
Subject: Re: extension of <levelCrossing> in railML2.4nor
Posted by [Thomas Nygreen JBD](#) on Thu, 23 May 2019 12:37:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Janne and Christian,

I agree with Christian's reservation regarding @length, and his proposal regarding @dir.

However, I would like to extend Christian's proposal for @length to allow @pos to be anywhere on the level crossing, while keeping the middle as the default. I see the following alternatives:

As today. It is not clearly defined what the reference point on a level crossing is. Any writing or reading system can define it for itself, providing flexibility, but resulting in ambiguity.

Define @pos to refer to the location of the (geometric) middle of the level crossing. Half the @length is before @pos and the other half is after. This gives clarity, but can be too unflexible for some systems. Further specification for systems that need more information is left to extensions.

Introduce one or more new attributes to allow specifying where the linear position of the level crossing. Some sub-alternatives:

A) as in railML2.4nor, with @roadStartPos and @roadEndPos, resulting in a redundancy with @length;

B) add just @roadStartPos and keep @length, so the level crossing extends from @roadStartPos to @roadStartPos + @length;

C) add an @offset (inspired by ocpTT/@offset) specifying how many metres of the @length is before @pos;

D) add a fractional @offset, e.g. 0.6 if 60 % of @length is before @pos, with a default of 0.5.

My favourite is 3C, with 2 as the fallback/default when @offset is not specified. While 3D includes a default that can be coded into the XSD, it does not operate in the same unit as the other alternatives and the sources for the information, resulting in extra calculations (and round-off errors) both when writing and when reading the information.

Best regards,
Thomas Nygreen - Common schema coordinator
