Subject: Additional questions for screen visualisation (Re: New extention element "nor:visualisationElement".)
Posted by Tobias Bregulla on Wed, 02 May 2018 23:22:14 GMT

Dear Christian, dear railML community!

We intend to support screen coordinates in our software in future
version too. Therefore we would like to have a possibility to store them
too. From our side it is not finally decided if we will use it on short
term (means: railML 2.3 with own extensions or railML 2.4) or mid term
(means: railML 3.x). Therefore we'll be more neutral in the current
decision of railML.org.

Regarding the proposed solution of railML's ticket #104 (see below) we
see some more additional questions to be answered before a new modelling:
   - What about a percentage value for screen coordinates? This would
give opportunities to fit to very different screen resolutions from
mobile phones (portrait) up to Retina displays (landscape) like we are
supporting in our VIA infrastructure database viewer project.
   - What about the adressing of more than one screen and the
cross-screen adressing?
   - What about the adressing of linear and areal elements (maybe with
out of right angles)?

Maybe a "re-animating" the visualization scheme for railML 2.4 and more
complete solution with longer development time for railML 2.5 and/or
railML 3.1 could be a more hassle free idea.

Best regards,

Tobias Bregulla and the whole Bahnkonzept team

Am 30.04.2018 um 17:06 schrieb Christian Rahmig:
> Dear all,
>
> based on the feedback from the 33rd railML Conference in Berlin two
> weeks ago, I suggest to implement the proposed solution from Trac ticket
> #104 [1] with railML 2.4, ok?
>
> Best regards
> Christian
>
>> Am 20.03.2018 um 15:35 schrieb Christian Rahmig:
>>> Dear Torben,
>>>
>>> the subject of screen coordinates is essential to the use case
>>> "Schematic Track Plan" and thus will be solved with railML 3.1 for sure.

```
>>>
>>> I read your proposed solution for railML 2.3 NOR several and I came to
>>> the conclusion that for railML 2.x there could be a simpler solution:
>>>
>>> Instead of "re-animating" the visualization scheme, I suggest to
>>> introduce a new element <screenCoord> at the same level like <geoCoord>.
>>> <screenCoord> shall have attributes @x, @y (mandatory) and @z (optional)
>>> defining a pixel position. A small example may look like this:
>>>
>>> <levelCrossing id="lcr01" pos="123.45">
>>>   <geoCoord coord="52.26125 10.58776" epsgCode="4326"/>
>>>   <screenCoord x="250" y="550"/>
>>> </levelCrossing>
>>>
>>> What do you think about that proposal? Any feedback is appreciated...
>>>
>>> Best regards
>>> Christian
```