DR 15-0013 — SML: Handling empty rIds

Status: Open

Subject: SML: Handling empty rIds

Qualifier: Request for clarification

Submitter: Charlie Clark Organization: None

Contact Information: charlie@begeistert.org

Submitter’s Defect Number: none

Supporting Document(s): none

Date Circulated by Secretariat: 2015-10-01

Deadline for Response from Editor: 2015-12-01

IS 29500 Reference(s): none

Related DR(s): none

Nature of the Defect:

I recently came across an SML file with the following in the workbook.xml

<sheet name="Module3" sheetId="12" state="veryHidden" r:id=""/>

<sheet name="Module1" sheetId="13" state="veryHidden" r:id=""/>

<sheet name="Module2" sheetId="14" state="veryHidden" r:id=""/>

This is valid XML because r:id is required but can be empty but there is no guidance on how to deal with such sheet definitions. As there are no relations and no entries in the content types manifest, but the sheets exist when Excel opens the file, it seems like the entries are a request to the application to create the sheets.

Could you clarify this?

Here is a sample file that demonstrates this.



Solution Proposed by the Submitter:

None

Schema Change(s) Needed:

No

**Editor’s Response:**

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

Response from Darrin’s team:

Request is out of scope for documentation. The language from the standard indicates that this rId field specifies the identifier of the sheet part where the definition for the sheet is stored. The request seems to be asking about Excel’s recovery mechanisms for the scenarios where the rId doesn’t point to a valid sheet part.

FWIW the specific files that he’s looking at wouldn’t be valid as an xlsx file. The empty string rIds are used as legacy support for basic-sheets which were used for handling VBA code before the VBA project being in the file. It shouldn’t be possible to create any new files that would have those in them. Any use of that syntax outside of this very narrow scenario is viewed by excel as unreadable content.

Charlie wrote:

But this a valid XLSX file!

<sheet name="Module3" sheetId="12" state="veryHidden" r:id=""/>

<sheet name="Module1" sheetId="13" state="veryHidden" r:id=""/>

<sheet name="Module2" sheetId="14" state="veryHidden" r:id=""/> is valid, but nonsense

<sheet name="Module3" sheetId="14" state="veryHidden" /> would be invalid, but equivalent

An id that is an empty string is a direct breach of the specification that requires such relationships to be explicit and unique, but also one that conforms to the schema.

From the std: “The id attribute is of type ST\_RelationshipID, and therefore the relationship with ID must be the target of an explicit relationship from the source part, based on the context of the parent XML element.”

As this is not the case, consuming applications must implement workarounds.

Caroline wrote:

This was a request for clarification.  Crafting a response to Charlie based on the response from the Excel team seems the extent of what is needed.

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

MS’s experts say, “Yes, this is a valid xlsm file.”

Based on the discussion above, I see that the example file submitted was an xlsm file, which is definitely outside the scope of 29500. I propose that we close this DR without action and tell the submitter to submit a new DR if the problem can be reproduced in an xlxs file.

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