Subject: questions & suggestions 2
Posted by Matthias Hengartner on Wed, 07 Apr 2004 16:48:23 GMT
View Forum Message <> Reply to Message

Hello again,

here some more points about miscellaneous topics:

- We have three element types (levelCrossings, tunnels, bridges) containing the attribute "posInd", which has the possible values "begin", "end", "middle" or "unknown". Farther, we have the "dir" attribute, which belongs to each line element type. I think this could cause some confusion. The use of these attributes has to be very well-defined. Is it possible to map the values of "posInd" to the values of "dir" (up=begin, down=end, both=middle) and vice versa, or not? If yes, why do we have both these attributes? If no, how should they be used?

attribute "absPos" in two parts, one for the actual (absolute) position [km], and

importing
and exporting programs can still translate it to the "traditional" pattern.

- What about an new line element type "emptyElement", which has the type "elementBaseType"? I see three ways (maybe there are more) how they could be applied:
1. For visualisation purposes
2. For elements, which have no line element specific information yet, but are intended to get such information to a later point of time
3. For line elements, which are not (yet) supported by railML. Exporting application could at least export some basic information about such an element.

I propose to integrate <emptyElement>. The remaining question is, where to locate it, as child of <trackTopology>, <trackData> or even both? Suggestions?

- Should we leave the infrastructure visualisation part as a direct child element of the <railML> root, or should we take it as a child of <infrastructure>? Or should there be a container for all thinkable types visualisations (not only for infrastructure)?

Thank you for your opinions!

Best regards,
Matthias Hengartner

Subject: First posting...
Posted by Matthias Hengartner on Wed, 07 Apr 2004 16:50:55 GMT
View Forum Message <> Reply to Message

Sorry, please don't mind the first posting!

Matthias Hengartner