Subject: [railML2] Consistent definitions of formationTT@load @weight and @timetableLoad
Posted by Torben Brand on Mon, 23 Nov 2020 19:04:38 GMT
View Forum Message <> Reply to Message

Dear railML-community,

I would like to point your attention towards the forum posting for missing load attribute in RS:vehicle
https://www.railml.org/forum/index.php?t=msg&th=767& start=0&

In this context we would like to ask the TT community to suggest a clearer definition of the three attributes (bellow with today's wiki definitions):

@weight: the real weight of the formation while in use, including any possible engine(s)
@load: (introduced with version 2.1) the real load of the formation while in use, excluding any possible engine(s)
@timetableLoad: (introduced with version 2.1) planned load according to the timetable, excluding any possible engine(s)

What is meant with "real"? How do the definitions relate to <formation>/<vehicle> @bruttoWeight, @nettoWeight and @tareWeight?

Please see attached illustration in RS forum posting, where I try to illustrate the definitions.
What does the TT community think? Any thoughts are much appreciated.

Kind regards

Torben Brand
Jernbanedirektoratet

Subject: Re: [railML2] Consistent definitions of formationTT@load @weight and @timetableLoad
Posted by Thomas Nygreen on Mon, 30 Nov 2020 16:09:49 GMT
View Forum Message <> Reply to Message

Dear Torben,

I found this trac ticket from the introduction of timetableLoad: https://trac.railml.org/ticket/59 (in German)

And this old related thread: https://www.railml.org/forum/index.php?t=msg&goto=784

Best regards,
Thomas

Subject: Re: [railML2] Consistent definitions of formationTT@load @weight and @timetableLoad
Posted by Torben Brand on Fri, 19 Feb 2021 13:21:08 GMT
View Forum Message <> Reply to Message

Dear TT comunity,

I would like to address two issues here:
1. Change the use of the term "real" to "actual"
2. Indicate the use of wagons with the use of the attribute @load or @timetableLoad

Change the use of the term "real" to "actual"
According to the forum post and ticket referenced above by Thomas Nygreen and the forum post in rolling stock by Jörg von Lingen (https://www.railml.org/forum/index.php?t=msg&th=767&goto=2597&#msg_2597), we seem to be in conclusion that we should change the word "real" with "actual".
The primary purpose in formationTT is to indicate the planned values of the formations length, weight/load and/or speed per individual train in the timetable if different than indicated in the formation. The formation values being the physically maximum capable values. The actual driven values seem to bee of less importance for the community. Thus the suggestion to change the definitions for @length and @speed to being defined as for the planned values. Alternative we could keep the definition open if they are for the planned or the actual values (since there is only one set as opposed to load and timetableLoad).

Indicate the use of wagons with the use of the attribute @load or @timetableLoad
As defined in the existing Notes section a zero load indicates no wagons in the formation. We suggest thus to also define the opposite to be true. That if there is defined a load or timetableLoad of >0 to indicate that the formation contains wagon(s). If the formation only contains an engine then the use of formationTT@load or @timetableLoad and a formation@length adds virtual wagons with the defined load and length values. If there are defined wagons in the formation (either real wagons with real weight and length values or a dummy wagon of 1m length and 1t tareweight) the formationTT values overwrite these values.

wiki page revision suggestion for formationTT
@weight: the actual weight of the formation while in use, including any possible engine(s)
@load: (introduced with version 2.1) the actual driven load of the formation, excluding any possible engine(s)
@lenght: the planned length of the complete formation according to the timetable.
@speed: the planned maximum speed of the formation according to the timetable.
@timetableLoad: (introduced with version 2.1) the planned load according to the timetable, excluding any possible engine(s)

Notes / Anmerkungen / Notes
Due to the habits of railway terminology, "load" here means the mass of any wagons and carriages including their payload (passengers, freight) but does not include the mass of any engines. In the case the formation consists of engines only (including multiple units), load is defined to be zero and does not include the mass of any passengers or freight (which may be conveyed or transported inside the engines or multiple units), with the use of vehicle@bruttoWeight.

A load or timetableLoad of >0 indicates that the formation contains wagon(s). If the formation only contains an engine then the use of formationTT@load adds virtual wagons with the defined load and length values. If there are defined wagons in the formation the formationTT values overwrite these values.

The attribute weight contains the total physical mass, always including all engines, multiple units and payload.

On the difference between ocpTT/@trainReverse and formationTT/@orientationReversed, see Reversing trains and formations.

Note that the formation values in rolling stock are the physically maximum capable values of the formation.

Kind regards

Torben Brand on behalf of Jernbanedirektoratet

Subject: Re: [railML2] Consistent definitions of formationTT@load @weight and @timetableLoad
Posted by on Tue, 12 Jul 2022 13:29:03 GMT
View Forum Message <> Reply to Message

Dear Torben and all,

since Milan asked to get a reply on this post: There are no objections from our side. We agree that a @load or @timetableLoad at a <formation> with no wagons is a valid way to encode a load with "virtual" wagons.

Some hints on wording:

> If there are defined wagons in the formation (either real wagons with real weight and length values or a dummy wagon of 1m length and 1t tareweight) the formationTT values overwrite these values.

Acceptable, but very much on the borderline... I would object against any intentionally coding of wrong train lengths. So a wagon of a placeholder-length 1 m or mass 1 t would _have_ _to_ _be_ overwritten by the formation or train. It would not be acceptable that any of these placeholders come into effekt at the final train. (I would prefer to define wagons without length and mass instead of placeholder values.)

> @weight: the actual weight of the formation while in use, including any possible engine(s)
> @load: (introduced with version 2.1) the actual driven load of the formation, excluding any

possible engine(s)
> @lenght: the planned length of the complete formation according to the timetable.
> @speed: the planned maximum speed of the formation according to the timetable.
> @timetableLoad: (introduced with version 2.1) the planned load according to the timetable, excluding any possible engine(s)

I recommend avoiding the word "actual" since we are still in <timetable>: Everything here is planned, in advance. It is usually not the task of <timetable> to encode data in retrospect, so "actual" data of the past. This would open up the problem that a train can only have a certain (actual) load per day, so the load would differ from day to day. In case you do not agree to this and you want to allow to encode data of the past here, there should be at least the constraint that in such a use case only _one_ operating day (of the past) is allowed.

Best regards,
Dirk.

Subject: Re: [railML2] Consistent definitions of formationTT@load @weight and @timetableLoad
Posted by Milan Wölke on Mon, 25 Jul 2022 09:45:45 GMT
View Forum Message <> Reply to Message

Hi all,

regarding the wording I have to agree with Dirk. We shouldnt encourage encoding of intentionally wrong data in the documentation. The semantics you are trying to describe here (i.e. the rule of overwriting), however, I agree with.

Best regards, Milan