
Subject: Re: Issue with unique tID in commons3

Posted by [Thomas Nygreen](#) on Tue, 28 Mar 2023 10:38:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi all,

I'll try to sum up the email discussion I had with Mark.

It is correct that there is no uniqueness constraint on UUIDs in the railML 3 XSDs. Similarly, the large number of type matching constraints on references that you find in railML 2 are gone in railML 3. Partly, there is a simple practical reason for this: Sparx Enterprise Architect, the modelling tool used for railML 3, simply does not support them. If we decided to include such constraints, we would have to add them to the XSDs after exporting them, or find another (set of) tool(s).

JAXB seems to be mapping the provided IDs to the first pattern that matches, which is not so strange. So, since rail3:tGenericID is the first type to match against, UUIDs that are valid XML IDs will be interpreted as that. The order in the type definition should therefore preferably be

```
<xs:union memberTypes="rail3:tUUID rail3:tGenericID"/>
```

This would mean all UUIDs were recognised as such. Unfortunately, I have not found a way to implement this in Sparx Enterprise Architect, as it sorts the types alphabetically.

Finally, even if we had a uniqueness constraint on UUIDs in the XSDs, this constraint would only ensure that UUIDs were not reused within one single file. As an important part of the purpose of UUIDs is to reference elements that are not present in the same file, the writing and reading systems will still have to take care of the proper use and handling of UUIDs across files.

Best regards,
Thomas
