

[Draft] Minutes of the Teleconference of

ISO/IEC JTC 1/SC 34/WG4, 2013-08-13

Rex Jaeschke (rex@RexJaeschke.com)

2013-08-15

1. Opening remarks

The meeting started at 13:10 GMT. The convener, Murata-san, welcomed everyone to the 54th teleconference of WG4.

2. Roll call of delegates

The following members were present during part or all of the meeting:

Name	Affiliation	Employer/Sponsor
Makoto Murata	WG4 Convener, JP	International University of Japan
Rex Jaeschke	Ecma, Project Editor	Consultant
Isabelle Valet-Harper	Ecma (TC45 chair)	Microsoft
Caroline Arms	Ecma	Library of Congress
John Haug	Ecma, US	Microsoft
Chris Rae	Ecma	Microsoft
Francis Cave	GB	Francis Cave Digital Publishing

Present were 7 people representing 3 NBs, and 1 liaison.

3. Adoption of the agenda

The agenda (SC 34 [N 1927](#)) was adopted as published.

4. Administration

Approval of Previous Meeting Minutes [WG4 N 0262]

The draft minutes were approved, as circulated.

Outstanding Action Items

1. John H. to report re the addition of Digital Signature support for OPC. **Pending**
2. Jesper to write a Wiki page on the Assembla site
([https://www.assembla.com/spaces/IS29500/wiki/Package-level_encryption_\(OPC\)](https://www.assembla.com/spaces/IS29500/wiki/Package-level_encryption_(OPC))) proposing how to move this idea forward. **Pending**
3. Murata-san will revise his proposed resolution to DR 11-0010 to include multi-column and vertical writing. **Pending** (depends on the resolution of DR 11-0008)
4. Rex will distribute the latest MCE draft. Separately, he will circulate emails with proposals to review the new MCE Scope text; Murata-san's proposed new Terms and Definitions, and related terms; and John's proposed text rewrites, in an attempt to get some/all of these proposals adopted by consensus via email or at the next teleconference. **Done**
5. Rex will ask Kimura-san about the state of the WG4 document site and also when old/pre-LiveLink SC 34 documents might be available. **Done** (The ITSCJ web server should be running again in Mid-August.)

Report from the WG4 Secretariat

The following NBs and liaisons have registered delegates to WG4: BR, CA, CH, CI, CN, CZ, DE, DK, Ecma, FI, FR, GB, IN, IT, JP, KR, NL, NO, OASIS, PL, US, W3C, XML Guild, and ZA. All requests for additions, deletions, and changes to the delegate list should be sent to the WG4 Secretariat (rex@RexJaeschke.com).

The WG4 email list is e-SC34-WG4@ecma-international.org. The document repository is at http://lucia.itscj.ipsj.or.jp/itscj/servlets/ScmDoc10?Com_Id=w4.

5. Revising Part 3 (Semantics of Markup Compatibility and Extensions)

We looked at all the MCE-related email threads since the previous meeting.

MCE Revision Issue: Correcting the Scope Clause

Rex Jaeschke, 2013-07-22:

One of the action items I took from the Bellevue review of the MCE spec was to propose new wording for the Scope Clause, which was tied to Office Open XML documents, but should not be. After drafting some words and reviewing them with Francis and John H., I have the following proposal:

Replace:

“This Part of ISO/IEC 29500 describes a set of conventions that are used by Office Open XML documents to clearly mark elements and attributes introduced by future versions or extensions of Office Open XML documents, while providing a method by which consumers can obtain a baseline version of the Office Open XML document (i.e., a version without extensions) for interoperability.”

With:

“This Part of ISO/IEC 29500 describes a set of conventions that can be used by any valid XML document to mark clearly elements and attributes representing features introduced by future versions or extensions of the document format, while providing a method by which consumers can obtain a baseline version of that document (i.e., a version without extensions) for interoperability.”

Jim Thatcher, 2013-07-22:

This change looks good to me.

Francis Cave, 2013-07-22:

Yes, works for me too.

Gareth Horton, 2013-07-22:

Any objection to inserting "those" as below? It seems to read a little strangely: "... to mark clearly those elements and attributes representing features introduced by future versions or extensions of the document format ..."

Makoto Murata, 2013-07-23:

I think that "valid" should be deleted [from "any valid XML document"], since MCE is applicable to markup specifications that lack DTDs or schemas.

Dennis Hamilton, 2013-07-23:

It could be more clear who this provision applies to. This seems to apply to producers of XML documents. If it "can be used in any XML document ...", that might be over-reaching.

It would appear to require an agreement among XML document producers and consumers that the occurrence of MCE will be treated properly. One important case of such agreement is when the occurrence of MCE is an explicit interoperability provision in the specification of an XML-based format for some class of applications.

Makoto Murata, 2013-07-23:

Dennis, Having seen you comment, I tried to reformulate the proposal. While doing so, I start to feel that I am describing too much details in the scope.

I would say that even the original is too detailed to provide a high-level overview.

W3C has established a nice framework for versioning and evolvability of markup specifications. A nice introduction is with <http://www.xml.com/pub/a/2004/10/27/extend.html?page=1> and it provides links to relevant W3C documents.

I tried to reformulate the scope on the basis of the W3C framework.

Here is my proposal:

"This Part of ISO/IEC 29500 defines a set of conventions for forward compatibility of markup specifications. These conventions allow XML documents created by applications of later versions or extensions to be handled by applications of earlier versions."

Dennis Hamilton, 2013-07-23:

Perfect! Thank you.

Teleconference, 2013-08-13:

We agreed to adopt Murata-san's final proposal, as highlighted above.

MCE Revision Issue: §6, "Overview"

Rex Jaeschke, 2013-07-22:

A few days ago, I posted to this email list, WG N 0264, "29500-3 (MCE) Revision, WD0.91". It contains proposed changes by John H. based on an action item he took at the Bellevue F2F meeting. This action item was: Rework this so we don't appear to be introducing any new terms. Use plain English.

Francis Cave, 2013-07-22:

The edits to §6, "Overview", are mostly OK, I think, but I'm concerned about the wording of the final bullet point. Here it is with the changes applied:

"Application-defined extension elements specify a method for defining extensibility points within the markup specification. This allows markup producers to introduce new features scoped to particular elements within the markup specification. Markup Consumers can disregard these self-contained blocks of additional features."

I think that the first two sentences are OK, but the final sentence looks wrong to me. Markup Consumers MAY be able to disregard these extension elements, but only if the markup specification permits a Consumer to disregard them. The way that ext is specified in Part 1 means that it must be preserved if not understood, so they may never be disregarded. See Part 1, §18.2.7.

John Haug, 2013-07-23:

Good comment. What I think I was getting at was that consumers that don't understand the extension can safely ignore it. The ext definition in SpreadsheetML requires preserving unknown extensions (18.2.7), the ones in PresentationML (19.2.1.11) and DrawingML (20.1.2.2.14, 21.2.2.62) don't.

Perhaps the best thing for Part 3 is to omit that last sentence. That paragraph is intended to give a rough informative overview of the features in MCE, so saying anything about how they work (as opposed to generally what they do) is probably not a good idea.

Makoto Murata, 2013-08-13:

I think that the 2nd bullet in the 2nd itemized list (which is about ProcessContent) requires some wordsmithing. It might be a good idea to convert this bullet to a sub-bullet of the 1st bullet (which is about ignorable namespaces).

Makoto Murata, 2013-08-13:

[Re John's posting of 2013-07-23] I think that this para should mention preserving rather than ignoring. Ignorable namespaces are never preserved, but app-defined ext elements can be (but need not to be) preserved.

How about:

"Application-defined extension elements specify a method for defining extensibility points within the markup specification. Markup consumers are expected to be ready to encounter any extensions at these points. In particular, they might preserve extensions even if they cannot handle them."

Teleconference, 2013-08-13:

We agreed to adopt the idea behind John's latest proposal made during the call on the understanding that he'll post a final draft soon.

John Haug, 2013-08-13:

From the discussion on the call, how about this: "Application-defined extension elements specify a method for defining extensibility points within the markup specification. This allows producers to introduce new features scoped to particular elements within the markup specification. Consumers can disregard these self-contained blocks of additional features and they might preserve extensions even if they cannot handle them."

John Haug, 2013-08-13:

[Re Murata-san's posting of 2013-08-13] From the call:

Elements [[ADD: in ignorable namespaces]] may be marked for [[ADD: their content to be processed | DELETE: processing]] even if they would otherwise be ignored. This allows producers to prevent loss of content nested within ignored content when processed by consumers that do not understand the ignorable namespace.

John Haug, 2013-08-13:

My late homework. From the further discussion on the call, how about this. This incorporates the points raised about preserving extensions and noting that before the words about disregarding them. I also made a few cleanups.

Existing:

“Application-defined extension elements specify a method for defining extensibility points within the markup specification. This allows producers to introduce new features scoped to particular elements within the markup specification. Consumers can disregard these self-contained blocks of additional features.”

Proposed:

“Application-defined extension elements specify a method for defining extensibility points within a markup specification. This allows producers to introduce new self-contained blocks of additional features scoped to particular elements. Consumers might preserve extensions even if they are ignored and cannot be handled.”

I suspect there will be a question about “ignored” in the last sentence. I tried using disregard to avoid using the term *ignored* since we have a feature for ignorable namespaces, but that caused confusion. I'd like to get across that consumers can safely ignore extensions if they're not understood. If someone can think of a better word, please suggest it. I'm fine with either ignored or disregarded since this is all an informative overview.

Francis Cave, 2013-08-14:

I'm not so keen on the phrase “blocks of additional features”. A “block” is something fairly concrete, whereas a “feature” is definitely an abstraction.

I would prefer “blocks representing additional features”.

One way around the ignore/disregard problem would be to leave out both, e.g., “Consumers might preserve extension elements even if they are unable to handle them in any other way.”

[Editor: I made John's proposed fix re Murata-san's comment on "the 2nd bullet in the 2nd itemized list". I also adopted John's final proposal for the final bullet text even though Francis raised some doubts in a later email (see above). I noted Francis' concern in a comment in the new draft. We can review that in Delft.]

MCE Revision Issue: §10.6, "Justification of Ignorable Foreign Children of AlternateContent"

Rex Jaeschke, 2013-07-22:

A few days ago, I posted to this email list, WG N 0264, "29500-3 (MCE) Revision, WD0.91". It contains proposed changes by John H. based on an action item he took at the Bellevue F2F meeting. This action item was: Needs wordsmithing – this was an e-mail John sent during the London 2012 meeting to capture some discussion.

Francis Cave, 2013-07-22:

I would suggest the following text for this paragraph:

"The way that AlternateContent is specified to contain only Choice and Fallback elements from the Markup Compatibility namespace (see §8.5) prevents the use of other Markup Compatibility elements that would otherwise allow extension of AlternateContent in future versions of MCE. For this reason, the specification of AlternateContent allows elements from Ignorable namespaces as children of AlternateContent. Any MCE processor that encounters a child element of AlternateContent that is in the namespace of an intended future extension of MCE will not fail to process the document, provided the namespace of the extension element is ignorable, because it will discard all elements in ignorable namespaces that are not understood before making a selection between the remaining Choice and Fallback elements."

I think that a re-write along these lines is clearer than attempting to revise the existing paragraph word-by-word.

Francis Cave, 2013-08-13:

The addition of the first sentence from the original text makes it necessary to remove the later sentence that says the same thing. I've also done some wordsmithing to correct the spelling of AlternateContent and to avoid use of the term "extension element", which has a special meaning.

"Ignorable elements are allowed as child elements of AlternateContent to allow for future extensions to this construct. If AlternateContent were specified to contain only Choice and Fallback elements from the Markup Compatibility namespace (see §8.5), this would prevent the use of other Markup Compatibility elements that

would allow extension of AlternateContent in future versions of MCE. Any MCE processor that encounters a child element of AlternateContent that is in the namespace of an intended future extension of MCE will not fail to process the document, provided the namespace of this child element is ignorable, because it will discard all elements in ignorable namespaces that are not understood before making a selection between the remaining Choice and Fallback elements.”

Teleconference, 2013-08-13:

We agreed to adopt Francis’ final proposal, as highlighted above.

MCE -- terms and definitions

Teleconference, 2013-08-13:

We agreed to adopt Murata-san’s edits, as they existed in WD0.91.

Caroline Arms, 2013-08-13:

Extracted from the message I sent yesterday with primarily editorial suggestions -- my preliminary thoughts on definitions brought up by Murata-san. Note that I wrote these before I understood the ISO policy that requires replaceability in the text, but I don't think that invalidates these suggestions.

On the call, I suggested that the definitions should be revisited by Murata-san and Francis as they revise sections 7 and 9.

Markup consumer (and Markup producer): I agree with Murata-san that “and further conforms to the requirements of a markup specification” is not relevant and should be dropped from the definition of markup consumer. Similarly markup producer definition should probably be adjusted to “tool that can generate a markup document that conforms to a markup specification.” That is, it is the document that must conform, not the tool.

Markup specification: When you look at “markup specification”, you suddenly see that this seemingly generic phrase is defined as specific to “this part of ISO/IEC 29500.” I would prefer the use of a term that included reference to MCE in some way. That is, if the term really needs to be specified so tightly. Personally, I can see no reason for the limitation to this part of ISO/IEC 29500 -- but someone else need to check that out. In places where the discussion is specific to MCE use, that seems clear to me from the context.

Makoto Murata, 2013-08-14:

Caroline, “Markup specification” in MCE is a special term. It includes a set of expanded names, which are declarations of application-defined extension elements.

Francis Cave, 2013-08-14:

Murata-san, Are you confusing “markup specification” with “markup configuration”? The special term in MCE is surely “markup configuration”. My understanding is that “markup specification” is the specification of the application of MCE (e.g. OOXML).

DRAFT 0.91 of the MCE Spec

Makoto Murata, 2013-08-11:

I am going to send a few more e-mails about this draft, but this mail proposes two minor changes.

1. Move the only para in §7.1 after the itemized list in §10.1, and remove Clause 7.
2. In §10.5 (Step 4), add a note: [Note: With the exception of those in application-defined extension elements, elements and attributes in the output document belong to understood namespaces.] end note]

Teleconference, 2013-08-13:

We agreed to move the only paragraph in §7.1 to the end of §10.1, and to remove the §7.1 heading”, but to keep the empty Clause 7 with a place-holder for us to consider what, if anything, should go there. We also agreed to add the proposed note text to the existing, penultimate note in §10.5.

Proposed text for Clause 9

Makoto Murata, 2013-08-13:

Here is a proposed rewrite of Clause 9. It is based on Francis' wording but has been modified and further expanded by two examples. Thanks Francis.

9. Extension elements defined by a markup specification (informative)

A markup specification that uses Markup Compatibility elements and attributes to allow extensions in namespaces other than those defined by the markup specification may also define one or more specific

extension elements in the namespaces that it defines. [*Note: If the markup specification includes a schema, any extension elements would normally be constrained by the schema to occur only in specific markup contexts. end note*].

If an MCE processor is configured to recognize extension elements, it preserves them together with their attributes and contents. See Clause 9 for details.

[*Example:*

<https://subversion.assembla.com/svn/IS29500/trunk/Part3/TestData/ExtensionElements/example1.xml>

In this example, the e1:foo element contains the unknown:foo element.

Suppose that a markup configuration contains an expanded name ("<http://www.example.com/e1>", "foo") and that an application configuration does not contain "<http://www.example.com/unknown>".

Then, the element e1:foo is an application-defined extension element.

Although the unknown:foo element does not belong to an understood or ignorable namespace, the MCE processor preserves it and does not report any errors. *end example*]

[*Example:*

<https://subversion.assembla.com/svn/IS29500/trunk/Part3/TestData/ExtensionElements/example2.xml>

In this example, Markup Compatibility elements and attributes occur within the extensionElement element, which is the only child of the root element example. Suppose that a markup configuration contains an expanded name ("<http://www.example.com/e1>", "extensionElement").

Then, the MCE processor preserves the extensionElement element.

Therefore, MCE elements and attributes within it, namely mce:Ignorable="i1", mce:ProcessContent="i1:bar1", mce:MustUnderstand="e1", mce:AlternateContent, mce:Choice, and mce:Fallback, appear in the output document. *end example*]

After receiving the output of an MCE processor, application programs may further invoke an MCE processor to handle Markup Compatibility elements and attributes within extension elements.

Francis Cave, 2013-08-13:

A couple of comments:

1. Your proposed text (especially the first sentence of the second paragraph, and the final paragraph) defines an extension element somewhat more narrowly than I had proposed in my draft text. Yours is closer to what is currently in OOXML, which may be a good thing (it shouldn't break any OOXML implementations), but I still wonder whether the MCE spec should be quite so proscriptive about the processing of extension elements or should instead leave it to the markup specification to define how MCE processors should handle extension elements in each specific case. Putting it another way, your proposed Clause 9 seems to be saying something normative about the processing of extension elements, although you have made it clear in the Clause heading that the Clause is informative.
2. At the end of the second paragraph you have a cross-reference to "Clause 9". This is self-referential, so clearly wrong. What is meant?

Makoto Murata, 2013-08-13:

[Re 1.] I rather think that your wording states too much about application programs. I think that we should limit our concern to behaviours of MCE processors and try to avoid describing behaviours of application programs.

MCE processors simply pass application defined extension elements to application programs. The behaviours of MCE processors have to be very clear for interoperability.

My 2nd para may look normative at a first glance, but it is not. It just gives a high-level overview without providing details. Details are provided in the itemized lists in Steps 1 and 2.

[Re 1.] Oops. Clause 10, "Semantic Definitions and Reference Processing Model".

Francis Cave, 2013-08-13:

Your second paragraph says:

"If an MCE processor is configured to recognise extension elements, it preserves them together with their attributes and contents."

I think what you are saying is that an MCE processor that is configured to recognise extension elements will ALWAYS preserve them. In other words, "recognise" in this context means "preserve". If this is ALWAYS the case,

that sounds to me like a normative provision. If it is up to the application to specify this (as I thought we had agreed), it may not always be the case, in which we should not imply that in the MCE spec.

Makoto Murata, 2013-08-13:

Perhaps, the problem is the ambiguity in our terminology: markup consumer , MCE processor, and application program. We might want to make this point clear in Clause 7.

[Re Francis' 2nd point] Exactly. But I am talking about the MCE processor. I am not talking about markup consumers.

[Re "recognise" in this context means "preserve"] Well, I meant that the markup configuration, which is a set of expanded names, contain the expanded names of application-defined extension elements.

To me, markup consumers and MCE processors are very different.

Requirements on MCE processors have to be very clear. Their behaviours are completely predictable. But requirements on markup consumers are much more predictable.

Francis Cave, 2013-08-13:

I understand the distinction between MCE processors and markup consumers.

[Re predictability] I think you mean that the requirements on markup consumers are much LESS predictable.

But if the behaviours of MCE processors are completely predictable, should these not be specified normatively somewhere? If the behaviour of an MCE processor with respect to extension elements must be completely predictable, this behaviour must be specified normatively. Where is it to be specified?

Either in the MCE spec or in the markup spec.

I think that my confusion is between the specification of extension elements (in the markup spec) and the specification of MCE processor behaviour (in the MCE spec). If we can completely specify MCE processor behaviour with respect to extension elements, this should be done normatively in the MCE spec. If we cannot - i.e. it depends upon how the extension elements are specified in the markup spec - we cannot say anything normative about it in the MCE spec and we probably shouldn't make too many assumptions about it.

Teleconference, 2013-08-13:

After a lengthy discussion, we agreed that regarding required behaviors, we need to distinguish clearly markup consumers and MCE processors and to clearly define their relationship. To be continued online and in Delft.

Caroline's edits and comments in mail "DRAFT 0.91 of the MCE Spec"

[Editor: I responded to these by email on 2013-08-14.]

Action: Rex will produce MCE WD 0.92 in advance of the Delft F2F meeting.

6. Revising Part 2 (Open Packaging Conventions)

There was no discussion.

7. Other Business

We thanked Microsoft and John Haug for hosting the teleconference.

8. Future meetings

Face-to-Face Meetings:

- 2013-09-09/13, Delft, NL (with other WGs and Plenary, as follows:
09 Mon: 10:00 SC 34 opening plenary; 14:30-17:00 WG1
10 Tue: 09:00-12:00 WG5; 13:30-17:00 WG4
11 Wed: 09:00-17:00 WG4
12 Thu: 09:00-17:00 WG4
13 Fri: 09:00-12:00 WG4 and WG3 in parallel; 13:00 SC 34 closing plenary)
- 2014-03-03/07, Berlin, DE (with other WGs; specific days/times TBD)
- 2014-06-16/20, Prague, CZ (with other WGs; specific days/times TBD)

Teleconferences:

- None scheduled

9. Adjournment

Adjourned by unanimous consent at 14:50 GMT.