DR 14-0014 — SML: Merging Cells

Status: Further Consideration Required

Subject: SML: Merging Cells

Qualifier: Request for clarification

Submitter: Charlie Clark Organization: Ecma/Clark Consulting & Research

Contact Information: [charlie.clark@clark-consulting.eu](file:///E:\Std\SC%2034\WG4\DRs\Log\2014\charlie.clark@clark-consulting.eu)

Submitter’s Defect Number: None

Supporting Document(s): None

Date Circulated by Secretariat: 2014-12-11

Deadline for Response from Editor: 2015-02-11

IS 29500 Reference(s): 29500:2012, Part 1, §18.x, “”, p. xx

Related DR(s): None

Nature of the Defect:

I'm a confused as to how to handle merged cells, specifically formatting, as this is not covered in the specification. It looks like the implementations remove all but the top-left cell of a range of merged cells. However, when formatting is applied, such as a fill, alignment or border, cells are created and assigned values. In a read/write situation, this is a problem as such cells could also be assigned values that may or may not be ignored by consuming applications.

Formatting, at least the way it's done in Excel, creates empty cells and, if necessary, sub-formats for them. Sub-formats appear necessary in the case of borders. This allows inconsistencies as such sub-formats can be completely different to the format initially applied (a merged range with a border can contain cells with different fills). A solution might to have something similar to the array that is available for formulae.

Let's consider a worksheet with value in the cells A1:D4. Values are a series of integers from 1 to 16 with A1 being 1, D1 being 4 and D4 being 16.

What happens if we merge these cells?

As far as I can tell the cells B1:D4 and A2:A4 are removed because they are now represented by A1 and they cannot be assigned values because that would introduce ambiguity.

But what about styles? If a style is now applied to A1 how does that affect the merged cells? Do they share the style? If they share the style how are "edge cases" such as borders handled?

Looking at the source of some Excel files I can guess at some ways of handling this but the specification is silent.

Solution Proposed by the Submitter:

None

Schema Change(s) Needed:

No

**Editor’s Response:**

**2014-12-16 Chris Rae:**

If I'm understanding your issue correctly, I think it is out of scope for the standard, and not something that Microsoft have documented elsewhere either. The intended scope of the standard is to describe the markup that can appear in documents and what that means, but it is not intended to describe how applications should perform operations on documents. So, for example, it doesn't cover how to paginate WordprocessingML documents or how to create and parse a calculation chain in SpreadsheetML.

I know that there are a couple of places where the standard does describe runtime operations, but it's not supposed to.

**2014-12-16 Charlie Clark:**

> If I'm understanding your issue correctly, I think it is out of scope …

But this \*does\* cover markup appearing in a document. Syntax \*must\* be given a semantic context.

> I know that there are a couple of places where the standard does describe runtime operations, but it's not supposed to.

My point is that this shouldn't have anything to do with runtime and is all about interoperability: if cells that have been merged are reduced to the top-left cell of a range then they can safely be removed from a document; if they persist how should values assigned to them be interpreted?

**2014-12-16 Chris Rae:**

Ah - I think I understand a little better now. The case is something like this, where an A1:A2 span exists but A2 also has a (different) value:

<mergeCells count="1">

<mergeCell ref="A1:A2"/>

</mergeCells>

<row r="1">

<c r="A1" s="1">

<v>1</v>

</c>

</row>

<row r="2">

<c r="A2" s="10"/>

</row>

Out of curiosity, I just tried this in Excel and it reported the file as corrupt.

You're right, there appears to be very little in the standard regarding what to do about these. I'll work up some updated wording for the standard for WG4 to review.

**2014-12-16 Charlie Clark:**

I've created files like that that haven't been reported as corrupt; they are valid! ;-)

This seems to be \*essential\* for styling the range of cells that has been merged. This really needs clarification. The naive interpretation that the style applied to the "rump" cell should be applied to the cells if demerged/displayed does not seem to be the case, which is a pity as it would allow merged cells to be removed I can't think of what a relevant constraint would look like.

> You're right, there appears to be very little in the standard regarding what to do about these. I'll work up some updated wording for the standard for WG4 to review.

Thanks. My suggestion would be that if cells are required (such as for styling that they be kept in the mergedCells node where constraints could be defined and enforced.

**2015-02-24/26 Seattle F2F Meeting:**

Chris agreed that there was something to fix here. He’ll work on a proposal.

**2015-11-24 Charlie Clark:**

I've done some further digging the spec and it really isn't clear how this should be done (Part 4 seems to contain the "red-haired cousin" using VML and the Excel 2016 implementation seems a victim of this confusion. As far as I can tell clear dependency has to exist between comments and a worksheet. Maybe a relationship should also exist between the worksheet's canvas (drawingML) and a comment in the same way one exists between a chart and a canvas. Otherwise the information in the commentsPr (anchor, text formatting, etc.) would seem to be sufficient and any kind of canvas can be disintermediated.

[In separate mail later that day]

Part 1 probably needs to make an explicit reference to this implementation. A normative, non-VML implementation is what I'm really after, I guess.

I also noticed my own error in the relationshipType:

<http://schemas.openxmlformats.org/officeDocument/2006/relationships/comments>

(plural) is the correct spelling. Given that a drawing has <http://schemas.openxmlformats.org/officeDocument/2006/relationships/drawing>

(singular). But both are 1:1 relationships, are there any rules for these?

Can we do a lookup somehow to validate? And why is the mime-type not used?

**2016-02-29/03-02 Barcelona Meeting:**

Response from Darrin’s team:

Add a new Implementer note. Some questions we feel out of scope as they cover implementation details (such as what should be done when a user merges a range of cells that have values in them). The one question which seems in-scope is how to handle cases where cells other than the top-left of a merge cell are actually specified in the sheet content. Although it’s not explicit, we do feel there’s an implication in the standard that such files are not valid (while Excel can roundtrip this state we cannot produce it).

Charlie replied:

Excel does this whenever you have non-uniform styling for a merged range: typically borders.

Eg. cells a1:d4 are merged and a black, solid border is applied to the edges. Excel creates cells in order to style them for a1:d1, a2:a4, d2:d4 and b4:c4.

In beautiful ASCII:

----

| |

| |

----

Ideally, I suspect there would be a separate style definition (xfId) for the merge range. But this would require a change to the specification. In its absence, it would seem to make sense to accept and document Excel's (less desirable) behaviour.

Rex wrote:

If something is implementation-dependent, we should say so. But then a conforming implementation is obliged to document its behavior. Otherwise, we can say it’s unspecified, but saying something explicitly seems to be called for here.

**2016-08-08 Aarti Nankani:**

We at MS think this behavior should be made explicitly unspecified.

**2016-08-08 Charlie Clark:**

I cannot see how the proposed resolution does anything to promote interoperability. It specifically ignores the problem of cells "created" just for styling purposes and sketches no roadmap for any potential improvements [that] could resolve the problem:

* overlays as in ODF
* an extension to the format, such as the one I proposed

Interoperability would be relegated to reverse engineering application behaviour; this the raison d'être behind an open Ecma specification in the first place.

**2016-08-08 Francis Cave:**

The issue appears to boil down to the fact that there is no standardized definition of what it means to merge a range of cells. The nearest that the standard comes to doing this is in §L.2.2.13, i.e., in the (informative) Primer, where it says:

"L.2.2.13 Merged Cells

<mergeCells>

<mergeCell ref="D13:H14"/>

</mergeCells>

In the example, cells D13:H14 have been merged into a single, larger cell. Note that the cell table itself doesn't reflect this merge, but it does reflect the data content and formatting. Specifically, the top-left cell in a merged collection of cells contains the value, and all the cells reflect the various border formatting."

The last sentence sounds normative, although the Primer is non-normative. It contradicts what is said in the example (also non-normative) in §18.3.1.55:

"[Example:

This example shows that three ranges are merged. The formatting and content for the merged range is always stored in the top left cell.

<mergeCells>

<mergeCell ref="C2:F2"/>

<mergeCell ref="B19:C20"/>

<mergeCell ref="E19:G19"/>

</mergeCells>

end example]"

The safest—but least helpful—approach would be to say in §18.3.1.55 that when a range of cells is merged, the content and formatting of the merged range are combined, but the process by which they are combined is unspecified/implementation-dependent. My fear is that to try to specify anything more interoperable than that is going to get us into deep water trying to define what happens in the general case when merging ranges of cells.

We should certainly make sure that §18.3.1.55 and §L.2.2.13 are consistent, which they are not at present.

**2016-08-25 Teleconference:**

Based on feedback from MS experts and Francis’ mail from 2016-08-08, (see above), we agreed that this behavior should be made explicitly unspecified.

**Action**: Rex will write up the final text and will look at making sure that §18.3.1.55 and §L.2.2.13 are consistent, per Francis’ mail.

**2016-09-20 Rex Jaeschke:**

Here are the proposed changes:

**Part 1, §18.3.1.55, “mergeCells (Merge Cells)”, p. 1641**

This collection expresses all the merged cells in the sheet.

When a range of cells is merged, the content and formatting of the merged range are combined in an unspecified manner.

…

**Part 1, §L.2.2.13, “mergeCells (Merge Cells)”, p. 4635**

In the example, cells D13:H14 have been merged into a single, larger cell. Note that the cell table itself doesn't reflect this merge, but it does reflect the data content and formatting. ~~Specifically, the top-left cell in a merged collection of cells contains the value, and all the cells reflect the various border formatting.~~

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