
Subject: [railML3] Usage of <ownsPlatform> vs <ownsInfrastructureElement>
Posted by [Larissa Zhuchyi](#) on Tue, 27 Aug 2024 10:38:09 GMT

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

Dear all,

railML.org suggests to add in railML3 a semantic constraint IS:018 [1] (see text below). Please let us know if:

- 1) it is understandable;
- 2) you have anything against;
- 3) anything is missed in the wording.

Any functional infrastructure element that belongs to an <operationalPoint> may be listed as its equipment. This may be done by adding it to a specific container, such as <ownsPlatform>, <ownsTrack> and <ownsSignal> or it may be added to the generic container <ownsInfrastructureElement>. Any such added infrastructure element must be added to the most specific container available. No element shall be part of two such containers. If no specific container for the functional infrastructure element exists, it shall be listed in the generic container. Example: a <signalIS> must not be added to <ownsInfrastructureElement>. It shall be added to <ownsSignal>. As there is no container for levelCrossings a <levelCrossingIS> belonging to an <operationalPoint> shall be added to <ownsInfrastructureElement>.

See the list of all semantic constraints at [2].

[1] <https://wiki3.railml.org/wiki/IS:opEquipment>

[2] https://wiki3.railml.org/wiki/Dev:Semantic_Constraints

Thanks in advance!

Sincerely,
