

Reason and Goal Setting

Due to the constant development during the last years there exists a large number of computer programs in the railway section. In most cases a data transfer from and to other applications is necessary. For this transfer operation interfaces are needed to convert the relevant data. Caused by the historical development in most cases there is one existing specialized interface to handle the data transfer between two single programs. Due to its internal structure it cannot be used for other programs. If further programs are being included, the effort for exchanging data will increase to a non practicable level. This effort will increase further because of the changing of the data structure in connection with every new version of one of the programs to avoid any incompatibilities. Consequently a huge effort in care and maintenance is needed implicating high additional costs.

The railML® interface is also based on this data structure. The exporting program produces a valid XML file according to the appropriate schema and publishes it or hands it over directly to a specified receiving program. The receptor parses the incoming file to compute the demanded data only. The advantage is that in this mode an unspecified number of applications are able to use the exported data even if their internal structure is not entirely identical.

Structure

According to the XML syntax every document is opened with a key tag here called "railml". From this head four substructures are derived in the actual version containing the data. Each of the structures serves a specific topic. This is to reduce the effort for smaller amounts of data. To refer from one schema to another reference keys are used so that redundant information in different schemes will be avoided which might lead to an additional effort if not false information.

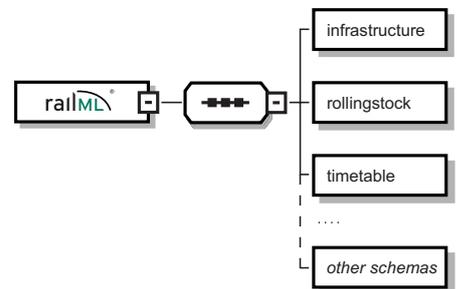


Fig.: railML.org schemas

Solution

To fit this demand a new interface called railML® has been developed by a consortium of the Fraunhofer Institute for Transportation an Infrastructure Systems IVI Dresden and the the Swiss Federal Institute of Technology Zurich based on XML (eXtensible Markup Language).

This language developed by the World Wide Web consortium is established as a quasi industry standard and is being supported by a large number of various applications.

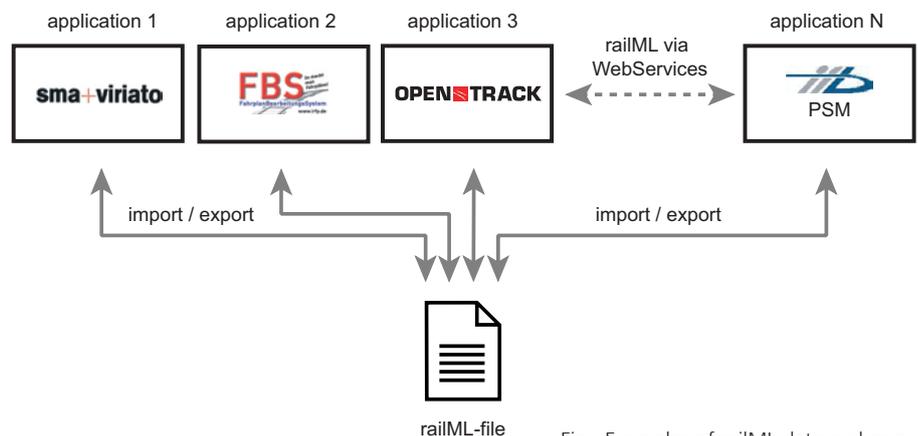


Fig.: Examples of railML data exchange

Contact:

DI Vasco Paul Kolmorgen
Dr. Daniel Huerlimann

Zeunerstrasse 1
01069 Dresden
Germany

Phone: +49 351 466 76 939
eMail: coordination@railml.org
Internet: www.railml.org

Data status: June 2014

Development

The railML® project is being consequently developed to scope with the different applications. As an open source project it is being supported and developed by all of the adjoining partners. The list of partners has meanwhile been strongly extended. The railML-consortium plays the role as the coordinator and collects and implements all changes and critics. Any interest and participation at the development

process via www.railml.org is very welcome. Programs providing interfaces according to the actual standard of railML® are welcome to apply for a certification of the supported schemas. A successful certification is required to use the railML®-Logo and to list the the railML®-functionality of the tool in the documentation of the application.

The **INFRASTRUCTURE** schema includes all infrastructural information of one line. The basis is a microscopic model, according to which one line may have a freely choosable number of detail.

The **TIMETABLE** schema serves for the transport of timetable data. Including the information from the infrastructure schema this can be inserted in an unlimitedly detailed form so that single time measuring points can be put within a one meter distance. Furthermore you have the possibility to split single courses e.g. in the context of train splitting or joining.

The **ROLLING STOCK** schema contains all data of engines and wagon material. Here the single types with their specific attributes can be saved at which the schema also respects the problem of possibly multiplied type declarations.

Version 2.2 of the ready developed schemas *Infrastructure*, *Timetable* and *Rolling stock* are released for productive use. A new schema for the *Interlocking* elements of the rail system is under development.

Each of the schemes mentioned can be freely combined with other ones of this kind so that there is no need to make an extra file for every single schema. Concerning the handle the inner structure of the railML® schemes is identical. According to the XML standard there are main elements with direct child attributes as well as child elements which themselves are freely sub-dividable. Elements and attributes can be declared as required or optional. Following the railML® philosophy as few elements as possible are being declared as required to allow the user to choose how many attributes he wishes to fill with data. All other attributes may remain empty without restricting the validity of the file. With this highly flexible data structures are available on the one hand fitted on the demands of a single user and on the other mostly universally applicable without becoming too complex to handle.

railML developers and users (Selection)



Fig.: selected railML.org partners

Applications with railML® interface (Selection)

The following applications have integrated a railML® interface. Further ones are in the development and certification phase.

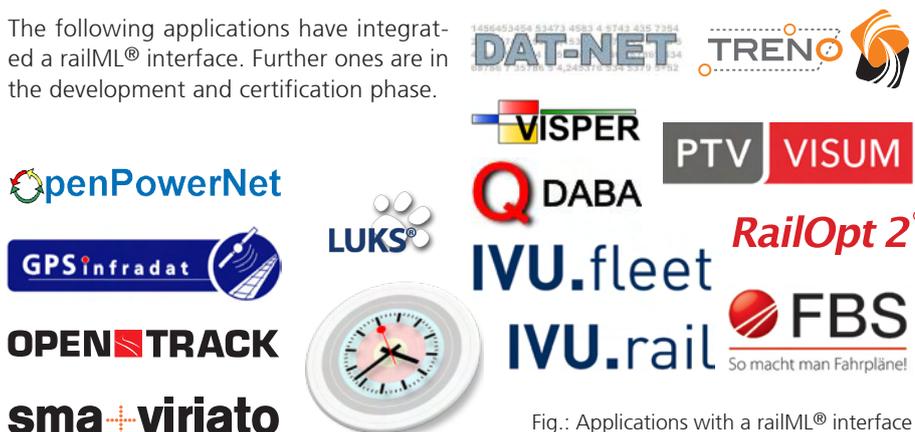


Fig.: Applications with a railML® interface

railML® certification

Programmes which passed the railML® certification process successfully are allowed to use the railML logo:

