
Subject: Sequence of ocpTT elements

Posted by [Susanne Wunsch railML](#) on Wed, 30 May 2012 15:31:46 GMT

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

Hi all,

The Wiki page about the <ocpTT> element states (Constraints) [1]:

"The sequence of the ocpTT elements inside a trainPart has to be according to the train path."

A general rule for XML design is not to evaluate the order of elements unless it is of importance, e.g. mixed content issues in document specific markup.

In this case the logical sequence of the <ocpTT> elements is defined by its arrival and departure times (including days). There is no need to require this order with the XML syntax.

We introduced an additional attribute for ordering if it was needed.

It's the same issue with all <trackElements> in the Infrastructure sub-schema that don't have to be ordered neither by the relative nor by the absolute mileage.

An export interface possibly orders its ocpTT elements chronologically. But an import interface should be aware of the possible chronological mix of ocpTT elements.

I would change the Wiki page after some possible discussion.

[1] <http://wiki.railml.org/index.php?title=TT:ocpTT>

Kind regards...

Susanne

--

Susanne Wunsch

Schema Coordinator: railML.common

Subject: Re: Sequence of ocpTT elements

Posted by _____ on Wed, 30 May 2012 18:21:52 GMT

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

I agree with Susanne.

Hi Susanne,

couldn't there be the case that the times in two ocpTts are identical,
in particular, if times are minute-based?

I have to say that non-ordered ocpTts would currently break our import.
The wiki is not versioned, but if preconditions are altered it may pose
problems. In this case, one would have to add versioning to the
documentation and changes could be only to non-released versions.

Best,
--Andreas.

Am 30.05.2012 17:31, schrieb Susanne Wunsch:

- > Hi all,
- >
- > The Wiki page about the<ocpTT> element states (Constraints) [1]:
- >
- > "The sequence of the ocpTT elements inside a trainPart has to be
- > according to the train path."
- >
- > A general rule for XML design is not to evaluate the order of elements
- > unless it is of importance, e.g. mixed content issues in document
- > specific markup.
- >
- > In this case the logical sequence of the<ocpTT> elements is defined by
- > its arrival and departure times (including days). There is no need to
- > require this order with the XML syntax.
- >
- > We introduced an additional attribute for ordering if it was needed.
- >
- > It's the same issue with all<trackElements> in the Infrastructure
- > sub-schema that don't have to be ordered neither by the relative
- > nor by the absolute mileage.
- >
- > An export interface possibly orders its ocpTT elements chronologically.
- > But an import interface should be aware of the possible chronological
- > mix of ocpTT elements.
- >
- > I would change the Wiki page after some possible discussion.
- >
- > [1] <http://wiki.railml.org/index.php?title=TT:ocpTT>
- >
- > Kind regards...

> Susanne

>

Subject: Re: Sequence of ocpTT elements

Posted by [Susanne Wunsch railML](#) on Thu, 31 May 2012 08:52:03 GMT

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

Hi Andreas,

Andreas Tanner <ata@ivu.de> writes:

> couldn't there be the case that the times in two ocptts are identical,
> in particular, if times are minute-based?

I think, in that case we should introduce an attribute for ordering. The geographical/linear reference may not help out in cases where the ocp is only defined with an "id" and a "name".

> I have to say that non-ordered ocptts would currently break our
> import. The wiki is not versioned, but if preconditions are altered it
> may pose problems. In this case, one would have to add versioning to
> the documentation and changes could be only to non-released versions.

That's the reason why I opened a thread for this issue. Thanks for your quick response.

We could also change the currently documented behaviour with a next release not changing such sensitive portions of wiki documentation inbetween schema releases.

Other/same opinions?

Kind regards...
Susanne

--

Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: Sequence of ocpTT elements

Posted by on Thu, 31 May 2012 11:43:07 GMT

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

Dear Susanne and Andreas,

> Other/same opinions?

From my side, it is up to the reading software

- either to sort the times chronologically
- or to declare that OCPTTs have to be ordered on input (additionally to RailML).

I do not see a big problem in the additional declaration. I think that there always will be additional demands on the softwares dealing with RailML.

Actually, we currently have arrival/departure times which come sometimes in a non-sorted order (from an Austrian Infrastructure Company) for reasons which I do not know. We sort them on input and refuse the input if there are two with the same time. It is up to the data source to secure data integrity.

From our side, a kind of "ordering index" as an additional attribute does not change the situation very much: Either we sort by arrival/departure times and refuse if there are two OCPTTs with the same times or we sort by index and refuse if there are two OCPTTs with the same index...

If you consider introducing a new "ordering attribute", may be a "running length" (meters calculated from the beginning of train's route) would be solution which also allows a unique order but includes the additional value of the distances which many reading programmes want to have and which otherwise can only be calculated more difficulty.

Best regards,
Dirk.

Subject: Re: Sequence of ocpTT elements

Posted by [Andreas Tanner](#) on Fri, 01 Jun 2012 08:51:17 GMT

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

If the standard is relaxed in this point, any software has the problem of solving sorting ambiguities. The case of identical times is not purely academic but does occur in practice. So why, without need, introduce ambiguities?

If a condition of the order of child elements is bad XML style, I would follow Susannes suggestion for an ordering index, and introduce the precondition (by documentation) that the times must be weakly ascending by the index.

The index would have to be mandatory, and that's why I would not implement it as a metering index because possibly one would have to write fictitious data if the metering is unknown. Moreover, in theory (maybe someone uses railML for model trains?) there could be two ocpTts

with same meter...

It seems as if this was a breaking change, so I would prefer leaving it for 3.0.

--Andreas.

Am 31.05.2012 13:43, schrieb Dirk Bräuer:

> Dear Susanne and Andreas,

>

>> Other/same opinions?

>

> From my side, it is up to the reading software

> - either to sort the times chronologically

> - or to declare that OCPTTs have to be ordered on input (additionally to RailML).

>

> I do not see a big problem in the additional declaration. I think that

> there always will be additional demands on the softwares dealing with

> RailML.

>

> Actually, we currently have arrival/departure times which come sometimes

> in a non-sorted order (from an Austrian Infrastructure Company) for

> reasons which I do not know. We sort them on input and refuse the input

> if there are two with the same time. It is up to the data source to

> secure data integrity.

>

> From our side, a kind of "ordering index" as an additional attribute

> does not change the situation very much: Either we sort by

> arrival/departure times and refuse if there are two OCPTTs with the same

> times or we sort by index and refuse if there are two OCPTTs with the

> same index...

>

> If you consider introducing a new "ordering attribute", may be a

> "running length" (meters calculated from the beginning of train's route)

> would be solution which also allows a unique order but includes the

> additional value of the distances which many reading programmes want to

> have and which otherwise can only be calculated more difficultly.

>

> Best regards,

> Dirk.

Subject: Re: Sequence of ocpTT elements

Posted by [Joachim Rubröder railML](#) on Mon, 04 Jun 2012 15:58:20 GMT

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

Hello everybody,

- > The index would have to be mandatory, and that's why I would not
- > implement it as a metering index because possibly one would have to
- > write fictitious data if the metering is unknown. Moreover, in theory
- > (maybe someone uses railML for model trains?) there could be two ocpTTs
- > with same meter...
- > It seems as if this was a breaking change, so I would prefer leaving it
- > for 3.0.

In order to avoid breaking changes, what do you think about introducing an optional attribute "sequence" for the ocpTT in version 2.2 and declare that it will become required for 3.0?

Kind regards,
Joachim

--

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

Subject: Re: Sequence of ocpTT elements
Posted by on Tue, 05 Jun 2012 15:15:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Joachim and Andreas,

- > In order to avoid breaking changes, what do you think about introducing
- > an optional attribute "sequence" for the ocpTT in version 2.2 and declare
- > that it will become required for 3.0?

I would of course welcome it.

I would also not see any problem in declaring it required from 2.2. Anybody who implements 2.2 has to change at least something (the namespace location?). It should not be demanded too much to add such a simply counting attribute. So if we consider it being required we can - from our side - do it from the beginning.

If there would be a "sequence" attribute in 2.2 - whether required or not - we would always write it from the first release of 2.2 and also we would require it on input. There is no reason to move forward the programming effort - it will not become easier.

Concerning my suggestion of a "distance" instead of a "sequence": I accept that it would not be a good idea. It should be possible to write RailML files without knowing the distance. So: forget it.

It was never a question that there may be two OCPs with the same times. It is typical for railway timetables to use 1/10 of a minute as the time resolution. Nowadays, one can travel several hundred meters during 0,1 minutes, passing some blocking signals or even stations.

Well, a "sequence" from 2.2 would be fine, I would opt for it being required but also accept if it would be optional.

Best regards,
Dirk.

Subject: Re: Sequence of ocpTT elements
Posted by [Andreas Tanner](#) on Thu, 07 Jun 2012 14:01:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear everyone,

I would vote for Joachim's proposal to avoid an - even if only formal - incompatibility between minor versions.

--Andreas.

Am 05.06.2012 17:15, schrieb Dirk Bräuer:

> Dear Joachim and Andreas,
>
>> In order to avoid breaking changes, what do you think about
>> introducing an optional attribute "sequence" for the ocpTT in version
>> 2.2 and declare
>> that it will become required for 3.0?
>
> I would of course welcome it.
>
> I would also not see any problem in declaring it required from 2.2.
> Anybody who implements 2.2 has to change at least something (the
> namespace location?). It should not be demanded too much to add such a
> simply counting attribute. So if we consider it being required we can -
> from our side - do it from the beginning.
>
> If there would be a "sequence" attribute in 2.2 - whether required or
> not - we would always write it from the first release of 2.2 and also we
> would require it on input. There is no reason to move forward the
> programming effort - it will not become easier.
>

Subject: Re: Sequence of ocpTT elements
Posted by [Joachim Rubröder railML](#) on Fri, 08 Jun 2012 13:52:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi everyone,

there seems to be an agreement about the usefulness of this 'sequence' attribute.

I opened a 2.2-ticket for it (<http://trac.assembla.com/railML/ticket/149>) and another one (#150) for 3.0 to make it mandatory.

It will be available in the next revision.

Kind regards,

Joachim

--

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

Subject: Re: Sequence of ocpTT elements
Posted by [Susanne Wunsch railML](#) on Fri, 09 Nov 2012 09:41:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Dirk, Andreas, Joachim and others,

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

> there seems to be an agreement about the usefulness of this 'sequence'
> attribute.
> I opened a 2.2-ticket for it (<http://trac.assembla.com/railML/ticket/149>)
>) and another one (#150) for 3.0 to make it mandatory.

The optional 'sequence' attribute is already introduced by Joachim as positive integer value starting with "1":

<http://trac.assembla.com/railML/changeset/422>

Kind regards...
Susanne

--

Susanne Wunsch
Schema Coordinator: railML.common
