

Dear Dirk, Andreas and others,

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

>> That's why I would, with caution, pledge for being flexible here:
>> allow for an ocpTT to refer to a platform without referring to a
>> track or stop post, as well as referring via the track.

+1 for flexibility

> Please note that currently there is no <platform> in RailML in the
> "Czech sense". We do have <platformEdges> but even _one_ "Czech"
> platform would have _two_ <platformEdges>. There is, as far as I
> see, currently no possibility to "to refer to a platform".

+1

But the 'platform edge' at least has a 'name' that could be the platform name and should be duplicated in the Czech example for both platform edges in a short-term solution.

>> There may be more than one stop post that refers to the same platform
>> edge. I thought the train part should refer to its appropriate stop
>> post if it exists otherwise to the platform edge.

>
> It is not common to do so in timetables because the operational rules
> clarify which stop post to take and would always have "higher
> priority" over any timetable information of that kind.

>
> However, I agree that this could be a "theoretic wish", pure
> informational: To get a part of the operational rules (which stop post
> to take) into RailML. If we deem it necessary, we should provide the
> "stopPostRef". Again, it would be a kind of redundant.

OK, I thought of some kind of simulating software. There is no driver to stop at the appropriate position. It can be done with the now introduced stopPost element.

>>> So why not "forcing" to use <ocpTT>.trackRef to come from a train to
>>> platformEdges and stopPosts?

>>

>> Sorry, I don't understand this suggestion.

>

> In my opinion, a platform and the stop posts always belong to the same
> station. The "belonging" (linkage) of platforms and stop posts to a

> station is an infrastructure information.

It may be defined by the operationControlPoints/ocp/propEquipment/trackRef elements. An ocp defines which tracks belong to itself.

But there are problems with `_really long_` tracks through several ocps. With the help of the tracks/track/trackTopology/crossSections/crossSection/@ocpRef one may determine where the ocp is located along the "long track".

By traversing from ocp/./trackRef to the track one may detect several stopPosts and platformEdges valid for multiple ocps that may be distinct by their positions. But that's a really awful model.

Maybe we need the somewhat redundant reference from stopPost and platformEdge to the ocp for the sake of "long tracks".

> So why repeating this infrastructure information at each train? Isn't
> it simpler to express is only one time in the infrastructure?

The timetable should only express references it has knowledge about. That may be a certain track, a certain platform edge or a certain stop post for the ocpTT.

* None of the three references have to be given.

* If a stop post is referred to, the track and platform edge references are redundant and not needed but not forbidden.

* If a platform edge is referred to, the track reference is redundant and not needed but not forbidden.

> <platformEdge> gets an "ocpRef".

>

> --> We then also have the references from the stop posts to its ocp's
> via the platformEdges, but there may be stop posts without platforms
> (freight-only tracks), so: providing references from a <stopPost> to
> an <ocp>: <stopPost> gets an "ocpRef".

Thanks for pointing this out. At least for the "long tracks" we probably need these attributes. We should include Christian in this discussion, he will probably not follow each timetable thread.

We quite forgot the 'serviceSection'! It is really similar to the 'platformEdge' element. But nevertheless we should provide the appropriate reference attributes for this element, too.

Should stop posts also refer to service sections, such as parking/fueling/cleaning/... areas? I don't think so, but have too less experiences in this field.

> --> A <trainPart><ocpTT> already refers to an <ocp> and a
> <track>. With this information one can find the <platformEdges> and
> <stopPosts> through the infrastructure.

Not at all, there may be several stop posts and platform edges at a certain track for a certain ocp. If the platform height varies there have to be several platform edges for each height at the same track.

All these thoughts are only to find the best fitting model, I'm sure that only rare software tools handle these detailed information.

>>> I don't think so. I would provide only one way and wait until somebody
>>> claims and explains why there is a need for another one which is not
>>> redundant.

>>

>> I would like to ask in advance because of the experiences in other
>> cases: Changing an attribute into an element is not so easy done.

>

> I was not aware that this is about <element> or attribute. I thought
> the question is more or less attributes only.

An attribute can only occur once per element. If we need multiple information of the same type, we may allow lists in attribute (currently only practiced for 'coordinates') or define repeatable elements (common way in railML).

> However, our experience shows us that often we learn much more when
> implementing the schemes.

I'm really keen on your implementation experiences.

Kind regards...

Susanne

--

Susanne Wunsch

Schema Coordinator: railML.common