

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
Ecma	9.1.1.1		te	<p>It is possible to get XML values whose set of <code>[[InScopeNamespaces]]</code> is inconsistent with the set of namespaces used by that value. For example, this might occur when an element is extracted from its parent.</p> <p>While the specification contains many notes of the form</p> <p>NOTE: The E4X data model does not enforce the constraint: for all <i>x</i> belonging to XML: <i>x</i>.<code>[[InScopeNamespaces]]</code> is an improper superset of <i>x</i>.<code>[[Parent]].[[InScopeNamespaces]]</code>.</p> <p>to allow implementations freedom of representation, it must also be true that the namespaces used by an XML value must always be in the <code>[[InScopeNamespaces]]</code> of that value.</p>	<p>There are four changes necessary to address this issue:</p> <p>(1) In <code>MapInfoltemToXML</code>:</p> <p>Step 6: remove step g and modify the new step g (was step h) to look like:</p> <p>g. For each attribute information item <i>a</i> in the <code>[in-scope namespaces]</code> property of <i>i</i>, except for the attribute information item whose <code>[prefix]</code> property is equal to "xml"</p> <p>i. Map a member <i>ns</i> of <code>x.<code>[[InScopeNamespaces]]</code> to <i>a</i> as follows:</code></p> <ol style="list-style-type: none"> 1. Map <i>ns.prefix</i> to the <code>[prefix]</code> property of <i>a</i> 2. Map <i>ns.uri</i> to the <code>[namespace name]</code> property of <i>a</i> <p>(2) Constrain <code>[[InScopeNamespaces]]</code> of an XML value to include every namespace used by that XML value</p> <p>(3) Change the algorithm of <code>ToXMLString()</code> to emit a <code>xmlns</code> declaration for the default xml namespace rather than generating an prefix for it</p> <p>(4) Change the algorithm in <code>[[AddInScopeNamespaces]]</code> to replace the default namespace.</p>	
Ecma	9.1.1.1, 9.1.1.3		ed	There is redundant code that should be shared between these two sections.	Define an abstract procedure to describe the common logic shared between these two clauses.	
Ecma	9.1.1.2		te	<code>ToString</code> is called unnecessarily, which is inefficient and can have side effects.	Move steps 3 and 4 to before 1 and 2	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
Ecma	9.1.1.2		te	Step 6. GetDefaultNamespace is called much earlier than its one use.	Inline or move down	
Ecma	9.1.1.2		ed	Step 9. isValidName misspelling of isValidName	Replace isValidName with isValueName	
Ecma	9.1.1.3, 9.2.1.3, and elsewhere		ge	<p>Throughout the document, the handling of conditional statements is inconsistent. I.e.</p> <p>The following pattern where there are extra steps with else-after-return and extra returns:</p> <pre> 2 If foo 2(a) If bar, return true 2(b) Else 2(b)(i) blah 2(b)(ii) blah etc. 2(c) Return true </pre> <p>Should be rewritten as</p> <pre> 2 If foo 2(a) If !bar 2(a)(i) blah 2(a)(ii) blah etc. 2(b) Return true </pre>	(see comment)	
Ecma	9.1.1.4		ed	The terminology "shift up" vs. "shift down" is unclear.	Use the terminology: "shift higher" and "shift lower" instead of "shift up" and "shift down", respectively.	
Ecma	9.1.1.13		ed	[[AddInScopeNamespaces]] wrongly plural in 3(d)(iii)	Replace [[AddInScopeNamespaces]] with [[AddInScopeNamespace]]	
Ecma	9.2.1.2		te	Step 2(c)(ii) sets y. [[Parent]] = r where r is the result of [[ResolveValue]] called on x. [[TargetObject]] in 2(a)(i).	To match insertChildAfter, insertChildBefore, prependChild, and setChildren, we should silently	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
				This can result in text parenting text. E.g. <pre>var MYXML = new XML(); MYXML.appendChild(new XML("<TEAM>Giants</TEAM>"));</pre>	do nothing in this case.	
Ecma	9.2.1.2		ed	Step 2(c)(vii)(1) could test r instead of y. [[Parent]], since we know from 2(c)(ii) that they're identical.	(see comment)	
Ecma	9.2.1.2		te	Step 2(c)(vii)(3) what is V. [[PropertyName]]? Should be [[TargetProperty]]	(see comment)	
Ecma	9.2.1.2		te	Step 2(f)(iv, vi). Off-by-one error	Replace (iv) and (vi) with these steps: iv. For j = x. [[Length]] - 1 downto i + 1, rename property j of x to ToString(j + c. [[Length]] - 1) vi. Let x. [[Length]] = x. [[Length]] + c. [[Length]] - 1	
Ecma	9.2.1.2		te	Step 2(g)(iii)V may not be of type XML, but all index-named properties x[i] in an XMLList x must be of type XML, according to 9.2.1.1 Overview and other places in the spec.	Thanks to 2(d), we know V is either a string or an XML/XMLList object. If V is a string, call ToXML on it to satisfy the constraint before setting x[i] = V.	
Ecma	9.2.1.2 Step 2(e)(i, ii), 9.2.1.2 Step 7(e)(i), 9.2.1.3 Step 2(b)(ii)(1)(a)		te	All uses of a. [[Name]] for an attribute a in these sections that pass that QName object to [[Delete]] must pass an AttributeName cloned from a. [[Name]]. The [[Name]] internal property is always a QName instance and never an AttributeName or AnyName instance. But [[Delete]] will not operate on x. [[Attributes]] when given a QName by these sections, so a child could be wrongly deleted instead of the attribute of the same name.	Need to convert QName into an attribute name before calling [[Delete]]	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
Ecma	9.2.1.10		ed	2(a) tests for null after testing for a non-null value	2(a) transpose 2nd and 3rd terms in if condition's disjunction	
Ecma	10.2.1		te	Step 3 is premature, given the early returns in steps 4-7. Unless steps 4-7 should prepend s to their specified return values, which upon testing seems like the right thing! So the errata are that (4-7) do not prepend s to their "Return" results.	Add s+return_value where returns do not already do that.	
Ecma	10.2.1		te	Step 11 seems to make a copy of the in-scope namespace prematurely. Only if Step 12's "If (namespace.prefix == undefined)" test is true does it need the copy, in order to set namespace.prefix.	Move copying closer to use	
Ecma	10.2.1		te	Step 12 seems confused: if namespace.prefix is set to an arbitrary prefix not used by any namespace in the union set, then 12(b)'s "If" condition is always true, and we'll always add the newly-prefixed copy of the namespace found in the in-scope namespaces to namespaceDeclarations.	Remove test make 12.b.i 12.b	
Ecma	10.2.1		ed	Step 17(b)(i) typo: [[GetNamespace]]	Replace with [[GetNamespace]]	
Ecma	10.2.1		te	Step 17(e-f): The XML spec say these need EscapeAttributeValue	Add a step to call EscapeAttributeValue with the attribute value as the argument before concatenating the result	
Ecma	10.2.1		ed	Step 21(a) Typo: indentLevel is wrongly capitalized.	Fix capitalization	
Ecma	10.2.1		te	Step 24(a) It seems a new line character should be concatenated to s before the indentLevel spaces, to put the end-tag on its own line.	Add a step before Step 24(a), which adds line terminator character to s.	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
Ecma	10.2.1.1		te	Step 2(b) is unnecessary and contrary to user expectations because '>' characters do not need to be escaped in element content.	Remove step 2(b).	
Ecma	10.6.1		te	Step 1 uses P, should be s. Step 1 excludes any number values property when it should only exclude unsigned integer valued properties.	Change S to p. Change conversion to ToUInt32()	
Ecma	11.1		te	Step 3(a)(i) n::x given valid Namespace reference n, @n::b, *, etc. => undefined if not found in scope chain. This goes against Editions 1-3 and the implementations that led to the EcmaScript standard, and it's not good human engineering.	Throw ReferenceError if not found	
Ecma	11.3.2		ed	The specification seems to be ambiguous as to what should be returned as section 11.3.2 The typeof Operator says in its text When UnaryExpression evaluates to a value of type XMLList, the typeof operator returns the string "xml", while the table defining the results says: Type Result XML "xml" XMLList "xml"	Return the string "xml", instead of "xml"	
Ecma	11.1.4		te	The grammar for XML initialisers includes markup that is not well-formed XML. A tighter grammar will allow syntax errors to be caught while parsing the program, rather than at runtime.	Replace the grammar for XML initialisers with the following: XMLElement < XMLName XMLAttributes XMLWhitespaceOpt > < XMLName XMLAttributes > XMLElementContent </ XMLName XMLWhitespaceOpt > XMLName { Expression } XMLName XMLAttributes	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
					XMLWhitespace { Expression } XMLAttribute XMLAttributes empty XMLAttribute XMLWhitespace XMLName XMLWhitespaceOpt = XMLWhitespaceOpt XMLAttributeValue XMLAttributeValue { Expression } XMLAttributeValue XMLElementContent { Expression } XMLElementContent XMLMarkup XMLElementContent XMLText XMLElementContent XMLElement XMLElementContent empty	
Ecma	11.2.4		ed	6(a, d, e) should use l[i], not x[i].	Replace references to x[i] with l[i]	
Ecma	11.5.1		te	E4X specs the section 11.5.1 The Abstract Equality Comparison Algorithm states: ... 3. If Type(x) is the same as Type(y) ... c. If Type(x) is Object and x.[[Class]] == "Namespace", return the results of the comparison x.uri == y.uri The comparison in 3(c) is does not verify that y is a namespace value.	That should be changed to c. If Type(x) is Object and x.[[Class]] == "Namespace" and y.[[Class]] == "Namespace", return the results of the comparison x.uri == y.uri A similar treatment should be applied to QName part.	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
Ecma	11.3.1		te	<p>The delete Operator</p> <p>This section does not describe how delete x..@a works, for example, because the Descendants accessor, 11.2.3, does not return a Reference type (because [[Descendants]] for XML and XMLList do not return References, and the final steps of the semantics in 11.2.3 just propagate those return values), and the 11.3.1 Overview lists cases only for Reference types where the base object is XML and XMLList.</p> <p>Ecma-262 Edition 3, 11.4.1, The delete Operator, specifies that delete on a non-Reference type returns true (step 2), so delete x..@a does nothing except evaluate to true.</p>	Throw a TypeError exception of the operand of the delete operator is if type XMLList.	
Ecma	12.1		te	<p>default xml namespace</p> <p>Default namespace is scoped lexically, but not hoisted to the top of function bodies in the same way that var definitions are. This make it hard to use and hard to compile.</p>	In functions that define the default xml namespace, initialize the default xml namespace to the current value of the global default xml namespace, at the beginning of the function block.	
Ecma	12.3		ed	"NOTE The for-each-in statement behaves differently than the for-in statement." Should use "differently from" or "other than".	Replace "differently than" with "differently from"	
Ecma	12.3		ed	off-by-one (too great) step numbering in "The mechanics of enumerating the properties (steps 7 and 7a in the first algorithm, steps 8 and 8a in the second) is implementation dependent."	Change numbering to "6 and 6a" and "7 and 7a", respectively.	
Ecma	13.3.2		te	QName called with zero arguments is not specified don't want localName == "undefined"	Use empty string	
Ecma	13.4.4.1		te	XML.prototype.constructor cannot be accessed because XML [[Get]] does not lookup properties as Object [[Get]]	Make reserved for future use	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
				does. Ditto for XMLList.prototype.constructor. This may not be the case in the future, creating a potential compatibility issue if implementations or program give these properties values.		
Ecma	13.4.4.6		te	<p>XML.prototype.child Step 1(a) depends on the [[Get]] method of x returning an XMLList of all children, and 1(b) depends on [[Get]] of that XMLList, called with index P, returning a list containing that one child, with [[TargetObject]] referring to x and [[TargetProperty]] presumably being null.</p> <p>But [[Get]] on an XMLList, 9.2.1.1, does not return a list containing the indexed child, given a property index P -- it delegates to Object [[Get]]. Per 9.2, getting an indexed property from an XMLList will return undefined if P >= x.length(), otherwise it will return just the indexed child, not wrapped in a XMLList.</p> <p>This contradicts the wording in the Overview ("If P is a numeric index, the child method returns a list"), but not the example.</p>	<p>Change P to propertyName.</p> <p>Step 2 set temporary to result of [[Get]]</p> <p>Add Step 3 to return temporary converted to XMLList.</p>	
Ecma	13.4.4.31		ed	Step 9 "QNames" misspelled as "Qnames".	Replace "Qnames" with "QNames".	
Ecma	13.5.4.4		te	XMLList.prototype.child seems to be missing at least a 'return m' step at the end.	Add a Step 3, return m	
Ecma	13.5.4.9		ed	Step 2 typo: "Returnt" should be "Return".	Replace "Returnt" with "Return"	
Ecma	13.5.4.16		te	Steps 1 and 3 specify undefined return, not null as is done for XML.prototype.parent(), making them unnecessarily inconsistent.	Return null from XMLList.prototype.parent() in the case that there is no parent.	
Ecma	General		ge	Conformance section is missing. In particular it is not	Add a conformance section, and add to that	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.

Template for comments and secretariat observations

Date: 2005/01/22

Document: ISO/IEC DIS 22537

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
				clear how implementers can extend E4X.	conformance section the constraint that implementeors may not add to the set of methods of XML.prototype and XMLList.prototype	
Ecma	General		ge	ToXMLName spelled as ToXmIName	Replace globally ToXmIName with ToXMLName	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial

NOTE Columns 1, 2, 4, 5 are compulsory.