FRAMEWORK FOR SDMX STANDARDS

(VERSION 1.0)

Initial Release September 2004
First Revision December 2004

http://www.sdmx.org/

I. INTRODUCTION

The Statistical Data and Metadata Exchange (SDMX) initiative (http://www.sdmx.org) sets standards that can facilitate the exchange of statistical data and metadata using modern information technology, with an emphasis on aggregated data.

There are several sections to this technical specification:

1. The SDMX Information Model - the information model on which syntax-specific implementations described in the other sections are based. This document includes as appendixes a UML tutorial and a tutorial for those who are unfamiliar with key families as a way of describing statistical data structures. This document is not normative.
2. SDMX-EDI - the EDIFACT format for exchange of SDMX-structured data and metadata. This document contains normative sections describing the use of the UN/EDIFACT syntax in SDMX messages.
3. SDMX-ML - the XML format for the exchange of SDMX-structured data and metadata. This document has normative sections describing the use of the XML syntax in SDMX messages, and is accompanied by a set of XML schemas and sample XML document instances.
4. The SDMX Format Implementor's Guide – this is a guide to help those who wish to use the SDMX format specifications. It includes reference material for the use of the SDMX Information Model; a section describing the expressive differences of the various messages and syntaxes; and provides some best practices for assigning identifiers and designing key families. This document is not normative.
5. Web Services Guidelines – this is a guide for those who wish to implement SDMX using web-services technologies. It places an emphasis on those aspects of web-services technologies which will work regardless of the development environment or platform used to create the web services, and recommends the use of the WS-I version 1.0 specification.

II. PROCESSES AND BUSINESS SCOPE

A. Process Patterns

SDMX identifies three basic process patterns regarding the exchange of statistical data and metadata. These can be described as follows:

1. *Bilateral exchange:* All aspects of the exchange process are agreed between counterparties, including the mechanism for exchange of data and metadata, the formats, the frequency or schedule, and the mode used for

communications regarding the exchange. This is perhaps the most common process pattern.

2. *Gateway exchange:* Gateway exchanges are an organized set of bilateral exchanges, in which several data and metadata sending organizations or individuals agree to exchange the collected information with each other in a single, known format, and according to a single, known process. This pattern has the effect of reducing the burden of managing multiple bilateral exchanges (in data and metadata collection) across the sharing organizations/individuals. This is also a very common process pattern in the statistical area, where communities of institutions agree on ways to gain efficiencies within the scope of their collective responsibilities.
3. *Data-sharing exchange:* Open, freely available data formats and process patterns are known and standard. Thus, any organization or individual can use any counterparty's data and metadata (assuming they are permitted access to it). This model requires no bilateral agreement, but only requires that data and metadata providers and consumers adhere to the standards.

This document specifies the SDMX standards designed to facilitate exchanges based on any of these process patterns, and shows how SDMX offers advantages in all cases. It is possible to agree bilaterally to use a standard format (such as SDMX-EDI or SDMX-ML); it is possible for data senders in a gateway process to use a standard format for data exchange with each other, or with any data providers who agree to do so; it is certainly possible to agree to use the SDMX standards to support a common data-sharing process of exchange.

The standards specified here do not by themselves support a data-sharing process, but they provide the basic components of such a process. In the future, a more complete set of standards concerning the technology infrastructure for supporting a data-sharing process is expected be offered.

It is important to note that SDMX is primarily focused on the exchange of statistical data and metadata between providers and consumers. There may also be many uses for the standard model and formats specified here in the context of internal processing of data that are not concerned with the exchange between organizations and users. It is felt that a clear, standard formatting of data and metadata for the purposes of exchange can support any number of specific internal formats to facilitate processing by organizations and users.

B. Statistical Data and Metadata

To avoid confusion about which "data" and "metadata" are the intended content of the SDMX formats specified here, a statement of scope is offered. Statistical "data" are sets of numeric observations which typically have time associated with them. They are associated with a set of metadata “values” representing specific concepts, which act as identifiers and descriptors of the data. These metadata can be understood as the named dimensions of a multi-dimensional table, describing what is often called a "cube" of data.

SDMX identifies a standard technique for modelling, expressing, and understanding the structure of this multi-dimensional "cube", allowing automated processing of data from a variety of sources. This approach is widely applicable across types of data and attempts to provide the simplest and most easily comprehensible technique that will support the exchange of this broad set of data and metadata.

The SDMX standards offer a common model and formats which support the exchange of any type of statistical data meeting the definition above; an attempt has been made to optimize formats based on the specific requirements of each implementation, as described below in the SDMX-ML section.

The term "metadata" is very broad indeed, and the current version of SDMX is not designed to model or format every type of metadata. A distinction can be made between “structural” metadata – those concepts used in the description, identification and retrieval of statistical data – and “reference” metadata – that describe statistical concepts, methodologies for the generation of data and information on data quality.

Reference metadata, sometimes generated, collected or disseminated separately from the data to which they refer can be relevant to all instances of data described: entire collections of data, data sets from a given country, or for a data item concerning one country and one year. For this reason, some overlap may exist between "reference" metadata – which are often disseminated separately from the data they refer – and "structural" metadata used to identify data.

The present SDMX specifications are concerned with the structural metadata needed to identify, use, and process the data "cubes" described above. Future releases of the SDMX specifications are envisaged to address the exchange of reference metadata in more detail, as the general scope of the SDMX-ML format is to enable the exchange of structured data and metadata needed by users. In this version 1.0, the term "metadata" is used narrowly, to describe what can be termed "structural" metadata.

It should be noted that these specifications are not intended to cover all possible views of data "cubes". A "cube" is a rich, multi-dimensional construct, which can be viewed along any of its axis (or "dimensions"). SDMX takes a slightly narrower view of these requirements in its version 1.0 specifications. The view of data in these SDMX formats is primarily as time series – that is, as a set of observations which are organized around the time dimension, so that each observation occurs progressively through time. This is a clear way of organizing statistical data of many types, and has been proven to be a useful way of organizing data for exchange between counterparties.

There are, however, some types of statistical data which are not typically organized for exchange in this way – what we term "cross-sectional" data. SDMX provides some support for cross-sectional views of data cubes. In the 1.0 version of these standards, it is assumed that most data will be structured as time series. It is possible to describe a view such that a dimension other than time is used as the organizing dimension. If this type of a data structure is described – and if it has time as a dimension within the cube - then that data will be expressible in SDMX formats which are either organized along the chosen non-time dimension, or along time as a dimension. This approach gives time-series-based systems the ability to process many cross-sectional data sets as well as time series. In future versions of these specifications, it is intended that more flexible support for data cubes will be provided.

Another type of structure commonly found in statistical "cubes" of data is the use of hierarchical classifications to describe the points along any of its dimensions (or axes). In the 1.0 version, SDMX standards do not provide full support for this functionality. The introduction of these hierarchical classifications is anticipated in future versions of the standards.

C. SDMX and Process Automation

Statistical data and metadata exchanges employ many different automated processes, but some are of more general interest than others. There are some common information technologies that are nearly ubiquitous within information systems today. SDMX aims to provide standards that are most useful for these automated processes and technologies.

Briefly, these can be described as:

1. *Batch Exchange of Data and Metadata:* The transmission of whole or partial databases between counterparties, including incremental updating.
2. *Provision of Easily Processible Data and Metadata on the Internet:* Internet technology - including its use in private or semi-private TCP/IP networks - is extremely common. This technology includes XML and web services as primary mechanisms for automating data and metadata provision, as well as the more traditional static HTML and database-driven publishing.
3. *Generic Processes:* While many applications and processes are specific to some set of data and metadata, other types of automated services and processes are designed to handle any type of statistical data and metadata whatsoever. This is particularly true in cases where portal sites and data feeds are made available on the Internet.
4. *Presentation and Transformation of Data:* In order to make data and metadata useful to consumers, they must support automated processes that transform them into application-specific processing formats, other standard formats, and presentational formats. Although not strictly an aspect of exchange, this type of automated processing represents a set of requirements that must be supported if the information exchange between counterparties is itself to be supported.

The SDMX standards specified here are designed to support the requirements of all of these automation processes and technologies.

III. THE SDMX INFORMATION MODEL

SDMX provides a technique for modelling statistical data, and defines the set of metadata constructs used for this purpose. Because SDMX specifies formats in two syntaxes for expressing data and structural metadata, the model is used as a mechanism for guaranteeing that transformation between the different formats are lossless. All of the formats are syntax-bound expressions of the common information model. SDMX has based itself on GESMES/TS as an input to the model and formats, both to build on the proven success of this model for time series data exchange, and to ensure backward compatibility with existing GESMES/TS-based systems.

SDMX recognizes that statistical data is structured; in SDMX this structure is termed a "key family". "Data sets" are made up of one or more lower-level "groups", based on their degrees of similarity. Each group is in turn comprised of one or more "series" of data (or "sections" for non-time-series data). Each series or section has a "key" - values for each of a cluster of concepts, also called "dimensions" - which identifies it, and one or more "observations", which typically combine the time of the observation, and the value of the observation (e.g., measurement). Additionally, metadata may be attached at any level of this structure as descriptive "attributes". Code lists (enumerations) and other patterns for representation of data and metadata are also discussed, where they are representable across syntax-specific formats.

There is some similarity between "cube" structures commonly used to process statistical data, and the "key family" idea in the SDMX Information Model. It is important to note that the data as structured according to the SDMX Information Model is optimized for exchange, potentially with partners who have no ability to process a "cube" of data coming from complex statistical systems. SDMX time series can be understood as "slices" of the cube. Such a slice is identified by its key. A key consists of the values for all dimensions foreseen by the key family except time. It is certainly possible to reconstruct data cubes from SDMX-structured data, and to exchange such databases according to the proposed standards.

The SDMX Information Model is presented using UML, and is also described in prose. While the information model is not normative, it is a valuable tool for understanding and using the normative format specifications.

IV. SDMX-EDI

The SDMX-EDI format is drawn from the GESMES/TS version 3.0 implementation guide, as published as a standard of the SDMX initiative.

1. *Structure Definition:* All SDMX-EDI express the data and structural metadata covered by the SDMX information model in a UN/EDIFACT format.
2. *Compact Data:* Optimized for the batch exchange of large amounts of time series data between counterparties, it allows for extremely compact expression of large whole or partial data sets, and the structural metadata needed to understand them. Non time series data, such as cross-sectional, can be supported if represented as repackaged time series.

The SDMX Information Model provides the constructs which are found in the EDIFACT syntax used for SDMX-EDI, and those found in the XML syntax of SDMX-ML. Since both syntactic implementations reflect the same logical constructs, SDMX-EDI data and metadata messages can be transformed into corresponding SDMX-ML formats, and vice-versa. Thus, these standards provide for interoperability between the EDIFACT- and XML-based systems processing and exchanging statistical data and metadata.

V. SDMX-ML

While the SDMX-EDI format is primarily designed to support batch exchange, SDMX-ML must support a wider range of requirements. XML formats are used for many different types of automated processing, and thus must support more varied processing scenarios. That is why there are several types of messages available as SDMX-ML formats. Each is suited to support a specific set of processing requirements.

1. *Structure Definition:* All SDMX-ML message types share a common XML expression of the metadata needed to understand and process a data set.
2. *Generic Data:* All statistical data expressible in SDMX-ML can be marked up according to this data format, in agreement with the contents of a Structure

Definition message. 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. Data marked up in this format are not particularly compact, but they make easily available all aspects of the data set. This format does not provide strict validation between the data set and its structural definition using a generic XML parser.

3. *Compact Data:* This format is specific to the key family of the data set it encodes, and is created by following mappings between the metadata constructs defined in the Structure Definition message and the compact format. It supports the exchange of large data sets in XML format (similar to SDMX-EDI), and allows for the transmission of partial data sets (incremental updates) as well as whole data sets.
4. *Utility Data:* Many XML tools and technologies have expectations about the functions performed by an XML schema, one of which is a very direct relationship between the XML constructs described in the XML schema and the tagged data in the XML instance. Strong datatyping is also considered normal, supporting full validation of the tagged data. This message type, like the Compact Data message, is specific to the key family of the data set, but is designed to support validation and other expected XML schema functions. It is also derived from the Structure Definition message through the implemention of a set of standard mappings. It requires that a data set be complete in order to validate with an XML parser.
5. *Cross-Sectional Data:* Unlike data oriented towards processing as time series, some statistical data consists of large numbers of observations at a single point in time. This message type, like the Compact Data message, is specific to the key family of the data set, but is oriented for this different packaging of data. The cross-sectional format is based on the same data set structure description as other (time series) formats, so that searches can be made across time series data, and then formatted for this type of processing if desired.
6. *Query:* Data and metadata are often published in databases which are available on the web. Thus, it is necessary to have a standard query document which allows the databases to be queried, and return an SDMX-ML message. The Query document is an implementation of the SDMX Information Model for use in web services and database-driven applications, allowing for a standard request to be sent to data providers using these technologies.

Because all of the SDMX-ML formats are implementations of the same information model, and all the data messages are derivable from the Structure Definition message which describes a data set, it is possible to have standard mappings between each of the data formats. These mappings can be implemented in generic transformation tools, useful to all SDMX-ML users, and not specific to a particular data set’s key family (even though some of the formats they deal with may be). Part of the SDMX-ML package is the set of mappings between the key family-specific data formats and the Structure Definition format from which all are derivable.

VI. CONFORMANCE

This section is a normative statement of what applications must do to be considered conformant with the SDMX version 1.0 specifications. This section addresses both what application functionality must be supported, and the contents of an Implementor's Conformance Statement regarding SDMX conformance.

A. Conformance with the SDMX Specifications

SDMX standardizes the *exchange* of statistical data and metadata between counterparties. Thus, conformance is only meaningful for applications which have an exchange function between counterparties.

B. Implementor's Conformance Statement

In order to be SDMX-conformant, an application must have an Implementor's Conformance Statement (ICS), specifying the details of conformance. The ICS states which message types are supported, and how.

A "message type" is defined as an item from the following list, within groups as indicated by italicized headings:

Structure Message Types

SDMX-EDI Key Family
SDMX-EDI Concept
SDMX-EDI Codelist
SDMX-ML Key Family
SDMX-ML Concept
SDMX-ML Codelist
SDMX-ML Agency

Data Message Types

SDMX-EDI Data
SDMX-ML Generic Data
SDMX-ML Utility Data
SDMX-ML Compact Data
SDMX-ML Cross-Sectional Data

Query Message Types

SDMX-ML Query

The Implementor's Conformance Statement must declare for each supported message type if the application supports read functionality, write functionality, or both.

If the ICS declares support for SDMX-ML Key Families, SDMX-ML Codelist, SDMX-ML Concept, and/or SDMX-ML Agency, it must also state for each declaration whether structural dependencies on codelists, concepts, and agencies may be included in the message by reference, inline (that is, present within the message), or both.

If the ICS declares support for the SDMX-EDI Data message type, the SDMX-ML Generic Data message type, the SDMX-ML Compact Data message type, and/or the SDMX-ML Cross-Sectional Data message type, then it must declare for each message type whether it supports Delete actions.

C. Application Functionality

To be SDMX conformant, applications are required to perform specific functionality, according to which message types are supported.

For each message type supported with Read functionality, a valid message must be accepted as input by the application. For each message type supported with Write functionality, a valid message must be produced as output by the application. For all SDMX-EDI message types, validity is defined in [reference to SDMX-EDI spec]. For all SDMX-ML message types, validity is defined in [reference to SDMX-ML spec.]

For all applications declaring conformance for the SDMX-EDI key family structure message type, the messages read or written must be valid SDMX-EDI instances [normative reference], containing one or more complete key families with all concepts and codelists on which it has dependencies. (Codelists consist of VLI, CDV, and FTX segments; concepts of STC, and FTX segments; key family definitions consist of ASI, FTX, SCD, ATT, and IDE segments.)

For all applications declaring conformance for the SDMX-EDI concepts structure message type, the messages read or written must be valid SDMX-EDI instances [normative reference] with one or more STC segments containing complete concepts.

For all applications declaring conformance for the SDMX-EDI codelist structure message type, the messages read or written must be valid SDMX-EDI instances [normative reference] with at least one VLI segment containing a complete codelist.

For all applications declaring conformance for any SDMX-ML structure message types, the messages read or written must be valid XML instances [normative reference] with a root element of *StructureMessage*.

For all applications declaring conformance for SDMX-ML Key Family message types, the messages read or written must be valid XML instances which include an instance of the *KeyFamily* element [ref]. If support for referenced structural dependencies is declared, then the application must be capable of resolving references to the key family's codelists, concepts, and agencies. If support for inline structural dependencies is declared, then the concepts, codelists, and agencies must be read or written from within the key family's XML instance.

For all applications declaring conformance for SDMX-ML Concept message types, the messages read or written must be valid XML instances which include an instance of the *Concept* element [ref].

For all applications declaring conformance for SDMX-ML Codelist message types, the messages read or written must be valid XML instances which include an instance of the *Codelist* element [ref].

For all applications declaring conformance for SDMX-ML Agency message types, the messages read or written must be valid XML instances which include an instance of the *Agency* element [ref].

For all applications declaring conformance for the SDMX-EDI data message type, the messages read or written must be valid SDMX-EDI instances [normative reference] with DSI segments containing the data.

For all applications declaring conformance for the SDMX-ML Generic Data message, the messages read or written must be valid XML instances [ref] with a root element of *GenericDataMessage* or a root element of *GenericDataMessageGroup*.

For all applications declaring conformance for the SDMX-ML Utility Data message, the messages read or written must be valid XML instances [ref] with a root element of *UtilityDataMessage* or a root element of *UtilityDataMessageGroup*, and be validatable according to an XML schema [ref] derived from a valid key family according to the mappings specified in [reference to SDMX-ML section].

For all applications declaring conformance for the SDMX-ML Compact Data message, the messages read or written must be valid XML instances [ref] with a root element of *CompactDataMessage* or a root element of *CompactDataMessageGroup*, and be validatable according to an XML schema [ref] derived from a valid key family according to the mappings specified in [reference to SDMX-ML section].

For all applications declaring conformance for the SDMX-ML Cross-Sectional Data message, the messages read or written must be valid XML instances [ref] with a root element of *CrossSectionalDataMessage* or a root element of *CrossSectionalDataMessageGroup*, and be validatable according to an XML schema [ref] derived from a valid key family according to the mappings specified in [reference to SDMX-ML section].

For all applications declaring conformance for the SDMX-ML Query message types, the messages read or written must be valid XML instances [normative reference] with a root element of *QueryMessage*. [ref to Query spec].

For all applications declaring conformance for the SDMX-EDI data message type, and support for the Delete action, it must be able to create and/or meaningfully process a message where the STS segment has a status-type of “3” (data contents) and a status code of “6” (delete).

For all applications declaring conformance for the SDMX-ML Generic Data message type and support for the Delete action, the application must be able to create and/or meaningfully process a valid *SDMXGenericDataMessage* with an *Action* code value of “delete”.

For all applications declaring conformance for the SDMX-ML Compact Data message type and support for the Delete action, the application must be able to create and/or meaningfully process a valid *SDMXCompactDataMessage* with an *Action* code value of “delete”.

For all applications declaring conformance for the SDMX-ML Cross-Sectional Data message type and support for the Delete action, the application must be able to create and/or meaningfully process a valid *SDMXCrossSectionalDataMessage* with an *Action* code value of “delete”.

VII. DEPENDENCIES ON SDMX CONTENT STANDARDS

The technical standards proposed here are dependent on other SDMX standards which are more closely tied to the content and semantics of statistical data exchange. The SDMX Information Model works equally well with any statistical concept, but to encourage interoperability, it is also necessary to standardize and harmonize the use of specific concepts and terminology. To achieve this goal, SDMX will create and maintain a framework for the standardization of concepts, terminology, and key families within the statistical domain. There are three major parts to this effort.

A. Core Statistical Concepts

The SDMX Core Statistical Concepts is a content standard which collects and defines the cross-domain metadata items or categories which should be used wherever possible to enhance possibilities of interchange. Examples of such concepts are "data source used", "periodicity", "population coverage" and "seasonal adjustments".

This list is expected to grow rapidly, and to be subject to frequent revision as SDMX is used in a growing number of domains. The use of the SDMX Core Statistical Concepts where appropriate is not a requisite part of technical conformance, but provides a framework to promote interoperability among those who are also compliant with the technical standards presented here.

The harmonization of statistical concepts includes not only the definition of the concepts, and their names, but also, where appropriate, their representation with standard code lists (for instance for "periodicity") and the role they play within key family structures.

The intent of this standard is two-fold: to provide a core set of concepts which can be used to structure and describe statistical data, to promote interoperability between systems; and to promote the exchange of metadata more widely, with a set of harmonized concept names and definitions.

For the version 1.0 SDMX specifications, the focus is on the use of SDMX Core Statistical Concepts in their role as regards structural metadata. In future versions, it is anticipated that their applicability to reference metadata will become equally important.

B. Metadata Common Vocabulary

The Metadata Common Vocabulary (MCV) provides a set of standard metadata items and their definitions, that are commonly used for the description of statistical data. This enables the comparison and mapping of terminology found in key family descriptions and in other aspects of statistical metadata management. The MCV covers the main "reference" and "structural" metadata terms, as well as ISO/IEC 11179-compliant definitions. These definitions may be used directly, or facilitate interchange between systems which adopt different information models

The MCV provides definitions for terms on which the Core Statistical Concepts work is built. The Metadata Common Vocabulary is available today, maintained by SDMX, and will be subject to frequent expansion and revision.

C. Core Statistical Subject-Matter Domains

The Core Statistical Subject-Matter Domains is a listing of the breadth of statistical information for the purposes of organizing widespread statistical exchange and categorization. It acts as a standard scheme against which the categorization schemes of various counterparties can be mapped, to facilitate interoperable data and metadata exchange. It serves another useful purpose, however, which is to allow an organization of corresponding "domain groups", each of which could define standard key families, concepts, etc. within their domains. Such groups already exist within the international community – SDMX would use the Core Statistical Subject-Matter Domains list to focus the energies of these groups on producing the kinds of content standards which would be useful to support the interoperation of SDMX-complaint technical systems within each domain.

The Core Statistical Domains list will be created and maintained by the SDMX Initiative and will be subject to adjustment.

D. Non-SDMX Content Standards

In addition to the content standards created and maintained by SDMX, there are some international standards in the metadata arena which will be of importance to the use of SDMX technical specifications moving forward. While these need not be enumerated here, there is one which is very important: ISO/IEC 11179. This standard provides a structure for modelling metadata registries which may be a powerful tool for the interoperation of metadata across systems.

It should be noted that SDMX does not plan to define or adopt a specific metadata registry model. It does plan to define metadata registry interfaces for registration and discovery services of SDMX objects. Therefore, any underlying metadata source could be used to store, search, and retrieve SDMX objects, providing the appropriate middleware exists to map the SDMX interfaces to the native interfaces of the chosen registry. In this regard SDMX may, in the future, provide a mapping between the SDMX model and one or more metadata registry models such as ebXML and ISO 11179.

VIII. LOOKING FORWARD

The SDMX initiative sees this set of format standards as extremely useful in creating more efficient and open systems for statistical exchange. It is clear that some additional standards may provide even greater efficiencies. These concern the use of a standard infrastructure, based on the use of a registry framework.

It is anticipated that SDMX will provide these further standards in such a way as to build on the interoperability allowed by having a set of standard formats, based on a common information model. For more information about the status of this effort, please visit http://www.sdmx.org.