

railML® Use Case Definition

Interlocking Module Engineering Data

Abbreviation	IMED
Implementation	railML® 3.1 (release: 19.02.2019)
Date	17.02.2019
Version	1.0
Author	Jörg von Lingen

Revision History

Version	Date	Description	Author
0.1	10.11.2018	Initial Version	Jörg von Lingen
0.2	10.12.2018	Revision according schema update and comments	Jörg von Lingen
0.3	11.12.2018	Renaming "point" to "switch"	Jörg von Lingen
0.4	27.01.2019	Revision after schema refactoring	Jörg von Lingen
1.0	17.02.2019	Prepared for release	Jörg von Lingen

Executive Summary

The railML 3.1 provides an XML schema for detailed description of infrastructure and interlocking features. The interlocking part is based on the use cases "Routes for timetable simulation" (RSIM) and "Interlocking module engineering data" (IMED). Whereas the infrastructure part focuses on the physical objects and features the interlocking part is considering the functional or logical features of the objects. Thus the interlocking is using the infrastructure with references to their objects. The interlocking part allows the description of typical objects like:

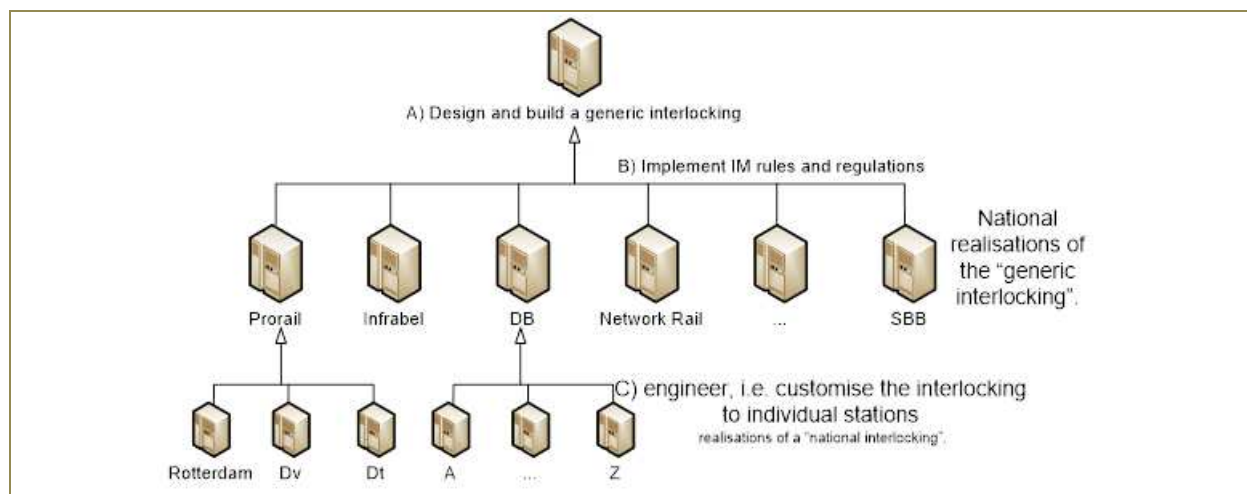
- TVD sections
- movable elements like switches, derailleurs and crossings
- signals with their aspects and aspect relations
- level crossings
- train routes with start area, running path and overlap
- route release
- special operational areas and element groups
- interfaces between different interlocking
- control relations from HMI to interlocking module and the element

Although some infrastructure manager dependent characteristics can be defined the interlocking sub-schema does not directly describe operational rules of that infrastructure manager.

1 Informal Use Case Description

The process of creating an interlocking is three-tiered.

1. The manufacturer designs an interlocking platform with an operating system and appropriate hardware to operate field elements such as switches, signals, track sections, axle counters, etc.
2. The manufacturer implements the Infrastructure Manager's (IM) rules and regulations ([Betriebsordnung](#)) in software. The behaviour of the interlocking is customised to meet the IM's requirements. This stage is a one-off process per IM and includes functions particular to railway administrations such as overlap, signal aspects and flank protection. Once these national rules have been implemented in generic software, they remain fixed. The functional behaviour of an interlocking now reflects the IM's rulebook for interlocking.
3. Customise, i.e. engineer, the interlocking to match individual stations. The interlocking is custom-tailored to operate switches, sections, signals and so on. This engineering process ([Projektierung](#), [définition des invariants](#)) of course depends on the layout of the individual station. Every interlocking is different.



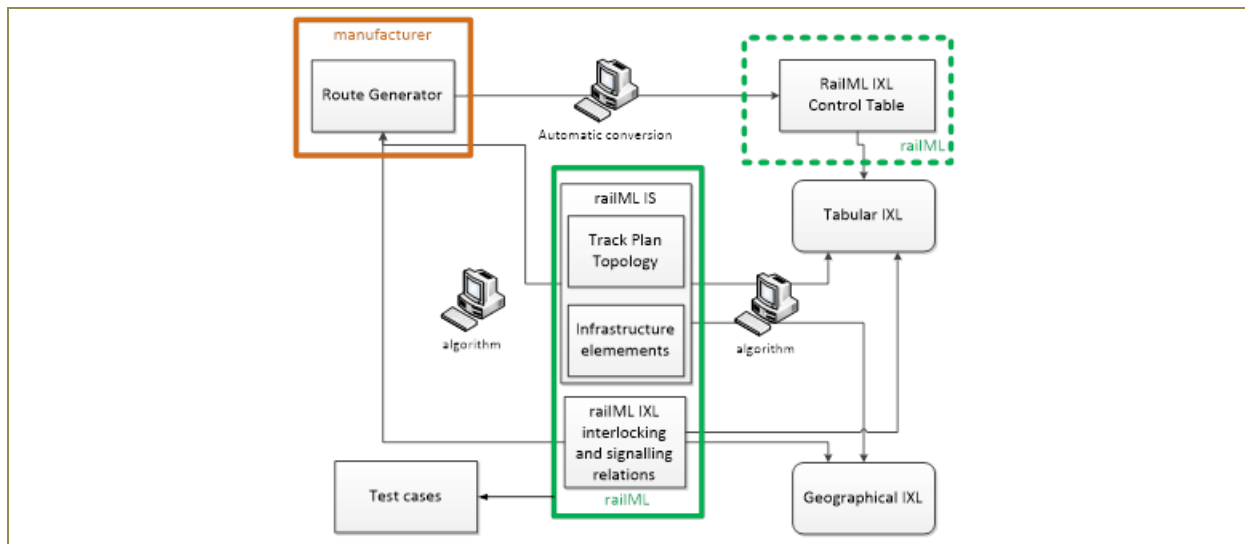
Stage C, the engineering process, is extremely costly. The overwhelming majority of data are taken from (paper) plans. Data exchange between IM and supplier takes the shape of graphical plans, tables or plain language requirement documents. These plans often are a print-out from suppliers proprietary tools. The manual conversion of the data from paper into the supplier's tool chain is likely to introduce errors that must be found and corrected by testing.

Automated exchange and conversion of railML data, rather than picking data from paper plans, into the suppliers tool chain will reduce cost. Better still, if the input data are known to be correct, validation costs will fall drastically. The latter should be the principal reason to adopt a common exchange format.

This use case concerns the exchange of engineering data for tailoring an interlocking to fit a station.

This use case focuses on "legacy" interlockings and signalling systems. The latter indicates that data can be used for the purpose of configuring ETCS systems. Note that the Eulynx project, with whom railML cooperates, defines data in terms of types and attributes mainly for the purpose of

asset Management. These Eulynx-defined asset data are internal to the IM whereas railML focuses on data exchange. Infrastructure Managers keep planning data needed to build an interlocking that they wish to convey to suppliers of interlocking equipment in a standard format. Note that data are safety relevant and must respect the restrictions of signalling rules of the railways. To-build data flow from IM to suppliers. The latter enrich the data with specific design information that flows back to the Infrastructure Manager (as-built). To-build and as-built data will be in railML format.



Above figure shows the position of railML in the data exchange process.

1.1 Data Flows and Interfaces

The railML IL scheme will hold data that manufacturers' tools can easily extract.

Information Category:

- Topology (origin: railML IS schema or classic track plan)
- Track Elements (signals, switches, sections...) (origin: railML IS schema or classic track plan)
- Interlocking relations (origin: railML IS schema or classic track plan or plain text requirements.)
- Control tables (origin: Track plan or spreadsheets)
- Signal aspect relations (origin: Classic signal plan)
- Signalled speeds (origin: Classic signal plan)
- Timers (e.g. for throwing switches, revoking signals, etc) (origin: Railway regulations or other plain text requirements)

The focus should be on data reliability because IL and IS data will be used for producing SIL-4 interlocking.

1.2 Data Characterization

This section serves to specify the required data regarding certain aspects.

How often do the data change (update)?

Depends on the frequency of resignalling, public works, corrections. Order of magnitude for a single station is 1/month – 1/decade. A typical modification is removal of a switch. This suggests that the frequency of update is fairly low.

How big are the data fragments to be exchanged (complexity)?

The size of the data to be exchanged typically match the size of an interlocking area, i.e. a station. The size of the railML IL file grows linearly with the number of routes and exponentially with the number of network nodes.

Which views are represented by the data (focus)?

- Signalling
- Schematic view of topology (Indispensable for interlocking engineers used to graphic presentation of signalling plans)

Interlocking engineers need the schematic view of data, i.e. a graphical representation of IL data because they are used to signal plans and track plans. Therefore, the railML data must contain coordinates that allow plotting schematic views. Such views support validation. Signalling engineers will only accept railML if combined with easy-to-use graphical tools. These tools must show obvious advantages with respect to the conventional way of working with paper plans and spreadsheets.

Which specific data do you expect to receive/send (elements)?

- Infrastructure:
 - Topology: Lines + operational points
 - Signalling: signals, train detection
 - Rails: track sections, switches, crossings, derailer, level crossings
 - Operational Points: name, type
- Interlocking:
 - IM specific Types: signal aspects, route types, restricted areas
 - Track Assets: TVD sections, movable elements, signals, key locks, interfaces
 - Logical Assets: routes, overlaps, partial routes, route relations, conflicting routes
 - Signal Plan: aspect relations
 - Control Relations: interlocking, extent of element control, element groups

2 Schema Implementation

2.1 Sub-schema References

The implementation of the IMED use case is based on elements of the following sub-schemas:

- Infrastructure
- Interlocking

2.2 Element Classification

This section serves to list all the railML® elements that are required for the implementation of the IMED use case. For all the related sub-schemas the listing is done using tables in the following structure:

{Topic}		
{mandatory element/@attribute}	m	{Description}
{optional element/@attribute}	o	{Description}

Mandatory elements have to be provided by an export interface and have to be understood by an import interface. However, this does not mean mandatory elements or attributes have to occur at each possible position. Such elements or attributes shall occur at least once in the import or export, if not set to mandatory in the schema. Optional elements are recommended for export and import interfaces.

2.2.1 Common

There are no railML® common elements required for the IMED use case.

2.2.2 Infrastructure

The infrastructure used for the IMED use case is comparable in its extent with the SCTP use case [2]. It shall contain all the related infrastructure elements as listed below with their physical characteristics. This means the topology of the network and element positions are crucial. However, the details of the infrastructure are not included here.

The following tables contain the railML® infrastructure elements required for the IMED use case:

Infrastructure		
topology	m	The topology of the described railway network has to be given.
positioning	m	
geometry	x	
functionalInfrastructure	m	The functional infrastructure objects with their features have to be given.

2.2.2.1 Topology

Topology		
netElement	m	
netRelation	m	
network	m	The <network> shall contain at least one child element <level> for describing the microscopic railway network on line section level and line level.

2.2.2.2 Positioning

Positioning		
geometricPositioningSystem	o	
linearPositioningSystem	m	There shall be at least one <linearPositioningSystem> for defining the mileage along the railway line.
screenPositioningSystem	x	

GeometricPositioningSystem		
@crsDefinition	o	The geometric positioning system shall be identified by its EPSG code to be found in this attribute.

LinearPositioningSystem		
anchor	m	All mileage changes and other mileage anomalies shall be modelled using <anchor> elements.
@units	m	Mileage shall be given in meters.
@startMeasure, @endMeasure	m	Mileage shall be given in meters with one digit after the comma.

2.2.2.3 Functional Infrastructure

Functional Infrastructure		
balise	o	
border	m	Borders between infrastructure manager responsibilities shall be modelled as well as national borders
bufferStop	m	
crossing	m	
derailer	m	
electrification	x	
embankment	x	
levelCrossing	m	It is recommended to provide elementary information about existence of level crossings along the line
line	o	
loadingGauge	x	
lock	m	
operationalPoint	m	
operationFacility	x	
overCrossing	x	
platform	x	
platformEdge	x	
restrictionArea	x	

serviceSection	x	
signal	m	
speed	m	
stoppingPlace	o	
switch	m	
track	m	
trackBed	x	
trackGauge	x	
trainDetectionElement	m	
trainProtectionElement	o	
trainRadio	x	
undercrossing	x	
weightLimit	x	

2.2.3 Interlocking

The following tables contain the railML® interlocking elements required for the IMED use case:

Interlocking		
specificIM	m	The specific types of the infrastructure manager of the described railway network have to be given. There may be more than one entry of specificIM.
assetsForIL	m	The list of assets used for interlocking purpose is essential for this use case. There is only one list of assets.
signalBox	m	The main characteristics of the particular interlocking module with all related assets have to be given. There may be more than one entry of signalBox.
controller	o	The controller definition is not essential for interlocking engineering data, i.e. an interlocking is working without a particular controller. There may be more than one entry of controller.

2.2.3.1 specificIM

specificIM		
ownsSetsOfAssets @ref	m	This reference to the list of assets is required for consistency check.
usesTypes	m	This parent element for the list of IM specific types is required.
designator @entry, @register	m	The name of the selected Infrastructure Manager is required for readability of the data set.
@id	m	The ID of the Infrastructure Manager will be used for references to it.

specificIM usesTypes		
has Aspect designator @entry, @register	m	Each defined signal aspect shall have a name (from the IM).
has Aspect @id	m	Each defined signal aspect shall have an ID to refer to it.

hasAspect @GenericAspect	m	The attribute provides a basic classification of the defined signal aspect. It uses an enumeration without the possibility to add "other"-entries.
hasTVDresetStrategy designator @entry, @register	m	Each defined TVD reset strategy shall have a name (from the IM).
hasTVDresetStrategy @id	m	Each defined TVD reset strategy shall have an ID to refer to it.
hasTVDresetStrategy @ResetStrategy	m	The attribute provides a basic classification of the defined reset strategy. It uses an enumeration without the possibility to add "other"-entries.
hasRouteType designator @entry, @register	m	Each defined route type shall have a name (from the IM).
hasRouteType @id	m	Each defined route type shall have an ID to refer to it.
hasRouteType @GenericRouteType	m	The attribute provides a basic classification of the defined route type. It uses an enumeration with the possibility to add "other"-entries.
hasLevelCrossingType designator @entry, @register	m	Each defined level crossing type shall have a name (from the IM).
hasLevelCrossingType @id	m	Each defined level crossing type shall have an ID to refer to it.
hasLevelCrossingType @isControlType	m	The attribute provides a basic classification of the defined level crossing type. It uses an enumeration without the possibility to add "other"-entries.
hasLevelCrossingType @allowsLocalOperation	o	The Boolean value might be omitted, if the information is unknown.
hasLevelCrossingType @hasBarrier	o	The Boolean value might be omitted, if the information is unknown.
hasLevelCrossingType @hasTrafficWarning	o	The Boolean value might be omitted, if the information is unknown.
hasElementGroupType designator @entry, @register	m	Each defined element group shall have a name (from the IM).
hasElementGroupType @id	m	Each defined element group shall have an ID to refer to it.
hasElementGroupType @elementGroupType	m	The attribute provides a basic classification of the defined element group type. It uses an enumeration with the possibility to add "other"-entries.
hasDetectorType designator @entry, @register	o	Each defined detector shall have a name (from the IM). This feature may not be needed in every network but if the element used the attributes are needed.
hasDetectorType @id	o	Each defined detector shall have an ID to refer to it. This feature may not be needed in every network but if the element used the attributes are needed.
hasDetectorType @detectorType	o	The attribute provides a basic classification of the defined detector type. It uses an enumeration with the possibility to add "other"-entries. This feature may not be needed in every network but if the element used the attributes are needed.

2.2.3.2 assetsForIL

assetsForIL		
tvSection designator @entry, @register	m	The list of TVD section is required as they shall be known to the interlocking. Each defined TVD section shall have a name (from the IM).
tvSection @id	m	Each defined TVD section shall have an ID to refer to it.
switchIL designator @entry, @register	m	The list of switches is required as they shall be known to the interlocking. Each defined switch shall have a name (from the IM).
switchIL @id	m	Each defined switch shall have an ID to refer to it.
derailerIL designator @entry, @register	m	The list of derailleurs is required as they shall be known to the interlocking. Each defined derailer shall have a name (from the IM).
derailerIL @id	m	Each defined derailer shall have an ID to refer to it.
movableCrossing designator @entry, @register	m	Each defined movable crossing shall have a name (from the IM).
movableCrossing @id	m	Each defined movable crossing shall have an ID to refer to it.
levelCrossingIL designator @entry, @register	m	The list of level crossings is required as they shall be known to the interlocking. Each defined level crossing shall have a name (from the IM).
levelCrossingIL @id	m	Each defined level crossing shall have an ID to refer to it.
key designator @entry, @register	o	The list of keys is optional as they are normally not known to the interlocking. Each defined key shall have a name (from the IM).
key @id	o	Each defined key shall have an ID to refer to it.
keyLockIL designator @entry, @register	m	The list of key locks or detector is required as they shall be known to the interlocking. Each defined key lock shall have a name (from the IM).
keyLockIL @id	m	Each defined key lock shall have an ID to refer to it.
genericDetector designator @entry, @register	o	The list of detectors is required as they shall be known to the interlocking. Each defined detector shall have a name (from the IM).
genericDetector @id	o	Each defined detector shall have an ID to refer to it.
signalIL designator @entry, @register	m	The list of signals is required as they shall be known to the interlocking. This applies only to switchable or virtual signals. Each defined signal shall have a name (from the IM).
signalIL @id	m	Each defined signal shall have an ID to refer to it.
<i>atpDevice designator @entry, @register</i>	x	not used in railML v3.1
<i>atpDevice @id</i>	x	not used in railML v3.1
interface designator @entry, @register	m	The list of interfaces is required as they shall be known to the interlocking. Each defined interface shall have a name (from the IM).
interface @id	m	Each defined interface shall have an ID to refer to it.
workZone designator @entry, @register	o	The list of work zones is optional as they may not be used in each project. Each defined work zone shall have a name (from the IM).
workZone @id	o	Each defined work zone shall have an ID to refer to it.

localOperationArea designator @entry, @register	m	The list of local operation areas is required as they shall be known to the interlocking. Each defined local operation area shall have a name (from the IM).
localOperationArea @id	m	Each defined local operation area shall have an ID to refer to it.
shuntingZone designator @entry, @register	o	The list of shunting zones is optional as they may not be used in each project. Each defined shunting zone shall have a name (from the IM).
shuntingZone @id	o	Each defined shunting zone shall have an ID to refer to it.
permissionZone designator @entry, @register	m	The list of permission zones is required as they shall be known to the interlocking. Each defined permission zone shall have a name (from the IM).
permissionZone @id	m	Each defined permission zone shall have an ID to refer to it.
routeReleaseGroupAhead designator @entry, @register	o	The list of route release groups ahead of the train is optional as they shall be known to the interlocking, if it supports the release of several elements in a group from set routes. Each defined route release group ahead of the train shall have a name (from the IM).
routeReleaseGroupAhead @id	o	Each defined route release group ahead of the train shall have an ID to refer to it.
routeReleaseGroupRear designator @entry, @register	o	The list of route release groups in rear of the train is optional as they shall be known to the interlocking, if it supports the release of several elements in a group from set routes. Each defined route release group in rear of the train shall have a name (from the IM).
routeReleaseGroupRear @id	o	Each defined route release group in rear of the train shall have an ID to refer to it.
route designator @entry, @register	m	The list of routes is required as they shall be known to the interlocking. Each defined route shall have a name (from the IM).
route @id	m	Each defined route shall have an ID to refer to it.
conflictingRoute designator @entry, @register	o	The list of conflicting routes is optional as they shall be known to the interlocking, if some routes need to be set incompatible explicitly. Each defined conflicting route shall have a name (from the IM).
conflictingRoute @id	o	Each defined conflicting route shall have an ID to refer to it.
routeRelation designator @entry, @register	o	The list of routes relation is optional as they shall be known to the interlocking, if there additional conditions to be configured for a defined route. Each defined route relation shall have a name (from the IM).
routeRelation @id	o	Each defined route relation shall have an ID to refer to it.

combinedRoute designator @entry, @register	o	The list of combined routes is optional as they shall be known to the interlocking, if it supports the setting of several routes by a single command. Each defined combined route shall have a name (from the IM).
combinedRoute @id	o	Each defined combined route shall have an ID to refer to it.
overlap designator @entry, @register	m	The list of overlaps is required as they shall be known to the interlocking. Each defined overlap shall have a name (from the IM).
overlap @id	m	Each defined overlap shall have an ID to refer to it.
dangerPoint designator @entry, @register	o	The list of danger points is optional as they shall be known to the interlocking, if it supports the use of it. Each defined danger point shall have a name (from the IM).
dangerPoint @id	o	Each defined danger point shall have an ID to refer to it.
destinationPoint designator @entry, @register	o	The list of destination points is optional as they shall be known to the interlocking, if it supports just overlaps or danger points related to a track asset without the definition of routes. Each defined destination point shall have a name (from the IM).
destinationPoint @id	o	Each defined destination point shall have an ID to refer to it.
powerSupplyL designator @entry, @register	m	The list of power supplies for interlocking is required as they shall be known to the interlocking. Each defined power supply shall have a name (from the IM).
powerSupplyL @id	m	Each defined power supply shall have an ID to refer to it.
designator @entry, @register	o	The name of the asset list supports readability of data set.
@id	m	The ID is required as it is referred to it from the IM specific data definition.

assetsForIL tvdSection		
hasDemarcatingBufferstop @ref	o	The reference to a buffer stop in the infrastructure part is optional as the railway network might not have any.
hasExitSignal @ref	o	The reference to a signal is optional as it might not affect the interlocking data.
hasDemarcatingTrainedetector @ref	m	The reference to train detectors in the infrastructure part is required as there are no TVD sections without them.
hasResetStrategy @ref	m	The reference gives the interlocking some information how to handle the TVD section in case of failures.
@isBerthingTrack	m	The Boolean value is used to indicate any special station tracks where trains usually start, end, split, join or reverse.
@residualRouteCancellationDelay	m	The Boolean value is mandatory in the schema.
@partialRouteReleaseDelay	m	The Boolean value is mandatory in the schema.
@technology	o	The type of detection technology may be given. It uses an enumeration with the possibility to add "other"-entries.
@frequency	o	The set frequency in Hertz of the track circuit may be given. It shall not be used with axle counters.
designator @entry, @register	m	The TVD section shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL switchIL		
refersTo @ref	m	The reference to an element in the infrastructure part is mandatory in the schema.
hasGaugeClearanceMarker @ref	x	
hasTvdSection @ref	m	The reference to a TVD section allows the relation of occupational status of the switch.
connectedToPowerSupply @ref	m	The reference to the related power supply is used to control the number of switching switch actuators.
relatedMovableElement @ref	o	The reference is only used if there is a related movable element. It shall be used in dependence of the value of switchIS @type.
hasFoulingTrainDetectors @ref	m	The references to train detection elements in infrastructure part mark the branches of the switch, which may receive lack of clearance.
branchLeft @ref	m	The reference to the underlying track in infrastructure part allows determining the neighbour relation. It is mandatory in schema.
branchRight @ref	m	The reference to the underlying track in infrastructure part allows determining the neighbour relation. It is mandatory in schema.
hasPositionRestriction designator @entry, @register	x	
hasPositionRestriction @id		
hasPositionRestriction relatedSwitchInPosition	o	Dependent on the relation to another movable element there might be a restriction in positions.
hasPositionRestriction relatedDerailerInPosition	o	Dependent on the relation to another movable element there might be a restriction in positions.
hasPositionRestriction @restrictedPosition	o	Dependent on the relation to another movable element there might be a restriction in positions.
@maxThrowTime	m	The duration value is mandatory in the schema. It is needed to supervise the switching process.
@typicalThrowTime	o	The duration value is optional and normally not needed for the interlocking.
@returnsToPreferredPosition	m	The Boolean value is needed for handling after use in a route.
@isKeyLocked	m	The Boolean value is needed to determine whether the switch is normally clamped by a locking device. If true there shall be no switch actuators connected.
@preferredPosition	m	In case the movable element has a preferred position (when not in use) this position shall be given. It uses an enumeration without the possibility to add "other"-entries. If there is no preferred position the attribute shall be omitted.
@numberOfBladeSwitchActuators	m	The non-negative integer value shall be given for switches. It shall be "0" in case of no switch actuator.
@numberOfFrogSwitchActuators	m	The non-negative integer value shall be given for switches. It shall be "0" in case of no switch actuator.
designator @entry, @register	m	The switch shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL switchIL relatedSwitchInPosition		
refersToSwitch @ref	m	The reference to an element in the asset list is mandatory in the schema.
@inPosition	m	The position of the related movable element
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to.

assetsForIL switchIL relatedDerailerInPosition		
refersToDerailer @ref	m	The reference to an element in the asset list is mandatory in the schema.
@inPosition	m	The position of the related movable element
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to.

assetsForIL derailerIL		
refersTo @ref	m	The reference to an element in the infrastructure part is mandatory in the schema.
hasGaugeClearanceMarker @ref	x	
hasTvdSection @ref	m	The reference to a TVD section allows the relation to the occupational status of the derailer.
connectedToPowerSupply @ref	m	The reference to the related power supply is used to control the number of switching switch actuators.
relatedMovableElement @ref	o	The reference is only used if there is a related movable element (switch).
@maxThrowTime	m	The duration value is mandatory in the schema. It is needed to supervise the switching process.
@typicalThrowTime	o	The duration value is optional and normally not needed for the interlocking.
@returnsToPreferredPosition	o	The Boolean value is optional for handling after use in a route.
@isKeyLocked	o	The Boolean value is optional to determine whether the derailer is normally clamped by a locking device. If true there shall be no switch actuators connected.
@preferredPosition	o	In case the movable element has a preferred position (when not in use) this position shall be given. It uses an enumeration without the possibility to add "other"-entries. If there is no preferred position the attribute shall be omitted.
@numberOfBladeSwitchActuators	m	The non-negative integer value shall be given for derailers. It shall be "0" in case of no switch actuator.
@numberOfFrogSwitchActuators	x	
designator @entry, @register	m	The derailer shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL movableCrossing		
refersTo @ref	m	The reference to an element in the infrastructure part is mandatory in the schema.

hasGaugeClearanceMarker @ref	x	
hasTvdSection @ref	m	The reference to a TVD section allows the relation of occupational status of the crossing.
connectedToPowerSupply @ref	m	The reference to the related power supply is used to control the number of switching switch actuators.
relatedMovableElement @ref	x	
branchUpLeft @ref	m	The reference to the underlying track in infrastructure part allows determining the neighbour relation. It is mandatory in schema.
branchUpRight @ref	m	The reference to the underlying track in infrastructure part allows determining the neighbour relation. It is mandatory in schema.
branchDownLeft @ref	m	The reference to the underlying track in infrastructure part allows determining the neighbour relation. It is mandatory in schema.
branchDownRight @ref	m	The reference to the underlying track in infrastructure part allows determining the neighbour relation. It is mandatory in schema.
hasFoulingTrainDetectors @ref	m	The references to train detection elements in infrastructure part mark the branches of the switch, which may receive lack of clearance.
@maxThrowTime	m	The duration value is mandatory in the schema. It is needed to supervise the switching process.
@typicalThrowTime	o	The duration value is optional and normally not needed for the interlocking.
@returnsToPreferredPosition	x	
@isKeyLocked	x	
@preferredPosition	o	In case the movable element has a preferred position (when not in use) this position shall be given. It uses an enumeration without the possibility to add "other"-entries. If there is no preferred position the attribute shall be omitted.
@numberOfBladeSwitchActuators	x	
@numberOfFrogSwitchActuators	o	The non-negative integer value may be given for crossings, if there is any switch actuator connected.
designator @entry, @register	m	The crossing shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.
assetsForIL levelCrossingIL		
isLevelCrossingType @ref	m	The reference to the basic level crossing type from IM specific data definition is required.
refersTo @ref	m	The reference to an element in the infrastructure part is mandatory in the schema.
hasInterface @ref	m	The reference to a physical interface is required for handling of inputs and outputs.
deactivatedBy	m	The definition of deactivation is required as it is mandatory in the schema.
activationCondition	m	The definition of activation is required as it is mandatory in the schema.

hasTvdSection @ref	m	The reference to a TVD section allows the relation of occupational status of the level crossing. It is mandatory in the schema.
@preferredPosition	o	In case the level crossing has a preferred position (when not in use) this position shall be given. It uses an enumeration without the possibility to add "other"-entries. If there is no preferred position the attribute shall be omitted.
@unprotectedSpeed	x	
@typicalTimeToClose	m	The duration value is required to control the switching process. It is mandatory in the schema.
@constantWarningTime	m	The duration value is required to set up the delay between activation and closing when road traffic is warned. It is mandatory in the schema.
@minimumOpenTime	m	The duration value is required to define the time span a reclosing command shall be rejected after opening. If unknown the attribute can be omitted.
@maximumClosedTime	m	The duration value is required to define the time after a closed level crossing is considered problematic. If unknown the attribute can be omitted.
@requiresStopBeforeUnprotected LevelCrossing	x	
designator @entry, @register	m	The level crossing shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL levelCrossingIL deactivatedBy		
tvdDetectorRef @ref	m	The reference to the train detection device or TVD section is required for the location.
@delay	m	The duration value is required to set up the delay between the trigger and the action. It is mandatory in the schema.
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

assetsForIL levelCrossingIL activationCondition		
delayBySwitchPosition designator @entry, @register	x	
delayBySwitchPosition @id	x	
delayBySwitchPosition @delay	o	If the element is used the value shall be given.
delayBySwitchPosition @inPosition	o	If the element is used the value shall be given. It uses an enumeration without the possibility to add "other"-entries.
delayBySwitchPosition refersToSwitch @ref	o	If the element is used the reference shall be given.
aspectRelatedDelay designator @entry, @register	x	
aspectRelatedDelay @id	x	
aspectRelatedDelay @delay	o	If the element is used the value shall be given.
aspectRelatedDelay refersToSignal @ref	o	If the element is used the reference shall be given.

aspectRelatedDelay showsAspect @ref	o	If the element is used the reference shall be given.
signalDelayTime designator @entry, @register	x	
signalDelayTime @id	x	
signalDelayTime @delay	o	If the element is used the value shall be given.
signalDelayTime delayedBySignal @ref	o	If the element is used the reference shall be given.
activatedBy designator @entry, @register	x	
activatedBy @id	x	
activatedBy @delay	m	The duration value is required to set up the delay between the trigger and the action. It is mandatory in the schema.
activatedBy refersTo @ref	m	The reference to the train detection device or TVD section is required for the location.
@andOr	o	The attribute shall be used if more than one activation condition is given here. It uses an enumeration without the possibility to add "other"-entries.
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

assetsForIL key		
@isPhysical	o	The Boolean value whether the key is of physical type may be given.
designator @entry, @register	o	The key may have a name (from the IM) to enhance readability.
@id	m	If the element is used the ID is required as it is used to refer to it.

assetsForIL keyLockIL		
takesControlOf @ref	m	The reference to the element is required that is controlled by this key lock.
hasInterface @ref	m	The reference to the physical interface of the key lock is required.
refersTo @ref	m	The reference to the key lock in the infrastructure part shall be given.
acceptsKey @ref	o	The reference to the matching key may be given but is without effect for this use case.
hasTvdSection @ref	o	The reference to a TVD section allows the relation of occupational status of the track at the key lock. It is needed for sidings on open line.
hasSlaveLock @ref	x	
@hasAutomaticKeyRelease	o	The Boolean value is used for controlling key release based on a trigger event.
@hasAutomaticKeyLock	o	The Boolean value is used for controlling locking of key on return.
@keyRequestTime	o	The duration value is used if there is a key request event asking for key release.

@keyAuthoriseTime	o	The duration value is used if the key release is time limited, i.e. automatic relock when not removed.
@description	o	A more detailed description for the locking device may be given.
designator @entry, @register	m	The key lock shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL genericDetector		
takesControlOf @ref	m	The reference to the element is required that is controlled by this detector. It might be omitted in case of just an informative detector type.
hasInterface @ref	m	The reference to the physical interface of the detector is required.
refersTo @ref	o	The reference to physical device in the infrastructure part may be given.
detectorType @ref	m	The reference to the type of detector is required.
@affectsRouteSignalling	m	The Boolean value of influence to routes may be given if such detectors exist.
@allowsSingleOverride	o	The Boolean value can indicate whether a related command shall be available.
@allowsPermanentOverride	o	The Boolean value can indicate whether a related command (override on/off) shall be available.
@hasTriggeredSelfTest	o	The Boolean value can indicate whether the interlocking shall trigger a self test for this detector.
@selfTestToleranceTime	o	The duration value can give the time the self test needs to run and any outputs during it are to be tolerated.
@selfTestInterval	o	The duration value can give the interval at which the interlocking shall trigger the self test. It has to be used in combination with @triggeredSelfTest.
@description	o	A more detailed description of the detector may be given.
designator @entry, @register	m	The master lock shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL signall		
refersTo @ref	m	The reference to an element in the infrastructure part is mandatory in the schema.
protectsBlockExit @ref	o	The reference to the track asset (train detection element) marking the end of the open line may be used.
@releaseSpeed	o	The speed value may be given but refers to the IM specific operational rules and is not directly used in an interlocking.
@malfunctionSpeed	o	The speed value may be given but refers to the IM specific operational rules and is not directly used in an interlocking.
@approachSpeed	o	The speed value may be given but refers to the IM specific operational rules and is not directly used in an interlocking.
@passingSpeed	o	The speed value may be given but refers to the IM specific operational rules and is not directly used in an interlocking.
@releaseDelay	m	The duration value is required to set up the delay for signal closure after the trigger event.

@function	m	The function of the signal is required. It uses an enumeration with the possibility to add "other"-entries.
@isVirtual	m	The Boolean value is required to mark whether the signal is connected to an interlocking for switching.
designator @entry, @register	m	The signal shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL workZone		
isLimitedBy @ref	m	The list of references to the track assets limiting the area shall be used for defining its extent.
activationLock @ref	o	The reference to a key lock may be used if the zone is secured by such a physical device.
switchInPosition designator @entry, @register	x	
switchInPosition @id	x	
switchInPosition givenPosition ...	o	The particular position of the switch may be given if defined so for the active work zone.
switchInPosition @protectingSide	o	The selection of protection use may be set in combination with the given position. It uses an enumeration without the possibility to add "other"-entries.
derailerInPosition designator @entry, @register	x	
derailerInPosition @id	x	
derailerInPosition givenPosition ...	o	The particular position of the derailer may be given if defined so for the active work zone.
derailerInPosition @protectingSide	o	The selection of protection use may be set in combination with the given position. It uses an enumeration without the possibility to add "other"-entries.
crossingInPosition designator @entry, @register	x	
crossingInPosition @id	x	
crossingInPosition givenPosition ...	o	The particular position of the movable crossing may be given if defined so for the active work zone.
crossingInPosition @protectingSide	o	The selection of protection use may be set in combination with the given position. It uses an enumeration without the possibility to add "other"-entries.
detectorInState designator @entry, @register	x	
detectorInState @id	x	
detectorInState givenState ...	o	The particular state of the detector may be given if defined so for the active work zone.
detectorInState @protectingSide	o	The selection of protection use may be set in combination with the given state. It uses an enumeration without the possibility to add "other"-entries.
signalWithAspect designator @entry, @register	x	
signalWithAspect @id	x	
signalWithAspect givenAspect ...	o	The particular aspect of the signal may be given if defined so for the active work zone.

signalWithAspect @protectingSide	o	The selection of protection use may be set in combination with the given aspect. It uses an enumeration without the possibility to add "other"-entries.
keyLockInState designator @entry, @register	x	
keyLockInState @id	x	
keyLockInState givenState ...	o	The particular state of the key lock may be given if defined so for the active work zone.
keyLockInState @protectingSide	o	The selection of protection use may be set in combination with the given state. It uses an enumeration without the possibility to add "other"-entries.
levelCrossingInState designator @entry, @register	x	
levelCrossingInState @id	x	
levelCrossingInState givenState ...	o	The particular state of the level crossing may be given if defined so for the active work zone.
levelCrossingInState @protectingSide	o	The selection of protection use may be set in combination with the given state. It uses an enumeration without the possibility to add "other"-entries.
releasedForLocalOperation @ref	o	The list of references to movable elements may be given if they can be operated locally with the zone active.
designator @entry, @register	m	The work zone shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.
assetsForIL localOperationArea		
isLimitedBy @ref	m	The list of references to the track assets limiting the area shall be used for defining its extent.
deactivationKey @ref	o	The reference to a track asset used for returning authority of operation back to the signal man may be given.
switchInPosition designator @entry, @register	x	
switchInPosition @id	x	
switchInPosition givenPosition ...	o	The particular position of the switch may be given if defined so for the active local operation area.
switchInPosition @protectingSide	o	The selection of protection use may be set in combination with the given position. It uses an enumeration without the possibility to add "other"-entries.
derailerInPosition designator @entry, @register	x	
derailerInPosition @id	x	
derailerInPosition givenPosition ...	o	The particular position of the derailer may be given if defined so for the active local operation area.
derailerInPosition @protectingSide	o	The selection of protection use may be set in combination with the given position. It uses an enumeration without the possibility to add "other"-entries.
crossingInPosition designator @entry, @register	x	
crossingInPosition @id	x	

crossingInPosition givenPosition ...	o	The particular position of the movable crossing may be given if defined so for the active local operation area.
crossingInPosition @protectingSide	o	The selection of protection use may be set in combination with the given position. It uses an enumeration without the possibility to add "other"-entries.
detectorInState designator @entry, @register	x	
detectorInState @id	x	
detectorInState givenState ...	o	The particular state of the detector may be given if defined so for the active local operation area.
detectorInState @protectingSide	o	The selection of protection use may be set in combination with the given state. It uses an enumeration without the possibility to add "other"-entries.
signalWithAspect designator @entry, @register	x	
signalWithAspect @id	x	
signalWithAspect givenAspect ...	o	The particular aspect of the signal may be given if defined so for the active local operation area.
signalWithAspect @protectingSide	o	The selection of protection use may be set in combination with the given aspect. It uses an enumeration without the possibility to add "other"-entries.
keyLockInState designator @entry, @register	x	
keyLockInState @id	x	
keyLockInState givenState ...	o	The particular state of the key lock may be given if defined so for the active local operation area.
keyLockInState @protectingSide	o	The selection of protection use may be set in combination with the given state. It uses an enumeration without the possibility to add "other"-entries.
levelCrossingInState designator @entry, @register	x	
levelCrossingInState @id	x	
levelCrossingInState givenState ...	o	The particular state of the level crossing may be given if defined so for the active local operation area.
levelCrossingInState @protectingSide	o	The selection of protection use may be set in combination with the given state. It uses an enumeration without the possibility to add "other"-entries.
releasedForLocalOperation @ref	m	The list of references to movable elements shall be given if they can be operated locally with the zone active.
designator @entry, @register	m	The local operation area shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.
assetsForIL ... switchInPosition givenPosition		
relatedSwitchAndPosition refersToSwitch @ref	m	The reference to the switch in the asset list is mandatory in the schema.
relatedSwitchAndPosition @inPosition	m	The position of the switch is mandatory in the schema.
@mustOrShould	o	The level of position enforcement is recommended to be given.

@proving	o	The way of proving the position is recommended to be given.
@isNegated	o	In case the position shall be anything else but this one the attribute is to be used. With only two positions possible it is not useful for a switch.
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to.

assetsForIL ... derailerInPosition givenPosition		
relatedDerailerAndPosition refersToDerailer @ref	m	The reference to the derailer in the asset list is mandatory in the schema.
relatedDerailerAndPosition @inPosition	m	The position of the derailer is mandatory in the schema.
@mustOrShould	o	The level of position enforcement is recommended to be given.
@proving	o	The way of proving the position is recommended to be given.
@isNegated	o	In case the position shall be anything else but this one the attribute is to be used. With only two positions possible it is not useful for a derailer.
designator @entry, @register	x	
@id	o	The marking of the derailer with @id is optional. The ID is never referred to.

assetsForIL ... crossingInPosition givenPosition		
relatedCrossingAndPosition refersToCrossing @ref	m	The reference to the movable crossing in the asset list is mandatory in the schema.
relatedCrossingAndPosition @inPosition	m	The position of the movable crossing is mandatory in the schema.
@mustOrShould	o	The level of position enforcement is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@proving	o	The way of proving the position is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@isNegated	o	In case the position shall be anything else but this one the attribute is to be used. With only two positions possible it is not useful for a movable crossing.
designator @entry, @register	x	
@id	o	The marking of the movable crossing with @id is optional. The ID is never referred to.

assetsForIL ... detectorInState givenState		
relatedDetectorAndState refersToDetector @ref	m	The reference to the detector in the asset list is mandatory in the schema.
relatedDetectorAndState @inState	m	The state of the detector is mandatory in the schema.
@mustOrShould	o	The level of state enforcement is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.

@proving	o	The way of state the position is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@isNegated	o	In case the state shall be anything else but this one the attribute is to be used.
designator @entry, @register	x	
@id	o	The marking of the detector with @id is optional. The ID is never referred to.

assetsForIL ... signalWithAspect givenAspect		
relatedSignalAndAspect refersToSignal @ref	m	The reference to the signal in the asset list is mandatory in the schema.
relatedSignalAndAspect showsAspect @ref	m	The reference to the shown aspect of the signal is mandatory in the schema. There may more than one reference if the complete aspect is composed of several ones.
@mustOrShould	o	The level of aspect enforcement is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@proving	o	The way of aspect the position is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@isNegated	o	In case the aspect shall be anything else but this one the attribute is to be used.
designator @entry, @register	x	
@id	o	The marking of the signal with @id is optional. The ID is never referred to.

assetsForIL ... keyLockInState givenState		
relatedKeyLockAndState refersToKeyLock @ref	m	The reference to the key lock in the asset list is mandatory in the schema.
relatedKeyLockAndState @inState	m	The state of the key lock is mandatory in the schema.
@mustOrShould	o	The level of state enforcement is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@proving	o	The way of proving the state is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@isNegated	o	In case the state shall be anything else but this one the attribute is to be used. With only two states possible it is not useful for a key lock.
designator @entry, @register	x	
@id	o	The marking of the key lock with @id is optional. The ID is never referred to.

assetsForIL ... levelCrossingInState givenState		
relatedLevelCrossingAndState refersToSwitch @ref	m	The reference to the level crossing in the asset list is mandatory in the schema.
relatedLevelCrossingAndState @inState	m	The state of the level crossing is mandatory in the schema.

@mustOrShould	o	The level of state enforcement is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@proving	o	The way of proving the state is recommended to be given. It uses an enumeration without the possibility to add "other"-entries.
@isNegated	o	In case the state shall be anything else but this one the attribute is to be used.
designator @entry, @register	x	
@id	o	The marking of the level crossing with @id is optional. The ID is never referred to.

assetsForIL shuntingZone		
isLimitedBy @ref	m	The list of references to the track assets limiting the area shall be used for defining its extent.
designator @entry, @register	m	The shunting zone shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL permissionZone		
canBeControlledBy @ref	m	The reference to the controllers which can have operational authority of this zone shall be given.
controlledElement @ref	m	The reference to all elements which can be commanded within this zone shall be given.
designator @entry, @register	m	The permission zone shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL interface		
command @bitNr	m	The non-negative integer value shall be given to define the bit position or just the order number of the command sent out.
command @description	m	The name or meaning of the command sent out shall be given.
command @normalState	m	The normal status of the command sent out (open/closed) shall be given.
command @pulseDuration	o	The duration value may be given if the command is sent out temporarily as pulse.
message @bitNr	m	The non-negative integer value shall be given to define the bit position or just the order number of the received message.
message @description	m	The name or meaning of the received message shall be given.
message @normalState	m	The normal status of the received message (open/closed) shall be given.
message @pulseDuration	o	The duration value may be given if the received message is expected as pulse.
initStatus @comString	o	The initial status of the command side may be given.
initStatus @messString	o	The initial status of the message side may be given.
@invalidTolerationTime	m	The duration value shall be given to control the valid status of the interface.

@switchoverTolerationTime	o	The duration value needed for normal status change may be given.
designator @entry, @register	o	The interface may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL routeReleaseGroupAhead		
@isAutomatic	m	The Boolean value shall be indicating whether the sections in the group are automatically released.
@delay	m	The duration value used for delay release after use shall be given.
@typicalDelay	o	The duration value used to indicate the typical time needed for release after use may be given.
hasTvdSection @ref	m	The list of references to the TVD sections contained in this partial route shall be given.
designator @entry, @register	o	The partial route may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL routeReleaseGroupRear		
@delay	m	The duration value used for delay release after use shall be given.
@typicalDelay	o	The duration value used to indicate the typical time needed for release after use may be given.
hasTvdSection @ref	m	The list of references to the TVD sections contained in this partial route shall be given.
designator @entry, @register	o	The partial route may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL route		
handlesRouteType @ref	m	The reference to the IM specific route type is required.
routeActivationSection	m	The route activation condition shall be given.
facingSwitchInPosition ...	m	The distinct definition of route path by relevant switch positions is required. It may be used for any other movable element type than switches if really needed to define the path.
routeEntry	m	The definition of the start of the route is mandatory in the schema.
hasTvdSection @ref	m	The reference to any TVD section within the running path of the route shall be given. This shall be preferable an ordered list starting from route entry and it shall conform to the path defined by the switch positions.
hasReleaseGroup @ref	o	The list of references to release groups may be given if needed for route release.
switchPositionInDepartureTrack ...	o	The defined position of any switch in the departure track (in rear of route entry) may be needed.
routeExit	m	The definition of the destination of the route is mandatory in the schema.

additionalRelation @ref	o	The reference to route relations which shall be considered in addition may be listed.
@locksAutomatically	o	The Boolean value may be given if automatic route setting is used.
@processingDelay	o	The duration value may be given if the route locking is delayed after the trigger event for setting it.
@proceedAspectDelay	o	The duration value may be given if there is a delay for opening the start signal after the route is locked.
@signalClosureDelay	o	The duration value may be given if there is a delay for closing the start signal after the trigger event by the passing train. It shall be coordinated with the @nonReplacement of the route entry.
@approachReleaseDelay	o	The duration value may be given if the interlocking supports the special handling of route release on approach.
designator @entry, @register	m	The route shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL route routeActivationSection		
@delayForLock	m	The duration value shall be given to allow control of route activation.
@automaticReleaseDelay	m	The duration value shall be given to allow control of release of activated route.
activationSection @ref	m	The reference to the TVD section used as trigger for route setting/locking is required.
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

assetsForIL route facingSwitchInPosition		
refersToSwitch @ref	m	The reference to the switch in the asset list is mandatory in the schema.
@inPosition	m	The required position of the switch is mandatory in the schema. It uses an enumeration without the possibility to add "other"-entries.
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

assetsForIL route routeEntry		
refersTo @ref	m	The reference to the track asset marking the start of the route is mandatory in the schema. This is typical the reference to a signal of the interlocking assets.
nonReplacement @ref	o	The reference to TVD sections in advance to the route entry may be given to control signal closure depending on train position. It shall be coordinated with @signalClosureDelay of the route.
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

assetsForIL route routeExit		
refersTo @ref	m	The reference to the track asset marking the destination of the route is mandatory in the schema. This is typical the reference to a signal of the interlocking assets.
hasDangerpoint @ref	o	The reference to a related danger point may be given.
hasOverlap @ref	m	The reference to one or more related overlaps shall be given.
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

assetsForIL route switchPositionInDepartureTrack		
refersToSwitch @ref	m	The reference to the switch in the asset list is mandatory in the schema.
@inPosition	m	The required position of the switch is mandatory in the schema:
designator @entry, @register	x	
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

assetsForIL conflictingRoutes		
refersTo @ref	m	The reference to the route having a conflict shall be given.
conflictsWithRoute @ref	m	The reference to a route causing the conflict shall be given.
reasonForConflict @origin	m	The reasons for the conflict shall be given. It uses an enumeration without the possibility to add "other"-entries.
reasonForConflict @refersTo	m	The reference to the element causing the conflict shall be given.
designator @entry, @register	o	The conflicting route pairing may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL routeRelations		
requiredSwitchPosition ...	o	instance of type AssetAndGivenState
requiredDeraillerPosition ...	o	instance of type AssetAndGivenState
requiredCrossingPosition ...	o	instance of type AssetAndGivenState
requiredDetectorState ...	o	instance of type AssetAndGivenState
requiredSignalAspect ...	o	instance of type AssetAndGivenState
requiredSectionState ...	o	instance of type AssetAndGivenState
requiredKeyLockState ...	o	instance of type AssetAndGivenState
requiredLevelCrossingState ...	o	instance of type AssetAndGivenState
... {AssetAndGivenState} designator @entry, @register	x	
... {AssetAndGivenState} @id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.
... {AssetAndGivenState} refersTo... @ref	m	The reference to a related track asset shall be given.
... {AssetAndGivenState} ...InPosition/State	m	The position or state of the related track asset shall be given.

@mustOrShould	m	The level of state enforcement is required. It uses an enumeration without the possibility to add "other"-entries.
@proving	m	The way of state proving is required. It uses an enumeration without the possibility to add "other"-entries.
@isNegated	o	The Boolean value may be set in case the position or state shall be any other but the listed one. It is only useful if there are more than two positions or states available for the element.
designator @entry, @register	o	The route relation may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL combinedRoute		
comboEntry @ref	m	The reference to the entry point of the route combination is required.
comboExit @ref	m	The reference to the exit point of the route combination is required.
containsRoute @ref	m	The list of references to the single routes in the combination is required. The list shall be preferable in the order of the routes.
designator @entry, @register	m	The combined route shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

assetsForIL overlap		
activeForApproachRoute @ref	m	The list of references to the related routes of the overlap shall be given. This is the indirect reverse link to the route exit elements.
relatedToTrackAsset @ref	o	The reference to the related track asset of the overlap may be given. This is the link to a signal or so used as route exit element.
requiresSwitchInPosition ...	m	The switch in overlap required to be in a particular position shall be given for definition of the overlap path, i.e. the reference to the switch in asset list and its position.
hasTvdSection @ref	m	The reference to any TVD section within the path of the overlap shall be given. This shall be preferable an ordered list starting from route exit and it shall conform to the path defined by the switch positions.
isLimitedBy @ref	m	The reference to the limiting train detection element shall be given.
overlapRelease	m	The condition for overlap release is required.
overlapRelease releaseTriggerSection @ref	m	The reference to the TVD section triggering the overlap release is required.
overlapRelease overlapReleaseTimer	m	The timer for overlap release is required.
overlapRelease overlapReleaseTimer timer	m	The duration value is required to set up the release timer.
overlapRelease overlapReleaseTimer overlapReleaseCondition	m	The detail of the trigger condition is required. It uses an enumeration without the possibility to add "other"-entries.
@overlapSpeed	o	The speed value in km/h may be given.
@releaseSpeed	o	The speed value in km/h may be given.

@overlapValidityTime	o	The duration value may be given.
@length	o	The length value of the overlap in metres may be given. It shall be coordinated with the limiting train detection element.
designator @entry, @register	o	The overlap may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL dangerPoint		
lastSupervisedSectionBeforeDP @ref	o	The reference to the TVD section may be given.
situatedAtTrackAsset @ref	o	The reference to the underlying track in infrastructure may be given.
@distance	m	The length value in metres until the danger point is required.
@releaseSpeed	o	The speed value may be given.
designator @entry, @register	o	The danger point may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL destinationPoint		
refersTo @ref	m	The reference to an element in the infrastructure part is mandatory in the schema.
hasDangerPoint @ref	o	The reference to a related danger point may be given.
hasOverlap @ref	o	The reference to one or more related overlaps may be given.
designator @entry, @register	o	The destination point may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

assetsForIL powerSupplyIL		
@numberOfSimultaneousSwitching Actuators	m	The non-negative integer value is required to control the switch actuator activation.
@signalVoltageMode	m	The control mode of the signal voltage is required. It uses an enumeration without the possibility to add "other"-entries.
designator @entry, @register	o	The power supply may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

2.2.3.3 signalBox

signalBox		
controlsSystemAsset	m	The list of system assets known to the interlocking is required.
controlsTrackAsset	m	The list of track assets known to the interlocking is required.
controlsRoute @ref	m	The list of references to the routes controlled by the interlocking is required.

controlsCombinedRoute @ref	o	The list of references to the route combinations controlled by the interlocking may be given.
controlsInterface	m	The list of physical interfaces known to the interlocking is required.
controlledBy @ref	o	The reference to the controllers that are connected to the interlocking may be given.
implementsSignalplan	m	The signal plan for the interlocking is required.
implementsElementGroup	m	The list of element groups known to the interlocking is required.
hasPermissionZone @ref	m	The list of references to the permission zones within the area controlled by this interlocking shall be given.
hasConflictingRoutes @ref	o	The list of references to the route conflicts known to the interlocking may be given.
hasConfiguration	o	The details of interlocking configuration may be given.
designator @entry, @register	m	The signal box (interlocking) shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

signalBox controlsSystemAsset		
connectedSystemAsset @ref	m	The reference to the system asset from the list is required.
@extentOfControl	m	The control level is required. It uses an enumeration without the possibility to add "other"-entries.
designator @entry, @register	o	The system asset relation may have a name (from the IM) to enhance readability.
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

signalBox controlsTrackAsset		
connectedTrackAsset @ref	m	The reference to the track asset from the list is required.
@extentOfControl	m	The control level is required. It uses an enumeration without the possibility to add "other"-entries.
designator @entry, @register	o	The track asset relation may have a name (from the IM) to enhance readability.
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

signalBox controlsInterface		
lastOwnTvdSection @ref	m	The reference to the TVD section is required.
firstRemoteTvdSection @ref	m	The reference to the TVD section is required.
incomingRoute @ref	m	The reference to the route is required.
outgoingRoute @ref	m	The reference to the route is required.
hasInterface @ref	m	The reference to the physical interface is required.
@interfaceLocation	m	The topological location of the interface is required. It uses an enumeration without the possibility to add "other"-entries.
@isOnCommandSide	m	The Boolean value is required to define whether the interlocking is on the command side of the physical interface. If false the commands of the physical interface are messages and vice versa.
designator @entry, @register	o	The interface relation may have a name (from the IM) to

		enhance readability.
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

SignalBox implementsSignalplan		
aspectRelation designator @entry, @register	m	The signal plan shall have a name (from the IM) to enhance readability.
aspectRelation @id	m	The ID of any aspect relation is required for reference to it.

SignalBox implementsSignalplan aspectRelation		
masterAspect ...	o	The reference to the signal and the aspect shown at the route exit may be given. It is required if this aspect is to be announced at the route entry by the slave aspect. Instance of type SignalAndAspect
slaveAspect ...	m	The reference to the signal and the aspect shown at the route entry is required. Instance of type SignalAndAspect
distantAspect ...	x	The reference to the signal and the aspect shown at the distant signal related to the signal of the masterAspect. Instance of type SignalAndAspect
... {SignalAndAspect} designator @entry, @register	x	
... {SignalAndAspect} @id	o	The ID of the aspect relation may be given.
... {SignalAndAspect} refersToSignal @ref	m	The reference to the particular signal in the asset list shall be given.
... {SignalAndAspect} showsAspect @ref	m	The reference to the aspect shown at the signal shall be given.
signalsSpeedProfile @ref	x	
appliesToRoute @ref	m	The reference to the applicable route is required.
@passingSpeed	m	The speed value related to the slave aspect is required.
@expectingSpeed	o	The speed value related to the master aspect may be given.
@endSectionTime	o	The duration value may be given for controlling route release.
designator @entry, @register	o	The aspect relation may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

SignalBox implementsElementGroup		
groupType @ref	m	The reference to the particular group type from the IM specific data is required.
refersToMember @ref	m	The list of references to the assets included in this group is required.
designator @entry, @register	o	The element group may have a name (from the IM) to enhance readability.
@id	m	The ID is required as it is used to refer to it.

SignalBox hasConfiguration		
@model	m	The string shall contain the manufacturer name and type of the interlocking.
@technologyType	m	The basic technology type of the interlocking is required. It

		uses an enumeration without the possibility to add "other"-entries.
@SWversion	o	The string may contain the software version used in the interlocking.
@signalSystem	o	The string may contain the name of the signaling system the interlocking is controlling.
designator @entry, @register	o	The configuration may have a name (from the IM) to enhance readability.
@id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

2.2.3.4 Controller

Controller		
controlledAssets	m	The container for controlled signalboxes and system assets may be given.
itineraries	o	The container for itineraries is required.
designator @entry, @register	m	The controller shall have a name (from the IM).
@id	m	The ID is required as it is used to refer to it.

controller controlledAssets		
controlledInterlocking connectedSignalBox @ref	m	The reference to the signal box controlled from this place shall be given.
controlledInterlocking @extentOfControl	m	The extent of control over this signal box shall be given.
controlledInterlocking designator @entry, @register	o	The control relation may have a name (from the IM) to enhance readability.
controlledInterlocking @id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.
controlledSystemAsset connectedSystemAsset @ref	m	The reference to the system asset controlled from this place shall be given.
controlledSystemAsset @extentOfControl	m	The extent of control over this system asset shall be given.
controlledSystemAsset designator @entry, @register	o	The control relation may have a name (from the IM) to enhance readability.
controlledSystemAsset @id	o	The marking of the element with @id is optional. The ID is never referred to in this use case.

controller itineraries		
itinerary comboEntry @ref	m	The reference to the entry point of the route combination is required.
itinerary comboExit @ref	m	The reference to the exit point of the route combination is required.
itinerary containsRoute @ref	m	The list of references to the single routes in the combination is required. The list shall be preferable in the order of the routes.
itinerary designator @entry, @register	o	The combined route may have a name (from the IM) to enhance readability.
itinerary @id	o	The marking of the element with @id is optional. The ID is

		never referred to in this use case.
--	--	-------------------------------------

2.2.4 Timetable

No elements are required.

2.2.5 Rollingstock

No elements are required.

3 Additional Conditions

None.

4 References

- [1] railML.org Wiki: *Use case Interlocking Module Engineering Data*. In:
<https://wiki.railml.org/index.php?title=IL:UC:InterlockingEngineering>; last access:
14.08.2018
- [2] railML.org Wiki: *Use case Schematic Track Plan*. In:
https://wiki.railml.org/index.php?title=UC:IS:Schematic_Track_Plan; last access:
18.06.2018