STATISTICAL DATA AND METADATA EXCHANGE INITIATIVE

FRAMEWORK FOR SDMX TECHNICAL STANDARDS

(VERSION 2.0)

November 2005

1 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 the SDMX 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. This document has normative sections.

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. This document has normative sections.

4. The SDMX Registry Specification provides for a central registry of information about available data and reference metadata, and for a repository containing structural metadata and provisioning information. This specification defines the basic services offered by the SDMX Registry: registration of data and metadata; querying for data and metadata; and subscription/notification regarding updates to the registry. This document has normative sections.

5. The SDMX Implementor's Guide – this is a guide to help those who wish to use the SDMX 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.

6. 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 (including, but not requiring, an SDMX-conformant registry) 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 1.1 specification. This document is not normative.

2 CHANGES FROM VERSION 1.0

The 2.0 version of this standard represents a significant increase in scope, and also provides more complete support in those areas covered in the version 1.0 specification. Version 2.0 of this standard is backward-compatible with version 1.0, so that existing implementations can be easily migrated to conformance with version 2.0.

Note that the idea of backward-compatibility in the standards is based on the information model. In this release, some non-backward-compatible changes have been made to the SDMX-ML formats. The same set of information required to use version 1.0 of the specification will permit the use of the same features in the version 2.0 specifications, however. Thus, a key family is easily translated from version 1.0 to version 2.0, without requiring any new information regarding structures, etc. There have been no changes to the SDMX-EDI format.

The changes can be briefly summarized:

- In addition to describing and specifying data structures and formats (along with related structural metadata), the version 2.0 specification also provides for the exchange of metadata which is distinct from the structural metadata in the 1.0 version. This category includes "reference" metadata (regarding data quality, methodology, and similar types – it can be configured by the user to include whatever concepts require reporting); metadata related to data provisioning (release calendar information, description of the data and metadata provided, etc.); and metadata relevant to the exchange of categorization schemes.
- Provision is made in the 2.0 standard for standard communication with registry services, to support a data-sharing model of statistical exchange. These services include registration of data and metadata, querying of registered data and metadata, and subscription/notification.
- The support for exchange of statistical data and related structural metadata has been expanded. Some support is provided for qualitative data; data cube structures are described; hierarchical code lists are supported; relationships between data structures can be expressed, providing support for extensibility of data structures; and the description of functional dependencies within cubes are supported.

3 PROCESSES AND BUSINESS SCOPE

3.1 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 possible to agree to use the full set of SDMX standards to support a common data-sharing process of exchange, whether based on an SDMX-conformant registry or some other architecture.

The standards specified here specifically support a data-sharing process based on the use of central registry services. Registry services provide visibility into the data and metadata existing within the community, and support the access and use of this data and metadata by providing a set of triggers for automated processing. The data itself is not stored in a central registry – these services merely provide a useful set of metadata about the data (and additional metadata) in a known location, so that users/applications can easily locate and obtain whatever data and/or metadata is registered. The use of standards for all data, metadata, and the registry services themselves is ubiquitous, permitting a high level of automation within a data-sharing community.

It should be pointed out that these different process models are not mutually exclusive – a single system capable of expressing data and metadata in SDMX-conformant formats could support all three scenarios. Different standards may be applicable to different processes (for example, many registry services interfaces are used only in a data-sharing scenario) but all have a common basis in a shared information model.

In addition to looking at collection and reporting, it is also important to consider the dissemination of data. Data and metadata – no matter how they are exchanged between counterparties in the process of their development and creation – are all eventually supplied to an end user of some description. Often, this is through specific applications inside of institutions. But more and more frequently, data and metadata are also published on websites in various formats. The dissemination of data and its accompanying metadata on the web is a focus of the SDMX standards. Standards for statistical data and metadata allow improvements in the publication of data – it becomes more easily possible to process a standard format once the data is obtained, and the data and metadata are linked together, making the comprehension and further processing of the data easier.

In discussions of statistical data, there are many aspects of its dissemination which impact data quality: data discovery, ease of use, and timeliness. SDMX standards provide support for all of these aspects of data dissemination. Standard data formats promote ease of use, and provide links to relevant metadata. The concept of registry services means that data and metadata can more easily be discovered. Timeliness is improved throughout the data lifecycle by increases in efficiency, promoted through the availability of metadata and ease of use.

It is important to note that SDMX is primarily focused on the *exchange* and *dissemination* of statistical data and metadata. 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, however. It is felt that a clear, standard formatting of data and metadata for the purposes of exchange and dissemination can also facilitate internal processing by organizations and users, but this is not the focus of the specification.

3.2 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 Processable 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.

3.3 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 often 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 values and concepts can be understood as the named dimensions of a multi-dimensional co-ordinate system, 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 related 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. A distinction can be made between “structural” metadata – those concepts used in the description and identification of statistical data and metadata – and “reference” metadata – the larger set of concepts that describe and qualify statistical data sets and processing more generally, and which are often associated not with specific observations or series of data, but with entire collections of data or even the institutions which provide that data.

The SDMX Information Model provides for the structuring not only of data, but also of “reference” metadata. While these reference metadata structures exist independent of the data and its structural metadata, they are often linked. The SDMX Information Model provides for the attachment of reference metadata to any part of the data or structural metadata, as well as for the reporting and exchange of the reference metadata and its structural descriptions. This function of the SDMX standards is intended to support many aspects of data quality initiatives, allowing as it does for the exchange of metadata in its broadest sense, of which quality-related metadata is a major part.

Metadata is also associated, not only with data, but also with the process of providing and managing the flow of data. The SDMX Information Model provides for a set of metadata concerned with “data provisioning” – metadata which is useful to those who need to understand the content and form of a data provider’s output. Each data provider can describe in standard fashion the content of and dependencies within the data- and metadata sets which they produce, and supply information about the scheduling and mechanism by which their data and metadata is provided. This allows for automation of some validation and control functions, as well as supporting management of data reporting.

SDMX also recognizes the importance of classification schemes in organizing and managing the exchange and dissemination of data and metadata. It is possible to express information about classification schemes and domain categories in SDMX, along with their relationships to data and metadata sets.

The formal objects in the information model are presented briefly below, but are also discussed in more detail elsewhere in this specification.

3.4 The SDMX View of Statistical Exchange

Version 1.0 of ISO/TS 17369 SDMX covered statistical data sets and the metadata related to the structure of these data sets - "key families". This scope was useful in supporting the different models of statistical exchange (bilateral exchange, gateway exchange, and data-sharing) but was not by itself sufficient to support them completely. Version 2.0 provides a much more complete view of statistical exchange, so that an open data-sharing model can be fully supported, and other models of exchange can be more completely automated. In order to produce technical standards that will support this increased scope, the SDMX Information Model provides a broader set of formal objects which describe the actors, processes, and resources within statistical exchanges.

It is important to understand the set of formal objects not only in a technical sense, however, but also in terms of what they represent in the real-world exchange of statistical data and metadata.

The first version of SDMX provided for data sets - specific statistical data reported according to a specific structure, for a specific time range - and for key families (data structure definitions) - the metadata which describes the structure of statistical data sets. These are important objects in statistical exchanges, and are retained and enhanced in the second version of the standards in a backward-compatible form. A related object in statistical exchanges is the "data flow" - this is the on-going publication of a data set, as new observations are added to the existing ones, or as subsequent data sets with the same subject and structure are published. "Data flows" can be understood as data sets which are not bounded by time. Data structures are owned and maintained by agencies - in a similar fashion, data flows are published by "data providers", and owned by maintenance agencies.

Version 2.0 – like version 1.0 – allows for the publication of statistical data (and the related structural metadata) but it also provides for the standard, systematic representation of reference metadata. Reference metadata is any metadata which is reported not as an integral part of a data set, but independent of the statistical data. SDMX provides for reference "metadata sets", for "metadata structure definitions", and also for "metadata flows". These objects are very similar to data sets, key families (data structure definitions), and data flows, but they concern reference metadata rather than statistical data. In the same way that data providers may publish statistical data, they may also publish reference metadata. Metadata structural definitions are maintained by agencies in a fashion similar to the way that agencies maintain key families, the structural definitions of data sets.

The structural definitions of both data and reference metadata associate specific statistical concepts with their representations, whether textual, coded, etc. In SDMX version 2.0, these concepts are taken from a "concept scheme" which is maintained by a specific agency. Concept schemes group a set of concepts, provide their definitions and names, and allow for semantic relationships to be expressed, when some concepts are specializations of others. It is possible for a single concept scheme to be used both for data structures - key families - and for reference metadata structures.

Inherent in any statistical exchange – and in many dissemination activities - is a concept of "service level agreement", even if this is not formalized or made explicit. SDMX incorporates this idea in objects termed "provision agreements". Data providers may provide data to many different data flows. Data flows may incorporate data coming from more than one data provider. Provision agreements are the objects which tell you which data providers are supplying what data to which data flows. The same is true for metadata flows.

Provision agreements allow for a variety of information to be made available: the schedule by which statistical data or metadata is reported or published, the specific topics about which data or metadata is reported within the theoretically possible set of data (as described by a key family or reference metadata structure definition), and the time period covered by the statistical data and metadata. This set of information is termed "constraints" in the SDMX Information Model. Constraints are associated with data providers (typically the schedules and time periods for their data), with data flows (typically describing the topics covered), and on the provision agreement (where a full description of time-related constraints and topical coverage is given).

A brief summary of those objects includes:

- ***Data Set:*** Data is organized into discrete sets, which include particular observations for a specific period of time. A data set can be understood as a collection of similar data, sharing a structure, which covers a fixed period of time.
- ***Key Family (Data Structure Definition):*** Each data set has a set of structural metadata. These descriptions are referred to in SDMX as “key families”, 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 identifying and descriptive (structural) metadata.
- ***Codelists:*** A code list is a maintained list of codes that may be used in a key family or metadata structure definition. Codelists enumerate a set of values to be used in the representation of dimensions, attributes, and other structural parts of SDMX. They can be supplemented by metadata which indicates how codes are organized into hierarchies.
- ***Metadata Set:*** A reference metadata set is a set of information regarding almost any object within the formal SDMX view of statistical exchange: they may describe the maintainers of data or structural definitions; 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. In SDMX, the creators of reference metadata may take whatever concepts they are concerned with, or obliged to report, and provide a reference metadata set containing that information.
- ***Metadata Structure Definition:*** A reference metadata set also has a set of structural metadata which describes how it is organized. This metadata identifies what reference metadata concepts are being reported, how these concepts relate to each other (typically as hierarchies), what their presentational structure is, how they may be represented (as free text, as coded values, etc.), and with which formal SDMX object types they are associated.
- ***Maintenance Agency:*** In SDMX, an organization which creates and maintain the structural definitions for data and metadata are called maintenance agencies. Every key family and code list, for example, has an agency.
- ***Dataflow Definition:*** In SDMX, data sets are reported or disseminated according to a data flow definition. The data flow definition identifies the key family and may be associated with one or more subject matter domains (this facilitates the search for data according to organised category schemes).

Constraints, in terms of reporting periodicity or sub set of possible keys that are allowed in a data set, may be attached to the data flow definition.

- ***Metadataflow Definition:*** A metadata flow definition is very similar to a data flow definition, but describes, categorises, and constrains metadata sets.
- ***Data Provider:*** An organization which produces data or reference metadata is termed a data provider.
- ***Provision Agreement:*** The set of information which describes the way in which data sets and metadata sets are provided by a data provider. A provision agreement can be constrained in much the same way as a data or metadata flow definition. 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. Importantly, the actual source of registered data or metadata is attached to the provision agreement (in terms of a URL). The term "agreement" is used because this information can be understood as the basis of a "service-level agreement". In SDMX, however, this is informational metadata to support the technical systems, as opposed to any sort of contractual information (which is outside the scope of a technical specification).
- ***Constraints:*** Constraints describe a subset of a data set or metadata set, and may also provide information about scheduled releases of data. They are associated with data providers, provision agreements, and data flows.
- ***Category Scheme:*** Categorization schemes are made up of a hierarchy of categories, which in SDMX may include any type of useful classification for the organization of data and metadata.
- ***Concept Scheme:*** A concept scheme is a maintained list of concepts that are used in key family and metadata structure definitions. There can be many such concept schemes. A "core" representation of the concept can be specified (e.g. a core code list, or other representation such as "date"). Note that this core representation can be overridden in the key family or metadata structure definition that uses the concept. Indeed, organisations wishing to remain with version 1.0 key family schema specifications will continue to declare the representation in the key family definition.
- ***Structure Set:*** Structure sets provide a mechanism for grouping structural metadata together to form a complete description of the relationships between specific, related sets of data and metadata. They can be used to map dimensions and attributes to one another, to map concepts, to map codelists, and to map category schemes. They can be used to describe "cubes" of data, even when the data within the cube does not share a single dimensionality.
- ***Reporting Taxonomy:*** A reporting taxonomy allows an organisation to link (possibly in a hierarchical way) a number of cube or data flow definitions which together form a complete "report" of data or metadata. This supports primary reporting which often comprises multiple cubes of heterogeneous data, but may also support other collection and reporting functions.
- ***Process:*** The process class provides a way to model statistical processes as a set of interconnected *process steps.* Although not central to the exchange and dissemination of statistical data and metadata, having a shared description of processing allows for the interoperable exchange and dissemination of reference metadata sets which describe processes-related concepts.

3.4.1 Notes on Data Structuring

A “cube” is a rich, multi-dimensional construct, which can be viewed along any of its axes (or “dimensions”). Whilst the full structure of cube data can be described in SDMX, the actual “data” specification of SDMX takes a slightly narrower view of these requirements in its version 2.0 specifications for the purposes of formatting the data for transmission. The view of data in most 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, where data are organized around some other, non-time dimension of the cube. SDMX provides support for cross-sectional views of data cubes. 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.

Another type of structure commonly found in statistical “cubes” of data is the hierarchical classification, used to describe the points along any of its dimensions (or axes). In the 1.0 version, SDMX standards did not provide full support for this functionality. The introduction of these hierarchical classifications is present in the current version of the standard.

Further, there is support for the expression of functional dependencies between the various dimensions of a cube, giving support for better processing of “sparse cubes”. This is an aspect of “constraints”, which allow for the framing of a cube region, or for the provision of a set of valid keys within the total set of keys described by the key family.

3.4.2 Notes on Reference Metadata Structuring

Metadata structures are based on the idea that concepts can be organised into semantic and presentational hierarchies, and that these hierarchies can form the basis for the structuring of XML reporting formats. There are three message types in SDMX-ML which serve this purpose: the Structure message (providing the metadata structure definition), the Generic Metadata message (providing a single format for any metadata structure definition), and the Metadata Report message (providing a metadata structure definition-specific format). Typically, this mechanism is suited to supporting reference metadata reporting and dissemination.

The Metadata Structure Definition takes *any* concept from concept schemes, and describes how they can be formed into a presentational structure – either as a flat list, or as a hierarchy. The concepts are assigned representations (coded, textual, etc.) The “target” of the metadata – that is, the class of process, information, organisation, exchange, etc. – which is the subject of the metadata is described. Because the SDMX Information Model gives a formalization of statistical exchange and dissemination, the model can be used as a typology of the different actors and resources within statistical activities. Thus, the “targets” (subjects) of reference metadata sets and metadata flows can be described as corresponding to some standard class by reference to this model.

As with data structures, there is a generic format for metadata sets and one that performs a higher degree of validation, derived specifically from a metadata structure definition.

3.5 SDMX Registry Services

In order to provide visibility into the large amount of data and metadata which exists within the SDMX model of statistical exchange, it is felt that an architecture based on a set of registry services is potentially useful. A "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 understood 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, located across the Internet or similar network.

Note that the SDMX registry services are not concerned with the storage of data or reference metadata. The assumption is that data and reference metadata lives on the sites of its data providers. The SDMX registry services concern themselves with providing visibility of the data and reference metadata, and information needed to access the data and reference metadata. Thus, a registered data set will have its URL available in the registry, but not the data itself. An application which wants that data would query the registry for the URL, and then have to go and retrieve the data from the data provider.

SDMX does not require a particular technology implementation of the registry – instead, it specifies the standard interfaces which may be supported by a registry. Thus, users may implement an SDMX-conformant registry in any fashion they choose, so long as the interfaces are supported as specified here. These interfaces are expressed as XML documents, and form a new part of the SDMX-ML language.

The registry services discussed here can be briefly summarized:

- ***Registration/Structural Metadata Submission:*** This registry service allows users with write access privileges to inform the registry that data sets, reference metadata sets, structural metadata, or data provisioning information exists. The registry stores a wide range of metadata about these objects. Objects in the registry are organized and categorized according to one or more categorization schemes.
- ***Querying:*** The registry services have interfaces for querying the metadata contained in a registry, so that applications and users can discover the existence of data sets and reference metadata sets, structural metadata, the providers/agencies associated with those objects, and the provider agreements which describe how the data and metadata are made available, and how they are categorized.
- ***Subscription/Notification:*** It is possible to "subscribe" to specific objects in a registry, so that a notification will be sent to all subscribers whenever the registry objects are updated.

4 THE SDMX INFORMATION MODEL

SDMX provides a way of 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 version 1.0 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. Version 2.0 expands upon the version 1.0 basis to provide a more comprehensive model.

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 and describe data cubes from SDMX-structured data, and to exchange such databases according to the proposed standards. In version 2.0, it becomes possible to more fully describe the structure of cubes, with hierarchical codelists, constraints, and relationships between key families.

In version 2.0, the SDMX standards also provide a view of reference metadata: a mechanism for referencing the meaningful “objects” within the SDMX view of statistical exchange processes (data providers, structures, provisioning agreements, dataflows, metadata flows, etc.) to which metadata is attached; a mechanism for describing a set of meaningful concepts, of organizing them into a presentational structure, and of indicating how their values are represented. This is based on a simple, hierarchical view of reference metadata which is common to many metadata systems and classification/categorization schemes. SDMX provides a model (and XML formats) for both describing reference metadata structures, and of reporting reference metadata according to those structures.

Version 2.0 also introduces support for metadata related to the process aspects of statistical exchange. A step-by-step process can be modelled; information about who is providing data and reference metadata and how they are providing it can be expressed; and the technical aspects of service-level agreements (and similar types of provisioning agreements) can be represented.

The SDMX Information Model formally describes all of the objects listed above, so as to present a standard view of the statistical exchange process.

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.

5 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.

6 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 or metadata set, and additional metadata about categorization schemes and organisations is included. Also, the structural aspects of data and metadata provision – dataflows and metadataflows – can be described using this format.

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 implementation 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. *Generic Metadata:* All reference metadata expressible in SDMX-ML format can be marked up according to this schema. It performs only a minimum of validation, and is somewhat verbose, but it does support the creation of generic software tools and services for processing reference metadata.

7. *Metadata Report:* For each metadata structure definition, an XML schema specific to that structure can be created, to perform validation on sets of reported metadata. This structure is less verbose than the Generic Metadata format, and is probably easier to use, because the XML mark-up relates directly to the reported concepts. It corresponds to the Utility format for data in its approach to the use of XML.

8. *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.

9. *Registry:* All of the possible interactions with the SDMX registry services are supported using SDMX-ML interfaces. These documents are almost all based on a synchronous exchange of documents – a “request” message answered by a “response” message. There are two basic types of request – a “Submit”,

which writes metadata to the registry services, and a "Query", which is used to discover that metadata. Registry interactions provide formats for all types of provisioning metadata, as well as for subscription/notification, structural metadata, and data and metadata registration.

Because all of the SDMX-ML formats are implementations of the same information model, and all the data and metadata messages are derivable from the Structure Definition message which describes a data set or metadata set, it is possible to have standard mappings between each of the similar 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 or metadata set's structure definition (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.

7 CONFORMANCE

This section is a normative statement of what applications must do to be considered conformant with the SDMX version 2.0 specifications. This section addresses both what application functionality must be supported, and the contents of an Implementor's Conformance Statement regarding SDMX conformance.

7.1 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.

7.2 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 Metadata Structure Definition
SDMX-ML OrganisationScheme (replaces Agency)
SDMX-ML Hierarchical Codelist
SDMX-ML Structure Set
SDMX-ML Reporting Taxonomy
SDMX-ML Process

Data Message Types

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

Metadata Message Types

SDMX-ML Generic Metadata
SDMX-ML Metadata Report

Query Message Types

SDMX-ML Query

Registry Message Types

SDMX-ML Registry Notification
SDMX-ML Submit Subscription Request
SDMX-ML Submit Subscription Response
SDMX-ML Submit Registration Request
SDMX-ML Submit Registration Response
SDMX-ML Query Registration Request
SDMX-ML Query Registration Response
SDMX-ML Submit Structure Request
SDMX-ML Submit Structure Response
SDMX-ML Query Structure Request
SDMX-ML Query Structure Response
SDMX-ML Submit Provisioning Request
SDMX-ML Submit Provisioning Response
SDMX-ML Query Provisioning Request
SDMX-ML Query Provisioning Response

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 Metadata Structure Definitions, SDMX-ML Codelist, SDMX-ML Concept, and/or SDMX-ML Organisation Scheme, it must also state for each declaration whether structural dependencies on codelists, concepts, and data providers, 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.

If the ICS declares support for the SDMX-ML Compact Data message type, then it must declare whether it supports time ranges.

7.3 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-ML message types, validity is defined in SDMX-ML:Schema and Documentation, Sections V and VI. For all SDMX-EDI message types, validity is defined in SDMX-EDI: Syntax and Documentation, Chapters 9 and 10.

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, 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 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 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 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. 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.

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.

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

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

For all applications declaring conformance for SDMX-ML Metadata Structure Definition message types, the messages read or written must be valid XML instances which include an instance of the *MetadataStructureDefinition* element.

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

For all applications declaring conformance for the SDMX-EDI data message type, the messages read or written must be valid SDMX-EDI instances 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 with a root element of *GenericData* or a root element of *MessageGroup* containing one or more *GenericData* elements.

For all applications declaring conformance for the SDMX-ML Utility Data message, the messages read or written must be valid XML instances with a root element of *UtilityData* or a root element of *MessageGroup* containing one or more *UtilityData* elements, and be validatable according to an XML schema derived from a valid key family according to the mappings specified.

For all applications declaring conformance for the SDMX-ML Compact Data message, the messages read or written must be valid XML instances with a root element of *CompactData* or a root element of *MessageGroup* containing one or more *CompactData* elements, and be validatable according to an XML schema derived from a valid key family according to the mappings specified.

For all applications declaring conformance for the SDMX-ML Cross-Sectional Data message, the messages read or written must be valid XML instances with a root element of *CrossSectionalData* or a root element of *MessageGroup* containing one or more *CrossSectionalData* elements, and be validatable according to an XML schema derived from a valid key family according to the mappings specified.

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

For all applications declaring conformance for the SDMX-ML Generic Metadata message, the messages read or written must be valid XML instances with a root element of *GenericMetadata* or a root element of *MessageGroup* containing one or more *GenericMetadata* elements.

For all applications declaring conformance for the SDMX-ML Metadata Report message, the messages read or written must be valid XML instances with a root element of *MetadataReport* or a root element of *MessageGroup* containing one or more *MetadataReport* elements, and be validatable according to an XML schema derived from a valid metadata structure definition according to the mappings specified.

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 applications declaring conformance for the SDMX-ML Compact Data message type and support for time ranges, the application must be able to create and/or meaningfully process non-first-position observations in a series supplied without time, and calculate the times for non-first position observations.

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”.

For all applications declaring conformance for the SDMX-ML Registry Notification message, the messages read or written must be valid XML instances with a *RegistryNotification* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Subscription Request message, the messages read or written must be valid XML instances with a *SubmitSubscriptionRequest* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Subscription Response message, the messages read or written must be valid XML instances with a *SubmitSubscriptionResponse* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Registration Request message, the messages read or written must be valid XML instances with a *SubmitRegistrationRequest* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Registration Response message, the messages read or written must be valid XML instances with a *SubmitRegistrationResponse* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Query Registraton Request message, the messages read or written must be valid XML instances with a *QueryRegistrationRequest* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Query Registraton Response message, the messages read or written must be valid XML instances with a *QueryRegistrationResponse* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Structure Request message, the messages read or written must be valid XML instances with a *SubmitStructureRequest* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Structure Response message, the messages read or written must be valid XML instances with a *SubmitStructureResponse* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Query Structure Request message, the messages read or written must be valid XML instances with a *QueryStructureRequest* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Query Structure Response message, the messages read or written must be valid XML instances with a *QueryStructureResponse* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Provisioning Request message, the messages read or written must be valid XML instances with a *SubmitProvisioningRequest* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Submit Provisioning Response message, the messages read or written must be valid XML instances with a *SubmitProvisioningResponse* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Query Provisioning Request message, the messages read or written must be valid XML instances with a *QueryProvisioningRequest* element, and comply with all requirements of the SDMX Registry Specification.

For all applications declaring conformance for the SDMX-ML Query Provisioning Response message, the messages read or written must be valid XML instances with a *QueryProvisioningResponse* element, and comply with all requirements of the SDMX Registry Specification.

8 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 guidelines for cross-domain concepts, terminology, and structural definitions. There are three major parts to this effort.

8.1 Cross-Domain Metadata Concepts

The SDMX Cross-Domain Metadata Concepts is a content guideline concerning concepts which are used across many statistical domains. 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 Cross-Domain Metadata 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 definitions of the concepts, and their names, but also, where appropriate, their representations with standard codelists, and the role they play within key family structures and metadata structure definitions.

The intent of this guideline is two-fold: to provide a core set of concepts which can be used to structure statistical data and metadata, to promote interoperability between systems ("structural metadata", as described above); and to promote the exchange of metadata more widely, with a set of harmonized concept names and definitions for other types of metadata ("reference metadata", as defined above.)

8.2 Metadata Common Vocabulary

The Metadata Common Vocabulary is an SDMX guideline which provides definition of terms to be used for the comparison and mapping of terminology found in key family descriptions and in other aspects of statistical metadata management. Essentially, it provides ISO/IEC 11179-compliant definitions for a wide range of statistical terms, which may be used directly, or against which other terminology systems may be mapped. This set of terms is inclusive of the terminology used within the SDMX Technical Standards.

The MCV provides definitions for terms on which the SDMX Cross-Domain Metadata Concepts work is built.

8.3 Statistical Subject-Matter Domains

The 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 Statistical Subject-Matter Domains list to facilitate the efforts of these groups to develop the kinds of content standards which could support the interoperation of SDMX-conformant technical systems within and across statistical domains.

SDMX Statistical Subject-Matter Domains will be listed and maintained by the SDMX Initiative and will be subject to adjustment.

8.4 Non-SDMX Standards

There are also some other 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 promises to be very important: ISO/IEC 11179. This standard provides a structure for modelling metadata constructs which may be a powerful tool for the interoperation of metadata across systems.

With version 2.0 and the introduction of full support for reference metadata, SDMX is positioned to provide a mapping from its own Information Model to ISO/IEC 11179, so that other ISO/IEC 11179-compliant metadata systems can consistently use the metadata found in SDMX-compliant systems.

Other standards of interest include OASIS' Universal Business Language, which offers guidelines about the creation of XML schemas; and ISO/TS 15000 (ebXML), which offers a useful registry specification (parts 3 and 4) and an approach to the use of models (part 5). SDMX is actively aligned with these standards. SDMX is also looking at how it can align with several other standards.

9 LOOKING FORWARD

The SDMX initiative sees this set of data and metadata formats and registry services interfaces standards as useful in creating more efficient and open systems for statistical exchange.

It is anticipated that SDMX will refine these standards further as they are implemented in such a way as to build on the interoperability allowed by having a set of standard formats and exchanges, based on a common information model. The review process for version 2.0 has suggested that future work could be usefully focused in the area of computational processing, especially to support a more complete approach to data quality.

For more information about the status of this effort, please visit http://www.sdmx.org.