

# **[Day 2 Draft] Minutes of the Bellevue Meeting of ISO/IEC JTC 1/SC 34/WG4, 2009-09-13/15**

**Rex Jaeschke ([rex@RexJaeschke.com](mailto:rex@RexJaeschke.com))**

**2009-09-15**

## **1. Opening remarks**

The meeting started at 10:10 on Sunday. The convener, Murata-san, welcomed everyone to Bellevue, Washington, USA, for the fourth face-to-face meeting of WG4.

## **2. Roll call of delegates**

The following members were present (\* indicates attendance by phone):

<b>Name</b>	<b>Affiliation</b>	<b>Employer/Sponsor</b>
Toshiko Kimura	SC 34 Secretariat	Japan Standards - ITSCJ
Makoto Murata	WG4 Convener	International University of Japan
Paul Cotton	CA	Microsoft
DongLin Wang	CN	Beijing Sursen International Information Technology Co. Ltd.
Liu Ning Sheng	CN	Beijing Sursen International Information Technology Co. Ltd.
Pia Elleby Lange	DK HoD	Danish Standards
Jasper Bojsen	DK	Microsoft
Mogens Kuehn Pedersen	DK	CBS
Jesper Lund Stocholm *	DK	Ciber
Rex Jaeschke	Ecma HoD, Project Editor	Consultant
Isabelle Valet-Harper	Ecma (TC45 co-chair)	Microsoft
Doug Mahugh	Ecma	Microsoft
Shawn Villaron	Ecma	Microsoft
Johann Granados Urena	Ecma	Staff DotNet

Name	Affiliation	Employer/Sponsor
Caroline Arms*	Ecma	US Library of Congress
Juha Vartiainen	FI HoD	Finnish Standards
Kimmo Bergius	FI	Microsoft
Mohamed Zergaoui	FR HoD, liaison from XML Guild and W3C	Innovimax
Klaus-Peter Eckert	DE HoD	Frauenhofer Fokus
Francis Cave	GB HoD	Francis Cave Digital Publishing
Alex Brown	GB	Griffin Brown Digital Publishing Ltd.
Gareth Horton	GB	Datawatch
P. Madhav	IN HoD	Institute of e governance
Andrea Valboni	IT HoD	Microsoft
Naoki Ishizaka	JP HoD	Microsoft
Toshiya Suzuki*	JP	Hiroshima University
Seung Yun Lee	KR	Microsoft
Shazhad Rana	NO HoD	ETRI
Keld Jørn Simonsen	NO	RAP
Grantham Daniels	ZA HoD	South African Standards - SABS
Johann Eksteen	ZA	Microsoft
Dave Welsh	US HoD	Microsoft
Andrew Rist	US	Oracle
John Peltonen	US	3Sharp
Jeffrey Chen	Invited guest (SME on MCE)	Microsoft
Zeyad Rajabi	Invited guest (SME on Custom XML)	Microsoft

14 NBs and 1 Liaison were represented.

### 3. Adoption of the agenda

The following items were added to the agenda (under “Other Business”), which was then adopted by unanimous consent:

- Request from W3C to review their Candidate Recommendation [1], "Widgets 1.0: Packaging and Configuration" (23 July 2009), in their liaison letter [2].
- Jesper’s presentation on the use of assembla.

## 4. Administration

### Approval of Previous Meeting Minutes [WG N 0084]

The minutes from the 2009-08-27 teleconference were adopted by unanimous consent as published.

### Outstanding Action Items

1. Rex will ask Ecma if it can provide public read-only access to the current mail archive. **A possible solution is still being tested**
2. Rex and Murata-san will see if the assembla system can be used to track schema changes. **Pending**
3. Shawn will get and distribute a copy of the schema containing foreign attributes, which was initially used to generate the strict and transitional schema versions. **Pending**
4. Jesper will explore the assembla system further as to its suitability for our work. **Done. He circulated a written submission, and requested time to present it at noon, local time (PDT), on Monday.**
5. Doug will see if he can get initial responses for the Open Font-related DRs for review in Seattle. **In progress**
6. Alex will produce a guidelines document regarding the relationship between transitional and strict. **Done**
7. Shawn will investigate the possibility of having Live Meeting available for members not attending the Bellevue meeting in person. **Done**

### Report from the WG4 Secretariat

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

For information about accessing the email list, and the document and email archive, please consult document WG4 N 0014 (2008).

Access to the documents on the Ecma site is restricted to registered members. For those documents that are to be made available to the public, Murata-san has provided copies of them at

<http://www.itscj.ipsj.or.jp/sc34/wg4/>.

## 5. Markup Compatibility and Extensibility

Jeffrey Chen, a subject matter expert on MCE at Microsoft, gave a presentation on that topic.

**Action:** Rex will make Jeffrey's presentation a committee document.

## 6. Custom XML and Smart Tags

Zeyad Rajabi, a subject matter expert on custom XML markup at Microsoft, gave a presentation on that topic.

**Action:** Rex will make Zeyad's presentation a committee document.

## 7. Defect reports

The latest version of the DR log was circulated as WG4 N 0083. The status of DRs at that time was as follows:

Status	Count
Open	20 (20 technical, 0 editorial)
Further Consideration Required	85 (77 technical, 8 editorial)
Last Call	0 (0 technical, 0 editorial)
Closed, to be incorporated in COR2	0 (0 technical, 0 editorial)
Closed, to be incorporated in AMD2	0 (0 technical, 0 editorial)
Closed, incorporated in COR1	175 (85 technical, 90 editorial)
Closed, incorporated in AMD1	24 (24 technical, 0 editorial)
Closed without action	9 (6 technical, 3 editorial)
<b>Total</b>	<b>313 (212 technical, 101 editorial)</b>

### Media Types

We have a number of DRs pertaining to media types. Murata-san provided an overview of the purpose of media types and the process for registering them with IANA. It was agreed that many media types in OOXML do not follow the guidelines of IETF MIME RFCs.

Some members asked how important such registration was.

There was concern about breaking existing implementations.

**Action:** Murata-san will circulate his presentation as a committee document.

Murata-san gave a summary of the open DRs relating to media types

**Action:** Murata-san will look at writing an RFC for registering a generic OPC package media type and establishing a naming convention (such as "+opczip") for specialized media types derived from OPC.

**DR-09-0034 — General: Numerous media types should be registered at IANA**

Murata-san is still working with IANA on this.

**DR-09-0160 — SML and Shared ML: Lack of Specialized Media Types**

The current design may well be inconsistent, but is not defective.

Closed without action.

**DR-09-0161 — Shared ML: Digital Signature Origin Part Media Type**

Although OOXML abuses MIME here, it was agreed that we cannot fix this "media type" without breaking existing OOXML documents. It was also agreed not to register this "media type", since it is likely to be rejected by IANA.

Closed without action.

**DR-09-0162 — Shared ML: Printer Settings Part Media Types**

Although OOXML abuses MIME here, it was agreed that we cannot fix this "media type" without breaking existing OOXML documents. It was also agreed not to register this "media type", since it is likely to be rejected by IANA.

Closed without action.

**DR-09-0163 — VML: Drawing Part Media Type**

The current design may well be inconsistent, but is not defective.

This media type still has to be registered at IANA. Murata-san will take care of this.

**DR-09-0164 — WML: Alternative Format Import Part Media Type**

It was noted that we might use the + OPC zip naming convention.

No progress.

**DR-09-0165 — SML and PML: Lack of Media Types**

It was noted that we might use the + OPC zip naming convention.

We need to register these media types.

**DR-09-0166 — SML: Custom Property Part Media Type**

Although OOXML abuses MIME here, it was agreed that we cannot fix this "media type" without breaking existing OOXML documents. It was also agreed not to register this "media type", since it is likely to be rejected by IANA.

Closed without action.

**DR-09-0167 — SML: Embedded Control Persistence Part Media Type**

We still need to do this.

Furthermore, Microsoft is requested to register this media type.

## Font-Related Issues

### DR-09-0012 — Parts, Font Part: Incomplete definition for Font Part

Shawn presented the solution as proposed in the DR log. This approach was accepted.

**Action:** Shawn and Murata-san will review the proposed text.

**Action:** Once the final wording has been accepted, Shawn will register the obfuscatedFont content type.

### DR-09-0036 — Shared, Parts, Font Part: Inappropriate media types

Although application/x-fontdata and application/x-font-ttf are inappropriate, it was agreed that we cannot fix these media types without breaking existing OOXML documents. Note that W3C continues to use application/x-www-form-urlencoded probably for the same reason. Since application/x-fontdata and application/x-font-ttf are in the special x. tree, they cannot be registered.

Shawn proposed that we close this without action. Agreed.

**Action:** Shawn will create a new DR to define the two existing media types.

### DR-09-0039 — Shared, Parts, Font Part: File format for "bitmapped font" is missing

Although another media type for Embedded OpenType Format has been already registered at IANA, we continue to use application/x-fontdata for the compatibility with existing OOXML documents.

Shawn presented a solution. Closed as proposed.

### DR-09-0040 — WML/DML: Complex scripts

Shawn presented a solution.

**Action:** Shawn will circulate his proposal to members during the meeting for review. [A1]

### DR-09-0041 — WML, Fonts: Font resource search

Shawn presented a solution.

**Action:** Shawn will circulate his proposal to members during the meeting for review.

### DR-09-0042 — WML, Fonts: notTrueType attribute missing from list

Shawn presented a solution.

**Action:** Shawn will circulate his proposal to members during the meeting for review.

**DR-09-0043 — WML, Fonts: notTrueType attribute and bitmap fonts**

Shawn presented a solution.

**Action:** Shawn will circulate his proposal to members during the meeting for review.

**DR-09-0045 — WML, Fonts: Character encodings of font names**

Shawn presented a solution.

Two issues were identified during the discussion:

1. How do we determine the first font?
2. We need to change the word “parser” to “processor”

**Action:** Shawn will circulate his proposal to members during the meeting for review.

**DR-09-0046 — WML, Fonts: Misleading example**

Shawn presented a solution. Closed as proposed

**DR-09-0047 — WML, Fonts: Identifying a face in an embedded font file**

Shawn presented a solution.

After some discussion, it was agreed that this needed more work.

**DR-09-0049 — WML, Fields: SYMBOL switch proper charset name usage**

Shawn presented a solution.

After some discussion, it was agreed that this needed more work.

**DR-09-0058 — DML, Main: Possible values of attribute script are unclear**

Shawn presented a solution. Closed as proposed.

**DR-09-0059 — DML, Main: Clarify relationship between the symbol font and Symbol character set**

Shawn presented a solution.

After some discussion, it was determined that the proposed solution was not satisfactory.

## Custom XML Markup

Zeyad Rajabi, a subject matter expert on custom XML markup at Microsoft, proposed responses for the following DRs:

**DR-09-0028 — WML, Custom XML and Smart Tags: Tighten data types**

**DR-09-0212 — WML: Custom XML and Smart Tags**

**DR-09-0219 — WML: Custom XML Attribute**

- Create a new simple type ST\_XmlName with restriction set to NCName
- We should change the name and element attributes to be of type ST\_XmlName
  - Change should be made to both transitional and strict
    - No backwards compatibility issue
  - This change affects the following elements
    - attr @name (§17.5.1.1)
    - attr @name (§17.5.1.2)
    - customXml @element (§17.5.1.3)
    - customXml @element (§17.5.1.4)
    - customXml @element (§17.5.1.5)
    - customXml @element (§17.5.1.6)
    - smartTag @element (§17.5.1.9)
- Create a new simple type ST\_Uri with restriction set to xs:anyURI
- We should change the uri attribute to have ST\_Uri type
  - Change should be made to both transitional and strict
    - Minor backwards compatibility issues
  - This change affects the following elements
    - attr @uri (§17.5.1.1)
    - attr @uri (§17.5.1.2)
    - customXml @uri (§17.5.1.3)
    - customXml @uri (§17.5.1.4)
    - customXml @uri (§17.5.1.5)
    - customXml @uri (§17.5.1.6)



- smartTag @uri (§17.5.1.9)
- schema @uri (§23.2.1)

Members liked the direction of the proposed changes.

**DR-09-0191 — Custom XML Schema: Allowed attribute values**

- Create a ST\_Uri with restriction set to xs:anyURI
- We should change the schemaLocation and manifestLocation attributes to have ST\_Uri type
  - Change should be made to both transitional and strict
    - Minor backwards compatibility issues
  - This change affects the following elements
    - schema @schemaLocation (§23.2.1)
    - schema @manifestLocation (§23.2.1)

Members liked the direction of the proposed change.

**DR-09-0207 — WML: Custom XML and Smart Tags**  
**DR-09-0208 — WML: Custom XML and Smart Tags**  
**DR-09-0209 — WML: Custom XML and Smart Tags**  
**DR-09-0210 — WML: Custom XML and Smart Tags**  
**DR-09-0211 — WML: Custom XML and Smart Tags**  
**DR-09-0213 — WML: Custom XML and Smart Tags**  
**DR-09-0216 — WML: Custom XML and Smart Tags**

## 17.5 Custom Markup

Within a WordprocessingML document, it is often necessary for specific documents to contain semantic information beyond the presentation information specified by ECMA-376. [*Example: An invoice document might wish to specify that a particular sentence of text is a customer name, in order for that information to be easily extracted from the document without the need to parse the text using regular expression matching or similar. end example*]

For these scenarios, multiple facilities are provided for the insertion and round-tripping of ~~customer defined~~[extra-standard](#) semantics within a WordprocessingML document. There are three distinct forms in which ~~customer defined~~[extra-standard](#) semantics can be inserted into a WordprocessingML document, each with their own specific intended usage:

- Smart tags
- Custom XML markup
- Structured document tags (content controls)

The elements and attributes which define each of these forms is described in the following clauses.

### 17.5.1 Custom XML and Smart Tags

The first ~~example form~~ of ~~customer-defined~~ extra-standard semantics that can be embedded in a WordprocessingML document are smart tags. Customers-Implementations can establish sets of smart tags that allow semantic labels to be added around an arbitrary run or set of runs within a document to provide information about the type of data contained within.

[*Example:* Consider the following text in a WordprocessingML document, with a smart tag around the stock symbol 'CNTS':

This is a stock symbol: CNTS

This text would translate to the following WordprocessingML markup:

```
<w:p w:rsidR="00672474" w:rsidRDefault="00672474">
<w:r>
<w:t xml:space="preserve">This is a stock symbol: </w:t>
</w:r>
<w:smartTag w:uri="http://www.example.com"
w:element="stockticker">
<w:r>
<w:t>CNTS</w:t>
</w:r>
</w:smartTag>
</w:p>
```

As shown above, the smart tag is delimited by the smartTag element, which surrounds the run (or runs) which contain the text which is part of the smart tag. *end example*]

The smart tag itself element in a document carries has two required pieces of information attributes, which together contain the customer semantics for this smart tag:

- ~~The first of these is the namespace for this smart tag (contained in the uri attribute). This allows the smart tag to specify a URI which should identifies the namespace of this smart tag to a consumer. It is intended to be used to specify a family of smart tags to which this one belongs.~~ [*Example:* In the sample above, the smart tag belongs to the `http://www.example.com` namespace. *end example*]
- The second of these is the element classification name for this smart tag (contained in the element attribute). This attribute, in combination with the namespace uri, should specify a classification which uniquely identifies this smart tag within its family and again available to a consumer ~~This allows the smart tag to specify a name which identifies this type of smart tag within its namespace and again available to a consumer. It is intended to be used to specify a unique name for this type of smart tag.~~ [*Example:* In the sample above, the smart tag specifies that its data is of classified as a style `stockticker`. *End example*]

The next example of ~~customer-defined~~ extra-standard semantics which can be embedded in a WordprocessingML document is custom XML markup. Custom XML markup allows the application of the XML elements defined in any schema syntax (XML Schema, NVDL, etc.) to be applied to the contents of a WordprocessingML document in one of two locations: around a paragraph or set of paragraphs (at the block level); or around an arbitrary run or set of runs within a document (at the inline level) to provide semantics to that content within the context and structures defined by the associated schema definition.

The distinction between custom XML markup and smart tags is that custom XML markup is based on a specified schema, which may be specified using the attached Schema element (§17.15.1.5). As a result, the custom XML elements can be validated against the schema. Also, as shown below, custom XML markup can be used at the block-level as well as on the inline (run) level.

[*Example:* Consider a simple XML Schema which defines two elements: a root element of `invoice`, and a child element of `customerName` - the first defining that this file's contents are an invoice, and the second specifying that the enclosed text as a customer's name:

This output would translate to the following WordprocessingML markup:

```
<w:customXml w:uri="http://www.example.com/2006/invoice" w:element="invoice">
<w:p>
<w:r>
<w:t>This is an invoice.</w:t>
</w:r>
</w:p>
<w:p>
<w:r>
<w:t xml:space="preserve">And this is a customer name: </w:t>
</w:r>
<w:customXml w:uri="http://www.example.com/2006/invoice"
w:element="customerName">
<w:r>
<w:t>Tristan Davis</w:t>
</w:r>
</w:customXml>
</w:p>
</w:customXml>
```

As shown above, each of the XML elements from the ~~customer~~-supplied XML schema is represented within the document output as a customXml element. *end example*]

Similar to the smart tag example above, a custom XML element in a document has two required attributes.

- The first is the uri attribute, whose contents specify the namespace of the custom XML element in the document. In the example above, the elements each belong to the `http://www.example.com/2006/invoice` namespace.
- The second is the element attribute, whose contents specify the name of the custom XML element at this location in the document. In the example above, the root element is called `invoice` and the child element is called `customerName`.

As well as the required information specified above, custom XML elements can also specify any number of attributes (as specified in the associated XML Schema) on the element. To add this information, the `customXmlPr` (properties on the custom XML element) specify one or more `attr` elements.

[*Example:* Using the example above, we can add a `type` attribute to the `customerName` element as follows:

```
<w:customXml w:uri="http://www.example.com/2006/invoice"
w:element="customerName">
<w:customXmlPr>
<w:attr w:uri="http://www.example.com/2006/invoice" w:name="type"
w:val="individual"/>
</w:customXmlPr>
<w:r>
<w:t>Tristan Davis</w:t>
</w:r>
</w:customXml>
```

The resulting XML, as seen above, simply adds an `attr` element which specifies the attribute for the custom XML element. *end example*]...

**Part 1, §17.5.2, “Structured Document Tags”, p. 549, will be updated as follows:**

The final form of ~~customer-defined~~-extra-standard semantics which can be embedded in a WordprocessingML document are structured document tags (SDTs).

As shown above, smart tags and custom XML markup each provide a facility for embedding ~~customer defined~~extra-standard semantics into the document: smart tags, via the ability to provide a basic namespace/name for a run or set of runs within a documents; and custom XML markup, via the ability to tag the document with XML elements and attributes specified by any XML Schema file.

However, each of these techniques, while they each provide a way to add the desired semantic information, does not provide a way to affect the presentation or interaction within the document. To bridge these two worlds, structured document tags allow both the specification of extra-standard~~customer~~ semantics as well as the ability to influence the presentation of that data in the document.

This means that the implementation~~customer~~ can define the semantics and context of the tag, but can then use a rich set of pre-defined properties to define its behavior and appearance within the WordprocessingML document's presentation.

**Part 1, §M.1.6, “Custom Markup”, p. 5087, will be updated as follows:**

~~Within a WordprocessingML document, it is often necessary for specific documents to contain semantic information beyond the presentation information specified by this Office Open XML specification. For example, an invoice document might wish to specify that a particular sentence of text is a customer name, in order for that information to be easily extracted from the document without the need to parse the text using regular expression matching or similar. For those cases, multiple facilities are provided for the insertion and round-tripping of customer defined semantics within a WordprocessingML document.~~

There are three distinct forms in which ~~customer defined~~extra-standard semantics can be inserted into a WordprocessingML document, each with their own specific intended usage: ...

**Part 1, §M.1.6.1, “Smart Tags”, p. 5087, will be updated as follows:**

The first form~~example~~ of ~~customer defined~~extra-standard semantics ~~which~~that can be embedded in a WordprocessingML document are smart tags. Smart tags allow semantic information to be added around an arbitrary run or set of runs within a document to provide information about the kind of data contained within.

**Part 1, §M.1.6.2, “Custom XML Markup”, p. 5089, will be updated as follows:**

The next form ~~example~~ of ~~customer defined~~extra-standard semantics which can be embedded in a WordprocessingML document is custom XML markup. ...

...

A producer can embed a custom XML element around or with block-level or run-level content in a WordprocessingML document in order to embed the structure of the ~~customer defined~~extra-standard XML Schema within the WordprocessingML content. This allows ‘tagging’ of specific regions of a document with the semantics from this schema, while ensuring that the resulting file can be validated to the WordprocessingML schemas.

A consumer can read this custom XML markup and provide additional functionality around this ~~customer defined~~-[extra-standard](#) XML markup, which might or might not be specific to that particular XML namespace. Examples of this functionality include: the ability to add/remove this XML markup via a user interface, ability to provide actions to operating in the context of this namespace, etc.

**Part 1, §M.1.6.3, “Structured Document Tags”, p. 5091, will be updated as follows:**

The final [formexample](#) of ~~customer defined~~-[extra-standard](#) semantics which can be embedded in a WordprocessingML document is the structured document tag (SDT).

**Part 1, §M.3.1.2.8, “Customer Data”, p. 5304, will be updated as follows:**

There is a set of utilities that facilitate the storage of customer XML data within the file format. Although a topic for a separate paper, essentially, this functionality comes down to the ability to store ~~customer defined~~-[extra-standard](#) XML in the file format in a way that it can be easily queried, modified and/or surfaced in the presentation. Suffice it to say, the data is stored in a separate part within the package, and hence the utility pairs the object using it with the part within the package.

#### DR-09-0209 — WML: Custom XML and Smart Tags

The first bullet on p. 529 has, "The first of these is the namespace for this smart tag (contained in the uri attribute). This allows the smart tag to specify a URI which should identify the namespace of this smart tag to a consumer."

What is the effect of omitting this attribute? Is the implied element conformant to XMLNames? Is it permitted for the attribute to be something other than a URI?

Also, correct, "which should identifies [sic]".

#### DR-09-0221 — WML: Custom XML Attribute

- We should add the following text to Part 1: §17.5.1.1, “attr (Custom XML Attribute)”, p. 531

The presence of two or more attr elements with identical name and uri attribute values, including attr elements with identical name attribute values and an empty or omitted uri attribute, shall be considered non-conformant.

Members liked the direction of the proposed change.

#### DR-09-0249 — Custom XML markup: well-formedness and validation

- It is a consumer defined operation to extract XML documents from custom XML markup. The standard allows for the uri attribute to be omitted or specified for custom XML elements and attributes. If uri is specified for custom XML elements or attributes then a consumer should add a prefix to the output XML elements or attributes. If uri is omitted for custom XML elements or attributes then a consumer should not add a prefix to output XML elements or attributes.

- The result of extracting XML from custom XML markup should always return well formed XML. In addition, custom XML markup within Open XML, if added according to the standard, should never create a non-conformant Open XML document.
- The result of extracting XML from custom XML markup can return an invalid document. For example, a user may need to create a document that contains custom XML markup in multiple sessions of an application. After each session the user saves and closes his document. There may be times where the user needs to save his document, even if the custom XML markup is not valid according to the custom XML schema. Such documents will not cause Open XML to be non-conformant as long as the custom XML markup was added according to the standard. Custom XML elements/attributes are stored using the customXML and attr elements as specified by Open XML. Custom XML elements/attributes are stored as name attributes on customXML and attr elements. In other words, having non-conformant custom XML does not cause Open XML to be non-conformant.

## 8. Conformance Testing and Methodology [A2]

xx

## 9. The Relationship between "Transitional" and "Strict"

Alex presented a revised version of his paper from Copenhagen.

Much of the discussion that followed was about the suggestion that not allowing future innovation into the Transitional class was a significant impediment to implementers using Transitional as a bridge to Strict.

**Action:** Alex will submit his paper for posting as a committee document.

There was also discussion about how Part 4 might be changed, so it can be used as a stand-alone document rather than being a set of edits to Part 1.

## 10. Environments for Maintaining Schemas [A3]

xx

## 11. Planning for Future Projects

1. Technical differences between ECMA-376:2006 and ISO/IEC 29500:2008 (see SC 34 N 1202 and SC 34 N 1234):

The ballot closed and passed. However, concern was raised that as the proposed new work item was to produce a document containing only non-normative text, perhaps it should not be an amendment. (JTC 1 Directives, §15.5.2: “An amendment is issued to publish a technical addition or change.”) A TR Type 3 seems like a better fit, and ITTF does allow such TRs to be made freely available. Separately, it was noted that the proposer of the work item declined to participate in its work or to make a submission, and there was a general lack of support for participation.

There was general agreement by the attendees that this effort should not result in an amendment.

**Action:** Isabelle will write up a resolution for the SC 34 plenary regarding making the new work item for a TR Type 3 instead of an amendment, and to request project editor nomination, NB participation, and contributions.<sup>[A4]</sup>

2. Attribute unqualification: |

<sup>[A5]</sup>

xx

3. ISO Dates:

Gareth presented a set of slides.

After some discussion, Murata-san concluded that the following issues needed addressing:

- 1) Whether we should continue to use ISO dates in Transitional
- 2) Subsetting of ISO dates
- 3) Fixing the definitions of quite a few functions in SML.

**Action:** Gareth will submit a revised version of his paper for posting as a committee document.

**Action:** Shawn, Mohamed, and Dave will help Gareth prepare to discuss this topic at the next teleconference.

4. CORs and Amendments for the rest of the defect reports:

There was consensus that WG4 forward a request to SC 34 for a project subdivision for a new amendment regarding ISO 8601 date-related issues, including subsetting of 8601 and fixing numerous SML functions.

**Action:** Francis will draft the resolution and circulate to WG4 for discussion and approval on Tuesday morning. |

<sup>[A6]</sup>

There was consensus that WG4 forward a request to SC 34 for a project subdivision for a new amendment set (for Parts 1, 2, and 4) to handle any DR resolutions that change/add functionality, just like we did for amendment set 1.

**Action:** Rex will draft the resolution and circulate to WG4 for discussion and approval on Tuesday morning.<sup>[A7]</sup>

5. Amendments vs. Revisions:

Kimura-san gave a presentation on the impact of the JTC 1 Directives revision, which is expected to take affect in June 2010. One implication is that after two change-documents (either COR or AMD) to a base standard, a new edition will be required.

We discussed the impact of the using the ISO Directives in future, especially with regard to the meaning of “edition” of a standard versus “revision”.

The problem we are looking to solve is how to get a consolidated version of the standard produced without making it a revision.

## 12. Other Business

- a. W3C requested us to review their Candidate Recommendation [1], "Widgets 1.0: Packaging and Configuration" (23 July 2009), in their liaison letter [2].

[1] <http://dev.w3.org/2006/waf/widgets/>

[2] <http://www.itscj.ipsj.or.jp/sc34/def/1267.htm>

Mohamed reported on this project.

**Action:** Murata-san will study [1] and prepare a response.

- b. Jesper’s feedback on the use of assembla

Jesper presented his paper.

The main limitation of assembla seems to be the lack of rich text support, which makes it unsuitable for storing most DR responses.

**Action:** Rex will make Jesper’s paper a committee document.

## 13. Future meetings<sup>[A8]</sup>

### Face-to-Face Meetings:

The schedule is as follows:

1. 2009-12-01/03, Paris, FR (in conjunction with WG1, WG5, and, possibly, Ad Hoc 3)
2. 2010-03-22/25, Stockholm, SE (in conjunction with the SC 34 plenary)
3. 2010-06-14/16, Helsinki, FI (in conjunction with WG5)
4. 2010-09-20/23 or 2010-09-27/30 (exact dates to be decided), ZA (in conjunction with the SC 34 plenary)
5. 2009-12-06/10 or 2009-12-13/17 (exact dates to be decided), tentative offer from CN

### Teleconferences:

[See <http://www.timeanddate.com/worldclock/meeting.html> for translation of GMT to your time zone. See WG4 document N 0021 for call-in details.]

The schedule is as follows:

1. 2009-10-01, xx:00 GMT (time to be determined)
2. 2009-10-15, xx:00 GMT (time to be determined)
3. 2009-10-29, xx:00 GMT (time to be determined)
4. 2009-11-12, xx:00 GMT (time to be determined)

## 14. Adjournment



Adjourned by unanimous consent at xx:xx on Tuesday.