# railML.org

# railML® 3.2
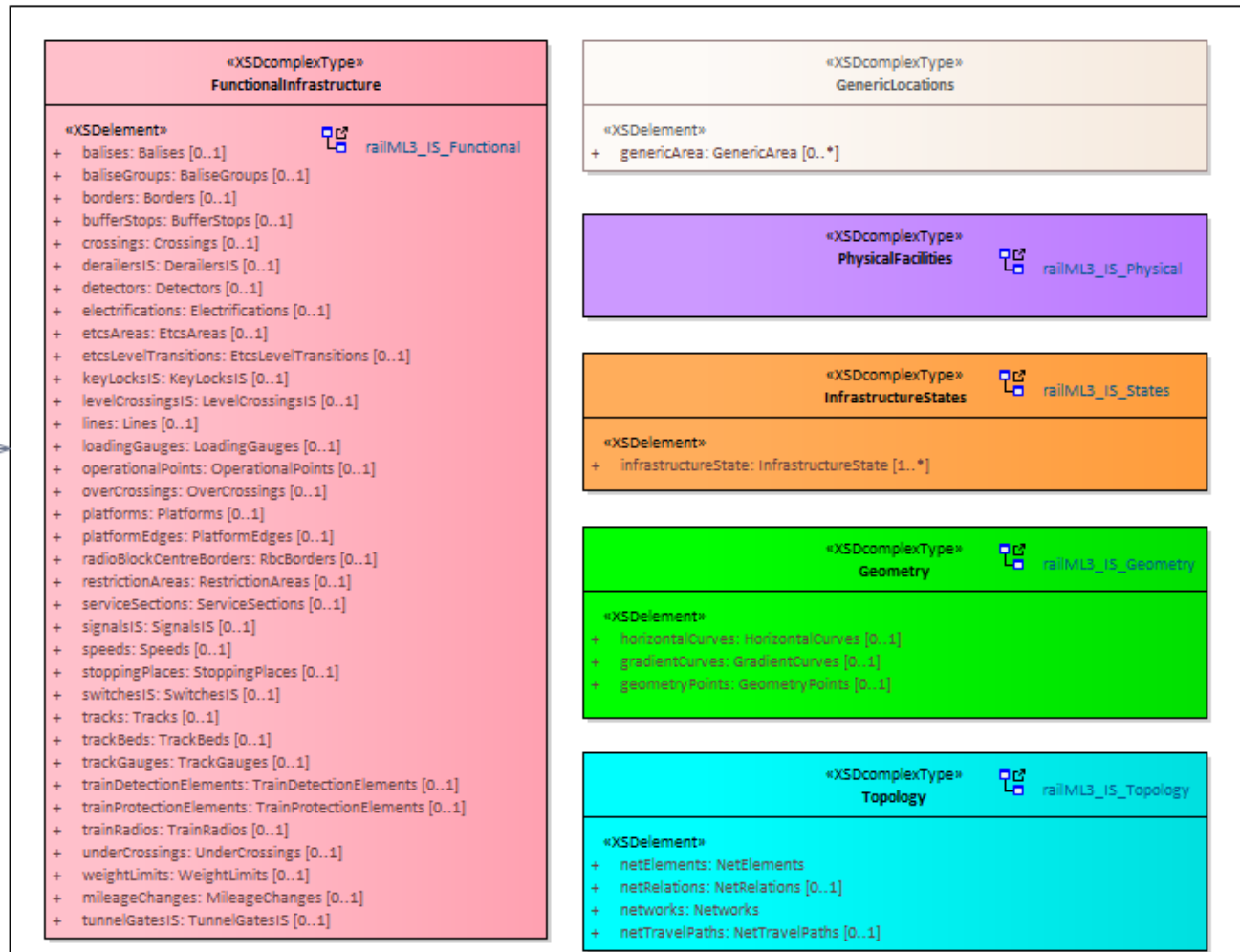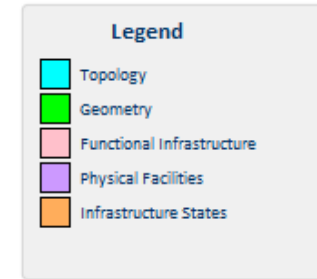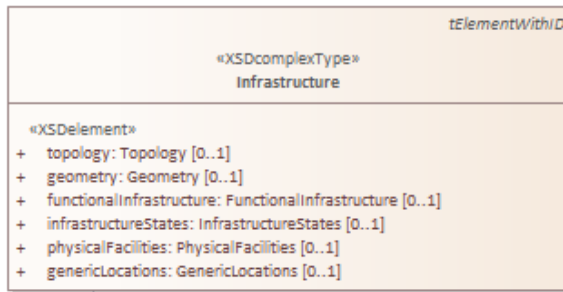# Modelling Infrastructure

Christian Rahmig  > railML 3.2 IS  > 26.04.2022 (41st railML Conference)

# Agenda

1. ETCS related modelling (infrastructure)
2. Balises and Balise Groups
3. Speeds
4. Tracks
5. Level Crossings
6. Switches and Crossings
7. Platforms
8. Topology
9. Line, Operational Point, Mileage Changes
10. Other infrastructure
11. Visualization

# Modeling

**tElementWithID**

«XSDcomplexType»
**Infrastructure**

«XSDelement»
+ topology: Topology [0..1]
+ geometry: Geometry [0..1]
+ functionalInfrastructure: FunctionalInfrastructure [0..1]
+ infrastructureStates: InfrastructureStates [0..1]
+ physicalFacilities: PhysicalFacilities [0..1]
+ genericLocations: GenericLocations [0..1]

**Legend**
- Topology
- Geometry
- Functional Infrastructure
- Physical Facilities
- Infrastructure States

«use»

«XSDcomplexType»
**FunctionalInfrastructure**

«XSDelement»                    railML3_IS_Functional
+ balises: Balises [0..1]
+ baliseGroups: BaliseGroups [0..1]
+ borders: Borders [0..1]
+ bufferStops: BufferStops [0..1]
+ crossings: Crossings [0..1]
+ derailersIS: DerailersIS [0..1]
+ detectors: Detectors [0..1]
+ electrifications: Electrifications [0..1]
+ etcsAreas: EtcsAreas [0..1]
+ etcsLevelTransitions: EtcsLevelTransitions [0..1]
+ keyLocksIS: KeyLocksIS [0..1]
+ levelCrossingsIS: LevelCrossingsIS [0..1]
+ lines: Lines [0..1]
+ loadingGauges: LoadingGauges [0..1]
+ operationalPoints: OperationalPoints [0..1]
+ overCrossings: OverCrossings [0..1]
+ platforms: Platforms [0..1]
+ platformEdges: PlatformEdges [0..1]
+ radioBlockCentreBorders: RbcBorders [0..1]
+ restrictionAreas: RestrictionAreas [0..1]
+ serviceSections: ServiceSections [0..1]
+ signalsIS: SignalsIS [0..1]
+ speeds: Speeds [0..1]
+ stoppingPlaces: StoppingPlaces [0..1]
+ switchesIS: SwitchesIS [0..1]
+ tracks: Tracks [0..1]
+ trackBeds: TrackBeds [0..1]
+ trackGauges: TrackGauges [0..1]
+ trainDetectionElements: TrainDetectionElements [0..1]
+ trainProtectionElements: TrainProtectionElements [0..1]
+ trainRadios: TrainRadios [0..1]
+ underCrossings: UnderCrossings [0..1]
+ weightLimits: WeightLimits [0..1]
+ mileageChanges: MileageChanges [0..1]
+ tunnelGatesIS: TunnelGatesIS [0..1]

«XSDcomplexType»
**GenericLocations**

«XSDelement»
+ genericArea: GenericArea [0..*]

«XSDcomplexType»
**PhysicalFacilities**                    railML3_IS_Physical

«XSDcomplexType»
**InfrastructureStates**                    railML3_IS_States

«XSDelement»
+ infrastructureState: InfrastructureState [1..*]

«XSDcomplexType»
**Geometry**                    railML3_IS_Geometry

«XSDelement»
+ horizontalCurves: HorizontalCurves [0..1]
+ gradientCurves: GradientCurves [0..1]
+ geometryPoints: GeometryPoints [0..1]

«XSDcomplexType»
**Topology**                    railML3_IS_Topology

«XSDelement»
+ netElements: NetElements
+ netRelations: NetRelations [0..1]
+ networks: Networks
+ netTravelPaths: NetTravelPaths [0..1]

railML.org

# ETCS related modeling

# #365: Extending enumeration for track condition areas

- The situation:
  - Current enumeration values of <restrictionArea>@type are not sufficient to cover all types of track condition areas as defined in ETCS SUBSET-026

- Idea:
  - Extend **<restrictionArea>@type** with new values „soundHorn", „tunnelStoppingArea", „changeTractionSystem", „changeAllowedCurrentConsumption", „bigMetalMasses"
  - General: only values from ETCS SUBSET-026 section 3.12.1.3 shall be added to the enumeration; further values may be put in an own extension

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=688&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/365 |
| Wiki: https://wiki3.railml.org/wiki/IS:restrictionArea/3.2 |

# #365: Extending enumeration for track condition areas

- The model:



tOtherEnumerationValue

«XSDunion»
tRestrictionAreaTypeExt

Further specific types like "avalancheSection" and "sandStormSection" can be added as extension via otherEnumerationValue.

string

«enumeration»
tRestrictionAreaType

airTightness
bigMetalMasses
changeAllowedCurrentConsumption
changeTractionSystem
lowerPantograph
mainPowerSwitchOff
noEddyCurrentBrakeForEmergencyBrake
noEddyCurrentBrakeForServiceBrake
noMagneticShoeBrake
noRegenerativeBrake
noSanding
noStopping
radioHole
soundHorn
tunnelStoppingArea

«XSDcomplexType»
RestrictionAreas

+restrictionArea

1          1..*

FunctionalInfrastructureEntity

«XSDcomplexType»
RestrictionArea

«XSDattribute»
+    type: tRestrictionAreaTypeExt

1) noSanding is not included in SUBSET-026 3.12.1.3; 2) platform is missing in list; 3) shall we use child element <etcs>

# #386: Radio Block Center

- Situation:
  - railML 3.1 data model is missing the Radio Block Centre (RBC) required by ETCS related applications

- Solution:
  - New element **\<radioBlockCentre>** in interlocking
  - New element **\<radioBlockCentreBorder>** in infrastructure

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=727&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/386 |
| Wiki: |

# #386: Radio Block Center

- The model:
  - <radioBlockCentreBorder>

Christian Rahmig  > railML 3.2 IS  > 26.04.2022 (41st railML Conference)
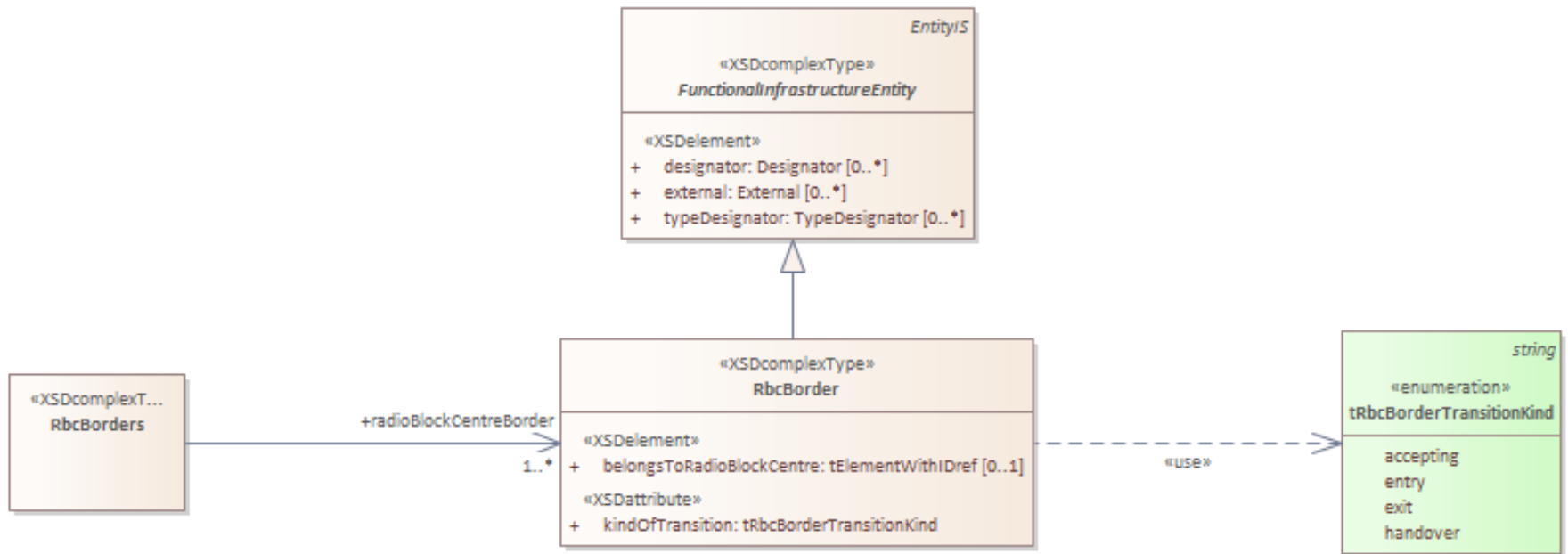
# #439: NID_CTRACTION for electrification model

- The situation:
  - railML 3.1 electrification model misses parameters to unambiguously derive ETCS variable NID_CTRACTION
- Idea:
  - Extend <electrificationSection> with child element **<etcsElectrification>**
    - Add **@nid_ctraction** (non-negative integer) to explicitly model ETCS variable values
    - Add **@mVersion** (non-negative integer) to specify the ETCS version (M_VERSION)

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=732&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/439 |
| Wiki: https://wiki3.railml.org/wiki/IS:electrificationSection |

# #439: NID_CTRACTION for electrification model

- The model:
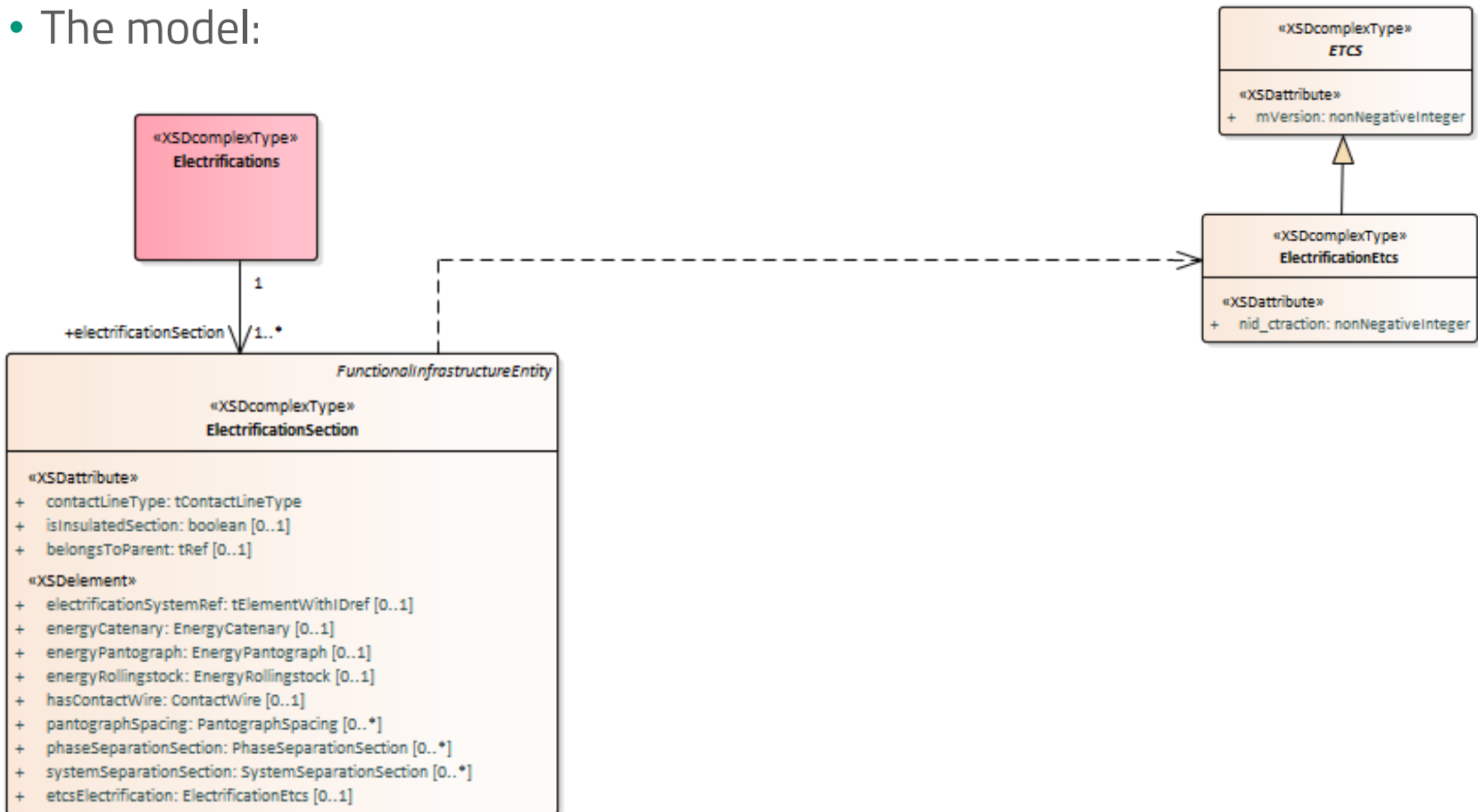
# #459: ETCS signal modeling update

- The situation:
  - ETCS SRS version number is implemented for signal: <signalIS / isEtcsSignal> @srsVersion, but seems to be not used / not needed

- Idea / solution:
  - DEPRECATE the not used attribute <signalIS / isEtcsSignal> **@srsVersion**
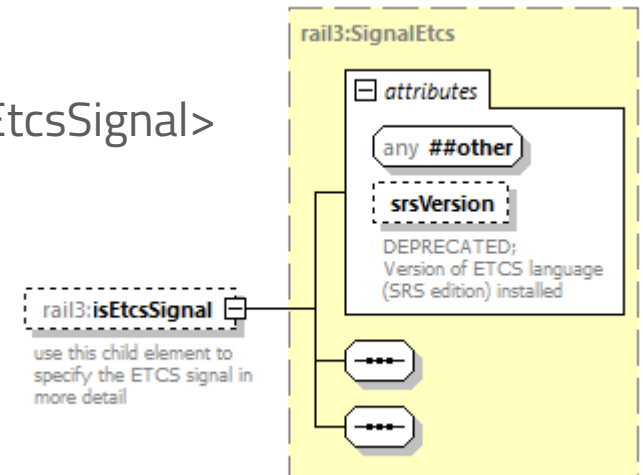
# #460: TrainProtectionElement vs ETCS

- The situation:
  - It is unclear if <trainProtectionElement> shall be used for ETCS based systems

- Solution:
  - Clarification: ***<trainProtectionElement> shall only be used for national and/or legacy train protection systems. ETCS based systems must not be modelled using <trainProtectionElement>.***

| Links |
|---|
| Forum: |
| Trac: https://development.railml.org/railml/version3/-/issues/460 |
| Wiki: https://wiki3.railml.org/wiki/IS:trainProtectionElement |

**rail**ML**.org**

# #504: ETCS / NTC level transitions

- Situation:
  - ETCS level transition (destination level) was at first defined in a combined variable <etcsLevelTransition / switchToLevel>@value (string)
  - Better to separate this information into two attributes

- Idea / solution:
  - Add new attributes in <etcsLevelTransition / switchToLevel>
    - **@levelType** (ETCS, NTC)
    - **@levelValue** (positive integer)

| Links |
|---|
| Forum: |
| Trac: https://development.railml.org/railml/version3/-/issues/504 |
| Wiki: |

# #504: ETCS / NTC level transitions

- The model:
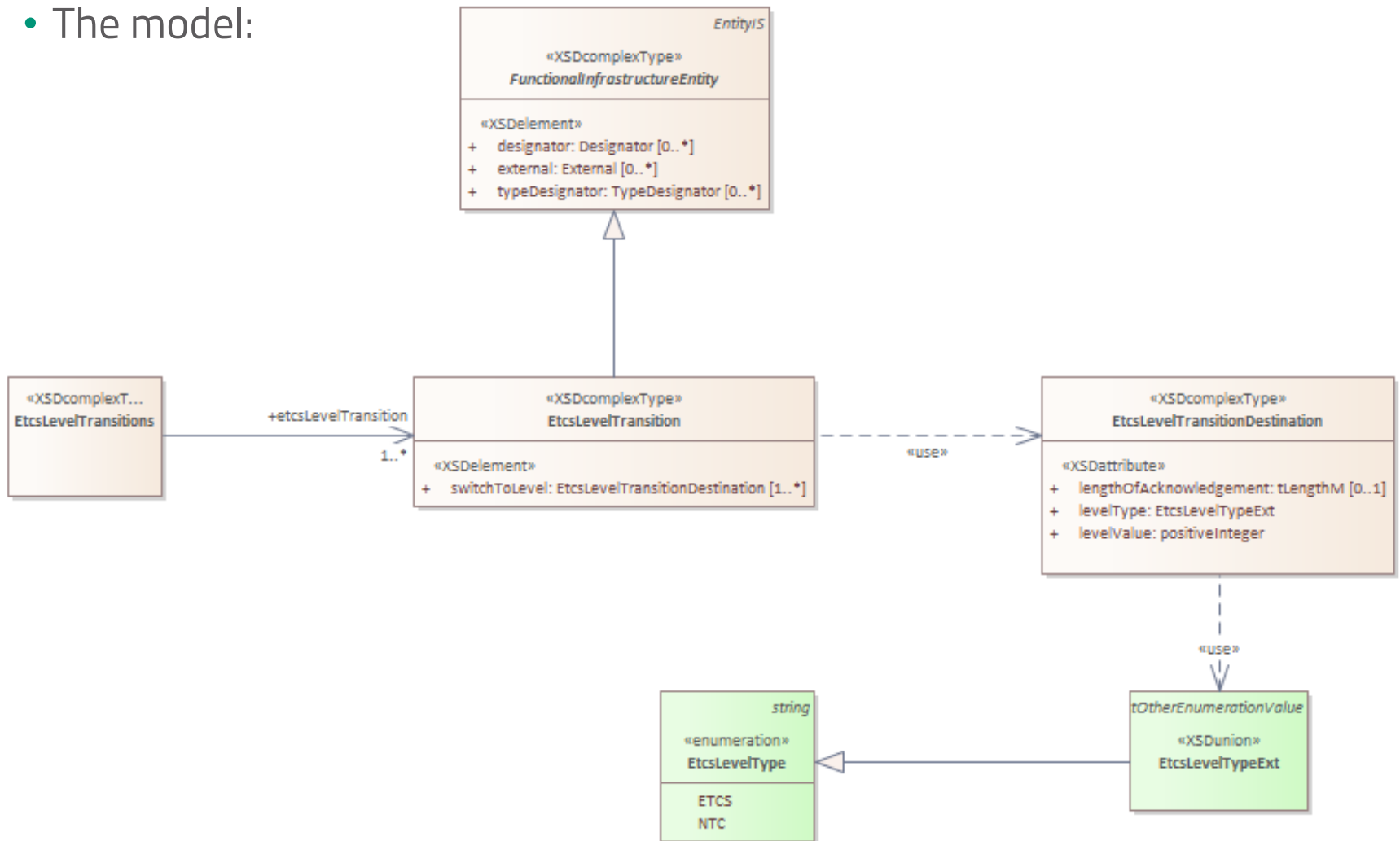
# Balises and BaliseGroups

# #366: Extending the <balise> element

- The situation:
  - railML 3.1 implementation of <balise> is insufficient w.r.t. parameters required by ETCS specification (ETCS SUBSET-026).

- Ideas:
  - Differentiate between **<balise>** and **<baliseGroup>**
    - DEPRECATE <balise>@belongsToParent
    - DEPRECATE <balise>@isBaliseGroup
    - DEPRECATE <balise>@baliseGroupType

# #366: Extending the <balise> element

- Ideas:
  - For **<balise>**:
    - Rename balise type „transparent" into „controlled"
    - Add Eurobalise as specific type of balise: **<balise / isEurobalise>**
      - Add ETCS version: **<balise / isEurobalise>@mVersion** (non-negative integer)
    - Add attributes for identification of a balise in a balise group,
      - **@distanceToPredecessorBaliseWithinGroup**
      - **@belongsToBaliseGroup**
      - **<isEurobalise>@positionInGroup**
      - **<isEurobalise>@duplicate**

# #366: Extending the <balise> element

- Ideas:
  - For **<baliseGroup>**:
    - Add Eurobalise group: **<baliseGroup/isEurobaliseGroup>**
      - Add new attributes for describing Eurobalise group linking reactions: **@linkReactionNominal**, **@linkReactionReverse** (trainTrip, applyServiceBrake, noReaction) and **@isLinked** (bool)
      - Add new attribute **@locationAccuracy** (decimal; -63..63 Meter)
      - Add ETCS related information: **@countryID** (integer, 0..1023; NID_C), **@groupID** (integer, 0..16383; NID_BG), **@usesPackage44** (integer, 0..511; NID_XUSER), **@virtualCoverageID** (integer, 0..63, NID_VBCMK) and **@mVersion** (non-negative integer, M_VERSION)
    - Add child element **<baliseGroup/applicationType>** → ETCS, GNT, NTC…
    - Add child element **<baliseGroup/functionalType>** → announcement, border, handover, … (direction dependent!)
    - Add new attribute **@coverage** (physical, virtual, both, none)
    - Add new attribute **@numberOfBalisesInGroup** (positive integer)
    - Add (repeatable) **<connectedWithInfrastructureElement>** for physical and logical connections between a balise group and other infrastrtucture

# #366: Extending the <balise> element

- Ideas:
  - Add reference from signal to (protecting) balise, e.g.
    **<signalIS>@protectedByBaliseGroup**

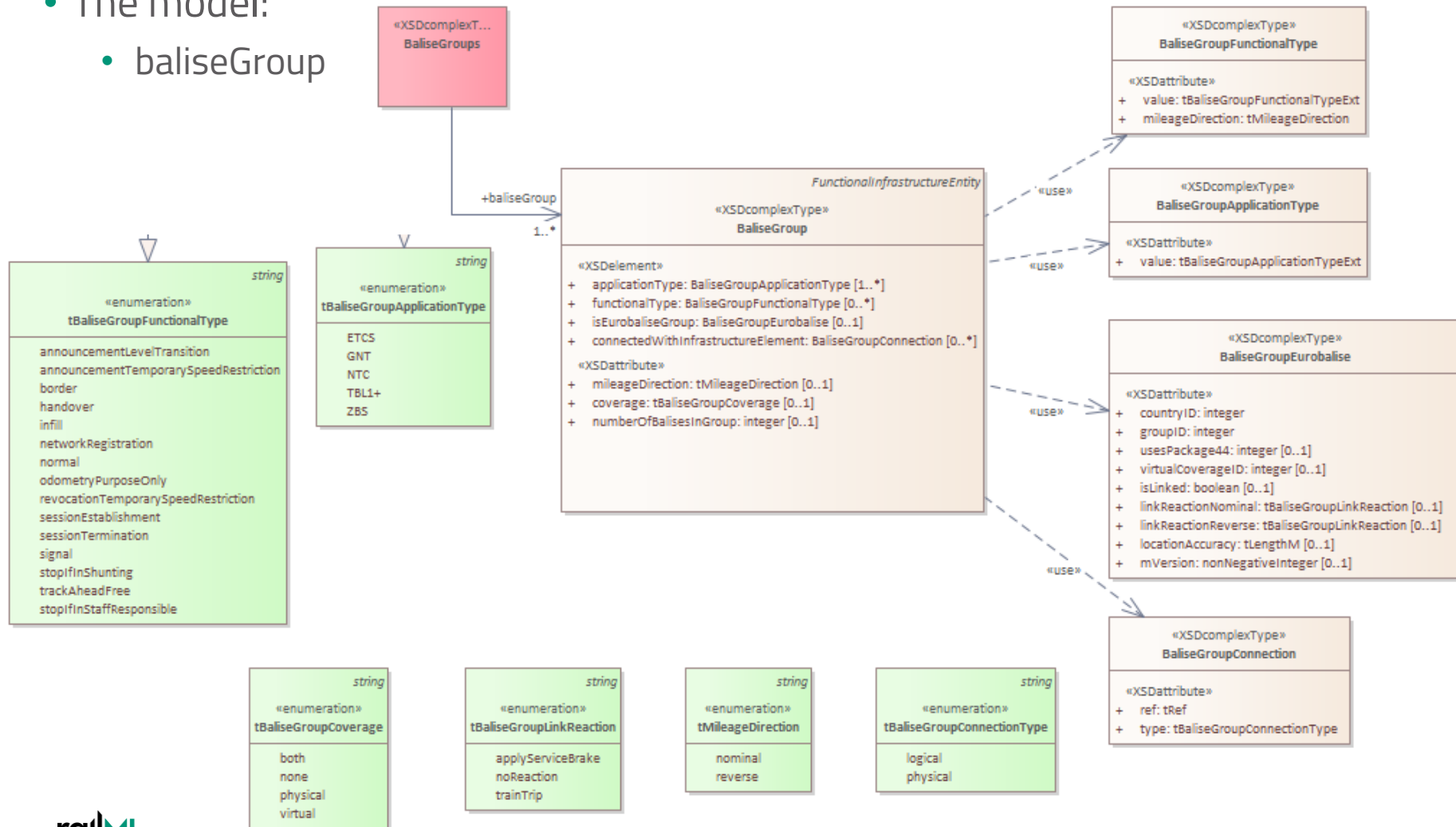| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=687&start=0&; https://www.railml.org/forum/index.php?t=msg&th=135&goto=513&#msg_513; https://www.railml.org/forum/index.php?t=msg&th=651&goto=2140&#msg_2140; https://www.railml.org/forum/index.php?t=msg&th=725&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/366; https://trac.railml.org/ticket/174 |
| Wiki: https://wiki3.railml.org/wiki/IS:balise; https://wiki3.railml.org/wiki/IS:baliseGroup https://wiki3.railml.org/wiki/IS:signalIS |

# #366:

- The model:
  - balise



«XSDcomplexType»
**Balises**

*FunctionalInfrastructureEntity*

+balise

0..*

«XSDcomplexType»
**Balise**

«XSDattribute»
+ type: tBaliseType [0..1]
+ belongsToBaliseGroup: tRef [0..1]
+ distanceToPredecessorBaliseWithinGroup: tLengthM [0..1]
+ basedOnTemplate: tRef [0..1]
+ belongsToParent: tRef [0..1]
+ isBaliseGroup: boolean [0..1]
+ baliseGroupType: tBaliseGroupTypeExt [0..1]

«XSDelement»
+ isEurobalise: BaliseEurobalise [0..1]

*tOtherEnumerationValue*

«XSDunion»
**tBaliseGroupTypeExt**

DEPRECATED

*string*

«enumeration»
**tBaliseGroupType**

fixed
infill
signal

*string*

«enumeration»
**tBaliseType**

controlled
fixed
transparent

«use»

«XSDcomplexType»
**BaliseEurobalise**

«XSDattribute»
+ positionInGroup: integer [0..1]
+ duplicate: tBaliseDuplicate [0..1]
+ mVersion: nonNegativeInteger [0..1]

*string*

«enumeration»
**tBaliseDuplicate**

no
ofNextBalise
ofPreviousBalise

# #366: Extending the <balise> element

- The model:
  - baliseGroup

# Speeds

# #367: Extending the <speedProfile> element

- The situation: railML 3.1 implementation of speed profiles is insufficient w.r.t. parameters required by ETCS specification.

- Ideas:

  - Add boolean flag to identify basic speed profiles **@isBasicSpeedProfile**

  - Add new attribute to specify the maximum allowed cant deficiency: **@maxCantDeficiency** (integer, 80..300)

  - Adapt enumeration values of attribute **<trainType>@type** to cover „mixed" and „all" trains; deprecate „tiltingPassenger"

  - Leading parameters of speed profile: train type, air brake application position, maximum cant deficiency (derive ETCS train category number)

  - Change cardinality of <trainType> from 0..1 to 0..*

  - Add new child element <**trainType** / **etcsSpeedProfile**> with attribue **@etcsTrainCategoryNumber** (derived information!)

  - Deprecate **<trainType>**@etcsTrainCategoryNumber and **<trainType>**@cantDeficiency

**rail**ML.org

# #367: Extending the <speedProfile> element

• The model (1/2):
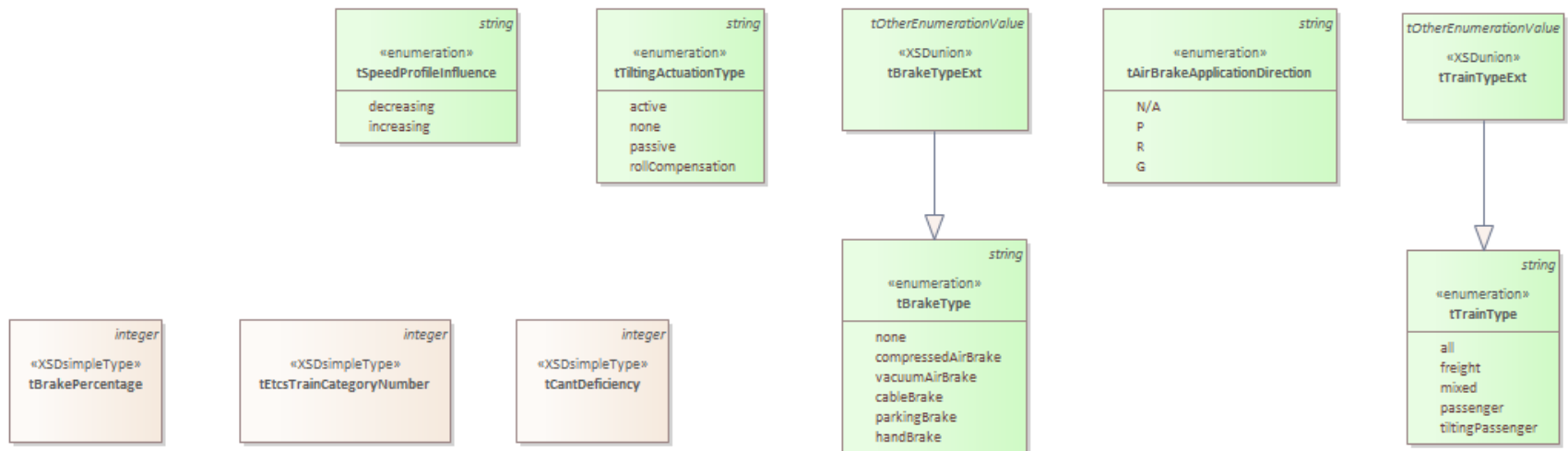


| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=627&goto=2053&#msg_2053; https://www.railml.org/forum/index.php?t=msg&th=686&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/367 |
| Wiki: https://wiki3.railml.org/wiki/CO:speedProfile |

# #367: Extending the <speedProfile> element

- The model (2/2):
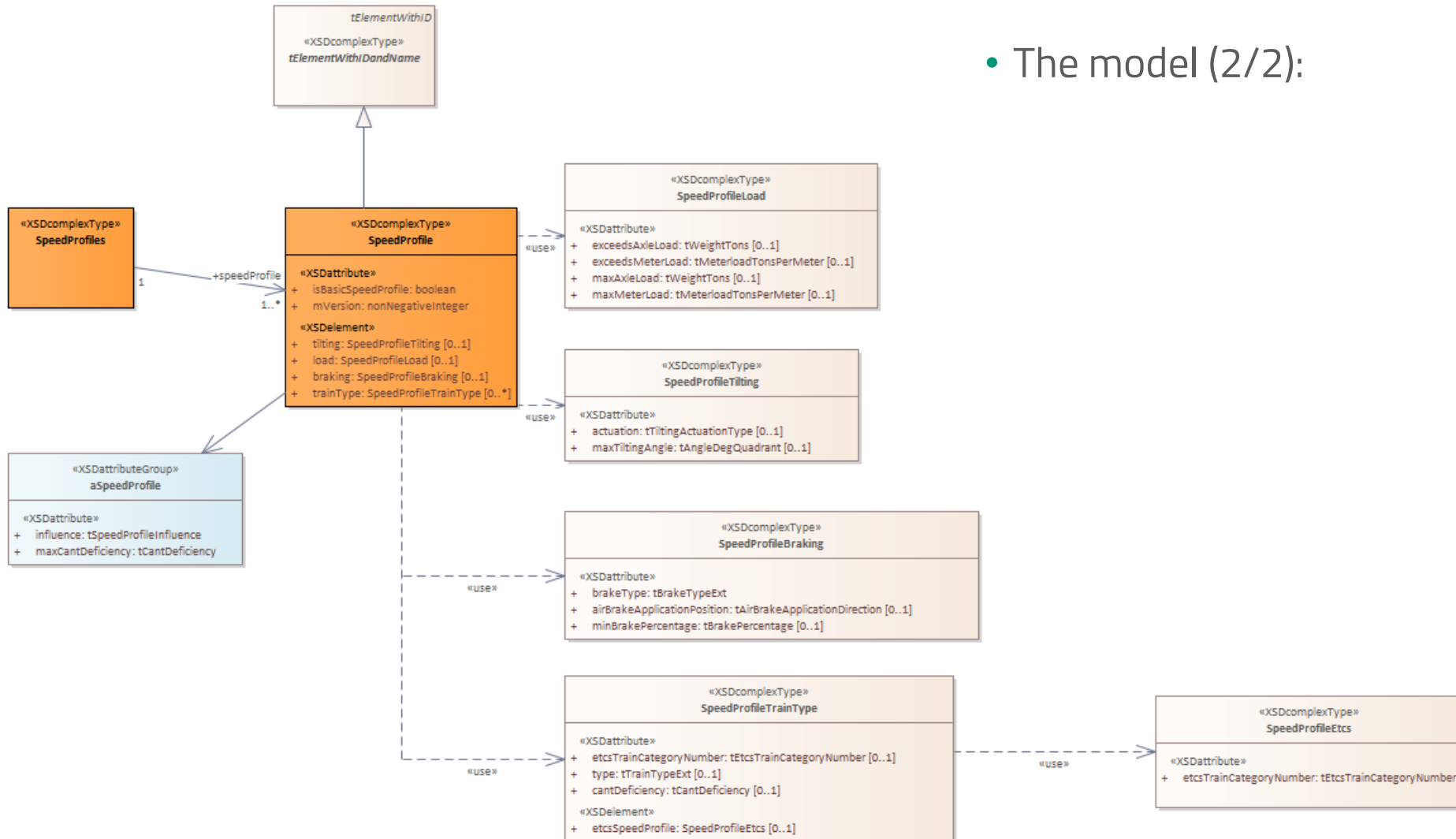
# #496: Reference to train from speedSection

- The task:
  - It is necessary to have a reference from a speedSection to the part of the train, for which the speed is valid.

- Solution:
  - In element **<speedSection>** new attribute **@refersToTrain** (tTrainRelation) shall be added
  - In element **<signalIS><isSpeedSignal>** the attribute **@trainRelation** (tTrainRelation) shall be marked DEPRECATED
  - Enum value "midOfTrain" shall remain

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=840&goto=2858&#msg_2858 |
| Trac: https://development.railml.org/railml/version3/-/issues/496 |
| Wiki: https://wiki3.railml.org/wiki/IS:speedSection |

# #496: Reference to train from speedSection

- The model:
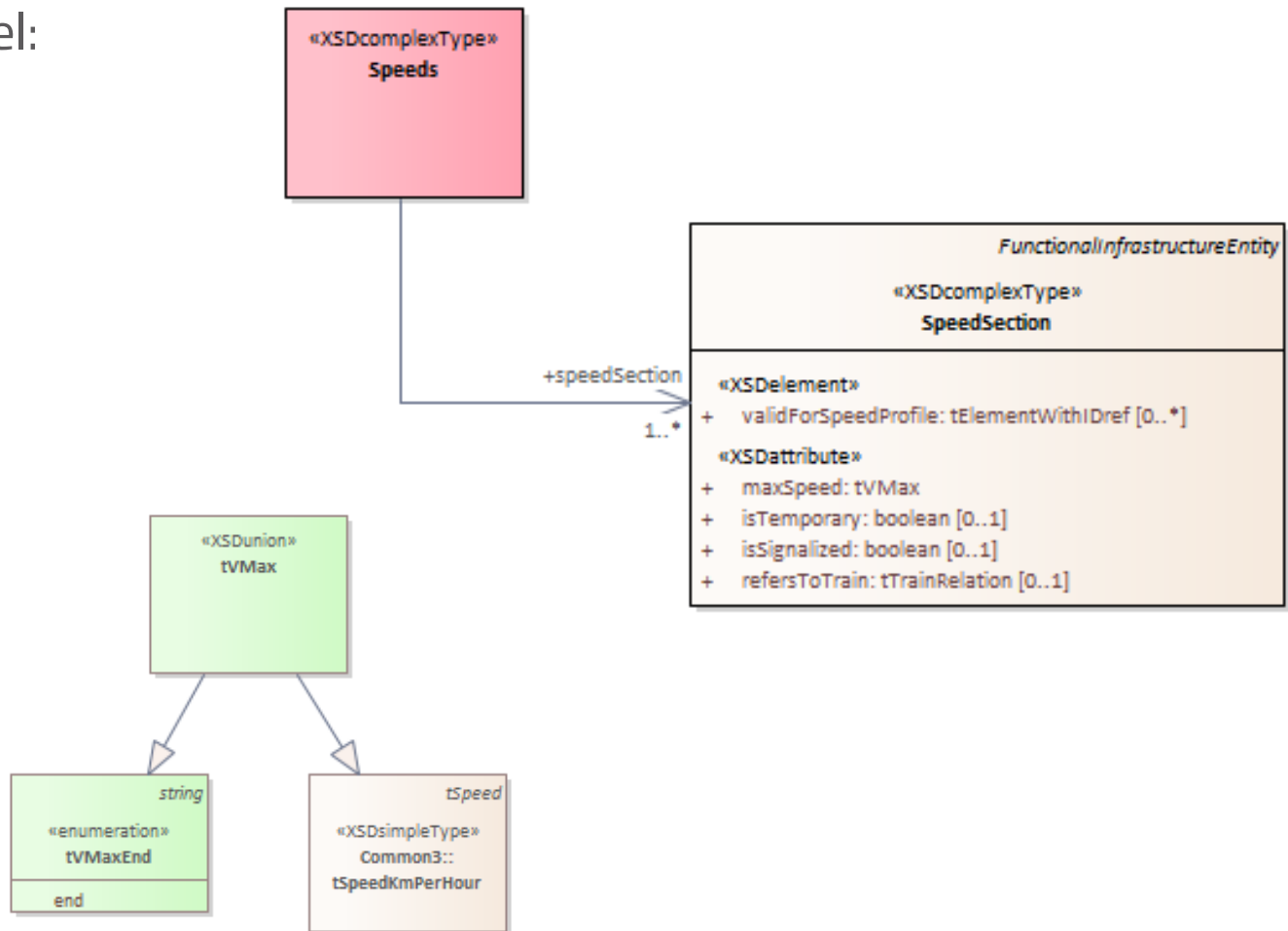
# #501: Move attribute @mVersion in speedProfile

- The situation:
  - The element <speedProfile> is missing an attribute for defining the ETCS version. Instead, the child element <speedProfile / trainType / etcsSpeedProfile> has it (@mVersion)

- Idea / Solution:
  - Attribute **@mVersion** shall be moved from <speedProfile / trainType / etcsSpeedProfile> to parent element **<speedProfile>**

# #502: Axle load related speed profiles

- The situation:
  - The definition of axle load related speed profiles is not clear. Therefore, a renaming and extension of attributes in the related speed profile sub type is required.

- Solution:
  - Rename attributes in element **\<speedProfile\>\<load\>**
    - @maxAxleLoad into @exceedsAxleLoad
    - @maxMeterLoad into @exceedsMeterLoad

| Links |
|---|
| Forum: |
| Trac: https://development.railml.org/railml/version3/-/issues/502 |
| Wiki: |

# #502: Axle load related speed profiles

- The model:



tElementWithID
«XSDcomplexType»
tElementWithIDandName

«XSDcomplexType»
**SpeedProfiles**

«XSDcomplexType»
**SpeedProfile**

«XSDattribute»
+ isBasicSpeedProfile: boolean
+ mVersion: nonNegativeInteger

«XSDelement»
+ tilting: SpeedProfileTilting [0..1]
+ load: SpeedProfileLoad [0..1]
+ braking: SpeedProfileBraking [0..1]
+ trainType: SpeedProfileTrainType [0..*]

+speedProfile

1    1..*

«XSDcomplexType»
**SpeedProfileLoad**

«XSDattribute»
+ exceedsAxleLoad: tWeightTons [0..1]
+ exceedsMeterLoad: tMeterloadTonsPerMeter [0..1]
+ maxAxleLoad: tWeightTons [0..1]
+ maxMeterLoad: tMeterloadTonsPerMeter [0..1]

«use»

«XSDcomplexType»
**SpeedProfileTilting**

«XSDattribute»
+ actuation: tTiltingActuationType [0..1]
+ maxTiltingAngle: tAngleDegQuadrant [0..1]

«use»

«XSDattributeGroup»
aSpeedProfile

«XSDattribute»
+ influence: tSpeedProfileInfluence
+ maxCantDeficiency: tCantDeficiency

«XSDcomplexType»
**SpeedProfileBraking**

«XSDattribute»
+ brakeType: tBrakeTypeExt
+ airBrakeApplicationPosition: tAirBrakeApplicationDirection [0..1]
+ minBrakePercentage: tBrakePercentage [0..1]

«use»

# Tracks

# #368: Definition of a Track

- The situation:
  - The current definition of a track is very strict: „A Track is defined by a railway section between two switches/crossings or between a switch/crossing and a buffer stop. "

- Idea:
  - allow for more flexible definition of a <track> in order to allow for very short tracks and very long tracks

# #368: Definition of a Track

- The situation:
  - The current definition of a track is very strict: „A Track is defined by a railway section between two switches/crossings or between a switch/crossing and a buffer stop. "

- Idea:
  - allow for more flexible definition of a <track> in order to allow for very short tracks and very long tracks

# #368: Definition of a Track

- The situation:
  - The current definition of a track is very strict: „A Track is defined by a railway section between two switches/crossings or between a switch/crossing and a buffer stop. "

- Idea:
  - allow for more flexible definition of a <track> in order to allow for very short tracks and very long tracks

# #368: Definition of a Track

- The situation:
  - The current definition of a track is very strict: „A Track is defined by a railway section between two switches/crossings or between a switch/crossing and a buffer stop. "
- Idea:
  - allow for more flexible definition of a <track> in order to allow for very short tracks and very long tracks
- Solution:
  - ***Track … is a railway section that can be traversed by a train in a continuous motion*.**

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=684&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/368 |
| Wiki: https://wiki3.railml.org/wiki/IS:track |

# #369: Track length

- The situation:
  - In railML 3.1 each <track> element has to have at least one child element <length>

- Idea:
  - Make **<track><length>** completely optional, because there are use cases (e.g. in timetable) that don't need the length information → change cardinality of <track / length> from 1..* to 0..*

- To be clarified: how about the version downwards compatibility of such a change?

# #369: Track length

- The model:
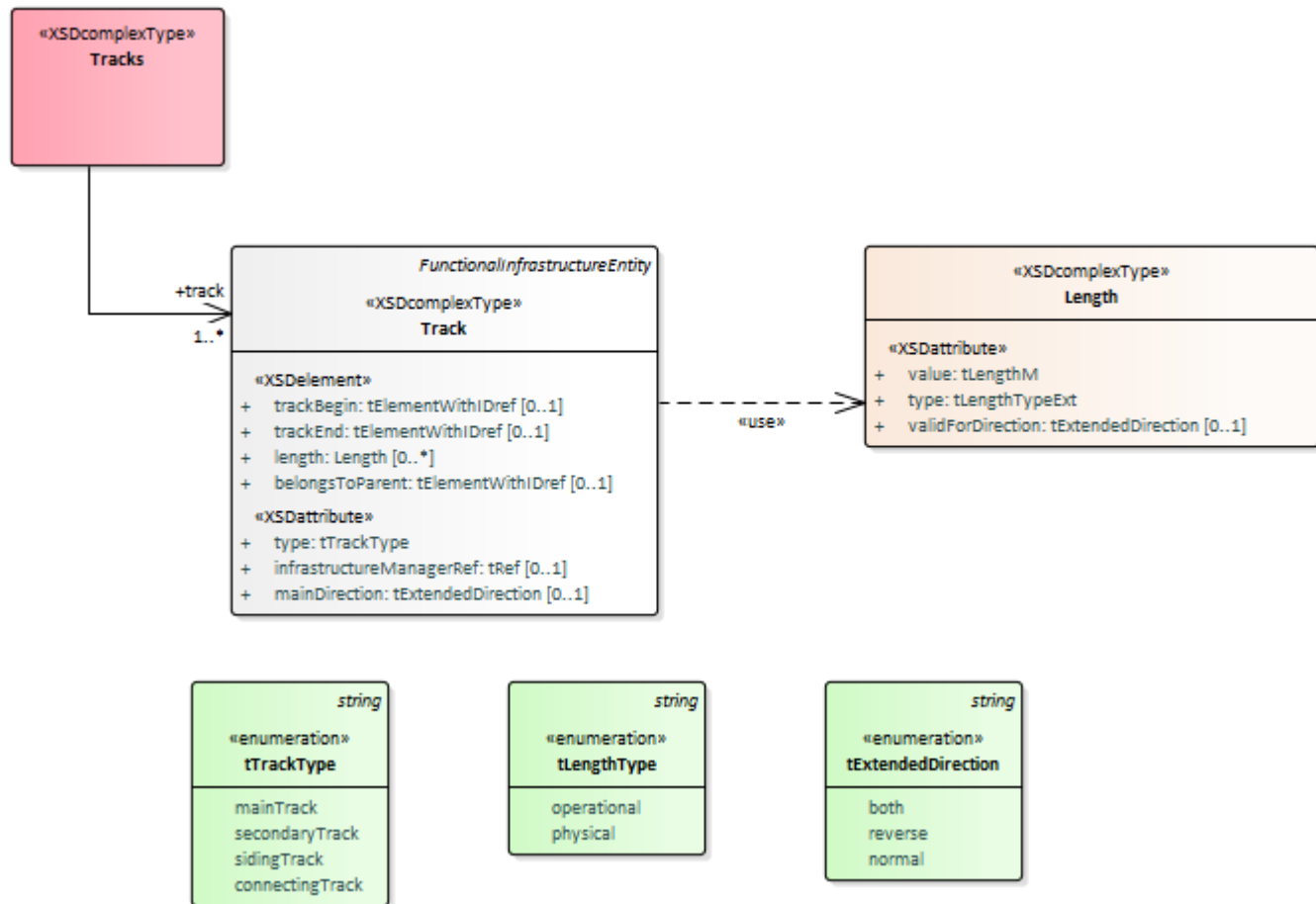
# #475: Make <track> @type optional

- The situation:
  - In railML 3.1 attribute <track>@type is mandatory

- Idea / Solution:
  - Attribute **<track>@type** shall become optional
  - This change will become effective with railML 3.3 (compatibility reasons)

# Level Crossings

# #377: Extending the Level Crossing Model

- The situation:
  - railML 3.1 model of LX is not sufficient to meet ETCS Track Net requirements

- Ideas:
  - Add new child element **<etcsLevelCrossing>** for ETCS related attributes of the Level Crossing
    - Add parameter **@etcsID** that corresponds with ETCS variables NID_TSR or NID_LX
    - Add **@mVersion** (non-negative integer) to specify the ETCS version

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=555&goto=2399&#msg_2399; https://www.railml.org/forum/index.php?t=msg&th=759&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/377 |
| Wiki: https://wiki3.railml.org/wiki/IS:levelCrossingIS |

# #377: Extending the Level Crossing Model

- Ideas (continued):
  - Add element **\<linkedSpeedSection\>** to reference a \<speedSection\> that defines the speed for passing the LX in unprotected mode
  - Add attribute **@lengthOfStoppingAreaBeforeLevelCrossing** to put distance between stopping point in front of LX and LX itself
  - Deprecate \<levelCrossingIL\>@unprotectedSpeed (replaced by linked speedSection information)

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=555&goto=2399&#msg_2399; https://www.railml.org/forum/index.php?t=msg&th=759&start=0& |
| Trac:  https://development.railml.org/railml/version3/-/issues/377 |
| Wiki: https://wiki3.railml.org/wiki/IS:levelCrossingIS |

# #377: Extending the Level Crossing Model

- The model:
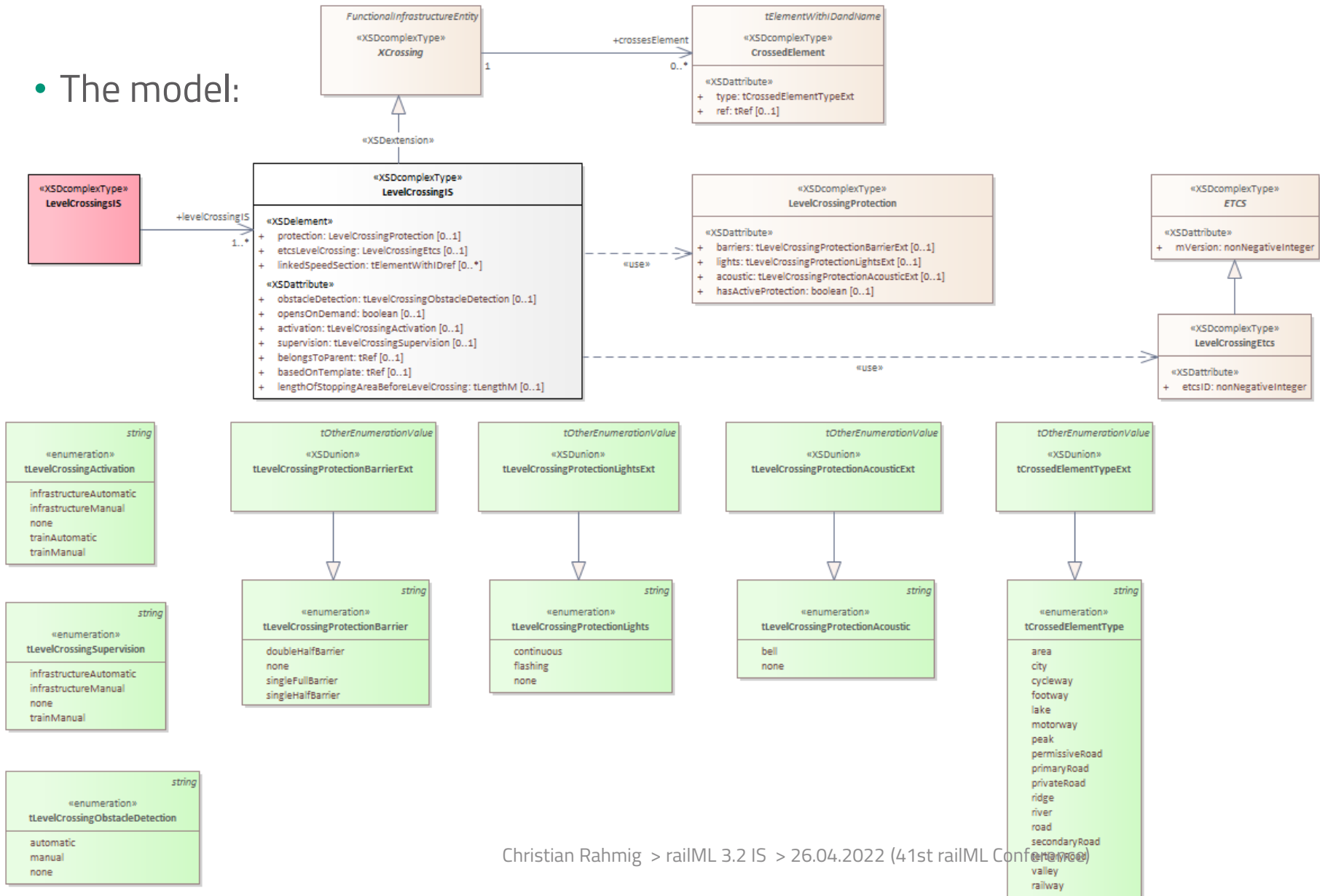
# #377: Extending the Level Crossing Model

- Open issues:
  - More detailed model of barriers (locat... ...relation to railway and road; name)
  - More detailed model of lights (loca... ...t and road; name; signal screen)

**Topics for railML 3.3 …**

# #506: Updating crossed element type

- Situation:
  - Type of crossed element (for bridge, tunnel, level crossing) misses values for specifying different types of roads

- Idea / solution:
  - New values (primaryRoad, secondaryRoad, tertiaryRoad, permissiveRoad, privateRoad) shall be added to existing enum attribute **<levelCrossingIS>@type**.

tOtherEnumerationValue
«XSDunion»
tCrossedElementTypeExt

string
«enumeration»
tCrossedElementType

area
city
cycleway
footway
lake
motorway
peak
permissiveRoad
primaryRoad
privateRoad
ridge
river
road
secondaryRoad
tertiaryRoad
valley
railway

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=874&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/506 |
| Wiki: https://wiki3.railml.org/wiki/IS:crossesElement |

# Switches and Crossings

# #380: Adding branches to crossing

- The situation:
  - railML 3.1 <crossing> element is missing information about its two straight branches

- Idea:
  - Add new child element **<straightBranch>** with cardinality 2 with same parameters like <*Branch> of <switchElement>
    - Reference to topology element <netRelation>: @netRelationRef
    - Length of branch: @length
    - Radius of branch shall be zero (=straight): @radius=„0"
    - Speed along the branch: @branchingSpeed

**rail**ML**.org**

# #380: Adding branches to crossing

- The model:
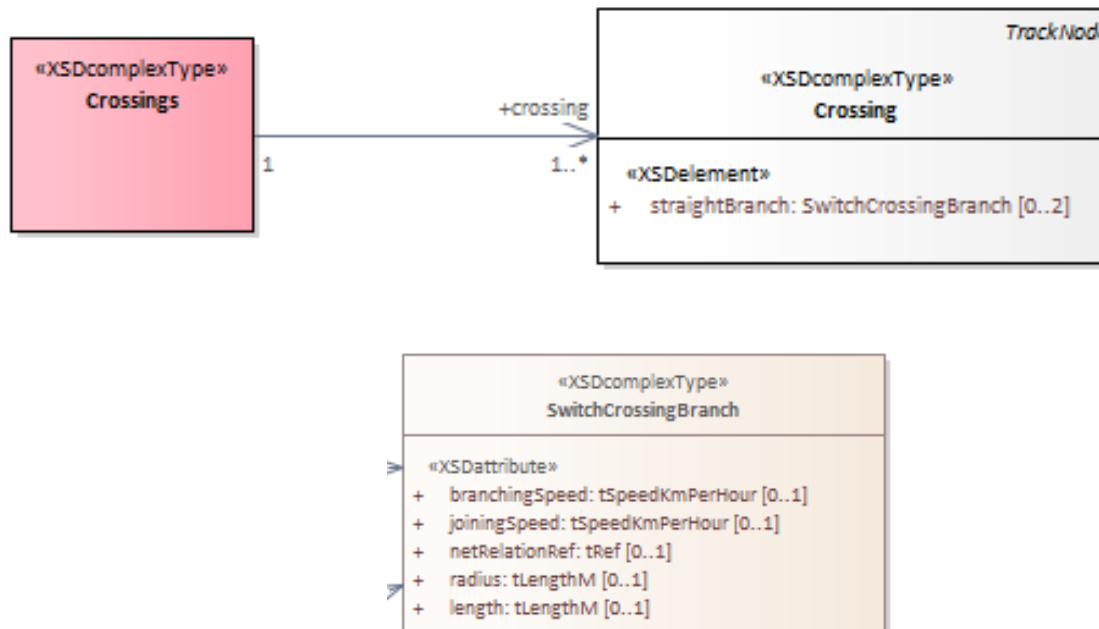
# #484: Modelling switch crossings in infrastructure

- The situation:
  - Sometimes, switch crossings are modelled as composition of simple switches. This modelling approach is currently not supported.

- Idea / Solution:
  - Introduce new **\<switchIS\>@type** value „switchCrossingPart"
  - Add new attribute **\<switchIS\>@belongsToParent** to allow for referencing from a switch crossing part to a switch crossing consisting of these parts

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=839&goto=2855&#msg_2855 |
| Trac:  https://development.railml.org/railml/version3/-/issues/484 |
| Wiki: https://wiki3.railml.org/wiki/IS:switchIS; https://wiki3.railml.org/wiki/IL:switchL |

# #484: Modelling switch crossings in infrastructure

- The model:
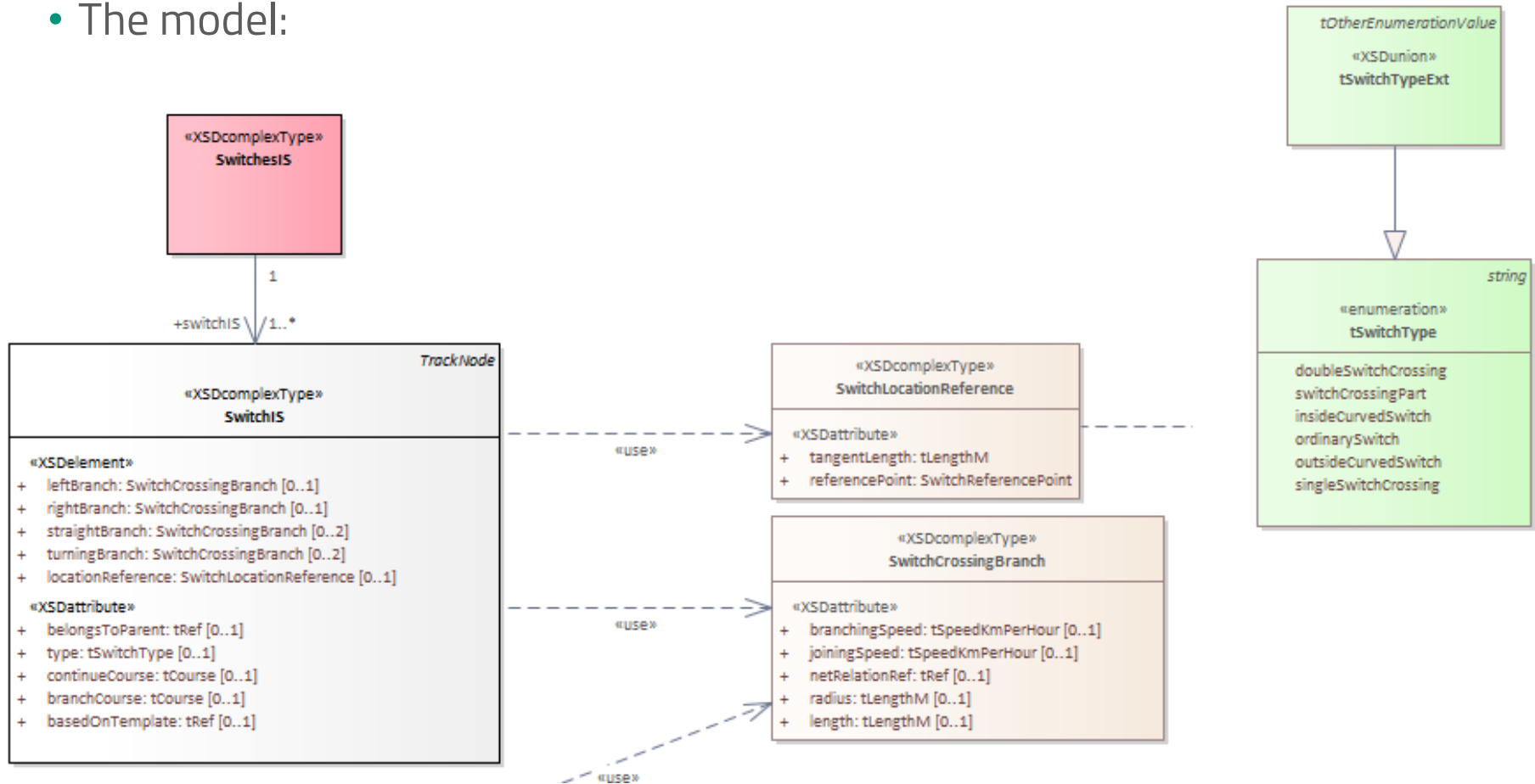
# #493: Switches and their reference points

- The situation:
  - railML 3 switch data model is missing information about switch tangent length that is required to calculate the alternative switch reference point

- Idea / Solution:
  - Introduce new optional child element **\<switchIS\>\<locationReference\>** with attributes **@tangentLength** (tLengthM) to describe the switch tangent length and **@referencePoint** (switchBegin, switchCenter) to define the switch location reference point given by the spot location
  - \<switchIS\>\<locationReference\> shall only be used if the switch is located using a \<spotLocation\>

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=858&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/493 |
| Wiki: https://wiki3.railml.org/wiki/IS:switchIS |

# #493: Switches and their reference points

- The model:

railML.org
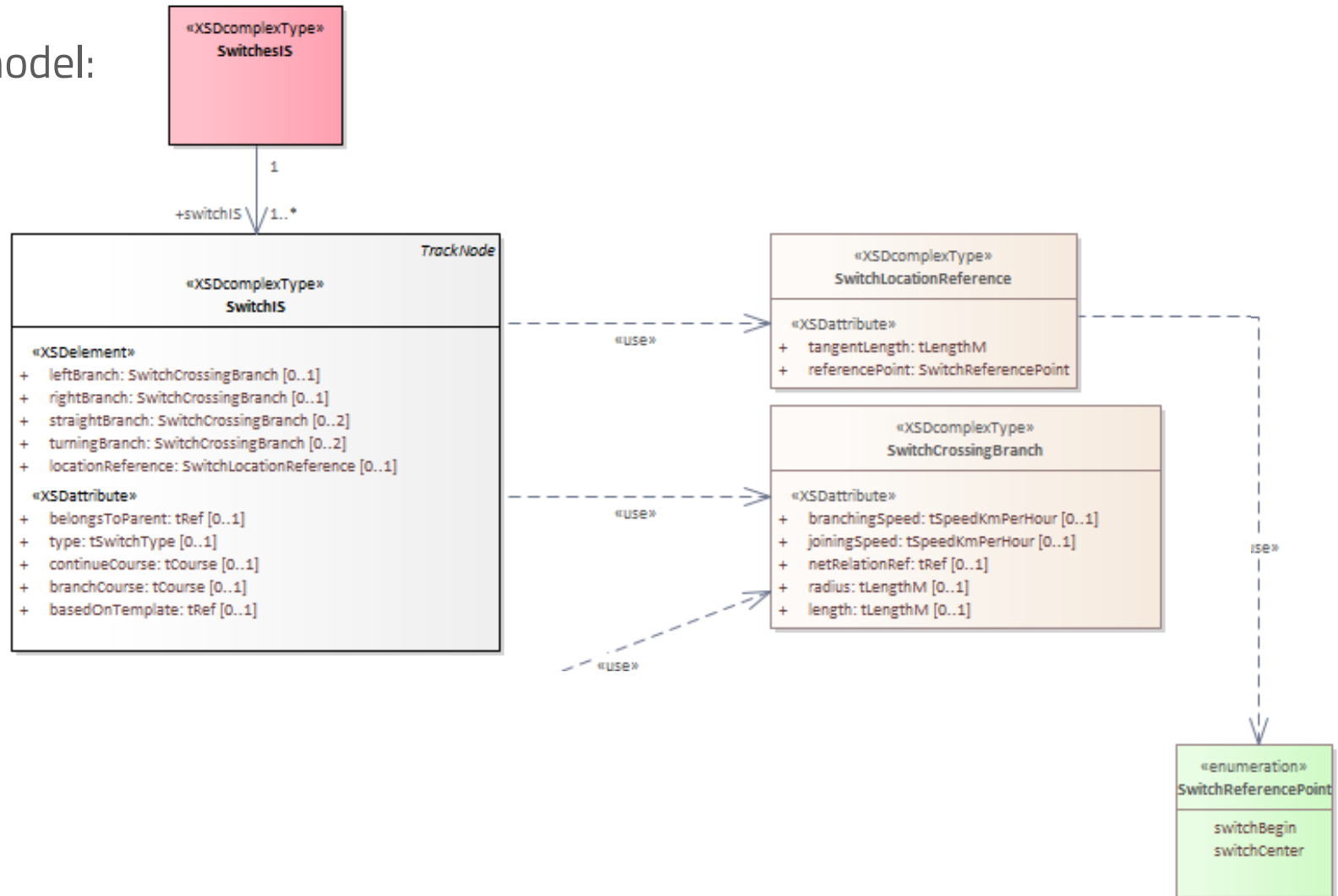
# Platforms, Platform edges and stopping places

# #438: Introduce <platformEdge>

- The situation:
  - railML 3.1 element <platform> is not sufficient to model both, platforms and platform edges
- Ideas:
  - Add new functional infrastructure element **<platformEdge>** (derived from FunctionalInfrastructureEntity) with parameters @belongsToParent, @belongsToPlatform, @height and <length>
  - Deprecate @height and <length> in **<platform>**

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=650&goto=2133&#msg_2133 |
| Trac:  https://development.railml.org/railml/version3/-/issues/438 |
| Wiki: https://wiki3.railml.org/wiki/IS:platform; https://wiki3.railml.org/wiki/IS:platformEdge |

# #438: Introduce <platformEdge>

- The model:
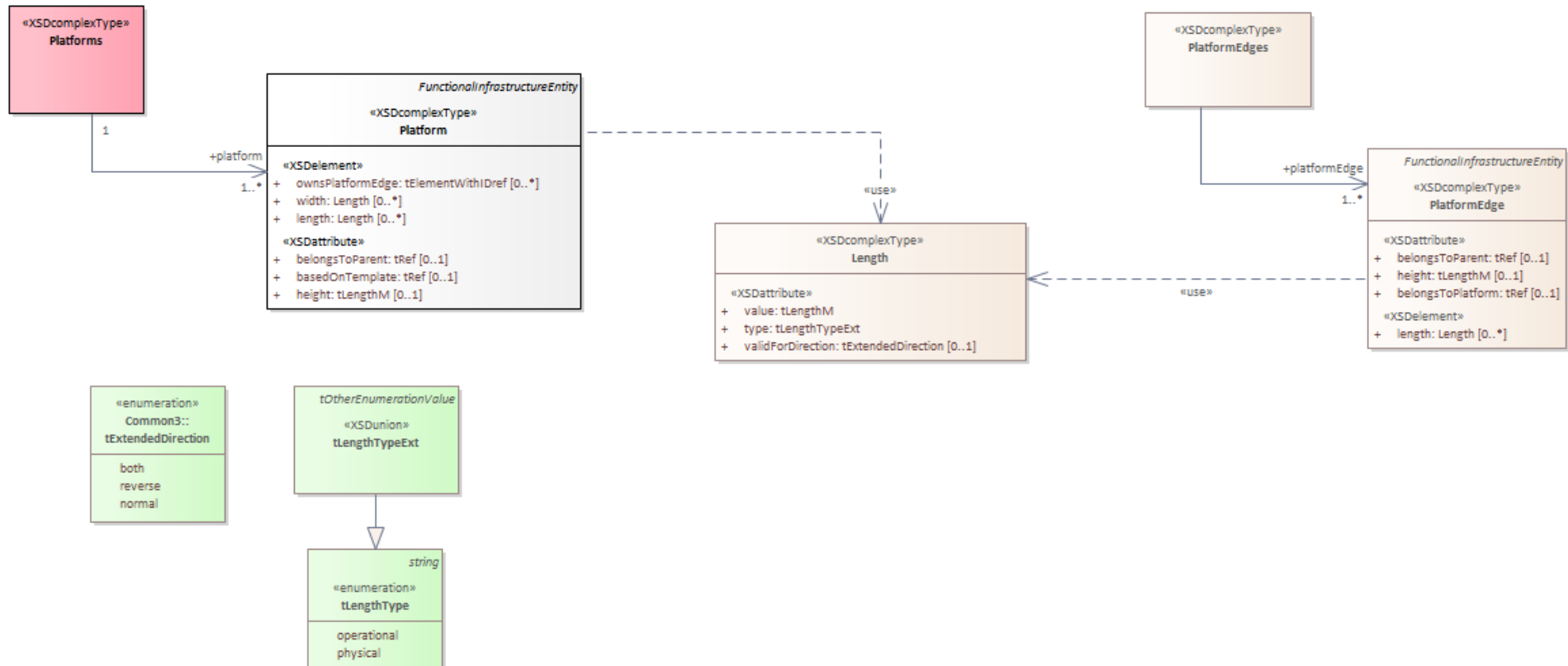
# #454: Stopping places and platform edges

- The situation:
  - In railML 3.1 a <stoppingPlace> can reference only one <platformEdge> with the attribute @platformEdgeRef
  - There is a need for referencing more than one platform edges...

- Idea:
  - Existing attribute <stoppingPlace>@platformEdgeRef shall be marked DEPRECATED
  - A new optional and repeatable child element **<allowsUsageOfPlatformEdge>** shall be introduced to reference a <platformEdge> element

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&goto=2644&#msg_2644 |
| Trac:  https://development.railml.org/railml/version3/-/issues/454 |
| Wiki: https://wiki3.railml.org/wiki/IS:stoppingPlace |

# #454: Stopping places and platform edges

- The model:
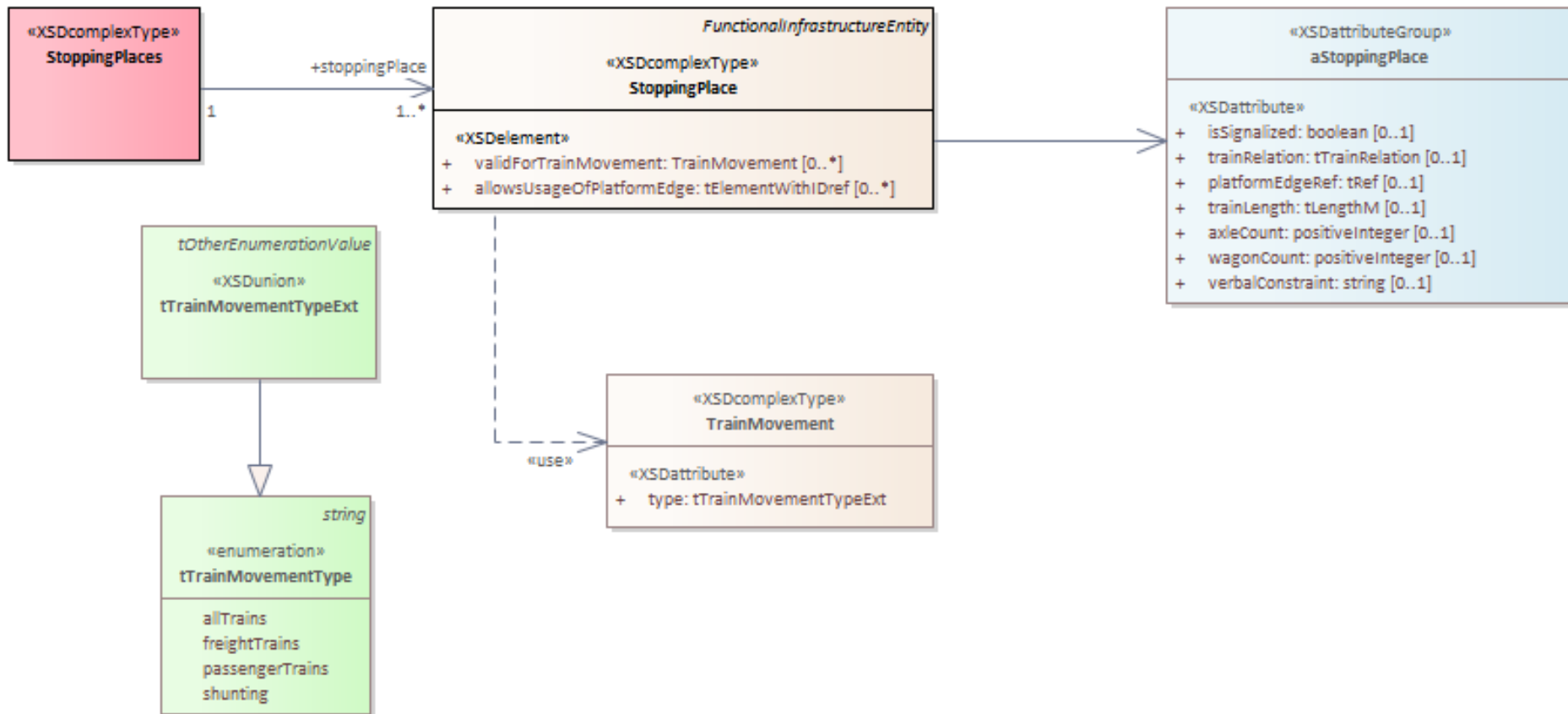
# Topology

# #325: railML data in one vs splitted files

- The situation:
    - In case of big railway networks it may be necessary to cut it into smaller parts that shall be put into separate railML files
    - How to realize the splitting in the data?
- Ideas:
    - Make use of UUIDs to enable element referencing from file externals
    - Realize clear cutting in railML topology layer

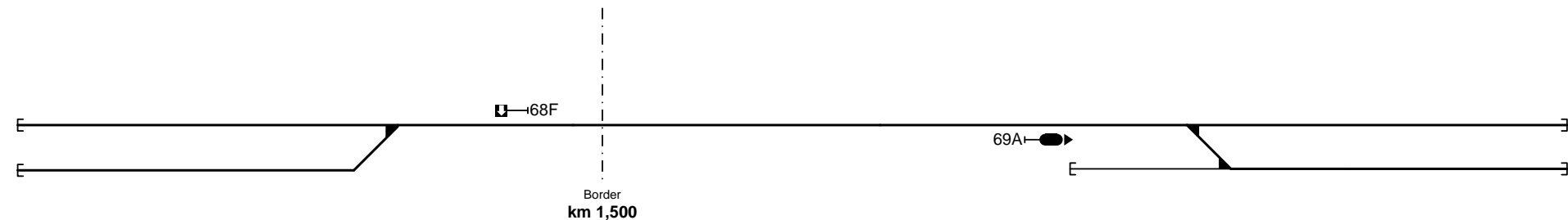| Links |
| --- |
| Forum: https://www.railml.org/forum/index.php?t=msg&th=637&goto=2083&#msg_2083 |
| Trac: https://development.railml.org/railml/version3/-/issues/325; https://development.railml.org/railml/version3/-/issues/363 |
| Wiki: |

# Splitted Infrastructure
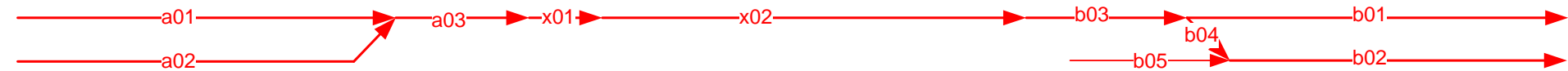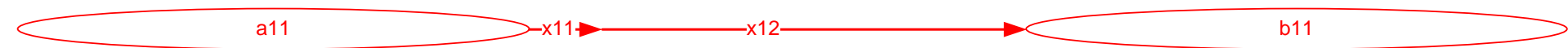
- Splitting of infrastructure networks starts at topology level

# Splitted Infrastructure

- Splitting of infrastructure networks starts at topology level
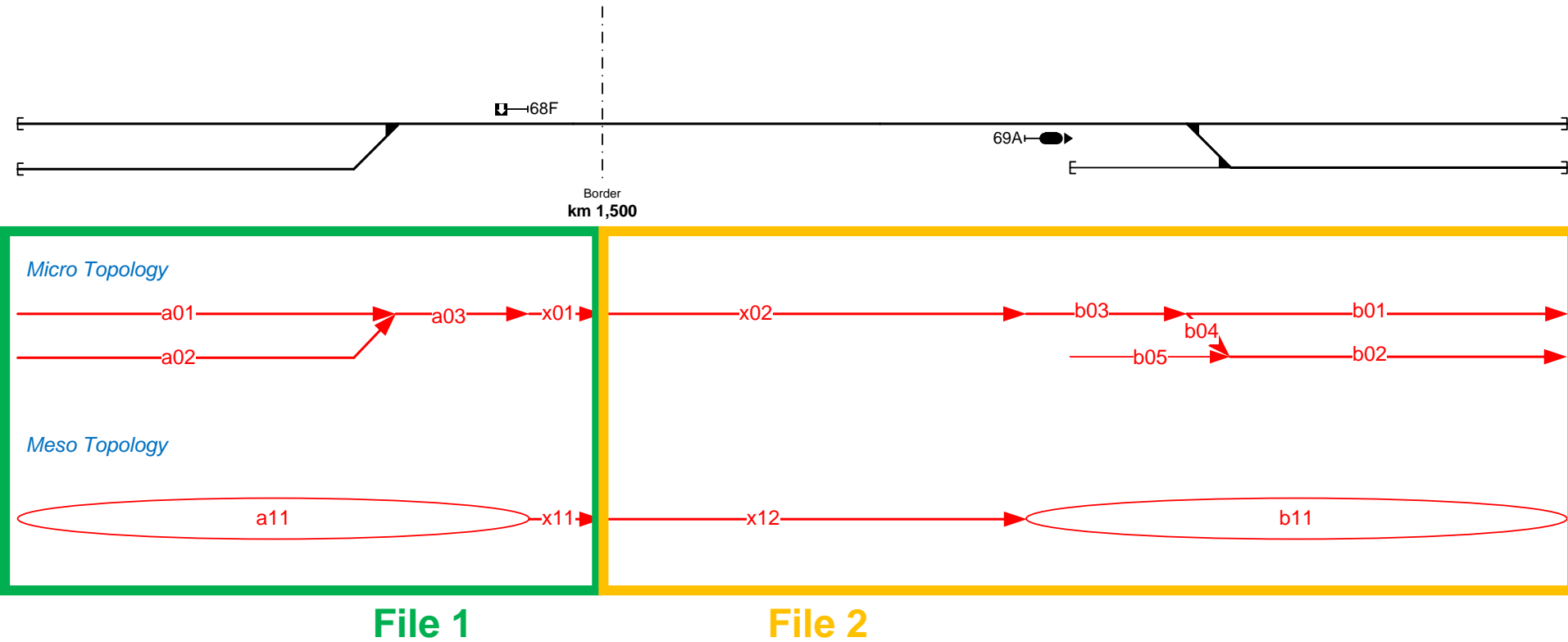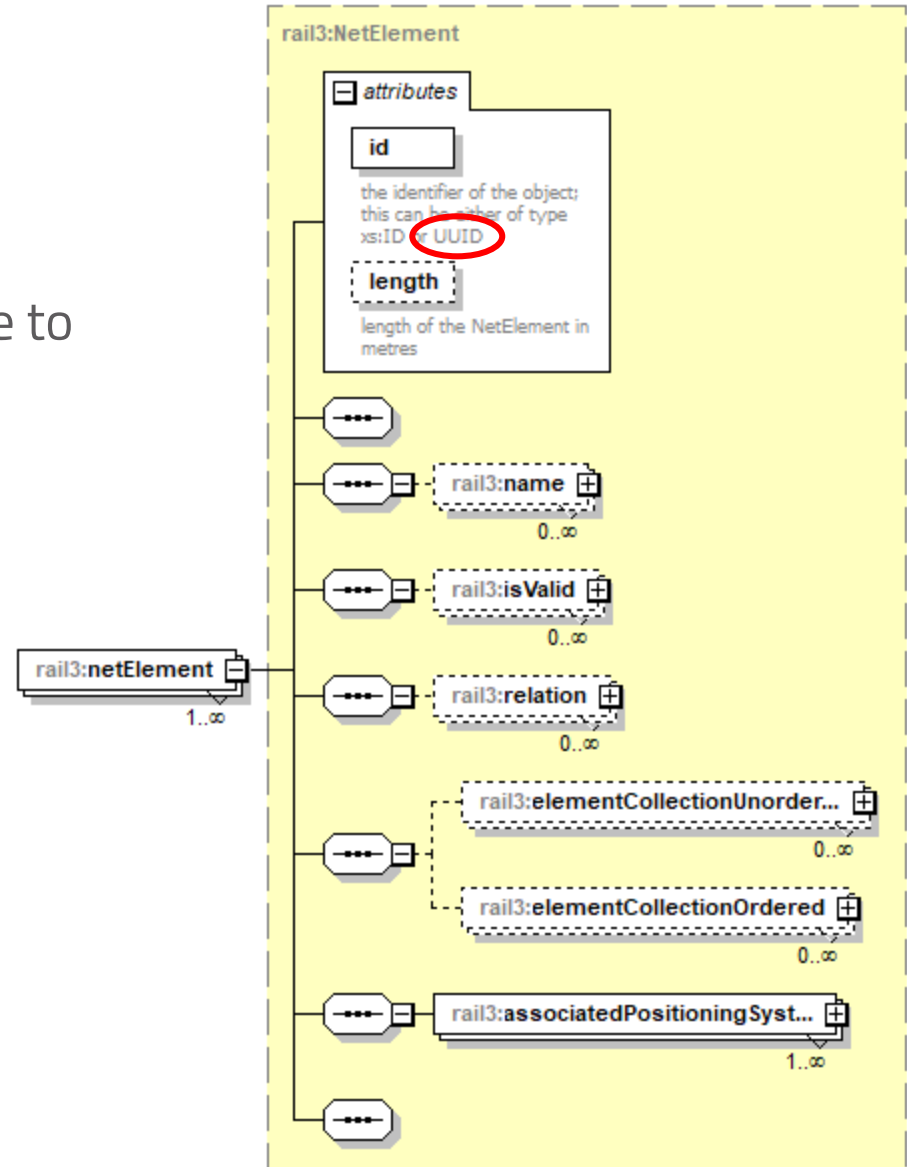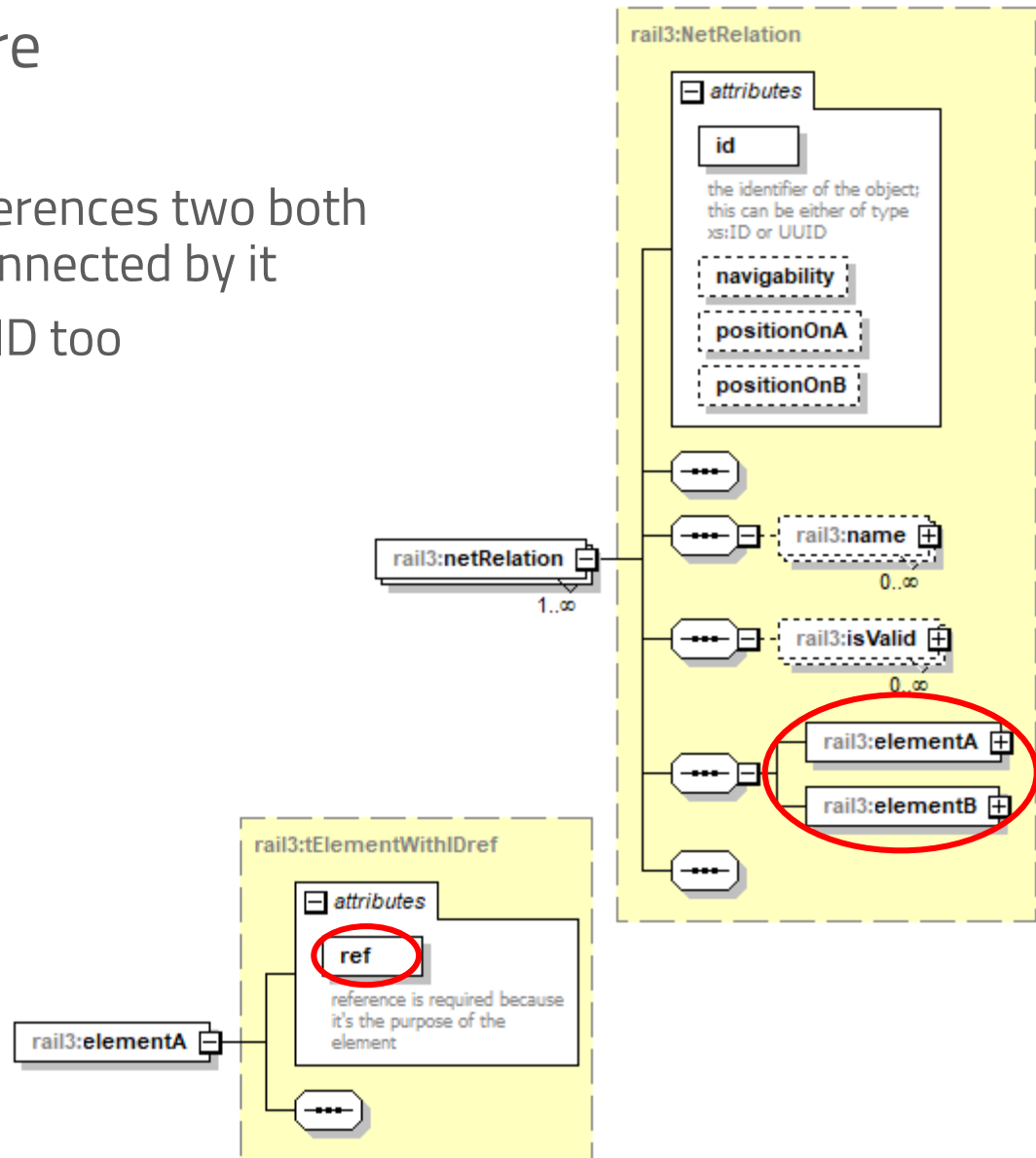
# Splitted Infrastructure

- UUID may be helpful for external referencing of NetElements
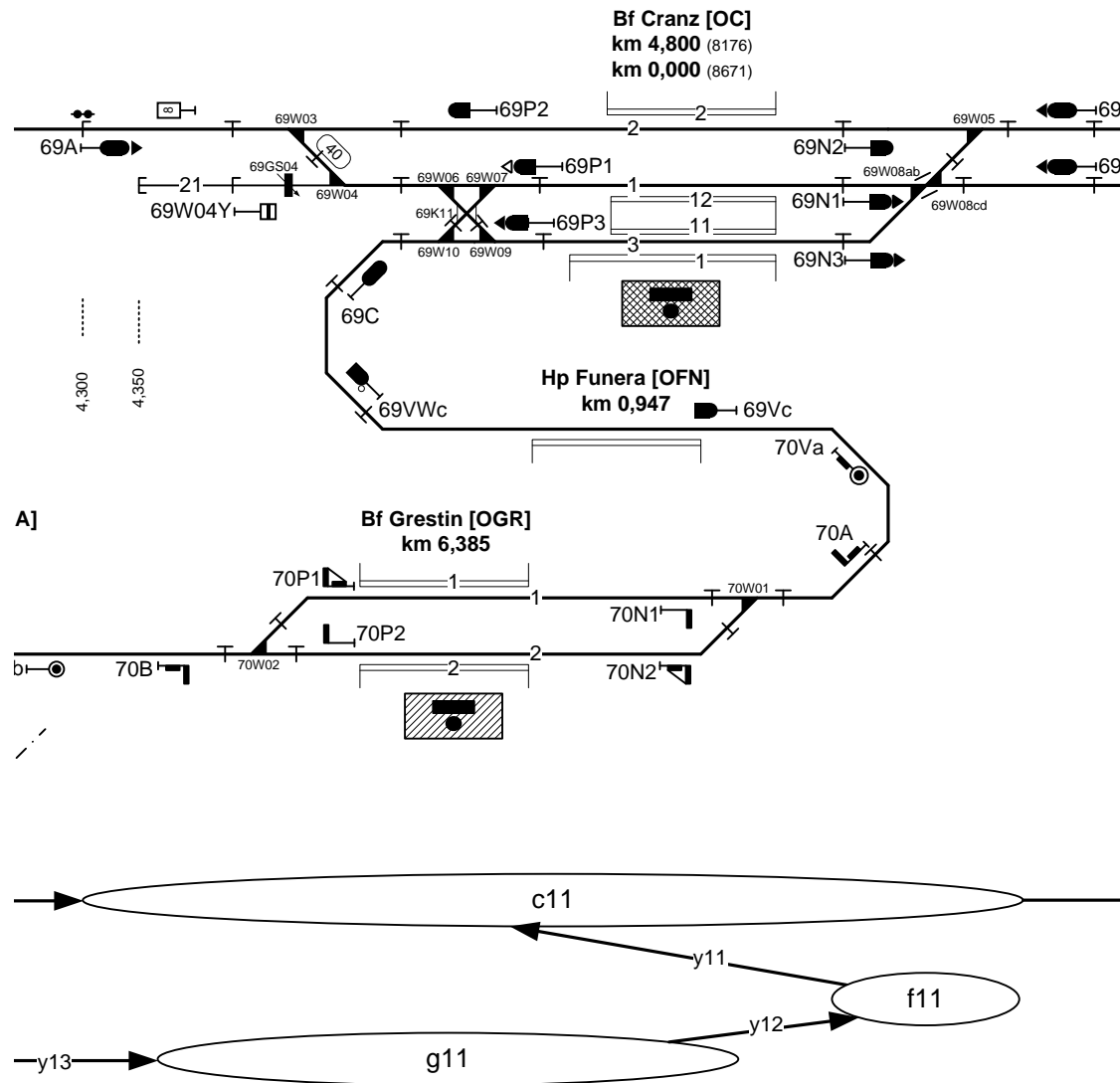- NetElements don't necessarily have to know connected NetRelations

# Splitted Infrastructure

- NetRelation requires references two both NetElements that are connected by it
- References allow for UUID too

# #452: Driving directions in macroscopic nodes

- Motivation:

# #452: Driving directions in macroscopic nodes

- The situation:
  - Without a microscopic model of the connections within a macroscopic node, it is not possible to identify possible direction changes of the railway vehicle
  - Question: how to model connections in the macroscopic node without microscopic modelling?
    - Topology external proposal (by Thomas Langkamm)
    - Topology internal proposal (by Christian Rahmig)

| Links |
| --- |
| Forum: https://www.railml.org/forum/index.php?t=msg&th=838&start=0& |
| Trac:  https://development.railml.org/railml/version3/-/issues/452 <br> https://development.railml.org/railml/version2/-/issues/413 (railML 2) |
| Wiki: |

# #452: Driving directions in macroscopic nodes

- Idea (proposal Thomas):
  - New element **&lt;travelPathInformation&gt;** in **&lt;topology&gt;**
  - &lt;travelPathInformation&gt; shall refer to three &lt;netElement&gt; objects with **&lt;from&gt;**, **&lt;via&gt;** and **&lt;to&gt;**
  - Attributes
    - @directed (boolean)
    - @distance (tLengthM)
    - @lengthRestriction (tLengthM)
    - @changesDirection (boolean)
    - @numberOfDirectionChanges (positive integer)
    - @requiresShunting (boolean)
    - @crossesContraflowTraffic (boolean)

**For details of solution proposal by Thomas see forum**

# #452: Driving directions in macroscopic nodes

- The model:
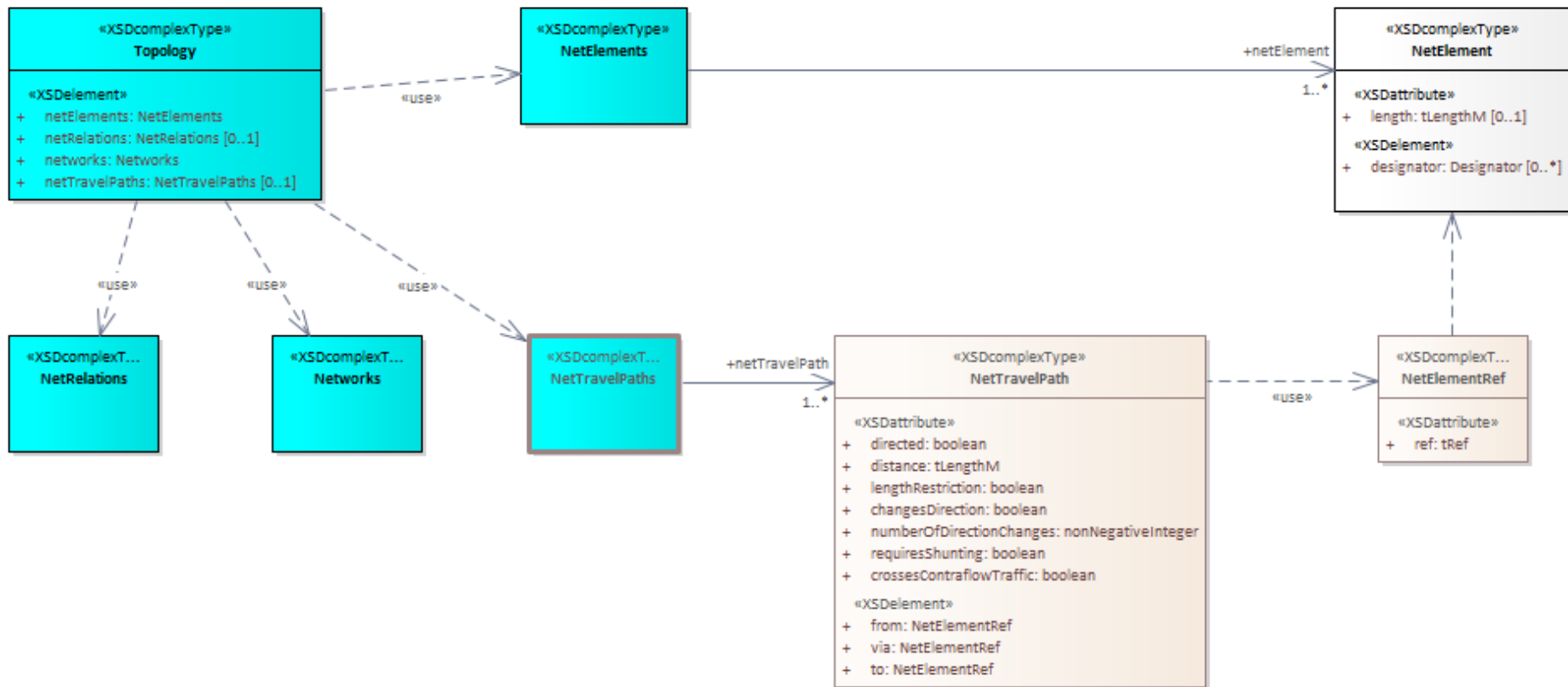
# #476: Extend <netElement> with <designator>

- The situation:
  - railML 3 topology element <netElement> is missing option for putting designators

- Ideas:
  - Option 1: extend **RTM** with designator child element for NetElement
  - Option 2: extend **railML 3 infrastructure** element NetElement (deriving from RTM) with designator

- Solution: option 2 has been implemented

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&goto=2879&#msg_2879 |
| Trac: https://development.railml.org/railml/version3/-/issues/476; https://development.railml.org/railml/railtopomodel/-/issues/1 |
| Wiki: https://wiki3.railml.org/wiki/IS:netElement |

# #476: Extend <netElement> with <designator>

- The model:
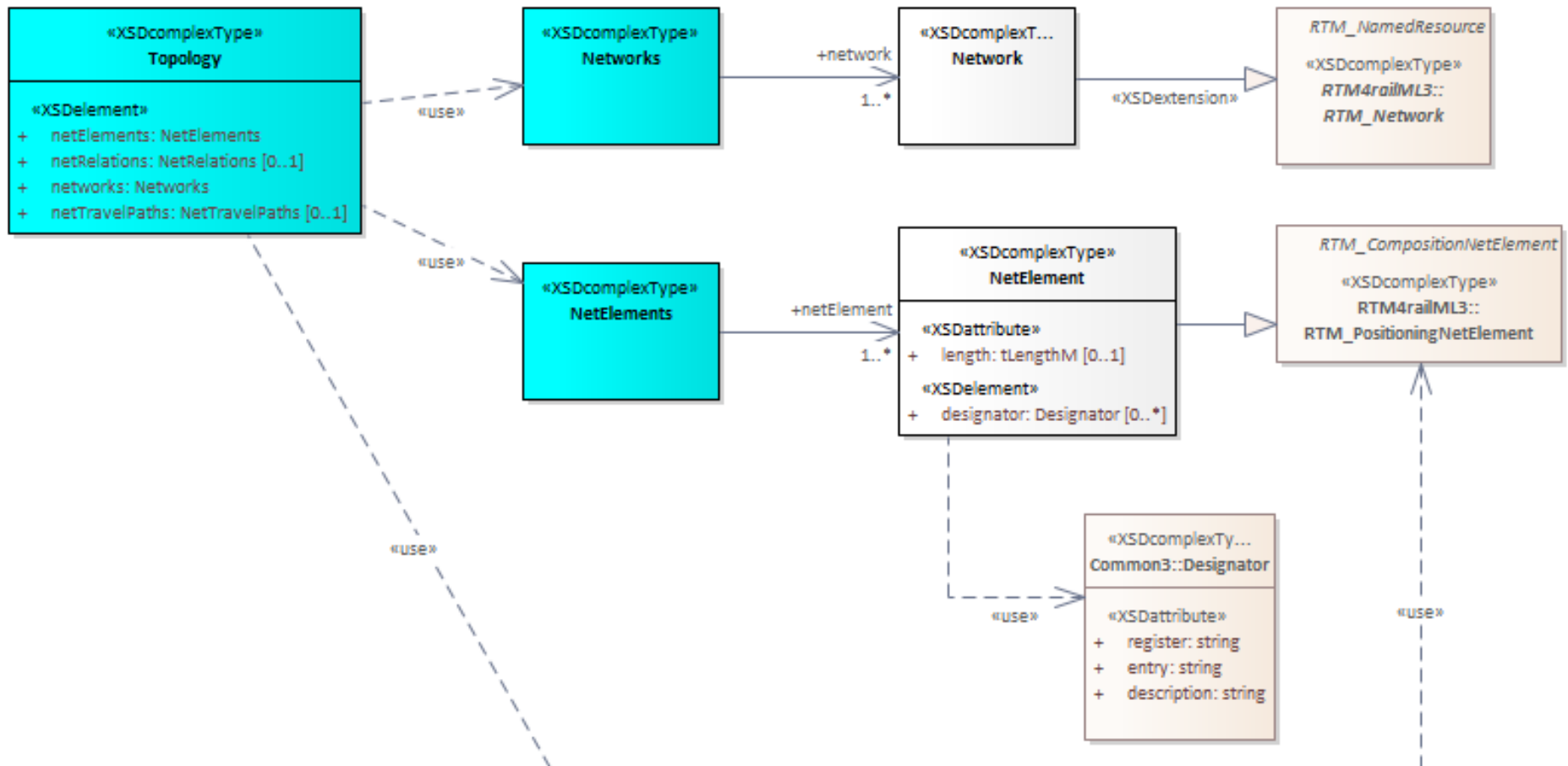
# #481: Redundant references in railML 3 topology

- The situation:
  - Reference from <netRelation> to <netElement> is mandatory
  - Reference from <netElement> to <netRelation> is optional
- Idea:
  - DEPRECATE reference from <netElement> to <netRelation> as it is redundant
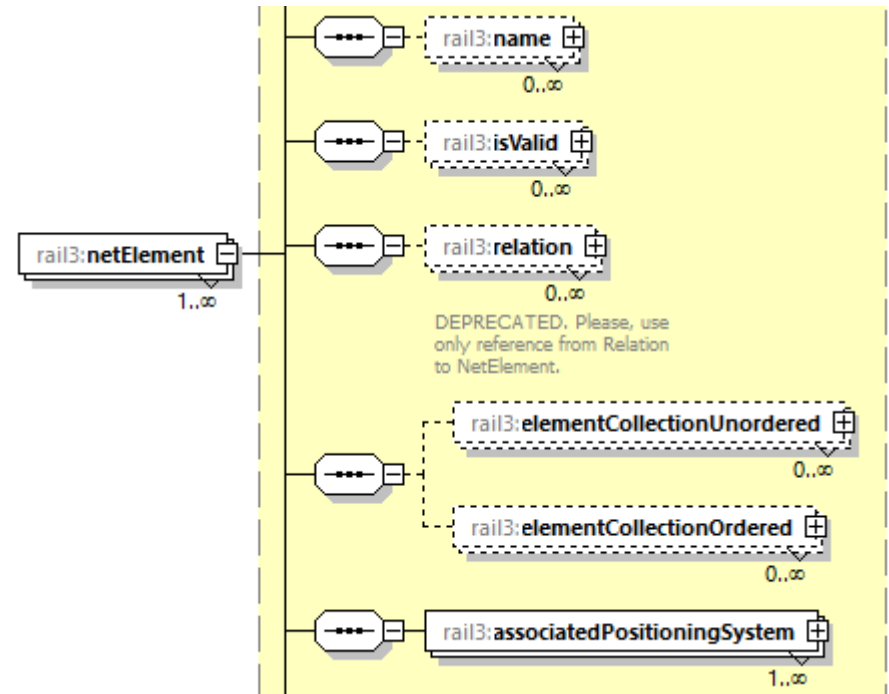


| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=828&goto=2827&#msg_2827 |
| Trac: https://development.railml.org/railml/version3/-/issues/481 ; https://development.railml.org/railml/railtopomodel/-/issues/10 |
| Wiki: |

# #487: Correct datatype of attribute @sequence

- The situation:
    - Some sequences are of type int ~~and~~ ~~positive~~ positive values

- Solution:
    - This is an RTM issue…
    - Change datatype of **@sequence** from <span style="color:red">int</span> to <span style="color:green">nonNeg…</span>
    - This affects elements:
        - <linearLocation / associatedNetElement>
        - <netElement / elementCollectionOrdered>

> Changes cannot be done in RTM due to unavailable base type in Java based UML of RTM
> → Topic for railML 3.3

| Links |
|---|
| Forum: |
| Trac: https://development.railml.org/railml/version3/-/issues/487 (old) https://development.railml.org/railml/railtopomodel/-/issues/9 |
| Wiki: |

# #500: Remove attribute @keepsOrientation

- The situation:
  - The attribute <associatedNetElement> @keepsOrientation (boolean) is unclear in its usage

- Idea / Solution:
  - Attribute **@keepsOrientation** (is mandatory in railML 3.1) will be marked „deprecated" in railML 3.2 and removed with railML 3.3

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=818&goto=2935&#msg_2935 |
| Trac: https://development.railml.org/railml/version3/-/issues/500 |
| Wiki: |

**rail**ML.org

# #500: Remove attribute @keepsOrientation

- The model:



rail3:RTM_AssociatedNetElement

attributes

netElementRef

intrinsicCoordBegin

intrinsicCoordEnd

keepsOrientation

As of version 3.2 this attribute is obsolete. It turned out that this attribute although required does not serve any meaningful purpose nor did anyone know what it was supposed to do. It will be removed as of version 3.3 of railML.

sequence

posBegin

posEnd

rail3:associatedNetElement
1..∞

rail3:geometricCoordinateBegin

rail3:linearCoordinateBegin

rail3:geometricCoordinateEnd

rail3:linearCoordinateEnd

# Line, Operational Point, Mileage change

# #441: Extension of <opEquipment>

- The situation:
  - OperationalPoint can reference platforms, tracks, signals, stoppingPlaces and serviceSections, but not switches and derailers
- Idea:
  - Option 1: extend <opEquipment> with **<ownsStoppingPlaces>**, **<ownsSwitch>** and **<ownsDerailer>**
  - Option 2: introduce generic child element **<ownsInfrastructureElement>**

**Option 2 has been implemented.**

# #441: Extension of <opEquipment>

- The model:
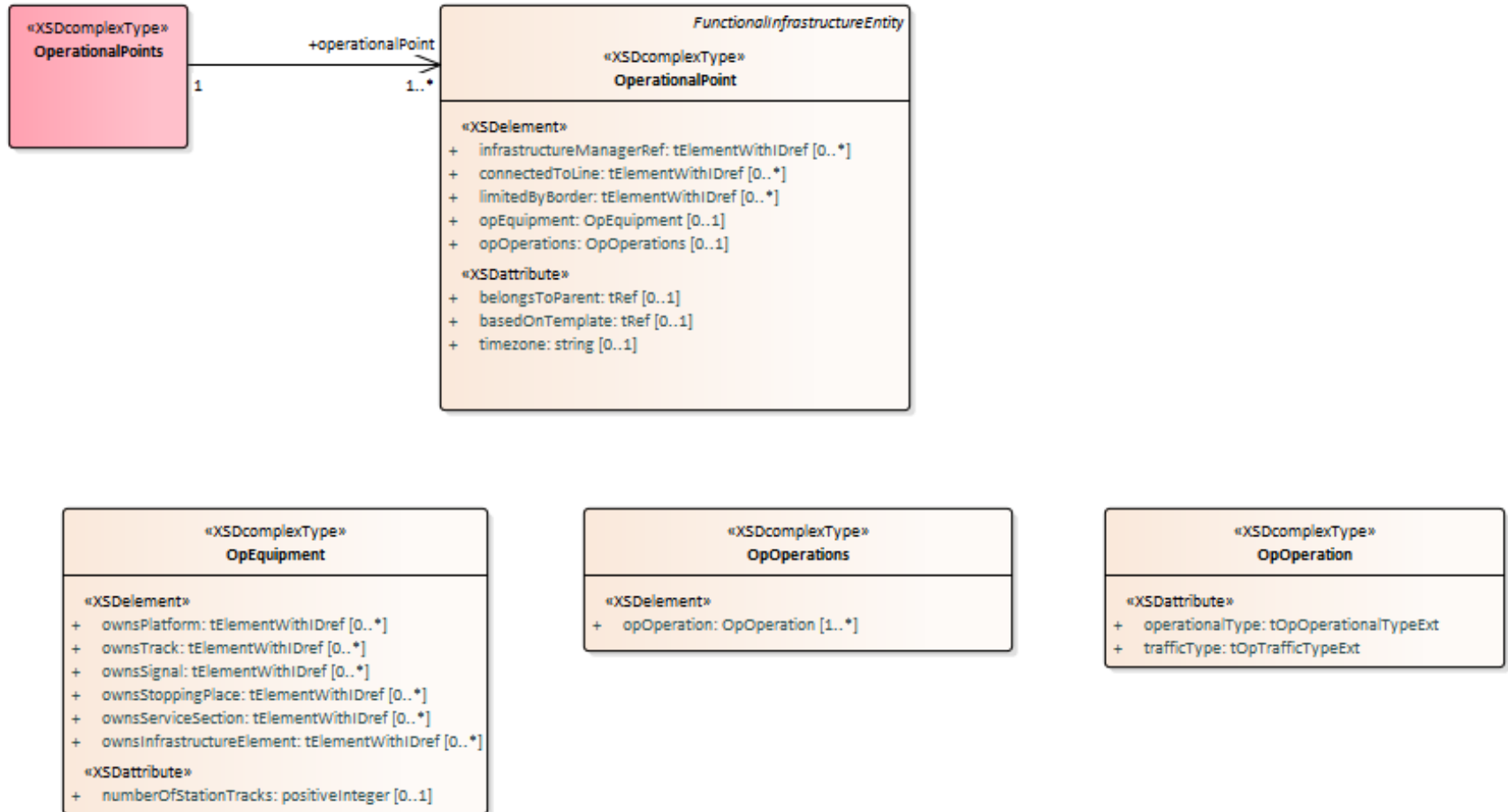
# #442: Transfer times for connections

- The situation:
  - In railML 2.x transfer times between trains are modelled in timetable domain with @minConnectionTime
  - In railML 3.x transfer times are not yet modelled
- Idea:
  - Since transfer times seem to be constant for platform relations, the idea has been formulated to implement transfer times in infrastructure domain (connected with platforms)

> **Decision from railML TT developers group: transfer times shall be modelled in TT**

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=714&goto=2382&#msg_2382 |
| Trac: https://development.railml.org/railml/version3/-/issues/442 |
| Wiki: |

**rail**ML**.org**

# #478: New element for mileage change modelling

- The situation:
  - In railML 3.1 mileage changes are being modelled using <linearPositioningSystem> with <anchor> points (RTM approach)
- Idea:
  - Introduce new functional infrastructure element **<mileageChange>**
  - <mileageChange>@type (gap, overlap) describes the km jump
  - <mileageChange> has two <spotLocation> child elements to describe the incoming and the outgoing mileage
  - <mileageChange>@from references „incoming" spot location
  - <mileageChange>@to references „outgoing" spot location

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=829&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/478 |
| Wiki: |

# #478: New element for mileage change modelling

- The model:
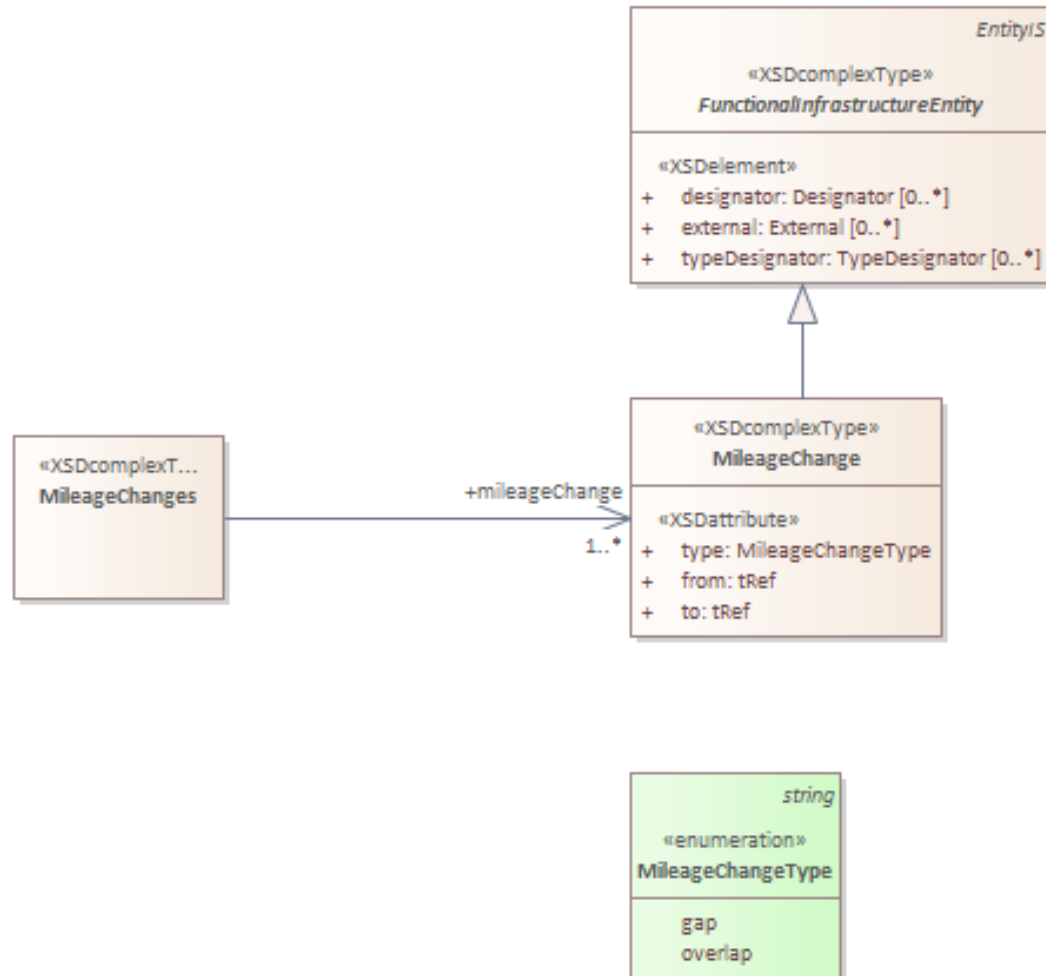
# #497: Support for line operation mode

- Task:
  - railML should support describing the mode of operation for a line or parts of it. This should be done by referring to either an enumeration or a register. As a minimum the values RIL408, RIL436, RIL437 and RIL438 should be supported.

- Solution:
  - Implement a new child element **\<line>\<lineOperation>** with extendable enum attribute **@modeOfOperation**

| Links |
|---|
| Forum: |
| Trac: https://development.railml.org/railml/version3/-/issues/497 |
| Wiki: https://wiki3.railml.org/wiki/IS:line |

**rail**ML.org

# #497: Support for line operation mode

- The model:



**Line** (FunctionalInfrastructureEntity) «XSDcomplexType»

«XSDelement»
+ beginsInOP: tElementWithIDref [0..1]
+ endsInOP: tElementWithIDref [0..1]
+ length: Length [0..*]
+ lineTrafficCode: LineTrafficCode [0..*]
+ lineCombinedTransportCode: LineCombinedTransportCode [0..*]
+ lineLayout: LineLayout [0..1]
+ linePerformance: LinePerformance [0..1]
+ lineOperation: LineOperation [0..1]

«XSDattribute»
+ infrastructureManagerRef: tRef [0..1]
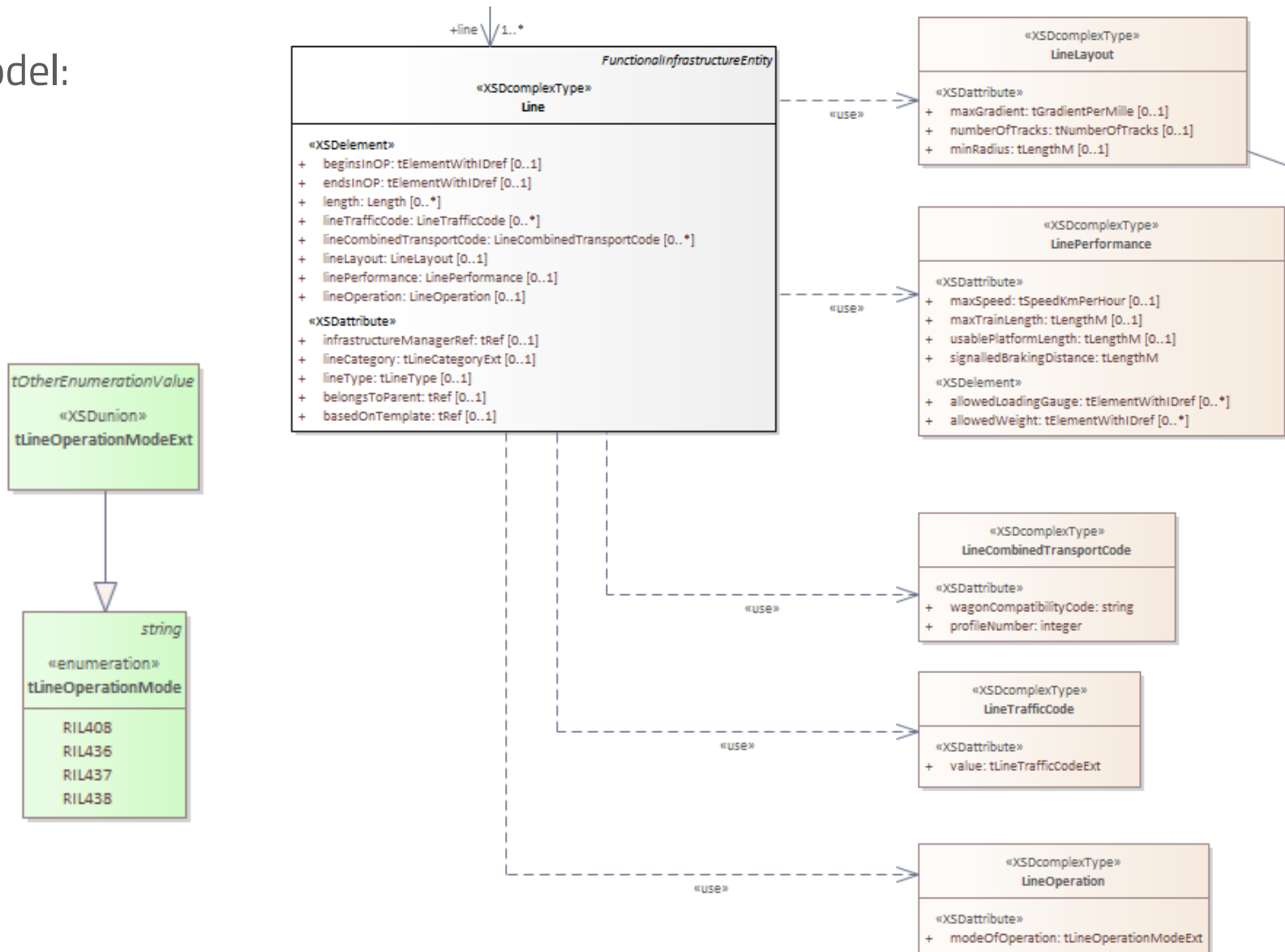+ lineCategory: tLineCategoryExt [0..1]
+ lineType: tLineType [0..1]
+ belongsToParent: tRef [0..1]
+ basedOnTemplate: tRef [0..1]

**LineLayout** «XSDcomplexType»

«XSDattribute»
+ maxGradient: tGradientPerMille [0..1]
+ numberOfTracks: tNumberOfTracks [0..1]
+ minRadius: tLengthM [0..1]

**LinePerformance** «XSDcomplexType»

«XSDattribute»
+ maxSpeed: tSpeedKmPerHour [0..1]
+ maxTrainLength: tLengthM [0..1]
+ usablePlatformLength: tLengthM [0..1]
+ signalledBrakingDistance: tLengthM

«XSDelement»
+ allowedLoadingGauge: tElementWithIDref [0..*]
+ allowedWeight: tElementWithIDref [0..*]

**tLineOperationModeExt** (tOtherEnumerationValue) «XSDunion»

**tLineOperationMode** (string) «enumeration»

RIL408
RIL436
RIL437
RIL438

**LineCombinedTransportCode** «XSDcomplexType»

«XSDattribute»
+ wagonCompatibilityCode: string
+ profileNumber: integer

**LineTrafficCode** «XSDcomplexType»

«XSDattribute»
+ value: tLineTrafficCodeExt

**LineOperation** «XSDcomplexType»

«XSDattribute»
+ modeOfOperation: tLineOperationModeExt

railML.org

# #505: Standard braking distance at a line

- Situation:
  - railML 3 misses data model for signalled braking distance (default distance between distant signal and main signal)

- Idea / solution:
  - New optional attribute **@signalledBrakingDistance** (tLengthM) shall be added in **<line / linePerformance>**

# Other infrastructure

# #422: Natural hazards detection

- The problem:
  - How to model areas where technical systems for natural hazard detection (e.g. avalanche, sand, camels, reindeer…) are installed
  - These detectors may trigger reactions in a TMS
- Ideas:
  - Introduce generic infrastructure element **<detector>** that @detects different types of hazards
  - Types of hazards as open enumeration list
  - <detector> can be <linkedWith>@ref restriction area that defines a certain operational reaction (e.g. „noStopping") on the detected hazard
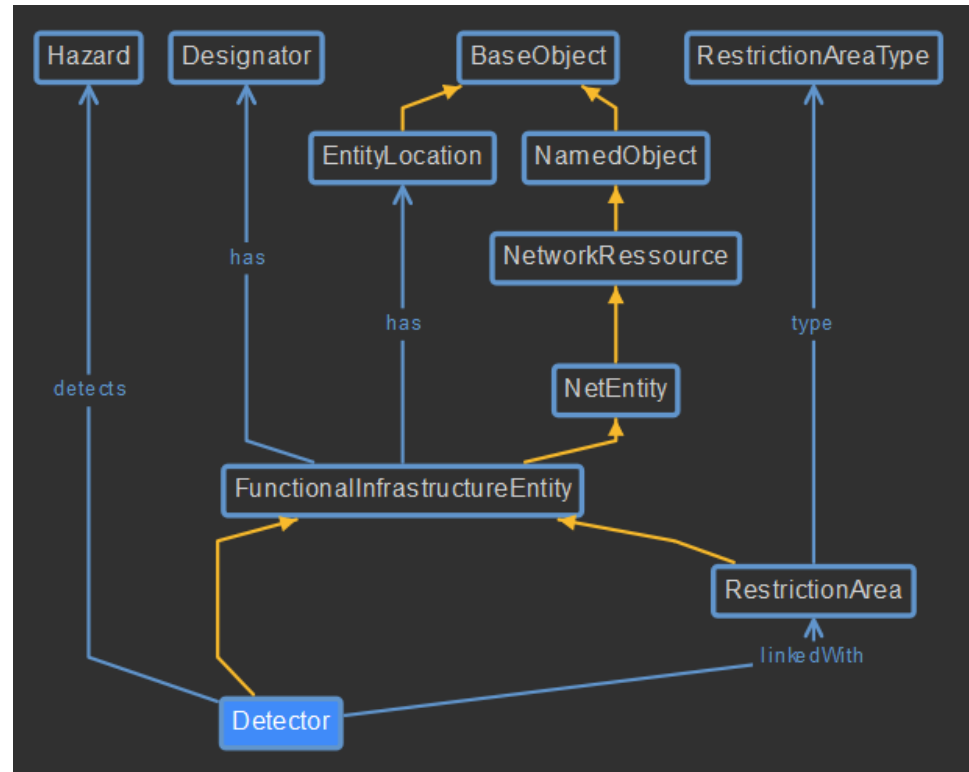
**rail**ML**.org**

# #422: Natural hazards detection

- Solution:
  - Detecting hazards

# #422: Natural hazards detection

- The model:
  - <detector>

railML.org
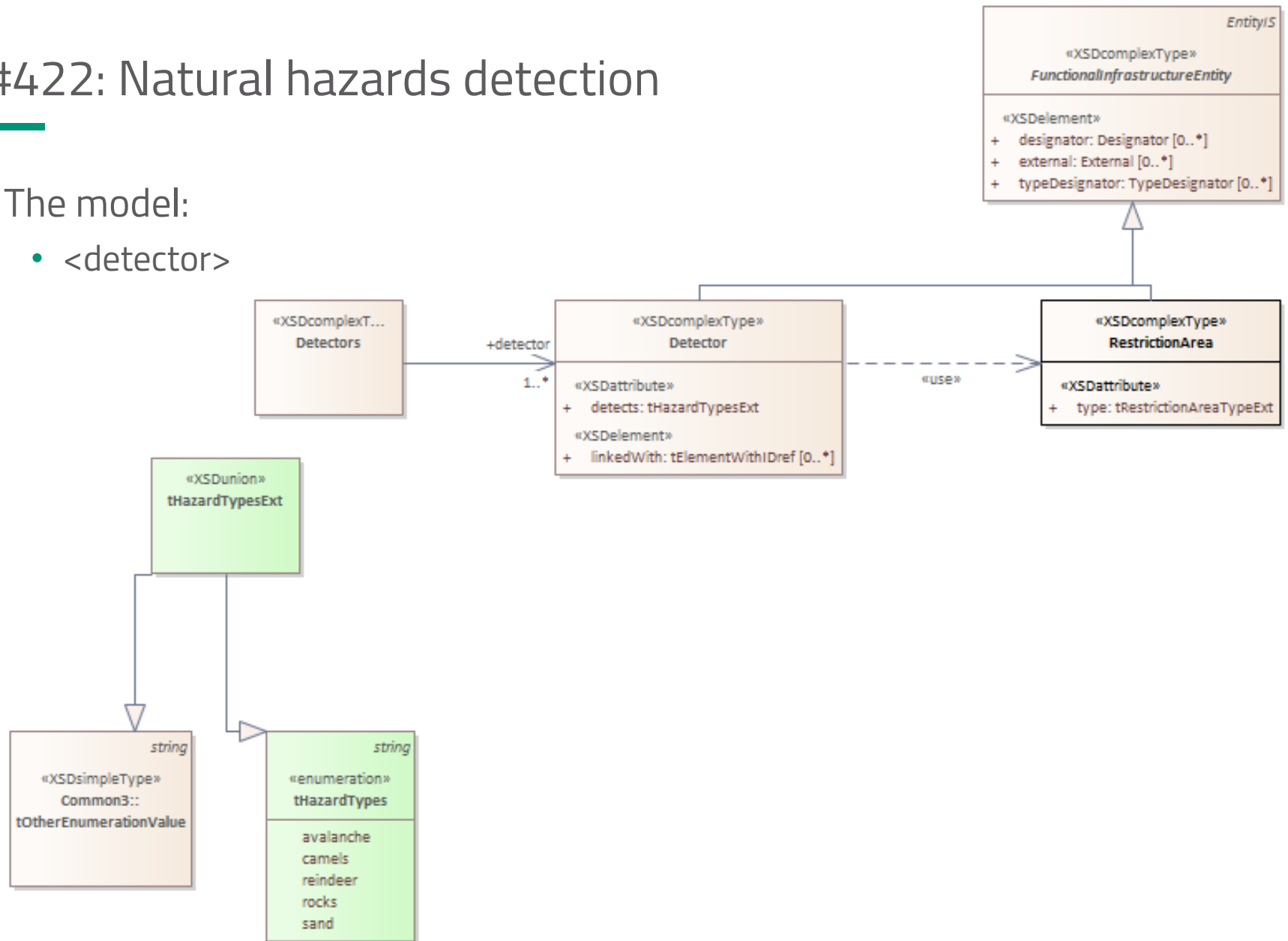
# #443: Re-introducing @ruleCode?

- The situation:
  - In railML 2.x a signal can linked with a rule book identifier using attribute @ruleCode
  - In railML 3.x the @ruleCode attribute is (so far) not modelled
- Idea:
  - ~~Option 1: implement attribute **@ruleCode** for signals (and other signalling related elements)~~
  - Option 2: A new "designator-like" element is introduced with attributes @rulebook and @entry. This could be called **&lt;typeDesignator&gt;**
  - ~~Option 3: use available child element **&lt;designator&gt;** to specify a rule code~~

# #443: Re-introducing @ruleCode?

- The solution:
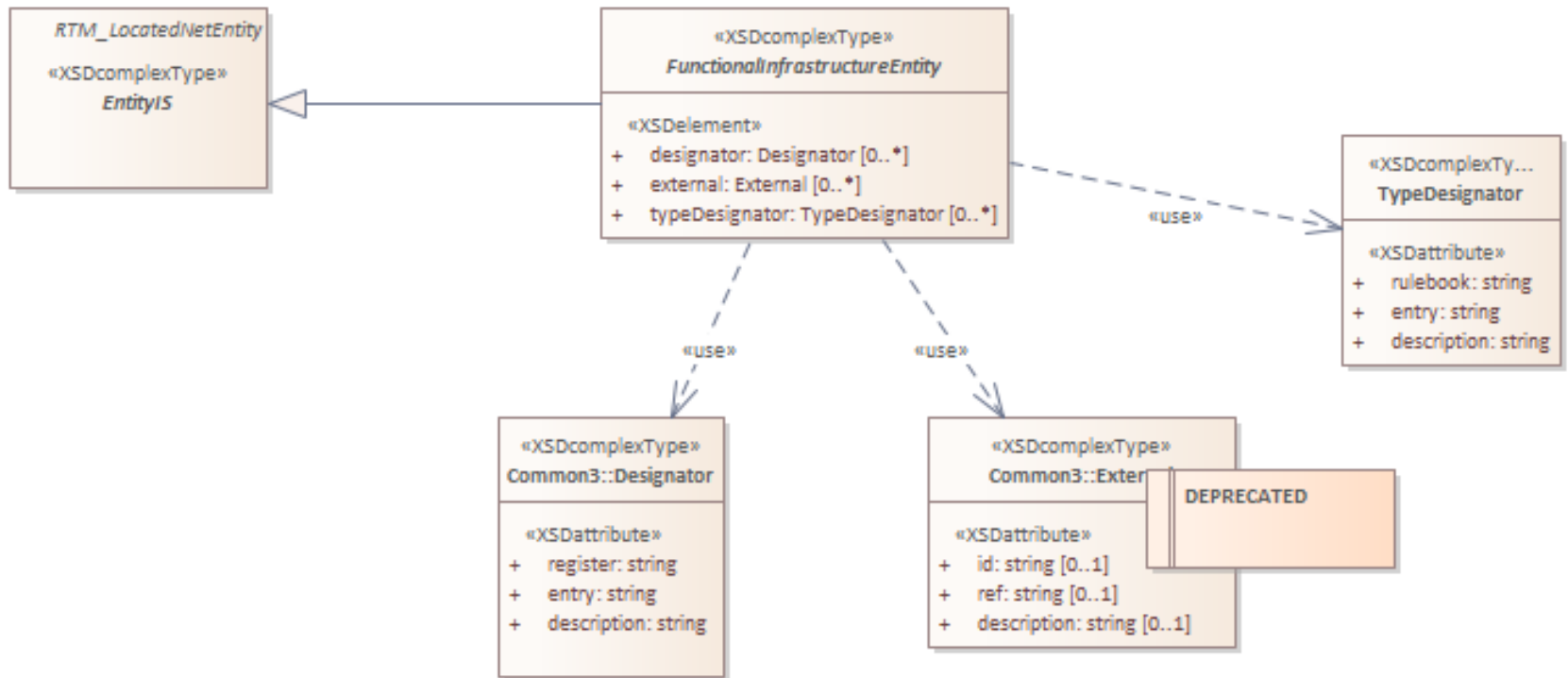
# #449: Movable bridge

- The situation:
  - There are bridges that can be temporarily lifted/moved e.g. for ships to pass; normal position of such a „bascule bridge" is „closed" (passable by train)

- Idea:
  - Extend enumeration **<*crossing>@constructionType** with new value „movableBridge"
  - (further adaptations in IL)



| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=781&start=0& |
| Trac: https://development.railml.org/railml/version3/-/issues/449 |
| Wiki: https://wiki3.railml.org/wiki/IS:overCrossing; https://wiki3.railml.org/wiki/IS:underCrossing |
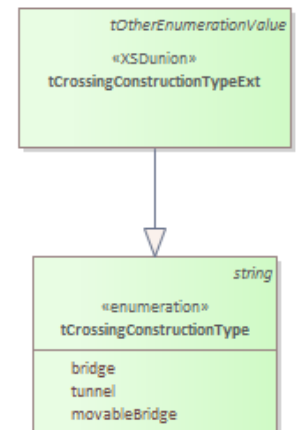
# #461: Loading gauge profiles

- The situation:
  - Current implementation of <loadingGauge> is missing static and kinematic reference profiles
- Idea / solution:
  - Add new child elements **<staticProfile>** and **<kinematicProfile>** in parent element <loadingGauge> with parameters @width (in meters) and @height (in meters)

**rail**ML**.org**

# #461: Loading gauge profiles

- The model:
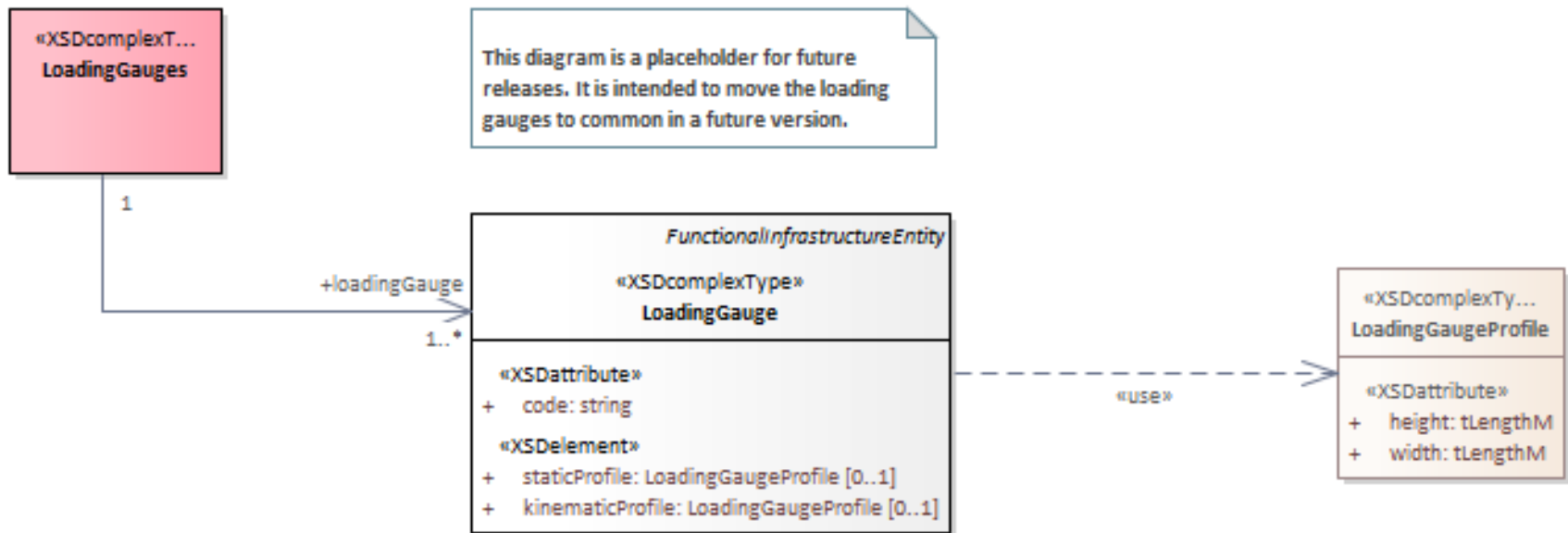
# #466: Tunnel Gate in Infrastructure

- The situation:
  - There can be gates installed at different locations inside a tunnel

- Idea:
  - Explicit modelling of these tunnel gates
  - Introduce new infrastructure element **\<tunnelGateIS\>**
    - Location
    - Reference to a tunnel (overCrossing)
      - **@installedInTunnel** (tRef)

| Links |
|---|
| Forum: https://www.railml.org/forum/index.php?t=msg&th=793&goto=2646&#msg_2646 |
| Trac: https://development.railml.org/railml/version3/-/issues/466; https://development.railml.org/railml/version3/-/issues/450; https://development.railml.org/railml/version3/-/issues/449 |
| Wiki: https://wiki3.railml.org/wiki/IL:tunnelGateIL |

# #466: Tunnel Gate in Infrastructure

- The model:
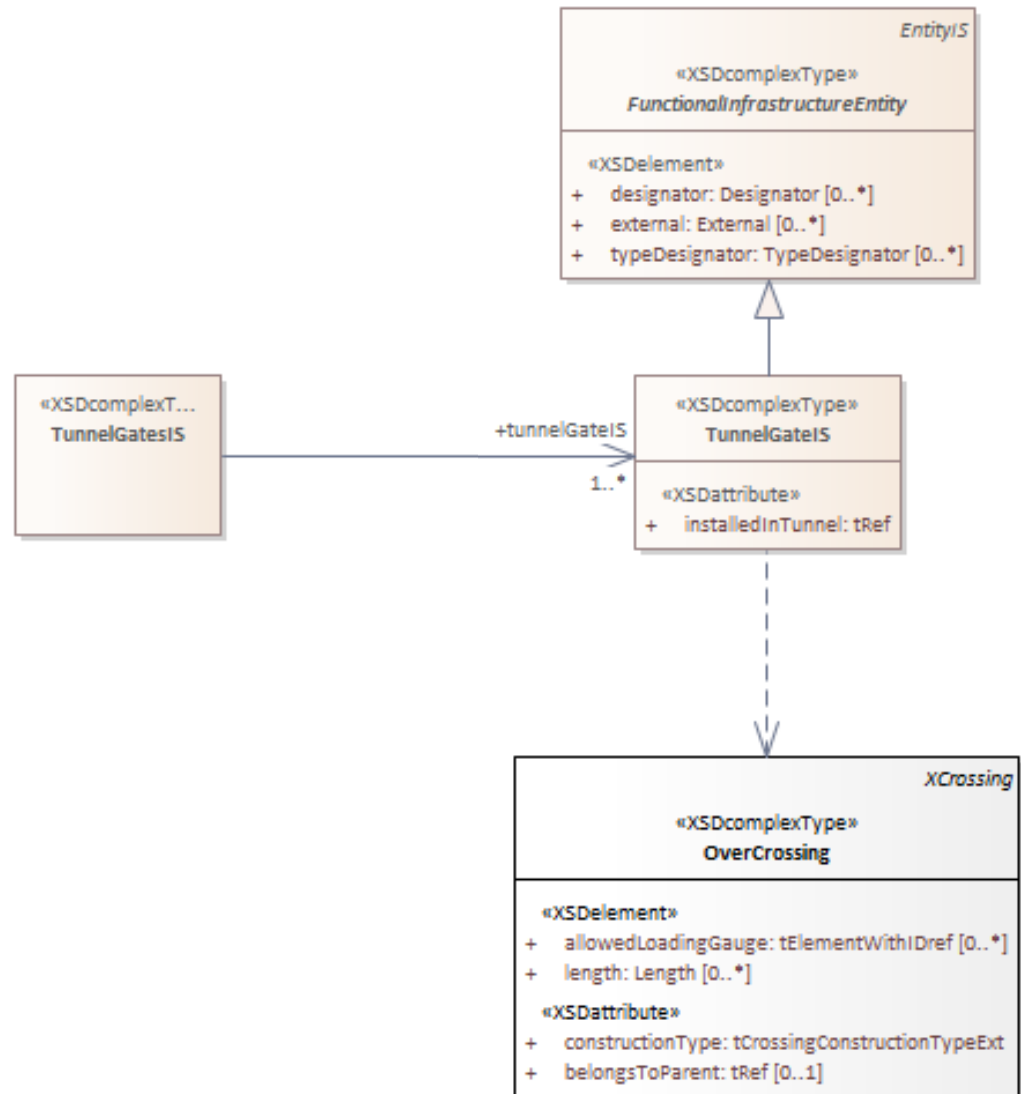
# #495: Different derailer types

- Task: It shall be possible to model that a derailer can be used to block several tracks at once.

- Solution:
  - Add new attribute **&lt;derailerIS&gt;@type** (enum: singleDerailer, doubleDerailer)
  - &lt;derailerIL&gt; can link several tracks that are locked by it



| Links |
| --- |
| Forum: https://www.railml.org/forum/index.php?t=msg&th=855&start=0& |
| Trac:  https://development.railml.org/railml/version3/-/issues/495 |
| Wiki: https://wiki3.railml.org/wiki/IS:derailerIS |

# #495: Different derailer types

- The model:
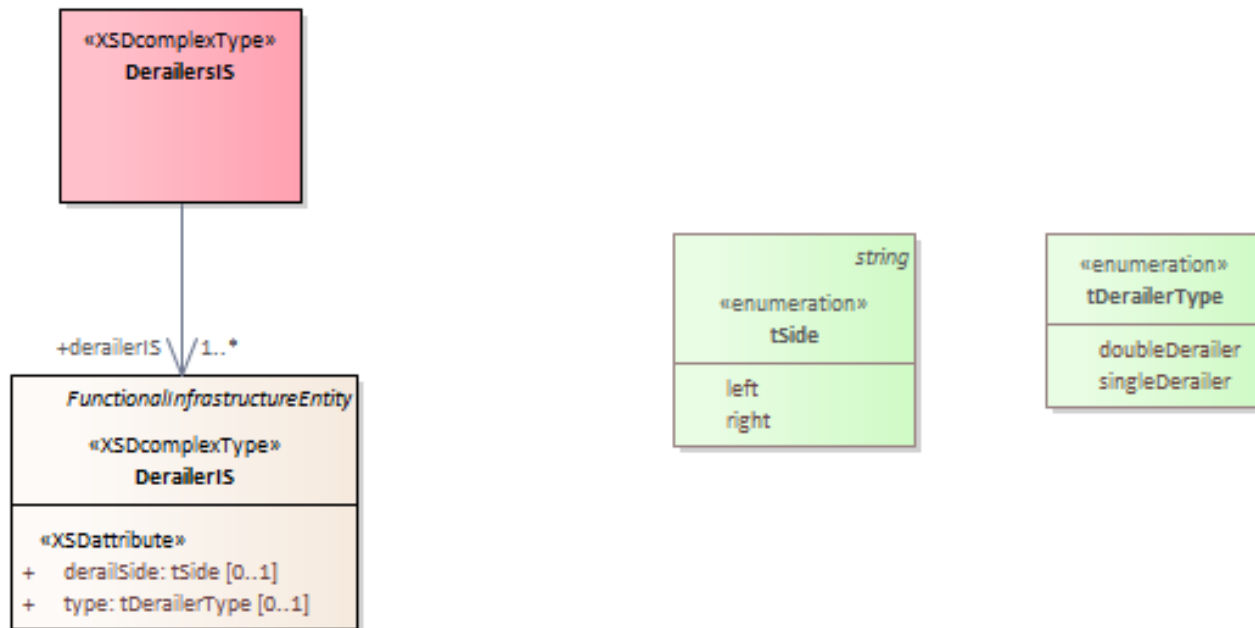
# #499: Renaming and extending serviceSection facilities

- The situation:
  - The serviceSection facilities are not complete and not named in a unified way.

- Solution:
  - The facilities of the **\<serviceSection\>** shall be renamed according to the following pattern:
    **@allows{*Domain*}{*Service*}**

«XSDattributeGroup»
aServiceSection

«XSDattribute»
+ allowsRollingstockCleaning: boolean [0..1]
+ allowsRollingstockFueling: boolean [0..1]
+ allowsRollingstockLoading: boolean [0..1]
+ allowsRollingstockMaintenance: boolean [0..1]
+ allowsRollingstockParking: boolean [0..1]
+ allowsRollingstockPreheating: boolean [0..1]
+ hasRamp: boolean [0..1]
+ allowsRollingstockToiletDischarge: boolean [0..1]
+ allowsRollingstockWaterRestocking: boolean [0..1]
+ allowsRollingstockSandRestocking: boolean [0..1]
+ hasElectricSupply: boolean [0..1]
+ allowsInfrastructureMaintenance: boolean [0..1]
+ allowsCleaning: boolean [0..1]
+ allowsFueling: boolean [0..1]
+ allowsLoading: boolean [0..1]
+ allowsMaintenance: boolean [0..1]
+ allowsParking: boolean [0..1]
+ allowsPreheating: boolean [0..1]
+ allowsSandRestocking: boolean [0..1]
+ allowsToiletDischarge: boolean [0..1]
+ allowsWaterRestocking: boolean [0..1]

| Links |
| --- |
| Forum: https://www.railml.org/forum/index.php?t=msg&th=862&start=0& |
| Trac:  https://development.railml.org/railml/version3/-/issues/499 |
| Wiki: https://wiki3.railml.org/wiki/IS:serviceSection |

# Visualization

# #370: Visualizations

- The situation:
  - railML 3.1 contains <infrastructureVisualization> scheme that is used to model graphical visualizations of the infrastructure
  - How about visualization of timetable, rollingstock, interlocking elements?

- Idea:
  - Generalize the concept of visualizations in new schema <visualizations>



| | Infrastructure | | |
|---|---|---|---|
| e | topology | [0..1] | Topology |
| e | geometry | [0..1] | Geometry |
| e | functionalInfrastructure | [0..1] | FunctionalInfrastructure |
| e | physicalFacilities | [0..1] | PhysicalFacilities |
| e | infrastructureVisualizations | [0..1] | InfrastructureVisualizations |
| e | infrastructureStates | [0..1] | InfrastructureStates |
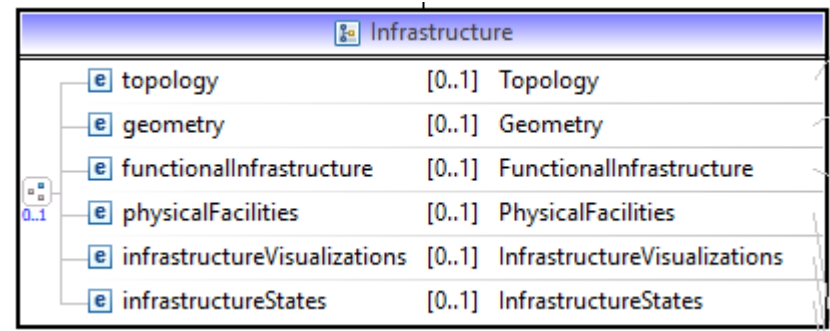
# #370: Visualizations

- Solution:
  - New sub-schema **\<visualizations\>**
  - Within \<visualizations\> child elements for domain-specific visualizations could be added → e.g. **\<infrastructureVisualizations\>**
  - Add new type of infrastructure visualization in form of child element \<ellipticalProjection\> to visualize circular and elliptical elements

# #370: Visualizations
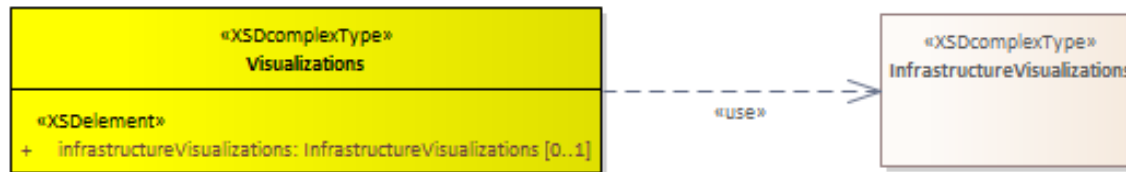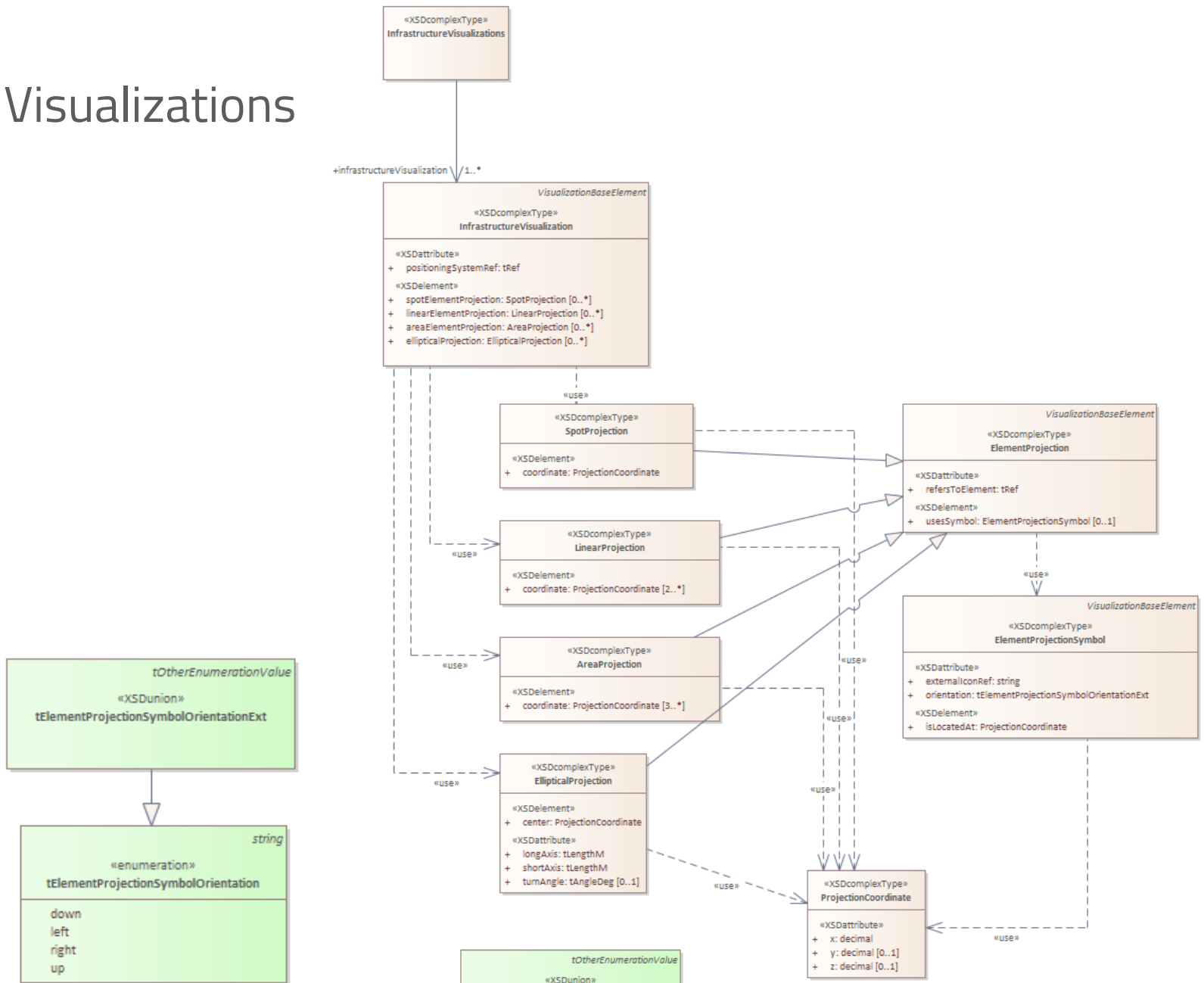
**InfrastructureVisualizations**
«XSDcomplexType»

+infrastructureVisualization  1..*

**InfrastructureVisualization**
*VisualizationBaseElement*
«XSDcomplexType»

«XSDattribute»
+  positioningSystemRef: tRef
«XSDelement»
+  spotElementProjection: SpotProjection [0..*]
+  linearElementProjection: LinearProjection [0..*]
+  areaElementProjection: AreaProjection [0..*]
+  ellipticalProjection: EllipticalProjection [0..*]

«use»

**SpotProjection**
«XSDcomplexType»

«XSDelement»
+  coordinate: ProjectionCoordinate

**ElementProjection**
*VisualizationBaseElement*
«XSDcomplexType»

«XSDattribute»
+  refersToElement: tRef
«XSDelement»
+  usesSymbol: ElementProjectionSymbol [0..1]

**LinearProjection**
«XSDcomplexType»

«XSDelement»
+  coordinate: ProjectionCoordinate [2..*]

«use»

«use»

**ElementProjectionSymbol**
*VisualizationBaseElement*
«XSDcomplexType»

«XSDattribute»
+  externalIconRef: string
+  orientation: tElementProjectionSymbolOrientationExt
«XSDelement»
+  isLocatedAt: ProjectionCoordinate

**AreaProjection**
«XSDcomplexType»

«XSDelement»
+  coordinate: ProjectionCoordinate [3..*]

«use»

**tElementProjectionSymbolOrientationExt**
*tOtherEnumerationValue*
«XSDunion»

**EllipticalProjection**
«XSDcomplexType»

«XSDelement»
+  center: ProjectionCoordinate
«XSDattribute»
+  longAxis: tLengthM
+  shortAxis: tLengthM
+  turnAngle: tAngleDeg [0..1]

**tElementProjectionSymbolOrientation**
*string*
«enumeration»

down
left
right
up

**ProjectionCoordinate**
«XSDcomplexType»

«XSDattribute»
+  x: decimal
+  y: decimal [0..1]
+  z: decimal [0..1]

«use»

*tOtherEnumerationValue*
«XSDunion»

# railML.org

Thank you very much for your attention!

**+ *contribution***

*All figures if not specifically stated: © railML.org*

**Deutsches Zentrum**
**für Luft- und Raumfahrt** e.V.
in der Helmholtz-Gemeinschaft

**Institut für Verkehrssystemtechnik**

**DLR**

Christian Rahmig

railML Infrastructure Coordinator

www.railml.org

coord@infrastructure.railml.org

+49 – (0)531 – 295 3461

+49 – 173 – 2714509