Subject: Re: NetElements vs. Tracks vs. TrainDetectionElements vs. TvdSections
Posted by Joerg von Lingen on Mon, 05 Nov 2018 14:14:53 GMT
View Forum Message <> Reply to Message

Dear Fabiana,

Christian Rahmig wrote on 02.11.2018 16:31:
> Dear Fabiana,
>
> Am 25.10.2018 um 17:08 schrieb Fabiana Diotallevi:
>> Hello everybody,
>> since I'm new to RailML community I'll briefly introduce
>> myself: I'm Fabiana Diotallevi from NEAT (www.neat.it), an
>> Italian design and development company, with solid
>> experience in creating HW&SW solutions for mission and
>> safety critical applications.
>> At the moment we are developing a tool for drawing and
>> visualizing fully equipped railway track plans, and for
>> easily editing, checking and importing and/or exporting the
>> relative objects properties in different formats (among
>> which, of course, railML).
>
> Welcome to the railML forum! I am looking forward to learn more about
> your visualization application, because it relates to one of our first
> railML 3 use cases: Schematic Track Plan (see [1]). So, if you are going
> to attend the upcoming railML conference (14.11.2018) [2] and railML 3.1
> Dissemination workshop (13.11.2018) [3] in Praha, we may discuss in detail.
>
>> I have read the documentation regarding the Infrastructure
>> and the Interlocking Scheme, and I have some doubts on how
>> to link the trackCircuit xml representation between the
>> Infrastructure and Interlocking Scheme.
>> Consider for example the situation depicted in the attached
>> figure: my goal is to find the correct representation of the
>> netElements, the tracks, the trainDetectionElements
>> (Infrastructure Scheme) and the TvDSection (Interlocking
>> Scheme) of this very unrealistic case study.
>>
>> In the figure there are 6 trackcircuits, delimited by 5
>> joints. The trackcircuits (in the real world) are composed
>> by the the following segments:
>>
>> •   TC01 = a
>> •   TC02 = b+c+e
>> •   TC03 = d
>> •   TC04= f+h+i
>> •   TC05 = g
>> •   TC06 = l

>>
>> According to what I understood reading the railML
>> documentation, the 6 trackcircuits correspond the 6
>> TvdSections in the Interlocking Scheme, is this correct?
>
> That is correct.
>
>> Another point I would like you to confirm me, is that, if I
>> have only one operational point, in the Infrastructure
>> scheme the netElement representation corresponds to the
>> Track representation.
>
> NetElements are topology elements and thus independent from "railway
> typical" tracks and lines. The <line> as well as the <track> is located
> as NetEntity on the underlaying topology (NetElement).
>
>> In particular, I would say that the netElements and tracks
>> representation of this case study should be the following:
>>
>> •   trc01 = ne_01 = a+b
>> •   trc02 = ne_02 = c+d
>> •   trc03 = ne_03 = e+f
>> •   trc04 = ne_04 = g+h
>> •   trc05 = ne_05 = i+l
>
> Yes, this approach is possible. In this specific microscopic model, the
> location of the <track> corresponds with the <netElement>.
>
>> For what concerns the limiting joints , they should be
>> represented in the following way as trainDetectionElements:
>>
>> •   J1 = tde01 => netElementRef="ne_a01"
>> •   J2 = tde02 => netElementRef="ne_a02"
>> •   J3 = tde03 => netElementRef="ne_a03"
>> •   J4 = tde04 => netElementRef="ne_a04"
>> •   J5 = tde05 => netElementRef="ne_a05"
>
> You mean "ne_01" instead of "ne_a01", don't you?
>
>> Finally, for the TvdSection we should have:
>>
>> •   Tvd01 = TC01 -> DemarcatingTraindetector ="j1"
>> •   Tvd02 = TC02-> DemarcatingTraindetector ="j1", "j2",
>> "j3"
>> •   Tvd03 = TC03-> DemarcatingTraindetector ="j2"
>> •   Tvd04= TC04-> DemarcatingTraindetector ="j3","j4","j5"
>> •   Tvd05 = TC05-> DemarcatingTraindetector ="j4"
>> •   Tvd06 = TC06-> DemarcatingTraindetector ="j5"

\>\>
\>\> Is all of this correct?

Yes, the TVD sections are correct w.r.t. rail joints. In addition the sections Tvd01, Tvd03, Tvd05, Tvd06 shall have
DemarcatingBufferstop as each section has at least two ends.

Regards,
Jörg von Lingen - Interlocking scheme coordinator