
Subject: Re: Include track length information in topology
Posted by [christian.rahmig](#) on Wed, 30 May 2018 08:57:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Martin,

thank you for your extensive conclusion on this discussion. I forwarded the main question to the RTM developing team [1]. We'll see what happens...

[1] <https://www.railml.org/forum/index.php?t=msg&th=579&start=0&>

Best regards
Christian

--

Christian Rahmig - Infrastructure scheme coordinator
railML.org (Registry of Associations: VR 5750)
Phone Coordinator: +49 173 2714509; railML.org: +49 351 47582911
Altplauen 19h; 01187 Dresden; Germany www.railml.org

Am 17.05.2018 um 10:21 schrieb Martin Karlsson:

- > The RailTopoModel contains no information about lengths of
- > tracks. It is concerned only with the relations between the
- > network elements, not with their physical dimensions.
- > However, practically all use cases imaginable will need the
- > track length information to calculate travel times, to
- > compare with train lengths, etc. So in the railML model, it
- > has to be included in some way.
- >
- > This issue has been discussed in a previous forum post
- > (<https://www.railml.org/forum/index.php?t=msg&th=472&start=0&>),
- > and the solution in version 3.1 beta was to introduce a
- > Track object within the FunctionalInfrastructure view. The
- > Track is mapped to a NetElement in the Topology view, but
- > adds a length attribute. A trackside object may be mapped to
- > a position at a distance on a Track through the child
- > elements trackPosition and trackSection, which were added to
- > the RTM originated location elements of spotLocation,
- > linearLocation and areaLocation.
- >
- > It has been argued that this solution adds unnecessary
- > complexity to the model, by in effect acting as a second
- > layer topology model, within the FunctionalInfrastructure
- > view. This could be avoided by instead including length
- > information in the Topology view.
- >
- > A proposal discussed in the In2Rail/railML collaboration

- > team is to
- > Add an attribute "length" of type tLengthM to the class
- > NetElement. NetElement is derived from the RTM class
- > LinearElement, which remains unaltered. tLengthM indicates
- > that the length is in meters, with decimals. The value of
- > "length" shall be the full length between agreed reference
- > points in the delimiting crossings, so that the aggregated
- > length over consecutive tracks always reflects the correct
- > total distance Add an attribute "pos" to the spotLocation
- > child element in the Entity class. In order to do this
- > without changing the RTM classes SpotLocation and
- > LocatedNetEntity, new derived classes may need to be
- > created. The value of "pos" shall be the distance from the
- > agreed reference point in the crossing at leg 0 of the net
- > element In the same way, add attributes "posBegin" and
- > "posEnd" of type tLengthM to the associatedElement element
- > used in linearLocation and areaLocation child elements of
- > the Entity class In the same way, add attributes "posBegin"
- > and "posEnd" of type tLengthM to the associatedElement
- > element used in linearLocation and areaLocation child
- > elements of the Entity class Remove trackPosition and
- > trackSection elements from class Entity Remove class Track
- > Remove abstract base class TrackNode, and reparent its
- > descendants (Switch, Crossing and BufferStop) to Entity. The
- > TrackNode was introduced to allow a Track to refer to the
- > functional assets at its ends, hence it is no longer needed
- >
- > The new attributes should all be optional in the XSD, but
- > the following application rules should apply:
- > "length" shall be mandatory in NetElements within a "Micro"
- > description level network (i.e. representing a track),
- > optional on other levels "pos", "posBegin" and "posEnd" may
- > only be used in Entities referring to a NetElement with a
- > "length" When "pos" is defined, "intrinsicCoord" may be
- > omitted, as it's value can be derived from "pos" and
- > "length" (same rule for posBegin/intrinsicCoordBegin and
- > posEnd/intrinsicCoordEnd respectively)
- >
- >