
Subject: How to represent open circulations in railML?

Posted by

on Fri, 17 Nov 2017 15:20:15 GMT

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

Dear railML Timetable Community!

When defining an rostering with an open (non-cyclic) circulation of blocks, there are two different possibilities to model them in railML.

Either with an final open circulation Element at the end:

```
<circulations>
  <circulation blockRef="b00380" operatingPeriodRef="op0103" nextBlockRef="b00381"
nextOperatingPeriodRef="op0104"/>
  <circulation blockRef="b00381" operatingPeriodRef="op0104" nextBlockRef="b00382"
nextOperatingPeriodRef="op0105"/>
  <circulation blockRef="b00382" operatingPeriodRef="op0105" nextBlockRef="b00383"
nextOperatingPeriodRef="op0106"/>
  <circulation blockRef="b00383" operatingPeriodRef="op0106" nextBlockRef="b00384"
nextOperatingPeriodRef="op0107"/>
  <circulation blockRef="b00384" operatingPeriodRef="op0107"/>
</circulations>
```

So you can see in the example there is a final circulation with no nextBlockRef and nextOperatingPeriodRef attributes. But in my point of view this last circulation element is redundant and may be left out - like the following example depicts:

```
<circulations>
  <circulation blockRef="b00380" operatingPeriodRef="op0103" nextBlockRef="b00381"
nextOperatingPeriodRef="op0104"/>
  <circulation blockRef="b00381" operatingPeriodRef="op0104" nextBlockRef="b00382"
nextOperatingPeriodRef="op0105"/>
  <circulation blockRef="b00382" operatingPeriodRef="op0105" nextBlockRef="b00383"
nextOperatingPeriodRef="op0106"/>
  <circulation blockRef="b00383" operatingPeriodRef="op0106" nextBlockRef="b00384"
nextOperatingPeriodRef="op0107"/>
</circulations>
```

As well I have read the railML-Wiki entry on circulations

(<http://wiki.railml.org/index.php?title=TT:circulation>). There I found following section on this question:

"There exists a <circulation> element for every block on every operational day. Via the attributes nextBlockRef and nextOperatingperiodRef the blocks are connected to a chain and form a rostering. ..."

I am not quit sure how to interpret this phrase. Is it sufficient to refer the final block and operatingPeriod in an nextBlockRef and nextOperatingPeriodRef attribute, or do I have to add an additional circulation element at the end with a blockRef and operatingPeriodRef pointing to the last block and operatingPeriod?

Best regards,
Leopold Kühschelm

Subject: Re: How to represent open circulations in railML?
Posted by _____ on Thu, 23 Nov 2017 11:55:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Leopold,

> Is it sufficient to refer the final block and operatingPeriod in an nextBlockRef and nextOperatingPeriodRef attribute, or do I have to add an additional circulation element at the end with a blockRef and operatingPeriodRef pointing to the last block and operatingPeriod?

Currently, as far as I know, there is only one usage of railML rostering for open circulations: The last block has a (redundant) <circulation> element without /nextBlockRef/ and without /nextOperatingPeriodRef/ attributes.

> But in my point of view this last circulation element is redundant and may be left out...

Yes, I agree, but this is apparently a bit too "indirect", implicit. However, as far as I am concerned, it was not the intention when the current structures were designed.

A closed circulation was regarded as the "normal" case - hence the word "circulation". The "open circulation" (which may be regarded as no circulation at all) was seen as a special case which is derived from the normal case. In a closed circulation, every block needs a <circulation> element. Therefore, to ease usage of structures and uniqueness, also in an "open circulation" every block should have a <circulation> element.

One could discuss whether a closed circulation is really the "normal" case; actually there are arguments against it. The question may be whether the alternative usage you describe is shall be valid railML or not. Currently, for the sake of compatibility, I would vote for "not valid", in spite of the obvious redundancy.

With best regards,
Dirk.

Am 17.11.2017 um 16:20 schrieb Leopold Kühschelm:

> Dear railML Timetable Community!

>

> When defining an rostering with an open (non-cyclic) circulation of blocks, there are two different possibilities to model them in railML.

>

> Either with an final open circulation Element at the end:

>

```
> <circulations>
>   <circulation blockRef="b00380" operatingPeriodRef="op0103"
nextBlockRef="b00381" nextOperatingPeriodRef="op0104"/>
>   <circulation blockRef="b00381" operatingPeriodRef="op0104"
nextBlockRef="b00382" nextOperatingPeriodRef="op0105"/>
>   <circulation blockRef="b00382" operatingPeriodRef="op0105"
nextBlockRef="b00383" nextOperatingPeriodRef="op0106"/>
>   <circulation blockRef="b00383" operatingPeriodRef="op0106"
nextBlockRef="b00384" nextOperatingPeriodRef="op0107"/>
>   <circulation blockRef="b00384" operatingPeriodRef="op0107"/>
> </circulations>
```

> So you can see in the example there is a final circulation with no nextBlockRef and nextOperatingPeriodRef attributes. But in my point of view this last circulation element is redundant and may be left out - like the following example depicts:

```
>
> <circulations>
>   <circulation blockRef="b00380" operatingPeriodRef="op0103"
nextBlockRef="b00381" nextOperatingPeriodRef="op0104"/>
>   <circulation blockRef="b00381" operatingPeriodRef="op0104"
nextBlockRef="b00382" nextOperatingPeriodRef="op0105"/>
>   <circulation blockRef="b00382" operatingPeriodRef="op0105"
nextBlockRef="b00383" nextOperatingPeriodRef="op0106"/>
>   <circulation blockRef="b00383" operatingPeriodRef="op0106"
nextBlockRef="b00384" nextOperatingPeriodRef="op0107"/>
> </circulations>
```

> As well I have read the railML-Wiki entry on circulations (<http://wiki.railml.org/index.php?title=TT:circulation>). There I found following section on this question:

```
>
> "There exists a <circulation> element for every block on every operational day. Via the
attributes nextBlockRef and nextOperatingperiodRef the blocks are connected to a chain and form
a rostering.
```

```
> ..."
```

```
>
> I am not quite sure how to interpret this phrase. Is it sufficient to refer the final block and
operatingPeriod in an nextBlockRef and nextOperatingPeriodRef attribute, or do I have to add an
additional circulation element at the end with a blockRef and operatingPeriodRef pointing to the
last block and operatingPeriod?
```

```
>
> Best regards,
> Leopold KÜhschelm
>
```

Subject: Re: How to represent open circulations in railML?
Posted by [Joachim.Rubröder](#) on Mon, 18 Dec 2017 07:02:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Leopold

Between these two alternatives I agree with Dirk and would vote for the first one because it is more explicit.
In ever such cases I would prefer a clear and easy understandable structure in spite of a possible redundancy.

With best regards,
Joachim Rubröder

Subject: Re: How to represent open circulations in railML?
Posted by [Andreas Tanner](#) on Mon, 26 Feb 2018 11:22:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello everybody,

agreed. Here
<https://wiki.railml.org/index.php?title=TT:circulation&oldid=7451&diff=cur>
is a proposal to clarify the documentation for the circulation element.
Is this acceptable?

Best regards, Andreas.

Am 18.12.2017 um 08:02 schrieb Joachim Rubröder:

> Dear Leopold
>
> Between these two alternatives I agree with Dirk and would
> vote for the first one because it is more explicit.
> In ever such cases I would prefer a clear and easy
> understandable structure in spite of a possible redundancy.
>
> With best regards,
> Joachim Rubröder
>

Subject: Re: How to represent open circulations in railML?
Posted by [Joachim Rubröder](#) on Tue, 13 Mar 2018 09:36:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Andreas,

I "internally" agree with all you have written. Means: I see no discrepancy in understanding.

I am not sure whether the writing is clear enough (and not misleading) for anybody who is new in railML circulation.

a) I would consider linking the "missing" /nextBlockRef/ attribute to the concept of "open" circulation plans, as the contrary of "closed" circulation plans which never have missing /nextBlockRef/s.

b) I think the sentence

"The presence of a circulation element that references this block via blockRef, in this case, merely expresses that fact that the block is considered as belonging to the roster."

is a bit of an "understatement" because such a <circulation> element does not only express that the block belongs to the roster. It can also express at which day it is formed by which (nominal) vehicle - by its attributes /operatingPeriodRef/ and /vehicleCounter/.

You write that such a circulation models a block that has no pre-/successor in the _linked_ chain (of this roster). This is formally true. But, in reality, it has of course a pre- and a successor, in the previous and following circulation plans. So actually the attribute /vehicleCounter/ is currently (with the current railML schemes) the only chance to find out _which_ vehicle exactly form this block (and unfortunately only an indirect kind). That's why I think such open <circulation> elements are still very important and not only "merely expressions".

I can write a suggestion for (a) later in case you do not want to do it now.

I want to ask you to extend or change your sentence for (b) if you agree with my argumentation.

With best regards,
Dirk.
