
Subject: [railML 2.5] Connections without track information
Posted by [Jerónimo Padilla](#) on Fri, 24 Nov 2023 12:18:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all

We are currently faced with the problem of how to model a connection between locations when no track information is available. This can happen when the connection is planned for the future (a new line is to be built and is required for capacity planning) or simply the responsible IM has not provided the number of tracks and line properties.

This is defined in RailNetEurope infrastructure systems as Segments and is used to connect TAF/TAP Primary Locations at macro level. The equivalent entity in RINF will be the Section of Lines that may exist without tracks (not mandatory). The main attributes that the Segments have are:

- Primary Location Code From (mandatory)
- Primary Location Code To (mandatory)
- Length_IM
- Length_Polyline
- Length_Lineal (mandatory)

We will appreciate any feedback from the railML community.
Best regards

Subject: Re: [railML 2.5] Connections without track information
Posted by [christian.rahmig](#) on Thu, 30 Nov 2023 10:41:24 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Jeronimo,

let me provide feedback from a railML coordinator's point of view:

Modelling a macroscopic track network in railML 2 is not that powerful as in railML 3. However, I am sure that we can realize it also with railML 2, step by step.

What is pretty clear seems to be that you need OCPs for modelling the primary locations that are the base for the connections you seek to describe. For each OCP you can use the <designator> element to provide the primary location code (PLC).

Connecting the OCPs can only be done using tracks in railML 2. You can use <trackBegin/macroscopicNode> and <trackEnd/macroscopicNode> to connect the OCPs on a macroscopic level. I would recommend you to use the element <state> to communicate the fact that a track connection is "planned" or its state is "unknown". You can also use <track>@type to express that it is a track of unknown property by using an extension value, e.g. "other:segment". The length of the track would be encoded using the @pos attribute of <trackBegin> and

<trackEnd>. Subtracting one from the other yields the length of the track. I am not quite sure about the precise semantic distinction between Length_IM, Length_Polyline and Length_Lineal, however I would assume one of these would probably be encoded here. Given that Length_Lineal is mandatory I would assume this one. This would mean that Length_Lineal would describe the actual physical length of the track. In case this is not correct, please clarify the meaning of the different length properties you need to encode. The other two may or may not need to be encoded using custom extension attributes, depending on their actual meaning.

I also understand that you are dealing with lines. Line information is encoded in railML 2 using the <trackGroups/line> element. There you can group tracks together to form a line. Although many tools, especially in the timetable domain, export double track lines using multiple connecting tracks between OCPs, it is not necessarily required to do so on a macroscopic level. Especially when describing the track state as unknown a macroscopic track reference from a line merely states that there is a rail connection between the connected OCPs. Be aware that each line has to refer to at least one track as it is syntactically required.

I hope this helps you in modeling your scenario. Let me know if you need further assistance.

Best regards
Christian
