railML®: The use of interlocking data for engineering and for simulation

Dr. Bob Janssen; railML.org-IL-coordinator, Den Haag

26th railML conference · Paris, October 8th, 2014
Use Cases

Simulation

- Design timetables
- Test timetables
- Test scenarios

Engineering

- Unified data exchange
- Automated data input
- Humanless tool chain
The making of an interlocking

A) Generic interlocking

B) Implementation of IM rules and regulation

National realisations of the “generic interlocking”.

C) tailor the interlocking to individual station

realisations of a “national interlocking”.

Prorail

Infrabel

DB

Network Rail

SBB

Rotterdam

Dv

Dt

A

... Z
Future engineering workflow

No human intervention
Validated data input

High quality for less money.
Interlocking families

Route tables
- Route is sequence of track elements
- Track elements are in a given state

Geographical
- Route is path through a network of track elements

Common to both families and defined in IS
Interlocking is all about relations

Two tastes of flank protection

- Trivial: signals A, B and points 12a, 12b interlock
- Non-trivial: signals A, B and point 13 interlock

Interlocking relationship

route relation

railML.org
Level Crossing relations

LX is a set of relations between a barrier, signals and train detection.

"close LX 5.7 when train crosses and point 1409 is left"

"close LX 5.7 seven seconds after train crosses and point 1405 is left"
Entry in a route table

Route element
Flank protection

Ordered sequence

Path 1

Path 3
How to model a signal plan

> US-inspired graphics
> Signals are objects
> Attributes are signal aspects, speeds and links to next signal
> Model the signal plan as a linked list

GL → R from 130 to stop

GR → GR@130km/h

V_p / V_a

railML® > October 8th 2014 > Slide 10
26th railML conference Paris
RailML model of a signal plan

railML® shall represent the relations between signals, aspects, speed profile and links between signals.
Power to XPath

Query: where can we go when signal 2 shows yellow?

XPath=//signal[@refId='2']/aspect[@code='GL']
Result=
<signal refId='2'>
  <aspect code='GL', Vp='13', Vz='0'>
    <target refId='4'/>
  </aspect>
</signal>
Conclusions

- IS models topological relationships
- IL models interlocking relationships
- Route tables are best modelled as ordered sequences with required states.
- Interlocking relationships are modelled as associations between elements.
- Searching routes through railML® is trivial.
- Visual representation of railML® IL is needed.