Subject: @keepsOrientation in Track object
Posted by Benoit Pahud on Fri, 11 Jun 2021 11:57:50 GMT

Hi everyone,

I hope you are doing well.

First, I want to introduce myself. My name is Benoit Pahud, I am a master's student at EPFL in transportation and mobility, and I will graduate this summer. I am currently working with RailML in the documentation.

Today's question is about the attribute @keepsOrientation in the Track object whether it is true or false. Should it be referenced with the first netElement or directly with the previous track orientation?

I am sharing an attachment for a better understanding of the question.

Thanks :grin:

File Attachments
1) keepsOrientation-question.pdf, downloaded 314 times

Subject: Re: @keepsOrientation in Track object
Posted by christian.rahmig on Fri, 25 Jun 2021 13:13:23 GMT

Dear Benoit,

welcome to the railML community and thank you very much for this very interesting question! Both solutions can be argumented and both solutions are possible from a technical perspective. Therefore, let's push this question to the railML 3.1 users: Which solution did you implement in your railML 3.1 exporting tool?

Any feedback is very much appreciated...

Best regards
Christian

Subject: Re: @keepsOrientation in Track object
Posted by Benoit Pahud on Mon, 09 Aug 2021 19:14:53 GMT

Hi everyone,
We are still interested in your answer. For those who cannot read the attachment, here is the simple example. We have a three-netelement track represented like this (-> -> ->). Keepsorientation indicates for each one of the netelements (True-True-True). But, if we change the orientation of the middle element to be like this (-> <- ->), what would be the keepsorientation indication? Should it be (True-False-True) or (True-False-False)?

I am looking forward to read your answers.

Best,

Ben

Subject: Re: @keepsOrientation in Track object
Posted by christian.rahmig on Mon, 07 Mar 2022 12:01:07 GMT
View Forum Message <> Reply to Message

Dear all,

as there has been no requests from the community and examples for intended usage, I propose to deprecate the attribute @keepsOrientation with upcoming railML 3.2. If you have any interest in keeping this attribute, please comment here in the forum until Friday, 11.03.2022. There is already a Git issue mentioning a missing documentation:
https://development.railml.org/railml/railtopomodel/-/issues/3

Best regards
Christian

Subject: Re: @keepsOrientation in Track object
Posted by christian.rahmig on Tue, 22 Mar 2022 09:09:32 GMT
View Forum Message <> Reply to Message

Dear all,

as announced, the attribute @keepsOrientation will be marked depracted with upcoming version railML 3.2. See [1] and [2].

[1] https://development.railml.org/railml/version3/-/issues/500
[2] https://development.railml.org/railml/railtopomodel/-/issues /3

Best regards
Christian

Subject: Re: @keepsOrientation in Track object
Posted by Jan Gruteser on Wed, 21 Jun 2023 08:28:56 GMT

Dear all,

I am late to this discussion, but I currently have an implementation that relies heavily on this attribute.

At the moment I am writing my master's thesis at the Heinrich-Heine-University in Düsseldorf in the context of formal methods. We are trying to import railML 3 into the formal B Method, namely ProB (prob.hhu.de), which allows to check and validate properties of the resulting model by formal methods, animation, and also simulation in combination with visualisation.

There is a need to describe formally how the netElements are connected and this is done by (mathematical) relations which take the intrinsic coordinates of the netElements and connects them as stated in netRelations. When defining tracks this is done by linear locations with certain associatedNetElements. As far as I understood, posBegin and posEnd are always used in an increasing way, i.e. posBegin < posEnd, even when the intrinsic coordinates for the netElement according to the specified sequence would decrease. To keep the directional data correct in the formal relation, I use the attribute "keepsOrientation". For me it seems, that it exactly stores this property, i.e. it is true, if posBegin < posEnd and according to the sequence we also travel from 0.0 to 1.0 within the netElement (false otherwise).
In my opinion, this technique can be observed in the Advanced Example.

As this attribute is deprecated now: How can I obtain this information in another way?

Thank you and kind regards
Jan Gruteser

Subject: Re: @keepsOrientation in Track object
Posted by christian.rahmig on Mon, 03 Jul 2023 10:39:41 GMT

Dear Jan,

you can model linear locations spanning over several associated NetElements also without the attribute @keepsOrientation.

Let's take the element <line> in the railML Simple Example. It spans over three NetElements (in mesoscopic level). Now, let's assume that the NetElement in the middle is oriented in opposite direction. If you model the location with intrinsic coordinates, the source code looks like this:

```
<linearLocation id="lin01_lloc01" applicationDirection="both">
  <associatedNetElement netElementRef="ne_a11" intrinsicCoordBegin="0"
intrinsicCoordEnd="1" />
```

```
  <associatedNetElement netElementRef="ne_x11" intrinsicCoordBegin="1"
intrinsicCoordEnd="0" />
  <associatedNetElement netElementRef="ne_b11" intrinsicCoordBegin="0"
intrinsicCoordEnd="1" />
</linearLocation>
```

If you want to use the intrinsic positioning system (usable for length and distance measurements)
instead of intrinsic coordinates, the source code looks like this:

```
<linearLocation id="lin01_lloc02" applicationDirection="both">
  <associatedNetElement netElementRef="ne_a11">
    <linearCoordinateBegin positioningSystemRef="ips" measure="0.0"/>
    <linearCoordinateEnd positioningSystemRef="ips" measure="700.0"/>
  </associatedNetElement>
  <associatedNetElement netElementRef="ne_x11">
    <linearCoordinateBegin positioningSystemRef="ips" measure="3600.0"/>
    <linearCoordinateEnd positioningSystemRef="ips" measure="0.0"/>
  </associatedNetElement>
  <associatedNetElement netElementRef="ne_b11">
    <linearCoordinateBegin positioningSystemRef="ips" measure="0.0"/>
    <linearCoordinateEnd positioningSystemRef="ips" measure="700.0"/>
  </associatedNetElement>
</linearLocation>
```

Please use one of these two approaches for modelling the linear location of the line. In addition,
you are free to add the mileage (line kilometer reference system) as another coordinate system.
The resulting source code may look like this (as you see, this mileage is independent from the
NetElement orientation):

```
<linearLocation id="lin01_lloc03" applicationDirection="both">
  <associatedNetElement netElementRef="ne_a11">
    <linearCoordinateBegin positioningSystemRef="lps01" measure="0.0"/>
    <linearCoordinateEnd positioningSystemRef="lps01" measure="700.0"/>
  </associatedNetElement>
  <associatedNetElement netElementRef="ne_x11">
    <linearCoordinateBegin positioningSystemRef="lps01" measure="700.0"/>
    <linearCoordinateEnd positioningSystemRef="lps01" measure="4300.0"/>
  </associatedNetElement>
  <associatedNetElement netElementRef="ne_b11">
    <linearCoordinateBegin positioningSystemRef="lps01" measure="4300.0"/>
    <linearCoordinateEnd positioningSystemRef="lps01" measure="5000.0"/>
  </associatedNetElement>
</linearLocation>
```

Does this sound fine for you? Please give us a feedback or write your further questions.

Best regards
Christian

Subject: Re: @keepsOrientation in Track object
Posted by Jan Gruteser on Mon, 10 Jul 2023 17:18:18 GMT
View Forum Message <> Reply to Message

Dear Christian,

thanks for your detailed answer. This sounds completely fine for me and it no problem as long as some positional data is given. But what if I would have something minimalistic like

```
<linearLocation id="trk1_lloc">
    <associatedNetElement keepsOrientation="true" netElementRef="ne1"/>
</linearLocation>
```

Here the @keepsOrientation gives me the information that the track should be defined from (ne1, 0.0) to (ne1, 1.0). If the intrinsicCoordinateBegin and End were present, this would be no problem without keepsOrientation.

However, without any (optional) positional information, it is difficult for me to determine the direction of the track.
Do you have a solution?

Thanks and best regards
Jan

Subject: Re: @keepsOrientation in Track object
Posted by christian.rahmig on Mon, 04 Sep 2023 07:50:21 GMT
View Forum Message <> Reply to Message

Dear Jan,

thank you for your feedback and apologies for the long answering time due to vacations...

The minimalistic example you presented is syntactically correct, but it is not usable. In railML 3 development working groups we discussed the topic of positioning in the past weeks and months and we came to the conclusion that we have to set some rules in order to make modelling of positions unambiguous. The analysis of the various positioning options and the concluded modelling approach are documented in the concept presentation [1].

So, according to this approach, it is not feasible to have a linear location without both, the intrinsic coordinates and linear coordinates in the intrinsic positioning system. You have to use one of these two options.

As this concept presentation is still a draft, railML community will have the chance to decide whether this approach will become active with railML 3.3 or already with railML 3.2.

[1] https://cloud.railml.org/s/m8JGaBsARjmoD2m

Best regards
Christian