
Subject: [railML2] modeling of a car ramp

Posted by [Torben Brand](#) on Mon, 03 Jun 2019 09:47:12 GMT

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

Problem description

The Norwegian railML community needs to model a car ramp (Example image <https://images.app.goo.gl/avBsC8MYFjGLvDML7>) on the end of a track for loading/unloading cars.

The purpose/use case is Network statement and schematic track plan.

Mapping

As I see it we can use either choose to use:

- Use <platformEdge>
- use <serviceSection>
- model a (nor) extension for freight equipment on a microscopic level

The wiki states that the <platformEdge> (<https://wiki.railml.org/index.php?title=IS:platformEdge>) is used for "A <platformEdge> A <platformEdge> defines the border line between platform and railway track, where passengers are meant to board and un-board the train."

We could use <serviceSection>

<https://wiki.railml.org/index.php?title=IS:serviceSection>

<https://www.railml.org/forum/index.php?t=msg&th=122> instead

(it is also listed in the railML2.4nor documentation <https://www.jernbanedirektoratet.no/railml> graphics on page 10 to be used as is in Norway).

However, the wiki states that "A <serviceSection> defines the border line between a railway service area and a railway track, where the service for the train takes place. A service section is always connected with exactly one railway track."

So strictly speaking it is meant for, as I understand it, the service of the train/ the trains maintenance. So, freight loading/unloading facilities are not part of it.

There are no freight loading facilities in the railML2.4 on a microscopic level (under trackElements). There are macroscopic elements under ocp<propService> (<https://wiki.railml.org/index.php?title=IS:propService>).

Solution suggestion

A. an extension

An extension should be done properly. So here a complete modelling should be made for all necessary elements and attributes. Probably a new element <freightEquipment> under <trackElements>?. This would demand a proper use case be defined. Which is not in place for now and is not anticipated.

B. Use <platformEdge>

The definition clearly states " where passengers are meant to board and un-board the train" Thus, I understand not to use <platformEdge> for loading facilities like a car ramp.

C. Use <serviceSection>

As an easy solution for now. We could choose to see the term "service" broader.

There is an attribute @ramp (and @loadingFacility for other freight loading). The definition is unfortunately not really helpful:

@ramp:" defining the service section as being a ramp"

@loadingFacility:"defining the service section as being equipped with loading facilities".

I find the suggested definitions in the initial forum posting far more helpful and precise:

"ramp" (ramp for loading / unloading goods)

"maintenance" (maintenance facilities e. g. in a depot)

"loadingFacility" (Goods can be (un-)loaded from the wagon's top / underfloor

I suggest going for @ramp for the car ramp.

Also more important the forum posting for serviceSection (

https://www.railml.org/forum/index.php?t=msg&th=122&goto=283&#msg_283) anticipated loading of freight. The attribute descriptions are clearly meant for loading/unloading of freight.

However, this has been "lost in translation" into the wiki.

I suggest improving the wiki page for <serviceSection> with a clear definition to mean service and freight handling of trains. Also, a more elaborate definition of the attributes would be helpful (like the initial suggested definitions).

Another problem is, as a car ramp is not alongside the track. There is another challenge for using <serviceSection> as it is defined to be alongside the track (like platformEdge). This both in the wiki definition, the presence of a @length attribute and in the @side attribute being restricted to the values "left" and "right".

This could be solved with a semantical rule for objects being at the end of the track instead of alongside it. This either with the rule:

1. No use of the attribute @side and a @length attribute of "0". This would restrict the attribute defining the length of the ramp outside the track area (in front of the track). This might be ok as @length in railML is always seen as along the tracks perspective.
2. No use of the attribute @side and a serviceSection@pos value corresponding to the @pos value of <trackEnd> or <trackBegin> with the sub element <bufferStop> placed.
3. In railML 2.5 extend serviceSection@side with the value "end"

I suggest C2 as the solution for now and C3 for later. What does the railML community think?
