Textual languages with Xtext

Fedor Biryukov

Overview

1. Railway Modeling Language

2. Railway Pattern Language

3. Automatic RAMification*

4. Demo

*RAMify is a shorthand for Reduce, Augment, Modify.

The family of railway formalisms

$$Railway \equiv\; < Segments, Trains, Schedules >$$
$$Segments = Stations \cup Straights \cup Turnouts \cup Junctions$$
$$Station \subset AllowedNames \times Segments \times Segments \times Trains \times Lights$$
$$Lights = \{RED, GREEN\}$$

Figure 1: Formal specification of a railway system model (excerpt)

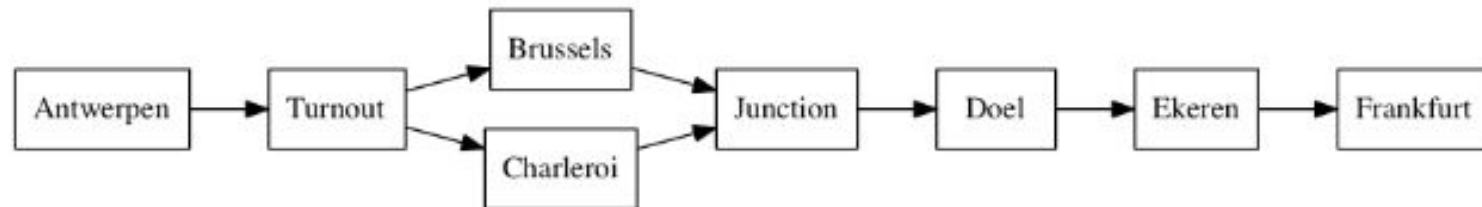https://eclipse.org/Xtext/index.html

Sample RML code

Figure 2: Sample railway system

```
railway RailwaySystem

// Segments
station Antwerp (; fromAtoBC)
turnout fromAtoBC (Antwerp; Brussels, Charleroi)
station Brussels (fromAtoBC; fromBCtoD)
station Charleroi (fromAtoBC; fromBCtoD)
junction fromBCtoD (Brussels, Charleroi; Doel)
station Doel (fromBCtoD; Ekeren)
station Ekeren (Doel; Frankfurt)
station Frankfurt (Ekeren; )

// Trains
train T0001
train T0002
train T0003

// Schedules
schedule S1 T0001 (Antwerp; Doel)
schedule S2 T0002 (Charleroi; Ekeren) STRAIGHT
schedule S3 T0003 (Frankfurt; Antwerp)
```

Sample RPL code

```
train t {
    name:
    segment: a
}
station a {
    name:
    trains: t
    light: 'it==Light.GREEN'
    in:
    out: b
}
station b {
    name:
    trains: t
    light:
    in: a
    out:
}
```

```
rule moveTrains
pre {
    train t {
        name:
        segment: a
    }
    segment a {
        matchSubclasses: true
        name:
        trains: t
        light: 'it==Light.GREEN'
    }
    segment b {
        matchSubclasses: true
        name:
        trains:
        light:
    }
}
post <<<
    if (b.train == null && a.isConnectedTo(b))
        t.moveTo(b)
>>>
```

https://eclipse.org/Xtext/index.html

What I generate

RML.xtext

-> RML.ecore

-> RPL.xtext + templates.xml

https://eclipse.org/Xtext/index.html

Screenshot of https://eclipse.org/Xtext/index.html - January 2016