

New Zealand Open Source Society

The BRM delegation asks,

We are writing to ask you for input on whether the ECMA dispositions should be applied to the spec. The sole criterion is: would accepting each response improve the quality of the spec? To stress the point: this is not about whether you think DIS29500 should be adopted by ISO, it's about using whether applying each ECMA response will make the spec better.



Our responses to this question can be found in the table below.








In some cases Ecma responses are insufficient in that they have only changed descriptions or fixed typos (Eg. NZ-0012) or added only a subset of requested features (Eg. NZ-0011) and therefore we have noted that this is “An improvement” over earlier versions. In some cases the response doesn't answer the comment (NZ-0004, NZ-0053) but the response is an improvement and we have responded with “An improvement”.
































Alternatively some Ecma responses have genuine improvements external to the specification (Eg. NZ-0013) with no internal specification changes and therefore we have noted that the Ecma response makes “No improvement” to the specification itself. In other areas the solution proposed is worse (Eg. NZ-0050 re: “BSTR”) and we have responded “No improvement”.






























In all cases the answers here do not reflect our opinion on the overall quality of each response.

These responses have been given under a very stressed timeframe. Any action to a response (particularly a approval) is therefore given only in the clearest of cases.

Key	Description
* Asterisk	Further detail is available below in <i>Clarifications and Explanations</i>
 No improvement	Response is not an improvement, should be disapproved
 An improvement	Response is an improvement, should be approved

Standards New Zealand Comment	Ecma Response
NZ-0001 Publish instead as a Type 2 Technical Report	 No improvement *
NZ-0002 Improve clarity, remove ambiguity	
● Compatibility Settings lineWrapLikeWord6, autoSpaceLikeWord95 etc.	 An improvement
● Security Descriptor	 No improvement *
● Passwords	 An improvement *
● Numbering Formats	 No improvement *
● Calendar Formats	 An improvement
● Page Borders	 An improvement

NZ-0003	Seek to harmonize with ODF		No improvement
NZ-0004	Clarification around non-required elements		An improvement
NZ-0005	Remove proprietary technology		No improvement *
	<ul style="list-style-type: none"> Compatibility Settings lineWrapLikeWord6, autoSpaceLikeWord95 etc. 		An improvement
NZ-0006	Remove proprietary technology		An improvement *
NZ-0007	Improve multiculturalism		An improvement
NZ-0008	XML naming conventions in OOXML		No improvement
NZ-0009	Fix propagation of historical bugs		An improvement *
NZ-0010	Clarify Intellectual Property: copyright, patents		An improvement *
NZ-0011	Improve Accessibility		An improvement *
NZ-0012	Remove proprietary technology		
	<ul style="list-style-type: none"> SSPI 		An improvement
	<ul style="list-style-type: none"> OLE 		No improvement *
NZ-0013	Legacy binary data formats		No improvement *
NZ-0014	Improve DEVMODE printer settings		No improvement *
NZ-0015	Feature parity: blinking text		No improvement *
NZ-0016	Feature parity: table cell protection		No improvement
NZ-0017	Feature parity: number of lines		No improvement
NZ-0018	Feature parity: table alignment		No improvement
NZ-0019	Feature parity: last line paragraph alignment		No improvement
NZ-0020	Feature parity: more than 63 columns maximum		No improvement
NZ-0021	Feature parity: leading line spacing		No improvement
NZ-0022	Feature parity: tabs fill character		No improvement
NZ-0023	Feature parity: 'Title' 'lowercase' style		No improvement
NZ-0024	Feature parity: table 'keep with next paragraph'		No improvement
NZ-0025	Feature parity: 'may break between rows'		No improvement
NZ-0026	Feature parity: hidden sections		No improvement
NZ-0027	Feature parity: table background images		No improvement
NZ-0028	Feature parity: evenly distribute table columns		No improvement
NZ-0029	Feature parity: rotate text by 90 or 270 degrees		No improvement
NZ-0030	Feature parity: multi-row repeated heading		An improvement *
NZ-0031	Feature parity: copy heading split table		No improvement

NZ-0032	Feature parity: table shadow		No improvement
NZ-0033	Feature parity: vertical list numbering		No improvement
NZ-0034	Feature parity: absolute frame image positioning		No improvement
NZ-0035	Feature parity: text background colour		No improvement
NZ-0036	Feature parity: before/after footnote references		No improvement
NZ-0037	Feature parity: keep ratio for frames		No improvement
NZ-0038	Feature parity: columns in frames/textboxes		No improvement
NZ-0039	Feature parity: page colours		No improvement
NZ-0040	Feature parity: notes in text boxes		No improvement
NZ-0041	Feature parity: each image border properties		No improvement
NZ-0042	Feature parity: background opacity		No improvement
NZ-0043	Feature parity: auto page break		No improvement
NZ-0044	Feature parity: granular font weights (boldness)		No improvement
NZ-0045	Feature parity: block manual changes to TOC		No improvement
NZ-0046	Feature parity: shadow distance and colour		No improvement
NZ-0047	Feature parity: column separator attributes		No improvement
NZ-0048	Feature parity: textbox vertical alignment		No improvement
NZ-0049	Browser export profiles		No improvement *
NZ-0050	Remove Proprietary binary code		No improvement *
	● bitmasks		An improvement *
	● <i>(other areas of response)</i>		An improvement
NZ-0051	Provide mapping from legacy binary to OOXML		No improvement
NZ-0052	Improve Readability, Ease of Use		No improvement *
	● bitmasks		An improvement *
	● <i>(other areas of response)</i>		An improvement
NZ-0053	Cross-platform reverse engineering		No improvement *
	● bitmasks		An improvement *
	● <i>(other areas of response)</i>		An improvement
NZ-0054	Clarify Intellectual Property: copyright, patents		No improvement

Clarifications and Explanations

NZ-0001 - Publish instead as a Type 2 Technical Report

No changes to the text of the standard are proposed here which is why we have said that there is “No improvement”. However, we do not agree with the opinion of the Ecma response in which they state,

According to the ISO/IEC JTC 1 Directives, §16.2.2, “Type 2 Technical Report [...] When the subject in question is still under technical development or where for any other reason there is the possibility of an agreement at some time in the future, JTC 1 may decide that the publication of a TR would be more appropriate.”

We disagree that the specification for Office Open XML is still under technical development.

The NZOSS disagree that the standard is not under technical development. DIS-29500 is under significant technical development. These significant developments delayed the release of a final specification until the last minute before the BRM.

Although externally visible factors suggest significant development we note that the pace of internal development has been talked about on Microsoft blogs. To quote Microsoft's Brian Jones,

It's unbelievable how much work we've been able to accomplish within TC45. Similar to how we moved from a 2,000 page spec to a 6,000 in 2006, in 2007 we were able to respond to 3,500 comments and generated a 2,300 page document (a bit less that a page per comment) [...]

It's been suggested that the responses might add several thousands pages to the length of DIS-29500. If the specification is not under technical development then why can't we view a final version before the March vote?

NZ-0002 – Improve clarity, remove ambiguity

Security Descriptor

This is an inadequate solution. We feel that the suggested examples will only obscure the fact that this descriptor lacks any defined semantics for standards-based interoperability.

Therefore we do not view this as an improvement.

Passwords

This is “An improvement” based on the assumption that password documentation problems as raised in Spex under “NZ 006 (ECMA response)” and acknowledged by Brett Roberts QUOTE “*Excel password hashing error has been acknowledged*” have been fixed.

We have not have time to verify this for ourselves however due to the shortness of the review time however in the good faith of this process we will trust Microsoft.

Numbering Formats

Ecma's response refers to other responses that have some inadequate solutions.

Ecma Reference BR-0030 has `lowerLetter` (Lowercase Latin Alphabet) where formatting "is determined by using the language of the `lang` element" but what if the `lang` element contains a non-Latin language or a language without lowercase letters (Eg. Chinese). We suggest that the original description although ambiguous was more clear than this and therefore we do not believe this is an improvement.

Ecma Reference BR-0030 has `decimalEnclosedCircle` and `decimalEnclosedCircleChinese` which both appear to have identical definitions. Distinct usage remains unclear. This is not adequate.

Ecma Reference BR-0030 has `ordinal` and `ordinalText` which have identical descriptions and yet different examples. Are these identical? This is not adequate.

Therefore we view this as “No improvement”.

NZ-0005 – Remove proprietary technology

Aside from classifying `lineWrapLikeWord6`, `autoSpaceLikeWord95` etc. as proprietary this response does not address any particular issues of proprietary technology.

Therefore we view this as “No improvement”.

NZ-0005 – Remove proprietary technology

This response removes example text of Rich Text Format (RTF), a proprietary Microsoft format.

It is a welcome but unfortunately minor change. It is “An improvement”.

NZ-0009 – Fix propagation of historical bugs

Perhaps this is just a terminology difference but classifying features as deprecated when these features are required the stated purpose of DIS-29500 seems strange. We do not feel that this is a point worth arguing however, and we accept the change (it's perhaps not an improvement per se but we don't mind the change).

This response does not refer to *any* fixes of spreadsheet bugs as referred to in NZ-0009. There has been “No improvement” in this area.

Dates

These responses suggest 5 date formats some of which solve problems reported by Standards New Zealand. The documentation surround this is incorrect in saying “In the 1900 date-base system, the lower limit is January 1, 0000, which has a serial value -693959”. There was no year zero (“0000”) although there was a 1 BC (as defined by ISO 8601).

If we have to accept these dates as *all or nothing* then we would deem it as “An improvement” however the erroneous year 0 is “No improvement”.

NZ-0010 – Clarify Intellectual Property: copyright, patents

We note that the response from Ecma declares that it will make “no changes” based on this comment however the clipart mentioned in NZ-0010 is now included in DIS-29500 and therefore gains

copyright infringement protection. This is “An improvement”, and we welcome the gained Intellectual Property rights.

As we understand it, it is acceptable for NBs to ask for licensing considerations. This comment asked for licensing compatible with the normal concepts of open source development. No changes for the primary open source licence the GPL have been made. Infact the Microsoft OSP has been crafted in such a way to exclude Open Source competitors through its lack of compatibility with the GPL.

QUOTE:

"Instead of offering a patent license for its protocol information on the basis of licensing arrangements it knows are incompatible with the GPL

<http://www.press.redhat.com/2008/02/21/red-hat-statement-on-microsoft-announcement/>

From the Microsoft FAQ on the OSP about the GPL (yay acronyms!), QUOTE:

"Q: Is this Promise consistent with open source licensing, namely the GPL? And can anyone implement the specification(s) without any concerns about Microsoft patents?

A: The Open Specification Promise is a simple and clear way to assure that the broadest audience of developers and customers working with commercial or open source software can implement the covered specification(s). We leave it to those implementing these technologies to understand the legal environments in which they operate. This includes people operating in a GPL environment. Because the General Public License (GPL) is not universally interpreted the same way by everyone, we can't give anyone a legal opinion about how our language relates to the GPL or other OSS licenses, but based on feedback from the open source community we believe that a broad audience of developers can implement the specification(s)."

<http://www.groklaw.net/article.php?story=20080211032647670>

The primary open source licence has been excluded from licensing.

NZ-0011 – Improve Accessibility

These changes are “An improvement” as we appreciate the addition of accessibility features.

We do note however that further Standards New Zealand clarifications asked for an accessibility review comparing OOXML to the W3C Web Content Accessibility Guidelines (WCAG). We have several concerns with this draft report authored by Microsoft as noted on Spex. The draft report incorrectly declares OOXML as being compliant with WCAG, the facts of which we can demonstrate to Standards NZ at a later date.

Also several accessibility features were not added such as Table Summaries and Multi-level Table Headings.

The addition of Image Maps to OOXML would also be appreciated.

NZ-0012 – Remove proprietary technology

OLE

This is an inadequate solution. We feel that OLE has not been removed as asked and that the suggested examples will only obscure the fact that this lacks any defined semantics for standards-based interoperability. The given examples of cross-platform interoperability such as KParts are acknowledged as “erroneous” by Brett Roberts on Spex (see thread “KParts / Bonobo comment from 30Jan08”).

Therefore we view the addition of unworkable examples as being “No improvement”.

The minimal extent of the change that Ecma has proposed may be best expressed by a screenshot:

Child Elements	
oleItems (OLE-Object Link Items)	
Attributes	
id (OLE-Object Link Relationship)	Relationship ID that references a attribute in the associated relation for this oleLink.
Namespace: .../officeDocument/2006/relationships	The possible values for this attribute simple type (§7.8.2.1).
progId (OLE-Object Link ProgIDIdentifier)	The progID for the OLE-object link would store the name of the app would store the ProgID of the app

NZ-0013 – Legacy binary data formats

We would like to note that we appreciate the addition of legacy binary formats, albeit outside the DIS-29500 standard. We do mean to acknowledge that this is an improvement however it is “No improvement” as it doesn't appear to affect DIS-29500.

We do note however that necessary parts of these binary formats remains undisclosed and outside of OSP legal coverage, as shown in Spex comments (see thread “Office Binary Formats documentation to be released under OSP”).

NZ-0014 – Improve DEVMODE printer settings

This is an inadequate solution. We feel that the suggested examples will only obscure the fact that DEVMODE lacks any defined semantics for standards-based interoperability.

Therefore we view this as “No improvement”.

NZ-0015–NZ-0029 – Feature Parity

No move towards harmonization has been made by Ecma.

As shown in the 'Harmonization or Extensions' presentation at SNZ there is a great deal of harm and waste caused by this approach by Ecma.

This is “No improvement”.

NZ-0030 – Feature parity: multi-row repeated heading

We appreciate this clarification. This is “An improvement”.

NZ-0031–NZ-0048 – Feature Parity

No move towards harmonization has been made by Ecma.

As shown in the 'Harmonization or Extensions' presentation at SNZ there is a great deal of harm and waste caused by this approach by Ecma.

This is “No improvement”.

NZ-0049 – Browser export profiles

This is an inadequate solution. We feel that the suggested examples will only obscure the fact the `optimizeForBrowser` lacks any defined semantics for standards-based interoperability.

Therefore we view this as “No improvement”.

NZ-0050 – Remove Proprietary binary code

Generally speaking these responses are “An improvement” except where noted below.

“bstr”

This describes a syntax for subverting XML strings in order to allow non-XML string characters. This would create security issues on name coding.

This is a widely known issue and it's surprising that Ecma have not solved this.

Even Rick Jelliffe has commented on this,

“one area where I think OOXML goes seriously wrong: in a few places it provides a mechanism for circumventing XML’s character repertoire restrictions. I think the idea that just because someone generated an automatic name and used the backspace character as part of it, this should be regarded as acceptable practice in the standard is completely bogus. Several National Bodies have commented on it: I hope ECMA will have the good sense to remove it or severely deprecate it at the least. For example it is clearly a security hole to allow backspace in names, where the visible name may be coded differently than its readers expect: a kind of spoofing.”

In this case we find ourselves entirely in agreement with Rick.

Ecma should have used an `xsd:hexBinary` or `xsd:base64Binary` rather than a string. Some commentary suggests elements like `<backspace>` to provide the functionality. Any of these would be workable solutions.

This is why we believe the proposed Ecma response is “No improvement” to DIS-29500.

DEVMODE

See our clarifications and explanations under “NZ-0014 Improve DEVMODE printer settings”

BITMASKS

We agree that several cases of bitmasks should remain as bitmasks.

However - in the majority of cases - bitmasks are a unnecessary technique that cause complexity in parsing, extensibility problems, and problems in maintainability and human-readability. As we understand it Ecma have not fixed these particular areas of bitmask use in OOXML (unfortunately we don't have time to verify this against the Feb 22nd document, apologies)

- Paragraph conditional formatting
- Table cell conditional formatting
- Table row conditional formatting
- Table style conditional formatting settings
- Table style conditional formatting settings exceptions
- Suggested filtering for list of document styles
- Suggested sorting for list of document styles
- tableproperties attribute of shape group.

The fix to this one was as simple as mapping a multi-choice list to readable terms (XML attributes) and we find it strange that Ecma have not budged on this issue.

Ecma have not suggested an appropriate fix to the question, however they have provided a workaround in the shape of some complex multipage XSLT.

The addition of bloated and complex XSLT to work around an unnecessary technical problem is “An improvement” over the original text, but it is also depressing.

(other areas of response)

The other areas mentioned in the response are “An improvement”.

NZ-0051 – Provide mapping from legacy binary to OOXML

This response suggests no changes and therefore there is “No improvement”.

Providing such a mapping is the only way to validate OOXML against its stated goal of being backwards compatible with binary Microsoft files. Without this mapping its claims appear more as marketing material than as engineering criteria.

Additionally we note that Ecma's response to this comment appears to contradict their response to NZ-0003. In their response to NZ-0051 they say,

it is not possible to create a simple one-to-one mapping between specific Office Open XML functionalities and specific versions of Microsoft Office applications because the features have evolved over time with each version of each application.

However in Ecma's response to NZ-0003 there reads,

Ecma believes that one format cannot simultaneously meet the requirements that would come from the merge of the two formats and the stringent requirements of backward compatibility that drive the design of OpenXML.

This latter response is the argument against any moves to feature compatibility or harmonisation.

NZ-0052 – Improve Readability, Ease of Use

Ecma's response does not – in our opinion – respond to this comment accurately and we note that it is identical to the response to “*NZ-0050 – Remove Proprietary binary code*”. The NZ comment was distinct and deserved an answer.

Accordingly please see that comment for further clarifications and explanations.

NZ-0053 – Cross-platform reverse engineering

Ecma's response does not – in our opinion – respond to this comment accurately and we note that it is identical to the response to “*NZ-0050 – Remove Proprietary binary code*”. The NZ comment was distinct and deserved an answer.

Accordingly please see that comment for further clarifications and explanations.

NZ-0054 - Clarify Intellectual Property: copyright, patents

Please see our response to “*NZ-0010 – Clarify Intellectual Property: copyright, patents*”