
Subject: [railml3.1] Make @applicationDirection optional
Posted by [Thomas Nygreen JBD](#) on Fri, 28 Dec 2018 23:36:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

In the 3.1-RC @applicationDirection is required for both <spotLocation>s and <linearLocation>s. Consequently it is not possible to position an infrastructure element without including @applicationDirection, even if most element types do not have any specific direction (see the discussion on @dir, including my review of elements in 2.3/2.4). I fear that this will lead to the same confusion that exists in 2.x.

The problem can be reduced by making the attribute optional, although this would still allow misuse (which is better than forcing it). I would prefer an alternate implementation that also separated elements that need a direction from the ones that do not, but I acknowledge that it would go against the general design. (After all, one is free to choose any location type for any element.)

As a side note: so far, the documentation of this attribute and its values is scarce. This increases my fear that this attribute will be misinterpreted and misused.

Subject: Re: [railml3.1] Make @applicationDirection optional
Posted by [christian.rahmig](#) on Tue, 29 Jan 2019 15:03:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Thomas,
dear all,

Am 29.12.2018 um 00:36 schrieb Thomas Nygreen:

> [...]
> In the 3.1-RC @applicationDirection is required for both
> <spotLocation>s and <linearLocation>s. Consequently it is
> not possible to position an infrastructure element without
> including @applicationDirection, even if most element types
> do not have any specific direction (see
> <https://www.railml.org/forum/index.php?t=msg&th=607>,
> including
> https://www.railml.org/forum/index.php?t=msg&th=607&goto=1985&#msg_1985).
> I fear that this will lead to the same confusion that exists
> in 2.x.
>
> The problem can be reduced by making the attribute optional,
> although this would still allow misuse (which is better than
> forcing it). I would prefer an alternate implementation that
> also separated elements that need a direction from the ones
> that do not, but I acknowledge that it would go against the

- > general design. (After all, one is free to choose any
- > location type for any element.)

The railML 3.1 implementation considers your remarks and makes the attribute @applicationDirection optional (see Trac ticket #266 [1]). By doing so, a missing attribute @applicationDirection may either mean, that the infrastructure element has no application direction or that it is unknown. Consequently, the attribute @applicationDirection has to be provided whenever the direction is known (no default value).

- > As a side note: so far, the documentation of this attribute
- > and its values is scarce. This increases my fear that this
- > attribute will be misinterpreted and misused.

Thank you for your hint. In order not to forget about this issue, I moved the Trac ticket #266 [1] into the Wiki domain.

[1] <https://trac.railml.org/ticket/266>

Best regards
Christian

--

Christian Rahmig - Infrastructure scheme coordinator
railML.org (Registry of Associations: VR 5750)
Phone Coordinator: +49 173 2714509; railML.org: +49 351 47582911
Altplauen 19h; 01187 Dresden; Germany www.railml.org
