

Dear all,

first let me introduce myself: I am Karl Jerosch and I am working in the ETCS trackside engineering department of Siemens Mobility Germany and I am participant of the railML workgroup "ETCS Track Net".

To improve the practical use of railML files as data exchange file, the work group "ETCS Track Net" suggests to add the following 4 information to be implemented in railML 3.2:

1.) a new attribute <status> providing information of the quality status of a railML file with a closed value list:
draft/verified/released

Note: This attribute could be part of e.g. <metadata>.

2.) a new attribute <fileDocumentId> providing information of the document id of the railML file (as substitution group="any"):
e.g. "ID123467890-LineX-Station1/Station2-XYZrailways"

Note 1: This attribute could be part of e.g. <metadata>.

Note 2: The existing attribute <railML><metadata> @identifier is used for other purpose and should not be used to provide an fileDocumentId.

3.) a new attribute <fileVersion> providing the file version number of a railML file with values:
00.00, ..., 99.99

Note 1: This attribute could be part of e.g. <metadata>.

Note 2: There is an existing attribute <railML>@version, which provides the version of the used railML schema, but an attribute version is missing, which provides the file version of a railML file.

4.) a new attribute <md5checksum> containing a checksum over all following file contents covering (at least) <common>, <infrastructure>, <interlocking>, <rollingstock> and <timetable> (if existing in the railML file),

and if possible also as many elements of <metadata>.

Note 1: This attribute could be part of e.g. <railML>.

Note 2: The calculation of the 128-bit hash-value of <md5checksum> shall follow the common known "message-digest-algorithm 5" which is also used in various applications, for example to check software downloads from the internet.

Note 3: By the new attribute <md5checksum> it can be detected by software tools importing a railML file, if the content was modified during the data exchange from the railML exporting tool to the railML importing tool.

This feature is important to ensure high data quality required for SIL 4 applications.

Does the community agree with the suggested extension of the data model in railML 3.2?

Background:

For example, in huge railway signalling projects with different construction stages, it is necessary to exchange railML files several times describing the same railway topology area, but with modifications in it according to the construction stages.

To avoid using the wrong file version, and to avoid unintended file modifications, the railML scheme 3.x shall provide a modelling to store the suggested information above.

best regards

Karl Jerosch
Siemens Mobility GmbH
SMO RI ML PE ENG HW&SW