



ISO

International Organization for
Standardization

ISO/IEC JTC 1/SC 32
Data Management and Interchange
WG 3
Database Languages

Project: 1.32.03.05
Title: W3C Liaison Report
Status: For information
Author: Jim Melton, with contributions from Paul Cotton
Abstract: The author is liaison from ISO/IEC JTC1/SC32 to the W3C. This document is intended to inform ISO/IEC JTC1/SC32 of activities of the W3C that are likely to be of particular interest to ISO/IEC JTC1/SC32.

References:

- [XML-WD]** ANSI H2-2002-013 = WG3:VIE-011, (*Working Draft*) *XML-Related Specifications (SQL/XML)*, December, 2001
- [schema-1]** *XML Schema Part 1: Structures*, W3C Recommendation, 2 May 2001 (<http://www.w3.org/TR/xmlschema-1/>)
- [schema-2]** *XML Schema Part 2: Datatypes*, W3C Recommendation, 2 May 2001 (<http://www.w3.org/TR/xmlschema-2/>)
- [queryreq]** *XML Query Requirements*, W3C Working Draft, 15 February 2001 (<http://www.w3.org/TR/xmlquery-req/>)
- [usecases]** *XML Query Use Cases*, W3C Working Draft, 20 December 2001 (<http://www.w3.org/TR/xml-query-usecases/>)
- [queryDM]** *XQuery 1.0 and XPath 2.0 Data Model*, W3C Working Draft, 11 December 2001 (<http://www.w3.org/TR/query-datamodel/>)
- [query]** *XQuery 1.0: An XML Query Language*, W3C Working Draft, 20 December 2001 (<http://www.w3.org/TR/xquery/>)
- [semantics]** *XQuery 1.0 Formal Semantics*, W3C Working Draft, 26 March 2002 (<http://www.w3.org/TR/query-semantics/>)
- [F&O]** *XQuery 1.0 and XPath 2.0 Functions and Operators Version 1.0*, W3C Working Draft, 11 December 2001 (<http://www.w3.org/TR/xquery-operators/>)

- [xpath2]** *XML Path Language (XPath) 2.0*, W3C Working Draft, 20 December 2001 (<http://www.w3.org/TR/xpath20>)
- [xqueryx]** *XML Syntax for XQuery 1.0*, W3C Working Draft, 11 June 2001 (<http://www.w3.org/TR/xqueryx/>)
- [xpointer]** *XML Pointer Language (XPointer) 1.0*, W3C Candidate Recommendation, 11 September 2001 (<http://www.w3.org/TR/xptr>)
- [xinclude]** *XML Inclusions (Xinclude) 1.0*, W3C Candidate Recommendation, 21 February 2002 (<http://www.w3.org/TR/xinclude>)
- [xml1.1]** *XML 1.1*, W3C Working Draft, 13 December 2001 (<http://www.w3.org/TR/xml11>)
- [namespaces]** *Namespaces in XML 1.1*, W3C Working Draft, 3 April, 2002 (<http://www.w3.org/TR/xml-names11>), and *Namespaces in XML 1.1 Requirements*, W3C Working Draft, 03 April 2002 (<http://www.w3.org/TR/xml-names11-req>)
- [xslt2.0]** *XSL Transformations (XSLT) Version 2.0*, 20 December 2001 (<http://www.w3.org/TR/xslt20>)
- [charmod]** *Character Model for the World Wide Web*, W3C Working Draft, 20 February 2002 (<http://www.w3.org/TR/charmod>)

1. W3C Projects Related to SC32

The World Wide Web Consortium (W3C) is a consortium that develops and published specifications (“Recommendations”) related to the World Wide Web. The Recommendations published by the W3C are widely accepted as *de facto* standards.

Among the specifications developed by the W3C are several that are related to the Extended Markup Language known as XML, including several specifications that concern XML-related metadata and querying of XML documents. This report concentrates on the W3C *activity* known as the “XML Activity”, but includes brief mention of one specification from the Internationalization Activity ([charmod]) and one from the Style Activity ([xslt2.0]).

Before examining the individual specifications, readers should be aware that every well-formed XML document has an Information Set (Infoset) that describes the content of every node in the tree corresponding to the document. (In fact, an Infoset is a tree structure that is isomorphic to the XML document with which it is associated, or that it describes.)

1.1 Specifications

The several documents cited above in the References for this paper (other than [XML-WD], of course) are W3C documents that the author of this paper believes to be of significant interest to SC32 participants. (Clearly, SC32 participants will have interest in other W3C documents, but it is beyond the scope of this report to address everything potentially of interest in the W3C’s scope.) The W3C’s practices include a requirement that every Working Group demonstrate their activity through frequent publication of their working drafts (commonly known as the Group’s “heartbeat”). At least once every three months, WGs are expected to publish one or more documents that represent the current status of their development efforts; however, not every document being developed by a WG has to be published in every heartbeat cycle.

1.1.1 XML Schema ([schema-1] and [schema-2])

The XML Schema Recommendation was published by the W3C in three parts. Part 0 is non-normative and is a *primer* that explains many aspects of the other two parts. Part 1, named “Structures”, is arguably the most complex of the three parts. It “defines facilities for describing the structure and constraining the contents of

XML 1.0 documents, including those which exploit the XML Namespace facility”. Part 2, named “Datatypes”, specifies a taxonomy for XML Schema’s “built-in types”, using facilities defined in Part 1.

The facilities defined in Part 1 are intended to allow the definition of metadata for XML documents of significant complexity. This metadata differs in several important ways from the metadata associated with SQL databases. The most critical difference lies in the fact that XML documents can exist independent of any metadata, that such metadata as may be available can be provided in any of several forms (*e.g.*, DTD, or Document Type Definition, XML Schema, and other schema-like formats), and that the metadata to which an XML document conforms is only loosely coupled with the document itself.

Another significant complexity introduced by Part 1 is the notion of complex types, both named and anonymous. The “types” covered by Part 1 are *element types*, which do not correspond terribly well with anything in the SQL language, not even user-defined types. XML Schema types can be subtyped, both by restriction and by extension, while SQL user-defined structured types are subtyped only through extension.

Part 2’s datatypes correspond more reasonably with SQL’s predefined data types. There are rather more XML Schema datatypes than SQL has predefined data types, and XML Schema partitions its datatypes into *primitive*

types and *derived* types. For example, there are three primitive numeric types, called “decimal”, “real”, and “double”, while there are 13 derived numeric types, such as “integer”, “int”, and “nonPositiveInteger”. XML Schema’s derived types are derived from its primitive types by restriction through the use of *facets* that identify (arguably orthogonal) characteristics of primitive types such as precision, scale, and such.

When an XML document has been validated against one or more XML Schemata, the document’s Infoset is augmented with additional information provided during the course of that Schema validation. The augmented Infoset is known as a Post-Schema Validation Infoset, or PSVI. Among other information, a PSVI provides details of the types of each typed node in the XML document’s Infoset.

1.1.2 XPath [xpath2]

XPath is a language for “addressing parts of an XML document”. In particular, it is a language whose syntax bears some resemblance to the syntax of file paths in UNIX filesystems. It is used to “navigate” the content of XML documents. XPath generally makes no use of the information provided by an XML schema, but relies on the document’s Infoset.

XPath 1.0 is among the most widely used of the W3C’s Recommendations, so development of XPath 2.0 has proven to be quite challenging, particularly in terms of balancing the need for 1.0 compatibility against the demand for new features and the desire for consistency with XQuery.

The most prominent features of XPath are its use of *paths* and *steps*, its dependency on *node sets*, its use of predicates (placed in square brackets in path steps), and—perhaps most interesting—its use of *axes* along which document navigation occurs. XPath defines 13 axes (such as “parent”, “child”, “ancestor-or-self”, and “preceding-sibling”) and provides syntactic sugar for several of the axes (allowing a more compact notation).

The most recent revision, [xpath2], is very significantly different from its predecessor working draft, due in no small part to a merger of functionality (and, currently in progress, of text) with the XQuery specification. Reviewers will find little in common between the most recent publication and the preceding one, and are advised to review the document as though no previous version were available.

1.1.3 XQuery ([queryDM], [semantics], [query], [F&O], [usecases], [queryreq], and [xqueryx])

In 1999, the W3C established a new Working Group to pursue specifications for XML Query. The group has faced enormous challenges, including (but hardly limited to) development of yet another data model for XML, development of a typing philosophy, and establishment of formal semantics for the query language’s syntaxes.

The requirements for XML Query are specified in [xqueryreq] and drove the selection of a base language from which to start the design, as well as the on-going efforts to build a comprehensive set of specifications for querying XML documents. Assisting the Working Group in its understanding of the requirements, [usecases] provides a number of use cases for various language features that are thought to be necessary for a successful XML Query language.

The foundation of the XQuery efforts is, naturally, the data model described in [queryDM]. The XQuery data model is shared by XPath 2.0 and, consequently, by XSLT 2.0. The XQuery data model is frequently described as a superset, or extension, of the PSVI. However, in reality, the XQuery data model explicitly omits some PSVI information on the grounds that it is not needed. This has sometimes led to confusion, when other Query-related specifications have assumed that all PSVI information is present—the resolution frequently being the enhancement of the XQuery data model. Most (perhaps all) of the contents of the XQuery data model are made available to the XQuery semantics through the provision of *accessors* that return the value of each attribute. Those accessors are generally unavailable to XQuery application code.

The XQuery syntax is specified in [query] and continues to evolve, although at a relatively slow rate compared with the other specifications. XQuery syntax is strongly based on a language named Quilt, defined in part by the “father of the SQL language”, Don Chamberlin of IBM (the other Quilt initiators being Jonathan Robie and Daniella Florescu).

The [query] document provides the syntax of the language, but only minimal semantics. The formal semantics of XQuery are specified in [semantics] and are presented in several aspects. First, and perhaps most controversial, are the *static semantics* of the language, which primarily concerns itself with the determination of the types of expressions. While this is in many ways analogous to SQL’s syntax rules that determine the data types of expressions, the presentation in [semantics] is (in my humble opinion) needlessly baroque and mathematical in style. On the other hand, it is intended to be, and arguably is, very precise. Somewhat less controversial are the *dynamic semantics*, which is concerned with evaluating expressions at runtime to determine the results of those expressions. Readers should note that “expressions” in XQuery usually have types that are XML Schema types, but they may have static types that cannot be expressed in XML Schema (however, the runtime values of these expressions always have types that can be expressed by XML Schema). The dynamic semantics are expressed in the same notation as the static semantics. It is probably because of the complexity of the typing system (and the fact that it is poorly understood by most XML Query WG participants) that [semantics] got out of sync with the other documents and was only recently republished in an updated form.

[F&O] is a relatively recent specification that provides definitions of a great many functions that are used in the definition of XQuery and that are useful to XQuery application code. Some functions in [F&O] are specified only to assist in the definition of many XQuery (and XPath) operators—for example, the addition operator “+” is defined in terms of a function named “numeric-add”. Other functions in [F&O] are accessible to XQuery code. Some are accessible for both purposes. In general, the functions that are intended primarily for operator definition are defined in a namespace that is identified by the namespace prefix “op:”, while the remaining functions (intended for direct use by applications) are defined in a namespace identified by the prefix “xf:”.

Finally, [xqueryx] is a separate syntax that expresses XQuery in a fully-XML form. It has not been republished in several months largely because of the need to finalize the syntax and semantics of the “readable” form of XQuery, after which translation into the XML syntax is expected to be a fairly straightforward (perhaps even automatable) matter. The specific nature of XQueryX may be as trivial as wrapping tags around an XQuery “statement” (e.g., <xq:xquery>for...let...where...return...</xq:xquery>) or it may be as comprehensive as a fully XML representation of the parse tree of an XQuery “statement”...or something between those two ends of the spectrum. There are strong feelings in both directions.

The most recent publications of [query], [queryDM], and [semantics] differ from their preceding publications in many ways, which I do not find easy to summarize. It is inaccurate to say that [query] and [queryDM] were completely overhauled, but the scope of the changes was significant. By contrast, [semantics] was radically rewritten and does not resemble its previous publication in any useful way. The recent publication of [F&O] was its second appearance, which has been significantly extended (and, in some aspects, reduced) since its first publication in August; the essential nature of [F&O] did not change between publications, although such fundamental restructuring is always a possibility. As a result of these differences, interested parties will find themselves in the unenviable position of having to review all of these documents “from scratch” without being able to rely primarily on exploring the differences from previous editions.

1.1.4 XSL Transformations

[xslt2.0] is a major revision of the XSLT language, taking great advantage of the new facilities available in [xpath2]. The revision to XSLT takes further advantage of new functions defined in [F&O], and also adds new transformation-oriented features. Readers of this report are encouraged to examine [xslt2.0] for new features of interest.

1.1.5 Other XML-related Specifications

The XML pointer language defined in [xpointer] is characterized as “language to be used as the basis for a fragment identifier for any URI reference that locates a resource whose Internet media type is one of” several sorts of XML resources. It is based on XPath (XPath 1.0, that is, since XPath 2.0 is not yet finalized).

The ability to include XML documents into other XML documents is specified in [xinclude].

XML itself is currently undergoing revision. The so-called “Blueberry” revision principally adds support for more recent versions of Unicode by supporting non-alphabetic characters in element and attribute names. The new version is now known as XML 1.1 and is found in [xml1.1].

Namespaces are an important, if poorly understood, aspect of XML processing in which the names of elements, attributes, and other components can be protected from clashes with other elements, attributes, etc. The requirements for a minor revision to the Namespaces specification, as well as a Working Draft addressing those requirements can be found in [namespaces].

Finally, the Internationalization Working Group has been developing a Character Model for use on the World Wide Web. The character model describes the relationships of W3C specifications to Unicode, addresses Unicode normalization issues, and provides other character-related specifications. [charmod] contains the most recent revision.

1.2 Status

XML Schema (all three parts) became a W3C Recommendation in May, 2001. The XML Schema Working Group remains active and is said to be gathering requirements for a new version. Whether the WG produces a version 1.1 or goes directly to a version 2.0 has not yet been announced, although new requirements are being gathered. It is common for the elapsed time between gathering requirements and publication of a Last Call document for a major revision to range between one and three years (minor revisions normally take less time).

XPath 1.0 achieved Recommendation status in November, 1999, and is currently under revision as XPath 2.0. No public statement has been made regarding the planned publication schedule for XPath 2.0, but the state of the document would suggest that it could be released for Last Call some time late in the second half of 2002.

The XQuery-related documents referenced in this paper represent the status of the Working Group’s efforts as of early 2002. The [xqueryreq] document has not changed in some time, because the requirements themselves are widely agreed and generally stable. The [usecases] document changes only incrementally as the group better understands some of the requirements and enhances or corrects use cases to match the requirements.

The XQuery documents continue to be in a state of flux that reflects the complex nature of the technological problems being addressed by the WG. Some public statements have been made that would suggest publication of Last Call for most of the XQuery-related specifications some time late in the second half of 2002, but public skepticism has also been expressed about committing to that schedule.

It is reasonable to expect a “heartbeat” publication of the XPath and XQuery working drafts early in the second quarter of 2002, even though that would comprise a 4-month period following the preceding publication cycle.

[xpointer] recently reached Candidate Recommendation (CR) status after a somewhat turbulent birth. The outcome of the CR period is currently indeterminate and there is not yet an announced decision regarding further progression to Proposed Recommendation (PR) status or regression to an earlier stage (perhaps Working Draft or Last Call).

[xinclude] is currently in its CR period and the outcome of that period currently looks promising.

[xml1.1] is currently in Last Call.

[namespaces] is in the early stages of processing and I have seen only a single public Working Draft.

[charmod] is currently entering its second Last Call period, having received sufficient comments in its first Last Call to justify heavy revisions in style and structure, although the technical content has not been radically revised.

2. SC32 Feedback

While there is no formal liaison relationship between the W3C and SC32 that *explicitly* identifies SQL or XML Query, the Query WG in particular has made it quite clear that feedback from the standardizers of SQL is useful and interesting. (There is a Class C liaison between the W3C and SC32 that appears to concern XML as a whole, as well as the Dublin Core specifications, which are probably of interest primarily to SC32/WG2.) Speaking candidly, it is unlikely in the extreme that any formal written liaison from SC32 to the W3C as a whole would find any audience—indeed, this author would not know how to submit such a liaison other than to the W3C Director (Tim Berners-Lee), who would no doubt be astonished and nonplussed to receive it. Therefore, any liaison that SC32 might wish to send to the Query WG can be transmitted informally or in writing by your humble servant, but expectations of a formal response should be set accordingly low.

End of paper