Railway Domain Modeling

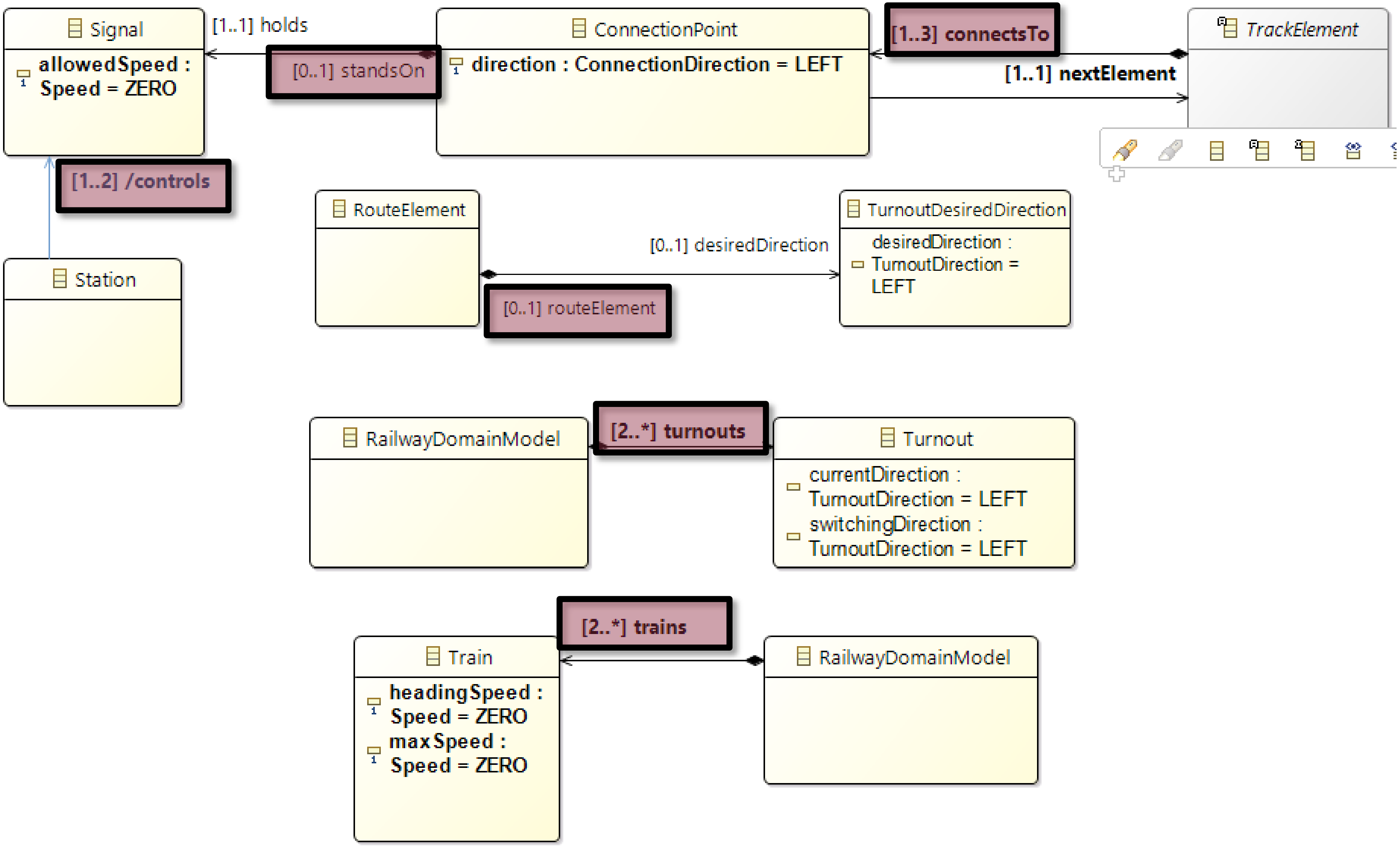
Homework Assignment #2
Team GoMRP
MDSD 2015

Benedek Horváth
Raimund Konnerth
Tamás Nádudvari

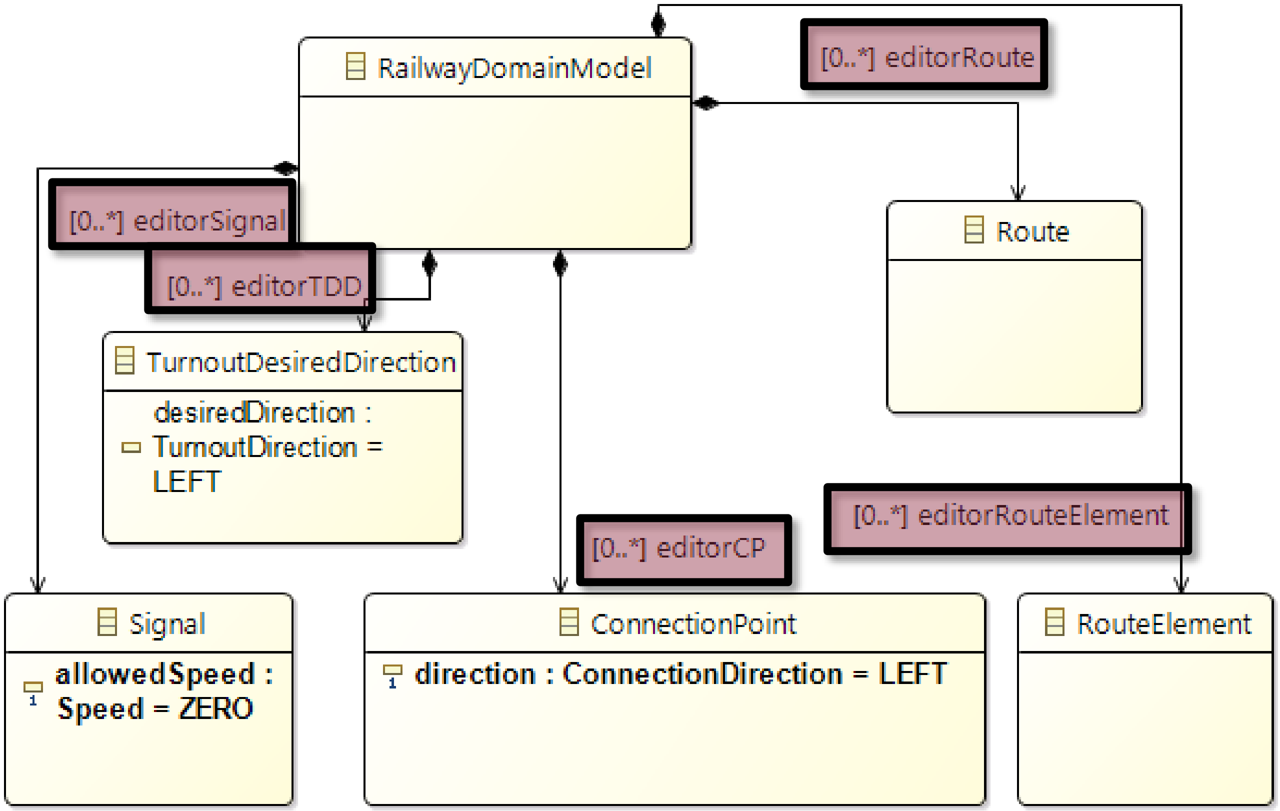
Homework specification

- Railway system
  - sections, turnouts and signals
  - trains
- Requirement: safe operation
  - prevent train collisions
  - prevent train derailment
- Homework Assignment 2
  - graphical and textual editors (Sirius and Xtext)
  - structural and behavioral instance models
  - research group’s model railway track’s instance model

Structural metamodel changes

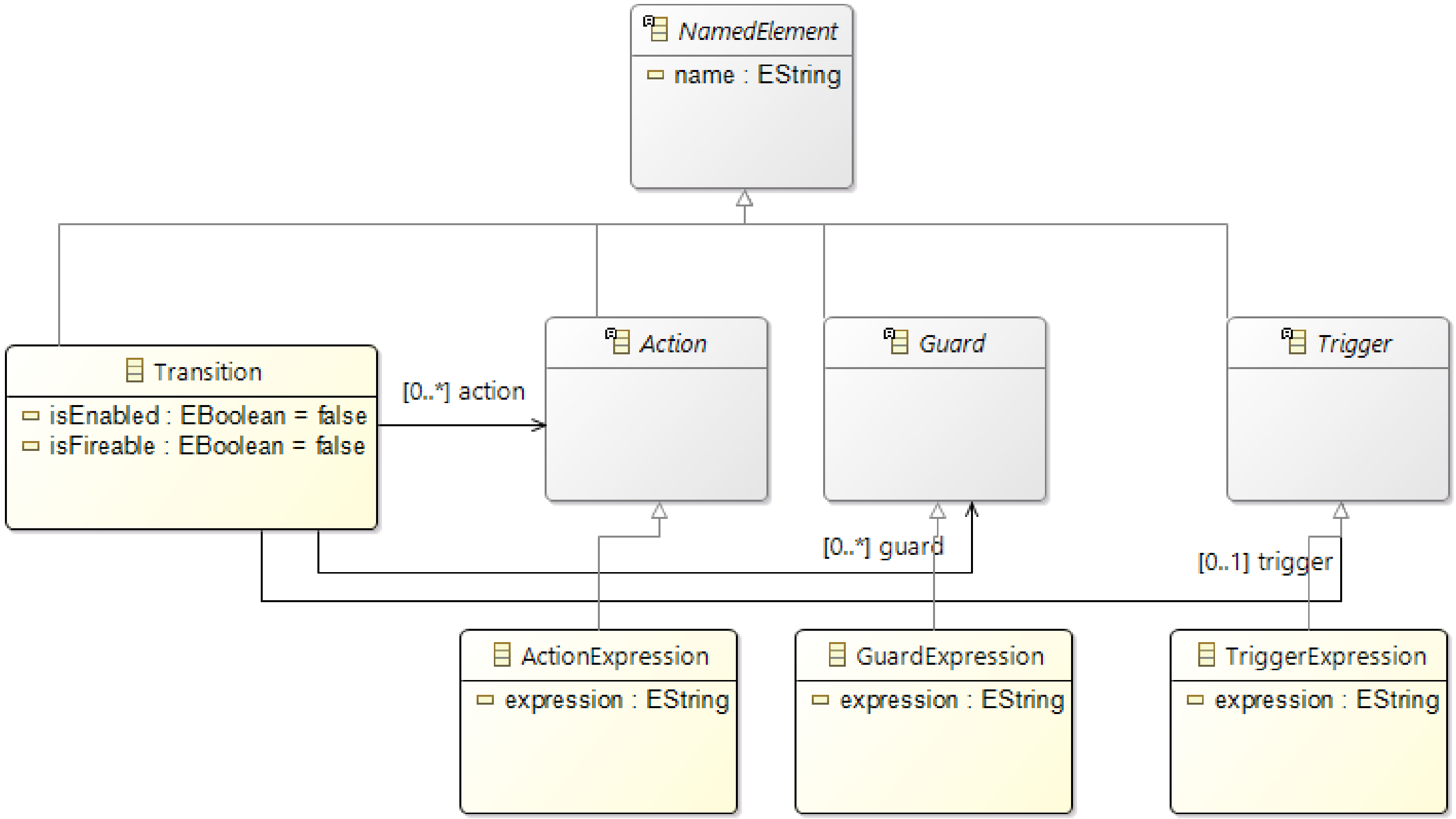


Structural metamodel changes



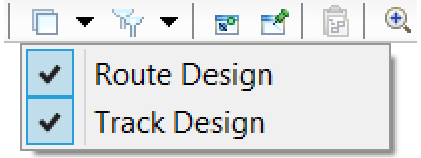
Behavioral metamodel changes

- Action, Guard, Trigger made abstract
  - ActionExpression, GuardExpression, TriggerExpression are inherited and made the superclass for the descendants

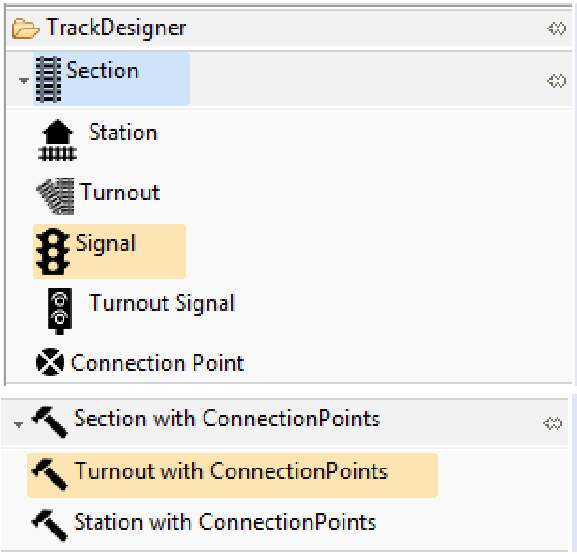


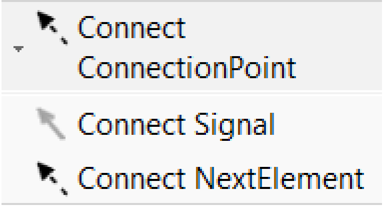
Graphical editor features

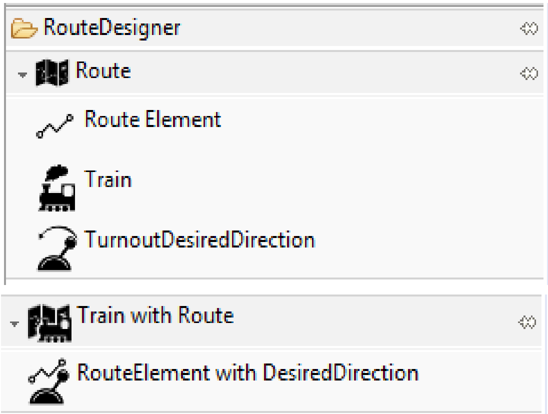
- Track and route elements are separated on different layers

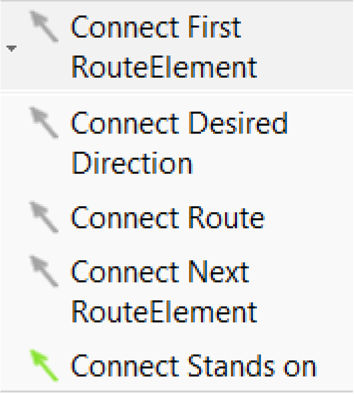


- Any elements can be placed as individual
- Related elements can be placed at once


TrackDesigner
Section
Station
Turnout
Signal
Turnout Signal
Connection Point
Section with ConnectionPoints
Turnout with ConnectionPoints
Station with ConnectionPoints






Connect First RouteElement
Connect Desired Direction
Connect Route
Connect Next RouteElement
Connect Stands on

Graphical editor features

- When an element is deleted, the related elements are also removed
- The edges are can be reconnect both at source and target ends
- Name of the element is validated with possible quick fix
- Unique ID can be generated for the elements
  - Uses IncQuery pattern to check uniqueness

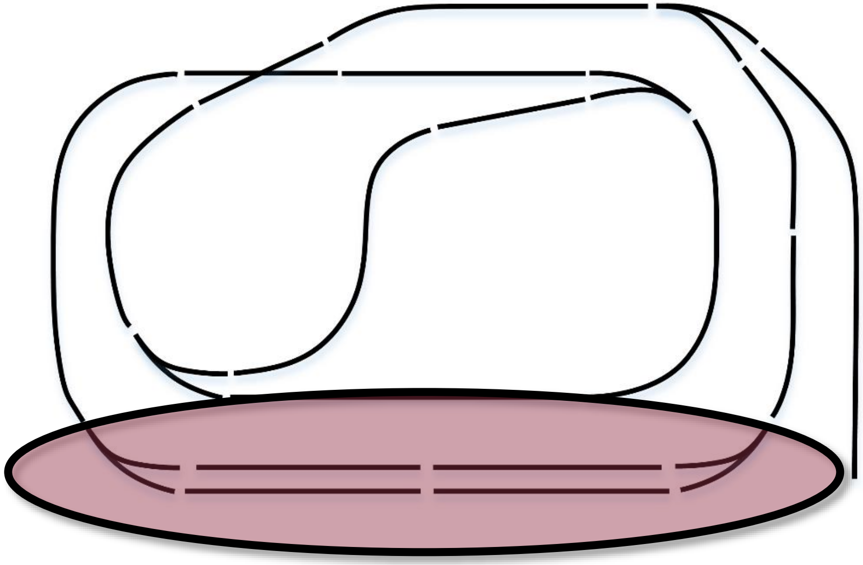
Graphical editor features

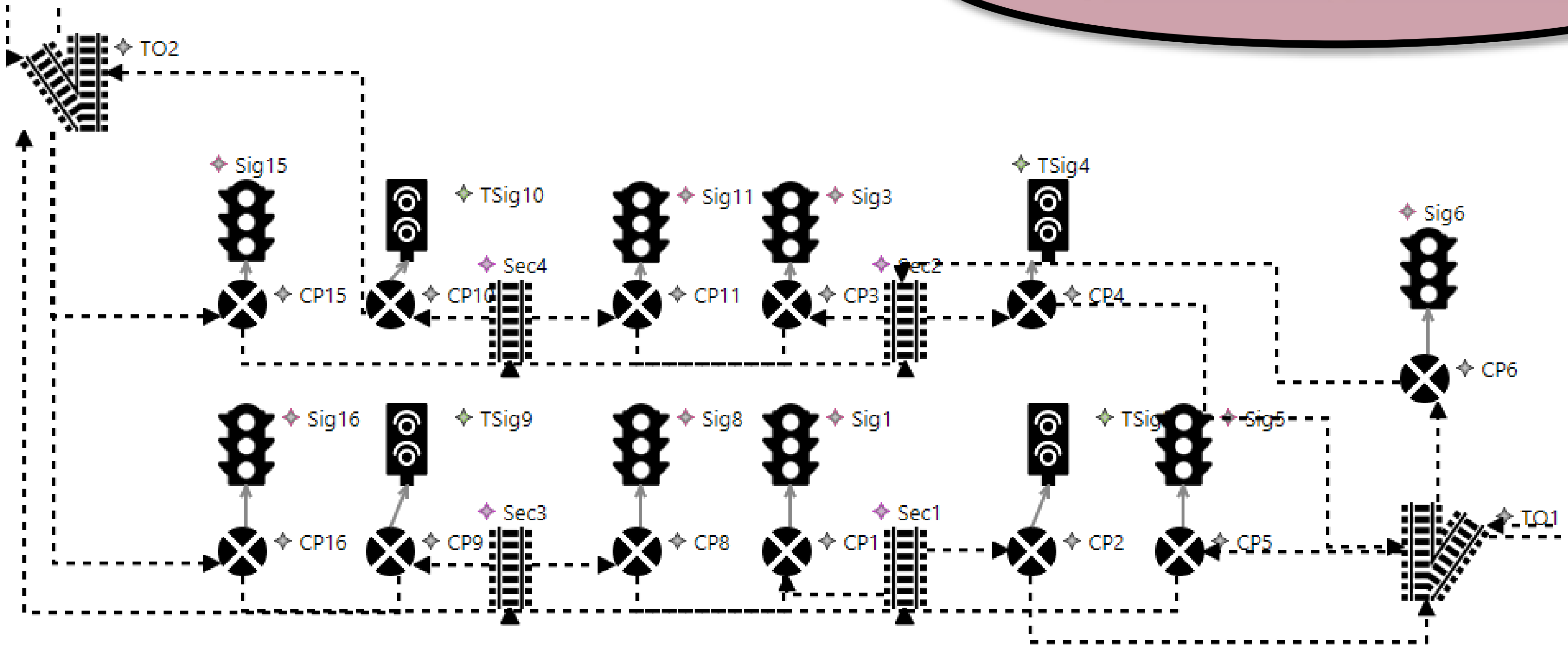
- New line of track can be started from an already placed track item
  - The placed item can be selected by double click
  - The function automatically chooses the correct track element and connects it
- New route for a train can be created by selecting the track elements from a popup menu
  - The order of the selection will be the order of route
- Condition based icons for turnouts
- IncQuery validations are enabled in the editor

Textual editor features

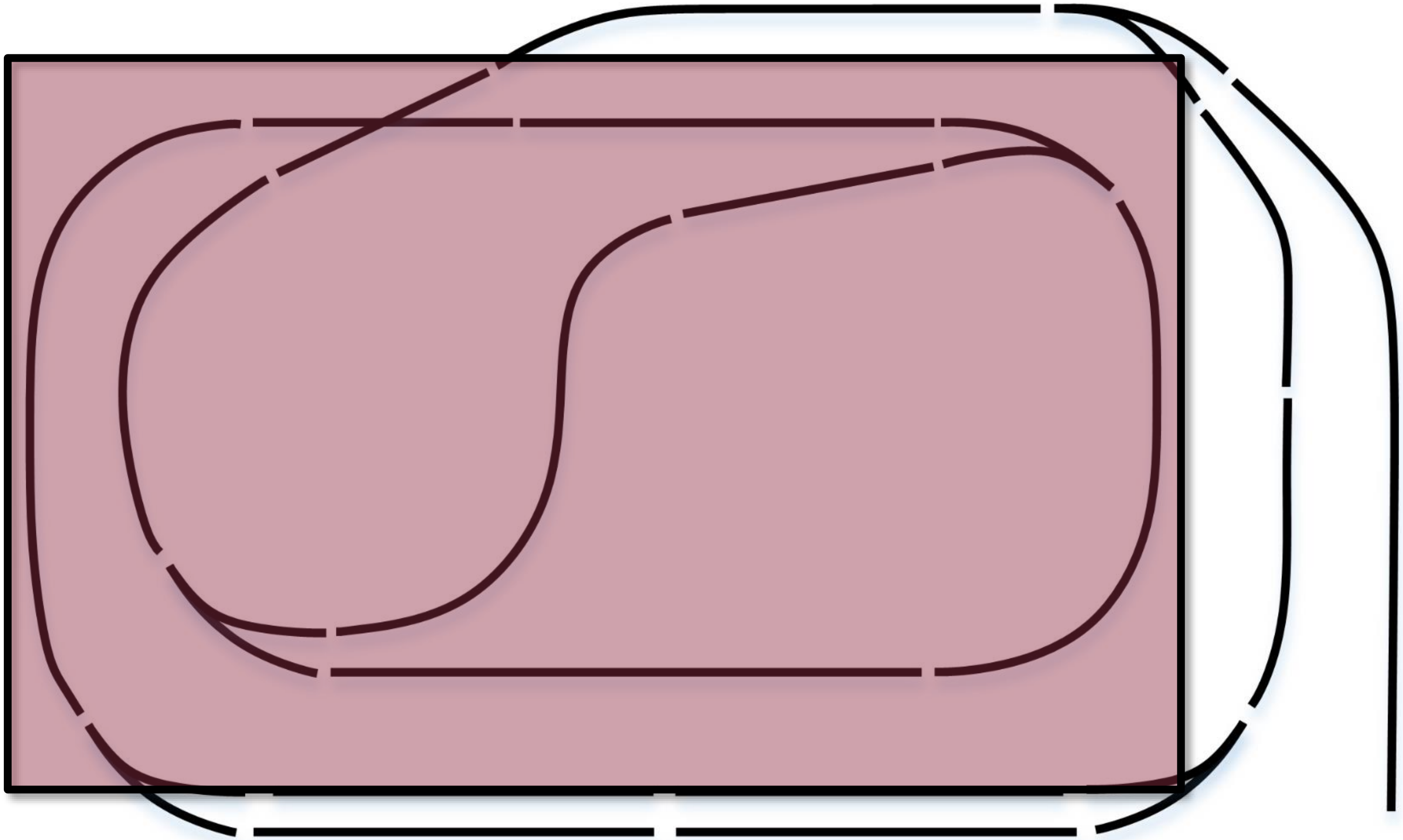
- Textual DSL for both behavioral and structural metamodel
- Structural DSL
  - limited (incomplete) grammar
  - created to allow cross-references from the behavioral DSL
  - multiple error markers in editor because of unsatisfied multiplicity constraints
- Behavioral DSL
  - Simplified grammar
  - transparency/understandability at the expense of completeness (full conformance) with the metamodel
  - cross-reference to the structural instance model elements
  - basic validations (only one state machine for areferred object)

Structural instance model

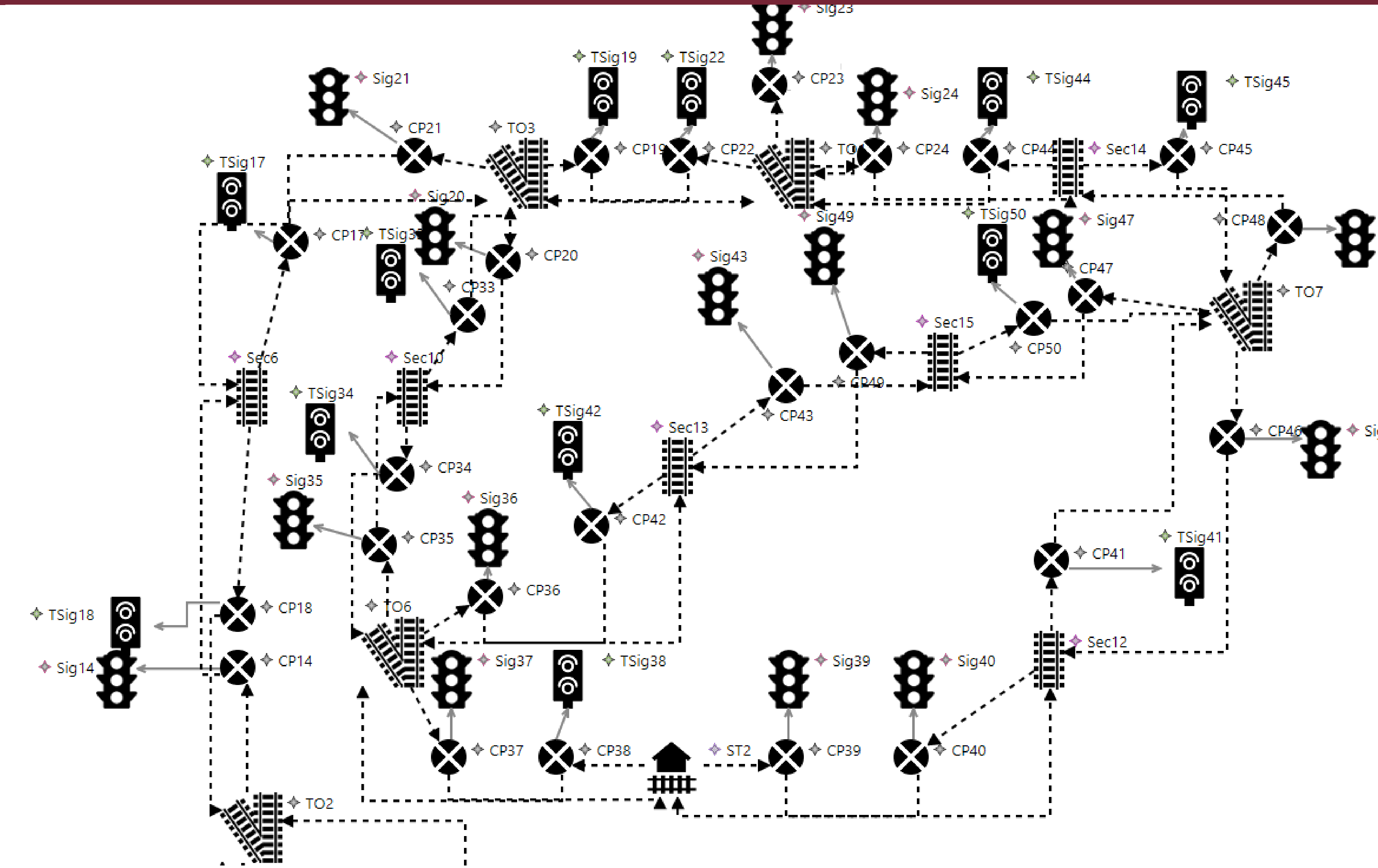




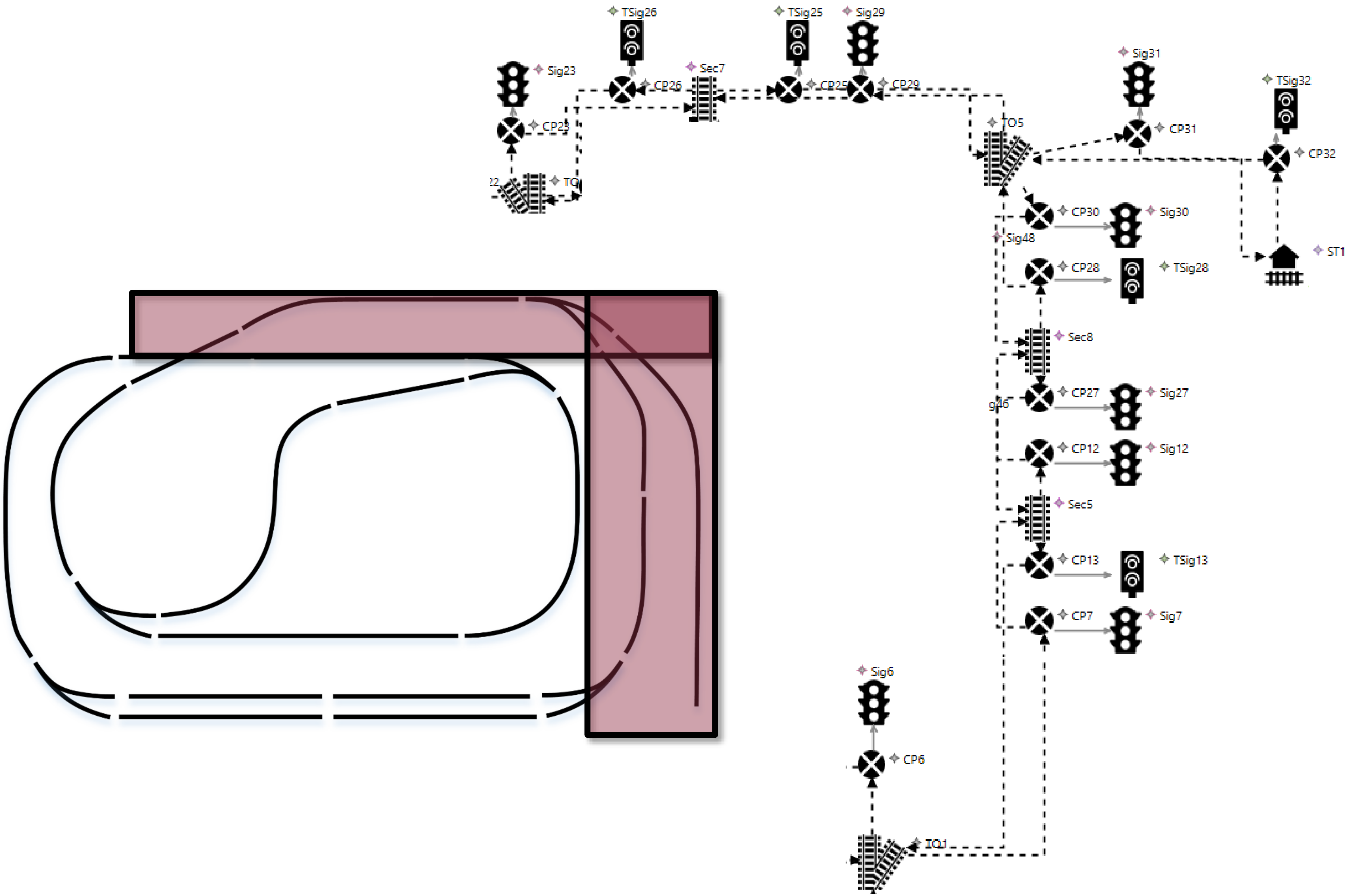
Structural instance model



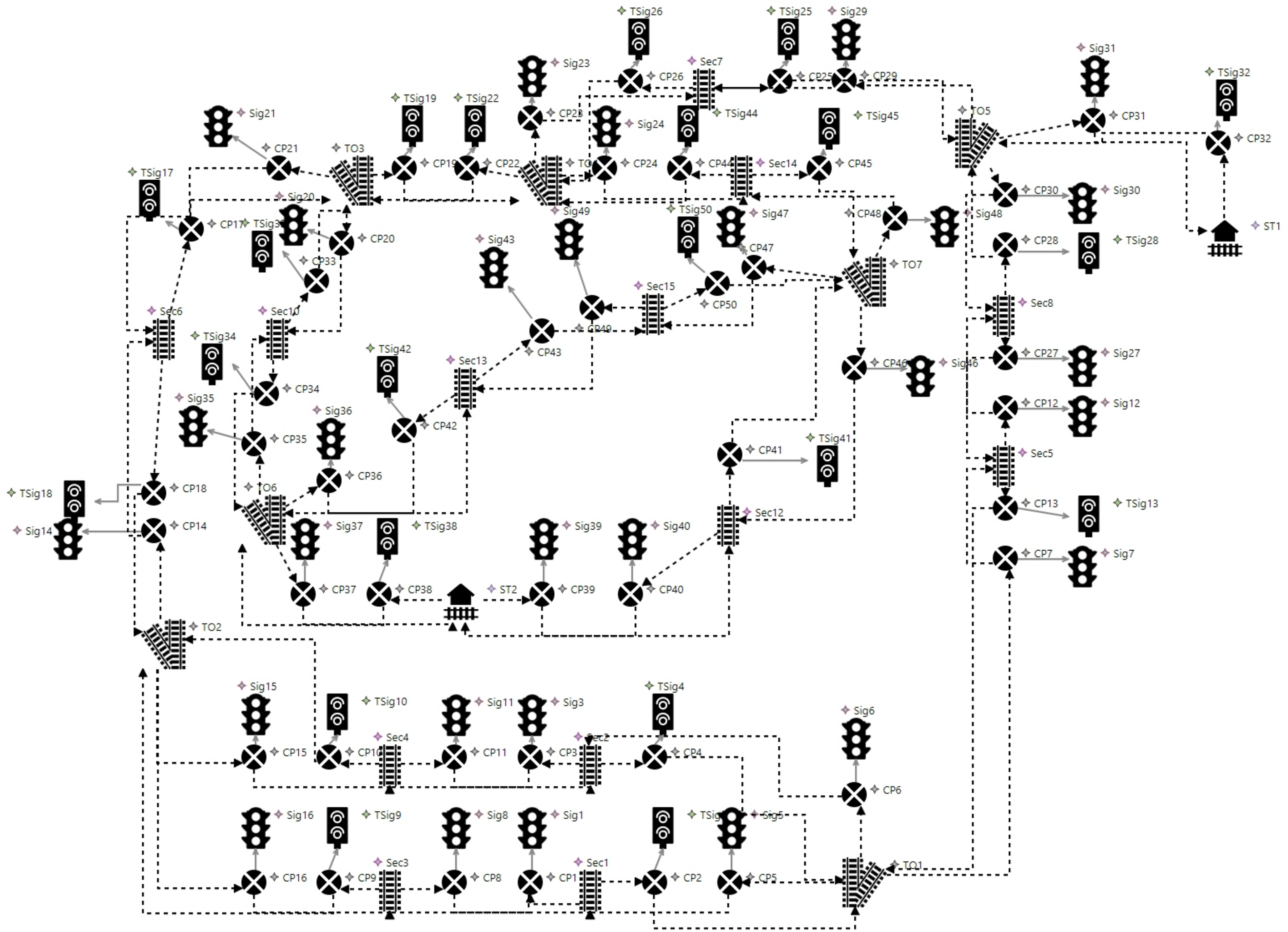
Structural instance model



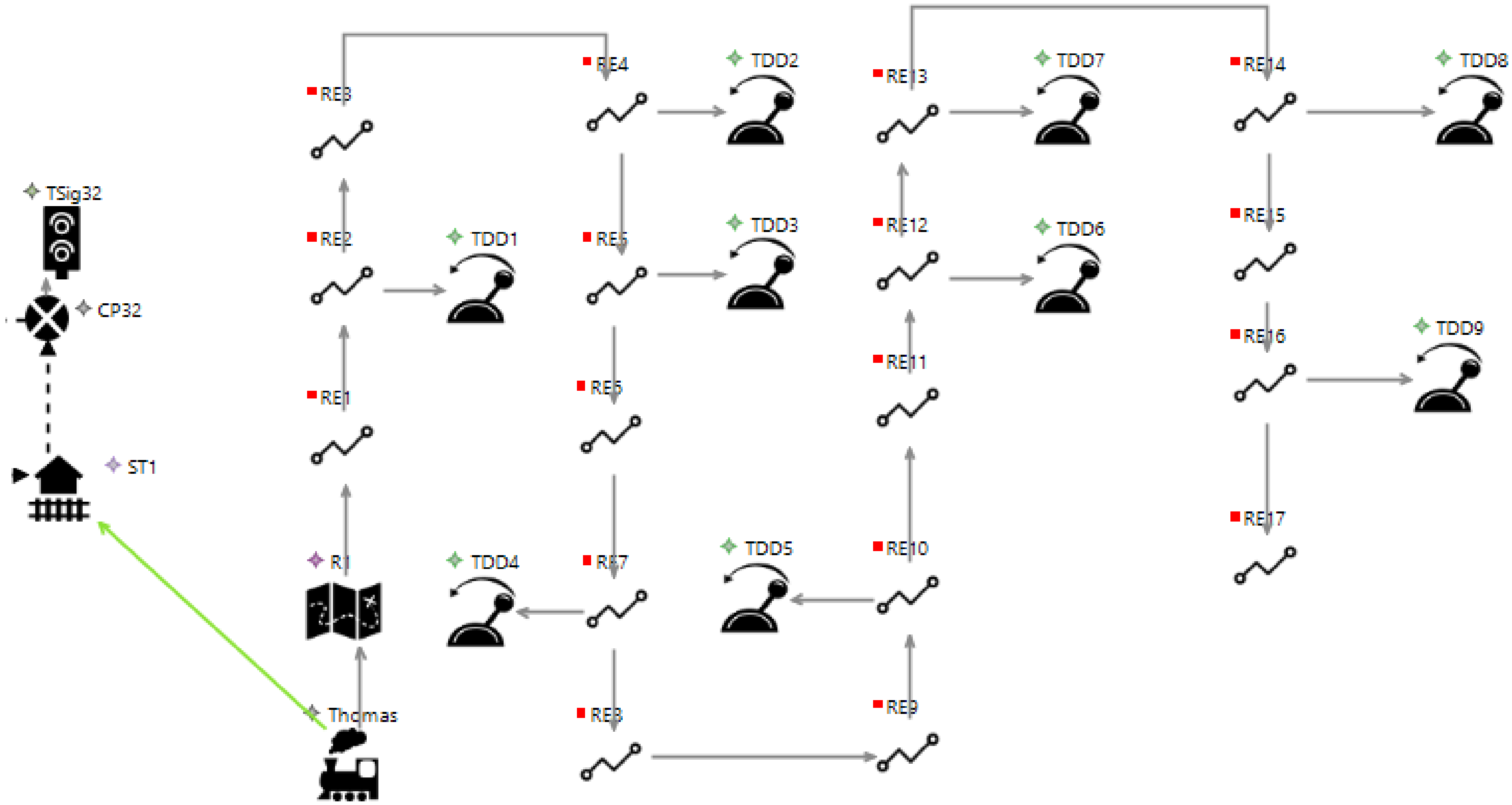
Structural instance model



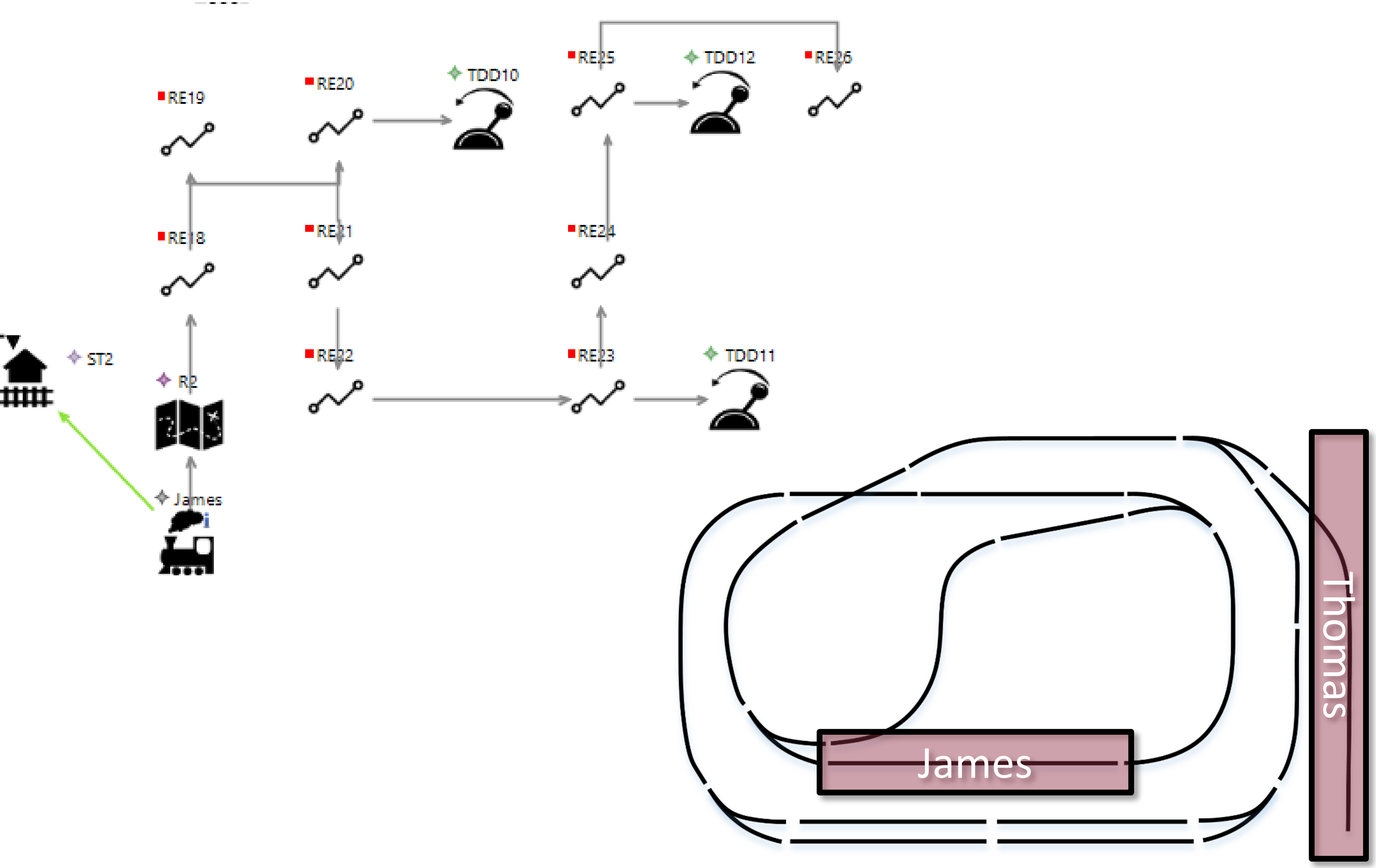
Structural instance model



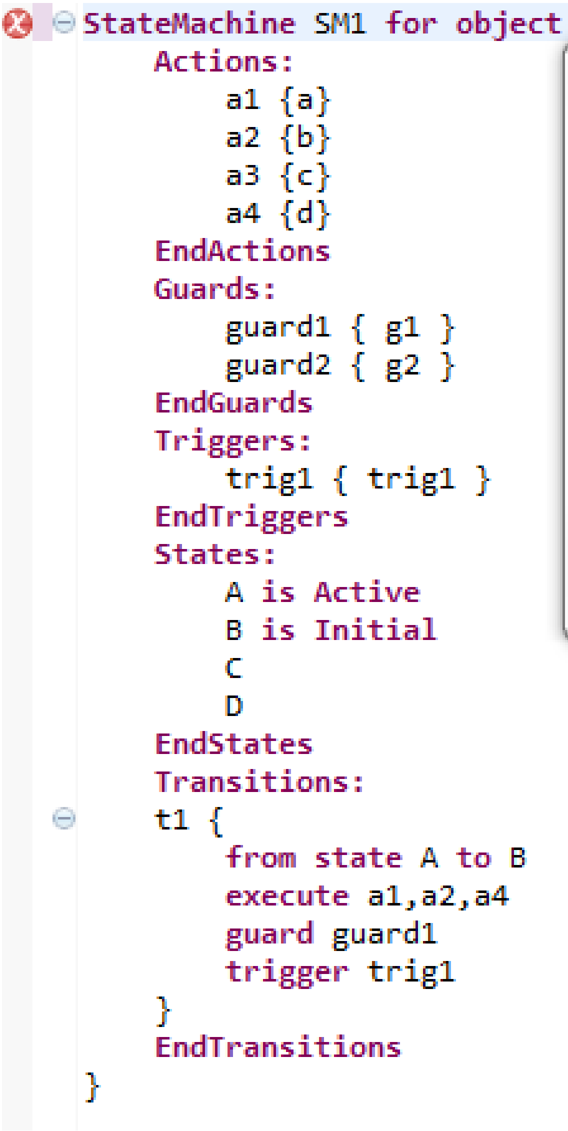
Structural instance model



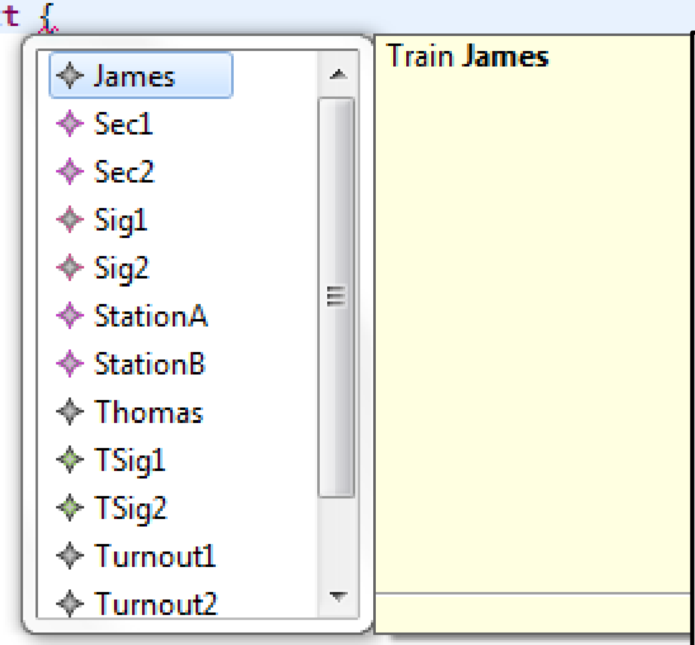
Structural instance model



Behavioral instance model



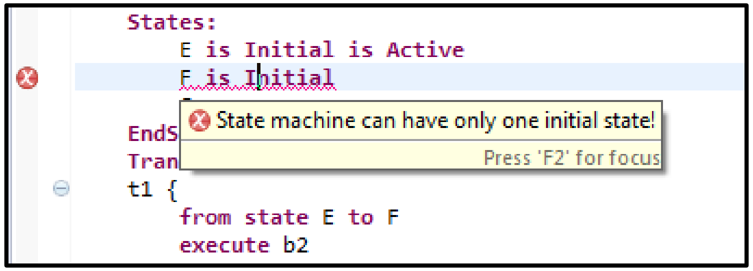
```
StateMachine SM1 for object {
    Actions:
        a1 {a}
        a2 {b}
        a3 {c}
        a4 {d}
    EndActions
    Guards:
        guard1 { g1 }
        guard2 { g2 }
    EndGuards
    Triggers:
        trig1 { trig1 }
    EndTriggers
    States:
        A is Active
        B is Initial
        C
        D
    EndStates
    Transitions:
    t1 {
        from state A to B
        execute a1,a2,a4
        guard guard1
        trigger trig1
    }
    EndTransitions
}
```



```
James
Sec1
Sec2
Sig1
Sig2
StationA
StationB
Thomas
TSig1
TSig2
Turnout1
Turnout2
```

Train James

```
RailwayDomainModel {
    Train Thomas
    Train James
    Section Sec1
    Section Sec2
    Station StationA
    Station StationB
    Turnout Turnout1
    Turnout Turnout2
    TurnoutSignal TSig1
    TurnoutSignal TSig2
    Signal Sig1
    Signal Sig2
}
```



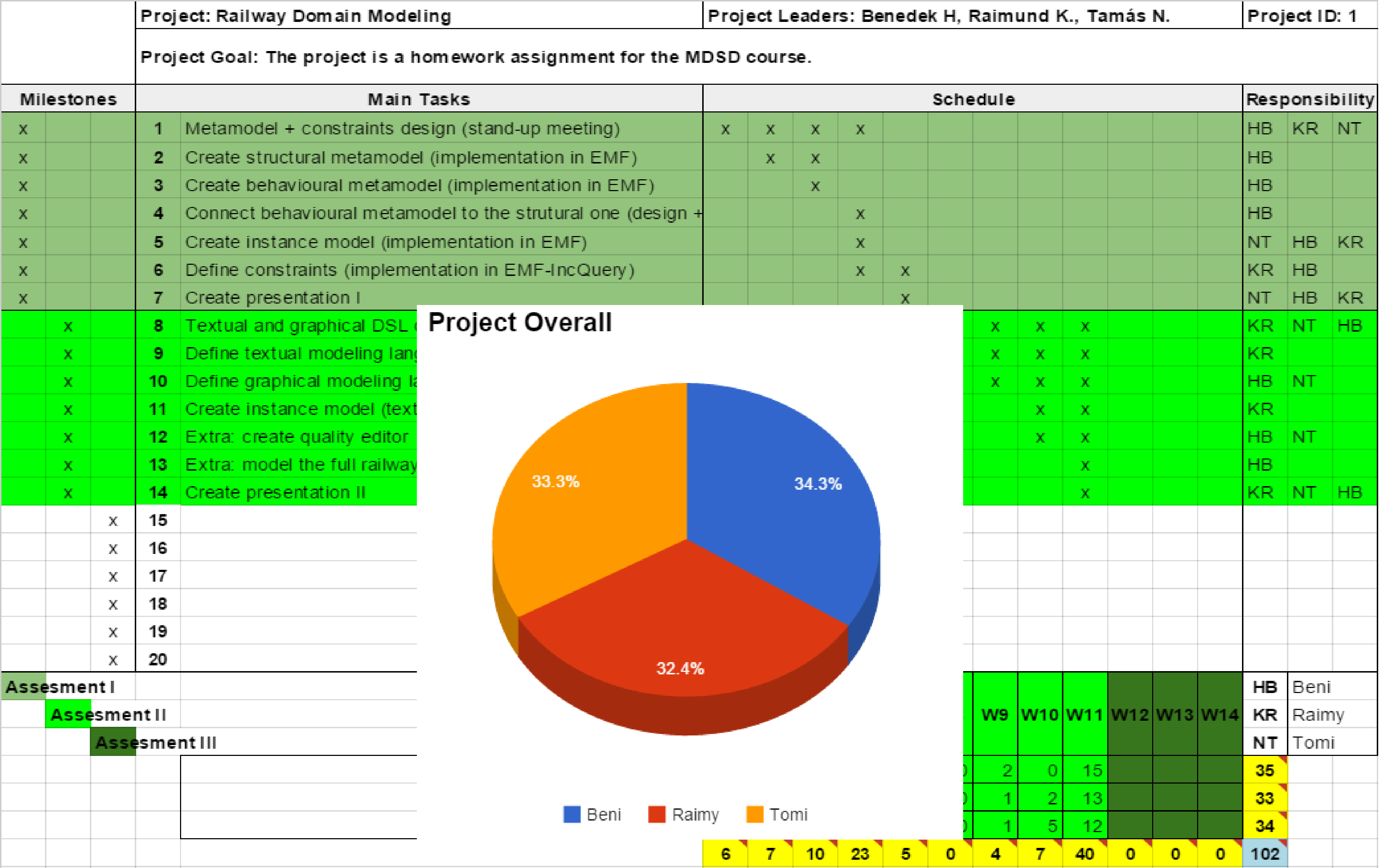
```
States:
    E is Initial is Active
    F is Initial
EndS
Tran
t1 {
    from state E to F
    execute b2
```

State machine can have only one initial state!

Press 'F2' for focus

Q & A

Division of labour



Project: Railway Domain Modeling | Project Leaders: Benedek H, Raimund K., Tamás N. | Project ID: 1

Project Goal: The project is a homework assignment for the MDSD course.

Milestones			Main Tasks		Schedule												Responsibility		
x			1	Metamodel + constraints design (stand-up meeting)	x	x	x	x									HB	KR	NT
x			2	Create structural metamodel (implementation in EMF)		x	x										HB		
x			3	Create behavioural metamodel (implementation in EMF)			x										HB		
x			4	Connect behavioural metamodel to the strutural one (design +				x									HB		
x			5	Create instance model (implementation in EMF)				x									NT	HB	KR
x			6	Define constraints (implementation in EMF-IncQuery)				x	x								KR	HB	
x			7	Create presentation I					x								NT	HB	KR
	x		8	Textual and graphical DSL							x	x	x				KR	NT	HB
	x		9	Define textual modeling lan							x	x	x				KR		
	x		10	Define graphical modeling l							x	x	x				HB	NT	
	x		11	Create instance model (text								x	x				KR		
	x		12	Extra: create quality editor								x	x				HB	NT	
	x		13	Extra: model the full railway									x				HB		
	x		14	Create presentation II									x				KR	NT	HB
		x	15																
		x	16																
		x	17																
		x	18																
		x	19																
		x	20																
Assesment I																	HB	Beni	
	Assesment II										W9	W10	W11	W12	W13	W14	KR	Raimy	
		Assesment III															NT	Tomi	
											2	0	15				35		
											1	2	13				33		
											1	5	12				34		
					6	7	10	23	5	0	4	7	40	0	0	0	102		