metadata needed to understand and process a data set;

- Supports annotations;
- Concepts and code lists could also be referenced by the message;
- This leverages Internet referencing mechanisms;
- Example file: <EUROSTAT_STS.xml>.

B.5.4 Generic Data Message

- Conveys data in a form independent of a data structure definition message, as the structure is imbedded in the message;
- Used when applications receiving the data do not have detailed understanding of the data set structure before they obtain the data set itself;
- For transmission of partial data sets (incremental updates) and whole data sets;
- Not a particularly compact format;
- All aspects of the data set easily available;
- Does not provide strict validation between the data set and its structural definition using a generic XML parser (the parser cannot validate the codes since they are not contained in the schema). In terms of XML syntax, all codes and observation values are elements.
- All key values specified at the series level;
- Attribute values attached at the same level as in data structure definition;
- Example file: <EUROSTAT_STS_STS_IND_PROD_M_BE_generic.xml>

B.5.5 Compact Data Message

- Exchange of large data sets in a data structure definition-dependent form;
- Specific to the data structure definition of the data set it encodes;
- Follows mappings between constructs in Structure Definition message and compact format;
- For exchange of large data sets in XML format;
- For transmission of partial data sets (incremental updates) and whole data sets;
- In terms of XML syntax, all codes and observation values are attributes. The permissible values of the codes are defined in the schema (which is specific to the data structure definition) so that a generic XML parser can be used to validate a data file against its structural definition.
- Key values can be at Group level;

- Example file: <EUROSTAT_STS_STS_IND_PROD_M_BE_compact.xml>.

B.5.6 Utility Data Message

- This message may be considered a special-purpose message. It is intended for schema-based functions in a data structure definition-dependent form:
- Specific to the data structure definition of the data set;
- For validation and other XML schema functions;
- Cannot be used for incremental updates (requires complete data set);
- Key values at the series level;
- Example file: < EUROSTAT_STS_STS_IND_PROD_M_BE_compact.xml>.

B.5.7 Cross-Sectional Data Message

- Exchange of many more than one observation type in a data structure definition-dependent form; it is intended for situations where the statistical data consist of multiple observations at a single point in time, or for each combination of dimension members in the multidimensional table. For example, in foreign trade statistics where, for combination of reporting country, partner country, commodity and time period there may be three observations: a value, a weight and a quantity;
- Specific to data structure definition of the data set;
- Key values from the Group down to the Observation level;
- Multiple observation values with different “measures”;
- Time at the Group level;
- Example file: < CrossSectionalSample.xml>

B.5.8 Query Message

- Used to convey a query to a database which then returns an SDMX-ML message;
- For web services and database-driven applications;
- Queries regard both data and structural metadata;
- Example file: < QuerySample.xml>

B.5.9 Deriving one SDMX-ML message from another

Much of the flexibility and elegance of SDMX standards arises from two facts. First, since all the messages are built on the underlying SDMX Information Model, some SDMX-ML messages can be derived from others, as shown in Figure 1. Tools are

freely available from the SDMX initiative¹⁴ to support the management of SDMX messages, including the translation between the different formats.

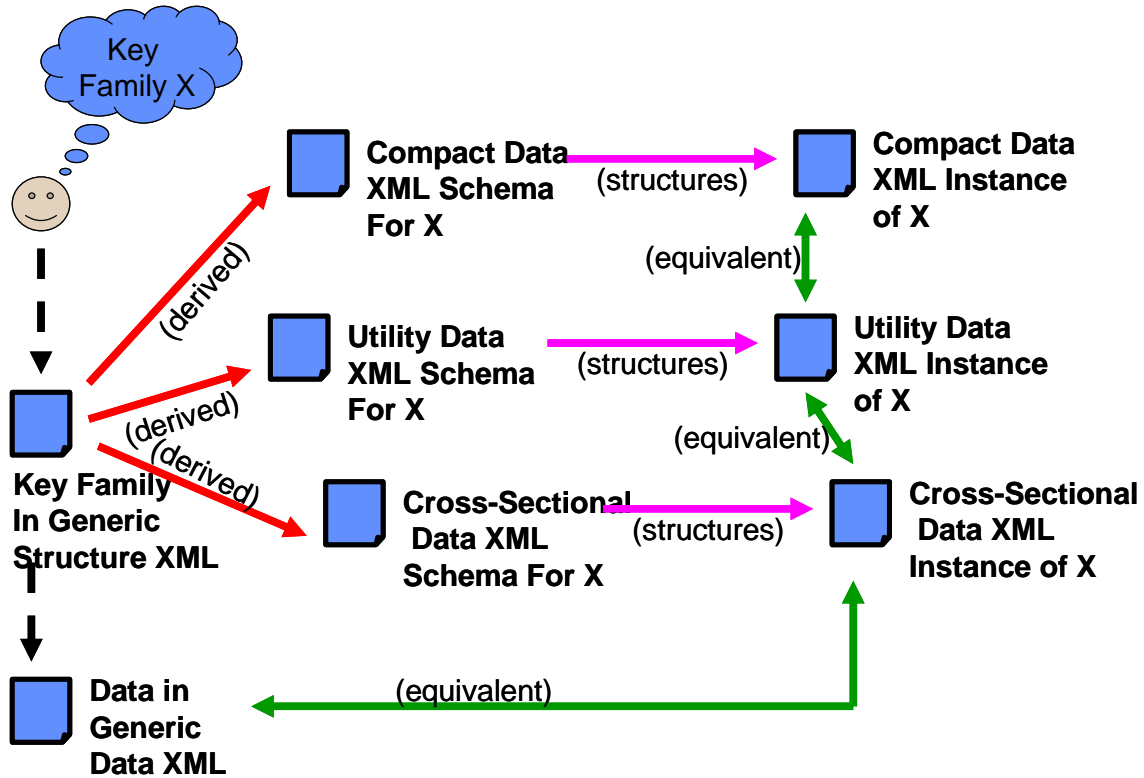


Figure B.5.1: Model-based equivalence of SDMX-ML messages

Second, because the SDMX-ML messages use standard XML, these transformations, and any other operations which the user may wish to perform on data conveyed with SDMX-ML, can be carried out using normal XML tools, such as XML editors and XSLT processors. High-quality tools are available – often for free – and well supported with documentation and training.

¹⁴ SDMX tools: available from the SDMX website.

B.6 SDMX message types for reference metadata

B.6.1 Scope of this chapter

This chapter explains the message types for reference metadata.

B.6.2 Message types

Three standard messages are used for reference metadata and MSDs: see Table B.6.1.

Table B.6.1: Standard message types for reference metadata and MSDs

	Name of message	Short description	Schema file
1	Structure Message	Contains a metadata structure definition	Fixed. This is the same schema as the structure definition message for data (see §B.5.2)
2	Generic Metadata Message	Conveys reference metadata in a form independent of a data structure definition.	Fixed
3	Metadata Report Message	Exchange of reference metadata according to a specific MSD.	Derived from metadata structure message

B.6.3 Structure Message

- The Structure Message contains the Metadata Structure Definition: all SDMX-ML Metadata Report Message types using the same MSD share this common structure of the metadata needed to understand and reuse a data set;
- It supports XML annotations containing explanations and further specification of the content of metadata to be provided (for instance guidelines for compilation);

Metadata concepts are described in terms of their content (definition, guidelines) representation (free text or code list) and usage (such as mandatory or optional). If a concept has to be represented by a code, the relevant code lists could also have to be referenced from within the message.

The MSD also describes the structure of the Metadata Report to be exchanged. The Metadata Report is normally composed of a hierarchy of metadata concepts which depends on the kind of metadata elements that a maintaining agency intends to exchange.

Table B.5.1 shows a sample of metadata attributes, as defined in the ESMS Report Structure defined by Eurostat. Information on where each referenced concept belongs to (for example, the SDMX cross-domain concepts list or any additional agency-specific concepts list) is included, as well as the 'Usage Status' (Mandatory/Conditional).

Table B.6.2: Metadata attributes as defined in the ESMS Report Structure

Concept Ref	Parent	ConceptScheme	Agency	Usage status
CONTACT		SDMX_CDC	ESTAT	Mandatory
CONTACT_ORGANISATION	CONTACT	SDMX_CDC	ESTAT	Mandatory
ORGANIZATION_UNIT	CONTACT	SDMX_CDC	ESTAT	Mandatory
CONTACT_NAME	CONTACT	SDMX_CDC	ESTAT	Mandatory
CONTACT_MAIL	CONTACT	SDMX_CDC	ESTAT	Mandatory
CONTACT_FAX	CONTACT	SDMX_CDC	ESTAT	Mandatory
STAT_PRES		ESTAT_ADD_CONCEPTS	ESTAT	Mandatory
SHORT_DESCR	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
CLASS_SYS	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
COVERAGE_SECTOR	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
STAT_CONC_DEF	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
STAT_UNIT	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
STAT_POP	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
REF_AREA	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
COVERAGE_TIME	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
BASE_PER	STAT_PRES	SDMX_CDC	ESTAT	Mandatory
UNIT_MEASURE		SDMX_CDC	ESTAT	Mandatory
REF_PERIOD		SDMX_CDC	ESTAT	Mandatory
FREQ DISS		SDMX_CDC	ESTAT	Mandatory
DISS_FORMAT		SDMX_CDC	ESTAT	Mandatory
NEWS_REL	DISS_FORMAT	SDMX_CDC	ESTAT	Mandatory
PUBLICATIONS	DISS_FORMAT	SDMX_CDC	ESTAT	Mandatory
ONLINE_DB	DISS_FORMAT	SDMX_CDC	ESTAT	Mandatory
MICRO_DAT_ACC	DISS_FORMAT	SDMX_CDC	ESTAT	Mandatory
ACCURACY		SDMX_CDC	ESTAT	Mandatory
ACCURACY_OVERALL	ACCURACY	SDMX_CDC	ESTAT	Mandatory
SAMPLING_ERR	ACCURACY	SDMX_CDC	ESTAT	Mandatory
NONSAMPLING_ERR	ACCURACY	SDMX_CDC	ESTAT	Mandatory
TIMELINESS_PUNCT		ESTAT_ADD_CONCEPTS	ESTAT	Mandatory
TIMELINESS	TIMELINESS_PUNCT	SDMX_CDC	ESTAT	Mandatory
PUNCTUALITY	TIMELINESS_PUNCT	SDMX_CDC	ESTAT	Mandatory
COMPARABILITY		SDMX_CDC	ESTAT	Mandatory
COMPAR_GEO	COMPARABILITY	SDMX_CDC	ESTAT	Mandatory
COMPAR_TIME	COMPARABILITY	SDMX_CDC	ESTAT	Mandatory

The MSD references the data flows that are described by the metadata which are structured according to the same Metadata Structure Definition.

For example, Table B.6.3 lists, following the Eurostat Dataset Naming Convention, the metadata flows that are structured according to the ESMS MSD, in the Short-Term Statistics domains. The variable numbers refer to the variables as defined in the Council Regulation 1165/98. The variable descriptions refer to the definition of variables as specified in the Commission Regulation 588/2001.

Table B.6.3: Examples of metadata flows

Metadata Flow Identifier	Variables	Description
SSTSIND_PRODR_MS	110	Reference metadata for production in industry
SSTSIND_TURNR_MS	120, 121, 122	Reference metadata for turnover in industry, total, domestic and non-domestic (total, Euro-zone, non-Euro-zone)
SSTSIND_ORDR_MS	130, 131, 132	Reference metadata for new orders received in industry, total, domestic and non-domestic (total, Euro-zone, non-Euro-zone)
SSTSIND_PRICR_MS	310, 311, 312, 340	Reference metadata for output prices in industry, total, domestic market, non-domestic market (total, Euro-zone, non Euro-zone), import prices (total, Euro-zone, non-Euro-zone)
SSTSCONS_PROD R (MS, QS)	110, 115, 116	Reference metadata for production in construction, total, building construction, civil engineering
SSTSRTD_TURNR_MS	120, 123	Reference metadata for turnover in retail trade, value or deflated
SSTSSERV_TURNR_QS	120, 123	Reference metadata for turnover in repair and other services, value or deflated (Quarterly)
SSTSSERV_TURNR_MS	120, 123	Reference metadata for turnover in repair and other services, value or deflated (Monthly)
SSTSSERV_PRICR_QS	310	Reference metadata for output prices in other services
SSTSSERV_EMPLR_QS	210, 211	Reference metadata for number of persons employed, Number of employees, in repair and other services

B.6.4 Generic Metadata Message

- Provides a single format that supports reporting metadata for any metadata structure definition: all reference metadata expressible in SDMX-ML format can be marked up according to this format.
- Performs only a minimum of validation
- Supports the creation of generic software tools and services for processing reference metadata

B.6.5 Metadata Report message

- For each MSD, an XML schema (specific to that MSD) can be created
- Performs validation (against the schema) on sets of reported data
- Less verbose than the Generic Metadata message
- Easier to use because the XML mark-up relates directly to the reported concepts

B.7 SDMX architectures using the pull mode for data sharing

B.7.1 Scope of this chapter

This chapter explains the SDMX architectures based on the pull mode, and provides guidance on how to implement these architectures. The explanation is based on a step by step description of a real-world example, based on the experience of implementing SDMX in a national statistical institute.

B.7.2 Introduction

SDMX, besides describing and specifying technical standards (the Information Model, message formats for data and metadata, Registry service definitions), comprises an IT architecture to be used for the efficient exchange and sharing of statistics.

For this purpose, SDMX identifies three basic process patterns (bilateral, gateway and data-sharing) and two modes (push and pull) regarding the exchange of statistical data and metadata.

In the data-sharing model a group of partners agree on providing access to their data according to standard processes, formats and technologies.

In the pull mode, the data consumer retrieves the data from the provider's web server. The data may be made available for download in an SDMX-conformant file, or they may be retrieved from a database in response to an SDMX-conformant query, via a web service running on the provider's server. In both cases, the data are made available to any organisation requiring them, in formats which ensure that data are consistently described by appropriate metadata, whose meaning is common to all parties in the exchange.

Data sharing using the pull mode is well adapted to the database-driven and data hub architectures. Both architectures provide the best benefits for the data producers because they can lessen the burden of publishing the data to multiple counterparties.

In both architectures, it is necessary to implement a notification mechanism, providing provisioning metadata in order to alert collecting organisations that data and metadata sets are made available by data providers, details about the online mechanism for getting data (for example, a queryable online database or a simple URL) and constraints regarding the allowable content of the data sets that will be provided.

At the heart of a data-sharing architecture there is often an SDMX Registry. This is a central location where structural and provisioning metadata can be found. In fact all the users/applications that need to access data can query the registry in order to know what data sets and metadata sets are available from data providers, and how to access them.

B.7.3 The database-driven architecture

The database-driven architecture is implemented by those collecting organisations that periodically need to fetch the data and to load them in their database. In general a batch process is used in order to automate the flow in which a whole or a partial dataset, including incremental updating, is used.

From the data management point of view, the pull approach within a database-driven architecture includes the following steps:

- 1) when new data are available, the data provider should:
 - a) create an SDMX-ML file containing the new data set

or

 - b) provide a web service (WS) that builds SDMX-ML messages upon request.

In both cases a provision agreement must be in place. In some SDMX on-going projects such as SODI provision agreements are provided in the format of an RSS file, in which a new feed entry is added, including the URL where the SDMX-ML file resides or an SDMX-ML Query message describing the new data set.
- 2) the data collector Pull Requestor reads the new feed entry and:
 - a) retrieves the SDMX-ML file from the specified URL, if it resides in a URL,

or

 - b) uses the Query Message included in the feed to query the data provider WS, if the data are prepared by the data provider WS.

Figure B.7.1 represents the database-driven architecture.

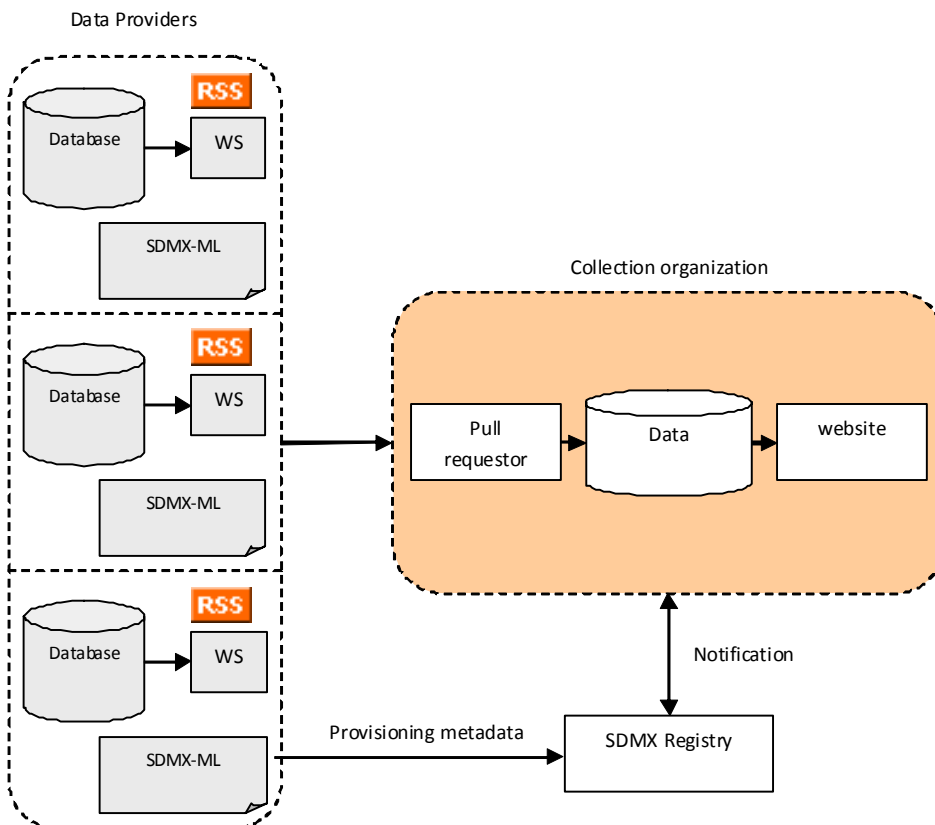


Figure B.7.1: Database-driven architecture

B.7.4 The data hub architecture

The data hub architecture consists of an accessible system providing involved actors with the following services:

- data providers can:
 - notify the hub of new sets of data and corresponding structural metadata (measures, dimension, code lists, etc.);

- make data available directly from their systems through a querying system.
- data users can:
 - browse the hub to define a dataset of interest via the above structural metadata;
 - retrieve the dataset from the data providers.

From the data management point of view, the hub is also based on agreed hypercubes or datasets, but here the hypercubes or datasets are not sent to the central system. Instead the following process operates:

- 1) a user identifies a dataset through the web interface of the central hub using the structural metadata, and requests it;
- 2) the central hub translates the user request in one or more queries and sends them to the related data providers' systems;
- 3) data providers' systems process the query and send the result to the central hub in a standard format;
- 4) the central hub puts together all the results originated by all interested data providers' systems and presents them in a human readable format.

Figure B.7.2 represents the data hub architecture.

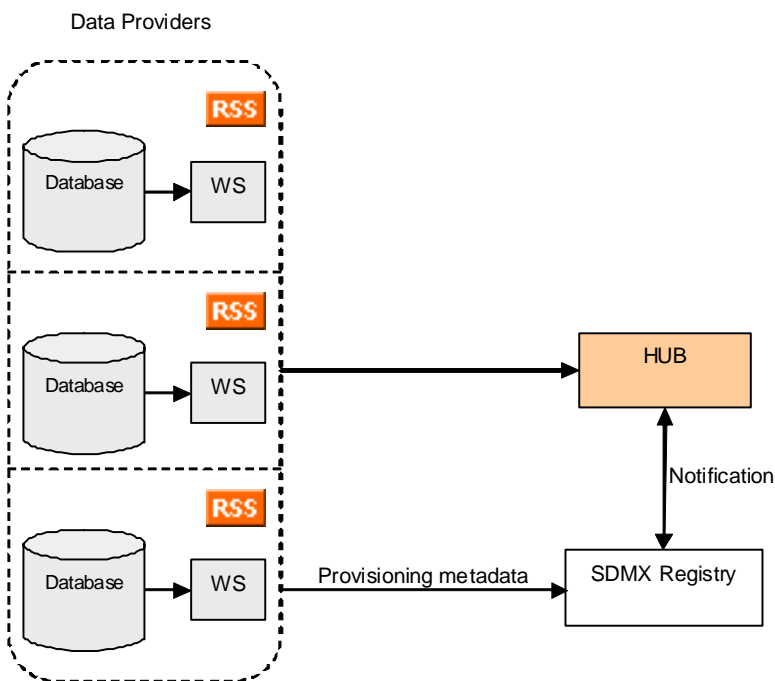


Figure B.7.2: Data hub architecture

B.7.5 Data producer architectures

In order to implement an SDMX IT architecture for data-sharing using the pull mode, several steps must be accomplished by a data producer and several questions must be considered:

- 1) which statistical domains are involved and where are the data currently stored?
- 2) which structural metadata are involved, and where are they currently stored?

- 3) what is the business process behind the data flow involved in the exercise?
- 4) will the SDMX data producer architecture be part of a data warehouse architecture, of a data hub architecture or of both?

Generally data and structural metadata that will be involved in the new SDMX information system are stored either in databases or in files. The two cases lead to different architectural approaches:

- a. data and structural metadata continue to be stored in files (for example: XLS, CSV, etc.) and the only need is to translate those files into SDMX-ML data files to be pulled by the data collector;
- b. data and structural metadata are already stored in a database and it is necessary to build suitable software interfaces in order to make the system “SDMX-compliant”.
- c. a separate special-purpose database is set up to store data and structural metadata. This database will be designed with the main aim of being part of an SDMX-compliant system. In this case the database can be modelled using the SDMX Information Model.

The cases (b) and (c) make it possible

- to extract SDMX-ML files from the database that will be made available to be pulled by data collectors;
- to allow the database to be queried directly through a web service.

Whichever type of data producer architecture is involved, a mapping process between structural metadata may be necessary, as explained below.

B.7.6 The mapping process

Generally data are described differently by data producers and data collectors using different concepts and code lists.

One of the main purposes of SDMX is to harmonize the structural metadata. When a new SDMX project starts, the first thing to do is define the necessary structural metadata (DSD and MSD) that will describe data and reference metadata sets (see §A.3 The SDMX information model: Data Structures and §A.4 The SDMX Information Model: Metadata Structures) Generally this task is performed by the institution that leads the project. Common structural metadata make it possible to exchange data among all the actors in a way that can be understood by everyone; unfortunately, the typical situation is that data producers have already their data in their databases described through local¹⁵ metadata. Therefore the first step to perform is to map the local structural metadata present in the data providers' system and those provided with the DSD.

In order to better explain this process the following real life example will be used.

The Eurostat SODI (SDMX Open Data Interchange) project deals with certain of the set of STS (short-term statistics) indicators defined by EU statistical legislation. This project implements a data-sharing architecture using the pull mode (although the push mode is also supported). Generally the majority of the involved data producers have their data already stored in a database and described using different local structural metadata.

¹⁵ The term “local” is used to indicate that the structure metadata are not SDMX-compliant. In general they are valid only for the system in which they are stored

This is the case for the Italian National Institute of Statistics (ISTAT), which disseminates those data through its short-term statistical databank Conlstat¹⁶.

Inside Conlstat, data are stored in a database using local structure metadata. A simplified snapshot of the database schema is provided in Figure B.7.3.

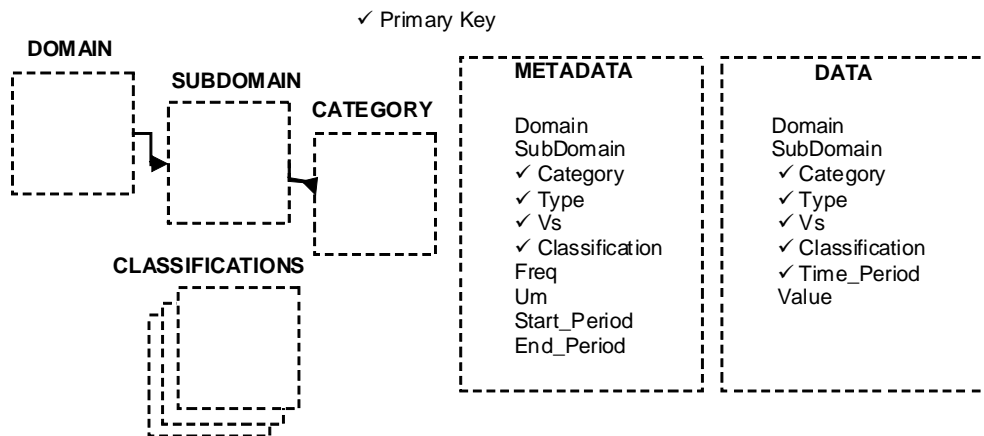


Figure B.7.3: Database schema for Conlstat

The schema is mainly based on two database tables: METADATA and DATA. The others tables can be considered as lookup tables useful to store code lists. Moreover two tables, respectively named DOMAIN and SUBDOMAIN, allow categorizing data in statistical subject-matter domains.

The main concepts used in order to describe each time-series are the following:

Category: short term statistical indicator

Type: adjustment indicator

Vs: stock/flow

Classification: NACE, SEC95, other classifications

Freq: frequency

Um: Unit of measure + Unit multiply + Base year

Start_Period: start period of the time-series

End_Period: last available period of the time-series

Each time-series is identified through a row in the METADATA table, and each field in that table has a correspondence in a particular lookup table representing a code list.

So the time-series *Monthly, neither seasonally or working day adjusted, Production in industry index base 2000, Mining and quarrying* is described in the following way:

Category: 11 (index of industrial production)

Type: g (neither seasonally or working day adjusted)

Vs: R (flow)

Classification: C (Mining and quarrying)

Freq: 12 (monthly)

Um: PE (index number - base 2000)

Start_Period: 01_1990

End_Period: 08_2008

¹⁶ Conlstat: <http://con.istat.it/>

The mapping process can be achieved by storing the resulting information in a special repository outside or inside the native database. In the case of the reported example it was chosen to use a repository inside the native database but without changing anything in the original tables. For this purpose, the following tables were added to the already existing schema:

- STS_METADATA: used to describe STS time-series (in order to describe other domains it would be necessary to add other tables, for example ESA_METADATA for National Accounts and so on);
- Some lookup tables useful to store within the local database some SDMX artefacts from the related DSD (for example: labels or even descriptions for concepts, code lists and dataflows)

The table STS_METADATA represents the place where the mapping process stores the mapping information. In fact, it inherits the base structure from METADATA, and some fields were added in order to cover all the concepts expressed in the SDMX DSD.

The resulting database schema after adding the new tables useful for the mapping process is shown in Figure B.7.4.

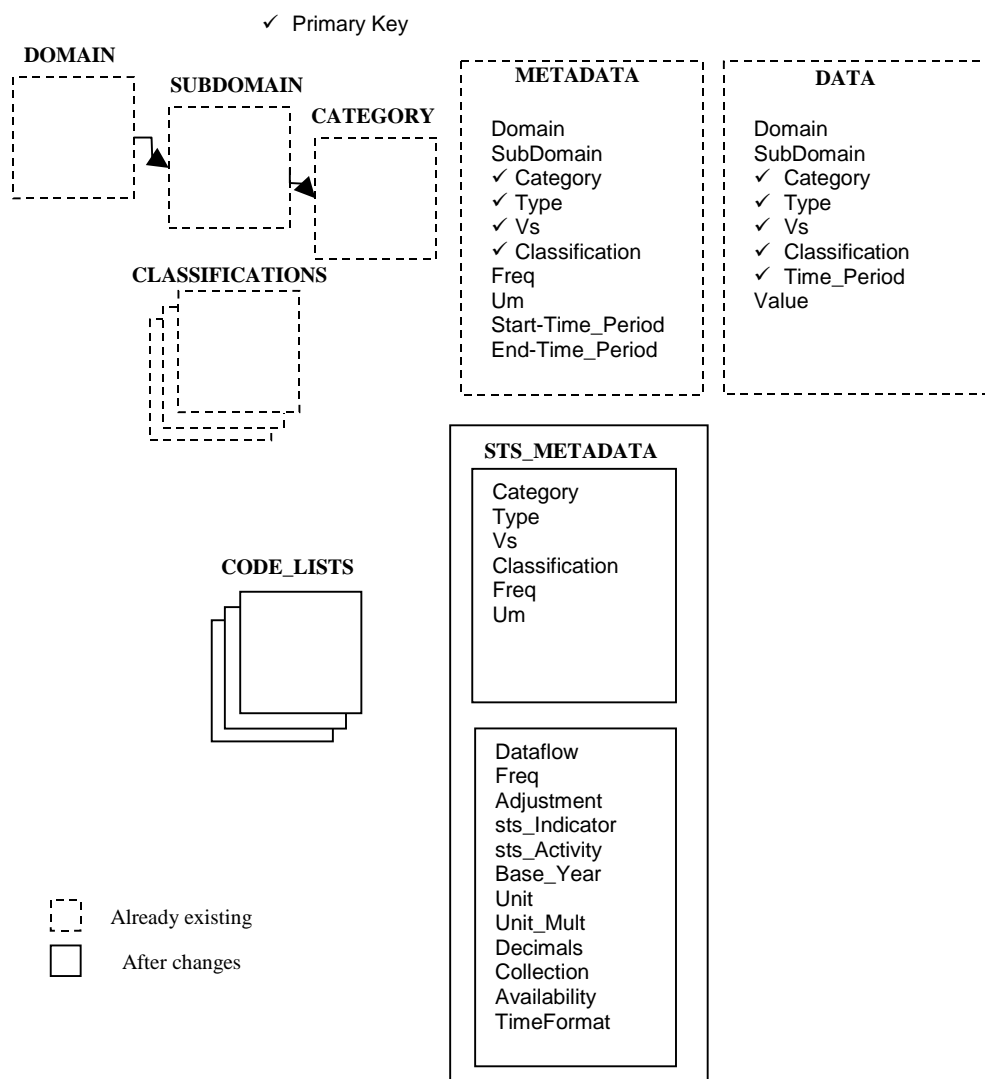


Figure B.7.4: Database schema with additional tables for mapping

In order to perform the mapping process correctly, it is necessary to consider different types of mapping: mapping of concepts and mapping of codes¹⁷.

Mapping of concepts

The first step is to identify all the statistical concepts involved in the exercise. The following circumstances can occur:

- 1) one concept in the DSD can be linked up with a single local concept. A typical example is the *measured value* in the data provider database that corresponds to the *Primary measure* in the STS DSD used for SODI;
- 2) one local concept must be linked up with two or more concepts in the DSD. For example in the local concept named Um there is an element as follows: “one million of Euro”. In the related STS DSD it corresponds to two concepts: Unit (Euro) and Unit multiple (one million);
- 3) one concept in the DSD is not directly linked up with any local concept. This could be the case of the concept “Reference area”, in fact that concept is generally not used in a National Organisation because it is the default (Italy);
- 4) one concept in DSD is linked up with two or more local concepts. For example the DSD concept “Adjustment” has no 1-to-1 correspondence with any single local concept; it is split into two different concepts “DAYADJ” (calendar adjusted) and “SEASADJ” (adjusted for periodical variations during the measurement period), which each has a Boolean value (true/false).

Mapping of codes

The second step is the mapping of the codes. Often a concept within a DSD can assume a code enumerated in a code list or a free value. The same thing can happen for a local concept. Assuming the concept used in the DSD and the local concept, used in the data provider's database, are both described using code lists, it may be possible to map one code in the first code list with a code in the second code list. The following example shows two such code lists:

- code list associated with the frequency local concept

CODE	DESCRIPTION
1	Annual
12	Monthly
365	Daily
4	Quarterly
52	Weekly

¹⁷ For further explanations of the usage of concepts and codes, see chapters XXXXXX.

- code list associated with the frequency concept in the code list used by the STS DSD

CODE	DESCRIPTION
A	Annual
M	Monthly
D	Daily
Q	Quarterly
W	Weekly
H	Half-yearly
B	Business

The mapping process will produce the following result:

DSD CODE	Local CODE	DESCRIPTION
A	1	Annual
M	12	Monthly
D	365	Daily
Q	4	Quarterly
W	52	Weekly
H		Half-yearly
B		Business

Often the map processing can be helped by some rules. For example, consider the CL_STS_ACTIVITY code list and the NACE Rev 1.1 classification. The rule is: remove all dots from the NACE code and add as many zeros as necessary in order to reach four digits. Then add the prefix N1, or NS in case of special codes.

After applying the above steps, the result of the mapping process in Conlstat can be set out as in Table B.7.1, in which columns represent both DSD concepts and local concepts, while rows represent a combination of their codes. The scheme shown here reflects the way in which the mapping tables are set up at ISTAT, which was chosen for performance reasons; the mapping table could be organised in other ways.

Table B.7.1: Mapping result example

CATE GORY	TYPE	CLASSI FICATION	FREQ	UM	DATAFLOW	STS_ INDICATOR	STS_ ACTIVITY	UNIT	BASE_ YEAR	ADJU STMENT	FREQUE NCY
18	G	DL300	12	PE	SSTSIND_ORD_M	ORDT	N13000	PURE_N UMB	2000	N	M
18	G	DL31	12	PE	SSTSIND_ORD_M	ORDT	N13100	PURE_N UMB	2000	N	M
18	G	DL311	12	PE	SSTSIND_ORD_M	ORDT	N13110	PURE_N UMB	2000	N	M

For example:

- the concept named CATEGORY that assumes the code 18 (Index of total orders), from the related local code list, is mapped with the concept named STS_INDICATOR that in the STS code list is represented by the code ORDT;
- the concept named TYPE that assumes the code G (neither seasonally or working day adjusted), from the related local code list, is mapped with the concept named ADJUSTMENT that in the STS code list is represented by the code N;
- the concept named FREQ that assumes the code 12 (Monthly), from the related local code list, is mapped with the concept named FREQUENCY that in the STS code list is represented by the code M;

- the concept named UM that assumes the code PE (index base=2000), from the related local code list, is mapped with the two concepts: UNIT that in the SDMX code list is represented by the code PURE_NUMB and BASE_YEAR that in the STS code list is represented by the code 2000;
- the concept named CLASSIFICATION that assumes the code DL300 (Manufacture of office machinery and computers), from the related local code list, is mapped with the concept STS_ACTIVITY that in the STS code list is represented by the code N13000.

B.7.7 From a data file to an SDMX data file

This solution is the simplest one and can be used only in the database-driven architecture where the data collector, driven by the RSS feed provided by the data producer, fetches the SDMX data file directly from the data provider's web server.

This solution is valid only in the database-driven architecture because the data collector can pull only entire data sets, and not sub-sets.

Generally this operation foresees a manual intervention every time an SDMX data file must be produced; in fact, the operator has to use an application that converts a source data file, for example in CSV or Excel format, into an SDMX data file. Moreover, the operator also has to produce the related the RSS file. Figure B.7.5 illustrates this solution:

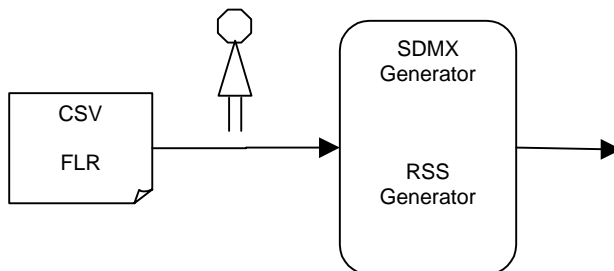


Figure B.7.5: Manual generation of SDMX data file

B.7.8 Disseminating SDMX data files starting from a database

This solution also is valid only within a database-driven architecture. The difference from the previous solution is the better automation of the workflow. In fact, the application that produces the SDMX data files and the RSS file can be started automatically. Logically this kind of solution requires more effort from a development point of view, but could eliminate or reduce the human intervention during the production phase. Figure B.7.6 illustrates this solution:

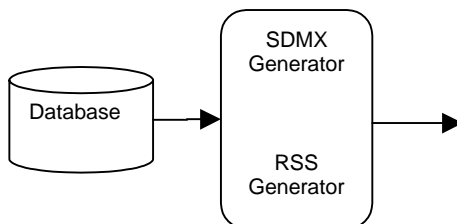


Figure B.7.6: Generation of SDMX data files starting from a database

B.7.9 An SDMX solution valid for both the database-driven and data hub architectures

This solution is the most advanced and cost-effective in operation, but the most costly in terms of development. This solution requires an architecture able to perform several functionalities. In particular data and metadata stored in a database are accessible directly by other systems. In order to simplify the meaning, this architecture can be separated in several building blocks, each of them performing some well defined functionalities. Figure B.7.7 illustrates this solution:

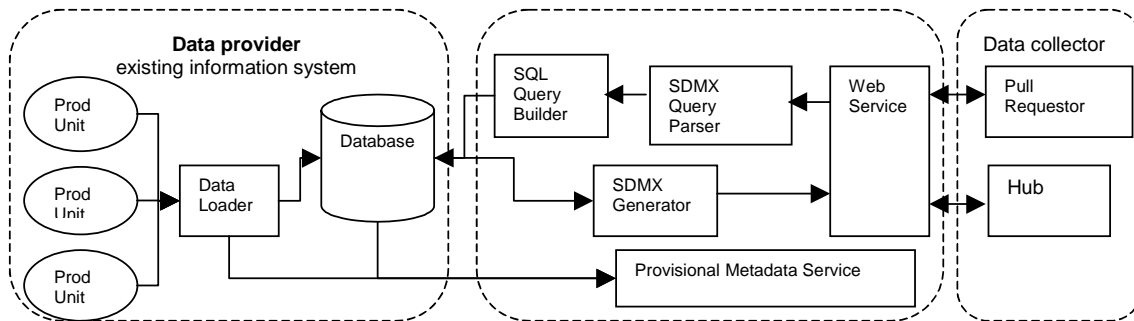


Figure B.7.7: A complete solution for database-driven and hub architectures

The Web Service

The Web Service represents the front-end for the data provider part of the SDMX architecture. It accepts SDMX-ML query messages and responds with SDMX-ML data or metadata messages. A client application knows how to interact with the Web Service through the WSDL information¹⁸.

Because SDMX offers a number of data formats (although it only requires one), and because it concerns itself both with data and with the structural metadata needed to understand and process that data, the SDMX web service is composed of a set of data exchanges. Thus, the SDMX web service implements a "multiple message exchange pattern" (in WSDL terminology) that directly encapsulates SDMX-ML messages. So just as the `GetCompactData(Query)` method accepts an SDMX-ML Query message through the Query parameter and responds with an SDMX-ML Compact message, the `GetCrossSectionalData(Query)` method respond with an SDMX-ML Cross-sectional message.

This module should implement also the handling of the SOAP errors and for simplicity could also implement the important functionality of validation of the incoming SDMX query message, which otherwise has to be done by another module. The validation is made to check the message syntactically and semantically, using the related XML schema (.xsd).

The SDMX Query Parser

The SDMX Query Parser interprets an SDMX query message and converts it into a data format easily usable from the next module. In general an SDMX query message

¹⁸ WSDL (Web Service Description Language) is an XML message where a service consumer can find information on the endpoint where the service resides, exposed operations and information to be passed as XML so that the service consumer is able to create requests and read responses for a service. Most of the SOAP frameworks available for most development platforms (.NET, Java, C/C++, Perl, etc.) have utilities that can dynamically generate classes for creating a client for the service.

contains a **header** element and a **where** element. The header element is equal for all kinds of SDMX query messages and could include information to identify the sender and the receiver. The **where** element could be different depending on what the user/application wants to query. For simplicity, this chapter takes into consideration only SDMX Query messages for querying data sets. In this case the starting node, regarding the **where** part of the query message, is represented by **DataWhere**.

The DataWhere node includes the following elements:

- DataSet
- KeyFamily
- Dimension
- Attribute
- Codelist
- Time
- Category
- Concept
- DataProvider
- DataFlow
- Version
- Or
- And

Most of the elements are composed of complex data types which make the SDMX Query message very complicated. Each of the above elements can appear 0 or 1 time in the **DataWhere** node, but only one of them. Moreover the two elements **Or** and **And** can include the above elements, but this time without any limit.

In general the **DataWhere** node and the minimum constraints such as **DataFlow**, **DataProvider** and **Time** are required. Moreover finer constraints on dimensions and attributes can be specified together with **AND** and **OR** operators.

The following example of an SDMX Query message asks for: two time series (the first is "Manufacture of machine tools" and the second is "Manufacture of office machinery and computers"), from ISTAT (IT1), neither seasonally or working day adjusted, from the New order received in industry – total" data set, for the period from Jan 2008 to Dec 2008:

```
<query:DataWhere>
  <query:And>
    <query:Dimension id="STS_INDICATOR">ORDT</query:
Dimension>
    <query:Dimension id="FREQ">M</query:Dimension>
    <query:Dimension id="ADJUSTMENT">N</query:Dimension>
    <query:Time>
      <query:StartTime>2008-01</query:StartTime>
      <query:EndTime>2008-12</query:EndTime>
    </query:Time>
    <query:DataProvider>IT1</query:DataProvider>
    <query>Dataflow>SSTSIND_ORD_M</query>Dataflow>
    <query:Or>
      <query:Dimension id="STS_ACTIVITY">
N12940</query:Dimension>
      <query:Dimension id="STS_ACTIVITY">
N13100</query:Dimension>
    </query:Or>
  </query:And >
</query:DataWhere>
```


The SDMX Query message is converted by this module into a format that can be used easily in order to build an SQL query that the database can understand. Table B.7.2 shows an example, where rows represent Or elements while columns represent all the others allowable elements (e.g; Dimension, Attribute, Time, DataProvider, DataFlow, etc.) in And relation between them. The schemes shown here reflect the way in which the tables are set up at ISTAT, which was chosen for performance reasons; the table could be organised in other ways.

Table B.7.2: Example of translated SDMX query

STS_INDICATOR	StartTime	EndTime	DataProvider	Dataflow	STS_ACTIVITY	FREQ	ADJUSTMENT
ORDT	2008-01	2008-12	IT1	SSTSIND_ORD_M	N12940	M	N
ORDT	2008-01	2008-12	IT1	SSTSIND_ORD_M	N13100	M	N

The SQL Query Builder

The SQL Query Builder module reads the result of the SDMX Query Parser module and converts the high level query containing SDMX terms to a native SQL query that will be able to be executed against the database. In order to build the SQL query, the builder has to know the schema of the database and how the SDMX structure metadata are mapped to the data stored in the database.

Taking into consideration the previous example, the related schema database and the table in Table B.7.1, the result of the SDMX Query Parser can be mapped as shown in Table B.7.3.

Table B.7.3: Example of result of SDMX Query Parser

Category	StartTime	EndTime	Classification	TYPE
18	2006-01	2006-12	DL300	G
18	2006-01	2006-12	DL31	G

The SQL that will be built is the following:

```

Select METADATA.*, DATA.*
From METADATA, DATA
Where
METADATA.CATEGORY = "18" And METADATA.CLASSIFICATION = "DL300" And
DATA.TIME Between 2006-01 And 2006-12 And METADATA.CATEGORY =
DATA.CATEGORY And METADATA.TYPE=DATA. TYPE And METADATA.VS =
DATA.VS And METADATA. CLASSIFICATION =DATA.CLASSIFICATION
Union
Select METADATI.*, DATI.*
From METADATI, DATI
Where
METADATI.CATEGORIA = "18" And METADATI.ATECO= "DL31" And DATI.TIME
Between 2006-01 And 2006-12 And METADATI. CATEGORY = DATA. CATEGORY
And METADATA.TYPE = DATA. TYPE And METADATA.VS = DATA.VS And
METADATA.CLASSIFICATION = DATA. CLASSIFICATION And METADATA.TYPE =
"G"

```

The SDMX Generator

This module collects the recordset which results from the SQL query and builds an SDMX-ML Data message. In order to do this, it uses the information within the related DSD and stored in the database during the mapping process.

The Provisioning Metadata Service

This module in its most simple extent allows to notify data consumers that a new or updated data set is available.

In general there are two ways to receive such notifications: the first is the subscription/notification mechanism, using SDMX-ML messages through an SDMX Registry (in this case the Provisioning Metadata Service has to provide a Provision agreement to an SDMX Registry). Another mechanism is the use of RSS feeds. In either case, the update can serve as a trigger for the receiving application – to go out and get the updated or new data set, or to perform some other automated process.

Benefits of this approach

This complete system is built on a set of separate modules each of which has well-defined functions and interfaces. This allows the modules to be built and tested independently, and the modules themselves can be re-used across domains and between organisations, which makes this a cost-effective approach to a full-scale SDMX implementation.

B.8 Building and operating an SDMX Registry

B.8.1 Scope of this chapter

It is intended that this chapter will explain technical issues relating to the implementation of an SDMX Registry.

For the time being, the reader is referred to the material and references in §A.7 Uses for an SDMX Registry, and to the documentation accompany the registry software which is available via the SDMX website (see §B.2 Obtaining and using SDMX Tools).

B.9 Data Structure Definitions: a tutorial

B.9.1 Scope of this chapter

This tutorial is intended to explain *data structure definitions* (DSDs) to those who are completely unfamiliar with the concept. Data structure definitions are an important part of the SDMX family of standards for exchanging statistical data, and they are modelled and explained in much greater detail in other documents. However, those documents are not written to explain the basics, and will make difficult reading for those new to the idea. This chapter provides a basic tutorial, to help provide the basic level of understanding needed to make sense of the SDMX standards.

A practical example for building a data structure is also provided in §B.10 Guidance on setting up Data Structure Definitions.

B.9.2 What is a data structure definition?

This chapter begins with the same example used in §A.3.2.

In order to answer this question, we need to look at statistical data. Statistical data are represented with numbers, such as:

17369

If you are presented with a number - as above - you will have no idea of what it actually represents. You know that it is a piece of statistical data, and therefore is a measurement of some phenomenon - also known as an "observation" - but you can't tell from the number alone what it is a measurement of. A number of questions come immediately to mind:

- What is the subject of the measurement?
- What units does it measure in?
- What country or geographical region, if any, does it apply to?
- When was the measurement made?

The list of questions is potentially endless. Behind each of these questions is a particular idea, or "concept", which is used to describe the data. In our questions above, these descriptor concepts are Subject, Unit of measure, Country, and Time. If I tell you the answers to these questions, the data will begin to make sense:

- the Subject is "total population"
- the Unit of measure is "thousands of people"
- the Country is "Country ABC"
- the Time is "1 January 2009"

This is a simplified and fictional example, but it does demonstrate how we can begin to make sense of statistical data with a set of descriptor concepts. We now know that our number represents the fact that the total population of Country ABC on 1 January, 2009, was 17,369,000.

The simplest explanation of a data structure definition is that it is a set of descriptor concepts, associated with a set of data, which allow us to understand what that data mean. There is more to it, however.

B.9.3 Grouping Data

Numbers are often grouped together in various ways, to serve as useful packages of information. One very common approach is to have a set of observations - known as a "series", or a "time series" - made over time. This allows us to see trends in the phenomenon being measured. Thus, if I measure the total population in Country ABC on 1 January of every year, I can see whether the population is growing or declining.

A time series always has a "frequency". This is a descriptor concept which describes the intervals of time between observations. Usually, this is a regular interval, so that the frequency can be expressed as "annual" or "monthly" or "weekly". Sometimes, the intervals are irregular. Notice that a single observation does not have a frequency - only series of observations have frequencies. Frequency is an example of a descriptor concept which only applies to series of data.

There are other, higher-level groupings of data as well. A number of series are often grouped together into a "Group". One such group of time series consists of a number of time series (i.e. known as a "Sibling Group") which are identical except that they are measured with different frequencies. Thus, a group of Series would be one given phenomenon measured as daily, monthly, and annually.

It is possible to have Groups which have variable values for descriptor concepts other than frequency, however: if I want to express the US daily exchange rate for all of the world's currencies over the past year, I have a different kind of group. All of the "frequency" descriptors would be the same - "daily" - but the descriptor concept which gives the "foreign currency" would be different for each series. This results in a two-dimensional table with the dimensions country and time, the concept measured being US daily exchange rate for each country's currency. There could be more than these two dimensions in the group, so the basic structure of (aggregated) statistical data is a multidimensional table (also called a cube), of which one of the dimensions can be time.

There is also a higher level of package known as a "Data Set". This represents a set of data that may be made up of several Groups. Typically, it is maintained and published by an agency, so that it becomes a known source of statistical data.

A basic structure is emerging: We have Observations, grouped into Series (if time is a dimension), which are grouped into Groups, i.e. multidimensional tables, which are grouped into Data Sets.

Note: It should be mentioned that there is another way of packaging Observations, which we call "cross-sectional" data. In cross-sectional data, a number of related Observations are presented for a single point or period in time and for a single member of each of the other dimensions. This organization of data is very similar to Time Series data in the way a set of descriptor concepts can be associated with it. A data structure definition can be used to describe both cross-sectional, time series data and multidimensional table data. For the purposes of this part of the tutorial, however, we will focus on time series data. Once we have described the data structure definition for time series data, we will go back and see how cross-sectional data are structured.

What is a data structure definition? (Answer #1)

A data structure definition is a way of associating a set of descriptor concepts with a specific set of statistical data, as well as a technique for packaging or structuring that set of data into groups and sub-groups. This is only one way of understanding the structure and meaning of statistical data, but it provides us with a solid, generic model.

B.9.4 Attachment Levels

Some descriptor concepts are not meaningful at the Observation level, but only at a higher level. The example we saw earlier was frequency, which means nothing for a single Observation, but has meaning when applied to a Series of Observations. This is because it represents the interval of time between Observations. Time, on the other hand, is meaningful at the Observation level - every Observation is associated with a specific point or period in Time ¹⁹. Data structure definitions provide information about the level at which a particular descriptor concept or dimension is attached: at the Observation level, the Series level, the Group level, or the Data Set level. This is known as the "attachment level" of the descriptor concept.

If we think about Groups, particularly, we can see how this works. Within a group, some descriptor concepts have values that are the same for all Series within the Group, while other descriptor concepts are changeable. For the Group described above, of all US exchange rates measured daily for all of the world's currencies, the descriptor concepts of Subject ("US exchange rate") and Frequency ("daily") will be the same for all members of the Group. The descriptor concept "Currency", however, will change for each Series within the group: there will be a Series for "Swiss Francs," a Series for the "Euro," a Series for "New Zealand dollars," etc.

The rule is that descriptor concepts are "attached" to the grouping level where they become variable. Thus, if, within a single set of data, all the contents of a Series share a single value for a descriptor concept, then that descriptor concept should be attached at the Series level. This rule also assumes that the chosen level is the highest structural level where all sub-groups will share the same value.

Attachment levels of descriptor concepts are always at least at the level where the concept is meaningful: thus, you cannot attach the descriptor concept frequency at the Observation level, because as a concept it only operates at the level of Series (that is, with multiple Observations made over time).

B.9.5 Keys

"Key" refers to the values for the descriptor concepts which describe and *identify* a particular set of data. Let's continue our initial simple example:

I have a set of statistical data which uses the following descriptor concepts:

- Time
- Frequency
- Topic
- Country

Time is always attached at the Observation level - the value for Time is the time at which the Observation was made. For time series data, Time, which is a concept connected to all statistical data, does not form part of the key. The other descriptor concepts - frequency, topic, and country - are all attached at the series level. For any given Series of Observations, they will all have a single value.

If we have a Series of data which is the annual measurement of the total population of Country ABC, we will have a key made up of the following values for each descriptor concept:

¹⁹ However, not all statistical data sets can be perceived as time series. Time may not be relevant to all observations, or there may be observations for only one time period or point in time

Frequency = "annual"
Topic = "total population"
Country = "Country ABC"

This set of values - "Monthly - total population - Country ABC" is the "key" for this data Series: it identifies what the data is.

Keys are most often associated with data at the Series level, but they also exist at other levels. For example, we could enlarge our example to be a Group including the annual total population data for all of the countries in the world. At the Group level, Frequency would have a value of "annual", and Topic would have a value of "total population", but we would not specify the Country descriptor concept, because it would change from Series to Series. The key for the Group is known as a "Group Key" - it identifies what the Group is, rather than identifying the Series. (In order to completely understand the Group, of course, we also need to know which descriptor concepts are changeable - in this case, Country.)

The key values are attached at the Series level, and are given in a fixed sequence. Frequency is the first descriptor concept, and the other concepts are assigned an order for that particular data set. This makes it much easier to share and understand statistical data.

B.9.6 Attributes

If you look back to our initial use of this example, you will notice that we have not been discussing the "Unit of measure" descriptor concept when defining the key. This is because the "key" only contains values for those descriptor concepts which identify the data. If we have the measurements made in thousands or in millions, the data are the same - they can be derived from one another by simply multiplying the numbers in the data by the appropriate conversion factor.

This points out a major distinction between the two types of descriptor concepts: the ones which both *identify* and describe the data are called "dimensions", and those which are *purely descriptive* are called "attributes". Only "dimensions" - that is, the descriptor concepts which also identify the data - are used in the "key", because the "key" is fundamentally a way of identifying a set of data.

B.9.7 Code lists and other representations

In order to be able to exchange and understand data, a data structure definition tells us what the possible values for each dimension are. This list of possible values is known as a "code list". Each value on that list is given a language-independent abbreviation - a "code" - and a language-specific description. This helps us avoid problems of translation in describing our data: the code can be translated into descriptions in any language without having to change the code associated with the data itself. Wherever possible, the values for code lists are taken from international standards, such as those provided by ISO for countries and currencies.

As stated, dimensions are always represented with codes. Attributes are sometimes represented with codes, but sometimes represented by numeric or free-text values. This is allowed because the attributes do not serve an identification function, but merely describe the data.

What is a data structure definition? (Answer #2)

We now have a more sophisticated understanding of what a data structure definition does: it specifies a set of concepts which describe and identify a set of data. It tells us which concepts are dimensions (identification and description), and which are attributes (just description), and it gives us an attachment level for each of these concepts, based on the packaging structure (Data Set, Group, Series, Observation). It also tells us which code lists provide possible values for the dimensions, and gives us the possible values for the attributes, either as code lists or as numeric or free text fields.

B.9.8 Cross-sectional data structures

Given the explanation of data structure definitions thus far, we understand that a data structure definition associates descriptor concepts with data, some of which also serve to identify the data – the “dimension” concepts which make up the Key.

Cross-sectional data structures do not apply a different set of concepts to the data: the same concepts still apply in describing and identifying the data. It attaches the concepts to the data differently, to create a different presentation of the data.

If we go back to our earlier example, we had the following concepts or dimensions:

- Time
- Frequency
- Topic
- Country

If we want to take a set of data which is described and identified by this set of concepts, and present it in a cross-sectional fashion, we would not change these concepts – we would merely change the way in which they are represented – that is, attached – to the data structure.

Take, as an example, the total population of each country in the world on January 1, 2001 as a set of data. In our earlier example, we measured the population of Country ABC over a period of years – that is, over time. Time was the concept we used to organize our data in a sequence of observations.

If we organize our data to reflect only a single point in time – in this case, January 1, 2001 – then organizing our data over time makes less sense. It is still a possible way to structure the data, but we may wish to view it as a cross-section.

Think about the term “cross-section” – it can be understood to mean a group of parallel series over time, from which a section is taken, across time. Thus, a cross-section is created.

In our example, it is easy to see how this applies: instead of organizing our data over time – that is, using the time concept - we are choosing to organize it over the Country concept. Thus, instead of having a single value for Frequency, Topic, and Country for all Observations in our series, with a Time value associated with each Observation, we will have a Country value associated with each Observation, and a single value for Frequency, Topic, and Time. Instead of calling the group of Observations a “Series”, we now use the term “Section”.

In our first example, we had a key which existed mostly at the Series level:

Frequency	=	"annual"
Topic	=	"total population"
Country	=	"Country ABC"

Time – our remaining concept, was associated with the Observations, with a different value for each one. Thus, we could have a Series which looks like this:

January 1, 2001	–	17369
January 1, 2002	–	17370
January 1, 2003	–	17405

For our cross-sectional presentation, we would have most of our key at the Section level (or, potentially, at a higher level of grouping):

Frequency	=	"annual"
Topic	=	"total population"
Time	=	"January 1, 2001"

With each Observation, we now have a Country value, instead of a Time value:

Country ABC	=	"17369"
Country XYZ	=	"24982"
Country HIJ	=	"37260"

In this cross-sectional presentation of our data set, we have chosen to present each Observation paired with a Country value, taken from our Code list of values for the concept Country. Other dimensions could as easily produce a cross-sectional view, by attaching their values at The Observation level, instead of the values for Country, as in our example.

Because the concepts themselves do not change, but only the way in which they are attached to the data structure, a single data structure definition can be used to describe both time-series and cross-sectional presentations.

In the SDMX standards, formats are capable of presenting cross-sectional data for any single dimension concept, as well as presenting the data as a time series. It is up to the developer of the data structure definition to select which non-Time concept, used as a dimension, will serve to organize a cross-sectional presentation. In future versions, it is possible that more complete support for the possible cross-sectional views for a data structure definition will be provided.

What are the SDMX terms?

Data Structure Definition: set of structural metadata associated to a data set, which include information about how concepts are associated with the measures, dimensions, and attributes of a data cube, along with information about the representation of data and related descriptive metadata. The term *data structure definition* is synonymous with the term *key family*.

Statistical concept: A statistical characteristic of a time series or an observation. It can be coded (= taking values from a code list) or uncoded. In the generic description above the term *descriptor concept* was used.

Time series: a set of ordered observations on a quantitative characteristic of an individual or collective phenomenon taken at different points of time.

Observation: the value, at a particular period, of a particular variable.

Group of sibling time series (or sibling group): a set of time series whose keys differ only in the value taken by the frequency dimension: the group of time series referring to exactly the same economic phenomenon as expressed in different frequencies

Dimension: a statistical concept used, in combination with other statistical concepts, to identify subsets of a multidimensional table (cube) or single observations in the multidimensional table (examples of dimensions: frequency, economic phenomenon, reference area). A dimension has a definite number of values, called dimension members.

Key: set of values taken by the dimensions in a multidimensional table (cube) describing identifying a specific series

Attribute: statistical concept qualifying observations, time series or datasets (e.g. title, availability, observation status, unit of measure, title in national language, source). In general terms, the attribute is a characteristic of an object or entity.

B.10 Guidance on setting up Data Structure Definitions

B.10.1 Scope of this chapter

This chapter provides some general principles that may guide the development of data structure definitions and also a practical example. It is assumed that the reader is familiar with idea of a data structure and has, in particular, read chapter A.3 of the User Guide. The principles relate to SDMX-specific issues such as the choice of dimensions, attachment levels and code lists as well as to organisational issues, such as institutional involvement and consultations.

B.10.2 Choice of Dimensions and Attributes

It is preferable to include in the list of dimensions of the DSD only the concepts that are absolutely essential to identify the data/time series (and to distinguish them from each other); and to define other relevant concepts (for this data flow) as attributes of the DSD. The SDMX Cross-Domain Concepts should be applied wherever possible.

For the dimensions, statisticians tend to prefer “clean” as opposed to “mixed” concepts (something that may contribute to an increase in the number of dimensions) and, thus, specifying relatively simple corresponding code lists (not mixing up concepts). The ability of the DSD to address potential new and other future requirements also, usually, requires the use of “more” than “fewer” dimensions. The appropriate choice of “dimensions” needs to take into account conflicting requirements to balance the associated trade offs in the wish for maximising:

- Simplicity and purity of the statistical concepts chosen to be used as “dimensions”;
- Flexibility and ability to respond to changing and new requirements;
- Possibility to address in a homogeneous way more than one stage of the data life cycle;
- In some situations, usability (short and manageable “keys” for the users).

B.10.3 Principles for deciding the order of dimensions in the data structure

Following a specific order in defining dimensions is of particular usefulness for end-users. Thus, gradually, they become familiar with the same type of identifying data, regardless of the domain and the DSD maintenance agency that provided and supports the corresponding DSD.

There are a set of dimensions that are likely to be common across different data sets both within an organisation and across organisations, however, they are likely to differ between statistical domains. While the concept of “Maturity” may be used in DSDs relating to financial data, it is unlikely to be used in, for example, labour market statistics. Hence it will again be a balancing act between potentially conflicting requirements. A possible rule could be to put those concepts first, that are likely to be used across a wide range of DSDs, e.g. those that are part of the SDMX Cross-Domain concepts.

B.10.4 Code lists

In order of priority, for each *coded* dimension and attribute, it is suggested to use (in order of priority):

- 1) Code lists as proposed in the SDMX Content-Oriented Guidelines (Annex 2 – Cross-Domain Code Lists, available on the Guidelines page of the SDMX website);
- 2) ISO code lists;
- 3) Code lists used by several supranational organisations;
- 4) Standardised code lists at a regional level (e.g. Europe, America)
- 5) Standardised code lists at the national level
- 6) Institutional-wide code lists
- 7) Departmental code lists

B.10.5 Change management

Ideally, the DSDs should also be designed in a way that would be somehow forward looking and optimal in a, let's say, also (at least) 3-5 year horizon. Frequent future changes in DSDs may have implications on costs for adjusting systems, applications and users' programs. That is why frequent changes should be avoided, as these would have an impact on the benefits enjoyed from using the standards. A typical particular example is the procedure for deciding the dimensions to be included in a DSD: redundant dimensions (not useful in identifying the observed phenomenon and distinguishing it from other phenomena) should be avoided, but also future needs (that might lead to a need for additional dimensions in future) should not be ignored.

When setting up a new or changing an existing DSD, sufficient lead time for implementation (and, possibly, testing) should be given.

B.10.6 DSDs and data life cycle

Ideally, the needs and requirements of both producers *and* users of statistics should be taken into account. Thus, for a particular data flow, the *data identification* and *metadata* requirements related to the collection, storage and dissemination should be carefully reviewed.

There may be cases, however, in which the data collection requires more dimensions than the ones needed at a higher level aggregation targeted for use in web dissemination. In such cases, there may be a need for two DSDs that may differ in the two phases (collection, dissemination for end users). In general, such differences in DSDs are better to be avoided, but, if unavoidable, then there should be an effort to maximise the number of common elements, e.g. dimensions, code lists, etc.

B.10.7 Organisational issues

When there is a requirement for a new DSD, the broadest possible consultations (within and across organisations) should take place, taking into account needs and requirements of statistical offices, central banks and international organisations. A quick search, on what is already available and in use, is strongly recommended even if a national or local implementation is targeted.

With respect the appropriate choice of the “maintenance agency” of the DSD, the following options are available and in use:

- The organisation having the lead in defining a DSD would also probably be the maintenance agency *for technically maintaining* the corresponding DSD.
- Organisations sharing the use of a given DSD appoint one among them to act as the maintenance agency.
- A DSD may be technically maintained by an agency which, practically, may not be the agency administering all code lists used.

In general, the following considerations apply

- the *more locally* a maintenance agency for a DSD is specified, the more flexibility exists for making changes in a DSD (however, with somewhat higher “reading”-interpreting-mapping costs for “translating” such data towards the more global representation methods used, if they exist for similar data, in other areas/regions);
- and *the more globally* a maintenance agency for a DSD is specified, the more global harmonisation is achieved for the benefit of users and of the overall statistical community (however, with somewhat higher consultation costs and a longer lead time for discussing and agreeing on possible changes that might be needed; these should be more globally discussed, planned, accepted and implemented).

When defining or changing a DSD of broader use, consultations need to take place not only across organisations, but also within institutions and, in some cases, possibly also among experts from different statistical domains due to potential interactions and overlapping requirements (e.g. a DSD for national accounts has an interest also for external or governmental statistics). Thus, another parallel desirable objective could be pursued, i.e. maximising the reusability of concepts and code lists across domains.

C FREQUENTLY ASKED QUESTIONS

C.1 What are the differences between SDMX Version 1.0 and Version 2.0

The SDMX Technical Specifications Version 2.0 expanded the scope of the standard considerably. They provide for the formatting and exchange of reference metadata and present a model and formats for use in working with the statistical exchange process itself. This includes a standard registry/repository specification for the coordination of statistical exchanges across a network of users. SDMX-ML was expanded to include interfaces for interacting with the registry/repository.

See §B.1.2 Technical standards: from Version 1.0 to Version 2.0

C.2 What is a key family?

See §C.3.

C.3 What is a data structure definition?

A data structure definition (previously known as a “key family”) is a set of structural metadata which describes a multidimensional statistical data set. It includes information about which concepts are associated with the data, and how they are represented using code lists. Further, attached “footnote” metadata (termed “attributes”) are also described.

The data structure definition is used in SDMX to configure all of the data exchange messages, in both XML and EDIFACT syntax, and can itself be exchanged in these formats. It provides a rich set of metadata, associated with SDMX data sets, which can be used by generic statistical applications to understand and process specific data sets. This can help reduce the cost of adding support for new data sets to existing applications.

See §A.3 The SDMX information model: Data Structures

C.4 Are tools available to help build a data structure definition? To do other things with SDMX?

Many IT tools are being distributed free of charge to support the use of SDMX.

Tools exist for the creation of data structure definitions, for working with SDMX formats, for visualizing SDMX data and metadata, and for implementing SDMX Registries.

See B.2 Obtaining and using SDMX Tools and the Tools page of the SDMX website.

C.5 What is the difference between structural and reference metadata?

Structural metadata act as identifiers and descriptors of the data.

Reference metadata describe the contents and the quality of the statistical data. They are sometimes called explanatory metadata.

See §A.1 What is SDMX?, §A.3 The SDMX information model: Data Structures and §A.4 The SDMX Information Model: Metadata Structures.

C.6 How and when will organizations implement SDMX?

Many organizations are already implementing SDMX, including all of the SDMX sponsors and many central banks and national statistical organizations. There has been discussion of the role SDMX will play within the international statistical system, and several organizations are moving forward with implementation. Many institutions developed GESMES/TS implementations, which are now technically SDMX-EDI implementers (SDMX-EDI being backward-compatible with SDMX).

Additionally, some of the vendors and organizations who produce software packages for working with statistics are also starting to provide support for SDMX standards, so that adoption by organizations which use these tools will be greatly facilitated.

The specific use of SDMX ranges from the use of XML and EDIFACT data and metadata formats to the implementation of SDMX-conformant registry/repositories as the basis for statistical exchanges between several counterparties.

See the *Implementations* pages of the SDMX website.

C.7 What is an SDMX Registry?

An SDMX Registry is an IT application providing information on structural metadata (data structure definitions, code lists, etc.) and on the location of data sets and reference metadata sets. In support of this second function, it also can house metadata regarding release calendars, the protocols needed to obtain specific data sets and reference metadata sets, and administrative metadata which supports statistical exchange processes.

An SDMX Registry is in fact a data and reference metadata registry, and a repository for structural and process-oriented metadata; thus, it can also be referred to as an SDMX registry/repository.

See §A.7 Uses for an SDMX Registry

C.8 Is there a single central SDMX Registry?

No, there is no one central SDMX Registry – it is a generic mechanism, made available as a free tool, which can be operated on behalf of a network of counterparties within a statistical community. An SDMX Registry functions in a way which allows many registries to be used by a single application – thus, the fact that SDMX registries are implemented within specific statistical communities still supports the exchange of data and metadata across domain boundaries.

See §A.7 Uses for an SDMX Registry.

C.9 What is the Metadata Common Vocabulary?

The Metadata Common Vocabulary (MCV) is part of the Content-Oriented Guidelines. Based on several earlier works, the MCV provides ISO-compliant definitions for many different statistical terms. These include many concepts and terms which are important to the use of SDMX and to statistical interoperability more broadly.

See §C.10 What are the SDMX Content-Oriented Guidelines?

C.10 What are the SDMX Content-Oriented Guidelines?

The SDMX Content-Oriented Guidelines are a set of recommendations regarding the concepts, terminology, classifications, and code lists which are common across many domains of statistics. To support statistical interoperability, it is a good idea if the content areas which are similar across domains be standardized. Thus, while each domain will have specific concepts and terms which are germane only to that domain, there will also be many “standard” concepts (country, currency, time, etc.) which could be the same everywhere.

The focus of the Content-Oriented Guidelines is to recommend a set of these common standards, for use by everyone who uses the SDMX technical standards. This is not required for the use of the technical standards, but it will facilitate interoperability and help in the definition of data and metadata structures.

There are many references to the Content-Oriented Guidelines throughout the User Guide, as they underly every aspect of the use of SDMX.

The latest version of the Content-Oriented Guidelines can be found on the Guidelines page of the SDMX website.

C.11 How can I use existing code lists to help me develop a data structure definition?

Data structure definitions (and metadata structure definitions) include the code lists which are used in the data formats. If there are code lists which exist inside statistical applications today, these form the basis of the code lists found in the data structure definition. Sometimes this is as easy as simply taking existing code lists and representing them in SDMX-ML. Sometimes the existing code lists need to be analyzed and cleaned up before inclusion. This depends on how the existing code lists are structured.

It should be noted that SDMX technical standards version 2.0 also supports hierarchical code lists.

C.12 If more than one code list exists for similar information, how do I find where they overlap?

It is often the case, even within a single organization, that several similar code lists are used to perform similar functions for different data sets. In version 2.0 of the SDMX technical standards, formats exist for expressing the relationships between two code lists. Thus, it is possible not only to exchange data with a counterparty, but also the metadata about how two similar-but-different code lists are related.

Human analysis is always required to determine the equivalence of the codes in two different lists, but SDMX provides a way of capturing this analysis so that it is not lost, and can be exchanged with counterparties and leveraged by applications.

C.13 Is SDMX multilingual?

SDMX-ML is designed to support multiple languages. Within the DSD, most possible values are provided as codes. Each code can be given multiple labels, each in a different language. Applications can refer to the labels to perform automatic translations on data displays.

Any fields within SDMX-ML which are language-specific allow for parallel language versions to be supplied. Thus, if an attribute contains plain text, it is possible to provide that text in several languages. While there is nothing in the SDMX-ML schemas which requires the provision of multiple language versions, this is a possibility, and can be agreed with counterparties who are supplying data.

Languages within SDMX-ML are identified using the normal XML mechanism - that is, with the `xml:lang` attribute carrying language and locale codes.

Those working with SDMX-ML should be aware that all XML applications are required to support the UTF-8 form of Unicode. While other character encodings may be useful, UTF-8 is assumed, as it provides support for many different character sets.