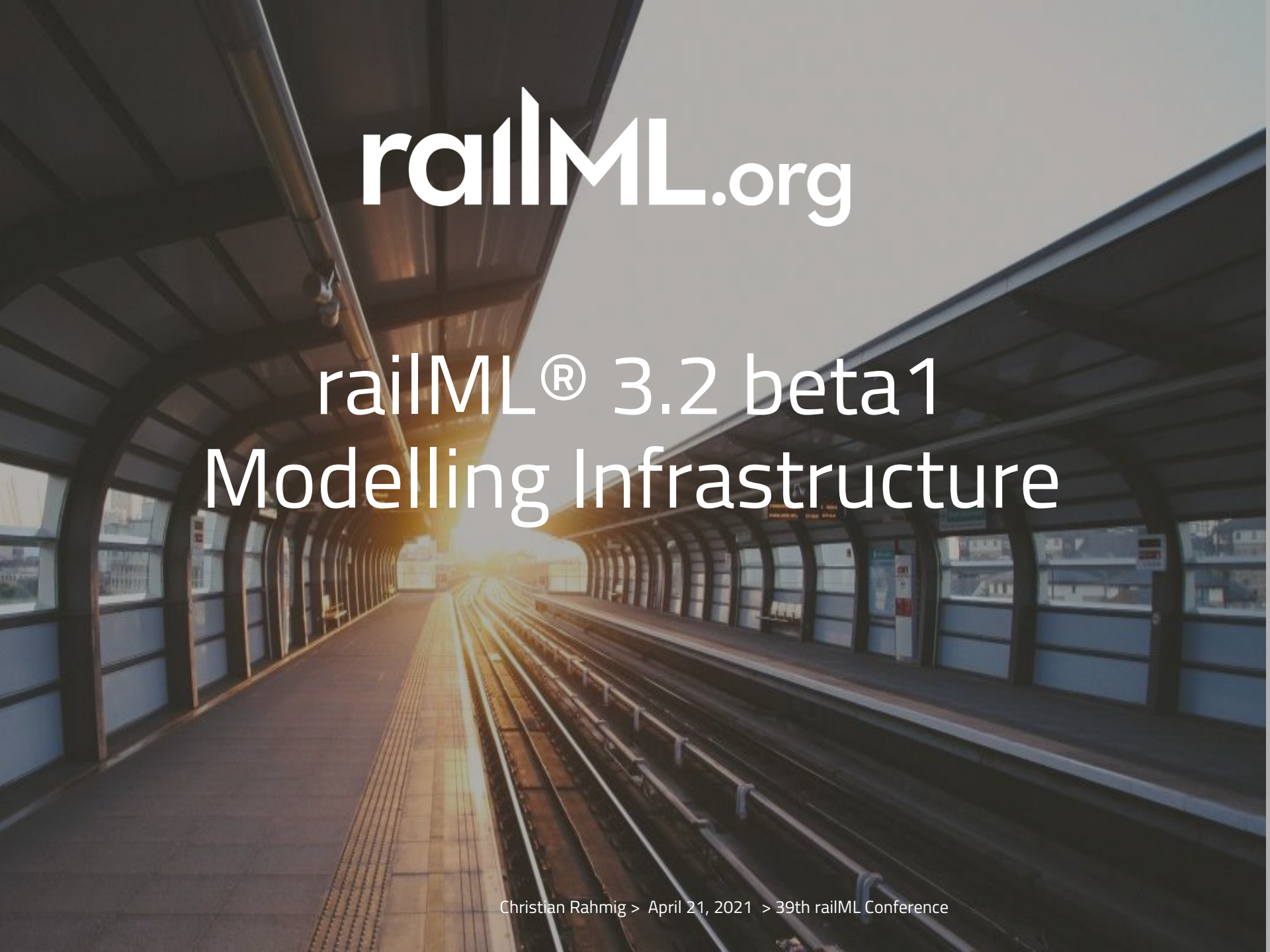
railML.org

railML® 3.2 beta1
Modelling Infrastructure



Christian Rahmig > April 21, 2021 > 39th railML Conference

Agenda

1. News from railML 3.2 Use Case Working Groups
   - ETCS Track Net Description (ETCS)
   - Integrated Traffic Management (ITMS)
   - Schematic Track Plan update (SCTP)
   - Track Geometry (TRGE)
   - *Ontology*
2. Modelling railML 3.2
3. railML Advanced Example

News from Use Case Working Groups

Working on railML 3.2

Use Case Working Groups

- railML 3.1 use cases
  - SCTP – Schematic Track Plan
  - NEST – Network Statement
  - RSIM – Routes for Simulation

- railML 3.2 use case **candidates**
  - Asset Status Representation
  - Track Geometry
  - Driver Advisory System
  - Traffic Management System
  - ETCS Track Net
  - Infrastructure Recording

Complete list of use cases can be found in: https://wiki3.railml.org/index.php?title=UC:Use_cases

Use Case Working Groups

Final Selection at railML Conference in Linz (03.04.2019)

- railML 3.1 use cases
  - SCTP – Schematic Track Plan
  - NEST – Network Statement
  - RSIM – Routes for Simulation

- railML 3.2 use case **candidates**
  - Asset Status Representation
  - **Track Geometry**
  - Driver Advisory System
  - **Traffic Management System**
  - **ETCS Track Net**
  - Infrastructure Recording

Complete list of use cases can be found in: https://wiki3.railml.org/index.php?title=UC:Use_cases

railML Use Case Development

- Initiating the use case working group

railML Use Case Development

- Use case working group writes the **informal use case description** → to be put into the wiki
  - If use case description already exists, this has to be reviewed and updated

Use case / Anwendungsfall / Scénario d'utilisation

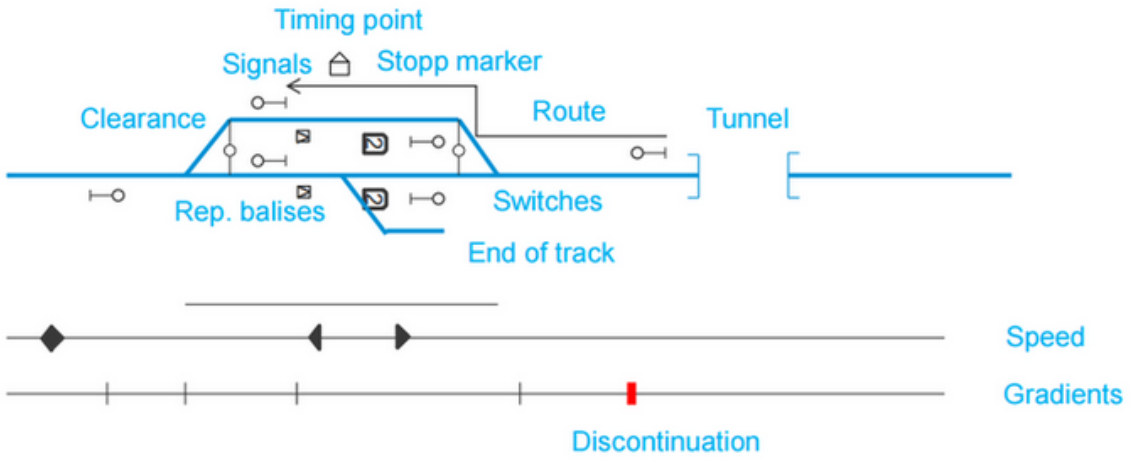
Schematic Track Plan

Description / Beschreibung / Description

The infrastructure manager uses maps for the visualization of their railway infrastructure. These maps comprise:

- Geographic maps
- Geoschematic maps / drawings
  - Operational points have geodetic coordinates
  - In between, the line layout is schematically interpolated
- Schematic drawings
  - All elements have screen coordinates resulting from their complete schematic layout.

The following figure depicts an example for a schematic drawing:

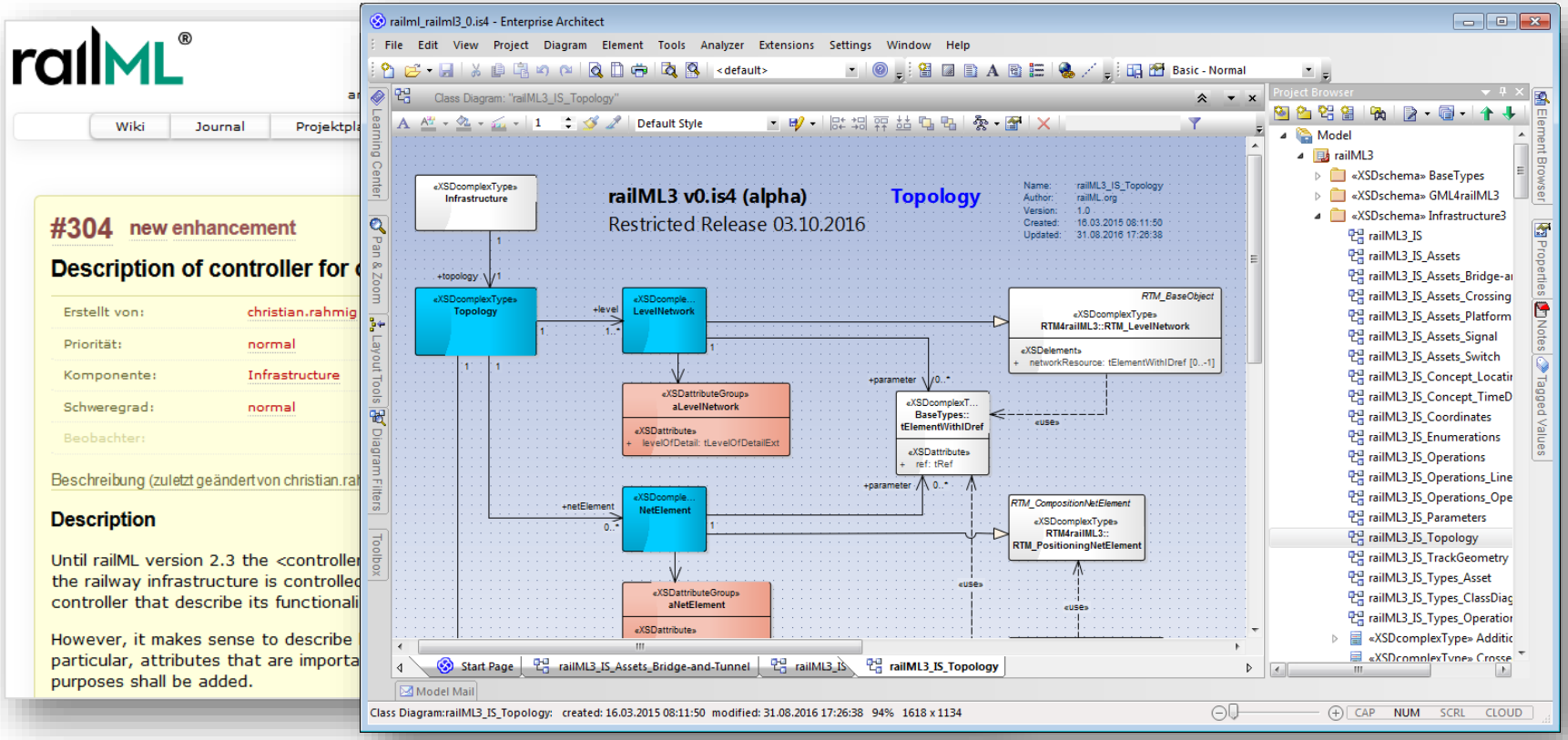


railML Use Case Development

- Check **which use case requirements can be already fulfilled** with existing railML 3.1
  - Derive list of requirements that have to be implemented in the railML 3.2 data model
- Discuss possible solutions for requirement implementation in the working groups and in the **forum**

railML Use Case Development

- Scheme coordinators derive **Trac ticket** and do the **UML model implementation**



railML Use Case D

- Set up the **element / attribute specification** (Excel file)

railML 3 Development Planning	Version Planning		lne Ca	Comments
Infrastructure				
	type			
Date: 30.10.2016	Time Plan	alpha		
Author: Christian Rahmig		beta		
		release candidate		
		release	SCTP	
Business Use Cases				
Topology			m	
Micro-Topology: Track + TrackNodes			m	
Meso-Topology: SectionOfLine			o	
Macro-Topology: Line + OPs			m	
Location referencing			m	
Point location			m	
Linear location			m	
Area location			o	
Orientation			m	
Positioning			m	
WGS84 coordinates 2D			o	
WGS84 coordinates 3D			o	
Mileage (general)			m	[km]
Mileage incl. special issues (mileageChanges)			m	[km]
coord system identification via EPSG code			o	
geodetic coordinates horizontal (EPSG code)			o	
geodetic coordinates vertical (EPSG code)			o	
screen coordinates (2D)			m	[x,y]
screen coordinates (3D)			x	[x,y,z]
Geometry			m	
Curvature			x	[1/m]
Gradient			m	[Promille]
Superelevation			x	[mm]
Point-based geometry			o	
Way-based geometry			o	

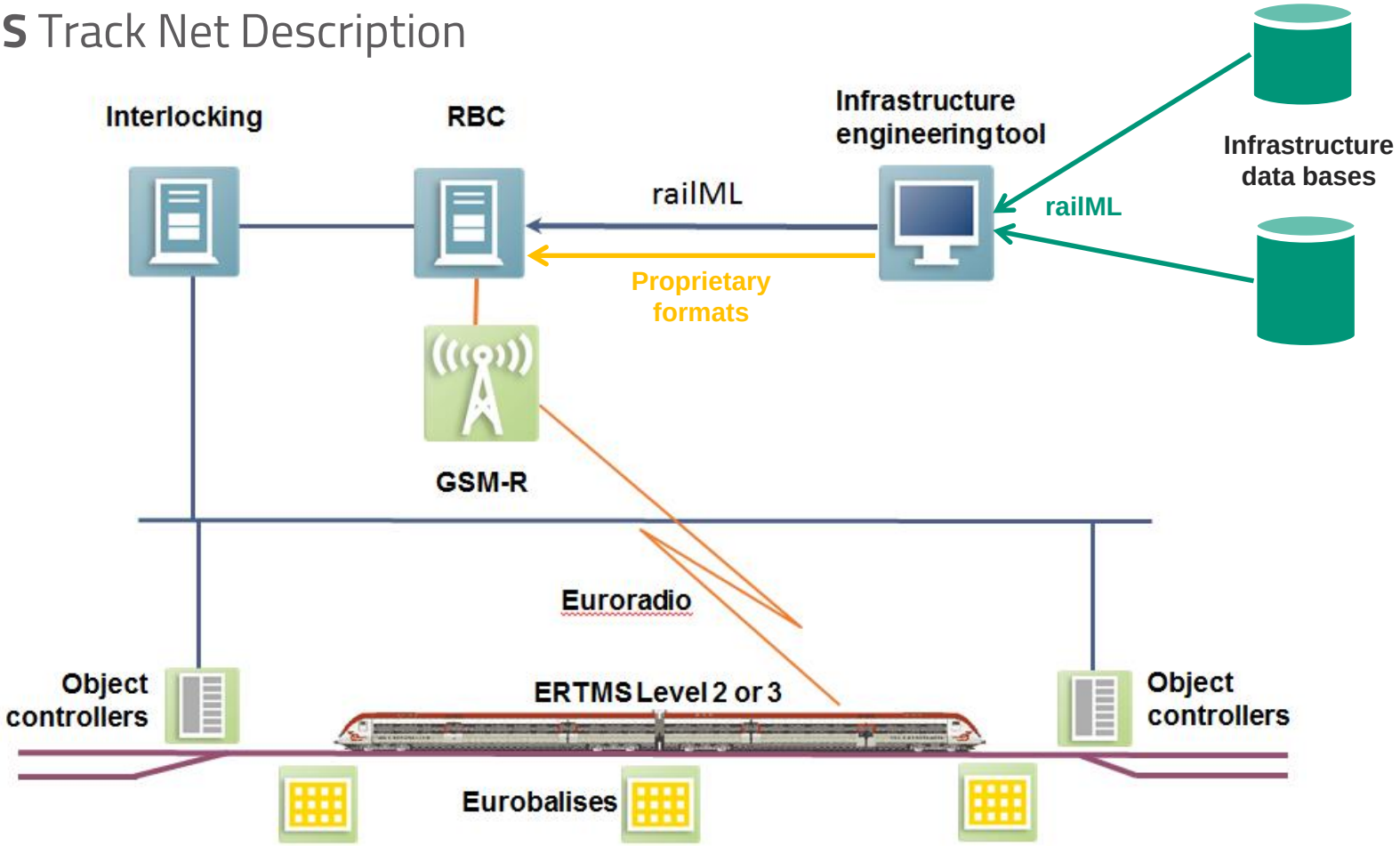
railML.org

railML Use Case Development

- Initiating the use case working group
- Use case working group writes the **informal use case description** → to be put into the wiki
  - If use case description already exists, this has to be reviewed and updated
- Check **which use case requirements can be already fulfilled** with existing railML 3.1
  - Derive list of requirements that have to be implemented in the railML 3.2 data model
- Discuss possible solutions for requirement implementation in the working groups and in the **forum**
- Scheme coordinators derive **Trac ticket** and do the model implementation
- Set up the **element / attribute specification** (Excel file)

Use Case Working Groups

- **ETCS** Track Net Description



Wiki: https://wiki3.railml.org/wiki/UC:IS:ETCS_track_net

Use Case Working Groups

- **ETCS** Track Net Description
  - Work agenda:
    - ✓ Finalizing table with functional elements and parameters
    - ⏳ Adding consolidated list of functional elements and parameters to use case wiki page
    - ✓ Comparison with railML 3.1 to identify already existing railML elements and attributes being usable for storing the functional information
    - ✓ Extending the model with missing elements and attributes
    - ✓ Filling the element specification table
    - ⏳ Prepare official use case document
    - Documentation of new elements and attributes in railML wiki
  - Next phone call: May 3, 2021, 13h CEST

Use case „ETCS“ requirements are realised with railML 3.2 beta1

Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - TMS: optimize rail traffic flow by providing decision support to dispatchers as well as automated functions
  - Typical tasks:
    - automatic route setting,
    - real time train graph display,
    - conflict detection,
    - conflict resolution and
    - traffic regulation
    - real time track layout display,
    - train describer tracking and command interface for route setting and wayside object control
  - includes also real time data, such as positions of trains and states of wayside objects

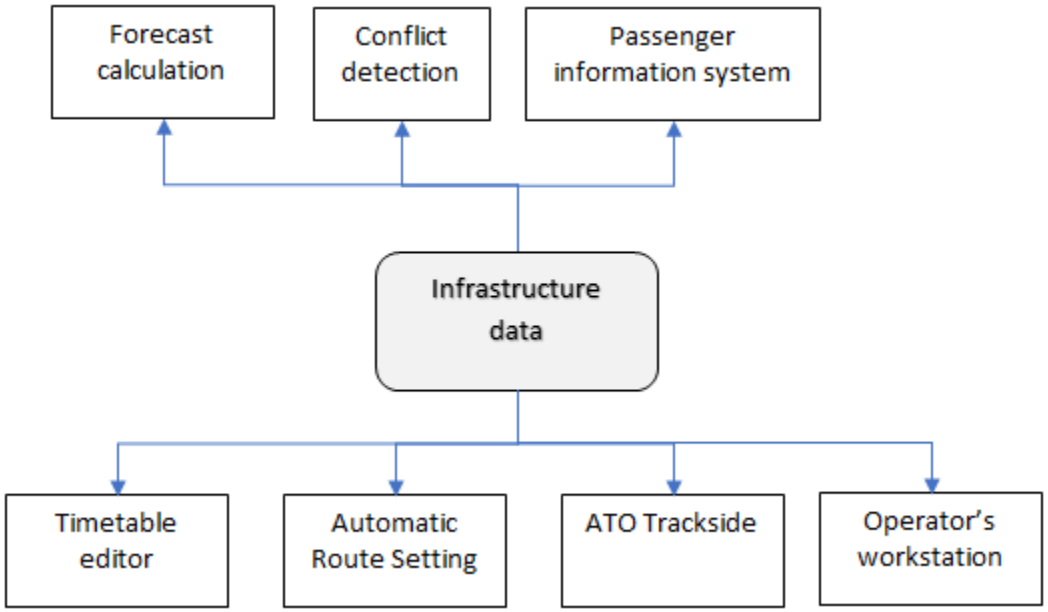
Wiki: https://wiki3.railml.org/index.php?title=UC:IS:IntegratedTMS

Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - Integrated TMS with operator-specific sandbox representations of real-time traffic situation
  - Focus: versioning of infrastructure and timetable information; scenarios; historic and forecasting information

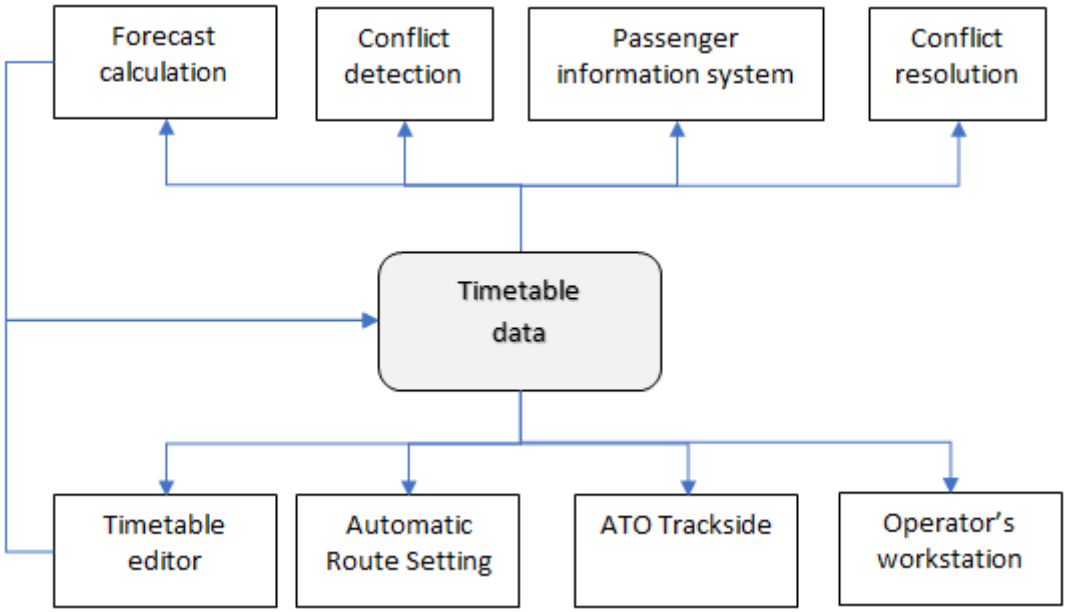
Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - Integrated TMS with operator-specific sandbox representations of real-time traffic situation
  - Focus: versioning of infrastructure and timetable information; scenarios; historic and forecasting information
  - Adresses **infrastructure**, timetable and interlocking information



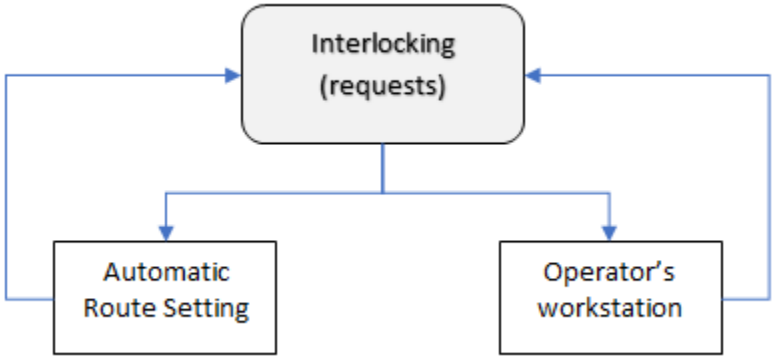
Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - Integrated TMS with operator-specific sandbox representations of real-time traffic situation
  - Focus: versioning of infrastructure and timetable information; scenarios; historic and forecasting information
  - Adresses infrastructure, **timetable** and interlocking information



Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - Integrated TMS with operator-specific sandbox representations of real-time traffic situation
  - Focus: versioning of infrastructure and timetable information; scenarios; historic and forecasting information
  - Adresses infrastructure, timetable and **interlocking** information



Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - Integrated TMS with operator-specific sandbox representations of real-time traffic situation
  - Focus: versioning of infrastructure and timetable information; scenarios; historic and forecasting information
  - Adresses **infrastructure**, **timetable** and **interlocking** information
  - Further: Adresses **real-time traffic** information

New schema?

Wiki: https://wiki3.railml.org/index.php?title=UC:IS:IntegratedTMS

Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - Work agenda:
    - ⏳ Identifying functional elements and their parameters that are relevant for this use case → listing them in scheme specific Excel files
      - Adding consolidated list of functional elements and parameters to use case wiki page
    - ⏳ Comparison with railML 3.1 to identify already existing railML elements and attributes being usable for storing the functional information
  - Next phone call: May 7, 2021, 14h CEST

Wiki: https://wiki3.railml.org/index.php?title=UC:IS:IntegratedTMS

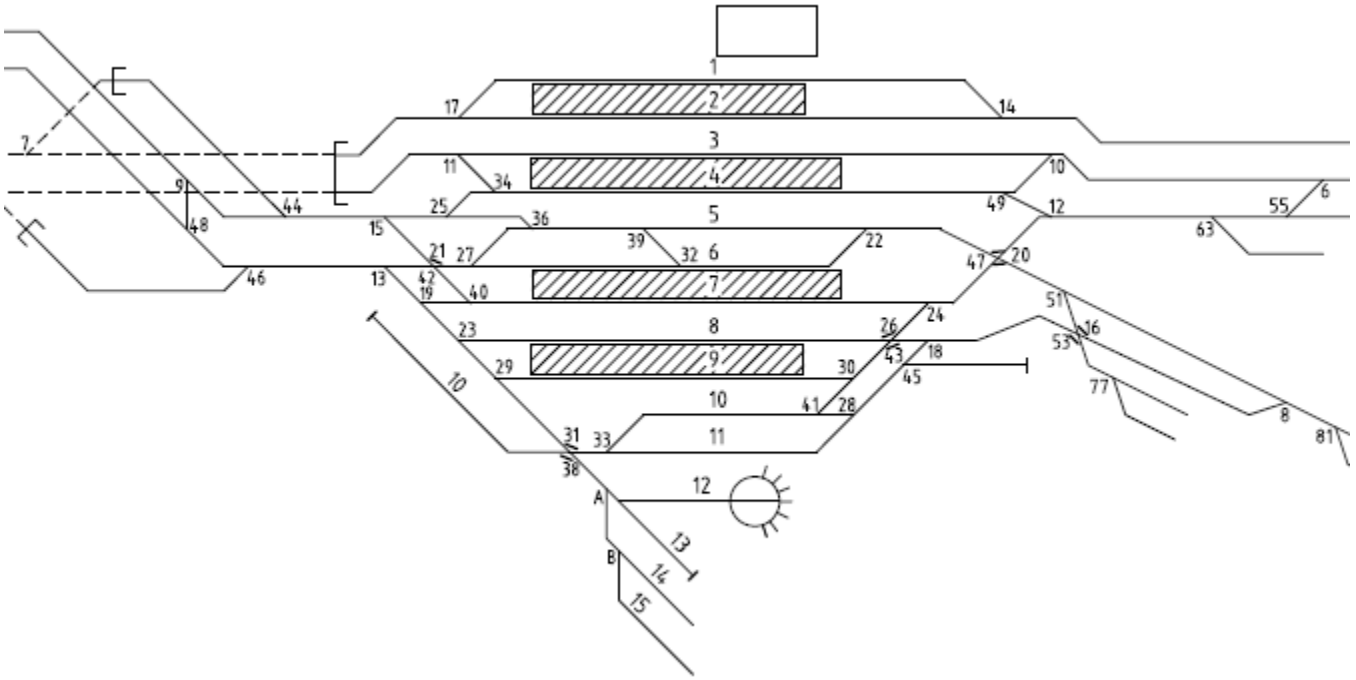
Use Case Working Groups

- Integrated Traffic Management System (**ITMS**)
  - Members:
    - Ansaldo / Hitachi
    - BaneNor
    - (Bombardier)
    - Siemens
    - Thales

Who else is interested in contributing?

Use Case Working Groups

- Schematic Track Plan (**SCTP**)
  - SCTP is already implemented in railML 3.1
  - Needed: update regarding graphical visualisation of infrastructure data



Wiki: https://wiki3.railml.org/index.php?title=UC:IS:Schematic_Track_Plan

Use Case Working Groups

- Schematic Track Plan (**SCTP**)
  - Work agenda:
    - ✓ Finalizing table with functional elements and parameters
    - ⏳ Adding updated consolidated list of functional elements and parameters to use case wiki page
    - ✓ Comparison with railML 3.1 to identify already existing railML elements and attributes being usable for storing the functional information
    - ✓ Extending the model with missing elements and attributes
    - ⏳ Updating the element specification table
    - Update official use case document
    - Documentation of new elements and attributes in railML wiki
  - Next phone call: April 23, 2021, 13h CEST

Wiki: https://wiki3.railml.org/index.php?title=UC:IS:Schematic_Track_Plan

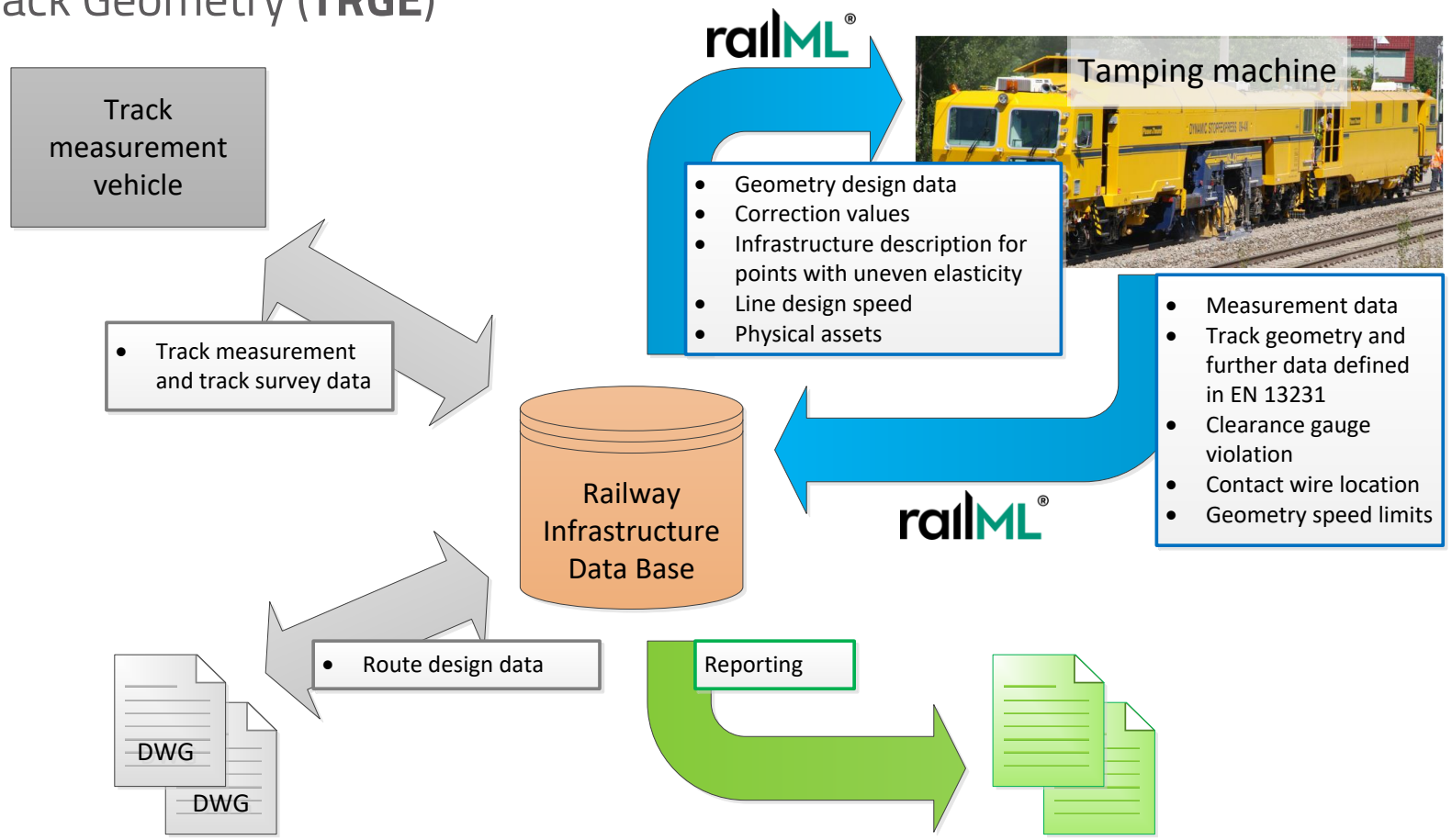
Use Case Working Groups

- Schematic Track Plan (**SCTP**)
  - Members:
    - Ansaldo / Hitachi
    - Neat
    - PSI

Who else is interested in contributing?

Use Case Working Groups

- Track Geometry (**TRGE**)



Wiki: https://wiki3.railml.org/wiki/UC:IS:Track_Geometry

Use Case Working Groups

Currently paused

- Track Geometry (**TRGE**)
  - Work agenda:
    - Reviewing use case description in railML wiki
    - Finalizing table with functional elements and parameters
    - Adding consolidated list of functional elements and parameters to use case wiki page
    - Comparison with railML 3.1 to identify already existing railML elements and attributes being usable for storing the functional information
    - Extending the model with missing elements and attributes
    - Filling the element specification table
    - Prepare official use case document
    - Documentation of new elements and attributes in railML wiki
  - Next phone call: not scheduled…

Wiki: https://wiki3.railml.org/wiki/UC:IS:Track_Geometry

Use Case Working Groups

- Track Geometry (**TRGE**)
  - Members:
    - CVUT
    - Infrabel
    - Plasser & Theurer
    - Railcomplete
    - SZDC

Who else is interested in contributing?

Modelling railML 3.2

Overview on proposed developments in infrastructure

Overview

- *Description of controller for operational purposes (#304) → IL*
- railML data in one vs splitted files (#325)
- *External IDs and References (#363) → CO*
- Extending enumeration for track condition areas (#365)
- Extending the <balise> element (#366)
- Extending the <speedProfile> element (#367)
- Definition of a Track (#368)
- Track length (#369)
- Visualizations (#370)
- Extending the Level Crossing Model (#377)
- […]

Overview

- […]
- *Dublin Core schema moved from http to https (#379) → CO*
- Adding branches to crossing (#380)
- *Metadata for revision management (#382) → CO*
- *Radio Block Center (#386) → IL*
- *Description field for train numbers (#388) → IL*
- Natural hazards detection (area) (#422) → CO
- Introduce <platformEdge> (#438)
- NID_CTRACTION for electrification model (#439)
- Extension of <opEquipment> (#441)
- Transfer times for connections (#442)
- […]

Overview

- […]
- Re-introducing @ruleCode? (#443)
- *Identification of preferred route path (#444) → IL*
- Bascule bridge (#449) → IL
- *Tunnel gates (#450) → IL*
- *Operator commands and indications (#451) → IL*
- Driving directions in macroscopic nodes (#452)
- *Level crossing nominal rise time (#453) → IL*
- Stopping places and platform edges (#454)
- *Remodel organizationalUnits (#456) → CO*
- Extension of Communication Settings (#457) → IL
- […]

Overview

- […]
- ETCS signal modeling update (#459)
- TrainProtectionElement vs ETCS (#460)
- Loading gauge profiles (#461)
- Tunnel Gate in Infrastructure (#466)

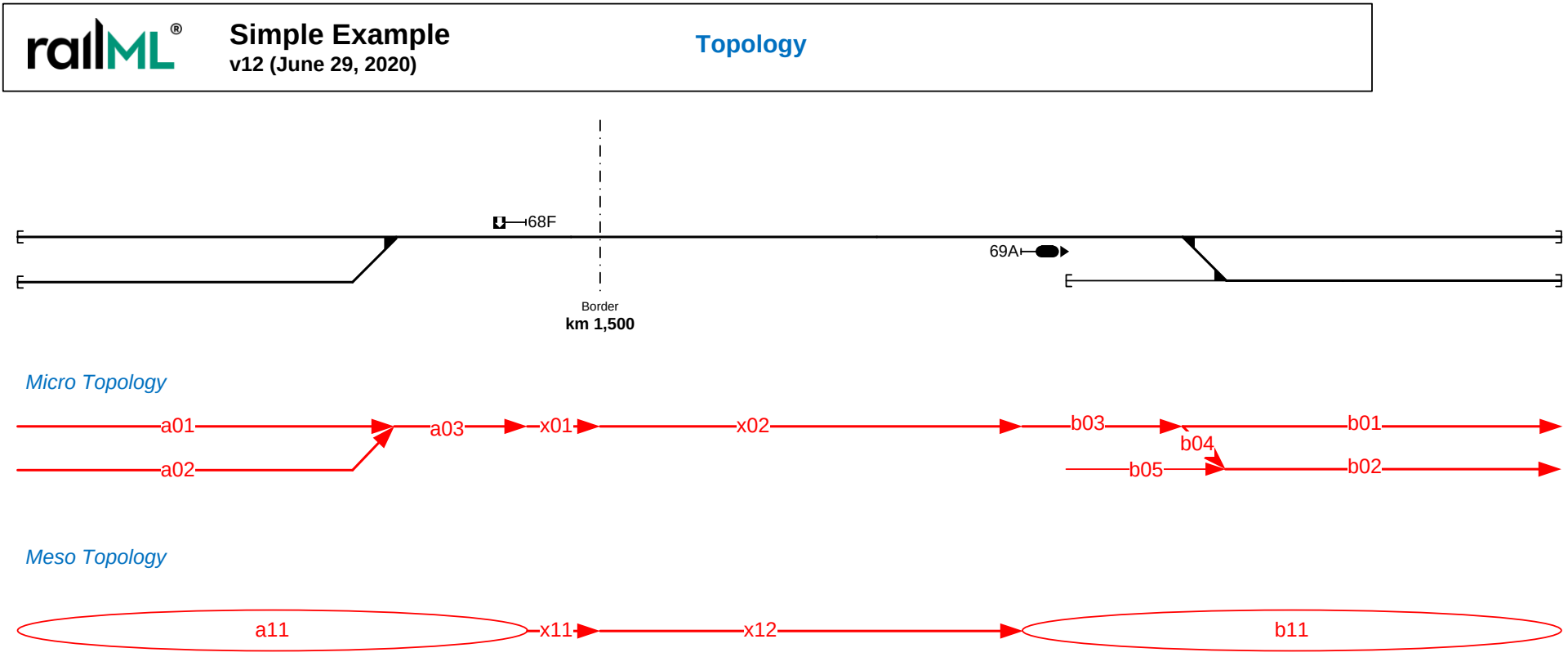
#325: railML data in one vs splitted files

- 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://trac.railml.org/ticket/325
Wiki:

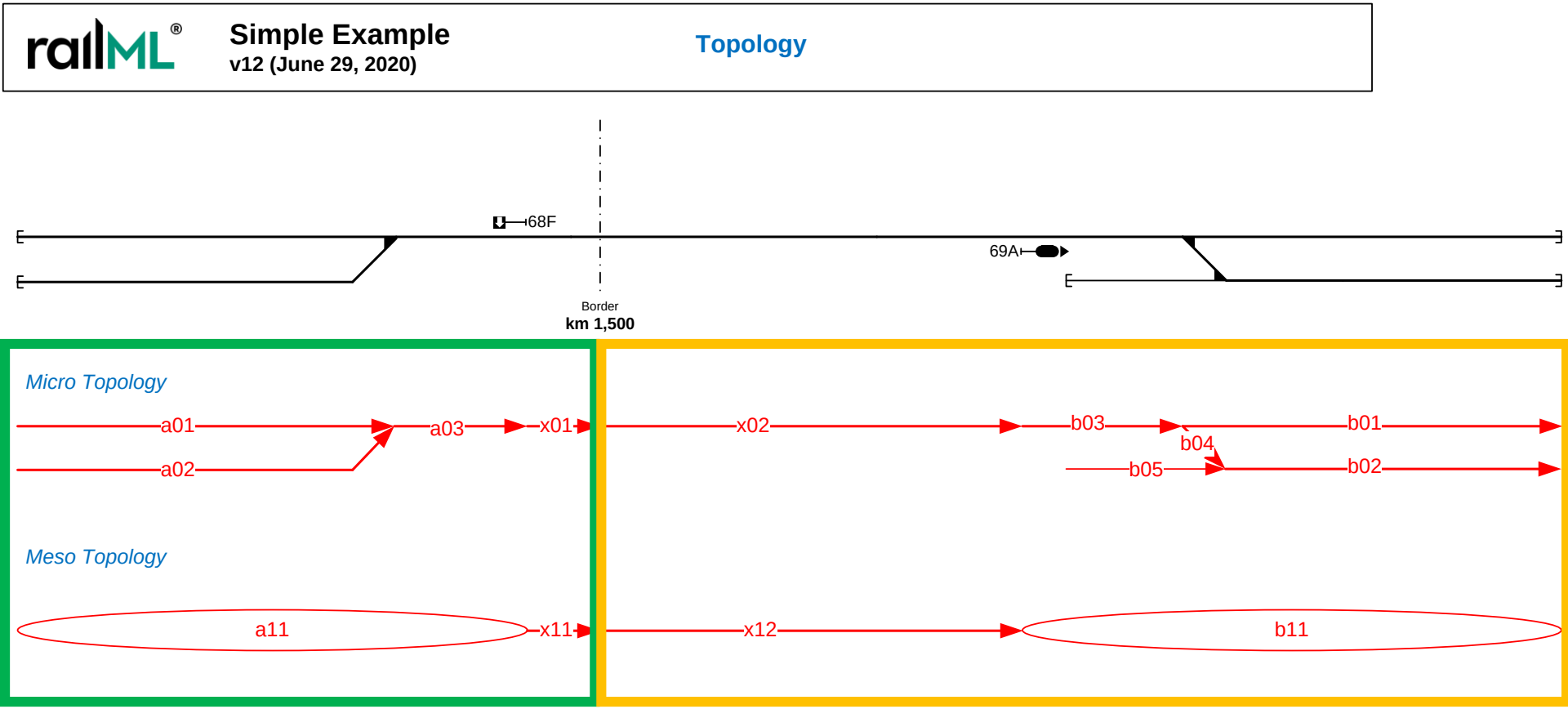
Splitted Infrastructure

- Splitting of infrastructure networks starts at topology level



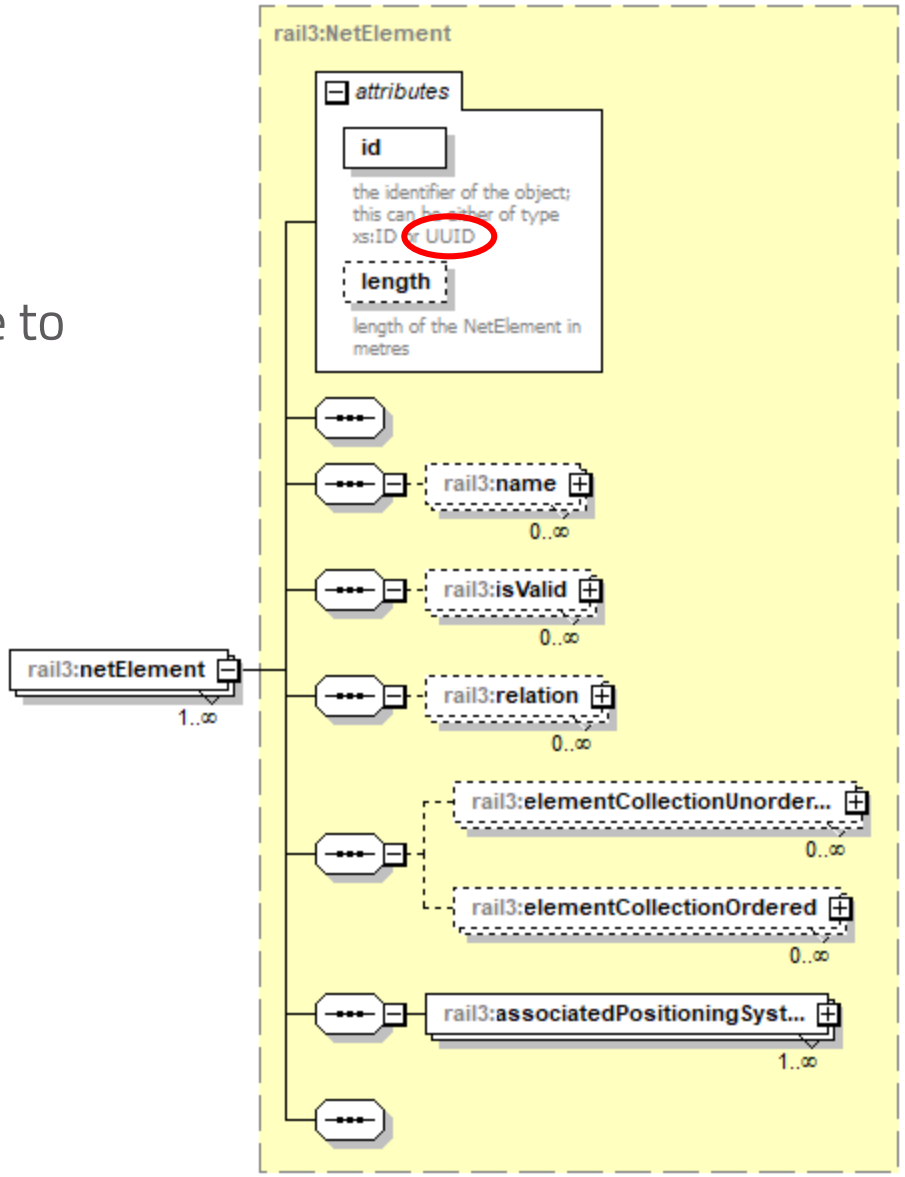
Splitted Infrastructure

- Splitting of infrastructure networks starts at topology level



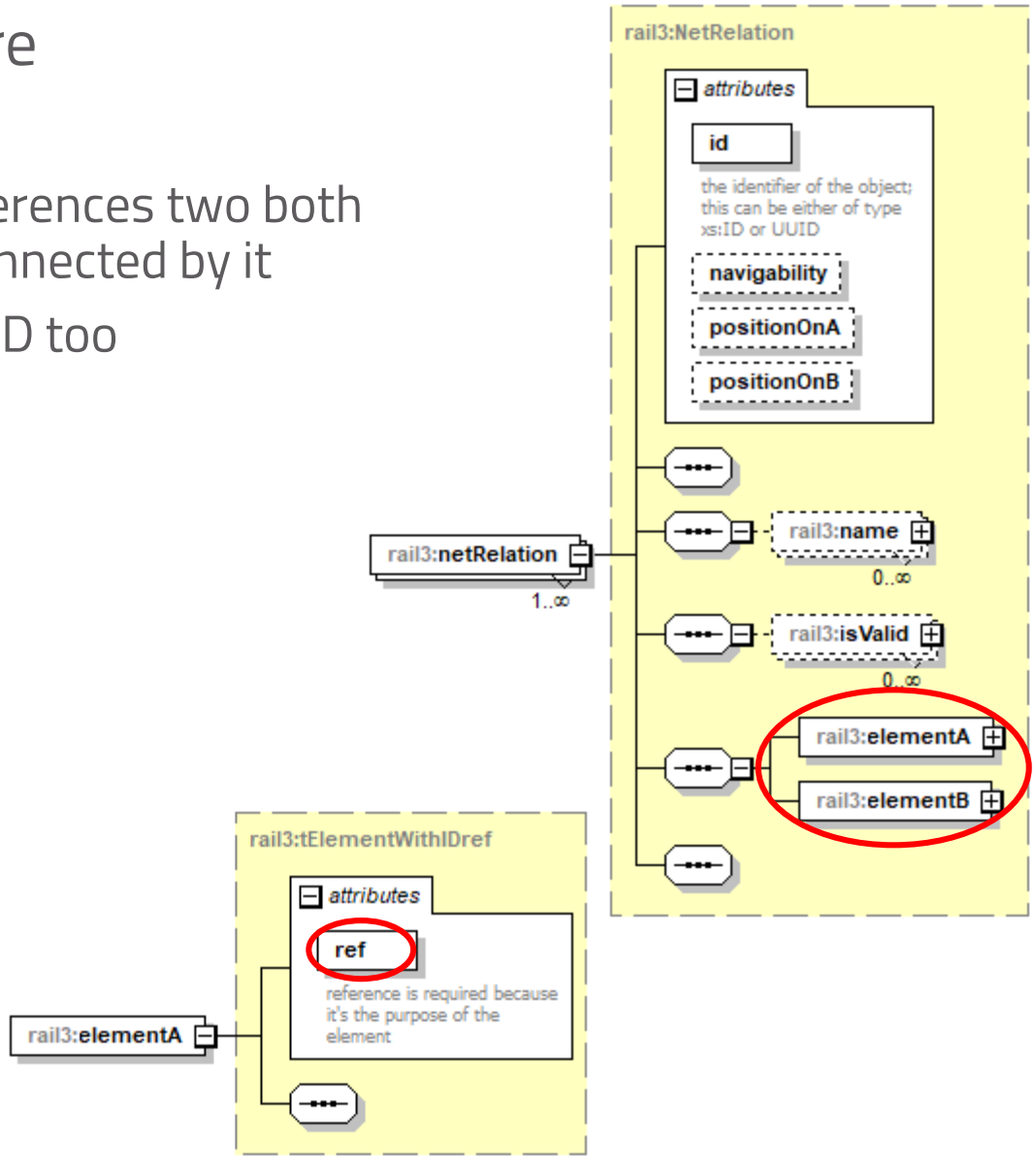
Splitted Infrastructure

- 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



Splitted Infrastructure

- To be discussed on basis of best practices / examples:
  - Are redundant NetElements / NetRelations necessary?
    - Option 1: linking NetRelation in both files
    - Option 2: linking NetRelation only in one of the two files
  - **How about elements on top of topology, e.g. tracks, lines or operationalPoints?**
    - Option 1: strict cut over all layers
    - Option 2: allow for „cross file elements“ with location „outside“ of the file

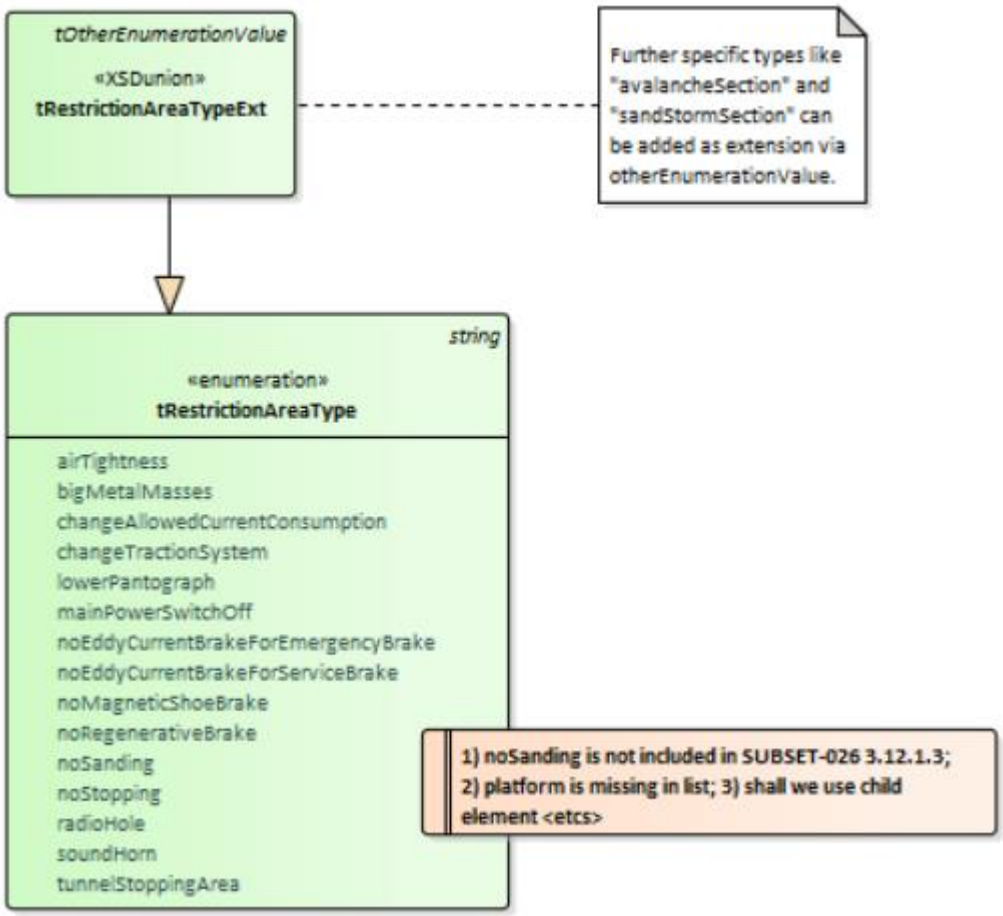
#365: Extending enumeration for track condition areas

- 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://trac.railml.org/ticket/365
Wiki:

#365: Extending enumeration for track condition areas

- The model:



#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

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://trac.railml.org/ticket/366; https://trac.railml.org/ticket/174
Wiki:

#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>@isDuplicate**

#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** (float; -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

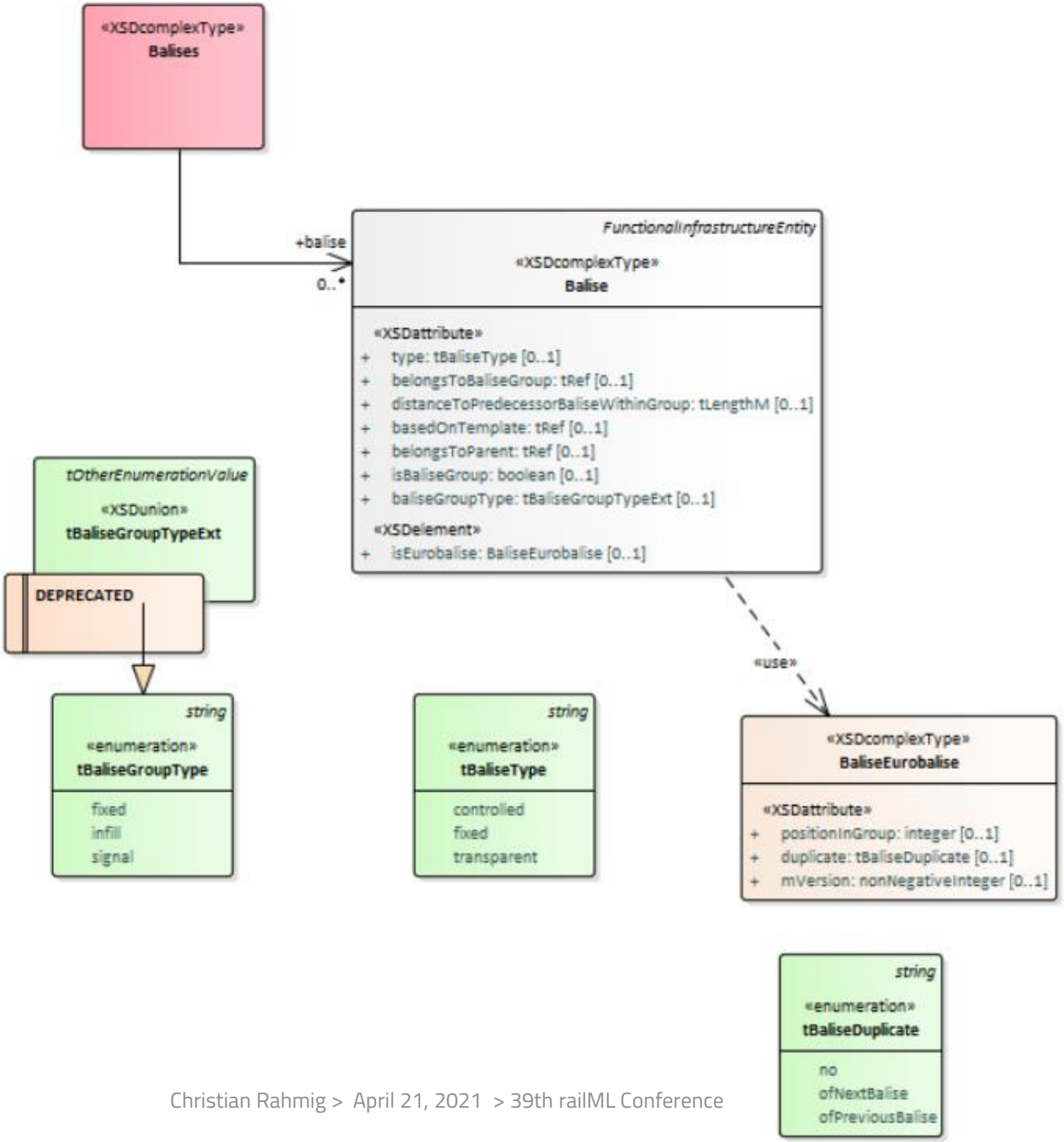
#366: Extending the <balise> element

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

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://trac.railml.org/ticket/366; https://trac.railml.org/ticket/174
Wiki:

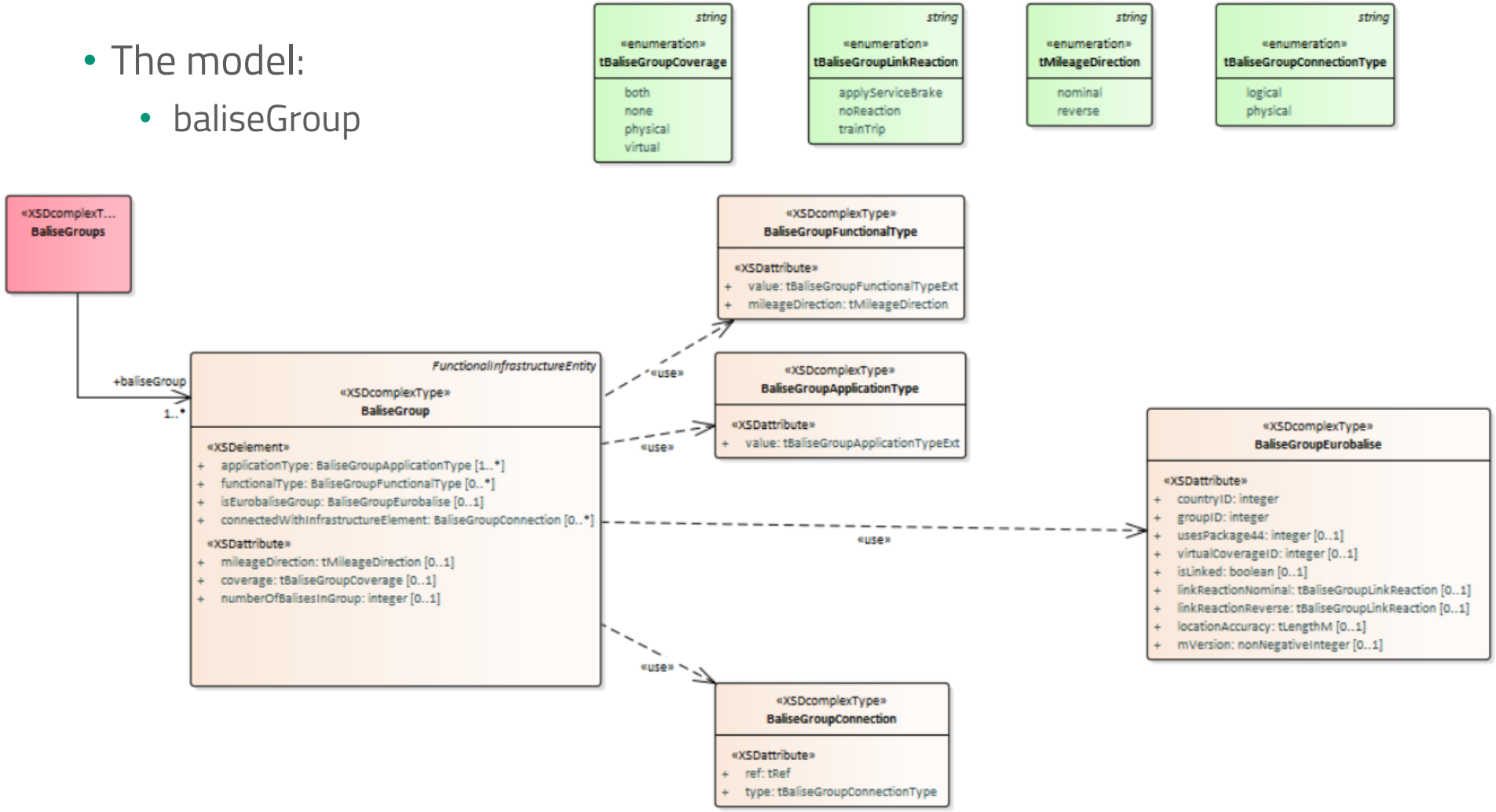
#366:

- The model:
  - balise



#366: Extending the <balise> element

- The model:
  - baliseGroup



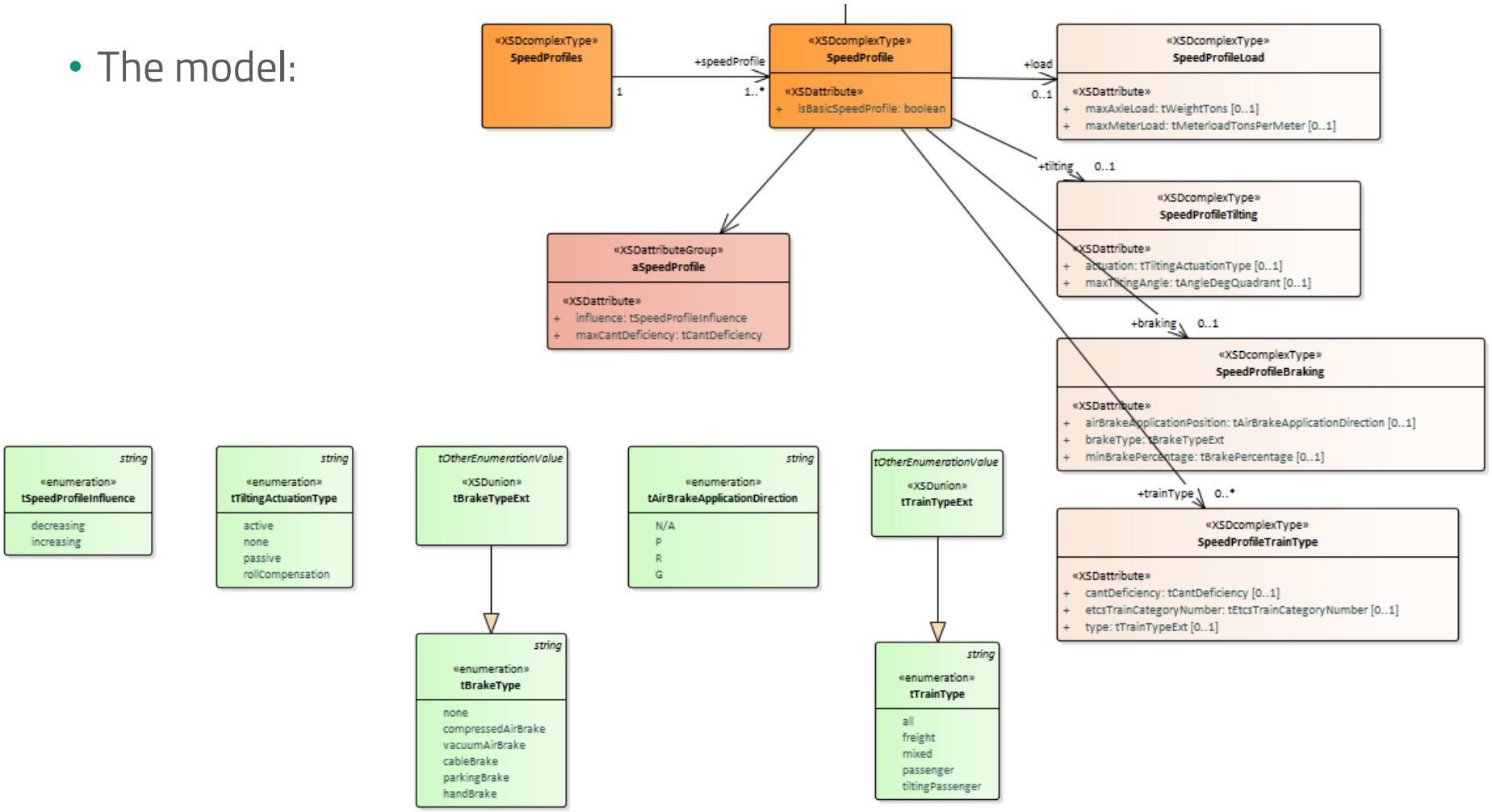
#367: Extending the <speedProfile> element

- 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..*
  - Deprecate **<trainType>**@cantDeficiency

Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=686&start=0&
Trac: https://trac.railml.org/ticket/367
Wiki:

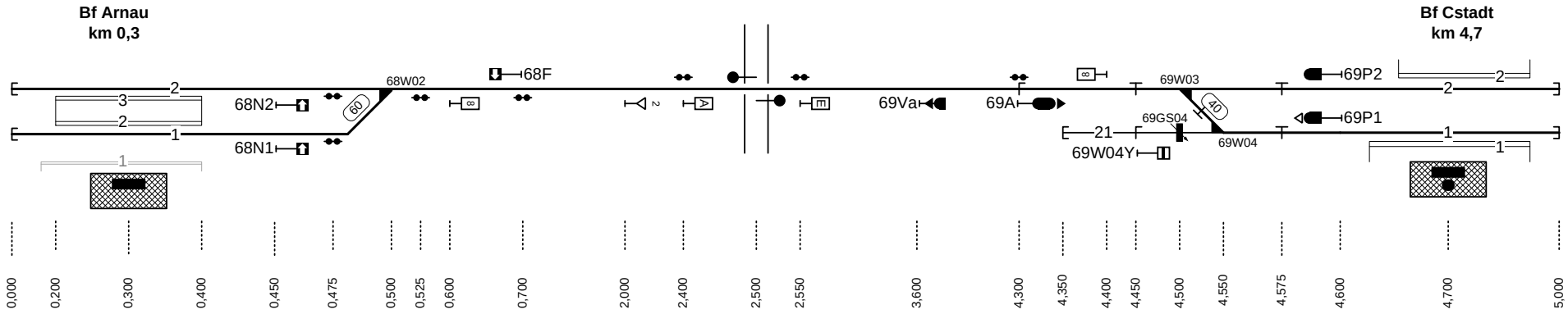
#367: Extending the <speedProfile> element

- The model:



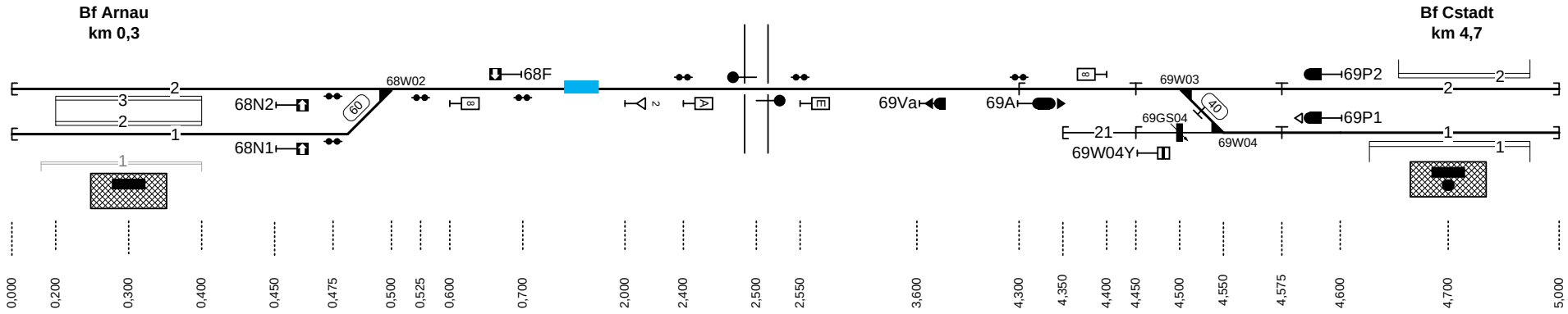
#368: Definition of a Track

- 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



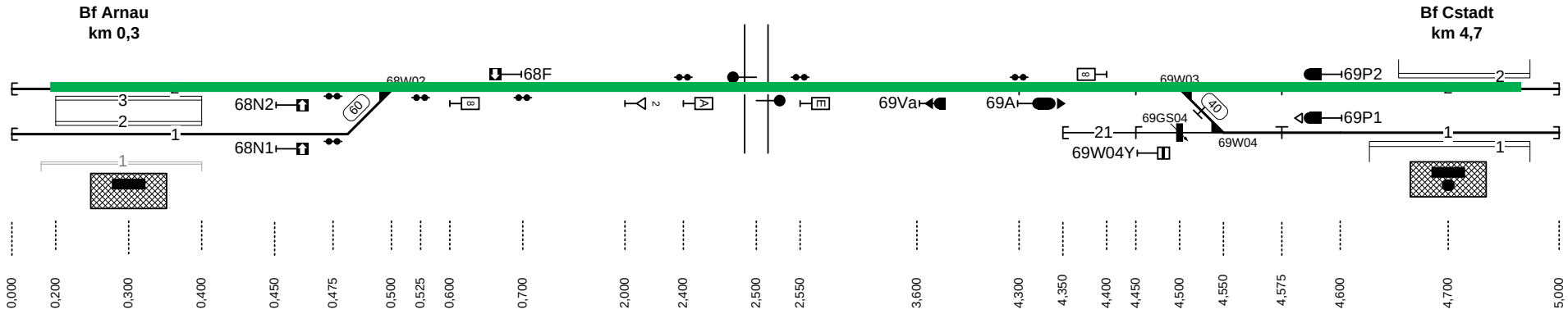
#368: Definition of a Track

- 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://trac.railml.org/ticket/368
Wiki:

#369: Track length

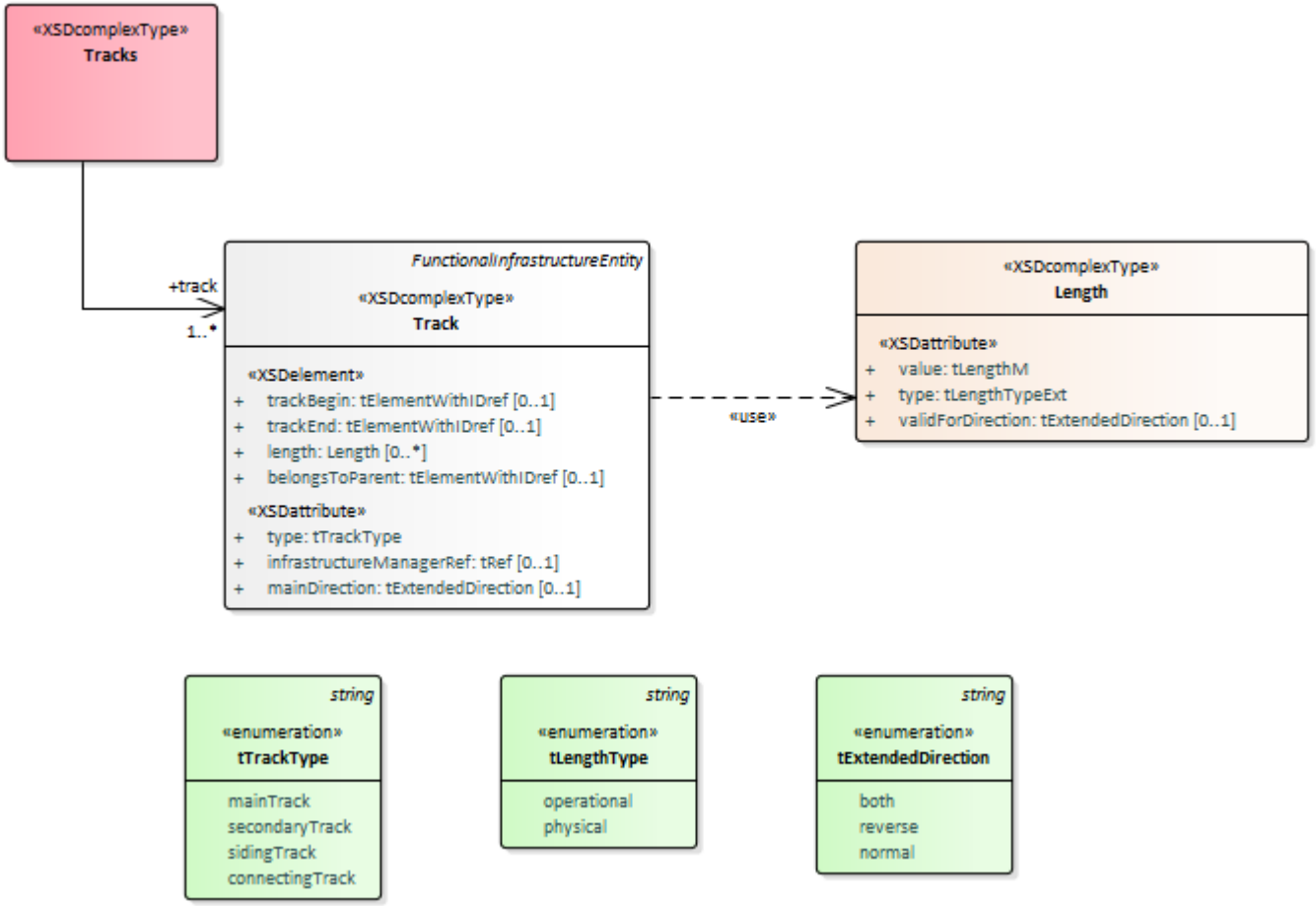
- The situation:
  - In railML 3.1 each <track> element has to have at least one child element <length>
- Idea:
  - Make **<track><length>** completely optional, because there are use cases (e.g. in timetable) that don't need the length information → change cardinality of <track / length> from 1..* to 0..*

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

Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=678&start=0&
Trac: https://trac.railml.org/ticket/369
Wiki:

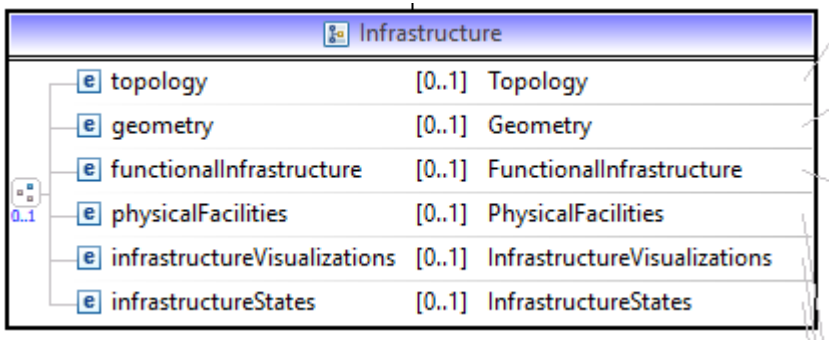
#369: Track length

- The model:



#370: Visualizations

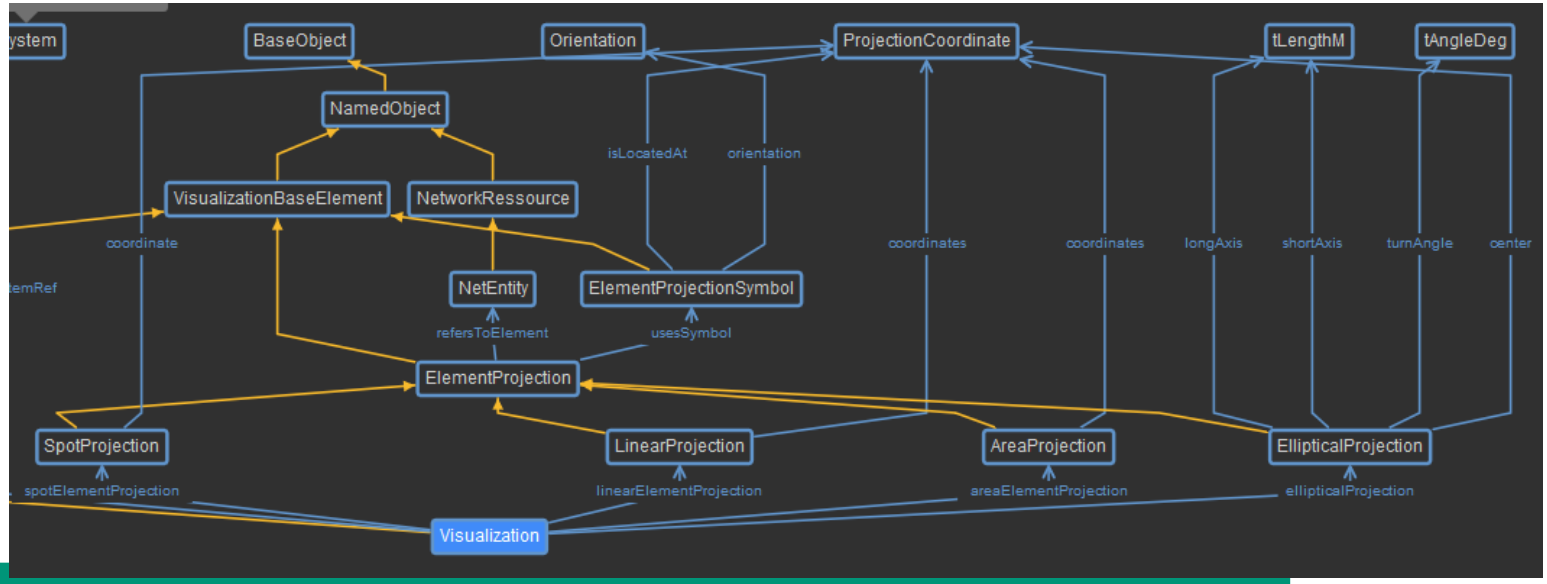
- The situation:
  - railML 3.1 contains <infrastructureVisualization> scheme that is used to model graphical visualizations of the infrastructure
  - How about visualization of timetable, rollingstock, interlocking elements?
- Idea:
  - Generalize the concept of visualizations in new schema <visualizations>



Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=683&start=0&
Trac: https://trac.railml.org/ticket/370
Wiki:

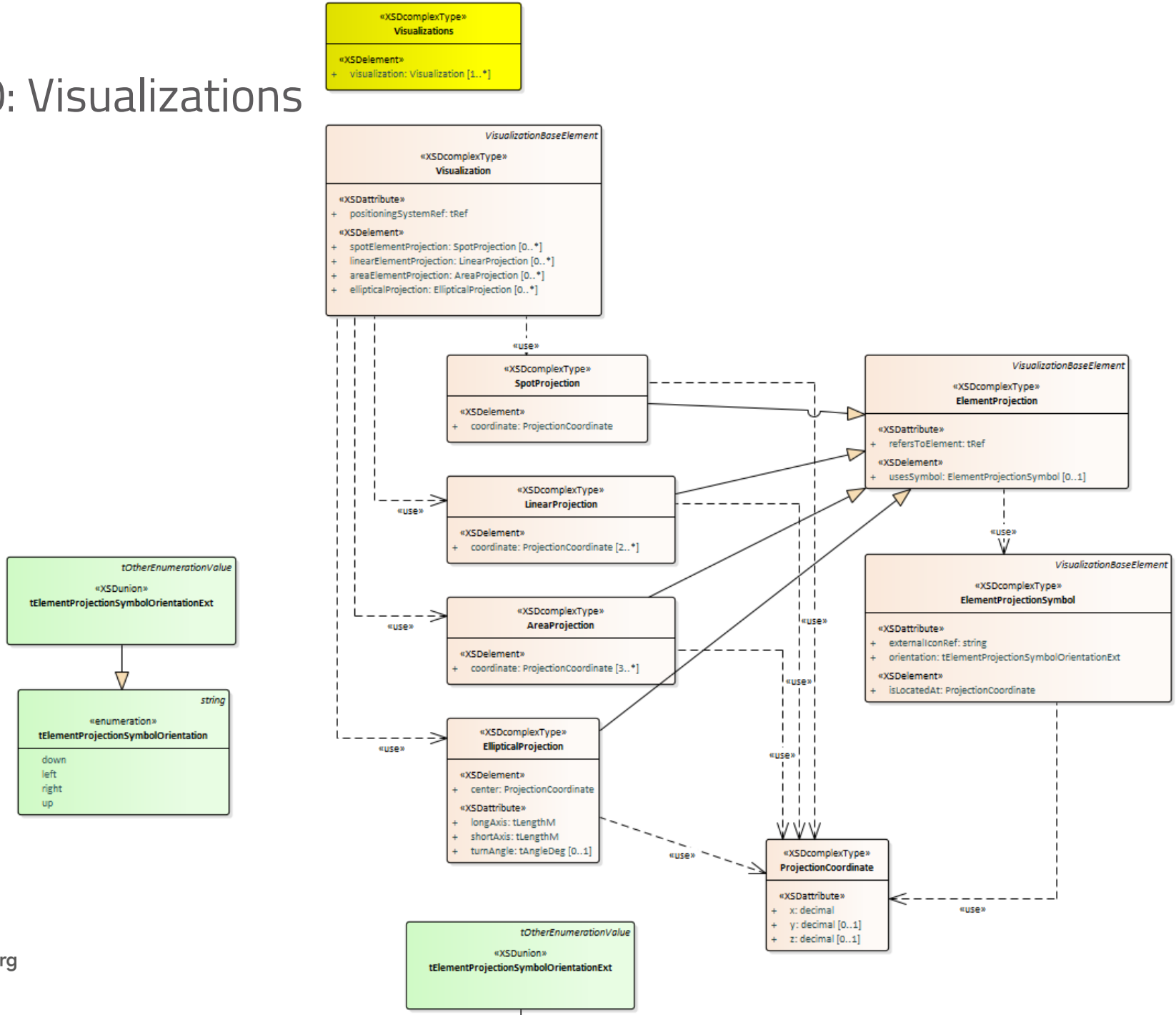
#370: Visualizations

- Solution:
  - New sub-schema <visualizations>
  - Add new child element <ellipticalProjection> to visualize circular and elliptical elements



Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=683&start=0&
Trac: https://trac.railml.org/ticket/370
Wiki:

#370: Visualizations



railML.org

#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://trac.railml.org/ticket/377
Wiki: https://wiki3.railml.org/wiki/IS:levelCrossingIS

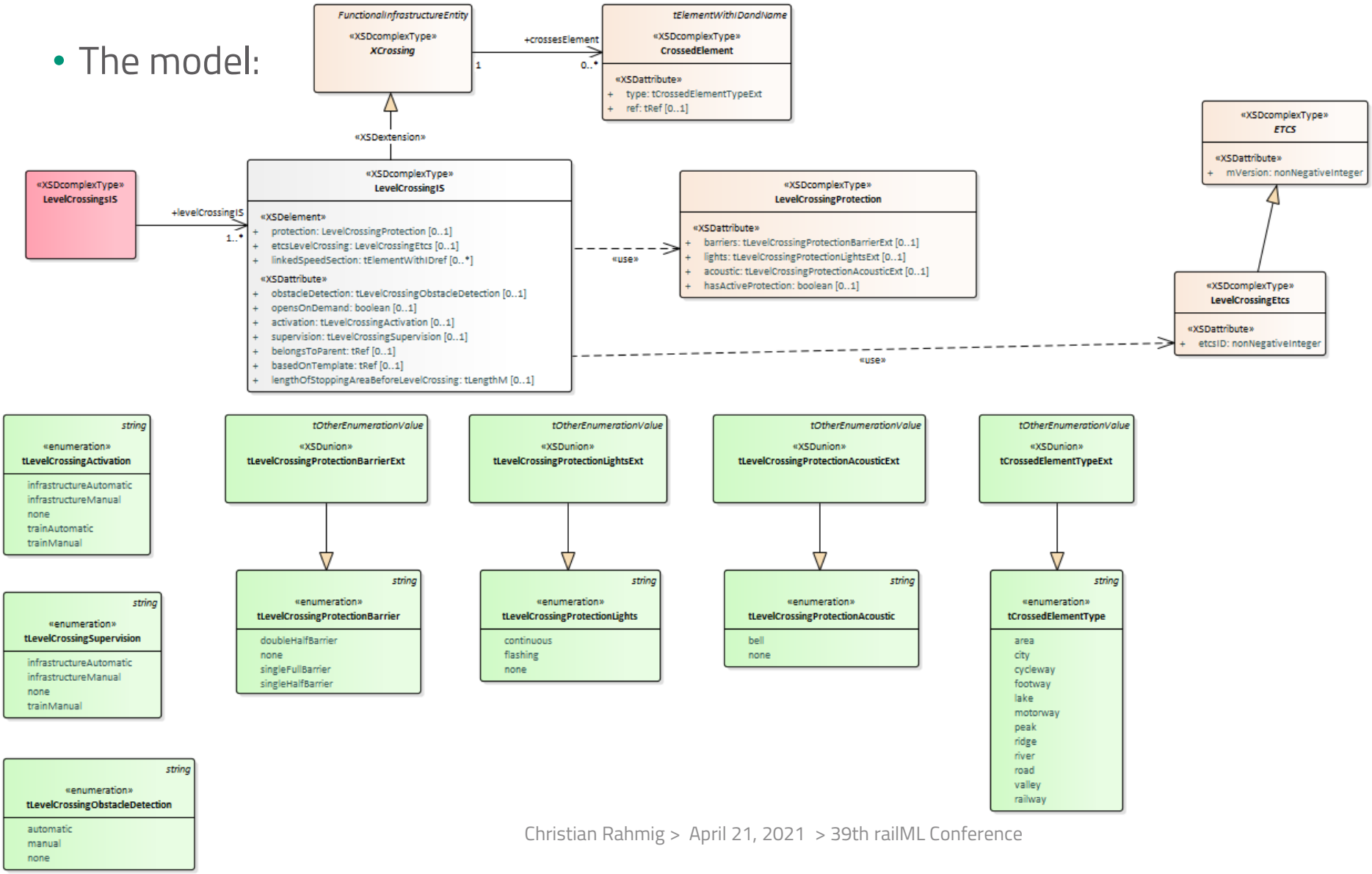
#377: Extending the Level Crossing Model

- Ideas:
  - 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://trac.railml.org/ticket/377
Wiki: https://wiki3.railml.org/wiki/IS:levelCrossingIS

#377: Extending the Level Crossing Model

- The model:



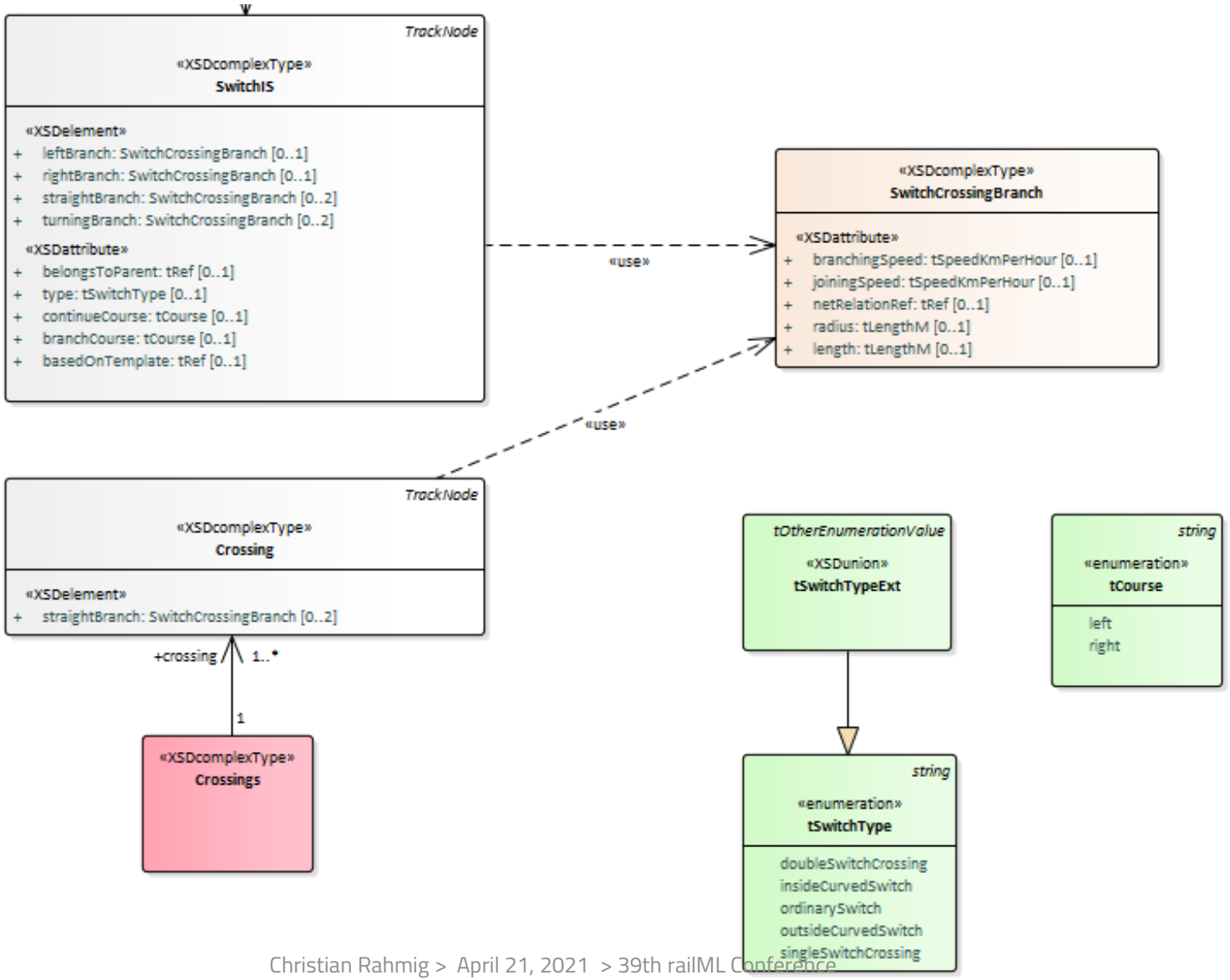
#380: Adding branches to crossing

- 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

Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=728&start=0&
Trac: https://trac.railml.org/ticket/380
Wiki:

#380: Adding branches to crossing

- The model:



#386: Radio Block Center

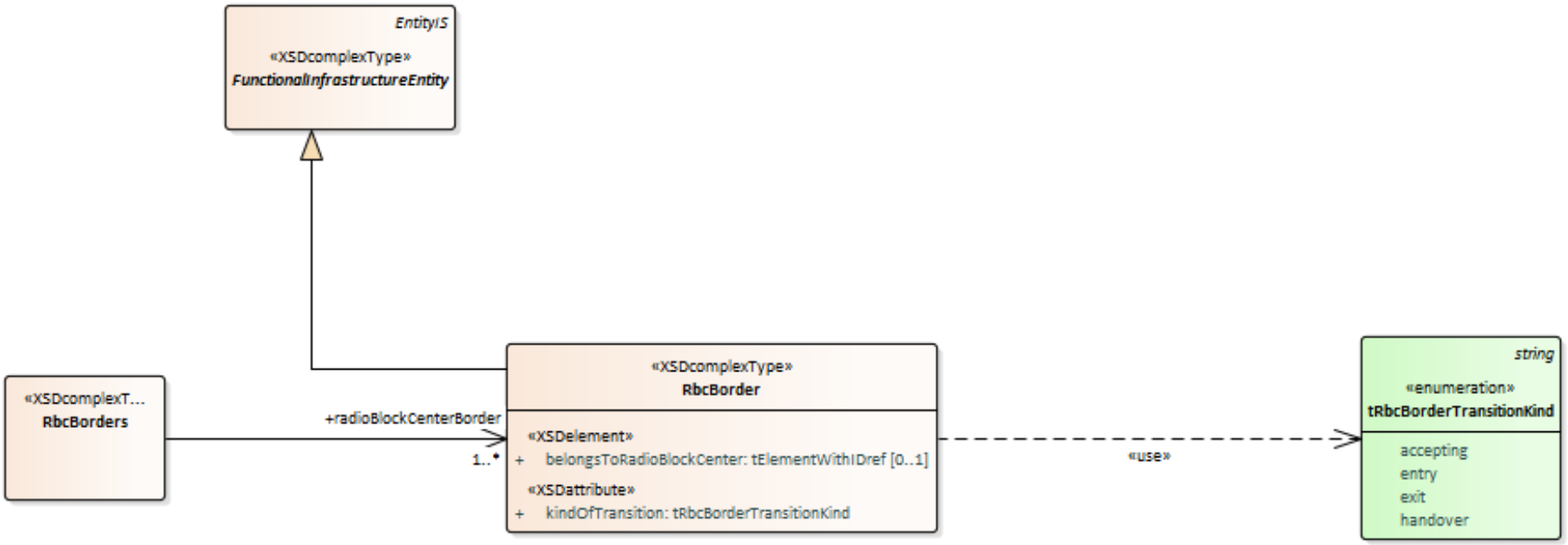
- Situation:
  - railML 3.1 data model is missing the Radio Block Centre (RBC) required by ETCS related applications

- 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://trac.railml.org/ticket/386
Wiki:

#386: Radio Block Center

- The model:
  - <radioBlockCentreBorder>



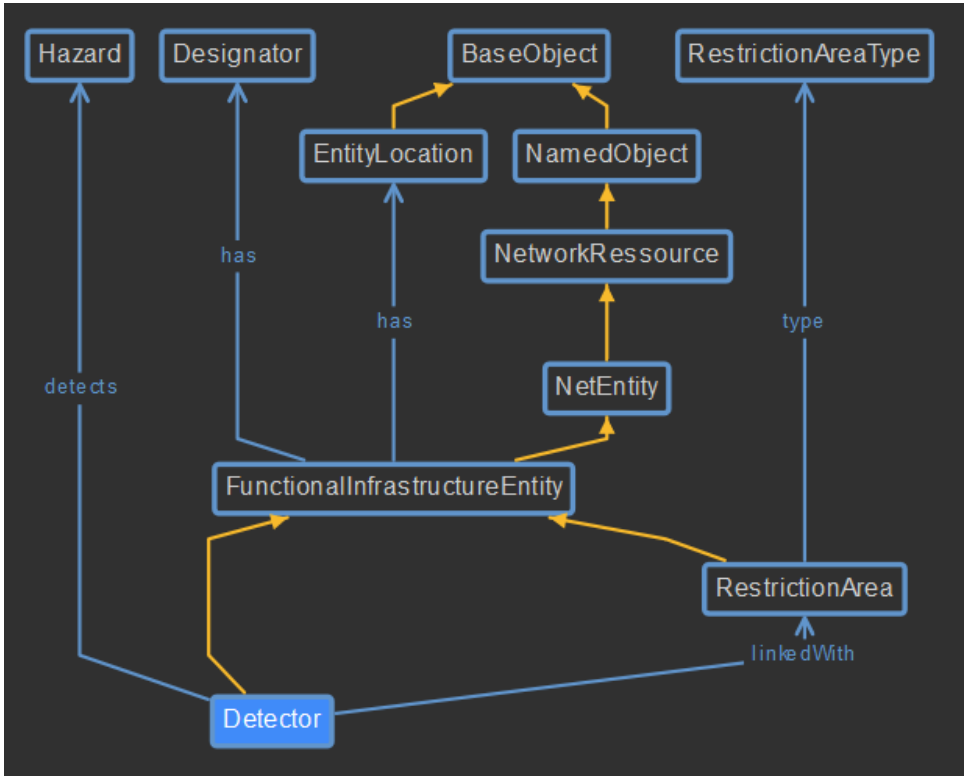
#422: Natural hazards detection

- 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 restriction area that defines a certain operational reaction (e.g. „noStopping“) on the detected hazard
  - How about **<protectionSystem>** that @protectsAgainstHazard?

Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=791&start=0&
Trac: https://trac.railml.org/ticket/422
Wiki:

#422: Natural hazards detection

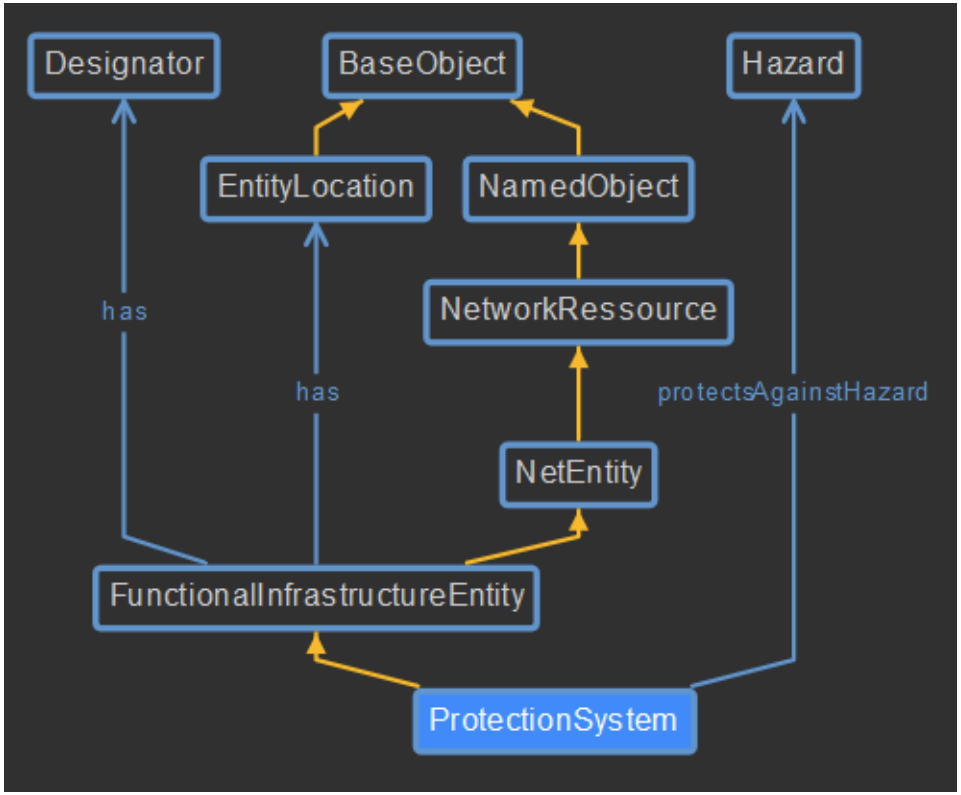
- Solution:
  - Detecting hazards



Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=791&start=0&
Trac: https://trac.railml.org/ticket/422
Wiki:

#422: Natural hazards detection

- Solution:
  - Protection against hazards (e.g. fences)



Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=791&start=0&
Trac: https://trac.railml.org/ticket/422
Wiki:

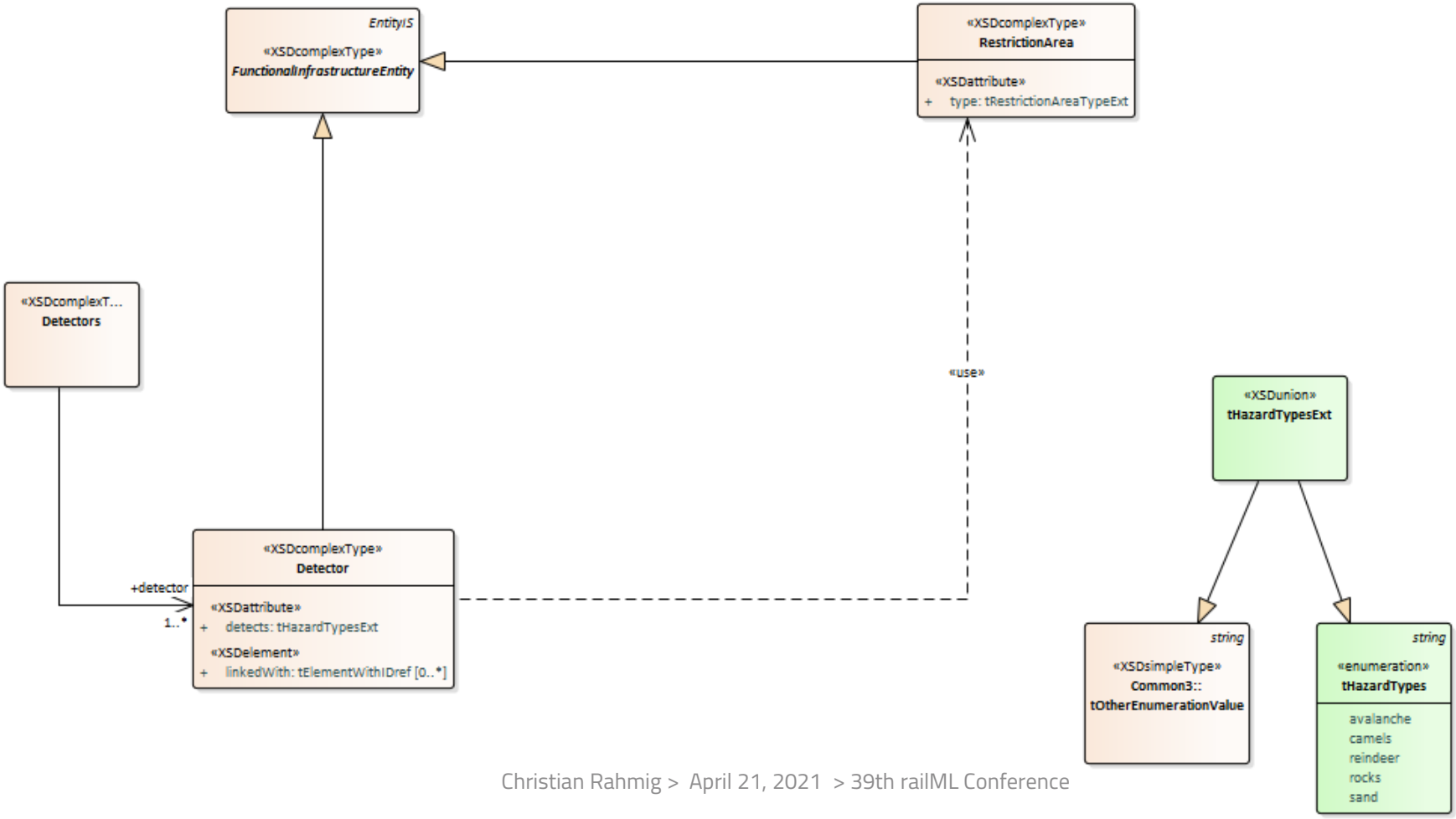
#422: Natural hazards detection

- **Open questions**:
  - Is there a need for <protectionSystems>?
  - Shall bridges and tunnels be linked with detectors or protection systems if installed?
  - How about TT, RS and other views on the topic of natural hazard detection?

Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=791&start=0&
Trac: https://trac.railml.org/ticket/422
Wiki:

#422: Natural hazards detection

- The model:
  - <detector>



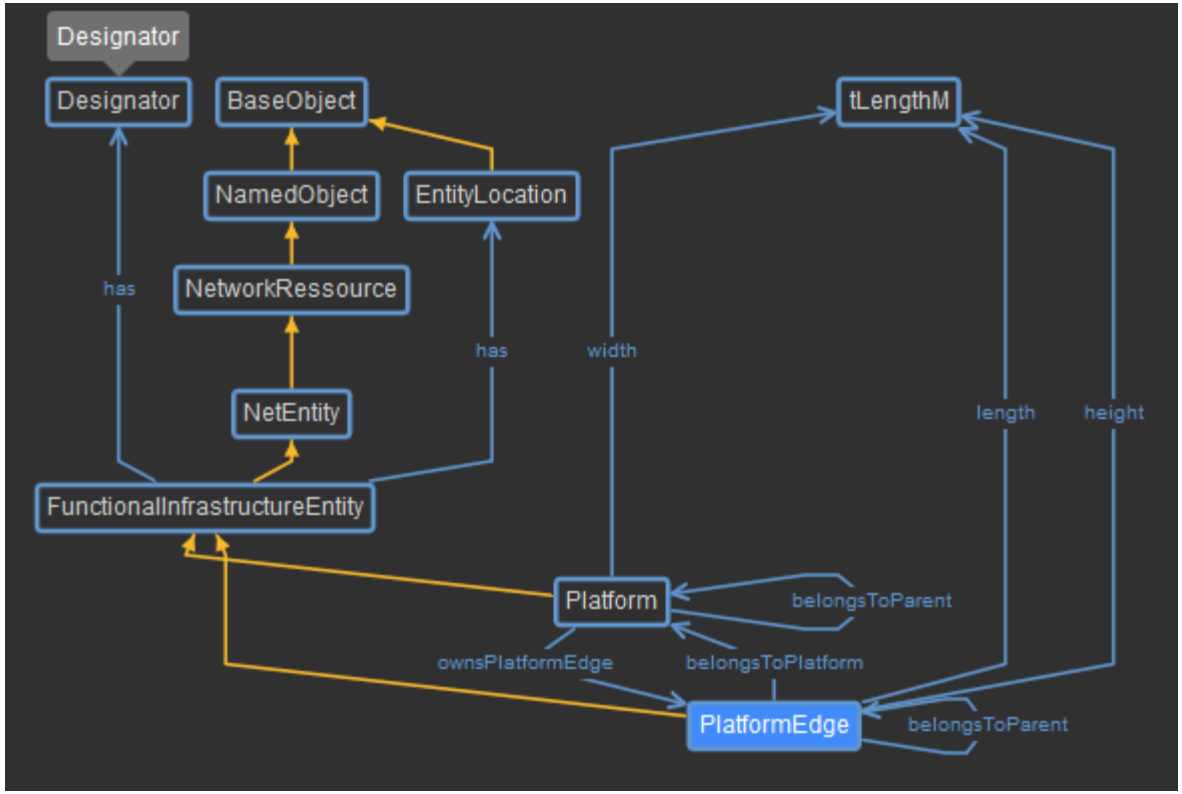
#438: Introduce <platformEdge>

- 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://trac.railml.org/ticket/438
Wiki:

#438: Introduce <platformEdge>

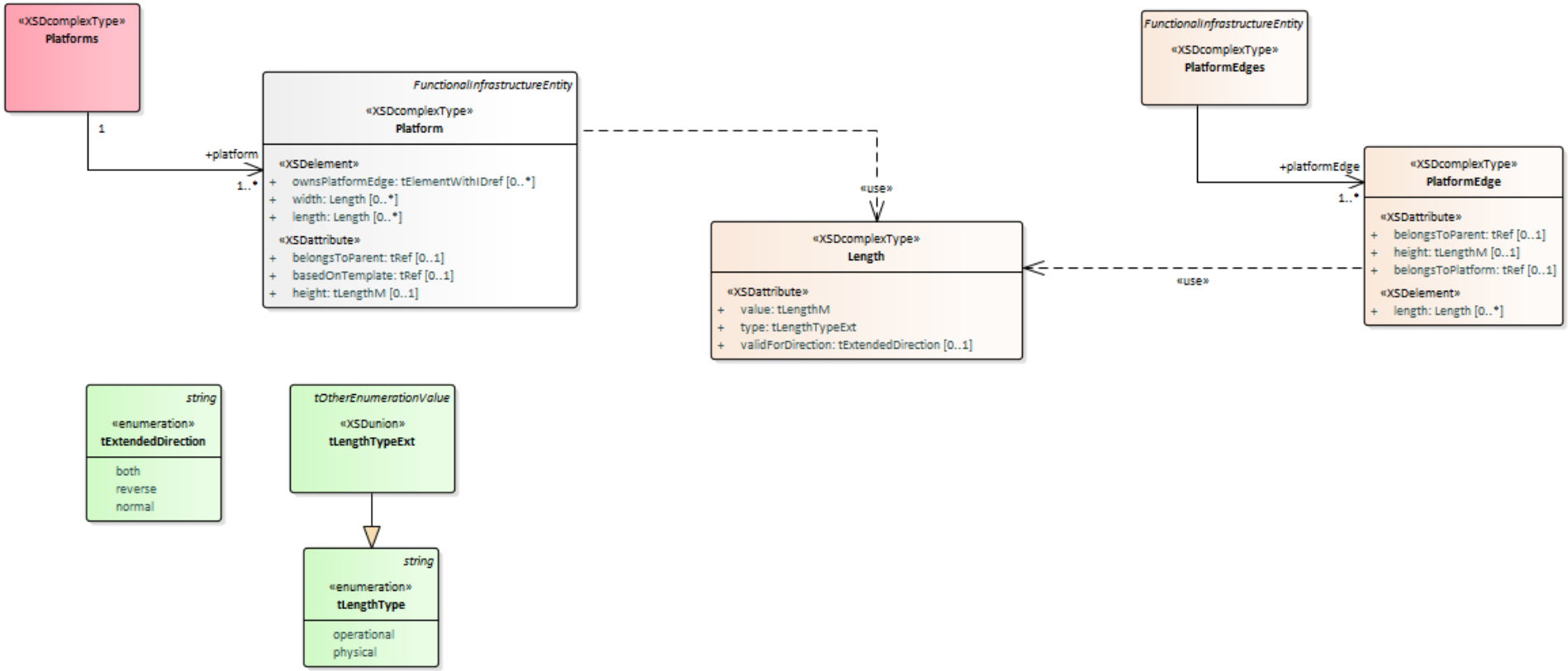
- Solution:



Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=650&goto=2133&#msg_2133
Trac: https://trac.railml.org/ticket/438
Wiki:

#438: Introduce <platformEdge>

- The model:



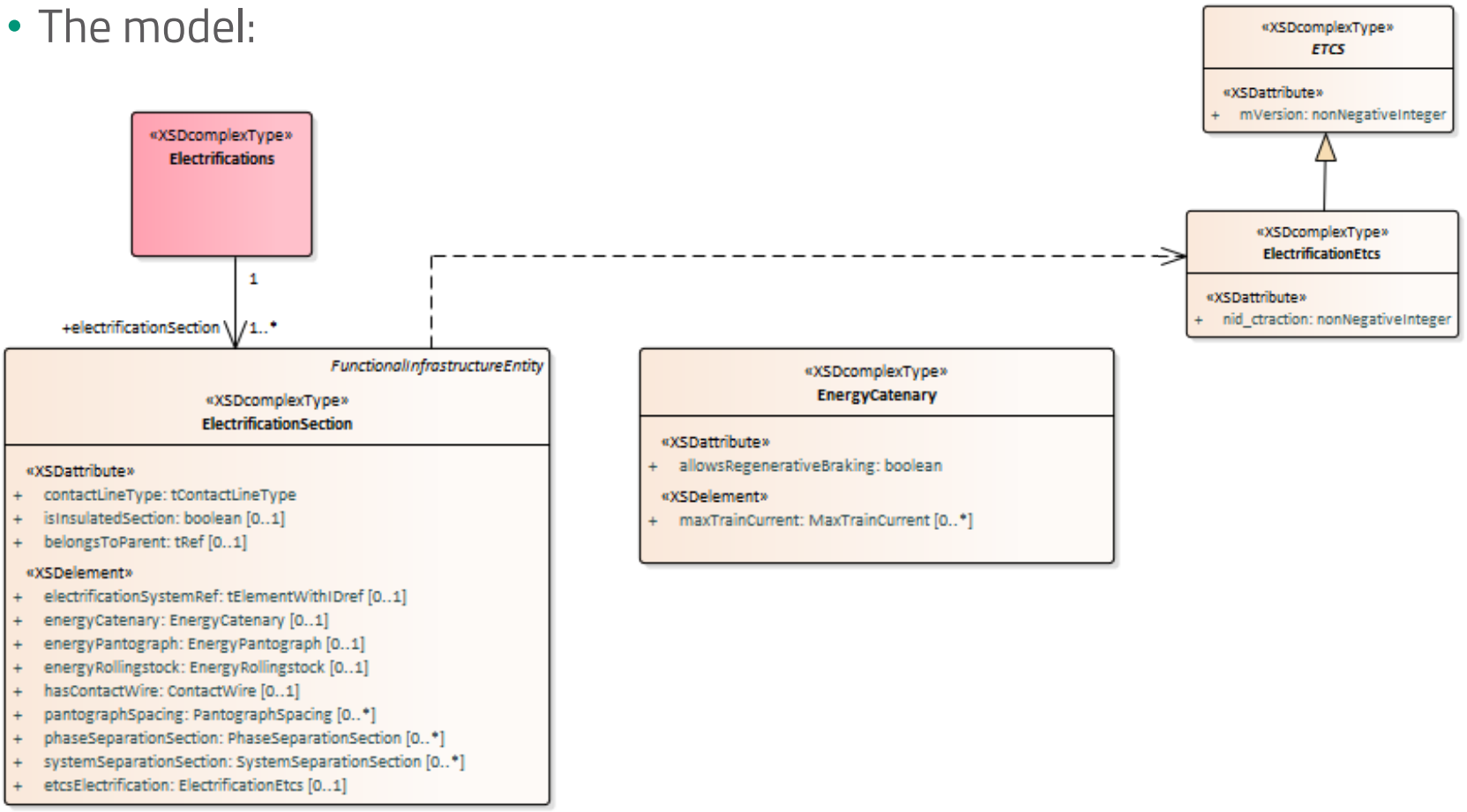
#439: NID_CTRACTION for electrification model

- 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://trac.railml.org/ticket/439
Wiki:

#439: NID_CTRACTION for electrification model

- The model:



#441: Extension of <opEquipment>

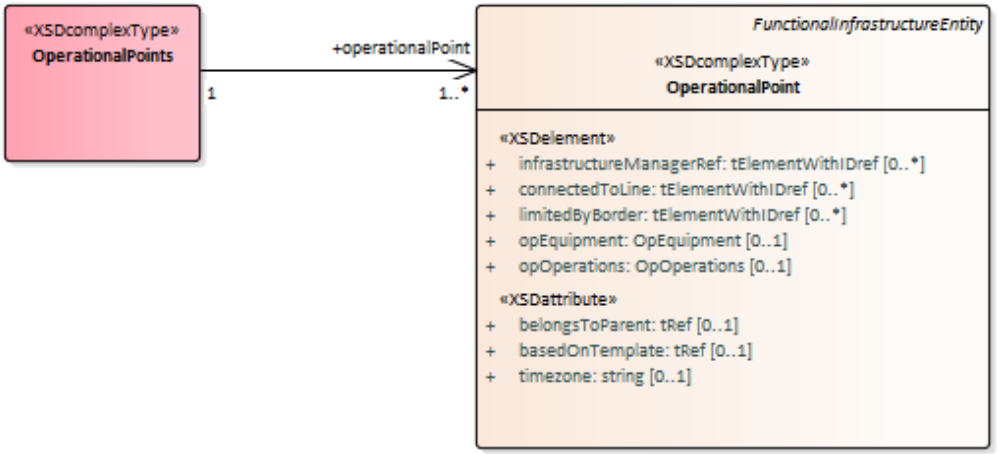
- The situation:
  - OperationalPoint can reference platforms, tracks, signals, stoppingPlaces and serviceSections, but not switches and derailers
- Idea:
  - Option 1: extend <opEquipment> with **<ownsStoppingPlaces>**, **<ownsSwitch>** and **<ownsDerailer>**
  - Option 2: introduce generic child element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

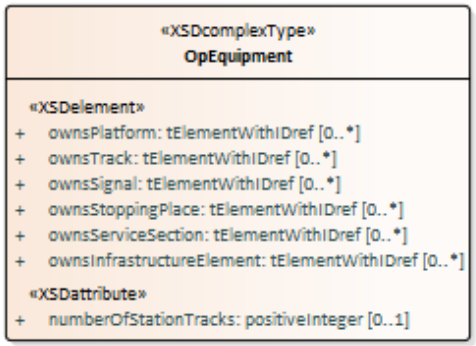
Which solution do you prefer?

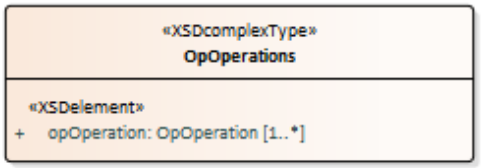
Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=667&start=0&
Trac: https://trac.railml.org/ticket/441
Wiki:

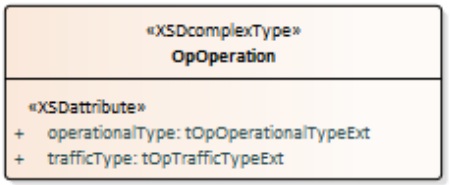
#441: Extension of <opEquipment>

- The model:
  - Option 2









#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)

Opinions from the community?

Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=714&goto=2382&#msg_2382
Trac: https://trac.railml.org/ticket/442
Wiki:

#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: use available child element **<designator>** to specify a rule code

Which solution do you prefer?

Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=712&start=0&
Trac: https://trac.railml.org/ticket/443
Wiki: https://wiki2.railml.org/index.php?title=IS:signal

#452: Driving directions in macroscopic nodes

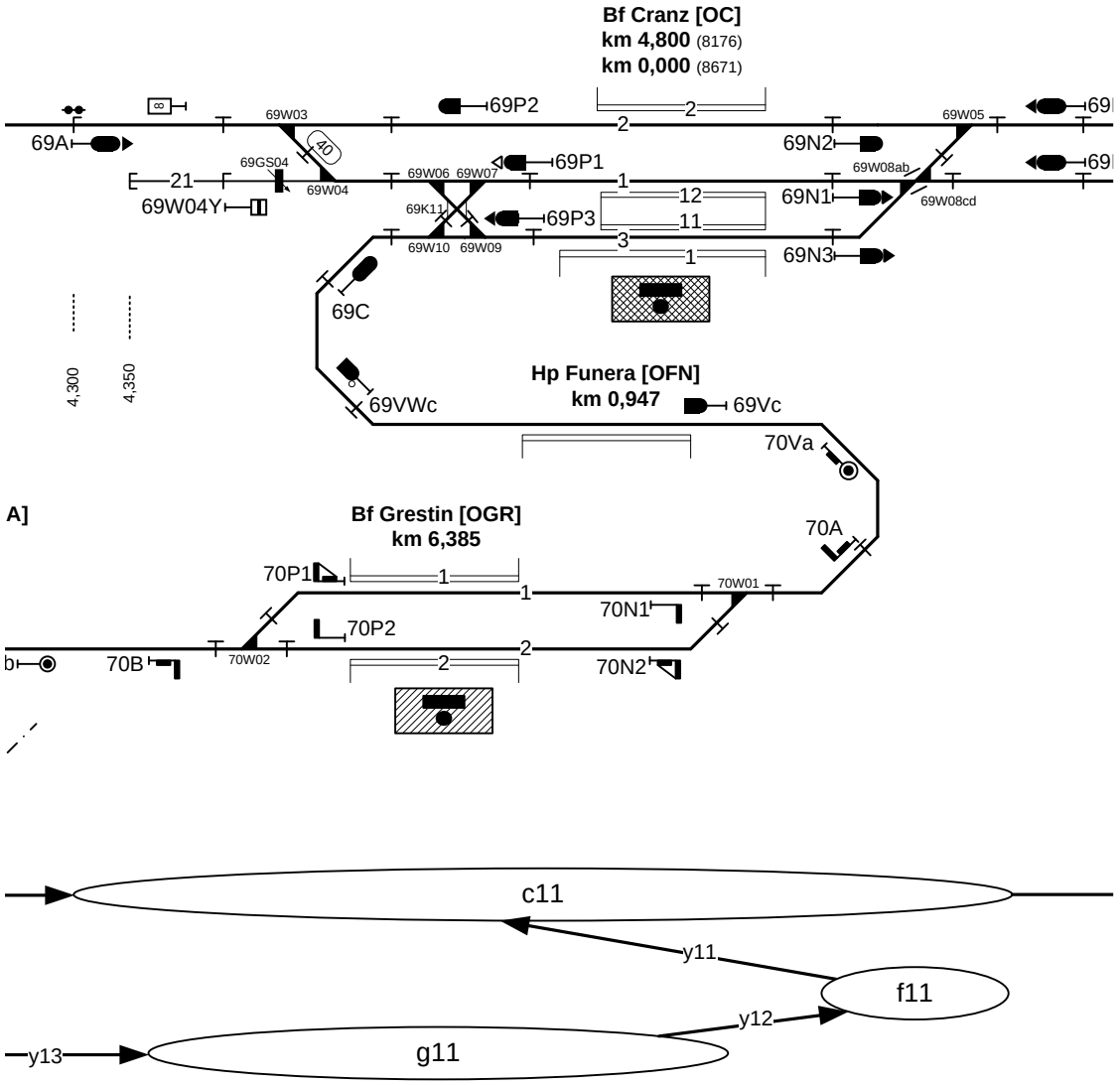
- The situation:
  - Without a microscopic model of the connections within a macroscopic node, it is not possible to identify possible direction changes of the railway vehicle
  - Question: how to model connections in the macroscopic node without microscopic modelling?

Solution to be found...
ideas from the community?

Links
Forum:
Trac: https://trac.railml.org/ticket/452
Wiki:

#452: Driving directions in macroscopic nodes

- Example:



#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 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://trac.railml.org/ticket/454
Wiki:

#454: Stopping places and platform edges

- The model:

Links
Forum: https://www.railml.org/forum/index.php?t=msg&goto=2644&#msg_2644
Trac: https://trac.railml.org/ticket/454
Wiki:

#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**

Links
Forum:
Trac: https://trac.railml.org/ticket/459
Wiki:

#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://trac.railml.org/ticket/460
Wiki:

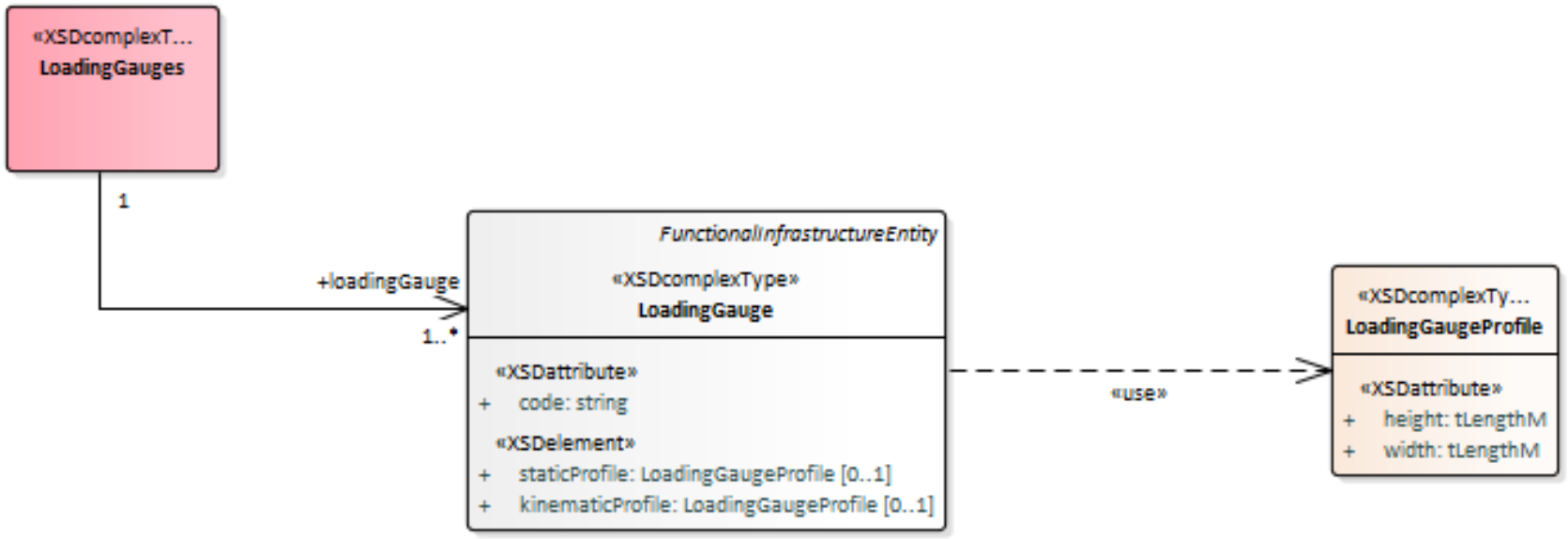
#461: Loading gauge profiles

- 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)

Links
Forum:
Trac: https://trac.railml.org/ticket/461
Wiki:

#461: Loading gauge profiles

- The model:



Links
Forum:
Trac: https://trac.railml.org/ticket/461
Wiki:

#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)

Implementation pending
Feedback from community?

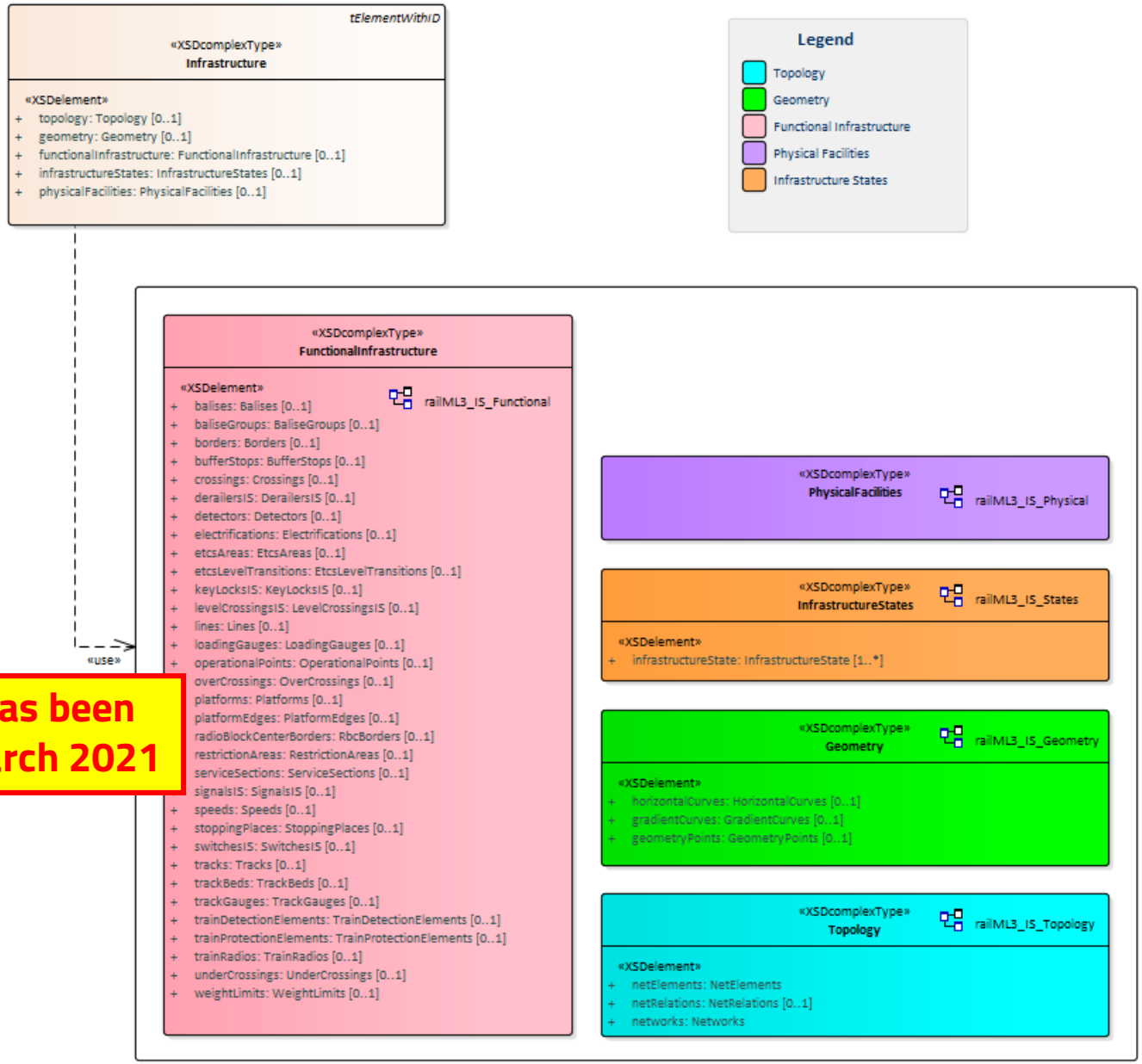
Links
Forum: https://www.railml.org/forum/index.php?t=msg&th=793&goto=2646&#msg_2646
Trac: https://trac.railml.org/ticket/466; https://trac.railml.org/ticket/450
Wiki:

Modeling

railML 3.2
Alpha March 19, 2021

Infrastructure

Name: railML3_IS
Author: railML.org
Version: 3.2
Created: 04.05.2016 15:26:29
Updated: 02.10.2020 13:48:07



railML 3.2 beta 1 has been published end of March 2021

railML.org

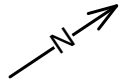
railML Advanced Example

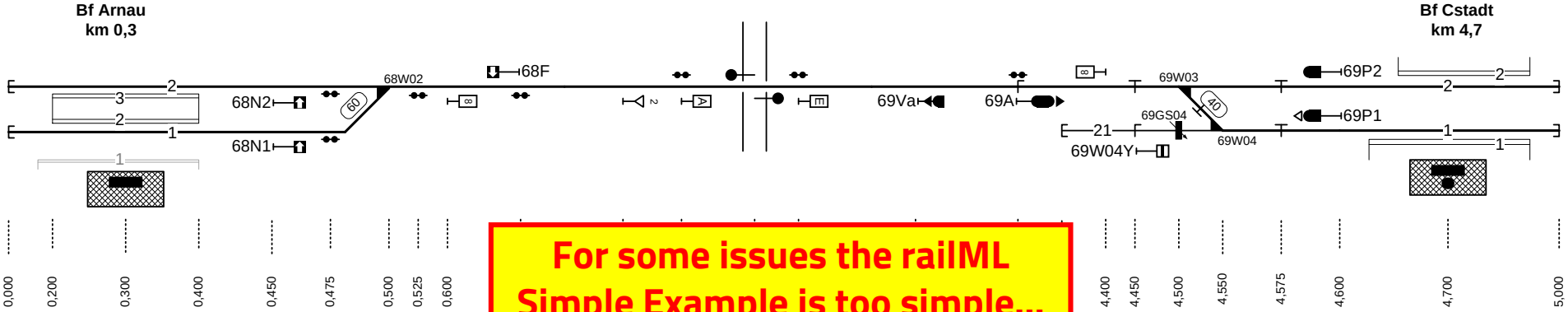
Increasing complexity in modelling with railML 3

railML Simple Example

railML®

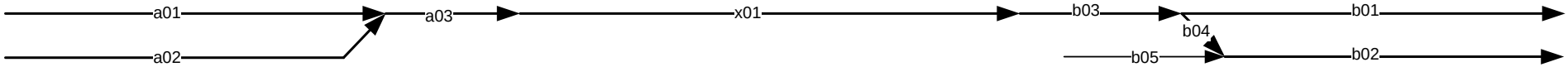
Simple Example
v11 with Topology (September 3, 2018)



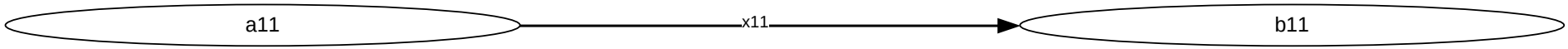


For some issues the railML Simple Example is too simple...

Micro Topology



Meso Topology



railML Advanced Example

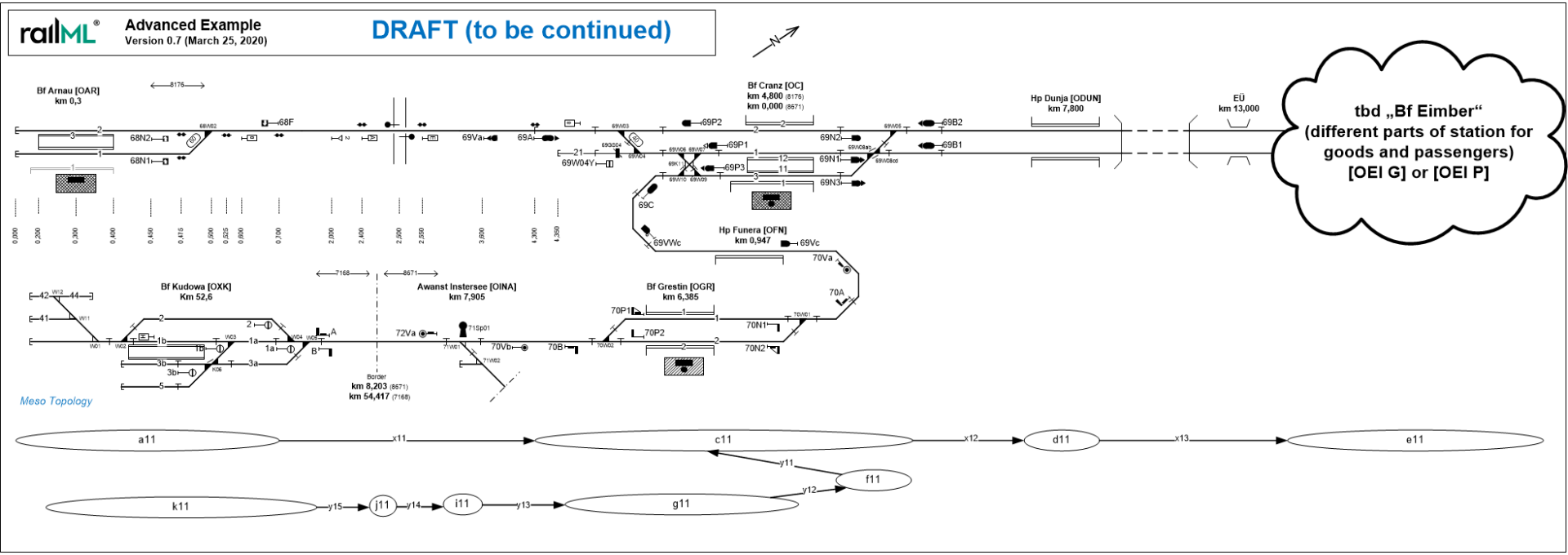
- Things to be modelled (in infrastructure):
  - Balises and balise groups
  - Mileage changes (gaps, overlaps)
  - Electrifications
  - Divided station tracks within a station
  - Two railway lines in form of an „Y“
  - A stopping point
  - Crossings, single switch crossing
  - Stabling tracks (siding tracks)
  - …

What else is interesting for you to be implemented?

Links
Forum:
Trac: https://trac.railml.org/ticket/351
Wiki:

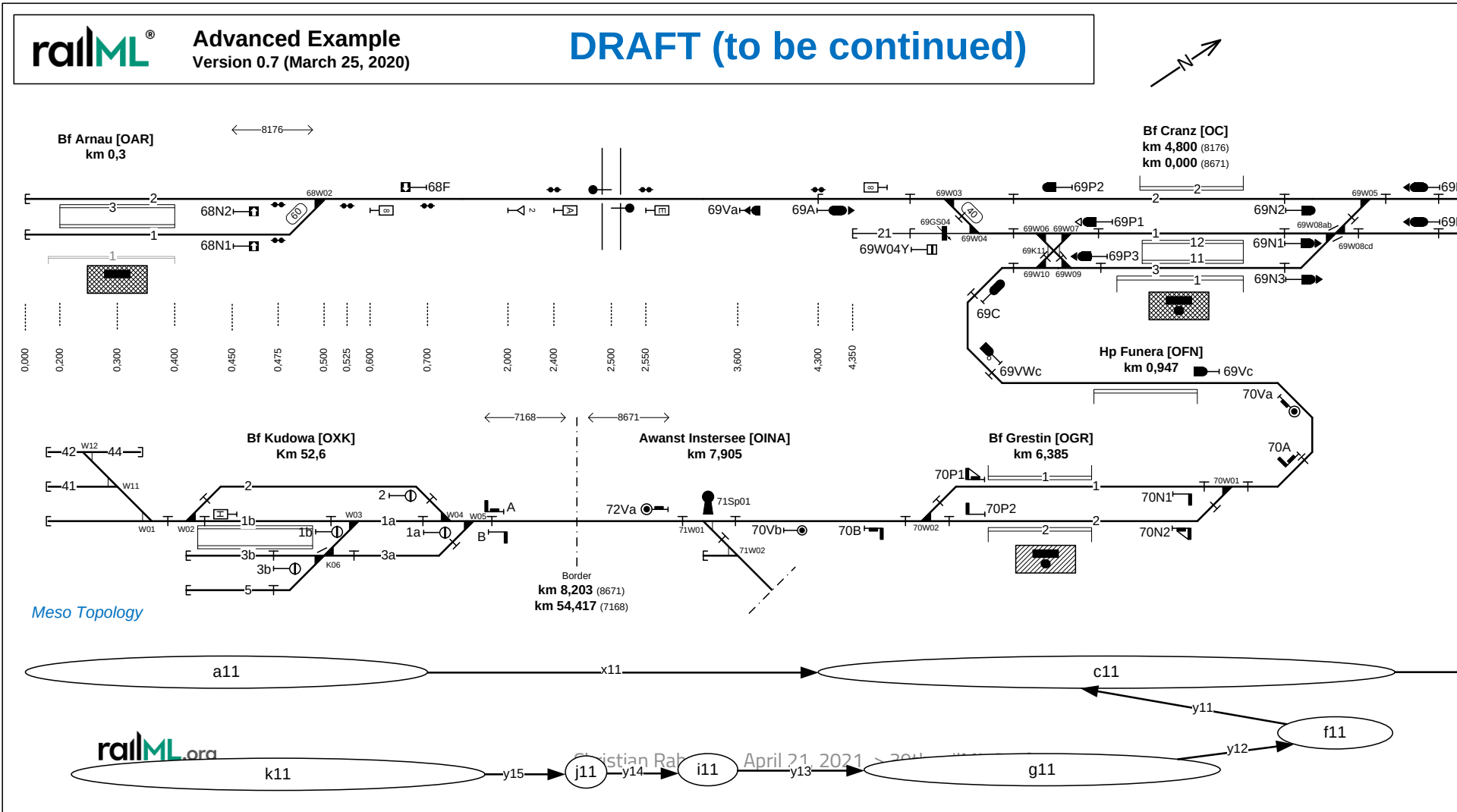
railML Advanced Example

- Overview:



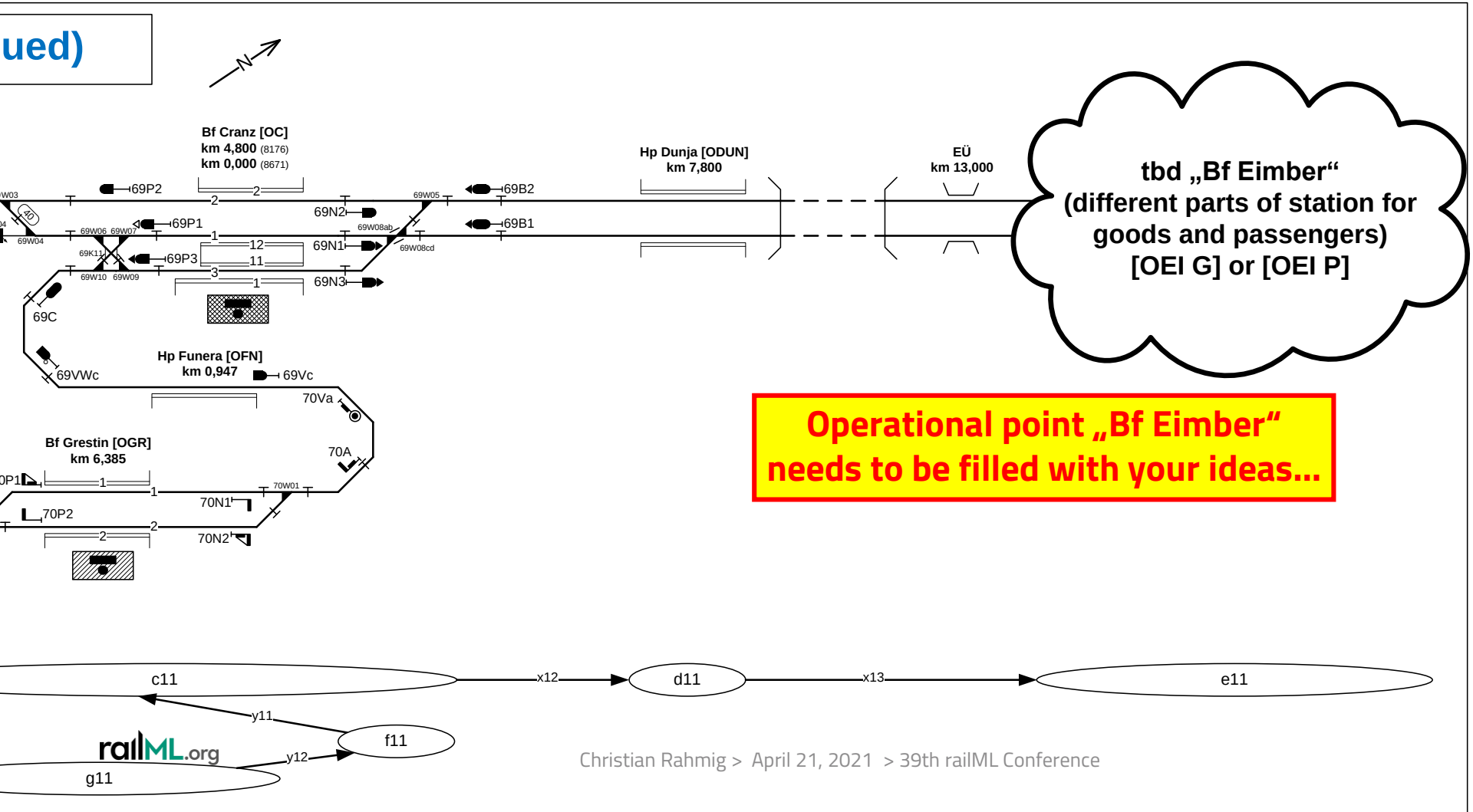
railML Advanced Example

- Detail left:



railML Advanced Example

- Detail right:



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

Institut für Verkehrssystemtechnik

Christian Rahmig
railML Infrastructure Coordinator

www.railml.org

coord@infrastructure.railml.org

+49 – (0)531 – 295 3461
+49 – 173 – 2714509