

---

Subject: Re: Extension rolling stock for capacity planning  
Posted by on Thu, 18 May 2017 13:16:20 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Dear Torben,

Concerning:

Resistance factors for calculating train resistance,  
well established formulas in the sector (Strahl, Sauthoff and Davies)

I agree from the basics.

But I am not convinced that including directly such rather  
"old-fashioned" formulas like Strahl and Sauthoff into railML. At least,  
if we do so, we should first provide more general formulas before the  
specials.

Strahl, Sauthoff and Davies include a combination of several different  
resistances in their formulas and lack other resistances. So, for  
instance, they normally include air resistance and head wind but only in  
"open air" much by empiric.

There is already a solution to provide general formulas in railML:  
mathml. So if necessary, to keep railML as the general exchange format  
it is intended to be, Strahl, Sauthoff and Davies with their  
coefficients could be written in mathml. Everything else would be a  
redundancy, wouldn't it?

We also have such values and a <railML> implementation of it. It  
produces <railML> with formulas in mathml. If now there will be a  
special solution for Strahl, Sauthoff and Davies, I would

- add some more general formulas from my side,
- like to have a very conscious decision to avoid "uncontrolled growth",
- ask what we should do with our implementation: Change to special  
solution, import redundantly both mathml and special solution or only  
support mathml.

---

Concerning

<formation><trainEngine>@trainMinTimeHoldSpeed

Again, I agree from the basics.

But I do not see that this depends on certain rolling stock. From my  
experience, it is a general value, may be network-wide or depending on  
certain lines but not vehicles.

We would use a kind of @minimumSpeedMaintainingTimeDelay as a

network-wide default value and in <train>.<trainPartSequence> to optionally overwrite the default value but currently there is no demand for it in <railML>.

If such a value will be introduced, I want to suggest there should be

- a default value for all <railML> file,
- a value per <line> or <track> in infrastructure to overwrite the default value,
- a value per vehicle or formation if this will be agreed as necessary,
- a value per <trainPartSequence> to overwrite the value coming from default, infrastructure or vehicle.

The priority shall be from top to bottom of my list.

---

Concerning

<vehicle><vehicleBrakes><vehicleBrake>@brakeSupervision:"none/ATP/ETCS/other: "

Such an attribute would be useful in general. Please consider that braking does not only depend on the vehicle properties. It also depends on the line-side infrastructure and possibly on operation rules. So, the same vehicle may have different @brakeSupervision (along with other brake-related properties) depending on the line and train it operates.

It would be desirable to have

- an enumeration of such attributes at vehicles with a priority,
- the same basic types for <infrastructure>, <rollingStock> and <timetable>.

I agree with you and with Jörg von Lingen concerning

> In case other deceleration curves are needed it would be better to enhance the tBrakeSupervision type.

For the sake of completeness I want to add:

Currently there is a differentiation between emergency brake values and regular (operational) brake values in <train>.<trainPartSequence>.<brakeUsage> and <rollingStock>...<vehicleBrakes>.

The intention is that regularBrakePercentage, regularBrakeMass, meanDeceleration a.s.o. define the behaviour of the train when either @brakeSupervision=none or in all other cases as the maximum possible regular brake acting. (@brakeSupervision=ATP/ETCS/other can only \_reduce\_ the brake acting, not raise it as long as we keep in planning and do not plan emergency brakes).

With best regards,

Dirk.

---