
Subject: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Marharyta Vyskarka](#) on Thu, 03 Apr 2025 16:54:24 GMT

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

Dear all,

I propose some constraints and edits to be considered for railML3, all related to references of <baseltineraryPoint>s from <range>s either in <itinerary>, <operationalTrainVariant> or <commercialTrainVariant>.

I propose these constraints:

TT:009: The sequence of <baseltineraryPoint>s referenced by <itinerary>'s <range>s must be a continuous path.

TT:010: <baseltineraryPoint>s referenced by <itinerary>'s <range>s must be increasing in time.

TT:011: <baseltineraryPoint>s referenced by <operationalTrainSection>'s <range>s within an <operationalTrainVariant> must be increasing in time.

TT:012: <baseltineraryPoint>s referenced by <commercialTrainSection>'s <range>s within a <commercialTrainVariant> must be increasing in time.

TT:013: <baseltineraryPoint>s referenced by <identifier>'s <range>s within an <operationalTrainVariant> or <commercialTrainVariant> must be increasing in time.

And based on similarity of <identifier> to <commercialTrainSection> and <operationalTrainSection> in this context, I also propose similar to TT:005 and TT:007 constraint:

TT:014: The first(last) <baseltineraryPoint> of each <identifier> within either <operationalTrainVariant> or <commercialTrainVariant> must either be the referenced <itinerary>'s first(last) <baseltineraryPoint>, or coincide with another <identifier>'s last(first) <baseltineraryPoint>.

Additionally constraints TT:005 and TT:007 and proposed TT:014 would need to be reworded. They describe the relation of first and last<baseltineraryPoint>s of the sequence of train sections' referenced points to the train variant's <itinerary>, but they do not mention the relation of the non-initial/ending points to the <itinerary>, only that they should coincide with another section's point. Adding "that belongs to the referenced <itinerary>" at the end of constraint should fix this.

Also existing constraints TT:004 and TT:006 should be deprecated, as existing constraints TT:005 and TT:007 and proposed constraints TT:011 and TT:012 would make violation of these constraints impossible without violation of one the mentioned ones (TT:005, TT:007, TT:011, TT:012). Constraints TT:005 and TT:007 do not allow direct overlap of <baseltineraryPoint>s in the section due to continuous sequence of coinciding end-start points, and TT:011 and TT:012 would not allow repeating of the coinciding <baseltineraryPoint>s in the sequence due to their increase in time. The constraints TT:011 and TT:12 make more general sense, so they need to be considered instead of existing TT:004 and TT:006.

Please let me know what you think.

Best regards,

Marharyta Vyskarka

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by on Tue, 08 Apr 2025 09:50:30 GMT

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

Hi Marharyta,

so far I have not had the time to think about all of the proposed SemCons in detail, but I wanted to give some feedback in advance:

- Some of your suggestions concern increasing times in baseltineraries. Times in baseltineraries consist of 2 components: a xs:time attribute (00:00:00-24:00:00) and a day offset (integer). So, if we talk about increasing times, both components have to be considered. I would mention this in the SemCon description to avoid misunderstandings.

- Generally I agree, that times within (base-) itineraries should be increasing, but in railML2 we make a difference between times in operational and in commercial trains: We require increasing times for operational trains, but are more tolerant for commercial trains, see:

https://wiki2.railml.org/wiki/Dev:Midnight_overrun, example in section "Reference place for day indexes / midnight overruns". For example, it is not strictly forbidden to end at one trainpart/baseltinerary with "23:58:00, dayOffset=0" and to continue the next one at the same station with "00:01:00, dayOffset=0". We will have to discuss, whether we want to keep this "contradiction" in railML3 or find another solution.

Best regards

Christian

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Marharyta Vyskarka](#) on Wed, 09 Apr 2025 17:30:09 GMT

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

Hello everyone,

Thank you for the feedback!

The brought up above first point does make sense and the wording for constraints could be changed to having "which is expressed with the components //baseltineraryPoint//times//@time, //baseltineraryPoint//times//@dayOffset and //itinerary/range/@offset" at the end of the definitions.

I also realized a constraint is missing for <baseltinerary> itself. So similar constraint could make

sense:

"The times of <baseltineraryPoint>s of the same scope, which are expressed with the components //baseltineraryPoint//times//@time and //baseltineraryPoint//times//@dayOffset, within the same <baseltinerary> must be increasing in time."

As for the second point, yes, the handling of this situation for railML 3 would need to be discussed to determine how it influences the proposed constraints.

Best regards,
Marharyta Vyskarka

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Marharyta Vyskarka](#) on Wed, 09 Apr 2025 17:35:27 GMT

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

Hello everyone,

I also wanted to note that constraints TT:004 and TT:006 should be deprecated regardless of proposed constraints TT:011 and TT:012, as I came to conclusion that existing constraints TT:005 and TT:007 fully cover this case. Which means that if all the section's end-start <baseltineraryPoint>s coincide as constraints TT:005 and TT:007 describe, the situation described in TT:004 and TT:006 is not possible.

If you have any idea as to why they should not be deprecated, please let me know.

Best regards,
Marharyta Vyskarka

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [David Lichti](#) on Thu, 10 Apr 2025 07:42:26 GMT

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

Good Morning Marharyta,
I have quite some objects to these proposals:

TT:009 What exactly do you mean by continuous path? Whether a sequence of base itinerary points is connected by a continuous, drivable track path is beyond railML.org's capacity to validate. So, it should not be a semantic constraint.

TT:010, TT:011, TT:012, TT:013 This would amount to the railML format enforcing business rules.

But railML.org is not a business regulation body. I don't think railML as a data format would gain much from enforcing any notion of timetable validity. In particular, it would render railML unusable in any context where the validation of schedule times is done on the receiving side of the communication.

Do you mean that the referenced base itinerary points must be well-ordered in the base itinerary? This would be a reasonable constraint. But it should not be based on any timing data.

TT:014 It was an explicit design goal of identifiers that they can overlap each other, even if they are of the same type. Also, I don't think that there is a general need for railML as a data format to enforce that every part of a train is covered by an identifier of any kind. This is in stark contrast to the situation with itinerary ranges, which is why the principles of TT:005 and TT:007 do not apply to identifiers.

TT:004, TT:005, TT:006, TT:007 These constraints were developed together, and they depend on each other to achieve their goal: Each section of a train variant's itinerary must be covered by exactly one train section.

For example, with TT:004 removed, there could be two train sections, both covering the entire itinerary. This would satisfy TT:005, since all referenced base itinerary points are first or last in the underlying itinerary. But such an itinerary sectioning would not make any sense, because each section should describe different parts of the itinerary.

Similarly, with TT:005 removed, there could be just one single train section, with a range from some intermediate base itinerary point to some other intermediate base itinerary point. TT:004 would be satisfied, since there are no other itinerary sections that could overlap. But the train variant would be completely missing a schedule description for the first and last part of the underlying itinerary.

There is, in fact, one flaw with these constraints: They do not enforce a well-ordering of the base itinerary points referenced by a range in a train section that is cancelled or on request. That is: It would be allowed to have a range where the start base itinerary point comes after the end base itinerary point. If it satisfied TT:005, it would necessarily overlap with some other train section. But that would not be a violation of TT:004 if the parent train section was cancelled or on request.

This would be worth addressing with semantic constraints everywhere ranges are used. But, as explained above, these constraints should be formulated based on the order of base itinerary points in their respective itineraries and base itineraries, not based on any timing information which may or may not be present.

Best regards

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Milan Wölke](#) on Mon, 19 May 2025 14:26:26 GMT

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

Hi all,

we discussed this at our last timetable developer meeting.

Regarding TT:009 we found that we shouldn't try to make it as strict as the wording above implies. As David pointed out railML.org cannot really verify if the path is actually continuous. Additionally it may actually be required to specify a path that is specifically not continuous in the sense that a train arrives at one track but departs from another. There is railways that actually operate trains like that. In the end we would propose an updated wording for TT:009:

When combining baseltineraries in an itinerary it shall be ensured that the last baseltineraryPoint of the ending baseltinerary references that same operational point as the first baseltineraryPoint of the starting baseltinerary.

Regarding the proposed semantic constraints TT:010-TT:013 we came to the conclusion that railML should be able to also communicate timetables that are not finished or not consistent. A possible use case for this which we actually supported in railML 2.x was that a system exports a timetable and another specialized system analyses that timetable and outputs issues and errors with it. For this use case to work it is necessary to also communicate times that are inconsistent with each other.

We also reviewed your proposal to deprecate the semcons TT:004 and TT:006 in favor of TT:005 and TT:007 because a possible redundancy. During the meeting a colleague was able to show that there actually is no redundancy. Your argument was that it is not possible to violate TT:005/TT:007 which explains that one part of a trainSection needs to end at the point another starts without also violating TT:004/TT:006 which ensures that there is no overlaps between trainSections of a trainVariant. One example where you could violate TT:005 without breaking TT:004 is if you have 2 train sections starting and ending at the same point. This would be in line with TT:005 but is a violation according to TT:004. As such we would argue that the existing semcons TT:004, TT:005, TT:006 and TT:007 shall remain.

Best regards, Milan

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Milan Wölke](#) on Tue, 20 May 2025 10:28:40 GMT

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

Another idea we discussed was the introduction of yet another rule, saying that the baseltineraryPoints referenced by start and end of the range of an operationalTrainSection or a commercialTrainSection should be increasing with regard to their sequence numbers. What makes things hard to find a good wording, is that these baseltineraryPoints may belong to different baseltineraries. Logically this is not a problem as the ranges of the itinerary have a sequence number as well but when it comes to creating a semcon wording for this it gets complicated.

I would propose the following wording for this issue:

In an <operationalTrainSection>, the <range> element's @start and @end must reference <baseltineraryPoint> elements such that their order aligns with the sequencing defined by the <itinerary>.

Specifically:

If the start and end points belong to different <baseltinerary> elements, the <range> elements referencing these <baseltinerary>s in the <itinerary> must have increasing @sequenceNumber.

If both points belong to the same <baseltinerary>, their own @sequenceNumber must increase from start to end.

The same would probably be needed for commercialTrainSections.

What do you think would this wording work? Is it understandable enough? Am I missing a case?

Best regards, Milan

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Milan Wölke](#) on Thu, 26 Jun 2025 10:54:07 GMT

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

Hi all,

during our last timetable developer meeting we discussed the proposed wording for TT:009 and found that it lacked definition of how the order of referenced baseltineraries is described in the semcon. With the following updated wording I am trying to improve on that.

When combining baseltineraries in an itinerary it shall be ensured that the last baseltineraryPoint of the preceding baseltinerary references that same operational point as the first baseltineraryPoint of the succeeding baseltinerary. The order of the baseltineraries is determined by the sequenceNumber specified in the range of the itinerary.

Please let me know if this is clear. When you read this, do you understand how things are meant to be?

Thanks in advance for any feedback.

Best regards, Milan

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Milan Wölke](#) on Thu, 26 Jun 2025 11:43:59 GMT

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

We also discussed the wording for the newly proposed semcon for `operationalTrainSection` and `commercialTrainSection`. We decided that we should only stick to the general rule in the semcon and move the explanatory sentences to the best practice section of the wiki page. This would mean that the semcon wording would be like this:

In an `<operationalTrainSection>`, the `<range>` element's `@start` and `@end` must reference `<baseltineraryPoint>` elements such that their order aligns with the sequencing defined by the `<itinerary>`.

For the explanatory text it was pointed out that also baseltineraries with just a single `baseltineraryPoint` are possible and should be considered in the wording. This would mean the core explanation for the semcon would be like this:

If the start and end points belong to different `<baseltinerary>` elements, the `<range>` elements referencing these `<baseltinerary>`s in the `<itinerary>` must have increasing `@sequenceNumber`.

If both points belong to the same `<baseltinerary>`, their own `@sequenceNumber` must increase from start to end, unless the baseltinerary consists of only a single `baseltineraryPoint`.

Please let us know what you think. Clear enough, or would you have trouble understanding the intended regulation.

Best regards, Milan

Subject: Re: [railML3] Proposal for new semantic constraints and change of existing ones

Posted by [Milan Wölke](#) on Wed, 09 Jul 2025 12:26:09 GMT

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

Hi all,

during the last timetable developer meeting we decided go forward with the two semantic constraints above:

TT:009:

When combining baseltineraries in an itinerary it shall be ensured that the last `baseltineraryPoint` of the preceding baseltinerary references that same operational point as the first `baseltineraryPoint` of the succeeding baseltinerary. The order of the baseltineraries is determined by the `sequenceNumber` specified in the range of the itinerary.

TT:010:

In an <operationalTrainSection>, the <range> element's @start and @end must reference <baselineraryPoint> elements such that their order aligns with the sequencing defined by the <itinerary>.

We will decide about approval of these in one of the next modelling telcos among the coordinators.

Best regards, Milan
