
Subject: Re: railML 2.3 infrastructure extension proposal - a new "state"

Posted by [christian.rahmig](#) on Mon, 11 Sep 2017 07:36:03 GMT

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

Dear Torben,

let me briefly comment on your remarks and questions:

Am 07.09.2017 um 10:29 schrieb Torben Brand:

- > To summarize (as I read you answer):
- > 1. time dimension only possible partially in railML 2.4 and
- > fully in railML3
- > 2. suggested railML 2.3nor extension is limited
- > 3. not possible to use more than one <infrastructure> in
- > railML 2.3

Correct.

- > [...]
- >
- > How to integrate different states in one data set
- > There are different ways to create a data set with different
- > states with a time dimension in railML2.3nor. I would like
- > to hear you opinions.
- > Alternative 1 "no" time dimension time dimension in the
- > metadata
- > A railML data set only handles one time dimension at a time
- > (snap-shot). There can be complete snap shots (1A) or delta snap shots
- > (1B). Complete snap shots could for instance be a data set
- > with a complete national network today and another data set
- > with a complete national network planned at a certain date.
- >
- > Delta snap shots would be first a data set with a complete
- > national network today and then delta data sets of
- > individual planned projects. The challenge with the first is that the
- > distinctions of the
- > changes disappear together with their the time dimension.
- > The project distinctions could somewhat be kept with the use
- > of <border> with @name and @description for the project name
- > and validity date. The challenge with the latter is to merge the
- > projects into
- > the existing network. The tracks cannot be connected as the
- > id's could be inconsistent in the different data sets. But
- > implicit information is available to place the project
- > infrastructure on the existing infrastructure through the
- > Norwegian consistent use of the values <line>@name,<absPos>
- > and <track>@name

I prefer alternative 1A: complete snap-shots of the infrastructure for different points in time. Since then all the related "scenario information" need to be put into the metadata, I suggest to add the already existing <state> sub element also to <infrastructure>. This would help to have the basic information about the infrastructure directly connected to the element.

- > Alternative 2 - with time dimension
- > Use time dimension as suggested with use @nor:state and
- > @operatingPeriodRef. There exist different sub alternatives
- > for placement.
- >
- > Alternative 2A - Integrated
- > As suggested use @nor:state and @operatingPeriodRef and
- > place them on individual elements in the structure for
- > individual element (planned) states. Place them on a
- > separate track with (planned) state if the @state spans an
- > entire track. All elements on the track inherit the change.
- > For states that span more than one ocp the state info is
- > placed on the <infrastructure> element.
- > There is a challenge using more than one <infrastructure>.
- > The wiki says "An <infrastructure> defines the header
- > container for various infrastructure versions." So multiple
- > <infrastructures> should be allowed. But the XSD defines
- > maximum of one occurrence of <infrastructure>. Can we ignore
- > this?

The wiki comment is misleading (I corrected it now). It is not planned to allow for having more than one <infrastructure> element. However, of course you are free to have infrastructure elements of different states included in this container, but the <states> element is currently limited in usage.

- > The reason we would like to use <infrastructure> for
- > different states is that having multiple time dimensions in
- > one <infrastructure> under either separate <track>s with
- > @nor:state and @operatingPeriodRef or even as loose elements
- > in the structure itself can possible create inconsistencies
- > if not kept completely cleanly structured.
- > Alternative 2A1 multiple <infrastructure> Alternative 2A2
- > with everything in one <infrastructure>
- >
- > Alternative 2B - Semi integrated time dimension time
- > dimension in the metadata
- > A work around would be to create one data set from one
- > source application with multiple railML files zipped
- > together. The individual railML files would contain
- > <infrastructures> for each time dimension. The id's and the

- > track connections between the infrastructures would have to
- > be created consistent. Loading the files together as one zip
- > creating one model with time dimension.
- > Based on what Christian writes and previous feedback I
- > suggest the following:
- >
- > Alternatives 2A1 is not valid.
- > We use alternative 1A for simple transfer between the
- > systems. More complex systems with DB handling of scenarios
- > can export either 1A, 1B or 2B depending on user choice.
- > Alternative 2A2 seems quite complex... And should be left
- > for railML3 I have made a simple illustration:
- >
- > What are your opinions?

Alternatives 1B and 2B are currently not possible, because of the very static structure of the railML schema. This means that in order to stay valid, a lot of information have to be considered in an export, even when they did not change. railML3 may solve this problem by reducing element hierarchy and limiting mandatory elements.

The complete "time dimension problem" is best to be solved with alternative 2A, but you are right: it is also the most complex one and it can only be solved with railML v3.

Other opinions and preferences?

Best regards
Christian

--

Christian Rahmig - Infrastructure scheme coordinator
railML.org (Registry of Associations: VR 5750)
Phone Coordinator: +49 173 2714509; railML.org: +49 351 47582911
Altplauen 19h; 01187 Dresden; Germany www.railml.org
