Subject: Wendevorgaben
Posted by Christoph.Jobmann on Wed, 16 Mar 2011 07:37:47 GMT
View Forum Message <> Reply to Message

Hallo miteinander,

ich hatte im Rahmen des gestrigen Treffens in Innsbruck angemerkt, dass Interesse besteht, Wendevorgaben in RailML abbilden zu kÃ¶nnen. ZunÃ¤chst mÃ¶chte ich kurz ausfÃ¼hren, was ich mir darunter vorstelle:

In EinzelfÃ¤llen kann es angebracht sein, fÃ¼r bestimmte Fahrten die jeweils nachfolgende Fahrt bereits als Bestandteil des Fahrplans (nicht des Umlaufplans) zu Ã¼bergeben. Ein auf Basis solch eines Fahrplans erstellter Umlaufplan sollte solch eine Vorgabe dann natÃ¼rlich auch berÃ¼cksichtigen. In diesem Zusammenhang ist - je nach Situation - mal von einer Durchbindung, mal von einer Zwangsbindung oder auch einer Wendevorgabe die Rede.

Nach einer kurzen Diskussion kamen wir zu dem Schluss, dass dies am ehesten im Einklang mit dem timetable-Teilschema zu lÃ¶sen ist, indem jene Elemente verwendet werden, die fÃ¼r die Abbildung von UmlÃ¤ufen vorgesehen sind - auch wenn de facto noch kein (vollstÃ¤ndiger) Umlauf vorliegt.

Ein Kritikpunkt an dieser Vorgehensweise war zunÃ¤chst noch, dass diese Vorgaben nicht mehr erkennbar wÃ¤ren, wenn fÃ¼r diesen Fahrplan ein Umlaufplan erstellt und wieder exportiert wÃ¼rde. Die "fixed"-Attribute der Elemente "blockPart" und "block" sehen zwar vielversprechend aus, kÃ¶nnen jedoch nicht sinnvoll genutzt werden, da die zu fixierenden

vorliegen.

und bitte hiermit um RÃ¼ckmeldung, ob sie fÃ¼r brauchbar gehalten wird.

Es wird ein neuer Typ "tRosteringScope" eingefÃ¼hrt, welcher als Vorgabe die Werte "timetable" oder "schedule" annehmen kann.
Weiter erhÃ¤lt das Element "rostering" ein zusÃ¤tzliches Attribut namens "scope", welches den Typ "tRosteringScope" hat.
Ein "rostering"-Element mit "scope=timetable" wÃ¼rde dann Wendevorgaben / Zwangs- / Durchbindungen beschreiben. Dabei wÃ¼rden vermutlich nicht alle Elemente zum Einsatz kommen: "circulations" und "circulation" wÃ¼rde besipielsweise nicht benÃ¶tigt.
Ein "rostering"-Element mit "scope=schedule" wÃ¼rde - wie bisher - einen Umlaufplan abbilden.

FÃ¼r RÃ¼ckmeldungen jeglicher Art bin ich dankbar.

Folgende Punkte habe ich dabei noch bewusst offen gelassen:
- Wie sind Zwangsbindungen anzulegen, die Ã¼ber die Grenze eines "block"-Elements hinausgehen? [Vorschlag: fÃ¼r "scope=timetable" kann die LÃ¤nge eines "block" von der LÃ¤nge fÃ¼r "scope=schedule" abweichen.]
- Ist fÃ¼r jede Zwangsbindung ein eigenes "rostering"-Element anzulegen, oder kÃ¶nnen/sollten alle in einem untergebracht werden? [Vorschlag: Ein "rostering"-Element pro Zwangsbindung]

Christoph Jobmann

DB Mobility Logistics AG

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

Subject: Re: Wendevorgaben
Posted by Joachim Rubröder railML on Wed, 23 Mar 2011 10:02:11 GMT
View Forum Message <> Reply to Message

Hallo Christoph,

die Idee, solche durch den Fahrplan bedingten Wendevorgaben als eine Form der Umlaufplanung anzusehen finde ich konsequent. Da gehÃ¶ren sie meiner Ansicht nach hin.

Die Unterscheidung beim Element "rostering" Ã¼ber eiin zusÃ¤tzliches Attribut namens
"scope" mit Typ "tRosteringScope" finde ich auch gut. Allerdings empfinde ich die Namenswahl "schedule"/"timetable" etwas unglÃ¼cklich. Wir haben ohnehin schon zweierlei Auffassungen, was so ein "rostering" eigentlich ist. Mein Vorschlag wÃ¤re daher:

"timetable" - fÃ¼r im Fahrplan vorgesehene Wendevorgaben (Zugformationen, Verkehrstage)
"conceptional" - fÃ¼r konzeptionelle UmlaufplÃ¤ne (Fahrzeuggruppen, Normwochen)
"operational" - fÃ¼r betriebliche UmlaufplÃ¤ne (einzelne Fahrzeuge, Kalendertage)

Wenn fÃ¼r den neuen timetable-rostering die "circulations" nicht benÃ¶tigt werden, haben wir allerdings ein kleines Problem, da es sich dabei um Pflichtelemente handelt.

Was spricht denn genau dagegen, je Wendevorgabe genau ein Element circulation zu genau einen "fixed" block mit genau zwei blockParts anzulegen? Damit entsprÃ¤che jede Wendevorgabe einem eigenen kleinen

offenen Teil-Umlauf.

Viele GrÃ¼sse,
Joachim RubrÃ¶der

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

Subject: Re: Wendevorgaben
Posted by Christoph.Jobmann on Thu, 24 Mar 2011 06:14:57 GMT
View Forum Message <> Reply to Message

Hallo Joachim,

vielen Dank fÃ¼r die Antwort.

Die Anmerkungen und Bedenken kann ich nachvollziehen und bin mit den Alternativ-VorschlÃ¤gen einverstanden.

Christoph Jobmann

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

Subject: Re: Wendevorgaben
Posted by Joachim Rubröder railML on Wed, 30 Mar 2011 07:19:40 GMT
View Forum Message <> Reply to Message

Hallo Christoph,
das entsprechende scope-Feld habe ich aufgenommen.

Sind damit aus Deiner Sicht die dringensten Anforderungen fÃ¼r eine Version 2.1 erfÃ¼llt, oder benÃ¶tigst Du noch weitere Felder?

Viele GrÃ¼sse,
Joachim RubrÃ¶der

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

Subject: Re: Wendevorgaben

Posted by Joachim Rubröder railML on Fri, 01 Apr 2011 07:38:12 GMT
View Forum Message <> Reply to Message

Hallo Christoph,

als Kosequenz der jetzt gefundenen Modellierungsalternativen habe ich die alten AusprÃ¤gungsmÃ¶glichkeiten der connOperation entsprechend reduziert und die Werte 'join', 'split' und 'turnaround' als DEPRECATED deklariert (commit 383).

Viele GrÃ¼sse,
Joachim RubrÃ¶der

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

Subject: Re: Wendevorgaben
Posted by on Mon, 04 Apr 2011 12:49:52 GMT
View Forum Message <> Reply to Message

Hallo Herr Jobmann und hallo miteinander,

das Interesse an den Wendevorgaben ist für mich durchaus nachvollziehbar. Wir sollten nur klar unterscheiden (und das ist ja wohl auch so angesprochen worden) zwischen Soll-Vorgaben und dem oder den konkreten Umlaufplan/-plänen.

Bei vollständiger Angabe von Wendevorgaben an allen Zügen ließe sich der Umlaufplan eindeutig rekonstruieren. Es sollte hier aber nicht darum gehen, eine zweite Möglichkeit zur Umlaufbeschreibung in RailML einzubringen - Redundanzen sind immer schlecht vor allem für ein lesendes Programm. Vielmehr sollten es wirklich "Wendevorgaben" im Sinne von Zwangsübergängen (-> Teilmenge eines Umlaufs) oder vor allem für den Fall, dass noch gar kein Umlauf besteht, sein.

Ein etwaiger Widerspruch zwischen den Umlauf und Wendevorgabe sollte mit klarer Priorität versehen sein, vmtl. auf dem Umlauf. Die Wendevorgaben haben also den Charakter "nice to have".

Solche Wendevorgaben waren unter dem Oberbegriff "fahrplantechnische Bindungen" bereits vorgesehen. Leider ist dies (wie so vieles) in den Wirren von RailML 0.9x und der "Verenglischung" untergegangen. Wir finden die unsterblichen Reste in der fahrplantechnischen Bindungen in trainPart -> ocpTT -> connections. Dass es hier wirklich um alle fahrplantechnischen

Bindungen und nicht nur um Anschlüsse (wie die Übersetzung suggeriert) geht, erkennt man am Element "connOperation": Hier gibt es auch "turnaround", was auf die Wendevorgaben abzielte.

Natürlich sind die alten fahrplantechnischen Bindungen - egal, ob nun Anschlüsse oder Wendevorgaben - nicht hinreichend (sind sie in RailML nie gewesen, da vmtl. etwas schwieger-, äh stiefmütterlich behandelt). Vor allem fehlt eine Möglichkeit, sie verkehrstage- und zugteilabhängig zu machen. Für Anschlüsse mag es egal (oder selbsterklärend) sein, auf welchen Zugteil der Anschluss stattfindet; spätestens für Wendevorgaben ist es nicht mehr egal.

Alternativ würde ich vorschlagen, im Umlauf selbst bei den Übergängen (Abfolgen) eine Möglichkeit zu schaffen, dort anzugeben, ob der Übergang aus einer Zwangsbindung heraus festgelegt wurde oder ob er "frei" (z. B. aus einer Optimierung heraus) entstanden ist. Damit ließen sich die Wendevorgaben ermitteln durch "scannen" des Umlaufs auf Zwangsbindungen hin. Als solche Zwangsbindungs-Kennzeichnung würde theoretisch zunächst ein boolscher Wert reichen, und wir kämen damit zunächst viel einfacher zum Ziel. (Im Prinzip geht es schon indirekt durch die Zusammenfassung von blockParts zu blocks, allerdings ist dies nicht zwangsläufig auf Zwangsübergänge beschränkt.)

Fazit (mein Vorschlag):
- Ausbau des connection-Elements auf die Möglichkeit, die fahrplantechnische Bindung etwas exakter zu formulieren (z. B. explizit Zwangsübergang) und sie verkehrstage- und zugteilabhängig machen (Aufnahme von operatingPeriodRef und trainPartRef)

- Aufnahme eines boolschen Werts bei den connections zur Kennzeichnung von Übergängen im Umlauf, die auf Grund von Zwangsbindungen entstanden sind.

- Da wir eine (im Prinzip derzeit schon bestehende) Redundanz haben: Beschreibung der Priorität, dass im Zweifelsfall der konkrete Umlauf (circulation) Vorrang vor der Wendevorgabe (connection) hat und die Wendevorgabe mehr so ein Sollwert sein soll...

Viele Grüße,
Dirk Bräuer.

Am 16.03.2011, 08:37 Uhr, schrieb Christoph Jobmann
<christoph.jobmann@deutschebahn.com>:

> Hallo miteinander,
>

> ich hatte im Rahmen des gestrigen Treffens in Innsbruck angemerkt, dass
> Interesse besteht, Wendevorgaben in RailML abbilden zu kÃ¶nnen. ZunÃ¤chst
> mÃ¶chte ich kurz ausfÃ¼hren, was ich mir darunter vorstelle:
>
> In EinzelfÃ¤llen kann es angebracht sein, fÃ¼r bestimmte Fahrten die
> jeweils nachfolgende Fahrt bereits als Bestandteil des Fahrplans (nicht
> des Umlaufplans) zu Ã¼bergeben. Ein auf Basis solch eines Fahrplans
> erstellter Umlaufplan sollte solch eine Vorgabe dann natÃ¼rlich auch
> berÃ¼cksichtigen. In diesem Zusammenhang ist - je nach Situation - mal
> von
> einer Durchbindung, mal von einer Zwangsbindung oder auch einer
> Wendevorgabe die Rede.
>
> Nach einer kurzen Diskussion kamen wir zu dem Schluss, dass dies am
> ehesten im Einklang mit dem timetable-Teilschema zu lÃ¶sen ist, indem
> jene
> Elemente verwendet werden, die fÃ¼r die Abbildung von UmlÃ¤ufen
> vorgesehen
> sind - auch wenn de facto noch kein (vollstÃ¤ndiger) Umlauf vorliegt.
>
> Ein Kritikpunkt an dieser Vorgehensweise war zunÃ¤chst noch, dass diese
> Vorgaben nicht mehr erkennbar wÃ¤ren, wenn fÃ¼r diesen Fahrplan ein
> Umlaufplan erstellt und wieder exportiert wÃ¼rde. Die "fixed"-Attribute
> der Elemente "blockPart" und "block" sehen zwar vielversprechend aus,
> kÃ¶nnen jedoch nicht sinnvoll genutzt werden, da die zu fixierenden
> EntitÃ¤ten die ÃœbergÃ¤nge sind - fÃ¼r welche aber keine konkreten
> Objekte
> vorliegen.
>
> FÃ¼r diese Problemstellung habe ich schlieÃŸlich eine LÃ¶sung gefunden -
> und bitte hiermit um RÃ¼ckmeldung, ob sie fÃ¼r brauchbar gehalten wird.
>
> Diese LÃ¶sung stelle ich mir folgendermaÃŸen vor:
> Es wird ein neuer Typ "tRosteringScope" eingefÃ¼hrt, welcher als Vorgabe
> die Werte "timetable" oder "schedule" annehmen kann.
> Weiter erhÃ¤lt das Element "rostering" ein zusÃ¤tzliches Attribut namens
> "scope", welches den Typ "tRosteringScope" hat.
> Ein "rostering"-Element mit "scope=timetable" wÃ¼rde dann Wendevorgaben /
> Zwangs- / Durchbindungen beschreiben. Dabei wÃ¼rden vermutlich nicht alle
> Elemente zum Einsatz kommen: "circulations" und "circulation" wÃ¼rde
> besipielsweise nicht benÃ¶tigt.
> Ein "rostering"-Element mit "scope=schedule" wÃ¼rde - wie bisher - einen
> Umlaufplan abbilden.
>
> FÃ¼r RÃ¼ckmeldungen jeglicher Art bin ich dankbar.
>
> Folgende Punkte habe ich dabei noch bewusst offen gelassen:
> - Wie sind Zwangsbindungen anzulegen, die Ã¼ber die Grenze eines

> "block"-Elements hinausgehen? [Vorschlag: fÃ¼r "scope=timetable" kann
> die
> LÃ¤nge eines "block" von der LÃ¤nge fÃ¼r "scope=schedule" abweichen.]
> - Ist fÃ¼r jede Zwangsbindung ein eigenes "rostering"-Element anzulegen,
> oder kÃ¶nnen/sollten alle in einem untergebracht werden? [Vorschlag: Ein
> "rostering"-Element pro Zwangsbindung]
>
> Viele GrÃ¼ÃŸe
> Christoph Jobmann
>
> DB Mobility Logistics AG
>

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

Subject: Re: Wendevorgaben
Posted by Christoph.Jobmann on Wed, 06 Apr 2011 03:01:41 GMT
View Forum Message <> Reply to Message

Hallo Herr BrÃ¤uer und weitere Mitlesende,

ich hielt es zunÃ¤chst auch fÃ¼r die naheliegenste LÃ¶sung, das connection-Element zu nutzen, was aber beim Treffen in Innsbruck weitgehend abgelehnt wurde. Die Variante der rosterings mit scope timetable wurde als sauberer im Sinne von railML gehalten und daher auch durch Herrn RubrÃ¶der umgesetzt.

Den vorgeschlagenen Ausbau des connection-Elements halte ich damit nicht fÃ¼r zielfÃ¼hrend. Hinzu kommt, dass ich den Nutzen des vorgeschlagenen boolschen Attributs zur Identifikation von durch Zwangsbindung vorgegebene

Interessant finde ich jedoch den Punkt der PrioritÃ¤t fÃ¼r Wendevogaben: TatsÃ¤chlich ist es Bestandteil unserer Optimierung, dass grundsÃ¤tzlich eine Wende vorgegeben werden kann, die jedoch nicht zwangslÃ¤ufig eingehalten werden muss. Ob bzw. wie eine entsprechende Information jedoch Ã¼bergeben werden sollte ist mir zunÃ¤chst nicht klar, da wir beipsielsweise mit "Modellkosten pro nicht umgesetzter Zwangsbindung" arbeiten und nicht mit PrioritÃ¤ten, weshalb ich eine entsprechende Erweiterung vermutlich nicht fÃ¼r 2.1 fÃ¼r sinnvoll halte.

Christoph Jobmann

Dirk BrÃ¤uer wrote:
>
> Hallo Herr Jobmann und hallo miteinander,
>
> das Interesse an den Wendevorgaben ist fÃ¼r mich durchaus nachvollziehbar.
> Wir sollten nur klar unterscheiden (und das ist ja wohl auch so
> angesprochen worden) zwischen Soll-Vorgaben und dem oder den konkreten
> Umlaufplan/-plÃ¤nen.
>

> Umlaufplan eindeutig rekonstruieren. Es sollte hier aber nicht darum
> gehen, eine zweite MÃ¶glichkeit zur Umlaufbeschreibung in RailML
> einzubringen - Redundanzen sind immer schlecht vor allem fÃ¼r ein lesendes
> Programm. Vielmehr sollten es wirklich "Wendevorgaben" im Sinne von
> ZwangsÃ¼bergÃ¤ngen (-> Teilmenge eines Umlaufs) oder vor allem fÃ¼r den
Fall,
> dass noch gar kein Umlauf besteht, sein.
>
> Ein etwaiger Widerspruch zwischen den Umlauf und Wendevorgabe sollte mit
> klarer PrioritÃ¤t versehen sein, vmtl. auf dem Umlauf. Die Wendevorgaben
> haben also den Charakter "nice to have".
>
> Solche Wendevorgaben waren unter dem Oberbegriff "fahrplantechnische
> Bindungen" bereits vorgesehen. Leider ist dies (wie so vieles) in den
> Wirren von RailML 0.9x und der "Verenglischung" untergegangen. Wir finden
> die unsterblichen Reste in der fahrplantechnischen Bindungen in trainPart
> -> ocpTT -> connections. Dass es hier wirklich um alle fahrplantechnischen

> geht, erkennt man am Element "connOperation": Hier gibt es auch
> "turnaround", was auf die Wendevorgaben abzielte.
>
> NatÃ¼rlich sind die alten fahrplantechnischen Bindungen - egal, ob nun
> AnschlÃ¼sse oder Wendevorgaben - nicht hinreichend (sind sie in RailML nie
> gewesen, da vmtl. etwas schwieger-, Ã¤h stiefmÃ¼tterlich behandelt). Vor
> allem fehlt eine MÃ¶glichkeit, sie verkehrstage- und zugteilabhÃ¤ngig zu
> machen. FÃ¼r AnschlÃ¼sse mag es egal (oder selbsterklÃ¤rend) sein, auf
> welchen Zugteil der Anschluss stattfindet; spÃ¤testens fÃ¼r Wendevorgaben
> ist es nicht mehr egal.
>

> aus einer Zwangsbindung heraus festgelegt wurde oder ob er "frei" (z. B.

> Wendevorgaben ermitteln durch "scannen" des Umlaufs auf Zwangsbindungen
> hin. Als solche Zwangsbindungs-Kennzeichnung wÃ¼rde theoretisch zunÃ¤chst
> ein boolscher Wert reichen, und wir kÃ¤men damit zunÃ¤chst viel einfacher
> zum Ziel. (Im Prinzip geht es schon indirekt durch die Zusammenfassung von
> blockParts zu blocks, allerdings ist dies nicht zwangslÃ¤ufig auf

> ZwangsÃ¼bergÃ¤nge beschrÃ¤nkt.)
>
> Fazit (mein Vorschlag):
> - Ausbau des connection-Elements auf die MÃ¶glichkeit, die
> fahrplantechnische Bindung etwas exakter zu formulieren (z. B. explizit
> ZwangsÃ¼bergang) und sie verkehrstage- und zugteilabhÃ¤ngig machen (Aufnahme

> von operatingPeriodRef und trainPartRef)
>
> - Aufnahme eines boolschen Werts bei den connections zur Kennzeichnung

> sind.
>
> - Da wir eine (im Prinzip derzeit schon bestehende) Redundanz haben:
> Beschreibung der PrioritÃ¤t, dass im Zweifelsfall der konkrete Umlauf
> (circulation) Vorrang vor der Wendevorgabe (connection) hat und die
> Wendevorgabe mehr so ein Sollwert sein soll...
>

> Dirk BrÃ¤uer.

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

Subject: Re: Wendevorgaben
Posted by Christoph.Jobmann on Wed, 06 Apr 2011 03:03:52 GMT
View Forum Message <> Reply to Message

Hallo Joachim,

vielen Dank fÃ¼r die Erweiterung und die Klarstellung bezÃ¼glich des connOperation-Wertebereichs.
DarÃ¼ber hinaus benÃ¶tige ich dringend fÃ¼r 2.1 keine weiteren Felder.

Christoph Jobmann

Joachim RubrÃ¶der wrote:
>
> Hallo Christoph,
> das entsprechende scope-Feld habe ich aufgenommen.
>
> Sind damit aus Deiner Sicht die dringensten Anforderungen fÃ¼r eine
> Version 2.1 erfÃ¼llt, oder benÃ¶tigst Du noch weitere Felder?

>
> Viele GrÃ¼sse,
> Joachim RubrÃ¶der
>
>

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