
Subject: Re: [railML2] Extension proposal: pattern trains, distributions and slots
Posted by [Torben Brand](#) on Tue, 19 Jan 2021 08:58:43 GMT

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

Dear TT community,

Thank you for valuable input for extension of the pattern train. As suggested we at Jernbanedirektoratet are implementing a separate element for both <patternTrain> and <patternTrainPart> with it's own sub elements and attributes. We have selected attributes and sub-elements of <train> and <trainPart> that also fit the use case for pattern trains. The selected elements and attributes are mirrored 1:1 in the implementation of <patternTrain> and <patternTrainPart>, with the same descriptions and types. You find the description of the suggested model in the Word file (chapter 4.4 and 4.5), Illustration Png file (I'll also try to add it in the forum bellow as an image) and Excel file and in the railML cloud for the TT working group (folder /railML2.5/patternTrain). Unfortunately I'm not able to attach the files here in the Forum post reply, due to file attachment seems to be dissabled for me.

How to interpret the times (@arrival and @departure <times>) set in a pattern train relative to the rest of the trains in the group must be defined. We have 4 different alternatives. See also illustration in Word document.

Alternative 1: use <times> representing a possible instance of the pattern train.

In this alternative, <times> is used in the same manner as with normal <trainPart>s. The times used represent a train that can be formed from the pattern train, e.g. the first train of the resulting train group. The drawback with this alternative is that the times can be interpreted as an actually desired path, not just a template.

Alternative 2: use <times> with departure at midnight from first OCP

As alternative 1, but fix the departure from the first OCP to 00:00. The following arrival and departure times will then be interpreted as durations from the origin. The drawback with this alternative is that durations should use type xs:duration, not xs:time.

Alternative 3: do not use <times> and depend on <runTimes> and <stopTimes> for time differences per ocp

In this alternative, <runTimes> and <stopTimes> are used in the same manner as with normal <trainPart>s. This is also allowed in the other alternatives, but in this alternative it is the only information about arrival and departure times. The drawbacks with this approach are that it requires adding together all the time differences and that the separation between @minimalTime, @operationalReserve and @additionalReserve is not necessarily known.

Alternative 4: implement something new

This can for instance be an implementation similar to alternative 2, but with time shifts from the origin using xs:duration instead of timestamps using xs:time.