Subject: Re: [railML2] Modelling of crossings/overtakings
Posted by on Thu, 19 Mar 2020 09:44:38 GMT
View Forum Message <> Reply to Message

Dear Janne,
dear community,

I dare to write that so far, it was not the intention of railML to explicitly model crossings or overtakings of trains. Therefore, I have some doubts on what you suggest:

1)
During the timetable developer meeting on 21st January 2020 in Dresden, it was discussed to use the <connection> element for more timetable linkings than passenger-taffic connections. There was no general agreement on this. So we must conclude that so far, passenger-taffic connections are the only timetable linkings which <connection> is intended to be used for - unfortunately. That's also why join/split/turnaround have been deprecated there since r2.1, and "meet", "IsWaitingFor" and "IsExpectedBy" have been clarified to be linked with _passengers_.

2)
There is probably no general accepted definition of what is a crossing and what is an overtaking. Consider that more than two trains can be involved - a train can be "isWaitingFor" and "isExpectedBy" at the same time, a crossings and an overtaking can occur at the same time. Some people would say that a crossing can also happen at the first or last station of a train. Other people would say that at the last station of each train, sooner or later always a next train begins, but despite this there is not always a crossing or overtaking at the first or last station of all trains.

This is of course no criterion for exclusion of crossings and overtakings from railML, but as you see, this leads very far. Surely the usage could only be a matter of use case.

3)
There is, as far as I can see, no hard exclusion to use <stopActivity> at ocpType='pass'. But it is surely not intended.

Please note that <stopActivity @type="occupationBlock"> does not necessarily encode an overtaking. The block can also be blocked because of a train in advance which never did overtake. On the contrary, this could be misleading because it should mean that the reason for the stop is that the block in advance is occupied when the train arrives (when the stop begins). This is normally not the case at the beginning of an overtaking.

However, my personal opinion:

I would welcome a possibility to encode crossings and overtakings in railML because we also have them in some German driver's timetables. But it would have to be a very flexible model (taking into account more than one train, operating days of trains etc.) and the actual definition of what is a crossing and an overtaking should be left to the use case.

In current railML 2.x, I would welcome to use <connection> for more timetable linkings, especially

for (use-case dependant) crossings and overtakings. But at the same time, we must avoid that <connection> is used to create redundancies to <rostering> - and especially avoid that it is used instead of <rostering>.

I would prefer not to spread one background to two different places in railML. Means: keep <stopActivity> out of this, hold it all together at <connection>. Extend <connection> to encode more information on crossings and overtakings if necessary. Do not allow <stopActivity> at ocpType='pass'.

I think that your demand on encoding crossings and overtakings is general enough to be met by railML and should not be solved purely by extensions.

Best regards,
Dirk.
