DR 16-0014 — WML: RSID Issues

Status: Further Consideration Required

Subject: WML: RSID Issues

Qualifier: Request for clarification

Submitter: Courtenay Inchbald Organization: Ecma

Contact Information: c@inchbald.com

Submitter’s Defect Number: None

Supporting Document(s): None

Date Circulated by Secretariat: 2016-04-18

Deadline for Response from Editor: 2016-06-18

IS 29500 Reference(s): 29500:2016, Part 1, §17.15.1.70–§17.15.1.72, p. 1,060–1,063

Related DR(s): None

Nature of the Defect:

**Issue #1**

§17.15.1.70 and §17.15.1.71 both correctly define an RSID as an ST\_LongHexNumber (defined in §17.18.50), but incorrectly describe this as a four-digit hex number instead of an eight-digit hex number. The examples shown are, correctly, eight-digit hex numbers (although the implementation I use [MS-Word] uses only 6 of the 8 digits).

**Issue #2**

In §17.15.1.72, RSIDroot is defined as “the revision save ID which was associated with the first editing session for this document”.

In §17.15.1.72, RSIDS is defined as “the set of revision save ID values for the current document”.

In §17.15.1.70, RSID is defined as “the revision save ID that was associated with a single editing session for a document”.

The implication of these in combination is that a document's RSIDroot is its first revision save ID and that all the others listed in its RSIDS element postdate it. In fact, in my implementation, the RSIDS element inherits all the revision save IDs from the template from which the document was made.

The note added to the definition of RSID amplifies the implication saying, “the meaning of two revision save IDs is not defined for documents with a different RSIDroot”. I presume it should say "the meaning of a revision save ID shared by two documents with a different RSIDroot is not defined". If so, although it does not say so explicitly, it implies that a shared revision save ID is not meaningful unless the rsidRroot is also shared. In fact, when the rsidRoot is different, although a single shared revision save ID might be coincidental, several shared revision save IDs almost certainly originate from a common template.

I would very much like to know whether ECMA-376/IS 29500 is a bit misleading, or if my implementation is non-compliant in its use of templates (in which case it should be mentioned in its implementer notes).

Solution Proposed by the Submitter:

None

Schema Change(s) Needed:

No

**Editor’s Response:**

**2016-10-14 Rex Jaeschke:**

From MS experts: If the RSIDRoot is different then you know NOTHING about the relationship between the two documents.  Even if all the other RSID’s are the same you still know nothing.  It might be that Word copies the RSID list from the template into documents based on the template or it might not.  But it doesn’t matter.  By changing the RSIDRoot, Word is telling you that it would not be fruitful to run a merge between these two documents assuming a common base.

Does this response clarify the question?

**2016-10-26 Courtenay Inchbald:**

The product team's reply is true for the purposes of running a Combine or a Compare (although Word does not object to combining or comparing documents with different rsidRoots). However, I am using RSIDs for a different purpose in which RSIDs inherited from templates are important. If the note's scope is limited to Merge and Compare, it would be helpful to me to say so in ECMA-376.

Word does not follow the rule in the rsid definition, repeated in the note at the end of the definition, that RSIDs are generated from the date. It would be helpful to me to add this exception to MS-OE376.

**2016-11-16 Aarti Nankani:**

It doesn’t seem like updating the documentation to specifically say “rsids are about merges and compares” is the right thing to do- the standard says “this information can be used as desired”. Another app might implement their own merge/compare algorithm, and it might not use rsids at all. It’s just what Word uses them for. Is it right to document that they won’t work for scenarios a, b, or c, as they come up?

The other issue mentioned is already done- we already document that we don’t follow the rules of rsid generation in MS-OI29500, and in section 2.1.424 in MS-OE376.

**2016-11-17 Courtenay Inchbald:**

The problems arise because MS-OI29500's description of RSID generation contain errors. As explained in MS-OE376, Microsoft deals with these errors by generating RSIDs at random, although for an unspecified reason, it uses only 23 of the 32 available bits. As far as I know, every other application using the standard, e.g. Google Docs, deals with the error by replacing all RSIDs with "00000000" or otherwise ignoring them.

The note creates confusion by giving guidance on the basis of assumptions that are not true for either Word documents (because the RSID generation rules quoted in the note are wrong, because Word documents inherit RSIDs from templates creating multiple RSID matches - but not rsidRoot matches - between unrelated documents, and because Word's Merge function allows unrelated documents to be merged giving the merged document the rsidRoot of one of the merged documents creating rsidRoot matches between unrelated documents), or those produced by any other application (because they replace RSIDs with "00000000" or otherwise ignore them).

I think you should correct the errors in the RSID generation rules, and, if you do not want to qualify the confusing note, I think you should remove it. I realise, however, that this is a very complicated thing to do because there are so many interrelated documents.

**2016-11-?? Aarti Nankani:**

From MS experts:

For “RSID matches – but not rsidRoot matches – between unrelated documents” that’s fine.  It’s even called out ISO-29500.  RSIDs are not globally unique.  The fact that two documents share some RSIDs but do not share the same rsidRoot means that they are different documents and the overlapping RSIDs are coincidence.  Choosing random numbers will yield duplicates eventually and we make no attempt not to BECAUSE the rsidRoot is different.

Word can merge two documents that do not have the same rsidRoot but the merge will not be as accurate as it would be if the rsidRoots were the same (see [Appendix A](#AppendixA)).  The fact that the resulting document shares the same rsidRoot as one of the two documents in the merge makes sense – you started with a document and did a series of edits (by doing a merge).  It’s still the same document.

For the green stuff, I do not think we should make a code change in Word.  So, if that’s the correction the finder wants, the answer is (politely) no.  I don’t know what description of RSID generation contains errors – we document [here](https://msdn.microsoft.com/en-us/library/ff533216%28v%3Doffice.12%29.aspx) and [here](https://msdn.microsoft.com/en-us/library/ff530756%28v%3Doffice.12%29.aspx) that the RSIDs are random.

**Appendix A**

Consider two documents:

|  |  |
| --- | --- |
| Document 1 | Document 2 |
| ABCDE | ACD |

Suppose the user asked Word to combine them.  In the absence of any other information, there are many interpretations of what happened here.  In one interpretation, Document 2 is the original document and Document 1 has two paragraphs added to it.  The correct merge there would be: A B C D E.  In another interpretation, Document 1 is the original document and Document 2 had two paragraphs deleted.  The correct merge there would be: A C D.

Now let’s add RSIDs to the mix

|  |  |
| --- | --- |
| Document 1 (rsidRoot = 12) | Document 2 (rsidRoot = 12) |
| A (rsid = 12)B (rsid = 12)C (rsid = 12)D (rsid = 12)E (rsid = 50) | A (rsid = 12)C (rsid = 12)D (rsid = 12) |

Now there is only one interpretation.  Neither document is the original.  The user who edited Document 1 added paragraph E.  The user who edited Document 2 removed paragraph B.  The correct combination of these two documents is:

A (rsid = 12)

C (rsid = 12)

D (rsid = 12)

E (rsid = 50)

But there is no way to arrive at that conclusion without the help of RSIDs, and that’s what Word uses them for.

**2016-12-07 Teleconference:**

[It was pointed out that a detailed email response from MS experts was missing from this log. Rex has added it after the teleconference ended; see the entry directly above this.]

Murata will look at this and send his feedback to Aarti, and Rex will send the latest log to the submitter.

**2016-12-08 Murata-san:**

I would like to propose three changes.

1) In §17.15.1.70 and §17.15.1.71, replace "four-digit" by "eight-digit".

Rationale: An obvious mistake.

2)  Section 17.15.1.70, rsid (Single Session Revision Save ID)

In the first bullet, remove "that is larger than all earlier ones in the same file".

Rationale: MS Office does not enforce this requirement, and thus existing OOXML documents are unlikely to satisfy it.

MS-OI29500:

2.1.414 Part 1 Section 17.15.1.70, rsid (Single Session Revision Save ID)

a. The standard states that the rsid should be randomly chosen and that every editing session shall be assigned an rsid that is larger than all earlier ones in the same file. If both are observed, the application would quickly run out of allowable rsid's.

Word generates a random number for every rsid that is not necessarily larger than all earlier ones in the same file.

3) Replace

"However, the meaning of two revision save IDs is not defined for documents with a different rsidRoot. Applications can use this information as desired."

in 17.15.1.70

by

"However, if two documents specify rsidRoot (§17.15.1.71) differently, an identical rsid value between the two documents does not indicate the same editing session."

Rationale: Clearer.

**2017-01-16 Caroline Arms:**

I've been trying to get my head round this DR with some difficulty.

In the most recent message Murata-san is suggesting some changes, but doesn't indicate whether he considers them to be a complete solution.

SUGGESTION 1

On the first suggestion, I am not certain that the 4 vs 8 "digit" issue needs addressing in exactly this way.

The confusion here is that each Hex "digit" is two characters long as encoded in the val attribute. A link to 17.18.50 ST\_LongHexNumber (Eight Digit Hexadecimal Value) might be useful. There the phrase "four octet (eight digit) hexadecimal number" is used, which might be clearer.

Note that the val attribute table entry attempts to explain this in

>>>

Specifies a number value specified as a four digit hexadecimal number), whose contents of this decimal number are interpreted based on the context of the parent XML element.

[Example: Consider the following value for an attribute of simple type

ST\_LongHexNumber: 00BE2C6C.

This value is permitted, as it contains four hexadecimal digits, each an encoding of an octet of the actual decimal number value. It can therefore be interpreted as desired in the context of the parent XML element, end example]

>>>

I believe we need a change to the normative sentence here, whose meaning seems unclear and has a spurious ) character. Same change (whatever we decide on) will be needed in 17.15.1.71.

SUGGESTION 2

The 2nd suggestion seems problematic without a requirement for uniqueness within the file, implicit in the clause Murata-san proposes to drop. I don't have a problem with dropping the "larger' requirement.

SUGGESTION 3

The third suggestion, although possibly clearer, could also possibly be improved. Rather than

"However, if two documents specify rsidRoot ( 17.15.1.71) differently, an identical rsid value between the two documents does not indicate the same editing session."

I would say:

"However, if two documents have different values for rsidRoot ( 17.15.1.71), an identical rsid value between the two documents does not indicate the same editing session."

MORE THOUGHTS ON THE DR

I see the following in a response "from the experts"

"For the green stuff, I do not think we should make a code change in Word. So, if that s the correction the finder wants, the answer is (politely) no. I don t know what description of RSID generation contains errors we document here and here that the RSIDs are random."

I see nothing in green and therefore don't know what change is being declined, but it may be related to my musings below:

I was struck by one issue that I don't see as adequately addressed in terms of explanation -- although it may not require changes

The 10/26 message in the DR from submitter Courtenay Inchbald says:

"Word does not follow the rule in the rsid definition, repeated in the note at the end of the definition, that RSIDs are generated from the date. It would be helpful to me to add this exception to MS-OE376. "

The 11/16 response from Aarti says:

"The other issue mentioned is already done- we already document that we don t follow the rules of rsid generation in MS-OI29500, and in section 2.1.424 in MS-OE376. " Those are the links at "here and here" above.

However, what is said in those two locations says nothing about whether RSIDs are generated from the date (or in fact time which is what the standard says). The wording is "Word generates a random number for every rsid that is not necessarily larger than all earlier ones in the same file."

17.15.1.70 rsid

says normatively, "Revision save IDs should be randomly generated based on the current time (to minimize the chance that two disparate editing sessions starting with the same immediate predecessor are assigned the same revision save ID)" and informatively in a note, "Although in practice it is possible for two independent sessions to result in the same value, this outcome is extremely rare as the values are based on the current time."

Question from me: Is Courtenay correct in saying that RSIDs are not based on the date/time? If so she has a point. But she may be assuming that is not the case because she observes that RSIDs are not larger than previously generated ones. It is certainly possible to use a randomization mechanism that includes incorporating a date/time stamp in order to minimize the chance of generating the same one but that doesn't enforce any ordering. Can I assume that is what Word does? If so, we probably don't HAVE TO make any changes.

However, we could possibly change the second bullet in 17.15.1.70 to something like:

"Revision save IDs should be randomly generated in a way that minimizes the chance that two disparate editing sessions starting with the same immediate predecessor document are assigned the same revision save ID (perhaps by using the current time)."

and perhaps also make a small modification to the note that Murata-san has already suggested a change to.

My apologies for all the words here. I needed them for my own clarification.

**2017-01-25 Teleconference:**

Murata-san agrees with all of Caroline’s three points.

Caroline thinks maybe we need to say more w.r.t Part 1, §17.15.1.70, “rsid (Single Session Revision Save ID)”.

She would like an answer to her question in the mail (whether Office does or does not use timestamps in the generation of the random rsid values - as normatively required in 17.15.1.70). Please clarify the RSID generation rules.

**2017-02-08 Courtenay Inchbald:**

I suggest the following changes, which incorporate what you have all said.

**17.15.1.70 rsid (Single Session Revision Save ID)**

This element specifies the revision save ID that was associated with a single editing session for a document. An editing session is a span of time that begins and ends with any event that produces an editable file, such as a save or an e-mail send, and contains no such event. When revision save IDs are added to a document, they shall follow these rules:

~~•             Every editing session shall be assigned a revision save ID that is larger than all earlier ones in the same file~~

•             Revision save IDs should be randomly generated ~~based on the current time (to minimize the chance that two disparate editing sessions starting with the same immediate predecessor are assigned the same revision save ID)~~

(I have deleted the reference to time because I don’t think it is necessary to specify the method of random number generation. I have deleted the note because it seems to be relate to the use of the time as part of random number generation)

•             Revision save IDs should be unique within a document.

(As suggested by Caroline Arms. Applications like Google Docs which always uses “00000000” will be non-compliant, but they are already non-compliant in not using rsids.)

•             Changes to document content in an editing session shall be stamped with the current revision save ID using the appropriate rsid\* attributes

•             An identical rsid value between two documents with the same rsidRoot (§17.15.1.71) shall indicate the same editing session.

[Note: A revision save ID should be treated as unique within the context of all documents with the same rsidRoot value. Although in practice it is possible for two independent sessions to result in the same value, this outcome is extremely rare ~~as the values are based on the current time~~. ~~However,~~ The meaning of two revision save IDs is not defined for documents with a different rsidRoot although applications might operate such that a new document (i.e. with a new rsidRoot) created by duplicating an old document or template will inherit the old document’s rsids element. Applications can use this information as desired. end note]

(The change to the note addresses my Issue #2, although I perceive it as rather clumsy. I prefer it to the other suggestions which seem to cause more problems than the original. Mt proposal acknowledges that rsids identifying the same editing sessions might be shared by documents with different rsidroots but says that this relationship is not defined as useful information in accordance with Aarti Nankani’s contributions. I am not suggesting any programming changes. I understand how rsids could be used as he explained and assumed that to be their purpose. However, my experience of the Compare and Combine functions caused me to conclude that they were not using rsids because the result often (possibly always – I cannot remember) depends on which document is selected first.)

**17.15.1.72 rsids (Listing of All Revision Save ID Values)**

This element specifies the set of revision save ID values for the current document. Revision save ID values refer to ~~four digit hexadecimal values~~ ST\_LongHexNumber numbers which uniquely identify ~~an editing session in the life of~~ the editing sessions associated with the current document.

(As suggested by Murata-San and Caroline Arms. I further suggest that the definition of an rsid as an ST\_LongHexNumber belongs in 17.15.1.70 rsid, not in 17.15.1.72 rsids. Later in the sentence I have made "session" plural to correspond with the rest of the sentence and I have replaced "in the life of" to deal with my issue #2, i.e. that if a document's life starts with its rsidRoot, as implied elsewhere, some of its rsids predate its life.)

An editing session is the period of time between two subsequent save operations by an application.

(This also belongs in 17.15.1.70, and is already although with different wording.)

[Guidance: The set of revision save IDs stored with a document only supplies information about ~~the~~ editing ~~session in~~ sessions which ~~document components~~ were ~~last~~ saved, which can be used by applications in any manner desired. end guidance]

(There seem to be errors here, to which I have proposed corrections.)

**2017-03-02 Seattle Meeting:**

Caroline would still like to get a response to her question, “Does Office use timestamps in the generation of the random rsid values - as normatively required in §17.15.1.70? Please clarify the RSID generation rules.”

We agreed to ask the following as well: If another implementation generates RSIDs that do not increment the suffix, as allowed by the current standard, can Word make sense of those? Basically, are RSIDs opaque strings, or are implementations required to interpret them as being in a particular sequence?

We discussed the submitter’s proposed edits from 2017-02-08.

**17.15.1.70 rsid (Single Session Revision Save ID)**

This element specifies the revision save ID that was associated with a single editing session for a document. An editing session is a span of time that begins and ends with any event that produces an editable file, such as a save or an e-mail send, and contains no such event. When revision save IDs are added to a document, they shall follow these rules:

~~•             Every editing session shall be assigned a revision save ID that is larger than all earlier ones in the same file~~

•             Revision save IDs should be randomly generated ~~based on the current time (to minimize the chance that two disparate editing sessions starting with the same immediate predecessor are assigned the same revision save ID)~~

We agreed to strike completely the 2nd bullet above. Saying how the IDs are generated is an implementation detail, and need not be stated. (See edit below.)

•             Revision save IDs should be unique within a document.

We agreed to replace the above bullet with the following:

•             Revision save IDs shall be unique within a given editing session. [*Note*: Care should be taken to minimize the chance that two separate editing sessions are assigned the same revision save ID. *end note*]

•             …

No changes to the remainder of submitter’s proposed changes.

The following text (used in various other places with respect to rsid\*) should be adapted for use in §17.15.1.70, as appropriate: “All rsid\* attributes throughout this document of an equal value, if present, shall indicate

that those regions were modified during the same editing session.”

What we need is a complete set of all proposed changes, shown in-context.

Rex will take ownership of this and will work with the submitter, Caroline, Francis, and Murata-san to make a complete proposal.

**2017-04-07 Rex Jaeschke:**

Here’s my attempt at a revised proposal:

**Part1: §17.15.1.70, “rsid (Single Session Revision Save ID)”, p. x**

This element specifies the revision save ID that was associated with a single editing session for a document. An editing session is a span of time that begins and ends with any event that produces an editable file, such as a save or an e-mail send, and contains no such event. When revision save IDs are added to a document, they shall follow these rules:

* ~~Every editing session shall be assigned a revision save ID that is larger than all earlier ones in the same file~~
* ~~Revision save IDs should be randomly generated based on the current time (to minimize the chance that two disparate editing sessions starting with the same immediate predecessor are assigned the same revision save ID)~~
* Revision save IDs shall be unique within a given editing session. [*Note*: Care should be taken to minimize the chance that two separate editing sessions are assigned the same revision save ID. *end note*]
* Changes to document content in an editing session shall be stamped with the current revision save ID using the appropriate rsid\* attributes. All rsid\* attributes in a document, having the same value, indicate regions that were modified during the same editing session.
* An identical rsid value between two documents with the same rsidRoot (§17.15.1.71) shall indicate the same editing sessions

[Note: A revision save ID should be treated as unique within the context of all documents with the same rsidRoot value. Although in practice it is possible for two independent sessions to result in the same value, this outcome is extremely rare ~~as the values are based on the current time~~. ~~However, tThe meaning of two revision save IDs is not defined for documents with a different rsidRoot~~. If two documents have different values for rsidRoot (§17.15.1.71), an identical rsid value between the two documents does not indicate the same editing session. Applications can use this information as desired. end note]

**…**

|  |  |
| --- | --- |
| Attributes | Description |
| val (LongHexadecimalNumber Value) | Specifies a number ~~value specified as a four digit hexadecimal number), whose contents of this decimal number are~~that is interpreted based on the context of the parent XML element.[*Example*: Consider the following value for an attribute of simple typeST\_LongHexNumber: 00BE2C6C.This value is permitted, as it contains four hexadecimal digits, each an encoding of anoctet of the actual decimal number value. It can therefore be interpreted as desired inthe context of the parent XML element, *end example*]The possible values for this attribute are defined by the ST\_LongHexNumber simple type (§17.18.50). |

**Part 1: §17.15.1.71, “rsidRoot (Original Document Revision Save ID)”, p. xx**

**…**

|  |  |
| --- | --- |
| Attributes | Description |
| val (LongHexadecimalNumber Value) | Specifies a number ~~value specified as a four digit hexadecimal number), whose contents of this decimal number are~~that is interpreted based on the context of the parent XML element.[*Example*: Consider the following value for an attribute of simple typeST\_LongHexNumber: 00BE2C6C.This value is permitted, as it contains four hexadecimal digits, each an encoding of anoctet of the actual decimal number value. It can therefore be interpreted as desired inthe context of the parent XML element, *end example*]The possible values for this attribute are defined by the ST\_LongHexNumber simple type (§17.18.50). |

**Part 1: §17.15.1.72, “rsids (Listing of All Revision Save ID Values)”, p. xx**

This element specifies the set of revision save ID values for the current document. Revision save ID values refer to ~~four digit hexadecimal values~~ST\_LongHexNumber (§17.18.50) numbers that~~which~~ uniquely identify ~~an editing session in the life of~~ the editing sessions associated with the current document. ~~An editing session is the period of time between two subsequent save operations by an application.~~

[Guidance: The set of revision save IDs stored with a document only supplies information about ~~the~~ editing session~~s~~ in which ~~document components~~ were ~~last~~ saved, which can be used by applications in any manner desired. end guidance]

**Part1: §17.18.50, “ST\_LongHexNumber (Eight Digit Hexadecimal Value)”, p. xx**

This simple type specifies a number value specified as a four-octet (eight-digit) hexadecimal number, whose contents are interpreted based on the context of the parent XML element.

…

Remaining questions for MS experts:

1. Does Office use timestamps in the generation of the random rsid values - as normatively required in §17.15.1.70? Please clarify the RSID generation rules.”
2. If another implementation generates RSIDs that do not increment the suffix, as allowed by the current standard, can Word make sense of those? Basically, are RSIDs opaque strings, or are implementations required to interpret them as being in a particular sequence?

**2017-04-10 Courtenay Inchbald:**

I attach my comments, all in green.

**Part1: §17.15.1.70, “rsid (Single Session Revision Save ID)”, p. x**

This element specifies the revision save ID that was associated with a single editing session for a document. An editing session is a span of time that begins and ends with any event that produces an editable file, such as a save or an e-mail send, and contains no such event. When revision save IDs are added to a document, they shall follow these rules:

* A Revision save ID is an ST\_LongHexNumber (§17.18.50) number (see my note about rsids below, from which this is moved)
* ~~Every editing session shall be assigned a revision save ID that is larger than all earlier ones in the same file~~
* ~~Revision save IDs should be randomly generated based on the current time (to minimize the chance that two disparate editing sessions starting with the same immediate predecessor are assigned the same revision save ID)~~
* Revision save IDs shall be unique within a given editing session. [*Note*: Care should be taken to minimize the chance that two separate editing sessions are assigned the same revision save ID. *end note*] I think the first sentence is redundant because the first paragraph already says, “This element specifies the revision save ID that was associated with a single editing session for a document.”; the purpose of this bullet point is to unsure uniqueness of rsids within the document. I think the second sentence is misleading and dangerous because it removes the requirement to generate the numbers randomly, which is the only way of taking care to avoid editing sessions in parallel documents (see my note to the second of the questions at the end). All that needs to be removed from the original bullet point is “based on the current time”.
* Changes to document content in an editing session shall be stamped with the current revision save ID using the appropriate rsid\* attributes. All rsid\* attributes in a document, having the same value, indicate regions that were modified during the same editing session.
* An identical rsid value between two documents with the same rsidRoot (§17.15.1.71) shall indicate the same editing sessions

[Note: A revision save ID should be treated as unique within the context of all documents with the same rsidRoot value. Although in practice it is possible for two independent sessions to result in the same value, this outcome is extremely rare ~~as the values are based on the current time~~. ~~However, tThe meaning of two revision save IDs is not defined for documents with a different rsidRoot~~. If two documents have different values for rsidRoot (§17.15.1.71), an identical rsid value between the two documents does not indicate the same editing session. Applications can use this information as desired. end note] I much prefer the original because it only implies something that is not true of any two Word documents created from the same template. The revision wrongly states this untruth to be a fact. Obviously, I think the original problem should be addressed.

**…**

|  |  |
| --- | --- |
| Attributes | Description |
| val (LongHexadecimalNumber Value) | Specifies a number ~~value specified as a four digit hexadecimal number), whose contents of this decimal number are~~that is interpreted based on the context of the parent XML element.[*Example*: Consider the following value for an attribute of simple typeST\_LongHexNumber: 00BE2C6C.This value is permitted, as it contains four hexadecimal digits, each an encoding of anoctet of the actual decimal number value. It can therefore be interpreted as desired inthe context of the parent XML element, *end example*]The possible values for this attribute are defined by the ST\_LongHexNumber simple type (§17.18.50). |

**Part 1: §17.15.1.71, “rsidRoot (Original Document Revision Save ID)”, p. xx**

**…**

|  |  |
| --- | --- |
| Attributes | Description |
| val (LongHexadecimalNumber Value) | Specifies a number ~~value specified as a four digit hexadecimal number), whose contents of this decimal number are~~that is interpreted based on the context of the parent XML element.[*Example*: Consider the following value for an attribute of simple typeST\_LongHexNumber: 00BE2C6C.This value is permitted, as it contains four hexadecimal digits, each an encoding of anoctet of the actual decimal number value. It can therefore be interpreted as desired inthe context of the parent XML element, *end example*]The possible values for this attribute are defined by the ST\_LongHexNumber simple type (§17.18.50). |

**Part 1: §17.15.1.72, “rsids (Listing of All Revision Save ID Values)”, p. xx**

This element specifies the set of revision save ID (rsid) values for the current document. Revision save ID values refer to ~~four digit hexadecimal values~~ST\_LongHexNumber (§17.18.50) numbers that~~which~~ uniquely identify ~~an editing session in the life of~~ the editing sessions associated with the current document. ~~An editing session is the period of time between two subsequent save operations by an application.~~ The information in the second sentence belongs in the definition of rsid (see my extra bullet point in rsid above)

[Guidance: The set of revision save IDs stored with a document only includes those of ~~supplies information about the~~ editing session~~s~~ ~~in~~ which ~~document components~~ were ~~last~~ saved, which can be used by applications in any manner desired. end guidance] This is potentially confusing without a warning that it includes revision save IDs that were saved as part of the template and inherited (see my note about the note to the rsid definition).

**Part1: §17.18.50, “ST\_LongHexNumber (Eight Digit Hexadecimal Value)”, p. xx**

This simple type specifies a number value specified as a four-octet (eight-digit) hexadecimal number, whose contents are interpreted based on the context of the parent XML element.

…

Remaining questions for MS experts:

1. Does Office use timestamps in the generation of the random rsid values - as normatively required in §17.15.1.70? Please clarify the RSID generation rules.” I think the only relevant rule is that the revision save ID is randomly generated, the only reliable way of ensuring that branched copies are not assigned the same revision save ID. If generation on the basis of the date relevantly affected the number’s random generation, its generation would not be random.
2. If another implementation generates RSIDs that do not increment the suffix, as allowed by the current standard, can Word make sense of those? Basically, are RSIDs opaque strings, or are implementations required to interpret them as being in a particular sequence? (Assuming suffix means revision save ID value) if Revision Save IDs were generated in a sequence (instead of randomly) and a document branched into two copies, both of their next editing sessions would be given the same revision save ID, which would cause problems if the two copies were ever recombined.

**2017-04-12 Aarti Nankani:**

Here are the responses from the Word team on the two questions at the end:

For question 1) We use the tick count and the system time to seed the random generator, which is how we interpret "based on the current time" in the standard.

For question 2) They have to be numbers, but any 32-bit numbers are fine.

**2017-04-12 Caroline Arms:**

In this context, but not in the DR per se please, Rex, I will note that Courtenay seems to be unaware that no “random numbers” generated by computers are truly “random.”  They are pseudo-random.  And, as the expert response indicates, the algorithms use seeds and seeds are usually based on something like the system time in order to simulate randomness.  If you give the algorithm the same seed, it will typically give the same answer (or the same sequence of answers) repeatedly – a feature that is useful when running simulations.  At least that is how things were when I wrote simulation programs a few decades ago!!

I was not immediately sure about in their response is what the “tick count” is.  But [https://msdn.microsoft.com/en-us/library/system.environment.tickcount(v=vs.110).aspx](https://msdn.microsoft.com/en-us/library/system.environment.tickcount%28v%3Dvs.110%29.aspx) tells me that it’s “the number of milliseconds elapsed since the system started” adjusted by the following remark:

“Because the value of the TickCount property value is a 32-bit signed integer, if the system runs continuously, TickCount will increment from zero to Int32.MaxValue for approximately 24.9 days, then jump to Int32.MinValue, which is a negative number, then increment back to zero during the next 24.9 days. “

I see that the TickCount function in Mac OS has been deprecated for some years.  So we can’t assume that tick count is used in Office for Mac, but my guess is that something equivalent is used.

**2017-04-12 Caroline Arms:**

Question 2 was generated at the face-to-face and like Courtenay, I was puzzled by the use of “suffix.”  If I make the same assumption as Courtenay made – that suffix means rsid value, I still have a question for the MS experts.  Is the order of rsid elements within the rsids element assumed to be chronological?

**2017-04-12 Teleconference:**

A new question for the MS experts: Are the rsid elements within an rsids element assumed to be chronological?

Changes to Part 1: Y Part 2: N Part 3: N Part 4: N