In response to NZ-0011 Accessibility (which is similar to Canadian comment CA-0075... high-level commentary on the necessity of basic accessibility features) here are my suggested definitions,
-- Matthew Cruickshank <matthew@holloway.co.nz>

ACCESSIBILITY AND OOXML

1. OOXML lacks the ability to link between form fields and their labels

This is about associating form fields with labels in order to describe them, and it helps a document that makes sense to a sighted person make sense to the non-sighted. The problem is this (in the following example "[...]" is used to denote a textbox):

```
Full Name:      Phone Number:
[............] [............]
```

A sighted person may understand where to write their first and last name, but if the document is read sequentially line-by-line from left to right (serially) then they would be read "Full Name Phone Number TEXTBOX TEXTBOX". The lack of a link between the text and the textbox makes understanding document difficult. To give a real world example (taken from the IRD's website, from IR 595 document) of problematic form layout here's an example where the textbox field keeps changing...

```
[...] Mr [...] Mrs [...] Miss [...] Other
First names [........................]
Surname [............................]
[....................................]
Street Address
```

This would make (some) sense when read by a sighted person, but when read serially it is much more difficult to understand. The need to label form fields is a well known issue within the accessibility community and its supported within HTML and ODF (see ODF accessibility guidelines http://urltea.com/2a28). Labeling form fields is specifically mentioned in the New Zealand E-government Standards <http://urltea.com/2a23>, the W3C WAI <http://urltea.com/2a24>, and numerous other accessibility initiatives <http://joeclark.org/book/sashay/serialization/Chapter12.html>, <http://www.webstandards.org/learn/tutorials/accessible-forms/intermediate/>.

This problem is recognised by Ecma in their comment ECMA-0039, and supported in Canadian comment CA-0069. I'm glad to see that Ecma recognise that accessibility of this type should be standardised and included within OOXML, rather than left as an non-standardised add-on feature. Standardised semantics for this feature will provide consistancy and make it far easier on accessibility software developers.

2. OOXML lacks the ability to give tables summaries or captions

This problem is similar in principal to the lack of form labels as table summaries and captions would allow for a description of tabular data that is often difficult for disabled users

to navigate. For example, take a 10x10 table where every table cell contains the value "1" except one that contains "100000". Sighted people could immediately see the peculiarity whereas a disabled person may have to drudge through each cell (what is more likely is that they may just misunderstand the table entirely after being read a pattern that they assume continues).

HTML and ODF support table summaries and captions. Giving summaries to tables is mentioned in the New Zealand E-government Standards <http://urltea.com/2a2m> and various accessibility initiatives <http://joeclark.org/book/sashay/serialization/Chapter10.html>.

Joe Clark, a notable accessibility expert, considers that in HTML the ability to have summaries and captions not a part of advanced or intermediate accessibility but a part of "basic accessibility". Personally I agree.

So it's for reasons like this that tables should be able to have summaries and captions.

3. OOXML lacks the ability to describe multi-level table headings

While sighted users can glance up a column of data to see the appropriate header disabled users may rely on speech synthesizers to speak out the relevant column header information of a table. As OOXML cannot represent multi-level table headings disabled users would be at a disadvantage.

The benefit of representing multi-level table headings is referred to in the New Zealand E-government Standards that say "For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells.". The issues to do with multi-level tables also mentioned in the W3C WAI at <http://urltea.com/2a2p>.

This concept is not unique to HTML, and there is need for this accessibility information in an Office format too.

OOXML should be changed to be able to describe multi-level table headings.

The following is new content added on December 10th, 2007.

4. OOXML lacks the ability to set a logical navigation order

Within HTML and ODF this feature is named “tab order”, named after the keyboard key that is pressed to advance between areas of the document.

When a document is not structured for serial use (as described in '1. OOXML lacks the ability to link between form fields and their labels'), or when users should be able to advance or bypass areas of a document then it's useful to be able to encode a navigation order.

The New Zealand E-government Web Standards describe this feature as used in HTML,

> “Not all users utilise a mouse or other “pointing” device for navigational purposes and rely on “tabbing” (usually via the “TAB” key) to move the cursor. The tab order is expected to follow the structural order of the web page elements. Not doing so gives a poor user experience from disorientation within the site, and can create confusion for assistive technologies such as screen and Braille readers.” -- http://tinyurl.com/3xryca

I note that ODF 1.0 provides this accessibility feature.

The W3C WAI (accessibility initiative) require a tab order in HTML in order for their 'Priority 3' accessibility. See <http://www.w3.org/TR/WCAG10-TECHS/#tech-tab-order>

I am very glad to see that this is mostly recognised by Ecma comment ECMA-0040, however I recommend that this not only apply to form controls but also to links and objects as supported by HTML (see above W3.org link).

5. OOXML Has Unreviewed Alternatives to Accessible W3C Technology

As OOXML suggests alternatives to existing standards that have gone through accessibility reviews (such as W3C XForms, XLinks, MathML, SMIL, and SVG) it is important that these OOXML alternatives do not have lesser support for accessibility.

I recommend that Ecma should provide accessibility documentation by contrasting OOXML's DrawingML, VML, WordProcessingML Forms, and PresentationML and with the accessibility features found in W3C XForms, XLinks, MathML, SMIL, and SVG. Where a W3C accessibility feature is not applicable to OOXML it should be explicitly stated. This would be a more specific approach to Canadian comment CA-0075 which requests an *Informative Annex* on accessibility features.

Further, I recommend that where possible W3C accessible formats should be allowed when they are currently excluded (for example, in OOXMLs definition of clipboard types). Canadian comment CA-0074 goes into some detail about extending clipboard types.