

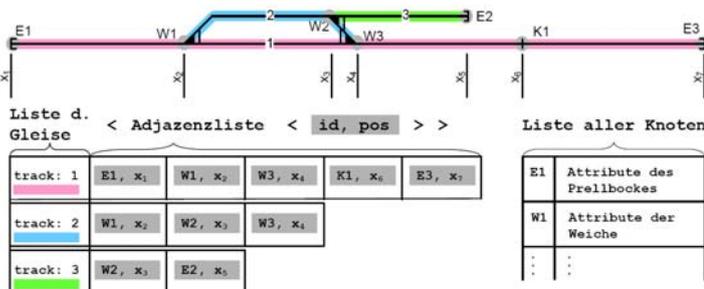
## Entwicklung einer Komponente für das Einlesen einer Streckennetz-Projektierung im Standardformat RailML

Im Rahmen der Diplomarbeit wurden eine Komponente zum Einlesen einer Streckennetz-Projektierung im Standardformat RailML für das Projekt Radio Block Center der Thales Rail Signalling Solutions GmbH.

Es wird ein Konzept entworfen, welches die Datentransformation beinhaltet, um die Erzeugung eines Graphen durch die RBC-Software zu ermöglichen. Die gewonnenen Erkenntnisse der Analyse fließen dabei in den Entwurf mit ein. Aufbauend auf dem ausgearbeiteten Konzept wird die Komponente als C++ Modul implementiert, welches den für das RBC geltenden Sicherheitsanforderung des SIL 4 genügen muss.

### Entwicklung des Datenmodells

Das für den Infrastructure Reader entwickelte Datenmodell, nimmt die Daten einer eingelesenen RailML-Streckentopologie auf. Jedes Gleis wird durch eine Adjazenzliste dargestellt. Eine Adjazenzliste enthält somit alle topologischen Knoten eines Gleises. Die Knoten sind entsprechend der Kilometrierung in der Liste sortiert. Die sortierten Adjazenzlisten werden in der Liste der Gleise gehalten.



### Entwurf der Komponente

Die Komponente muss die drei Hauptaufgaben bewältigt:

1. Das Einlesen von RailML-Streckendaten.
2. Die Überprüfen der gelesenen Daten im Bezug auf die Schemakonformität, die Vollständigkeit der Daten, die Einhaltung von Wertebereichen und Projektregeln.
3. Die Verarbeitung der Daten bzgl. der Generierung eines vollständigen Graphen.



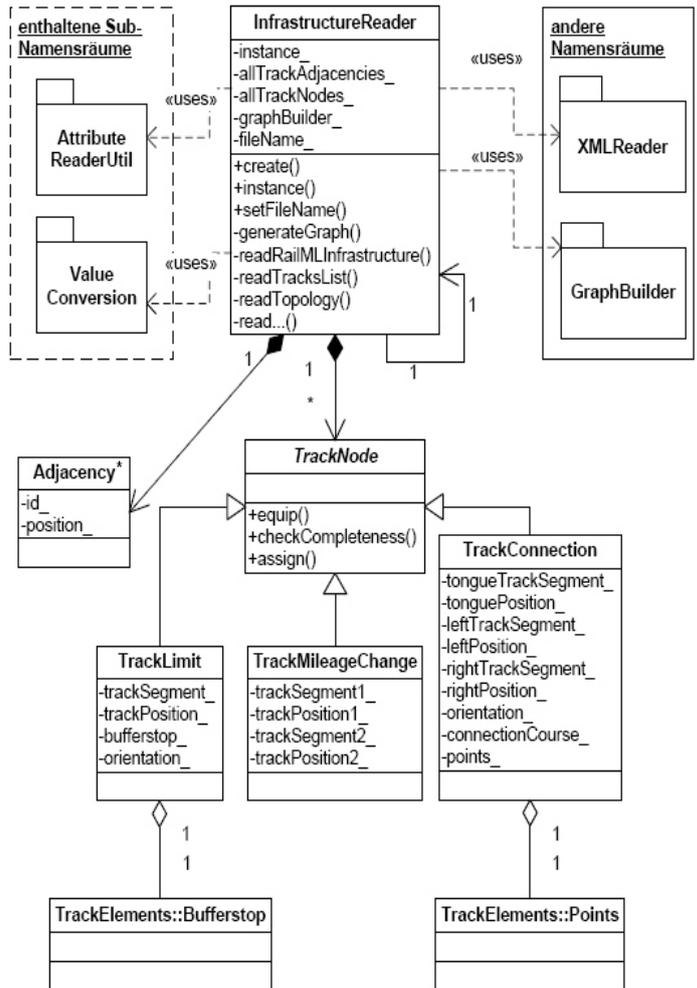
Student: Martin Lehmann

Betreuer: Dr.-Ing. T. Albrecht (TU Dresden)  
Dipl.-Ing. H. Jost (Thales RSS GmbH)

Kontakt: martin.lehmann3@mailbox.tu-dresden.de

### Implementierung

Der Namensraum InfrastructureReader enthält alle Klassen der implementierten Komponenten. Ein großer Teil des UML-Klassendiagramms betrifft das Datenmodell, in dem die gelesenen Daten zwischengespeichert werden und dessen Objekte in STL-Containern gehalten werden.



### Software-Prüfung

Bei der Entwicklung von sicherheitsrelevanten Anwendungen sind die Anforderungen an deren Prüfung besonders hoch, da hier Softwarefehler auf keinen Fall auftreten dürfen. Um die Fehlerfreiheit für die Komponente in einem hohen Maße zu gewährleisten, kamen während der Diplomarbeit die Testverfahren Code-Reviews und Unit-Test zum Einsatz.