

Ensure all GUIDs in format examples are valid but distinguishable #422



Closed

lgolding opened this issue on May 23, 2019 · 4 comments

Assignees



Labels

2.1.0-CSD.1-TC-comment

non-substantive

resolved-fixed



lgolding commented on May 23, 2019 • edited ▾

Contributor

Submitted by: lgolding

Many of the format examples in the spec use GUIDs. Some of them are invalid, for example: "88889999-AAAA-2BBB-BCCC-DDDDEEEEEFFF", which is invalid because it doesn't match the regex pattern `"^[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[1-5][0-9a-fA-F]{3}-[89abAB][0-9a-fA-F]{3}-[0-9a-fA-F]{12}$"`.

This is actually a good thing, because it forces anyone who copies and pastes from these examples to assign a "real" GUID (at least, their files won't validate unless they do so). So go through all the examples and make sure *all* the GUIDs are invalid in that way.

We can do this after the public comment period because it's clearly non-substantive.

Also, mention this convention in §2.2, "Format examples."

lgolding added the non-substantive label on May 23, 2019

lgolding self-assigned this on May 23, 2019



lgolding changed the title ~~Ensure all GUIDs in format examples are not valid~~ **Ensure all GUIDs in format examples are invalid** on May 23, 2019

lgolding added the 2.1.0-CSD.1-TC-comment label on May 30, 2019



lgolding commented on Jul 7, 2019

Contributor

Author

I've changed my mind about this. I just can't see providing examples that are intentionally invalid. Since this was my submission to begin with, I'm changing the title from "Ensure all GUIDs in format examples are invalid" to "... are valid". I don't think it's a realistic concern that people will paste these examples into output produced by their tool.

@michaelcfanning FYI,



lgolding changed the title ~~Ensure all GUIDs in format examples are invalid~~ **Ensure all GUIDs in format examples are valid** on Jul 7, 2019



lgolding commented on Jul 7, 2019

Contributor

Author

Darn it. I was right the first time. It is much easier to understand the examples with fake GUIDs -- for example, if you use 11111... as the guid of a toolComponent, and then later you *refer* to that toolComponent with 11111... in a `toolComponentReference`, it's much easier to see the relationship than it would be with some random GUID.

@michaelcfanning, really sorry for the randomizing noise on this one. I think my first instinct was right.



lgolding changed the title ~~Ensure all GUIDs in format examples are valid~~ **Ensure all GUIDs in format examples are invalid** on Jul 7, 2019



lgolding added the resolved-fixed label on Jul 7, 2019



packet-rat commented on Jul 16, 2019

Just so you know there are folks paying attention out here: I think you really had it right the 2nd time. Intentionally providing invalid examples just doesn't "feel right".

For the scenario you outline, Can't you still produce a valid representative GUID with some easily discernible pattern?





lgolding commented on Aug 5, 2019

Contributor

Author

Hi @packet-rat, it is nice to know people are paying attention, and at such a fine level of detail!

You're right, given the regex from [StackOverflow](#)

```
/^[0-9a-f]{8}-[0-9a-f]{4}-[1-5][0-9a-f]{3}-[89ab][0-9a-f]{3}-[0-9a-f]{12}$/i
```

I could make all the example regexes have the form

```
xxxxxxxx-yyy-1111-8888-zzzzzzzzzzzz
```

... and I'd have complete freedom to make the x's, y's, and z's easily distinguishable. I'll reactivate this and address it at the next stage of the process where we have an opportunity to make non-substantive changes. (FYI, that will happen after the Committee Specification is published and before we take the next step which is to submit the document for approval as an OASIS Specification).



lgolding removed the resolved-fixed label on Aug 5, 2019



lgolding changed the title ~~Ensure all GUIDs in format examples are invalid~~ Ensure all GUIDs in format examples are valid but distinguishable on Aug 5, 2019



lgolding added the resolved-fixed label 12 days ago



lgolding closed this 12 days ago

Assignees



lgolding

Labels

2.1.0-CSD.1-TC-comment

non-substantive

resolved-fixed