

Model Differencing

Joeri Reyns

Table of contents

- Introduction
- General approach
- Model representation in AToMPM ^[1]
- Difference calculation
- Difference representation
- Difference visualisation
- Conclusion
- Future work

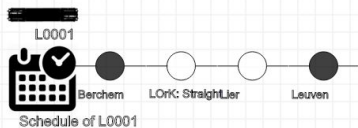
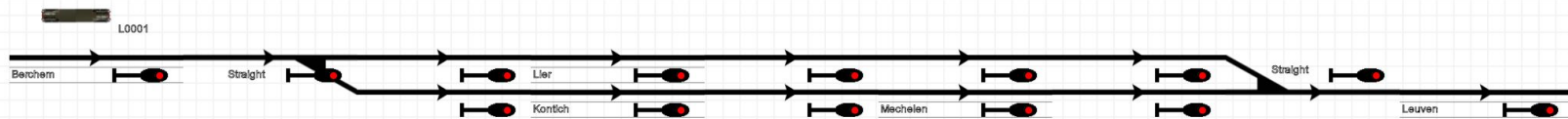
[1] E. Syriani, H. Vangheluwe, R. Mannadiar, C. Hansen, S. Van Mierlo, H. Ergin, Atompm: A web-based modeling environment., in: Demos/Posters/StudentResearch@ MoDELS, 2013, pp. 21-25.



Model Differencing: Introduction

- Important step during any type of development
- Determining model differences can be intrinsically complex
- Search space can be limited by model based optimisation

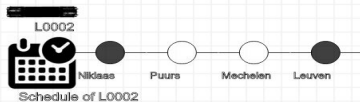
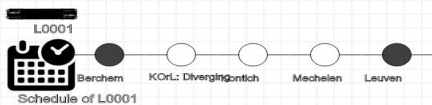
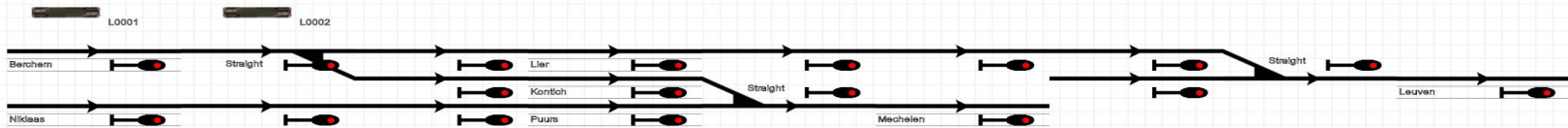
Figure 1: Original Model



General approach

- Calculation
Highly dependent on model representation
- Representation
Directed Delta versus Symmetric Delta
- Visualisation
Show the results in a domain specific way if possible

Figure 2: Final Model



Model representation in AToMPM

Models are represented in JSON format

- Abstract Syntax Model: asm
 - Edges
 - Metamodels
 - Nodes

```
▼ Object {csm: Object, asm: Object} ⓘ  
  ▼ asm: Object  
    ▼ edges: Array[68]  
      ▼ 0: Object  
        dest: "6"  
        src: "0"  
        ► __proto__: Object  
    ▼ Object {csm: Object, asm: Object} ⓘ  
      ▼ asm: Object  
        ► edges: Array[68]  
        ► metamodels: Array[2]  
        ▼ nodes: Object  
          ▼ 0: Object  
            $type: "/Formalisms/TrainSystem/Railway/Station"  
            ► name: Object  
            ► old_pos: Object  
            ► position: Object  
            ► scale: Object  
            ► __proto__: Object  
          ► 1: Object  
          ► 2: Object  
          ► 3: Object  
          ► 4: Object  
          ► 5: Object  
          ► 6: Object  
          ► 7: Object
```

- Concrete Syntax Model: csm
 - Edges
 - Metamodels
 - Nodes

```
▼ Object {csm: Object, asm: Object} ⓘ  
  ▼ asm: Object  
    ► edges: Array[68]  
    ► metamodels: Array[2]  
    ► nodes: Object  
    ► __proto__: Object  
  ▼ csm: Object  
    ► edges: Array[0]  
    ► metamodels: Array[2]  
    ▼ nodes: Object  
      ▼ 0: Object  
        ► $asuri: Object  
        ► $contents: Object  
        $type: "/Formalisms/TrainSystem/Railway.defaultIcons/StationIcon"  
        ► mapper: Object  
        ► orientation: Object  
        ► parser: Object  
        ► position: Object  
        ► scale: Object  
        ► typename: Object  
        ► __proto__: Object  
      ► 1: Object  
      ► 2: Object
```

Difference calculation

Optimisations based on model representation

- ID based matching
 - Unchanged elements are easy to find
- AToMPM doesn't fill in ID gaps
 - Additions and Deletions are easy to find
- All others should be modifications

Difficulties remain

- How to find moved elements
 - Not implemented but could be done by hashing all nodes and checking for partial matches
- How to handle type changes in models
 - Compare ramified models with their original models.

All found information is stored in the second model

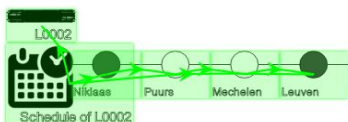
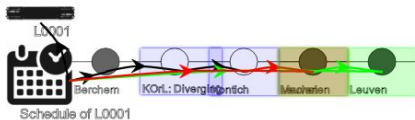
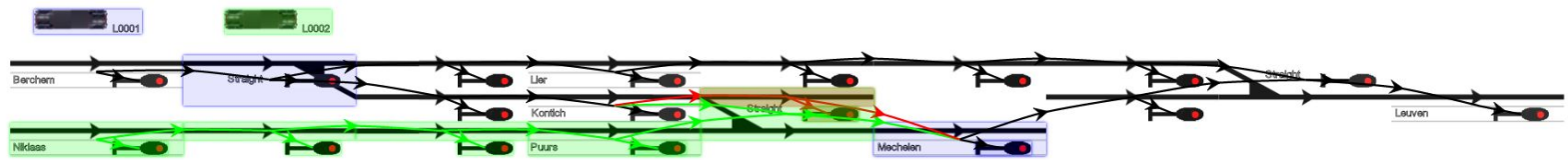
- Second model used as difference model

Difference representation

Difference model is changed into a difference graph
= symmetric delta approach

- Nodes which represent edges are:
 - Stored in a list of edges
 - Augmented with their source and destination information
- “Coloured” graph is constructed
 - Nodes = Nodes that weren't edges
 - Edges = The list of edges generated in previous step

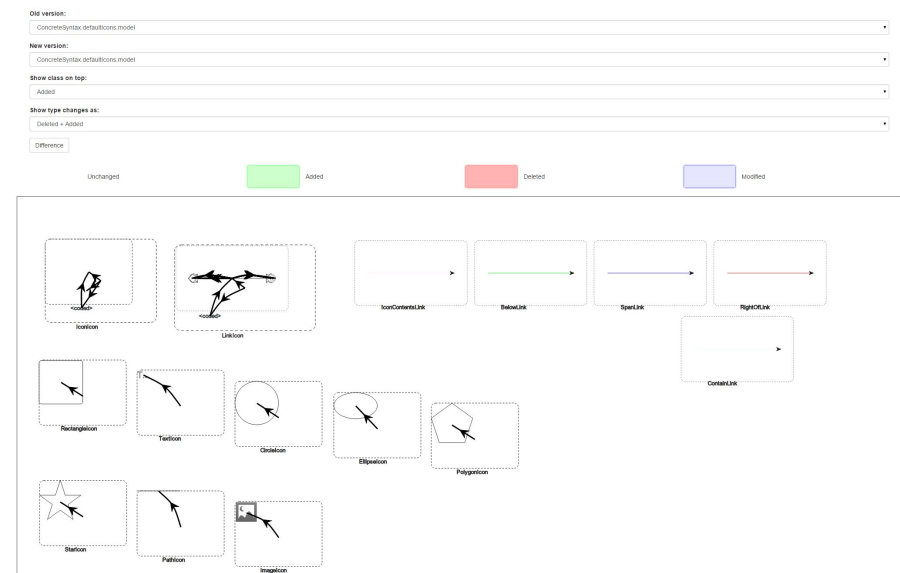
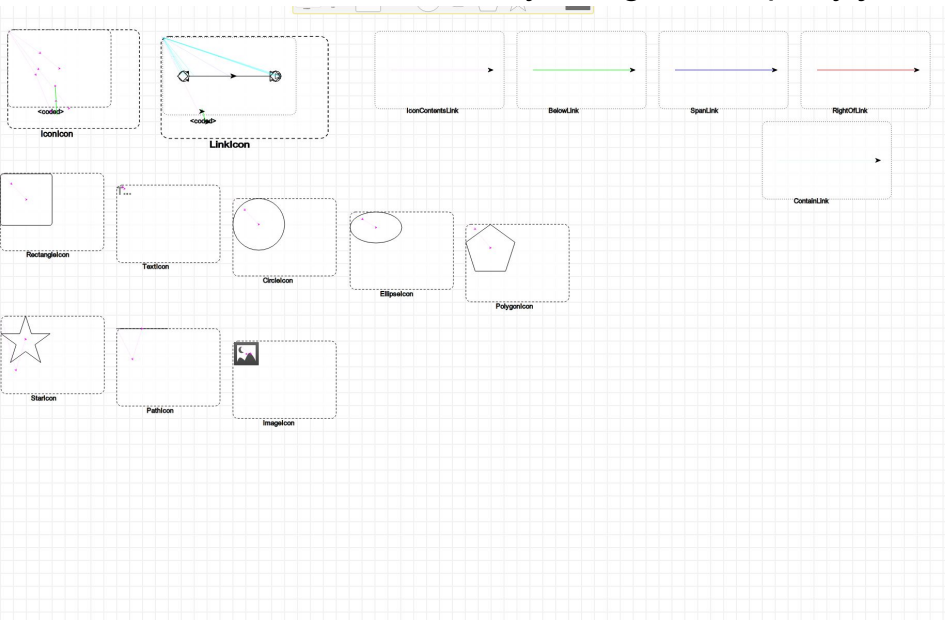
Figure 3: Difference graph



Difference Visualisation

Different technique used on nodes and edges

- Nodes are visualised in the same way as in AToMPM:
 - Content information stored in “\$content” attribute of csm
 - Visualised by using a third party javascript library named Raphaël [1]
- Edges are visualised in a more generic way for making sure all information is easily findable
 - Visualised by using a third party javascript library named jsPlumb [2]



[1] D. Baranovskiy, Raphaël JavaScript Library, [Online;<http://raphaeljs.com/>].

[2] I. jsPlumb, jsPlumb Build Connectivity Fast, [Online;<https://jsplumbtoolkit.com/>].

Conclusion

- General approach
 - Calculation step is representational dependent
- Cicchetti et Al.^[1] suggests:
 - meta-model based differencing leads to
 - Having the right expressiveness as needed by the domain
 - More intuitive for people within the domain
- AToMPM

Meta-modelling tools like AToMPM provide everything you need to build a meta-model based differencing tool which has enough expressiveness and can be more intuitive.

[1] Antonio Cicchetti, Davide Di Ruscio, Alfonso Pierantonio : A Metamodel Independent Approach to Difference Representation



Future work

- Use a less naive differencing algorithm
- Include the possibility for a directed delta approach
- Include the option to draw edges as they are visualised in AToMPM

Questions?

