

Dear Joerg,

Sorry for the late reply on this topic. Also, sorry for the amount of text in this post. I wanted to give some system background to

Bane NOR's use case for railML is, first and foremost, ETCS lines. These probably differ a bit from the conventional interlocking principles that (naturally) seem to be used as a basis for railML. So first, just a quick explanation on how release of train routes will work on ETCS lines in Norway (at least in the current state). In case of no specific design for release of train routes, the following principles are generic functions (taken from our interlocking guidelines):

- Objects in a train route and its flank protection are released when related TVP section is passed by the entire train.
- Objects in FS/OS-route and its flank protection are released when a train in the route has stopped and is deregistered.
- End point of a train route, that is extended with another route, is released when it is passed by the train's last axle.

Comments: A train route can release when train has passed TVP section specified in Ch 8.2.4. Releasing train route [Utløsning togvei], see the below points on specific release after a passed train detector.

- End point of a train route, at the border to a non-interlocked area, is released when it is passed by the train's last axle.
- FS/OS-route and subsequent overlap is released independently of each other when they are released by a train.
- FS/OS-route and subsequent overlap are released simultaneously when they are cancelled by command from the train dispatcher.
- An end point (signal) is released when both FS/OS-route to it and the overlap after it is released.

In addition there are separate rules for release of overlap in the generic applications, that are defined as follows (and do not necessarily correspond with the release of a train route to the signal that has the overlap):

- An overlap is not removed when FS/OS-routes are extended but is overtaken by the extended route.
- An overlap is released and release speed is set to 0 km/h when the last TVP section ahead of end point is occupied, and the train has stopped.
- An end point without an overlap is released when passed by the train's last axle. The release speed is kept even though the train stops ahead of the end point, as long as the overlap is not released.

In addition, we are able to define specific train detectors for release of the entire train route (also the remaining part of the route in question) as explained in the point referring to Ch. 8.2.4 above. In such a case, the following applies:

- The train route is released without time delay.
- All train routes passing this axle counter in the specified direction are released.
- The overlap is still released using the generic principles above, typically when the last TVP section in the respective route is occupied and the train as stopped.

Moving on to the parameter discussion:

Most of our signals have overlaps of XX meters. Additionally, since train route release and overlap release are not occurring at the same time in our system, it does not make sense to connect a train detector for route release to an overlap in railML. This means your option 2) is not optimal for our functionality.

Option 1) is a bit closer, I think, depending on how one interprets it. My understanding is that `IL:routeReleaseGroupRear` is meant to indicate the release of parts of a train route, and not the entire route (which would be the case in our system). One could say that, with the knowledge of how the specific signalling system works, such a parameter may be used to indicate the release trigger. And that the knowledge of the system functionality itself would help to know what is released and not. This may be the intention of the IL part of railML in general? If not, I think this option also does not cover our use case perfectly.

Either way, I think the closest railML parameter I have found to our system functionality is the `@releaseTriggerRef` for railML 2.4nor, as I mentioned in the original post. The only difference being whether the overlap is released at the same time.