

Dear Dirk, Christian and others,

Dirk Bräuer <dirk.braeuer@irfp.de> writes:

- > 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!

For all those "virtual" changes the special <trackElements> can be used.

- Currently railML lacks the train radio change element. [1] It should be introduced at minimum with a "channel" attribute based on the "tOrientedElement" type.
- railML further lacks some ETCS area element suitable for different ETCS levels.

Maybe the track/trackTopology/border/@borderType attribute could be used and enhanced for this.

- The typical catenary panels can be defined by the element track/trackElements/trackConditions/trackCondition with the attributes length and type (lowerPantograph, mainPowerSwitchOff).

Otherwise use electrificationChange/@type for non-electrified sections.

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

Yes we are aware of this fact. We don't want to repeat the information on the panel but to refer to its definition, e.g.

```
<signal id="s1" pos="10.5" dir="up" assembly="pole">  
  <signalFrame height="2">
```

```
<speed kind="announcement" switchable="false"
  speedChangeRef="sc1"/>
</signalFrame>
<signalFrame height="2.3">
  <trackCondition kind="execution" switchable="false"
    trackConditionRef="tr1"/>
</signalFrame>
</signal>
```

That's probably not the latest state of the signal/panel discussion.

The above examples shows why I prefer to define special sub-elements for each panel type instead of some general panel type.

- > 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?"

The XML validator should highlight an error if there is no such electrificationChange element referred to. But it can't detect if the appropriate <electrificationChange> element is to far away from its announcement panel.

I hope to clarify a bit and to correctly understand Christian's idea. ;-)

Any comments are highly appreciated.

Kind regards...
Susanne

[1] <https://trac.assembla.com/railML/ticket/43>

--
Susanne Wunsch
Schema Coordinator: railML.common
