Subject: Re: Extension suggestion for alternativeSectionTT
Posted by   on Thu, 02 Jul 2020 09:29:01 GMT
View Forum Message <> Reply to Message

Dear Torben,

I understand your point and I agree that multiple <sectionTT> would be an easy solution.

But
 - With multiple <sectionTT>s, you could encode different tracks between two stations. That's not the example you described, as far as I understand.
 - If you want to encode different tracks within one station, you would need different <ocpTT>s, I think.
 - If so (multiple <sectionTT>s and/or <ocpTT>s will be allowed), I would expect that they should have disjunctive operating days. That may not be necessary in your UC, but it is surely a much more common need for a "minor deviance" of a slot.
 - May be @priority can be introduced additionally to operating days. The @priority would clarify what should happen in case the operating days are not disjunctive.

> So that different arrival/departure/run times can be neglected.

That may be so in your UC but it is surely not so in general. I think railML should first orientate on general context, not on special cases at the first place.

> The alternative path(s) is only "minor" different to the
> primary path in relation to run time calculation.

Surely you are aware that there is no hard border between a "minor" difference and a major. It would rather be a grey zone in between. And that's the point where I am really concerned about multiple <sectionTT>s and/or <ocpTT>s.

Currently, one can encode all deviations ("minor" and major) of a train path by copying it. With your extension, there would be two option for the same purpose. That would make import harder (since both options should be understood) and leads to wild growth.

So, for the sake of consequence, I still have to say: A <trainPart> is designed to be the elementary, atomic and impartible part of a train. If we weaken this here today, we can hardly deny to weaken it again tomorrow there and so on...

> We could also not use the attribute @scope today as it only
> allows a "secondary" path and not "n" paths.

That's a misunderstanding. The official primary key is of a train in railML is @trainNumber+@scope+@additionalTrainNumber. So, there can be "n" paths with the same @trainNumber and @scope=primary, they only need to be counted by @additionalTrainNumber=1, 2, 3... and so on.

And more, you do actually _not_ need to copy/repeat all the <trainPart> if you only want to

encode a "minor" deviation in one <sectionTT> or <ocpTT>: You could use @scope=secondaryInner and only repeat the "inner" (minor) part of the path which deviates from the primary. Again, you could do that "n" different times by using @additionalTrainNumber=1, 2, 3...

That's the "official" way to do it now and that's why I say there would be a redundancy if you allow one more other way. However, I am no big friend of the current model, but I think that more important than that is consequence and less effort on import.

Best regards,
Dirk.

Am 01.07.2020 um 15:14 schrieb Torben Brand:
> Dear Dirk and TT development group,
>
> I will explain our UC in more detail for better
> understanding. We want to describe the n alternate path(s)
> of a train. With the path we mean the route of the train/
> the way along the network (also as explained in this forum
> posting:
> https://www.railml.org/forum/index.php?t=msg&th=710& start=0&).
> The alternative path(s) is only "minor" different to the
> primary path in relation to run time calculation. Usually
> (secondary) track 2 instead of (main) track 1 on a crossing
> loop. So that different arrival/departure/run times can be
> neglected. We would also primarily use the feature in
> trainParts forming the extension nor:patternTrains
> (norwegian extension documented on
> www.jernbanedirektoratet.no/railML)), that do not have any
> times (do not use <times>).
> As we are also concerned with "uncontrolled growth" we think
> that the solution suggested by Milan is the easiest
> (increase the multiplicity/cardinality to "n" and add
> @priority to <sectionTT> in railML 2.5. This would bloat the
> xml way less than having to create a new copy of the
> trainPart and train with only minor changes to the
> orriginal.
>
> We could also not use the attribute @scope today as it only
> allows a "secondary" path and not "n" paths.
>
> We welcome a discussion about the correct use of train@scope
> as we know it has been used somewhat differently as
> described in the wiki/in this forum posting. We think that
> the correct focus here is, as described in the wiki, is the
> train@scope is a separate slot allocation (DE:Trasse). For

> the solution with alternative sectionTTs this is within the
> same slot (order/allocation).