Subject: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Tue, 15 Nov 2011 14:39:20 GMT
View Forum Message <> Reply to Message

Hello,

during our last meeting in Karlsruhe there raised the "Request for enhancement" for clearer 'connection' elements in the 'ocp' context.

This topic was already touched by some threads in the forum.

The term "connection" is very misleading. In this context it is meant for passengers/goods to change from one train to another. German term: "Anschluss"

The 'circulation' branch serves for any vehicle connections/bindings.

There is currently no provision for staff connections/bindings. They may be modelled as "passenger connections", that must be met in any case.

For one 'ocpTT' there may be multiple 'connection' elements.

Each refers to a train.

(train or trainPart?)

Additional, there may be minimal and/or maximal connection times.

(maybe we need an equivalent transfer time matrix in the infrastructure branch, regarding platform distances)

The 'connOperation' attribute shows the kind of dependancy between the current train part and the referenced train/train part:

* none
* meet
* isWaitingFor
* isExpectedBy
* ...

Do these attributes fit for all "connection" purposes? Do we need further attributes for some reason?

Does somebody provide a short but complete example (as XML file)? (Joachim?)

Any hints, questions, remarks, comments, +/-1, welcome. :-)

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Tue, 29 May 2012 15:51:08 GMT
View Forum Message <> Reply to Message

Hello,

according to ticket #126, the following attributes are missing:
 * connInfo (text)
 * reason (commercial/operational)
 * anyAttribute

If there is no further input, I would implement it this way.

Kind regards,

Joachim

Susanne Wunsch wrote:
>
> Hello,
>
> during our last meeting in Karlsruhe there raised the "Request for
> enhancement" for clearer 'connection' elements in the 'ocp' context.
>
> This topic was already touched by some threads in the forum.
>
> The term "connection" is very misleading. In this context it is meant
> for passengers/goods to change from one train to another. German term:
> "Anschluss"
>
> The 'circulation' branch serves for any vehicle connections/bindings.
>
> There is currently no provision for staff connections/bindings. They
> may be modelled as "passenger connections", that must be met in any
> case.
>
> For one 'ocpTT' there may be multiple 'connection' elements.
>

> Each refers to a train.
>
> (train or trainPart?)
>
> Additional, there may be minimal and/or maximal connection times.
>
> (maybe we need an equivalent transfer time matrix in the
> infrastructure branch, regarding platform distances)
>
> The 'connOperation' attribute shows the kind of dependancy between the
> current train part and the referenced train/train part:
>
> * none
> * meet
> * isWaitingFor
> * isExpectedBy
> * ...
>
> Do these attributes fit for all "connection" purposes? Do we need
> further attributes for some reason?
>
> Does somebody provide a short but complete example (as XML file)?
> (Joachim?)
>
> Any hints, questions, remarks, comments, +/-1, welcome. :-)
>
> Kind regards...
> Susanne
>

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Tue, 29 May 2012 20:44:13 GMT
View Forum Message <> Reply to Message

Hello,

coord@timetable.railml.org (Joachim Rubröder) writes:

> according to ticket #126, the following attributes are missing:
> * connInfo (text)
> * reason (commercial/operational)
> * anyAttribute

>
> If there is no further input, I would implement it this way.

I would like to avoid the duplication of the element's name.

'description' (some textual message) instead of 'connInfo'?

I would like to implement an open enumeration list for the 'reason'
attribute: commercial / operational / other:xxx

just my 2 cents...

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by Andreas Tanner on Wed, 30 May 2012 08:21:50 GMT
View Forum Message <> Reply to Message

Dear all,

up to now, we (I mean: IVU) have not implemented the connection element.
But maybe we will at some point, and then the following would be, from
my point of view, good to consider:
* the terminus technicus we are talking about is "transfer connections"
* they come for different purposes:
** printout products
** online information systems
** instructions to drivers (AVL (de:RBL) systems)
* they have a priority and / or a maximal delay time

I don't know the purpose of the "reason" element but to me it looks
dubious. Maybe a "type" would be more appropriate and could have the
values "external" (for passenger information and printout) and
"internal" (for AVL systems). In fact, a further differentiation could
be needed for different types of printout products and information
systems, so maybe a free subtype element would be wise.

* For the "description" element I suggest the name "messageText" or
"infoText" as this is more specific.
* optional attributes "maximalDelayTime" and "priority" would allow to
specify how reliable these connections are. AVL systems use these
parameters.

And one more point: is the "trainRef" element intended for a reference
to a /train/? Shouldn't it refer to some ocptt? First, a connection may
be between two /different/ ocps (Berlin Hbf oben / unten), and second,
an ocp may be traversed more than once.

Best regards
--Andreas.

Am 29.05.2012 22:44, schrieb Susanne Wunsch:
> Hello,
>
> coord@timetable.railml.org (Joachim Rubröder) writes:
>
>> according to ticket #126, the following attributes are missing:
>> * connInfo (text)
>> * reason (commercial/operational)
>> * anyAttribute
>>
>> If there is no further input, I would implement it this way.
>
> I would like to avoid the duplication of the element's name.
>
> 'description' (some textual message) instead of 'connInfo'?
>
> I would like to implement an open enumeration list for the 'reason'
> attribute: commercial / operational / other:xxx
>
> just my 2 cents...
>
> Kind regards...
> Susanne
>

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Wed, 30 May 2012 12:16:45 GMT
View Forum Message <> Reply to Message

Dear Andreas,

> up to now, we (I mean: IVU) have not implemented the connection element.
> But maybe we will at some point, and then the following would be, from
> my point of view, good to consider:
> * the terminus technicus we are talking about is "transfer connections"
> * they come for different purposes:
> ** printout products
> ** online information systems

> ** instructions to drivers (AVL (de:RBL) systems)
> * they have a priority and / or a maximal delay time

We like to reuse the existing "connection" element to be considered as "transfer connections" but without renaming or lots of restructuring. Therefore we planned to just introduce some new attributes for version 2.2 with a clearly defined purpose.

> I don't know the purpose of the "reason" element but to me it looks
> dubious. Maybe a "type" would be more appropriate and could have the
> values "external" (for passenger information and printout) and
> "internal" (for AVL systems). In fact, a further differentiation could
> be needed for different types of printout products and information
> systems, so maybe a free subtype element would be wise.

The purose of "reason" was to distinguish between these two main types.
* internal / commercial for printouts or online information systems
* external / operational for AVL systems
* other:xxx for further differentiation
I think "commercial/operational/other:xxx" is more specific than "internal/external".

> * For the "description" element I suggest the name "messageText" or
> "infoText" as this is more specific.

I like the name "messageText". This is appropriate for printouts and also for the AVL system, indicating an instruction for the driver.

> * optional attributes "maximalDelayTime" and "priority" would allow to
> specify how reliable these connections are. AVL systems use these
> parameters.

We have already the two attributes "minConnTime" and "maxConnTime" that could be used instead of "maximalDelayTime". The "priority" is only for the AVL usage and is hard to define. I'd like to leave this for a future version.

> And one more point: is the "trainRef" element intended for a reference
> to a /train/? Shouldn't it refer to some ocptt? First, a connection may
> be between two /different/ ocps (Berlin Hbf oben / unten), and second,
> an ocp may be traversed more than once.

If we think of printouts, this wuould be part of the "messageText" like "connection to IC 7411 on platform 7b". We could not reference an ocpTT because there is no "id" to be referenced. I think a Reference to a "train" is correct.

Kind regards,

Joachim

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Wed, 30 May 2012 12:50:15 GMT
View Forum Message <> Reply to Message

Hi to all,

> We like to reuse the existing "connection" element to be considered as
> "transfer connections" but without renaming or lots of restructuring.
> Therefore we planned to just introduce some new attributes for version
> 2.2 with a clearly defined purpose.

I agree with Joachim.

>> I don't know the purpose of the "reason" element but to me it looks
>> dubious. Maybe a "type" would be more appropriate and could have the
>> values "external" (for passenger information and printout) and
>> "internal" (for AVL systems). In fact, a further differentiation could
>> be needed for different types of printout products and information
>> systems, so maybe a free subtype element would be wise.
>
> The purose of "reason" was to distinguish between these two main types..
> * internal / commercial for printouts or online information systems
> * external / operational for AVL systems
> * other:xxx for further differentiation
> I think "commercial/operational/other:xxx" is more specific than
> "internal/external".

As far as I remember, the background for the 'connection' element comes
from the Timetabling Theories as taught for instance here in Germany:

connection = "fahrplantechnische Bindung" (zweier Züge!)
fahrplantechnische Bindung --> Anschlussbindung, Umlaufbindung,
Belegungsbindung, Personalbindung

All of these kinds of "fahrplantechnische Bindung" should then be mapped
to 'connections'. There is no distinction between external and internal as
the railway is seen as a whole. So should we do in RailML.

I agree with Joachim to avoid 'external/internal' as they could be

misunderstood as 'in RailML' or 'outside RailML', for instance only.

> We have already the two attributes "minConnTime" and "maxConnTime" that
> could be used instead of "maximalDelayTime". The "priority" is only for
> the AVL usage and is hard to define. I'd like to leave this for a future
> version.

I agree with Joachim.

>> And one more point: is the "trainRef" element intended for a reference
>> to a /train/? Shouldn't it refer to some ocptt? First, a connection may
>> be between two /different/ ocps (Berlin Hbf oben / unten), and second,
>> an ocp may be traversed more than once.
>
> If we think of printouts, this wuould be part of the "messageText" like
> "connection to IC 7411 on platform 7b". We could not reference an ocpTT
> because there is no "id" to be referenced. I think a Reference to a
> "train" is correct.

I think it is not only about print-outs. It is also about a reading software being able to "understand" the connection - either because of transferring it into another data model or doing some function with it (e. g. calculating the connecting time or, like OpenTrack, simulating the delay of the connecting service).

Concerning the problem that one OCP may occur several times at a train, it may be left to the reading software to find the right instance (nearest departure time or so). But with the problem of connections between different OCPs it is not so easy. So, from my opinion, it would be good to have at lease an optional ocpRef there.

>> I would like to implement an open enumeration list for the 'reason'
>> attribute: commercial / operational / other:xxx

I would very much agree with Susanne.

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Andreas Tanner on Wed, 30 May 2012 12:56:01 GMT
View Forum Message <> Reply to Message

> The purose of "reason" was to distinguish between these two main types.
> * internal / commercial for printouts or online information systems
> * external / operational for AVL systems
> * other:xxx for further differentiation

> I think "commercial/operational/other:xxx" is more specific than
> "internal/external".
>

Ok, but to my ears, a connection would be of some /type/ rather than it
would have a "reason".

>> * optional attributes "maximalDelayTime" and "priority" would allow to
>> specify how reliable these connections are. AVL systems use these
>> parameters.
>
> We have already the two attributes "minConnTime" and "maxConnTime" that
> could be used instead of "maximalDelayTime". The "priority" is only for
> the AVL usage and is hard to define. I'd like to leave this for a future
> version.
>

Hmm, I read the description of these attributes in the wiki, but did not
come to the end that this was their intention. If maxConnTime is the
maximal delay, what then is minConnTime?

>> And one more point: is the "trainRef" element intended for a reference
>> to a /train/? Shouldn't it refer to some ocptt? First, a connection may
>> be between two /different/ ocps (Berlin Hbf oben / unten), and second,
>> an ocp may be traversed more than once.
>
> If we think of printouts, this wuould be part of the "messageText" like
> "connection to IC 7411 on platform 7b". We could not reference an ocpTT
> because there is no "id" to be referenced. I think a Reference to a
> "train" is correct.
>
>

I agree for printouts, but this would not work for online connection
finders and AVL systems where a driver wants to see the connection the
relevant location.
You are right, the ocpTTs are not referencable, could you please open a
ticket for future versions?
Trainparts would be somewhat more precise than trains, and they have a
validity. Maybe a connection holds only on certain dates.

Best, Andreas.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Andreas Tanner on Fri, 01 Jun 2012 08:56:16 GMT

> You are right, the ocpTTs are not referencable, could you please open a
> ticket for future versions?
> Trainparts would be somewhat more precise than trains, and they have a
> validity. Maybe a connection holds only on certain dates.
>
>

Actually, adding optional identifiers to ocpTTs would be non-breaking,
so why not implement it this way for the next minor release?

--A.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Thu, 27 Sep 2012 14:38:01 GMT

Hi,
here is my conclusion for the enhancement of connection within version 2.2:

* messageText (string), optional
* connType (commercial, operational, other:xxx), optional
* ocpRef (Ref), optional

http://trac.assembla.com/railML/ticket/165
I shall implement it this way. Please check if this is sufficient for now.

The issue of referencing an ocpTT is important and has a lot of
implications. Tthat should be futher discussed and be introduced within a
version 3.0. I therefore opend a new ticket:
http://trac.assembla.com/railML/ticket/165

Kind regards,
Joachim

Andreas Tanner wrote:
>
>
>> You are right, the ocpTTs are not referencable, could you please open a
>> ticket for future versions?
>> Trainparts would be somewhat more precise than trains, and they have a
>> validity. Maybe a connection holds only on certain dates.
>>
>>

>
> Actually, adding optional identifiers to ocpTTs would be non-breaking,
> so why not implement it this way for the next minor release?
>
> --A.
>
>

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Tue, 06 Nov 2012 09:47:01 GMT
View Forum Message <> Reply to Message

Hi Joachim, Andreas, Dirk and others,

coord@timetable.railml.org (Joachim Rubröder) writes:

> here is my conclusion for the enhancement of connection within version 2.2:
>
> * messageText (string), optional
> * connType (commercial, operational, other:xxx), optional
> * ocpRef (Ref), optional
>
> http://trac.assembla.com/railML/ticket/165
> I shall implement it this way. Please check if this is sufficient for now.

There is a typo, I think you wanted to cite the following URL:

http://trac.assembla.com/railML/ticket/126

Why not to implement the 'trainPartRef' instead of 'trainRef'? It is
much more specific as Andreas already mentioned.

> Andreas Tanner wrote:
>>> Trainparts would be somewhat more precise than trains, and they have a
>>> validity. Maybe a connection holds only on certain dates.

>>> Actually, adding optional identifiers to ocpTTs would be non-breaking,
>>> so why not implement it this way for the next minor release?

Currently no ocpTT may be referred because of the missing "id" attribute
in the <ocpTT> element.

Wouldn't it be sufficient to refer to a certain 'trainPart' and an 'ocp'? A 'trainPart' may traverse a certain 'ocp' only once. If it changes its direction this should be defined as a distinct 'trainPart'.

We currently speak about "heavy rails" not about tram systems where turning loops are common. But anyway the 'trainPart' changes running their.

> The issue of referencing an ocpTT is important and has a lot of
> implications. Tthat should be futher discussed and be introduced within a
> version 3.0. I therefore opend a new ticket:
> http://trac.assembla.com/railML/ticket/165

The reference to a certain platform stays to be resolved with the above mentioned ticket. But this is a more general topic: how to refer to a certain platform inside the timetable?

Currently the string-typed attribute "trackInfo" in the <ocpTT> element is used for this, but since railML 2.2 the infrastructure enables defining platformEdges for tracks that may be referred from the timetable.

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Tue, 06 Nov 2012 15:36:40 GMT
View Forum Message <> Reply to Message

Hi,

> Andreas Tanner wrote:
>>> Trainparts would be somewhat more precise than trains, and they have a
>>> validity. Maybe a connection holds only on certain dates.

Ok, I'm convinced.
A connection leads to a certain train. But more precisely to a certain trainPart within a train. If this trainPart is used within several trains, you will automatically have a connection to all of then. And a connection that holds on a certain operatingPeriod only, is expressed by referencing this special tainPart.

Therefore I will add a new "trainPartRef" within the still open ticket:

http://trac.assembla.com/railML/ticket/126

The existing (now redundant) "trainRef" is part of the existing version 2.0 and can't be renamed within 2.2. But we could declare it as deprecated for 3.0 if this is the consensus.

Kind regards,
Joachim

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Thu, 08 Nov 2012 21:05:02 GMT
View Forum Message <> Reply to Message

Dear all,

>> Andreas Tanner wrote:
>>>> Trainparts would be somewhat more precise than trains, and they have a
>>>> validity. Maybe a connection holds only on certain dates.

Joachim Rubröder wrote:
> Ok, I'm convinced.

I also agree. At least the difference may lay in the case two train parts, scheduled to run coupled together, must run separate under operational conditions. If you have a trainpart-relating connection then, it is clear which separate train need to wait for the connection and which may "run away". Having train-related connections, always all train parts would have to wait, whether it was it makes sense or not.

Susanne Wunsch wrote:
> Wouldn't it be sufficient to refer to a certain 'trainPart' and an
> 'ocp'? A 'trainPart' may traverse a certain 'ocp' only once. If it
> changes its direction this should be defined as a distinct 'trainPart'..

Unfortunately this is nowadays wrong. A train(part) may "traverse" an OCP more than once. I could mention plenty examples from practice, not only from Germany. For short, only one example which you can easily find at HAFAS: The CANTUS trains from Bebra to Göttingen and v. v., stopping two times at Niederhone whith the same train number (24090, 24096 ff.). If you like more examples: Don't hesitate to ask... ;-)

In former times, the local signalman and his books were the reasons why it was forbidden that one train number happens more than once a day at one station. There even was a special rule for that in the German rule book

(which by far wasn't able to avoid that it happened even in earlier times,
e. g. some trains from Leipzig to Görlitz, reversing at Dresden Hbf,
stopping two times in Dresden Neustadt).

Nowadays, it is very common in practice throughout many countries. It may
have to do that because of there are less local signalman, less books to
write or not enough train numbers or less knowledge about the rules...
Anyway, we have to handle it in RailML.

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Thu, 08 Nov 2012 21:54:26 GMT
View Forum Message <> Reply to Message

Dear Dirk,

Thanks for the example from the practice. :-)

Dirk Bräuer <dirk.braeuer@irfp.de> writes:
> Susanne Wunsch wrote:
>> Wouldn't it be sufficient to refer to a certain 'trainPart' and an
>> 'ocp'? A 'trainPart' may traverse a certain 'ocp' only once. If it
>> changes its direction this should be defined as a distinct 'trainPart'.
>
> Unfortunately this is nowadays wrong. A train(part) may "traverse" an
> OCP more than once. I could mention plenty examples from practice, not
> only from Germany. For short, only one example which you can easily
> find at HAFAS: The CANTUS trains from Bebra to Göttingen and v. v.,
> stopping two times at Niederhone whith the same train number (24090,
> 24096 ff.).

But this train reverses at Eschwege. That means there should be at least
two train parts in order to define the reversed vehicle order. [1] It's
unfortunately not yet described in the wiki page of trainPart. [2] :-(

Kind regards...
Susanne

[1] http://trac.assembla.com/railML/ticket/142
[2] http://www.wiki.railml.org/index.php?title=TT:trainPart

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Fri, 09 Nov 2012 10:43:36 GMT
View Forum Message <> Reply to Message

Dear all,

> But this train reverses at Eschwege. That means there should be at least
> two train parts in order to define the reversed vehicle order.

Dear Susanne: Please don't lose sight of the forest for the trees... ;-)

- If the train consists of one MU only (most of the trains do so) - what do you want to reverse there? (Please note that there is no possibility to describe the orientation of a single vehicle in a <formation>.)
- It is not necessary to specify a formation at all (<formationTT> is optional). So, for a simple timetable description - may be a passenger information like HAFAS - there is no need to use create two train parts.
- I can also send you an exempli gratia where a train passes a station twice without reversing...

But another question we should ask ourselves is: If we specify a connection with trainPartRef and ocpRef - may it be that the right interpretation follows from the contents?

Train #24090 stops at Niederhone 14.28 (direction to Eschwege) and again 14.38 (direction to Göttingen). A (hypothetical) bus could arrive at Niederhone on 14.25 an referring a connection to #24090.
- Do the min/maxConTime attributes help us to specify the right stop?
- Should we (alternatively) refer to <ocpTT>.sequence (the counter) instead of <ocp>?
- Should we (alternatively) provide optional "directionToOcpRef" and "directionFromOcpRef" attributes to clarify the situation?

With best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Fri, 09 Nov 2012 14:39:17 GMT
View Forum Message <> Reply to Message

Dear Dirk,

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

>> But this train reverses at Eschwege. That means there should be at least
>> two train parts in order to define the reversed vehicle order.

> - If the train consists of one MU only (most of the trains do so) -
> what do you want to reverse there?

It changes its running direction in Eschwege, that means "it reverses".

> (Please note that there is no possibility to describe the orientation
> of a single vehicle in a <formation>.)

In the element vehicle/wagon/driversCab/ you may define the 'orderNumber' of the drivers Cab and its 'position' (rear, middle, front). That could only serve for your mentioned MUs, of course. Otherwise the orientation is given by the order of vehicles in a formation.

> - It is not necessary to specify a formation at all (<formationTT> is
> optional). So, for a simple timetable description - may be a passenger
> information like HAFAS - there is no need to use create two train
> parts.

You don't have to give the formation reference at all, yes. But if you define one train part for both "vehicle orientations" there may be two railML-representations of the same train.

That is something the railML-Infrastructure-world has to deal with every day (short vs. long tracks). ;-)

> - I can also send you an exempli gratia where a train passes a station
> twice without reversing...

Would be interested in one example only.

> But another question we should ask ourselves is: If we specify a
> connection with trainPartRef and ocpRef - may it be that the right
> interpretation follows from the contents?
>
> Train #24090 stops at Niederhone 14.28 (direction to Eschwege) and
> again 14.38 (direction to Göttingen). A (hypothetical) bus could
> arrive at Niederhone on 14.25 an referring a connection to #24090.
> - Do the min/maxConTime attributes help us to specify the right stop?
> - Should we (alternatively) refer to <ocpTT>.sequence (the counter)
> instead of <ocp>?
> - Should we (alternatively) provide optional "directionToOcpRef" and
> "directionFromOcpRef" attributes to clarify the situation?

No questions would arise if we would model this case with two train parts. ;-)

Kind regards...

Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Fri, 09 Nov 2012 15:00:05 GMT
View Forum Message <> Reply to Message

Dear Susanne,

so your suggestion is: If a train passes the same station two times, two train parts have to be used?

Well, ok, I don't think that it will be widely accepted but I have no problem with it. (Actually, I would be very glad if DB Netz and OeBB Infra would accept it because it would make life much more easier with current interfaces.)

Can we then clarify that for RailML, there is the following rule:
--> No ocpRef is allowed to occur more than one time in the same <trainPart>.

I'll write it into Wiki.

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Fri, 09 Nov 2012 15:11:14 GMT
View Forum Message <> Reply to Message

Dear Susanne,

> It changes its running direction in Eschwege, that means "it reverses".

please do not mix the "trainReverse" attribute with the "orientationReversed" attribute.

- "trainReverse" is an attribute of <ocpTT>,
- "orientationReversed" is an attribute of <formationTT>.

If only the train reverses but not the formation, there is currently no need to separate two <trainParts>.

In the example of Eschwege with one MU, only the train reverses but not the formation. So until today: No need for two <trainParts>.

With your suggestion to _enforce_ two train parts if a train reverses (which I would welcome as mentioned in the previous post), the situation changes:

We could now declare "trainReverse" being obsolete since we could always use "orientationReversed" (also for single MUs by definition) because we always will have to have a new <trainPart>.

So: If your suggestion will be accepted (by Joachim and others), I herewith plead for declaring "trainReverse" obsolete to reduce redundancy.

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Fri, 09 Nov 2012 15:17:37 GMT
View Forum Message <> Reply to Message

Dear Dirk and others,

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

> so your suggestion is: If a train passes the same station two times,
> two train parts have to be used?
>
> Well, ok, I don't think that it will be widely accepted but I have no
> problem with it. (Actually, I would be very glad if DB Netz and OeBB
> Infra would accept it because it would make life much more easier
> with current interfaces.)
>
> Can we then clarify that for RailML, there is the following rule:
> --> No ocpRef is allowed to occur more than one time in the same
> <trainPart>.

> I'll write it into Wiki.

I'm sorry, it's only my understanding of this situation. As you wrote in the next posting, it should be accepted by the current railML community and at least by Joachim, as timetable coordinator.

We will speak about this aspect next week at the conference in Zurich.

I would be interested in other examples that match the problem of passing one ocp several times.

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Fri, 09 Nov 2012 15:29:07 GMT
View Forum Message <> Reply to Message

Dirk Bräuer <dirk.braeuer@irfp.de> writes:
>> It changes its running direction in Eschwege, that means "it reverses".
>
> please do not mix the "trainReverse" attribute with the
> "orientationReversed" attribute.
>
> - "trainReverse" is an attribute of <ocpTT>,
> - "orientationReversed" is an attribute of <formationTT>.

I'm sorry, you're right. I missed the attribute 'trainReverse' in the 'ocpTT' element.

> If only the train reverses but not the formation, there is currently
> no need to separate two <trainParts>.

I'm sorry, I don't see the difference between a train and a formation reverse.

Do you mean "Lok umsetzen" with "formation reverse"?

The wiki says about the 'trainReverse' in 'ocpTT' (since 2011-05-10):

trainReverse : is true if train changes direction at this station (xs:boolean, optional). After this station the rollingstock which is referenced in formationTT changes order. This attribute could also been set at the first station, indicating that the formation starts in the reversed order.

It is currently not forseen to split the train part because of the reversing - yes.

> With your suggestion to _enforce_ two train parts if a train reverses

> (which I would welcome as mentioned in the previous post), the
> situation changes:
>
> We could now declare "trainReverse" being obsolete since we could
> always use "orientationReversed" (also for single MUs by definition)
> because we always will have to have a new <trainPart>.

+1

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Fri, 09 Nov 2012 15:55:08 GMT
View Forum Message <> Reply to Message

Dear Susanne,

Am 09.11.2012, 15:39 Uhr, schrieb Susanne Wunsch <coord@common.railml.org>:
>> - I can also send you an exempli gratia where a train passes a station
>> twice without reversing...
>
> Would be interested in one example only.

I did not mention more than one! ;-)

Here it is:

The night train Amsterdam - Warsaw does not reverse at Köln Hbf: It goes
Düsseldorf - (line #2650) - Köln-Mülheim - Köln-Deutz -
(Hohenzollernbrücke) - Köln Hbf - Köln West - Köln Süd - (Südbrücke) -
Köln-Kalk - Köln-Mülheim - (line #2730) - Wuppertal.

So it passes two times through Köln-Mülheim and neighboring OCPs without
reversing between. (Actually, the aim of the action is to avoid reversing.)

Current train no. is EN 446/447. Current times are:
446: KKM 6.08 - 6.14 KK 6.17 - KKM 6.35
447: KKM 22.19 - 22.25 KK 22.28 - KKM 22.46

The interesting thing is: It actually does not go _back_ on its way
back... It always makes the circle anti-clockwise, never clockwise. So
both the trains to Amsterdam and Warsaw leave Köln Hbf in north-westerly

direction.

I'm aware that it has nothing to do with connections. It should only clarify that nowadays a train may pass a station twice, even without reversing. Concerning the connections, we seam to get a different solution.

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Fri, 09 Nov 2012 16:18:14 GMT
View Forum Message <> Reply to Message

Dear Dirk and others,

Dirk Bräuer <dirk.braeuer@irfp.de> writes:
> The night train Amsterdam - Warsaw does not reverse at Köln Hbf: It
> goes Düsseldorf - (line #2650) - Köln-Mülheim - Köln-Deutz -
> (Hohenzollernbrücke) - Köln Hbf - Köln West - Köln Süd - (Südbrücke) -
> Köln-Kalk - Köln-Mülheim - (line #2730) - Wuppertal.
>
> So it passes two times through Köln-Mülheim and neighboring OCPs
> without reversing between. (Actually, the aim of the action is to
> avoid reversing.)

But the train arrives Köln-Mülheim the second time "orientation reversed", doesn't it? Though we nevertheless have to indicate that aspect. :-(

> Current train no. is EN 446/447. Current times are:
> 446: KKM 6.08 - 6.14 KK 6.17 - KKM 6.35
> 447: KKM 22.19 - 22.25 KK 22.28 - KKM 22.46
>
> The interesting thing is: It actually does not go _back_ on its way
> back... It always makes the circle anti-clockwise, never clockwise. So
> both the trains to Amsterdam and Warsaw leave Köln Hbf in
> north-westerly direction.
>
> I'm aware that it has nothing to do with connections. It should only
> clarify that nowadays a train may pass a station twice, even without
> reversing. Concerning the connections, we seam to get a different
> solution.

Thank you for contributing this nice example. It would be interesting to see it in railML without splitting into two train parts. Which problems may occur regarding the timetable in general and rostering in special? I currently would expect none, but there are every time surprises in the

details. ;-)

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Fri, 09 Nov 2012 17:38:08 GMT
View Forum Message <> Reply to Message

> But the train arrives Köln-Mülheim the second time "orientation
> reversed", doesn't it?

No, it does not. The fist and the carriages are still the same. The orientation of a formation currently refers to its running direction but not to a geographic direction nor a mileage direction.

> It would be interesting to
> see it in railML without splitting into two train parts. Which problems
> may occur regarding the timetable in general and rostering in special?

So far: None. There already are these cases in practice including circulation plans w/o problems. There is no reference to an <ocpTT> in RailML so far so that's why two <ocpTT> may refer to the same <ocp> w/o problems.

Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Fri, 09 Nov 2012 18:38:54 GMT
View Forum Message <> Reply to Message

> Do you mean "Lok umsetzen" with "formation reverse"?

Of course not. Running ‘round with the engine does not reverse the formation, it changes the formation.

Concerning the meaning of <ocpTT>.trainReverse:

It simply tells that the train(part) changes the direction - no matter

whether the formation changes, reverses or neither of both.

trainReverse with change of formation = e. g. running around with the engine
trainReverse with reversing of formation = train(part) of several MUs or push-pull train
trainReverse with neither of them = train(part) consists of a single vehicle (MU or engine)

This information is mainly intended for passenger information (systems) which sometimes print a sign like <-> to notify the passenger where the running direction of the train changes.

Concerning the meaning of <formationTT>.orientationReversed:

It has nothing to do with a train changing the running direction. It simply shall avoid the necessity to create each formation two times for both orientations. A train does not need to change its running direction for <formationTT>.orientationReversed:

Let’s say a train with the formation
1. propelling control car
2. 2nd class carriage
3. 1st class carriage
4. engine

runs all day between Airport and a place called Pirna. For the one direction the <formation> is fine, but for the other direction - so for half of the trains in total - the formation would have to be created a second time:
1. engine
2. 1st class carriage
3. 2nd class carriage
4. propelling control car

To avoid this, the attribute <formationTT>.orientationReversed can be used at every second train.

Please note: None of the trains do ever change its running direction during a single run - as in practice between Airport and Pirna.

Until now, there was no need to use <formationTT>.orientationReversed at a formation consisting of one vehicle only. This would have been paradox since one cannot change the order of a list of one element.

> I'm sorry, I don't see the difference between a train and a formation

> reverse.

Additionally, one should be aware that <ocpTT>.trainReverse semantically applies to the whole train while <formationTT>.orientationReversed applies to the formation of one <trainPart> only. So, there is another way to change the orientation of the formation of a whole train w/o <formationTT>.orientationReversed: Each vehicle forms its own <trainPart>, may be due to different operating days or so.

```
<train>
  <trainPartSequence>
    <trainPartRef ref='TP1.1' position='1'>
    <trainPartRef ref='TP2.1' position='2'>
    <trainPartRef ref='TP3.1' position='3'>
  </trainPartSequence>
  <trainPartSequence>
    <trainPartRef ref='TP3.2' position='1'>
    <trainPartRef ref='TP2.2' position='2'>
    <trainPartRef ref='TP1.2' position='3'>
  </trainPartSequence>
</train>
```

Obviously the train reverses between both <trainPartSequences> but there would be no <formationTT>.orientationReversed at none of the <trainParts> if each consist of one MU only.

(This refers to the current situation in RailML. It changes if we declare <ocpTT>.trainReverse obsolete and declare <formationTT>.orientationReversed to be used by definition as recommended in the previous post.)

Summary:
1) A formation running w/o reversing in one direction ("forward"):
    <ocpTT>.trainReverse: not used
    <formationTT>.orientationReversed: not used

2) A formation of several vehicles running w/o reversing in the other direction ("backward"):
    <ocpTT>.trainReverse: not used
    <formationTT>.orientationReversed: shall be used

3) A formation of several vehicles reverses direction w/o 'running around':
    <ocpTT>.trainReverse: shall be used
    <formationTT>.orientationReversed: shall be used

4) A formation of several vehicles reverses direction with 'running around' of the engine:

<ocpTT>.trainReverse: shall be used
<formationTT>.orientationReversed: cannot be used since the formation changes

To avoid no. #4, the engine may be put in an own <trainPart> so that #4 becomes “two times #3”. This reduces the total number of necessary formations by trend.

Hope I was able to clarify the difference between trainReverse and orientationReversed.

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Mon, 12 Nov 2012 21:46:23 GMT
View Forum Message <> Reply to Message

Susanne Wunsch <coord@common.railml.org> writes:

> Dirk Bräuer <dirk.braeuer@irfp.de> writes:
>
>> so your suggestion is: If a train passes the same station two times,
>> two train parts have to be used?
>>
>> Well, ok, I don't think that it will be widely accepted but I have no
>> problem with it. (Actually, I would be very glad if DB Netz and OeBB
>> Infra would accept it because it would make life much more easier
>> with current interfaces.)
>>
>> Can we then clarify that for RailML, there is the following rule:
>> --> No ocpRef is allowed to occur more than one time in the same
>> <trainPart>.
>
>> I'll write it into Wiki.
>
> I'm sorry, it's only my understanding of this situation. As you wrote in the
> next posting, it should be accepted by the current railML community and
> at least by Joachim, as timetable coordinator.
>
> We will speak about this aspect next week at the conference in Zurich.

As mentioned I filed a Trac ticket for this issue:

https://trac.assembla.com/railML/ticket/197

Kind regards...

Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Mon, 12 Nov 2012 22:04:28 GMT
View Forum Message <> Reply to Message

Dear Dirk,

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

>> Do you mean "Lok umsetzen" with "formation reverse"?
>
> Of course not. Running ‘round with the engine does not reverse the
> formation, it changes the formation.
>
> ---
> Concerning the meaning of <ocpTT>.trainReverse:
> ---

[...]

Clarified. Thank you.

> ---
> Concerning the meaning of <formationTT>.orientationReversed:
> ---

[...]

Clarified. Thank you.

> ---
>> I'm sorry, I don't see the difference between a train and a
>> formation reverse.

[...]

Thanks for this clarification, too.

> (This refers to the current situation in RailML. It changes if we
> declare <ocpTT>.trainReverse obsolete and declare
> <formationTT>.orientationReversed to be used by definition as
> recommended in the previous post.)

I'm sorry this idea is based on my misunderstanding of the current situation. So what would be the pros and cons of changing the current situation?

PROs:
* Unambiguous connection definitions
* "Redundancy reduction": only one place in the schema where to define reversing of train parts and/or whole trains

CONs:
* Changing current implementations with no strong need

> Hope I was able to clarify the difference between trainReverse and
> orientationReversed.

I think you got it!

Thanks a lot.

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Tue, 13 Nov 2012 11:50:55 GMT
View Forum Message <> Reply to Message

Dear Susanne, thank you for clarifying what is clarified... ;-) (Honestly: It makes it easier to follow the discussion for someone outside.)

> So what would be the pros and cons of changing the current
> situation?

To focus this: You are writing about the following possible change:

>> Can we then clarify that for RailML, there is the following rule:
>> --> No ocpRef is allowed to occur more than one time in the same
>> <trainPart>.
>> We could now declare "trainReverse" being obsolete since we could
>> always use "orientationReversed" (also for single MUs by definition)
>> because we always will have to have a new <trainPart>.

This would simply be a change of rule and declaring 'orientationReversed' obsolete but no syntactic change.

> PROs:
> * Unambiguous connection definitions
> * "Redundancy reduction": only one place in the schema where to define
> reversing of train parts and/or whole trains

I agree these PROs are correct.

> CONs:
> * Changing current implementations with no strong need

I'm afraid I have to add one CON: The current 'trainReverse' attribute fits to the very common symbol <-> for reversing direction in timetables. I guess many public information systems have to handle this information. These public information systems do normally not handle train parts - a train parts is an object not very common in public information. So by deleting the 'trainReverse' attribute, we actually would reduce redundancy, but also we would make it much more difficult to recreate the 'trainReverse' information: One would have to test if all formations of all train parts have the 'orientationReversed' attribute negated and the order of the train parts (attribute 'position') has turned around...

Well, now it is 2:2 on the PROs and CONs.

If I see the many redundancies we create right now in <infrastructure>, it seams reasonable to keep this small redundancy here for the moment to ease reading of the files for programmes.

Anyway, as written earlier I would agree to and welcome that change.

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Wed, 14 Nov 2012 23:43:10 GMT
View Forum Message <> Reply to Message

Dear all,

Dirk BrÃ¤uer wrote:
> Dear Susanne, thank you for clarifying what is clarified... ;-) (Honestly:
> It makes it easier to follow the discussion for someone outside.)

Many thanks for both of you for this long thread and all the clarification which will indeed be very helpful for every outsider.

I appreciate to force the splitting in trainParts whenever the formation changes, also for orientation changes.

>>> Can we then clarify that for RailML, there is the following rule:
>>> --> No ocpRef is allowed to occur more than one time in the same
>>> <trainPart>.

A forced splitting of trainParts whenever an ocpTT would occur several times would be consequent. This would also solve the problem of referencing the correct ocpTT within a trainPart
(http://www.railml.org/forum/ro/?group=2&offset=0&thr ead=72&id=247).

>>> We could now declare "trainReverse" being obsolete since we could
>>> always use "orientationReversed" (also for single MUs by definition)
>>> because we always will have to have a new <trainPart>.
> I'm afraid I have to add one CON: The current 'trainReverse' attribute
> fits to the very common symbol <-> for reversing direction in timetables.
> I guess many public information systems have to handle this information.

I would like to keep the 'trainReverse' attribute, for this purpose which was also mentioned by T. Kauer (SBB) at the railML meeting. With the forced splitting of trainParts, the 'trainReverse' would mainly occur at the first ocpTT of a trainPart if you have any formations referenced. It should therefore no longer be seen as automatically reversing the formation. For a simple timetable information system (without dealing with formations) it could still be used within a long trainPart to indicate the symbol <->.

Best regards,
Joachim

--
Joachim RubrÃ¶der
Schema Coordinator: railML.timetable

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Thu, 15 Nov 2012 16:37:45 GMT
View Forum Message <> Reply to Message

Dear Joachim and all others,

thank you for your reply.

So it seams that our conclusion is to keep the 'trainReverse' attribute in spite of redundancy for easier reading and better understanding of the files. I agree.

There is only one question left:

>>>> --> No ocpRef is allowed to occur more than one time in the same
>>>> <trainPart>.

> A forced splitting of trainParts whenever an ocpTT would occur several
> times would be consequent. This would also solve the problem...

> For a simple timetable information system (without dealing with
> formations) it could still be used within a long trainPart to indicate
> the symbol <->.

Please specify whether

a) Reversing trains _must_ be splitted into several <trainParts>; the attribute 'trainReverse' is only allowed at fist <ocpTT>s.

b) Reversing trains _must_ be splitted into several <trainParts> if 'formationRef' is used; in this case, the attribute 'trainReverse' is only allowed at fist <ocpTT>s. If 'formationRef' is not used, 'trainReverse' may occur at each <ocpTT>.

c) Reversing trains need not to be splitted into several <trainParts>; the attribute 'trainReverse' may occur at each <ocpTT>. Both possibilities to express a change of direction (with 'trainReverse' or 'formationReversed') are allowed and shall be understood equally.

Please decide from one of these three options. I would like to fix this in the Wiki and in our software as soon as possible because we already have changes of direction without splitted train parts. If this will become invalid, I do not want to keep it in 2.2.

I would write it into Wiki after your decision.

Thank you and best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Susanne Wunsch railML on Thu, 15 Nov 2012 17:52:42 GMT
View Forum Message <> Reply to Message

coord@timetable.railml.org (Joachim Rubröder) writes:
> Dirk Bräuer wrote:

>>>> Can we then clarify that for RailML, there is the following rule:
>>>> --> No ocpRef is allowed to occur more than one time in the same
>>>> <trainPart>.
>
> A forced splitting of trainParts whenever an ocpTT would occur several
> times would be consequent. This would also solve the problem of
> referencing the correct ocpTT within a trainPart
> (http://www.railml.org/forum/ro/?group=2&offset=0&thr ead=72&id=247).

I would also prefer this way of modeling instead of references to single 'ocptTT's from within a 'trainPart'.

>>>> We could now declare "trainReverse" being obsolete since we could
>>>> always use "orientationReversed" (also for single MUs by definition)
>>>> because we always will have to have a new <trainPart>.
>> I'm afraid I have to add one CON: The current 'trainReverse' attribute
>> fits to the very common symbol <-> for reversing direction in timetables.
>> I guess many public information systems have to handle this information.
>
> I would like to keep the 'trainReverse' attribute, for this purpose which
> was also mentioned by T. Kauer (SBB) at the railML meeting. With the
> forced splitting of trainParts, the 'trainReverse' would mainly occur at
> the first ocpTT of a trainPart if you have any formations referenced.

It is some kind of redundancy, but it's a bit tricky to deduce it:

1. Find the commercial train, where this train part is used.
2. Look at the train parts at the previous trainPartSequence.
3. Look if the same formationTT is referred.

-> 'trainReverse' is true.

But if the formationTT refers some kind of general formation this deduction may be false.

+1 for keeping "trainReverse"

Instead of allowing the 'trainReverse' attribute only in the first 'ocpTT' we may include it in the 'formationTT' element as this may only occur once per 'trainPart'. This may be ensured by the XSD, but the occurence of the attribute in the first 'ocpTT' element may only be ensured by Schematron, not XSD.

That would mean, that both attribute 'trainReverse' and 'orientationReversed' will be in the same element, but with some kind of different meaning. I like to explicitly point to it, instead of "hiding" it in different elements:

* 'trainReverse' important for passenger information systems "<->"

* 'orientationReversed" referring to the definition of the formation in
  the rollingstock subschema

There may be a trainPart with 'trainReverse=true' but with
'orientationReversed=false' because of an already reversed formation in
the previous "train part sequence".

> It should therefore no longer be seen as automatically reversing the
> formation. For a simple timetable information system (without dealing
> with formations) it could still be used within a long trainPart to
> indicate the symbol <->.

I would be happy if the railML semantics would be covered by all
systems. That would mean, that already today a timetabling information
system has to split train parts if the formation changes, nevertheless
it does not know the formation type at all.

Kind regards...
Susanne

--
Susanne Wunsch
railML Common-Coordinator

Subject: Re: RFE for connection, DE:Anschluss
Posted by Daniel Prusseit on Fri, 23 Nov 2012 15:08:31 GMT
View Forum Message <> Reply to Message

Am 15.11.2012 17:37, schrieb Dirk Bräuer:

Dear Dirk Bräuer and the others,

in our case we need the trainReverse attribute to provide correct
destination information for the FIS (passenger information system) in
the several coaches / units of a train. Especially if there are 2 or
more trainparts running on the same day, the correct order is essential.

So I would suggest:

> b) Reversing trains _must_ be splitted into several <trainParts> if
> 'formationRef' is used; in this case, the attribute 'trainReverse' is
> only allowed at fist <ocpTT>s. If 'formationRef' is not used,
> 'trainReverse' may occur at each <ocpTT>.

In all of our cases we get the formationRefs, so it we would expect the
trainReverse at the beginning of a trainpart where a formation is used.

Additionally for other "more simple" use cases a trainReverse at each
ocpTT would provide flexibility.

> Thank you and best regards,
> Dirk.

Thanks in advance for the changes.
With kind regards,
Daniel Prusseit

Subject: Re: RFE for connection, DE:Anschluss
Posted by on Mon, 26 Nov 2012 12:09:04 GMT
View Forum Message <> Reply to Message

Dear Daniel Prusseit,

thank your for your statement. Of course no objections from my side
against solution b.

So we still wait for the decision of the scheme coordinator... ;-)

Best regards,
Dirk.

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Tue, 27 Nov 2012 15:48:17 GMT
View Forum Message <> Reply to Message

Hello all,

> So we still wait for the decision of the scheme coordinator... ;-)
If you both agree, there are no objections from my side.

Susanne mentioned, that the railML semantics should be covered by all
systems.:
> That would mean, that already today a timetabling information
> system has to split train parts if the formation changes, nevertheless
> it does not know the formation type at all.

This would mean to use case a) as a restriction of case b).

> a) Reversing trains _must_ be splitted into several <trainParts>; the
> attribute 'trainReverse' is only allowed at fist <ocpTT>s.

If the attribute 'trainReverse' occurs, it occours at the first <ocpTT>
indicating the symbol <-> for FIS purposes. This can't be checked within
the xsd.

The order of the vehicles could be reverted by 'orientationReversed' but
with no relation to the attribute 'trainReversed'.

Best regards,
Joachim

--
Joachim RubrÃ¶der
Schema Coordinator: railML.timetable

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Sat, 08 Dec 2012 21:03:15 GMT
View Forum Message <> Reply to Message

Hi,

>> * For the "description" element I suggest the name "messageText" or
>> "infoText" as this is more specific.

> I like the name "messageText". This is appropriate for printouts and also
> for the AVL system, indicating an instruction for the driver.

As requested by Peter Brandt at the railML conference in Zurich, the
'messageText' in the TT-element 'connection' should be internationalized.
https://trac.assembla.com/railML/ticket/211

This is done alike the internationalized ocp names:
https://trac.assembla.com/railML/changeset/505

Kind regards,
Joachim

--
Joachim RubrÃ¶der
Schema Coordinator: railML.timetable

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Mon, 02 Dec 2013 11:18:43 GMT
View Forum Message <> Reply to Message

Hi,
the issue was implemented with ticked:
https://trac.assembla.com/railML/ticket/197

Kind regards,
Joachim

Joachim RubrÃ¶der
Schema Coordinator: railML.timetable

Susanne Wunsch wrote:
>
> coord@timetable.railml.org (Joachim RubrÃ¶der) writes:
>> Dirk BrÃ¤uer wrote:
>>>> > Can we then clarify that for RailML, there is the following rule:
>>>> > --> No ocpRef is allowed to occur more than one time in the same
>>>> > <trainPart>.
>>
>> A forced splitting of trainParts whenever an ocpTT would occur several
>> times would be consequent. This would also solve the problem of
>> referencing the correct ocpTT within a trainPart
>> (http://www.railml.org/forum/ro/?group=2&offset=0&thr ead=72&id=247).
>
> I would also prefer this way of modeling instead of references to single
> 'ocptTT's from within a 'trainPart'.
>
>>>> > We could now declare "trainReverse" being obsolete since we could
>>>> > always use "orientationReversed" (also for single MUs by definition)
>>>> > because we always will have to have a new <trainPart>.
>>> I'm afraid I have to add one CON: The current 'trainReverse' attribute
>>> fits to the very common symbol <-> for reversing direction in timetables.

>>> I guess many public information systems have to handle this information.
>>
>> I would like to keep the 'trainReverse' attribute, for this purpose which
>> was also mentioned by T. Kauer (SBB) at the railML meeting. With the
>> forced splitting of trainParts, the 'trainReverse' would mainly occur at
>> the first ocpTT of a trainPart if you have any formations referenced.
>
> It is some kind of redundancy, but it's a bit tricky to deduce it:
>
> 1. Find the commercial train, where this train part is used.
> 2. Look at the train parts at the previous trainPartSequence.

> 3. Look if the same formationTT is referred.
> -> 'trainReverse' is true.
>
> But if the formationTT refers some kind of general formation this
> deduction may be false.
>
> +1 for keeping "trainReverse"
>
> Instead of allowing the 'trainReverse' attribute only in the first
> 'ocpTT' we may include it in the 'formationTT' element as this may only
> occur once per 'trainPart'. This may be ensured by the XSD, but the
> occurence of the attribute in the first 'ocpTT' element may only be
> ensured by Schematron, not XSD.
>
> That would mean, that both attribute 'trainReverse' and
> 'orientationReversed' will be in the same element, but with some kind of
> different meaning. I like to explicitly point to it, instead of "hiding"
> it in different elements:
>
> * 'trainReverse' important for passenger information systems "<->"
>
> * 'orientationReversed" referring to the definition of the formation in
> the rollingstock subschema
>
> There may be a trainPart with 'trainReverse=true' but with
> 'orientationReversed=false' because of an already reversed formation in
> the previous "train part sequence".
>
>> It should therefore no longer be seen as automatically reversing the
>> formation. For a simple timetable information system (without dealing
>> with formations) it could still be used within a long trainPart to
>> indicate the symbol <->.
>
> I would be happy if the railML semantics would be covered by all
> systems. That would mean, that already today a timetabling information
> system has to split train parts if the formation changes, nevertheless
> it does not know the formation type at all.
>
> Kind regards...
> Susanne
>

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

Subject: Re: RFE for connection, DE:Anschluss
Posted by Joachim Rubröder railML on Mon, 02 Dec 2013 11:26:56 GMT
View Forum Message <> Reply to Message

Hi,
in order to solve this issue we will prevent (by documentation) to have
two times the same ocpTT within one trainPart.

https://trac.assembla.com/railML/ticket/235

Kind regards,
Joachim

Joachim RubrÃ¶der
Schema Coordinator: railML.timetable

Dirk BrÃ¤uer wrote:
>
>> But the train arrives KÃ¶ln-MÃ¼lheim the second time "orientation
>> reversed", doesn't it?
>
> No, it does not. The fist and the carriages are still the same. The
> orientation of a formation currently refers to its running direction but
> not to a geographic direction nor a mileage direction.
>
>> It would be interesting to
>> see it in railML without splitting into two train parts. Which problems
>> may occur regarding the timetable in general and rostering in special?
>
> So far: None. There already are these cases in practice including
> circulation plans w/o problems. There is no reference to an <ocpTT> in
> RailML so far so that's why two <ocpTT> may refer to the same <ocp> w/o
> problems.
>
> Dirk.
>
>

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