Subject: Re: railML 2.3 infrastructure extension proposal line sections
Posted by Torben Brand on Mon, 06 Mar 2017 08:56:18 GMT
View Forum Message <> Reply to Message

These are my answers to Christian and Dirks comments:

We also distinguish between station tracks and line tracks in other countries, and we have many railML (IS 2.x) files where one railML<track> representing a "trough line track" goes through several stations and such "switches" between line and station track from home signal to end of station and so on.

To use the suggested new element "line section" tracks would have to be segmented on path/station border. This is comparable to when the track name changes. As this also requires a segmentation of tracks in railML today.

But, I am not convinced of (nor happy with) the solution
> <trackGroups>
> <NO:lineSection>
> </NO:lineSection>
> </trackGroups>

First, we should avoid that there is a need to make such distinction (as Christian wrote: North America...).

Secondly, there may be not "hard" or "exact" definition where a station begins or ends concerning a <line>. For instance, it may start at the home signal and end at the shunting limit marker board. On a double-track line, this may be different locations depending driving direction or which track is used. So, shouldn't this distinction (if at all) be made at <track>s rather than <trackGroup>s?

We should make the value optional so you do not need to use the description if you do not distinguish between path and station, or you do not have an exact border. You can have different borders/border rules per track. You then need to segment the tracks on the border. You could distinguish per direction if you added a direction element to the description. As we do not need this in Norway we have not added it to our extension.

Last, so far, we have avoided creating a construct like a "station" in railML, by intention! (It is no mistake that you do not find a <station> in railML so far!) So, the <lineSection> should not lead to the "accidental definition" of a <station>... which then would many un-wanted discussions (such as: Why no reference from <lineSection> to <ocp>? Is it allowed that there is more than one <ocp> in one <lineSection>?).

Could you refer me to the earlier discussion about not having a station defined? I think it's a great concept to optionally refer to an ocp with an @ocpRef, either on the track or in <NO:lineSection>@type"station" The ocpRef should only go to ocp's of @operationalType"station". This for better orientation in the railML structure. Today the user needs to deduct a tracks ocp reference by other means, like absPos values of the crossSection of the ocp.

In Norway a line section can have multiple ocp's on it.
<NO:lineSection>@type"station" can have either reference to one @operationalType"station","junction", "depot" or "crossover". For @operationalType"station" the line section can also have one or more @operationalType"stoppingpoint" on it.
<NO:lineSection>@type"path" can have no ocp's on it or one or multiple @operationalType"halt","siding"or "blockSignal".

Furthermore I refer to Christian Rahmigs comment on my forum posting for "ocp". Here he mentiones that we do not need to define which tracks are on a path and which are on a station as the <track>@type defines this. The values "connectingTrack" and "sidingTrack" are paths and the values "secondaryTrack",and "stationtrack" are stations. The problem is that, as I read the railML wiki, a main track can be both in a path and in a station.

The documentation would have to be clear about the interface towards the existing sister element <locallyControlledArea>. For instance, in Norway a line section of type "station" is usually, but not always, a local controlled area. Likewise, local controlled areas can be found on track sections on the path. To summarize: it seems to me that there are two different definitions and usage to the two elements.

Rather, I would prefer to describe exactly what is the functional
(operational?) background behind <lineSection>. So my question would be:
What is the operational background behind it?

Relevant use cases:
Network statement, line description (DE:"Streckenbuch"), capacity planning. In Norway different operational rules apply pending if you are on the path or in the station area. Also values could change. For instance a @operationalType"stoppingPoint" changes value to "halt" if it's on the path.
It would help the general orientation in the railML structure by showing which ocp the hundreds of <track>@name"1" belongs to.

Another very useful effect of grouping tracks together and referencing them to an ocp would allow all objects placed on the tracks to be referred to the same ocp by proxy. Many ocsElements need to be ocp referenced. This would make things simpler. If an object by exemption needs to reference to a different ocp, that ocp could be described on the specific object.

Bob Jansen mentioned "Please note that the interlocking schema introduces TVD sections. These shouldn't conflict with line sections.». If there are TVD sections present they correspond roughly to the line sections in Norway. This as the line section border is at a home signal and this usually has a TVD section border associated to it. So the line section suggestion adheres to TVD sections.

To round up the use case: The element lineSection is useful for us, but not essential.