



## Not as hard as it sounds – Using XML in metadata-enabled infrastructure

by Frank Farance, Project editor  
ISO/IEC 20944 series, and Dan  
Gillman, Information Scientist,  
US Bureau of Labour Statistics

**P**ublished as a World Wide Web Consortium (W3C) recommendation in 1998, the Extensible Markup Language (XML) is essentially a subset of the Standard Generalized Markup Language (SGML), designed for easier implementation and in particular for easier delivery and interoperability over the Web. However, XML is not only about marking up text. It also lends itself to complex validation and management of content, or data, which is the aspect of XML addressed in this article.

### The good and the bad

The information and communications technology industries have broadly adopted XML for use in data interchange applications, including a serialization technique for aggregate data. The advantages of XML are many, such as a readable text format, standardization, available tool suites and support in open source software. But XML has its disadvantages, as well. These include significant space inefficiency, common misunderstandings about the meaning of XML data, and multiple strategies for structuring data.

Joint technical committee ISO/IEC JTC 1, *Information technology*, subcommittee SC 32, *Data management and interchange*, has successfully used a heterogeneous metadata infrastructure (multiple types of metadata, metadata registries, metadata repositories, federated search) to overcome some of these disadvantages while facilitating automated and semi-automated transformation of data. The following are some of ISO/IEC JTC 1/SC 32's successful strategies.

### Don't think in terms of XML

**Strategy No. 1: Don't think in terms of XML. Avoid XML bias.** If your organization has an enterprise-wide man-

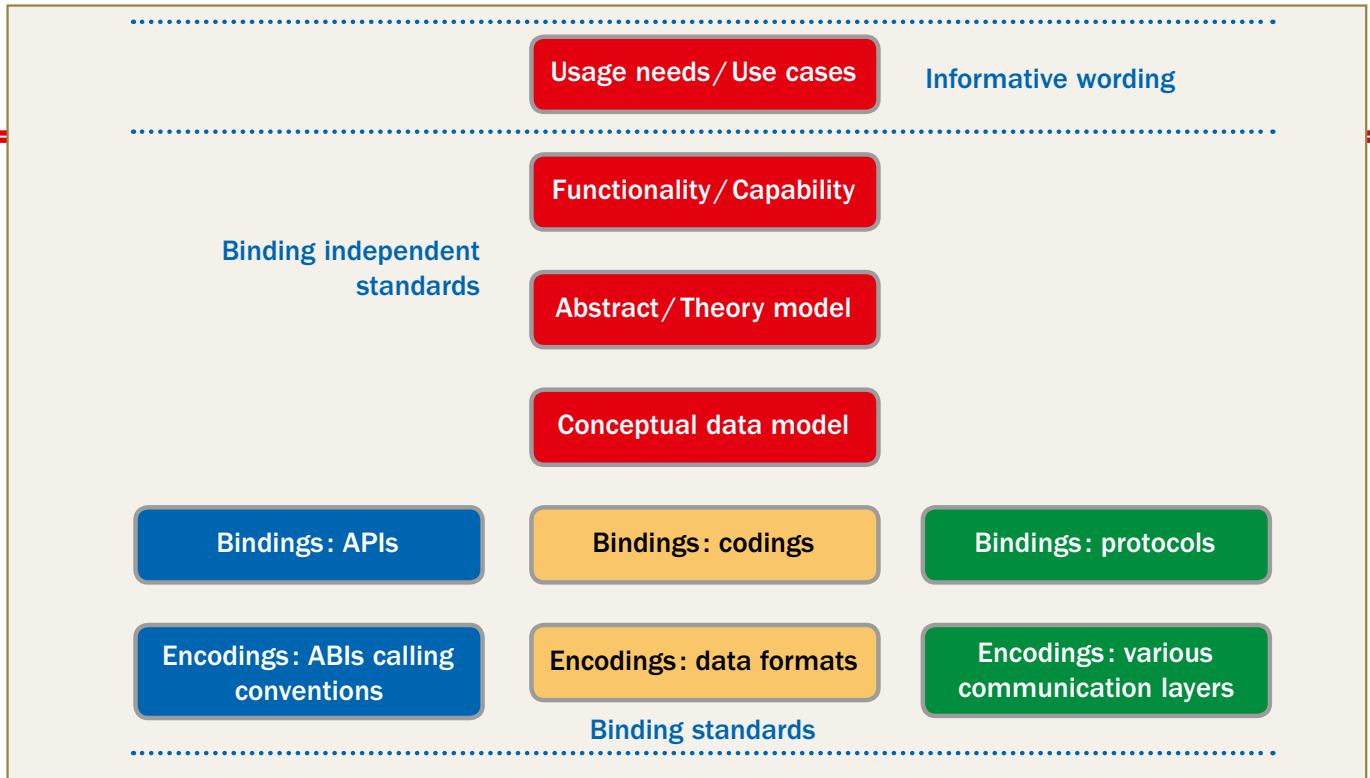


Figure 1 – A general layered data interoperability model

date to use XML, how can you not think about it? First, it is important to place XML in perspective with other technical decisions in developing interoperability specifications and implementing systems. The model shown in **Figure 1** may be helpful in framing a bigger picture.

Data interoperability can be decomposed into layers:

- Usage needs and use cases – typically, these are not standardized (e.g. a mailing list)
- Functionality and capability – agreement upon function and purpose (e.g. postal addresses used for mailing letters)
- Abstract/theory model – the theory or abstraction operation of the subject area (e.g. mailer sends letter, letter contains mailer’s postal address and addressee’s postal address, postal carrier delivers letter from mailer to addressee)
- Conceptual data model – binding-independent framing of data model (e.g. postal address is comprised of a delivery point and an addressee)
- Bindings – mapping to bindings, such as codings: e.g. XML, ASN.1<sup>1)</sup>; application programming interfaces: e.g. C, Java; and/or protocols: HTTP, WebDAV; (e.g. choice of tags for postal addresses such as <name>, <city>, <postal code>, etc.)

- Encodings – low-level bit-byte sequences, such as data encodings: e.g., UTF-8; application binary interfaces: Windows, Linux ELF; and/or communication layers: e.g. TCP/IP; (e.g. similar to showing <addressee> in English, Chinese, etc.)

If – instead of thinking specifically about XML – you design your data interchange specification as if you had to implement multiple codings (e.g. both XML

and ASN.1), and as if you had to implement a programming language interface, and as if you had to implement a session layer network protocol, then you are likely to use XML in a straightforward way (avoiding XML’s quirks). And you have allowed for growth if you have a future

1) Abstract Syntax Notation 1 (ASN.1) is specified in the ISO/IEC 8824 and ISO/IEC 8825 series (ITU-T Recommendations X.680 and X.690).

## About the authors



**Frank Farance** has been a developer of IT standards for more than 25 years, and the project editor for several ISO standards, including current projects:

metadata interoperability and bindings (ISO/IEC 20944 series) and metadata modules (ISO/IEC 19773); and published standards: general purpose data-types (ISO/IEC 11404:2007), and the C programming language (ISO/IEC 9899:1999). For the past ten years, Mr. Farance has been the ISO/IEC JTC 1 representative to the ISO Information technology strategies implementation group (ISO/ITSIG).



**Dan Gillman** is an Information Scientist in the Office of Survey Methods Research at the US Bureau of Labor Statistics. His work includes statistical metadata

management at the BLS, national and international statistical metadata initiatives, and national and international metadata standards. He has written extensively on metadata issues, chairs the UN Economic Commission for Europe working group on statistical metadata, chairs a US technical committee for metadata standards, and is editor for several international metadata standards.

## More on Strategy No. 3 features

need for another coding, application programming interface, or protocol.

The metadata-enabled environment being developed in SC 32's working group WG 2, *Metadata* (see **next article**), allows automated transformation between XML and ASN.1 codings. Data element metadata is recorded in the multipart ISO/IEC 11179, *Information technology – Metadata registries (MDR)*, to facilitate the automatic XML-ASN.1 conversion. This strategy allows us to use an efficient binary transfer method (ASN.1) on low bandwidth data links, and use XML elsewhere (which satisfies an enterprise-wide need). Also relevant is the ISO/IEC 20944 series, *Information technology – Metadata Registries Interoperability and Bindings (MDR-IB)*, currently under development, which has incorporated this multiple-binding approach for metadata/data.

## Married or not married?

**Strategy No. 2: XML Tags don't mean what you think. Define precise meanings.** We have heard many people say “the advantage of using XML is that you can understand the data by looking at the XML record”. First, humans are much better than computers when interpreting ambiguities; second, humans still get it wrong!

Take for example the XML fragment from a personnel record:

```
<maritalstatus>married</maritalstatus>
```

Is the value married from the set { single, married } or from the set { single, married, separated, divorced, widowed }? In the case of a person who is married but not living with his/her spouse, then married would be the right value if the first set was in use, and married would be the wrong value if the second set was in use. Even if the first set were agreed upon, the meaning would still not be clear: does married mean “currently married” (a divorced person is single) or does it mean “ever been married” (a divorced person is considered married).

Although a variety of XML methods (such as schema repositories) can help, XML provides no description of data semantics. We use ISO/IEC 11179 and a combination of its descriptive features: data elements, data element concepts, value domains and conceptual domains. For example, an ISO/IEC 11179 value domain and its concep-

**Characteristic:** concept that plays the role of a determinable in a determining relation. A characteristic is associated with a concept, whereas a property is associated with a subset of objects in the concept's extension.

**Property:** concept that plays the role of a determinant in a determining relation. For example, the characteristic “[has] mass” is a feature of humans, yet one human has the property “[mass is] 80 Kg” and another human has the property “[mass is] 110 Kg”. In this example, the determinable “mass” has a quantifiable determinant (mass measured in Kg). The same determinable could have a different range of determinants, such as a qualitative determinant (thin, fit, obese) or a boolean determinant (true-false, which would be “true” for all humans).

**Property valuespace:** the set of possible values for a property with respect to a characteristic.

tual domain would be used to precisely and unambiguously describe the marital-status feature above, regardless of whether it was implemented as an SQL column, a Java class, or an XML element. ISO/IEC 11404:2007, *Information technology – General purpose datatypes (GPD)*, is used for defining datatypes (ISO/IEC 11404 works well with ISO/IEC 11179).

Thus, the XML schemas specify syntax, while the metadata (ISO/IEC 11179, ISO/IEC 11404) specify the semantics.

## When cold becomes KLD

**Strategy No. 3: Use cross-binding techniques for data interoperability specifications.** The following features must be defined for every kind of data exchanged (see also **Box**):

- Standardized characteristics (e.g. temperature)
- Standardized property values for those characteristics (e.g. a quantitative scale such as degrees Celsius, or a qualitative scale such as cold, cool, warm, hot)
- Standardized property value codes (e.g. a signed 16-bit big endian binary integer, or the codes KLD, COO, WRM, HOT)
- Standardized datatypes for the property valuespace (e.g. an enumerated, ordered state datatype)
- Standardized naming/navigation for the value (e.g. human\_status.body\_temperature).

Without agreement upon these five features for each element of data exchanged, data interoperability might be limited and data exchange might be ambiguous or misunderstood.

As in Strategy No. 2, we store this type of information in our metadata repositories to automate and facilitate data interchange, and this kind of information applies across bindings: SQL columns, C programming language structures, and XML elements.

## Disadvantages rectified

The use of these strategies at specification-time, implementation-time, and run-time help to improve the interoperability of data, including XML data. By incorporating run-time support via a metadata infrastructure (metadata repositories, real-time access to metadata), many interoperability and data transformation operations can be automated or semi-automated. Finally, virtually all of the XML disadvantages are rectified with this metadata-enabled infrastructure. ■