Subject: Proposed NP to amend ISO/IEC 11404, Language-independent
             Datatypes
To: ISO/IEC JTC1/SC22/WG11
From: US national body
Status: US contribution to 20-22 March 2000 meeting in Dresden, Germany

The US National Body strongly supports this proposed effort to expand the scope of ISO 11404 (LID) with a few minor amendments that would make LID as useful to data-related languages as it now is to classic programming languages. See the below for the general approach conceived.

What Is ISO 11404?
*   A standard for Language Independent Datatypes
*   Intended to bind to virtually all programming languages and similar systems, e.g., databases
*   In use in many environments, including MPEG and XML Schema
*   Many class libraries based on ISO 11404 types
*   JTC1 will make document publicly available

"Why Haven't I Heard About ISO 11404?"
*   Bindings (conformance) are embedded in standards, not products
*   Binding is not visible to most users
*   Isn't the "XYZ committee/consortium" defining datatypes?
*   Since 1994, most datatyping is based on or harmonized with ISO 11404
*   ISO 11404 is the most comprehensive source for standards and datatyping

"What's So Great About ISO 11404?"
*   Very precise, polished standards words
*   Words have been interpreted/reviewed by SC22 working groups
*   Can describe very complex datatypes
*   Compatibility/harmonization with most programming environments

Other Features
*   LID is the "pseudo-code" for datatypes
*   XML is too low level
*   UML is too high level
*   LID can bind to XML, SQL, C/C++, Java, JavaScript

What Is Missing?
*   Support for semi-structured and unstructured data aggregates
         Unknown/unspecified datatyping/navigation
                    Important for effective XML bindings
*   Support for data longevity
                    Extensions: data element and vocabulary
                    Obsolete/reserved elements/vocabulary

Solution Approach [1/6]
* Add keyword "unordered"
* Used in records (aggregates)
* Supports unordered data, e.g.
  <R> <B>y</B> <A>x</A> </R>
  <R> <A>x</A> <B>y</B> </R>
         are the same
• Description: Add the keyword "unordered" as a type qualifier to the "record" type.
  Keyword would allow the elements of the record to appear in any order (may be an
  issue for some programming languages, definitely an issue for environments that have
  "self discovery" or "introspection", like XML). For example, "R : unordered record (X:
  XT, Y: YT)" permits both "X, Y" and "Y, X" as valid instances of R.

Solution Approach [2/6]
* Add keyword "extendable"
* Used in records (aggregates) and enumerations
* Supports extensions to data elements/vocabularies
  • Description: Add keyword "extendable" as a type qualifier to the "record" type and the
  • "selecting" type. This keyword would allow the additional elements in records, while
    still considering the datatype as valid, i.e., an instance of "R : extendable record (X:
    XT, Y: YZ)" that contains the elements X, Y, and Z is still a valid instance of R. For the
  • "selecting" type, "extendable" allows you to consider both open and closed
    vocabularies ("extendable selecting" is an open vocabulary).

Solution Approach [3/6]
* Support keyword "optional"
* Data elements may be optional
* Data elements may occur more than once, e.g., "optional(1..*)"
* Very useful for interoperability and compatibility among proprietary
  vendor solutions (e.g.: phone numbers)
  • Description: Add the keyword "optional". Type qualifier is used within a "record" type.
  • For example, "R : record (optional X : XT, Y : YT)" states that both "X, Y" and "Y" are
    considered valid instances of R.
    Keyword may take a range, such as "optional(0..8)" which states that the element may
    be repeated that many times ... this is necessary in bindings (e.g., XML, SGML) where
    there may be more than one instance of an element.

Solution Approach [4/6]
* Add keyword "extension"
* Used in to identify extensions
* Supports extensions to data elements/vocabularies
  • Description: Add keyword "extension". This type qualifier is used within a "record" or
  • "selecting" type. For example, "R : record extension X: XT, Y: YT)" requires "Y" in all
    instances of R, but "X" may exist (and must be of type XT) and X is considered an
    extension, i.e., the implementation may or may not "process" the element ... well-
    defined extensions are very necessary for long-term data modeling. Rationale: With
  • "optional", "extendable", and "extension", we would have all the necessary features

that closely map into common conceptual models which consider "mandatory data elements" (no keywords), "optional data elements" (use "optional" keyword), and "extended data elements" (using "extendable", "extension", or both).

Solution Approach [5/6]
*   Support keyword "alias"
*   Identify alternative names
*   Very useful for migrating old data structures to new data structures
  •   Description: Add the keyword "alias". This pseudo-type is just a name for another
  •   element. For example, "R : record (X: XT, old_name_of_X: alias X)" allows you to use both "X" and "old_name_of_X" equivalently.

Solution Approach [6/6]
*   Support keywords "obsolete" and "reserved"
*   This technique is very valuable for long-term data maintenance
        (both past and future data structures).
  •   Description: Add the keyword "obsolete". Has no effect on the data processing, but it
  •   will cause a "diagnostic" to be reported when the translation system actually "uses" the type/object attached to "obsolete".
  •   For example "R : record (X: XT, obsolete old_name_of_X: alias X)" will cause a diagnostic to be reported when the system translates/uses "old_name_of_X" (which just maps to using "X"), but the use of "X" causes no harm.
*   Description: Add the keyword "reserved". Similar to "obsolete" except a different diagnostic is reported.

Information Required For ISO/IEC JTC1 NP (New Work Item Proposal)
Title: Amendment N to ISO/IEC 11404
Scope: Providing enhancements for the following:
*   Supporting Semi-Structured and Unstructured data: aggregates with (1) unknown or unspecified, (2) data typing or data navigation/labeling.
*   Supporting data extensibility: identifying data element and vocabulary extensions
*   Supporting data longevity: identifying obsolete and reserved data elements and vocabulary Facilitating XML bindings
Purpose: To provide minor enhancements to ISO/IEC 11404 so that XML and other common bindings are practical within the Language Independent Datatypes framework
Target Date: 2001Q3

Relevant Documents:
*   XML specification
*   XML Schema specification
  •   IEEE 1484.14 Semi-Structured Data Access API specification
  •
Relationship to other International Activities:
*   SC22 programming languages
*   SC32 data management and interchange
*   W3C XML Liaison Organizations:
*   W3C
*   NCITS L8, J11, J16, J22, T2, V1

*   IEEE LTSC
*   Possibly, JTC1/SC36 Need for ISO and IEC coordination:
*   None identified Preparatory Work:
*   Draft Attached: Yes
*   Proposed Project Leader: F. Farance, US
Concerns known patented items?
*   No