Subject: Re: Speed Panels: types and reference to <speedChange>
Posted by   on Thu, 01 Nov 2012 15:29:15 GMT
View Forum Message <> Reply to Message

Dear Susanne and Christian,

two small thoughts to your conversation on modeling panels:

>> - gsm,
> more general: trainRadio
>
>   * fixed GSM signals indicating an GSM-R-equipped area, also the
>   negation indicating a non-GSM-R-equipped area, maybe with an
>   attribute to "enter"/"leave", also used for a "radio hole"

1)
Please consider how to handle the "virtual" aspects of all these kinds of
information: There may be line-side places where to leave/enter sections
of train radio, ETCS, catenary or such _without_ any sign/panel!

In Germany, we note such "train radio entering/leaving/changing places" in
the driver's timetable but there are normally no line-side panels. To
fully exchange or describe a timetable (may be for something like EBuLa)
we have to describe the virtual places too.

So please declare how to handle such cases:

a) Your conversation seams to be started because of the line-side "panels"
- not because of the virtual places. So may be you only want to handle the
real panels here only. But then we would need to have some kind of "train
radio change place" elements anywhere else in RailML to handle the virtual
cases, and both would be possibly redundant.

b) Alternatively, it may be wanted to handle the virtual and non-virtual
places in one structure, just to avoid redundancy. So, the "virtual:
boolean" attribute would be needed and the terminology should avoid
suggesting that there must be a real panel.

2)
In the cases we already have the "infrastructure property change element"
list, as with everything in <track>.<trackElements>... such as
<electrficationChange>:

Are you aware that you create a big amount of redundancy with the new
<panels> with type=.... enumeration which enumerates nearly all the same
we already have in <track>.<trackElements>.<...Change> ?

A reading software would always have the (additional) problems of the

kind: "What do to if I pass a panel 'electrification change' but there is no <electrificationChange> at the same place?"

I am aware that there may be panels not exactly at the same place where the change is (announcement or just problems of space). So that's why I take into account "optionally". But for the many, many cases where panels and changes are at the same place we should avoid redundancy _or_ we should explain that it is not redundancy.

If you decide that redundancy cannot be avoided here, please declare exactly how to handle the mixture of virtual and non-virtual change places in practice. I do not want to get RailML files where <panels> are used to describe virtual changes just because of an undefined situation in RailML.

Thank you,
best regards,
Dirk.