
abweichende Laufwege, ocps)

Posted by [Simon Heller](#) on Wed, 09 Mar 2011 08:58:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

Seitens der IVU-AG gibt es drei Diskussionspunkte zu railML 2.0

1. Darstellung der Identität der Fahrzeuge über mehrere Zugabschnitte hinweg
2. Darstellung von tageweise abweichenden Laufwegen im Allgemeinen und von Ergänzungsfahrplänen der DB Netz AG im speziellen
3. Semantik des <ocp> bzw. der Wertebereich von ocp.propOperational.operationalType

Da einige Aspekte leichter in Bilder als in Worte zu fassen sind, versuche ich, diese drei Punkte als pdf-Datei anzuhängen.

Gruß
Simon Heller

--

Erstellt mit Operas revolutionärem E-Mail-Modul: <http://www.opera.com/mail/>

Subject: Re: offene Diskussionspunkte (Fahrzeugidentitaet ueber Zugabschnitte, abweichende Laufwege, ocps)

Posted by [Joachim Rubröder railML](#) on Fri, 11 Mar 2011 04:55:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hallo Simon,
hier einige erste Anmerkungen zu Deinen Diskussionspunkten:

1, Darstellung der Identität der Fahrzeuge über mehrere Zugabschnitte
So es um Verkehrliche Aspekte geht (z.B. Sitzplatzbuchungen oder Fahrgastinformation) ist aus Sicht railML die Bildung von commercial trains der richtige Ansatz.

Wenn Du diese Variante nicht nützlich finden willst, da es Dir um betriebliche Aspekte und um Leerzüge geht, so bleibt meiner Meinung als saubere Lösung nur die Bildung von Umläufen.

Die Nutzung des connection-Elements für Durchläufe ist eher eine Krücke, die wir mit der Version 2.0 gerade beseitigen wollten. Dann schon lieber ein neues Element für diesen Zweck schaffen.

Einen Einfluss hat aber auch die Art, wie die Formations gebildet werden. Wenn jeder physische Zugteil eine eigene Formation bekommt, ist die Durchbindung klar ersichtlich.

2, Darstellung von tageweise abweichenden Laufwegen

Wir waren uns beim letzten Meeting glaube ich einig, dass die DB-Modellierung mit Erganzungsfahrplanen nicht der laut railML schrferen Modellierung mit kompletten Zuglaufen entspricht. Trotzdem muss man die DB-Modellierung aus Sicht railML auch als alternative gultige Modellierungsvariante zulassen.

3, Semantik des <ocp> bzw. der Wertebereich von ocp.propOperational.operationalType

Bei den ocp handelt es sich meist um Betriebsstellen, deren Typen naturlich von Land zu Land leicht differieren. Meist handelt es sich aber doch um etwa die gleichen Typen (station, junction, blockSignal, ...). Mit der Nutzung der bestehenden Typen (so identifizierbar) und der Moglichkeit der 'other:enumeration' fur weitere Typen (nicht identifizierbar) sollten die Bedurfnisse abgedeckt sein, oder?

Ich bin gespannt, was die anderen railML-Nutzer fur eine Meinung bei diesen Diskussionspunkten vertreten und bin gerne bereit, den Konsens dann ins Schema aufzunehmen.

Viele Grusse,
Joachim Rubroder

--

----- posted via PHP Headliner -----

Subject: Re: offene Diskussionspunkte (Fahrzeugidentitaet ueber Zugabschnitte, abweichende Laufwege, ocps)

Posted by _____ on Mon, 04 Apr 2011 13:48:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hallo Simon und Joachim,

- > 1, Darstellung der Identitat der Fahrzeuge uber mehrere Zugabschnitte
- > So es um Verkehrliche Aspekte geht (z.B. Sitzplatzbuchungen oder
- > Fahrgastinformation) ist aus Sicht railML die Bildung von commercial
- > trains der richtige Ansatz.

Ich stimme Joachim hier uneingeschrankt zu.

- > Wenn Du diese Variante nicht nutzen willst, da es Dir um betriebliche
- > Aspekte und um Leerzuge geht, so bleibt meiner Meinung als saubere
- > Losung nur die Bildung von Umlaufen.

Umlaue sind indirekt eine Alternative; allerdings mochte ich anmerken,

dass meinem Verständnis nach "commercialTrains" nicht auf "öffentliche" Züge beschränkt gedacht war. Vielmehr bilden wir auch für Leerzüge und -zugteile commercialTrains - überhaupt kann sich alles, was in der RailML-Datei so rumfährt, sowohl als operational- als auch als commercialTrain vorkommen. (In unseren RailML-Exporten kommen tatsächlich alle Züge doppelt - je einmal als commercial und einmal als operational - vor, auch die Leerzüge.)

Eine Abgrenzung von "öffentlichen" Zügen ist ohnehin schwer möglich - sind Güterzüge öffentlich? Was ist mit von einer Firma bestellten Sonderfahrten, in denen dann durchaus Reisende - Firmenkunden - mitfahren? Ich will damit sagen, dass es m. E. ohnehin Willkür wäre, für welche Züge "commercialTrains" gebildet werden, wenn sie nicht für alle Züge gebildet würden.

Vielmehr sollte der Umstand, ob ein Zug dem Reiseverkehr dient, über die Eigenschaft timetable -> category -> trainUsage = passenger (und ggf. deadRun) seiner Zuggattung abgebildet werden.

- > Die Nutzung des connection-Elements für Durchläufe ist eher eine Krücke, die wir mit der Version 2.0 gerade beseitigen wollten. Dann
- > schon
- > lieber ein neues Element für diesen Zweck schaffen.

Zustimmung.

- > 2, Darstellung von tageweise abweichenden Laufwegen
- > Wir waren uns beim letzten Meeting glaube ich einig, dass die
- > DB-Modellierung mit Ergänzungsfahrplänen nicht der laut railML
- > schärferen Modellierung mit kompletten Zugläufen entspricht. Trotzdem
- > muss man die DB-Modellierung aus Sicht railML auch als alternative
- > gültige Modellierungsvariante zulassen.

Dies ist ja auch gegeben - die DB-Modellierung kann rekonstruiert werden, indem die scope-Ausprägungen in "Art des Ergänzungsfahrplans" (der DB) umgerechnet wird (1:1-Zuordnung). Was fehlt (@Joachim: hatte ich schon mal beantragt, ist aber m. E. nicht dringlich) ist die lfd. Nr. des Ergänzungsfahrplans (der DB), für die es keine Entsprechung in RailML gibt und ohne die eine 100%ige Rekonstruktion in Extremfällen nicht möglich ist.

Meinem Verständnis nach geht es bei Simons Vorschlag im Prinzip "nur" um die Frage, ob alle Alternativfahrpläne (DB: "Ergänzungsfahrpläne") zusammen mit dem Stammzuglauf eine explizite Gruppe (Element-Ebene im RailML-Sinne) bilden oder ob sich der Zusammenhang nur implizit erschließt. Nach derzeitigem RailML-Modell geht es nur implizit - alle Züge mit gleicher Zugnummer bilden die Gruppe, den Stammzuglauf erkennt man am scope=primary. Simon wünschte sich eine explizite Gruppierung, z. B. dass die Alternativ-/Ergänzungsfahrpläne eine Untergruppe oder

gemeinsame Gruppe mit dem Stammzuglauf bilden.

Dagegen sprechen (ich versuche mal aus meiner Sicht zusammenzufassen)

a) dass es eine sehr starke und womöglich nicht allgemeingültige Bindung an das DB-Modell wäre. Im internationalen Vergleich ist das DB-Modell bei weitem nicht unumstritten - ganz im Gegenteil. Schon in Österreich funktioniert es nicht mehr, und auch in Deutschland war es früher anders (Parallelläufe mit gleicher Zugnummer waren möglich, die es jetzt nicht mehr sind). Insofern sollten wir in RailML kein womöglich zu einschränkendes Modell wählen (z. B. können im Ausland immer noch Parallelläufe zulässig sein).

b) dass es nun schon einmal die Abbildungsmöglichkeit mit scope in RailML 2 gibt und Redundanzen schlecht sind, weil ein lesendes Programm dann den doppelten Aufwand hätte. Selbst wenn sich herausstellte, dass die bisherige Modellierung in RailML von der Mehrheit als schlechter empfunden wird als die DB-Modellierung, könnten wir es nur schwer korrigieren, ohne die Abwärtskompatibilität zu brechen.

Ich bin mir bewusst, dass (a) kein Argument für oder gegen eine explizite oder implizite Gruppierung der Alternativfahrpläne ist - man könnte auch eine allgemeingültige explizite Gruppierung aufstellen. Ich fürchte daher, dass (b) jetzt das entscheidende Argument gegen eine zweite Möglichkeit ist.

- > 3, Semantik des <ocp> bzw. der Wertebereich von
- > ocp.propOperational.operationalType
- > Bei den ocp handelt es sich meist um Betriebsstellen...

Ich möchte noch anmerken, dass m. E. der Begriff ocp im Deutschen Fachwortschatz am besten durch "Fahrzeitmesspunkt" wiedergegeben wird. Ein Fahrzeitmesspunkt kann, muss aber keine Betriebsstelle sein - es kann z. B. auch ein Stellwerk, Halteplatz oder anderer Bahnhofsteil sein. Früher konnte jedes Stellwerk, für das ein eigener Fahrplanauszug (BFO) erstellt wurde, ein Fahrzeitmesspunkt sein. Fahrzeitmesspunkte sind (in Deutschland) die senkrechten Linien im klassischen Bildfahrplan. Beispielsweise waren die Stellwerke Leipzig Hbf B10, B21 und B26 eigene Fahrzeitmesspunkte, aber keine Betriebsstellen (sondern Bahnhofsteile von Leipzig Hbf). Sie haben eigene Fahrplanauszüge (BFO'en) erhalten, da bei ihnen nur jeweils etwa 1/3 der Züge vorbeikam wie an der Bahnhofshalle.

Ich empfehle daher, als deutschen Begriff für ocp "Fahrzeitmesspunkt" zu verwenden.

Viele Grüße,
Dirk.

--

Erstellt mit Operas revolutionärem E-Mail-Modul: <http://www.opera.com/mail/>

Subject: Re: offene Diskussionspunkte (Fahrzeugidentitaet ueber Zugabschnitte, abweichende Laufwege, ocps)

Posted by [Susanne Wunsch railML](#) on Fri, 13 Apr 2012 17:31:57 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hallo Simon, Dirk und Joachim,

ich wärme mal einen etwas älteren Thread wieder auf.

Dirk Bräuer <dirk.braeuer@irfp.de> writes:

>> 2, Darstellung von tageweise abweichenden Laufwegen
>> Wir waren uns beim letzten Meeting glaube ich einig, dass die
>> DB-Modellierung mit Ergänzungsfahrplänen nicht der laut railML
>> schärferen Modellierung mit kompletten Zugläufen entspricht. Trotzdem
>> muss man die DB-Modellierung aus Sicht railML auch als alternative
>> gängige Modellierungsvariante zulassen.

>

> Dies ist ja auch gegeben - die DB-Modellierung kann rekonstruiert
> werden, indem die scope-Ausprägungen in "Art des Ergänzungsfahrplans"
> (der DB) umgerechnet wird (1:1-Zuordnung). Was fehlt (@Joachim: hatte
> ich schon mal beantragt, ist aber m. E. nicht dringlich) ist die
> lfd. Nr. des Ergänzungsfahrplans (der DB), für die es keine
> Entsprechung in RailML gibt und ohne die eine 100%ige Rekonstruktion
> in Extremfällen nicht möglich ist.

Das ist mit dem Attribut "additionalTrainNumber" möglich.

> Meinem Verständnis nach geht es bei Simons Vorschlag im Prinzip "nur"
> um die Frage, ob alle Alternativfahrpläne (DB: "Ergänzungsfahrpläne")
> zusammen mit dem Stammzuglauf eine explizite Gruppe (Element-Ebene im
> RailML-Sinne) bilden oder ob sich der Zusammenhang nur implizit
> erschließt. Nach derzeitigem RailML-Modell geht es nur implizit - alle
> Züge mit gleicher Zugnummer bilden die Gruppe, den Stammzuglauf
> erkennt man am scope=primary. Simon wünschte sich eine explizite
> Gruppierung, z. B. dass die Alternativ-/Ergänzungsfahrpläne eine
> Untergruppe oder gemeinsame Gruppe mit dem Stammzuglauf bilden.

Für die nächste große Version habe ich die Gruppierung der Züge mit gleicher Zugnummer in einem Element <trainGroup> im Trac-Ticket #146 vorgeschlagen. [1]

Liebe Grüße...

Susanne

[1] <http://trac.assembla.com/railML/ticket/146>

--

Susanne Wunsch

Schema Coordinator: railML.common

Subject: scope and additionalTrainNumber / compatibility to "DB:

Posted by _____ on Mon, 16 Apr 2012 14:23:09 GMT

[View Forum Message](#) <> [Reply to Message](#)

Dear all,

this 'post' is mainly to bring the discussion to an intermediate agreement. (It is always confusing to have 'open' tasks for years without knowing the opinion of the others.)

> Das ist mit dem Attribut "additionalTrainNumber" möglich.

Yes. We have cleared that "additionalTrainNumber" has to be used for "Nummer des Ergänzungsfahrplans" in case of such DB(-compatible) timetables.

Summary:

- In RailML 2.x we only have so far the possibility to 'group' trains using attributes but not with structures.

- The attributes "scope" and "additionalTrainNumber" shall be used to group trains sharing the same train number.

- For cases with compatibility to the DB's terminology of "Ergänzungsfahrpläne", the following rules apply:

DB: "Stammfahrplan" = RailML: scope=primary

DB: "Startflügel" = RailML: scope=secondaryStart

DB: "Zielflügel" = RailML: scope=secondaryEnd

DB: "Doppelfahrplan" = RailML: sscope=econdaryInner

DB: "Nummer des Ergänzungsfahrplans" = RailML: additionalTrainNumber

> Für die nächste große Version habe ich die Gruppierung der Züge mit
> gleicher Zugnummer in einem Element <trainGroup> im Trac-Ticket #146
> vorgeschlagen. [1]

We would then have the redundancy of grouping this trains with attributes and with the structure <trainGroup> as well. However, I appreciate this solution for clearness. We should have done it years ago as Simon recommended.

Best regards,
Dirk.
