Subject: Protocol developer meeting 22nd January 2016 at PSI
Posted by   on Tue, 08 Mar 2016 14:54:14 GMT
View Forum Message <> Reply to Message

Dear all,

I put down some topics and conclusions we discussed during the last developer meeting. This protocol may be used as a base for the next meeting. Feel free to comment, if I got something wrong or missed important topics.

Protocol railML meeting 22nd january 2016

1) Presentation of new railML web site by Vasco Kolmorgen

- Description of partner companies
(http://www.railml.org/en/introduction/partners.html) and software tools
(http://www.railml.org/en/introduction/software.html) need to be updated
- registration on the web site is encouraged, because some parts (e.g. calendar) are only accessable for registered users

2) review of changes in railML version 2.3

2a) TAF/TAP-services (https://trac.railml.org/ticket/250)
For now implemented as code attribute. ERA provides a list which contains a description for each service code. This list is only available as a *.pdf document, therefore it cannot be integrated in the railML scheme. The wiki description page for this new element shall contain:
- the link to the *.pdf document of the ERA
- some examples for service codes and their description from this list

2b) Connections to trains outside the railML file
(https://trac.railml.org/ticket/244)
The diskussion showed, that the current implementation misses an attribute for the key of the referenced train (train number, unique train id, etc.).
Decision for railML 2.3: the target train of a connection should be referenced by one of the following elements in descending priority:
- trainRef
- TAF/TAP train Id
- trainNumber + operator
- free text field
The connection structure will be extended in that direction

For a next major release this model should be improved / extended, discussed ideas are:
- allow referencing of train lines in terms of "connection to RE4 in direction XX"

- referencing "dummy" trains that contain only its "key" (no itinerary or times) by trainRef, to avoid having several train identification structures scattered all over the railML scheme. This is not possible with the current scheme, because parts of the train id (organizationalUnitBinding operator) are referenced only by the trainPart but not by the train

- Open issue: What's the meaning of the attributes "minConnTime" / "maxConnTime" in the connection element? Wiki description should be added, otherwise the attributes should be marked as deprecated

2c) Operating period for trainPartSequence (suggestion by iRFP)
- needed for clearer definition of the trainParts of commercial trains, if operating periods of the trainParts change within the commercial train
- reason and possible solution were explained with a presentation
- resolution: should be implemented using an any-attribute, topic will be discussed again when refactoring relation between trains and trainParts for railML3

3) Meaning and handling of the "deprecated" flag. Shall new software versions get a certification if they produce railML with deprecated attributes?

General point of view: Writing of deprecated elements should not be introduced newly, but downward compatibility within minor version steps shall persist. It should be possible to implement a new railML version partially and step by step.

Conclusions:
- Exporting programs should get a certification for minor railML version steps, although they still write deprecated elements
- Importing programs should get a certification only if they support the "replacement modeling" of a deprecated element / attribute.

4) Wiki documentation
- Changes of railML2.3 have to be documented in the wiki by the developers
- Differences between railML versions should be mentioned on the general changes site (http://wiki.railml.org/index.php?title=CO:changes)

5) Further development of railML 3
- suggestion by Leopold Kühschelm to use inheritance of abstract railML elements instead of any elements / attributes. This would result in a stricter XML scheme for custom extensions

- Discussion about common understanding of the most important railML terms: Several railML elements are used and understood in different senses by the railML developers, which makes it almost impossible to discuss the further development direction. A common glossary of some of

the most fundamental terms and elements should be created before starting work on a next major release. This includes the following elements (not complete):
* unique train id
* operational and commercial trains, trainPart
* trainGroup
* operatingPeriod, timetablePeriod
* connections between trains
* itineraries (ocpsTT)

6) Miscellaneous

Next developer meeting will take place 2nd June 2016, again at PSI Transcom GmbH, Berlin

Kind regards
Christian Rößiger

--
iRFP e. K. · Institut für Regional- und Fernverkehrsplanung
Hochschulstr. 45, 01069 Dresden
Tel. +49 351 4706819 · Fax. +49 351 4768190 · www.irfp.de
Registergericht: Amtsgericht Dresden, HRA 9347