Subject: [Request for railML3] Different station tracks in one <ocpTT> for different <operatingPeriod>s
Posted by   on Fri, 15 Jan 2016 13:30:37 GMT
View Forum Message <> Reply to Message

Hello everyone,

For railML 3 we suggest to extent the possibilies for modelling stop
information in a way, that different kinds of stops can be modelled
within the same <ocpTT> instance distinguished by its operating periods.
Primarily, this affects a different station track usage (trackRef,
serviceSection, platformEdge, etc.), but also other attributes e.g.
commercial vs. operational stop, onOff, etc. Cancellation of a stop on a
subset of the total operating days (which is already now possible with
the stopDescription.operatingPeriodRef attribute) should also be
included in the new structure.
Arrival and departure times should not be separated into different
operation periods because this would most likely affect the times of the
previous or following <ocpTT>. For this reason, modelling of multiple
kinds of stops is only possible, if arrival and departure times are
identical on all operating days.

--- German translation ---
für railML 3 schlagen wir vor, die Möglichkeiten der Abbildung von
Halteinformationen so zu erweitern, dass innerhalb einer ocpTT-Instanz
unterschiedliche Haltearten separiert nach Verkehrstagen abgebildet
werden können. In erster Linie betrifft dies die unterschiedliche
Bahnhofs-Gleisnutzung (trackRef, serviceSection, platFormEdge, etc.),
aber auch andere Eigenschaften, wie kommerzieller vs. Betriebshalt, nur
Ein-/Aussteigen, usw. Auch ein Halteausfall an einer Teilmenge der
Gesamtverkehrstage (derzeit schon mittels
stopDescription.operatingPeriodRef möglich) sollte mit Hilfe dieser
neuen Struktur abbildbar sein.
Ankunfts- und Abfahrtszeiten sollten nicht nach Verkehrstagen separiert
werden, da dies höchstwahrscheinlich auch die Zeiten der benachbarten
<ocpTT> beeinflussen würde. Aus diesem Grund ist die Abbildung mehrerer
Haltearten innerhalb der gleichen <ocpTT> nur möglich, wenn Ankunfts-
und Abfahrts-Zeiten identisch sind.

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

Subject: Re: [Request for railML3] Different station tracks in one <ocpTT> for different <operatingPeriod>s
Posted by   on Tue, 22 May 2018 13:42:37 GMT
View Forum Message <> Reply to Message

Dear all,

since quite some time, there is an open demand for a functional "add-on" of railML <timetable>: It should be allowed to encode different platforms at different operating days of one <trainPart> without splitting it into two or more <trainPart>s.

The matter has been discussed lengthy during the last <TT> developer meetings and telephone conferences. The pros and cons of splitting <trainPart>s vs. introducing new elements have been discussed.

A short summary of the main cons is:
- <trainPart>s are intended to be the basic atom of trains and therefore should not be divided furtherly.
- There is already a solution for the problem by using more than one <trainPart>. Any additional solution would be a redundancy.
- This redundancy leads by tendency to higher effort and therefore higher costs for import interfaces (if there would be a need to support all possibilities).

A short summary of the main pros is:
- It seems to be state of the art to enumerate different platforms at _one_ train since several common software programs show this feature.
- There are already other sub-elements of <trainPart> with an operatingPeriodRef.
- For passenger information, it may be a too demanding effort to re-merge information in an import interface which only have been splitted before in an export interface. This applies especially for passenger information with "aggregated" information over more than one operating day.
- This additional effort leads by tendency to higher costs and possibly lower acceptance of railML at our customers where nobody is interested in.

However, at the last <TT> developer meeting on 19.04.2018 at Berlin there has been a suggestion as a compromise. This would allow to enumerate several <trackInfo> (working title) elements at <trainPart>.<ocpTT>.<stopDescription>. Each <trackInfo> would have one @operatingPeriodRef and one @description. All such @operatingPeriodRef's must be disjunctive and must cover (but not exceed) the @operatingPeriodRef of the <trainPart>.

It should be mentioned that this is a minimum solution which in any case needs a usage description or use case. Therefore, concerning the main "con", in my opinion, there is no general need to support all possible (redundant) solutions in one import interface. For instance, we (iRFP) will support both technologies on export and therefore, allow the import partner to select the best solution for it's demand. So, we regard this redundancy rather as more flexibility and better acceptance of railML.

The new "add-on" is up to be decided for railML 2.4 in near future. An example and suggestion for the scheme change can be found in the Wolke at
 https://cloud.railml.org/remote.php/webdav/TT%20working%20gr

oup/railML%202.4/Vorschlag%20Saisonierte%20Gleisbelegung.pdf

With best regards,
Dirk.

Dear all,

contrary to its title, the previous forum post does not refer to a
request for railML 3, but railML 2.4. Details on the implementation in
railML can be viewed at
 http://forum.railML.org/userfiles/2018-05-22_irfp_saisoniert e-gleisbelegung-railml24.pdf.
The implementation of this proposal in version 2.4 was decided on May
23. We ask for comments and remarks.

With kind regards
Christian Rößiger

Am 22.05.2018 um 15:42 schrieb Dirk Bräuer:
> Dear all,
>
> since quite some time, there is an open demand for a functional "add-on" of railML
<timetable>: It should be allowed to encode different platforms at different operating days of one
<trainPart> without splitting it into two or more <trainPart>s.
>
> The matter has been discussed lengthy during the last <TT> developer meetings and telephone
conferences. The pros and cons of splitting <trainPart>s vs. introducing new elements have been
discussed.
>
> A short summary of the main cons is:
> - <trainPart>s are intended to be the basic atom of trains and therefore should not be divided
furtherly.
> - There is already a solution for the problem by using more than one <trainPart>. Any additional
solution would be a redundancy.
> - This redundancy leads by tendency to higher effort and therefore higher costs for import
interfaces (if there would be a need to support all possibilities).
>
> A short summary of the main pros is:
> - It seems to be state of the art to enumerate different platforms at _one_ train since several
common software programs show this feature.
> - There are already other sub-elements of <trainPart> with an operatingPeriodRef.
> - For passenger information, it may be a too demanding effort to re-merge information in an

---

import interface which only have been splitted before in an export interface. This applies especially for passenger information with "aggregated" information over more than one operating day.
> - This additional effort leads by tendency to higher costs and possibly lower acceptance of railML at our customers where nobody is interested in.
>
> However, at the last <TT> developer meeting on 19.04.2018 at Berlin there has been a suggestion as a compromise. This would allow to enumerate several <trackInfo> (working title) elements at <trainPart>.<ocpTT>.<stopDescription>. Each <trackInfo> would have one @operatingPeriodRef and one @description. All such @operatingPeriodRef's must be disjunctive and must cover (but not exceed) the @operatingPeriodRef of the <trainPart>.
>
> It should be mentioned that this is a minimum solution which in any case needs a usage description or use case. Therefore, concerning the main "con", in my opinion, there is no general need to support all possible (redundant) solutions in one import interface. For instance, we (iRFP) will support both technologies on export and therefore, allow the import partner to select the best solution for it's demand. So, we regard this redundancy rather as more flexibility and better acceptance of railML.
>
> The new "add-on" is up to be decided for railML 2.4 in near future. An example and suggestion for the scheme change can be found in the Wolke at
> https://cloud.railml.org/remote.php/webdav/TT%20working%20gr oup/railML%202.4/Vorschlag%20Saisonierte%20Gleisbelegung.pdf
>
> With best regards,
> Dirk.
>

--
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

Subject: Re: [Request for railML2.4] Different station tracks in one <ocpTT> for different <operatingPeriod>s
Posted by   on Fri, 22 Jun 2018 10:10:49 GMT
View Forum Message <> Reply to Message

Dear all,

After internal discussion, we have decided to update our proposal for the modeling of seasonal track use in stations. The following points have been changed:

- Attribute "description" is omitted and replaced by the two attributes

"track" and "platform"
- All attributes of the new element <trackInfo> are now optional

Many Greetings
Christian Rößiger

Am 04.06.2018 um 11:55 schrieb Christian Rößiger:
> Dear all,
>
> contrary to its title, the previous forum post does not refer to a
> request for railML 3, but railML 2.4. Details on the implementation in
> railML can be viewed at
> http://forum.railML.org/userfiles/2018-05-22_irfp_saisoniert e-gleisbelegung-railml24.pdf.
> The implementation of this proposal in version 2.4 was decided on May
> 23. We ask for comments and remarks.

--
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