
Subject: Re: railML 3.x: Data Modelling Patterns
Posted by on Thu, 22 Nov 2018 10:28:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Christian and all,

there is already a lot of reply for this topic, so I reduce my words (this time ;-)) to a short summary concerning two issues which are very important for me:

1) Hierarchy

I think a rather flat hierarchy has no advantage. I prefer a good (possibly deep) hierarchy especially in a very 'technical' context. I already have often the problem of needing to 'jump' very often in the railML files (when reading manually) to resolve references.

Additionally, when I made the suggestion of a possible generic model for future <TT> (with a very flat hierarchy), it was widely refused because of too less structure. So, I am probably (obviously) not the only one with this opinion.

2) Default values

I want to encourage what Christian Rößiger wrote. We cannot avoid default values. It is highly unreasonable for everyone to write "operationalStopOrdered=..." each time when there is no operational stop up to the horizon. Same applies e. g. to "guaranteedPass=..." which does not make sense when there is no pass but a stop. I would simply be more confusion than helping for someone who reads the railML file.

We have plenty of such examples in <TT>.

Best regards,
Dirk.
