Subject: Re: the use of @dir in railML.
Posted by christian.rahmig on Mon, 13 Jan 2020 14:09:32 GMT
View Forum Message <> Reply to Message

Dear all,

as Thomas correctly pointed out in his post [1], the discussion here has not yet been concluded. Therefore, I would like to summarize the proposal about the usage of @dir in railML 2.x:

Elements without extent (without @length attribute)
examples: balise, border, derailer, signal, stopPost...
usage: @dir describes the direction of travel, for which the element applies. Possible values are "up", "down" and "both". A missing @dir attribute means that the application direction of this element is unknown.
proposal: DEPRECATE @dir for trackCircuitBorder

Elements with extent (with @length attribute)
examples: bridge, levelCrossing, platformEdge, serviceSection...
usage: @dir describes the direction of travel, for which the element applies. Possible values are "up", "down" and "both". A missing @dir attribute means that the application direction of this element is unknown. By standard, the elements' orientation (not their application direction!) shall be always in direction of track orientation (from trackBegin towards trackEnd).
proposal: DEPRECATE @dir for brigde, levelCrossing, platformEdge, serviceSection and tunnel

Elements that describe a change
examples: axleWeightChange, clearanceGaugeChange, electrificationChange, gaugeChange, speedChange...
usage: @dir describes the direction of travel, for which the change applies. Possible values are "up", "down" and "both". A missing @dir attribute means that the application direction of this change element is unknown. By standard, the change elements' orientation (not their application direction!) shall be always in direction of track orientation (from trackBegin towards trackEnd).
proposal: DEPRECATE @dir for elements where properties cannot differ by direction of travel, e.g. axleWeightChange, clearanceGaugeChange, electrificationChange, gaugeChange, ownerChange, powerTransmissionChange, radiusChange

Please let us know if you agree with these principles from your application point of view. Specifically: is that the same understanding how you use @dir currently and does it match with the concept of how you want to use @dir in future?

[1] https:// www.railml.org/forum/index.php?t=msg&th=665&start=0& amp;

Thank you very much and best regards
Christian