
Subject: missing bitMask at <trainPart><operatingPeriodRef>

Posted by _____ on Thu, 17 May 2012 11:32:25 GMT

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

Dear Joachim and all others,

there is one small issue which we should fix with RailML 2.2:

A <trainPart> references its operating days with <operatingPeriodRef>. Normally one should expect that there is a 'ref' to an operatingPeriod only and nothing more.

However, there are some more elements there for reasons which I do not know. They are repeated from 'operatingPeriod' and therefore tend to be redundant.

1) There are 'startDate' and 'endDate' which allow to reduce the given operatingPeriod. I suppose this is to reduce the number of operatingPeriods. It is easy to understand how it works and so I think we should keep that possibility in spite of its redundancy. But: There is currently no 'bitMask' for such a reduced operatingPeriod. Since the 'bitMask' becomes more and more the most important attribute of operating days we should provide it here also.

--> I herewith plead for an optional 'bitMask' attribute at <operatingPeriodRef> with the annotation: "to be used together with startDate and endDate".

2) More confusing, there is a sequence <specialService> at <operatingPeriodRef>. It seems that one can alter the referred 'operatingPeriod' using special days!

- To define an <operatingPeriod> and later alter it at <operatingPeriodRef> is very much confusing. It would be better to define one more <operatingPeriod> and not to alter them. The size of the file has never been a question with RailML.

- If we allow altering of operatingPeriods, why with <specialService> and not with <operatingDay>?

- The altered operatingPeriod would again have no bitMask.

From my opinion, we should clear that situation as soon as possible. We have two possibilities:

a) Simple to delete the sequence <specialService> from <operatingPeriodRef>.

b) To allow the definition of operating days without an <operatingPeriod>. This would mean

- to copy the sequence <operatingDay> into <operatingPeriodRef>,
- to add some attributes including 'bitMask',

- to declare the attribute 'ref' as optional,

--> I would plead for (a) for reasons of simplicity and less redundancy.

Best regards,
Dirk.

Subject: Re: missing bitMask at <trainPart><operatingPeriodRef>
Posted by [Andreas Tanner](#) on Mon, 21 May 2012 07:23:59 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

as a general rule, I see absolutely no point in providing alternative means of expressing one and the same thing. It only drives up the cost of implementing the standard.

I would therefore pledge for removing these attributes from the trainPart element and strictly restrain the standard to use of operatingPeriods.

Best regards, Andreas.

Am 17.05.2012 13:32, schrieb Dirk Bräuer:

> Dear Joachim and all others,

>

> there is one small issue which we should fix with RailML 2.2:

>

> A <trainPart> references its operating days with <operatingPeriodRef>.

> Normally one should expect that there is a 'ref' to an operatingPeriod

> only and nothing more.

>

> However, there are some more elements there for reasons which I do not

> know. They are repeated from 'operatingPeriod' and therefore tend to be

> redundant.

>

> 1) There are 'startDate' and 'endDate' which allow to reduce the given

> operatingPeriod. I suppose this is to reduce the number of

> operatingPeriods. It is easy to understand how it works and so I think

> we should keep that possibility in spite of its redundancy. But: There

> is currently no 'bitMask' for such a reduced operatingPeriod. Since the

> 'bitMask' becomes more and more the most important attribute of

> operating days we should provide it here also.

>

> --> I herewith plead for an optional 'bitMask' attribute at

> <operatingPeriodRef> with the annotation: "to be used together with

> startDate and endDate".

>

> ---
> 2) More confusing, there is a sequence <specialService> at
> <operatingPeriodRef>. It seems that one can _alter_ the referred
> 'operatingPeriod' using special days!
> - To define an <operatingPeriod> and later alter it at
> <operatingPeriodRef> is very much confusing. It would be better to
> define one more <operatingPeriod> and not to alter them. The size of the
> file has never been a question with RailML.
> - If we allow altering of operatingPeriods, why with <specialService>
> and not with <operatingDay>?
> - The altered operatingPeriod would again have no bitMask.
>
> From my opinion, we should clear that situation as soon as possible. We
> have two possibilities:
> a) Simple to delete the sequence <specialService> from
> <operatingPeriodRef>.
> b) To allow the definition of operating days without an
> <operatingPeriod>. This would mean
> - to copy the sequence <operatingDay> into <operatingPeriodRef>,
> - to add some attributes including 'bitMask',
> - to declare the attribute 'ref' as optional,
>
> --> I would plead for (a) for reasons of simplicity and less redundancy.
>
> Best regards,
> Dirk.

Subject: Re: missing bitMask at <trainPart><operatingPeriodRef>
Posted by _____ on Thu, 31 May 2012 12:48:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

> as a general rule, I see absolutely no point in providing alternative
> means of expressing one and the same thing. It only drives up the cost
> of implementing the standard.
> I would therefore pledge for removing these attributes from the
> trainPart element and strictly restrain the standard to use of
> operatingPeriods.

It is not so easy with redundancies. Of course we should avoid them as
much as possible. But sometimes we can hardly avoid them without risking
that RailML will not be accepted in practice.

In this special case I agree with Andreas; from our side no objection
against fully deleting (declaring deprecated) everything but the
<operatingPeriodRef>.ref.

Dirk.

Subject: Re: missing bitMask at <trainPart><operatingPeriodRef>
Posted by [Joachim Rubröder railML](#) on Mon, 17 Sep 2012 14:11:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

if you both agree, that the use of operatingPeriods should be restricted to references, we could implement it this way:
<https://trac.assembla.com/railML/ticket/158>

Kind regards...
Joachim

--
Joachim Rubröder
Schema Coordinator: railML.timetable

Dirk Bräuer wrote:

>
> Dear all,
>
>> as a general rule, I see absolutely no point in providing alternative
>> means of expressing one and the same thing. It only drives up the cost
>> of implementing the standard.
>> I would therefore pledge for removing these attributes from the
>> trainPart element and strictly restrain the standard to use of
>> operatingPeriods.
>
> It is not so easy with redundancies. Of course we should avoid them as
> much as possible. But sometimes we can hardly avoid them without risking
> that RailML will not be accepted in practice.
>
> In this special case I agree with Andreas; from our side no objection
> against fully deleting (declaring deprecated) everything but the
> <operatingPeriodRef>.ref.
>
> Dirk.
>
>

--

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

Subject: Re: missing bitMask at <trainPart><operatingPeriodRef>

Posted by on Tue, 02 Oct 2012 15:37:38 GMT

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

from our side no objection against fully deleting (declaring deprecated)
everything but the <operatingPeriodRef>.ref.

Subject: Re: missing bitMask at <trainPart><operatingPeriodRef>

Posted by on Fri, 19 Oct 2012 17:11:02 GMT

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

Dear Joachim,

now that <operatingPeriodRef> has "ref" as its only attribute: Do you want
to create a ticket for 3.0 to change <operatingPeriodRef> into a simple
attribute of <trainPart> just like "timetablePeriodRef"?

Dirk.

Subject: Re: missing bitMask at <trainPart><operatingPeriodRef>

Posted by [Joachim Rubröder railML](#) on Wed, 31 Oct 2012 18:16:19 GMT

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

Makes perfectly sense, unless someone want new attributes for
operatingPeriodRef.

<http://trac.assembla.com/railML/ticket/176>

Kind regards,
Joachim

--

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