
Subject: Re: [railML 3.1] border types

Posted by [Joerg von Lingen](#) on Sun, 26 Aug 2018 06:57:01 GMT

[View Forum Message](#) <> [Reply to Message](#)

In interlocking schema of railML3 we will have two different ways to reflect these needs:

1) RestrictedArea: It can be defined as LocalOperationArea with elements inside, elements as border and elements released for local operation.

2) ElementGroup: For an interlocking (SignalBox) a group of element can be defined which are operated together like setting a group of signals to stop aspect. Although the group type "catenary" just not yet exists it can be easily added.

Best regards,
Joerg v. Lingen

Rollingstock Coordinator

On 07/06/2018 16:56, Thomas Nygreen wrote:

> christian.rahmig wrote on Mon, 04 June 2018 15:25

>> Am 29.05.2018 um 18:45 schrieb Thomas Nygreen:

>>> In Norway we discussed just a week or two ago if

>>> <border>s

>>> were suitable for specifying shunting areas etc.

>>> in

>>> stations. Would this kind of use be in line with

>>> what the

>>> element is intended for? Two questions we had was

>>> how to

>>> group borders together to actually form an area,

>>> and how to

>>> specify what kind of area it is. The former can be

>>> solved by

>>> using a common name for all borders of the same

>>> area, and

>>> the latter by using type="other:...", but creating

>>> a way to

>>> group borders together by IDREF seems preferable.

>>

>> the situation that you describe here, is better being

>> solved with a different implementation: Instead of using border

>> elements, I suggest to define an <OperationalPoint> and to locate this

>> operational point as an area covering all the affected tracks. Further, this

>> <OperationalPoint> can be specified with an attribute

>> <propOperational>@operationalType="shuntingYard".

>> Finally, the interlocking element may reference this operational

>> point.
>
>
> A shunting yard is something else than what I am trying to
> describe. What we would like to do is to define areas within
> stations for different interlocking purposes. So Jörg is
> correct. Two common uses would be for defining areas that
> can be released from the interlocking for manual operation
> (probably fits locallyControlledArea in railML 2.x, except
> that it requires tracks to be split at the borders) or areas
> that are electrically separated in the signalling system,
> such that one can be shut down for maintenance without
> shutting down the whole station. It is too long since I
> looked at the railML 3 specs to remember if there are other
> groupings that work better.
>
> christian.rahmig wrote on Mon, 04 June 2018 15:25
>> So, to conclude: I think that grouping of borders is not
>> the best solution here. Borders shall be used where there is an
>> explicit point (e.g. on the track) where e.g. the ownership changes
>> (without knowing where else it will change too).
>
>
> I agree that grouping borders is not the best solution. It
> might be that my mind is to occupied at the moment with
> solving our needs using the elements that are already in
> railML 2.x.
>
> Best regards, Thomas Nygreen
