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	Response on each comment submitted

Ecma	9.1.1.1		te	It is possible to get XML values whose set of [[InScopeNamespaces]] is inconsistent with the set of namespaces used by that value. For example, this might occur when an element is extracted from its parent.
				
				While the specification contains many notes of the form
				
				NOTE: The E4X data model does not enforce the constraint:
				    for all x belonging to XML:
				    x.[[InScopeNamespaces]] is an improper superset of
				    x.[[Parent]].[[InScopeNamespaces]].
				
				to allow implementations freedom of representation, it must also be true that the namespaces used by an XML value must always be in the [[InScopeNamespaces]] of that value.
					There are four changes necessary to address this issue:
					
					(1) In MapInfoItemToXML:
					
					Step 6: remove step g and modify the new step g (was step h) to look like:
					
					g. For each attribute information item *a* in the [in-scope namespaces] property of i, except for the attribute information item whose [prefix] property is equal to "xml"
					    i. Map a member *ns* of *x*.[[InScopeNamespaces]] to *a* as follows:
					        1. Map *ns.prefix* to the [prefix] property of *a*
					        2. Map *ns.uri* to the [namespace name] property of *a*
					
					(2) Constrain [[InScopeNamespaes]] of an XML value to include every namespace used by that XML value
					
					(3) Change the algorithm of ToXMLString() to emit a xmlns declaration for the default xml namespace rather than generating an prefix for it
					
					(4) Change the algorithm in [[AddInScopeNamespaces]] to replace the default namespace.
						Accept. fix mapinfoitem and remove notes and unwinding the parent chain to find inscope namespaces.
						
						Keep 1, 3 and 4. Add 5 to remove notes and while loops that enforce the constraint in 2.
						
						In 9.1.12 specify that the replaced child inherits the children.

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.	This comment is technical. Defer to edition 4
Ecma	9.1.1.2		te	ToString is called unnecessarily, which is inefficient and can have side effects.	Move steps 3 and 4 to before 1 and 2	Accept

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.

ISO electronic balloting commenting template/version 2001-10

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	Response 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	Accept
Ecma	9.1.1.2		ed	Step 9. isValueName misspelling of isValidName	Replace isValidName with isValueName	Accept

Ecma	9.1.1.3, 9.2.1.3, and elsewhere		ge	Throughout the document, the handling of conditional statements is inconsistent. I.e.
				
				The following pattern where there are extra steps with else-after-return and extra returns:
				  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
				
				Should be rewritten as
				
				  2 If foo
				  2(a) If !bar
				  2(a)(i) blah
				  2(a)(ii) blah
				  etc.
				  2(b) Return true
					(see comment)	Accept

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.	Accept
Ecma	9.1.1.13		ed	[[AddInScopeNamespaces]] wrongly plural in 3(d)(iii)	Replace [[AddInScopeNamespaces]] with [[AddInScopeNamespace]]	Accept
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	Accept

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.

ISO electronic balloting commenting template/version 2001-10

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	Response on each comment submitted

				This can result in text parenting text. E.g.
				
				var MYXML = new XML();
				MYXML.appendChild(
				    new XML("<TEAM>Giants</TEAM>"));
					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)	Accept
Ecma	9.2.1.2		te	Step 2(c)(vii)(3) what is V.[[PropertyName]]? Should be [[TargetProperty]]	(see comment)	Accept

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
						Accept

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.	Accept
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]]	Accept

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.

ISO electronic balloting commenting template/version 2001-10

Template for comments and response by the Editor on the comments submitted

Date: 2005/01/27	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	Response 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	Accept
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.	Accept
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	Accept
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	Accept
Ecma	10.2.1		ed	Step 17(b)(i) typo: [[GetNamspace]]	Replace with [[GetNamespace]]	Accept
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	Accept. Replace "fo" with "of"
Ecma	10.2.1		ed	Step 21(a) Typo: indentLevel is wrongly capitalized.	Fix capitalization	Accept
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.	Accept

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.

ISO electronic balloting commenting template/version 2001-10

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	Response 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).	Reject. Continue to escape for parallelism with '<'.
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()	Accept
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	Accept

Ecma	11.3.2		ed	The specification seems to be ambigious 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 "xmllist", while the table defining the results says:
				  Type Result
				  XML "xml"
				  XMLList "xml"
					Return the string "xml", instead of "xmllist"	Accept

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
						Accept.

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.

ISO electronic balloting commenting template/version 2001-10

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	Response 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]	Accept

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 comparision 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.
						Accept

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.

ISO electronic balloting commenting template/version 2001-10

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	Response on each comment submitted
Ecma	11.3.1		te	The delete Operator\n\nThis 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.\n\nEcma-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.	Throw a TypeError exception of the operand of the delete operator is if type XMLList.	Accept
Ecma	12.1		te	default xml namespace\n\nDefault 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.	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.	Accept
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"	Accept
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.	Accept
Ecma	13.3.2		te	QName called with zero arguments is not specified don't want localName == "undefined"	Use empty string	Accept
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	Accept. Add note that says the constructor property can

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.

ISO electronic balloting commenting template/version 2001-10

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	Response 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.		be write only in practice.
Ecma	13.4.4.6		te	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. 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. This contradicts the wording in the Overview ("If P is a numeric index, the child method returns a list ...."), but not the example.	Change P to propertyName. Step 2 set temporary to result of [[Get]] Add Step 3 to return temporary converted to XMLList.	Accept with modifications that: 1.b. Set temporary to result of get 1.c. If temporary is undefined then return empty list
Ecma	13.4.4.31		ed	Step 9 "QNames" misspelled as "Qnames".	Replace "Qnames" with "QNames".	Accept
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	Accept
Ecma	13.5.4.9		ed	Step 2 typo: "Returnt" should be "Return".	Replace "Returnt" with "Return"	Accept
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.	Reject. Returning undefined is the correct behavior
Ecma	General		ge	Conformance section is missing. In particular it is not	Add a conformance section, and add to that	Accept

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.

ISO electronic balloting commenting template/version 2001-10

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**	**Response** 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 ToXmlName	Replace globally ToXmlName with ToXMLName	Accept

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.

ISO electronic balloting commenting template/version 2001-10

Template for comments and response by the Editor on the comments submitted

Date: 2005/01/27	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	Response on each comment submitted
US	8		te	Introduction of XML markup at the lexical level is a significant deviation in the underlying structure of an ECMAScript program and is inconsistent with what is done in other similar languages. Allowing XML markup at the lexical level allows the writing of straight XML text in an easier fashion as quotes are not needed around the text and multiline text does not need concatenation at the end of each line. (The example in 13.4.4.2 clearly shows this.) However, this change is a major shift in the lexical structure of the language and does not help in the writing of XML text with embedded data from variables. The "ToXML" operator provides sufficient functionality that XML markup does not need to be shoved into the lexical structure of the language.	Remove XML markup from the lexical level of the standard. A new form of XML literal, which could be multiline, could be added to the standard; However, it must have a clearly defined format with a beginning and end that does not change with the XML text being represented. Having such a literal would make processing the language much easier and error checking much better.	Rejected, see email comment. Ask rex if he has any more insight on this comment.
US	general		te	The document should use "shall" instead of "must" to be consistent with ISO/IEC phraseology.	Change all "must" verbs to "shall" when used to specify mandatory requirements; use "may" when used to specify permissiveness ("is permitted to")	Agreed
US	4		ed	Normative references should be clause 2, definitions clause 3, conformance clause 4, and everything else clause 5 and onward.		Agreed
US	3	p 1	ed	Change "must" ==> "shall"		Agreed
US	3	p 3	ed	Change "is permitted to provide" ==> "may"		Agreed
US	5		ed	Move Clause 5 to an informative annex.		Reject, but marking as informative
US	6		ed	Move Clause 6 to an informative annex.		Reject, but marking as informative
US	general		te	In the cases where "type" means "datatype", replace "type" with "datatype" to be consistent with JTC1 terminology.		Define type in the definition section

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.

ISO electronic balloting commenting template/version 2001-10

Template for comments and response by the Editor on the comments submitted | Date: 2005/01/27 | 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	Response on each comment submitted
US	10.4.1, general		ed	Change "var l = new XMLList" ==> "var List = new XMLList". The problem is that the lower case ell looks like a number 1 (one). So just spell out "List" instead of "l".	Note: This change (changing "l" ==> "List") is necessary in several places within the document.	Accept. E.g. var list = new List
US	13.5.4.3		te	What does "!!!1" mean? Is this a typo?		Not an issue. Only in PDF
US	14		te	The sentence "An implementation may define behaviour other than throwing a TypeError exception for the ToXML function and ToXMLList function when they are called with an argument of type Object" does not make sense because "an implementation may define behaviour" doesn't make sense.	Possibly the following was intended: "If the ToXML function or the ToXMLList function are called with an argument of type Object, then the error thrown is implementation-defined. Note: The implementation-defined behaviour may include throwing the TypeError exception."	Reject, the proposed wording requires that implementations throw an exception, which is not the case. Adding reference to section 16 of edition 3 to clarify. Replace controversial wording with "an implementation may choose to provide behaviour"

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.

ISO electronic balloting commenting template/version 2001-10

Template for comments and response by the Editor on the comments submitted					Date: 2005/01/27	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	Response on each comment submitted
JP	Overall		ge	Document style should be conforming to ISO/IEC standard	Change the document style to conforming to ISO/IEC standards	Reject. Not required for first edition of fast-track specifications
JP	Overall		ge	Clause for Definition of terminology is missing.	Add a clause for Definition of Terms.	Agreed
JP	3		te	Conformance definition given in Clause 3 is incomplete. Currently, the conformance to ISO/IEC 16262 is not required, but obviously the conformance to E4X without the conformance to ISO/IEC 16262 is not possible.	Add a requirement of ISO/IEC 16262 to Clause 3.	Agreed
JP	4		Te	Some references are not newest versions	Change the references to the newest versions.	Reject. Defer until edition 4 because this draft and implementations depend on earlier standards
JP	4		Te	References that are not actually cited should be moved to Informative.	References that are not actually cited should be moved to Notes.	Agreed. If reference is in a normative section then it goes in the normative references, otherwise it needs to be informative Add W3C XSLT as an informative references Change 4 to reference and add 4.1 normative and 4.2 informative
JP	4		Te	ISO/IEC 16262 must be referenced.	Add a reference to ISO/IEC 16262:2001	Agreed. Change to ISO reference.
JP	4		Te	References to Unicode Technical Reports #8 and #15 should be deleted when referring to Unicode 4.0 is added.	Change the Unicode 2.0 reference to Unicode 4.0 and delete the references to Unicode Technical Reports #8 and #15.	Reject. Defer until edtion 4, because of dependency of edition 3 on Unicode 3 support
JP	5		Ge	This must be Informative.	Declare informative, or move somewhere else.	Agreed. Add sentence to

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.

ISO electronic balloting commenting template/version 2001-10

Template for comments and response by the Editor on the comments submitted | Date: 2005/01/27 | 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	Response on each comment submitted
						declare section as informative.
JP	6		Ge	This must be Informative.	Declare informative, or move somewhere else.	Agreed. Add sentence to declare as informative.
JP	10.1.1	1st Paragraph	Te	XML encoding handling must be specified.	Add specific handling to the phrase, "ToString returns an XML encoded string representing the entire XML object".	Remove text "an XML encoded" so sentence reads "ToString returns a string representing the entire XML object,"
JP	14		ed	Period missing - E4X extends the list of erros	Add period.	Agreed
JP	14		ed	Capitalization missing – implementations are not….	Capitalize as head of sentence.	Agreed

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.

ISO electronic balloting commenting template/version 2001-10