Connecting IT and Transport

Modellbasierte Entwurfsmethoden für die Entwicklung einer neuen Generation SPS-basierter elektronischer Stellwerke

Elferik Schultz

Kiel, 11. Mai 2011

MENGES
KoSSE

funkwerk
information technologies

Das MENGES-Projekt
– Motivation und Ausrichtung des Vortrags

M	odellbasierte
E	ntwurfsmethoden für eine
N	eue
G	eneration
E	lektronischer
S	tellwerke

- Ausgangssituation, Chancen und Ziele des MENGES-Projekts
- Vorstellung der Domain Specific Language (DSL) und Umsetzung in den Sprachpartitionen
- Résumée und Ausblick

MENGES
KoSSE

funkwerk
information technologies

Die Chancen des modellbasierten Entwurfs in der Leit- und Sicherungstechnik

Ausgangssituation und Rahmenbedingungen

- hoher Rationalisierungsbedarf seitens DB Netz AG bei mechanischen Stellwerken / Relaisstellwerken
- neues Lastenheft ESTW-R: Reduzierung der Funktionalität bei gleicher Sicherheit
- Speicherprogrammierbare Steuerungen (SPS) als neue Trägertechnik (kompakte und modulare Standardprodukte mit langer Lebenszeit, geringe LCC, niedriger Stromverbrauch, vorhersagbares Zeit- und Schaltverhalten)
- modellgetriebener generativer Ansatz als Chance für Kostensenkungen und Effizienzsteigerungen in der Signaltechnik
- Kompetenzbündelung im Kieler Raum (Universität und Wirtschaft), Katalysatorfunktion der Förderprojekte

MENGES
KoSSE

funkwerk
information technologies

Erfahrungen und Hemmnisse im Bereich der traditionellen ESTW-Entwicklung ohne MDSE

- eingesetzte Methoden und Werkzeuge nur schwach integriert (Bsp. Zustandsdiagramme, Klassen, Wahrheitstabellen, Fallunterscheidungen)
- signifikanter Überprüfungs- und Korrekturaufwand für produzierte Artefakte, da es kein integriertes Modell gibt
- Brüche in der Entwicklungskette, so dass Umsetzung in die nächste Ebene zumeist manuell erfolgt
- „teure“ Fehler, da aufgedeckte Unzulänglichkeiten im Prüfling manuelle Korrekturen und Überprüfungen der gesamten Entwicklungskette bedingen (Anspruch First-time-right)
- Dokumentation nicht geschlossen aus Entwicklungs-artefakten produzierbar

→ **Erste positive Erfahrungen zum MDSE im Bereich der Verschub- und Rangierstellwerke 2008**

MENGES
KoSSE

funkwerk
information technologies

Die Ziele des MENGES-Projekts

Entwicklung einer domänenspezifischen Sprache und MDSE für SPS-basierte elektronische Stellwerke

- Automatisierung der Konstruktionsprozesse, Beseitigung von Brüchen in der Entwicklungskette
- hohe Qualtität und Sicherheit bei gleichzeitiger Aufwands- und Kostenreduktion
- frühzeitiges Testen und Erkennen von Fehlern auf der Modellebene (Veringerung von Round-Trip-Zeiten)
- Zusammenführung bekannter Modellierungsmethoden durch Erweiterung und Restriktion innerhalb der DSL (Zustandsmaschinen, Klassen, Regel- und Ablaufsprachelemente)
- offener Ansatz durch Transformation von Modellcode in auf Programmiersprachen einer Standardplattformen (PLCopen im Bereich SPS)

MENGES
KoSSE

funkwerk
information technologies

Wo setzt der modellbasierte Entwurf in der LST an?

Modellbasierter Entwurf

- abstrakte Repräsentation von Wissen (Gleisgraph, Fahrwegelemente, funktionales Verhalten der Stw-Elemente)
- unmittelbarer Bezug zur Anwendungsdomäne LST
- Identifikation von Partitionen und Design zugehöriger Sprachkonstrukte (DSL), in denen bestimmte Muster durchgängig und wiederkehrend auftreten

Transformation auf Zielplattform SPS

- direkte Abbildung der DSL auf SPS-Sprachen und -Plattformen (unter Einbezug von Sprachstandards, wie z. B. PLCopen XML)
- Erkennen von Entwicklungsmustern, Wiederverwendung von Modellen, Nutzung von Frameworks (Xtext, CoDeSys)
- erwarteter Automatisierungsgrad bei Modell-Transformationen ca. 90 %

→**Verstärkung über forcierte wissenschaftliche Kooperation und Zusammenarbeit mit regionalen Partnern**

MENGES
KoSSE

funkwerk
information technologies

Die MENGES-Projektpartner als Kompetenzverbund von Wirtschaft und Wissenschaft

- **AG Softwareengineering, CAU Kiel**
  - **Prof. Wilhelm Hasselbring, Reiner Jung**

- **AG Echtzeitsysteme und Eingebettete Systeme, CAU Kiel**
  - **Prof. Reinhard von Hanxleden, Christian Schneider**

- **Funkwerk Information Technologies GmbH, Kiel**
  - **Holger Neustock, Elferik Schultz, Heiko Schaefer**

- **b+m Informatik AG, Melsdorf**
  - **Dr. Wolfgang Goerigk, Thomas Stahl, Steffen Weik**

Vorstellung der Domain Specific Language

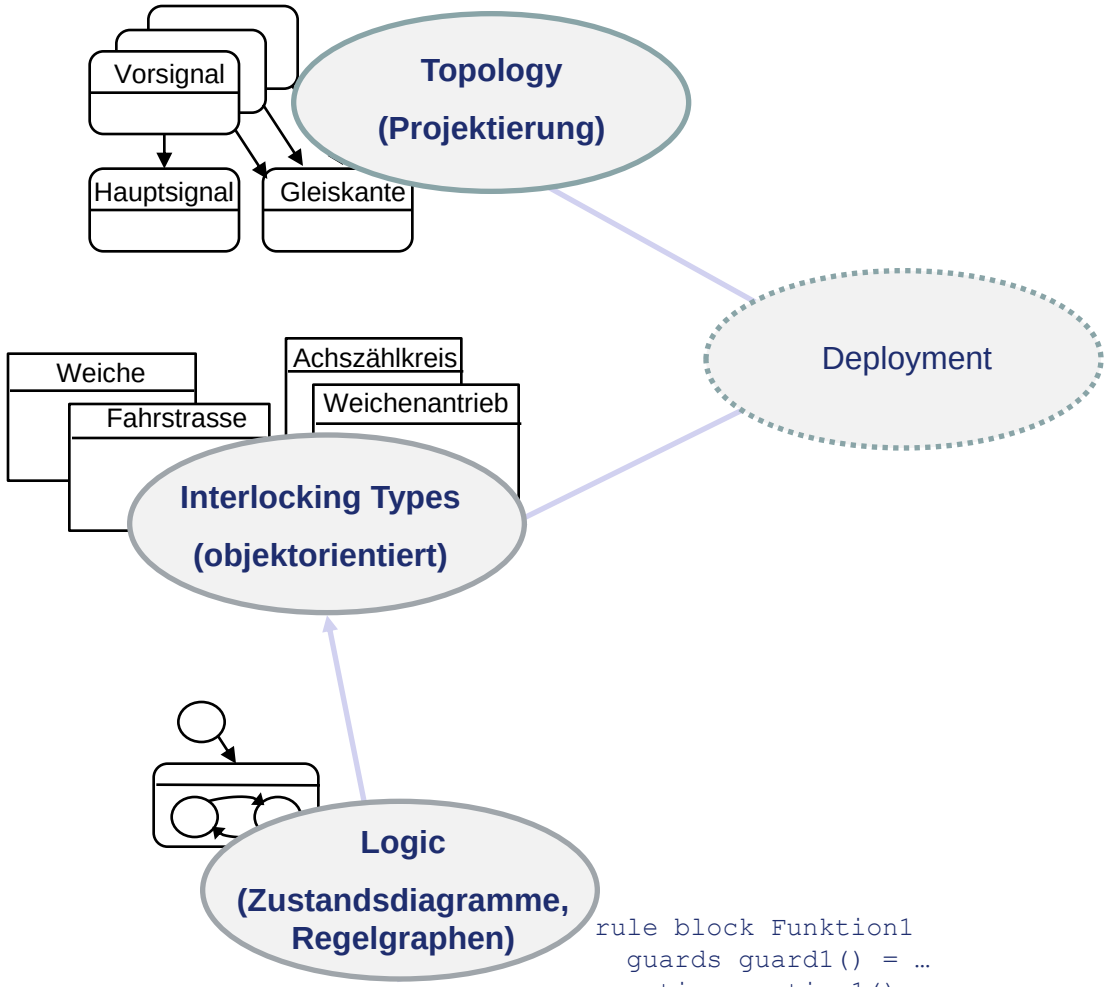
funkwerk
information technologies

Die MENGES-Partitionen als Mittel der Strukturierung der Anwendungsdomäne

Instanzen

Typen

Semantik



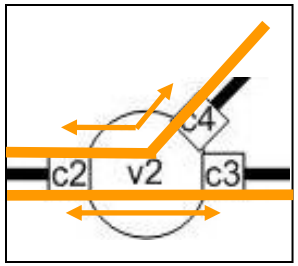
Aufbau der Topologie-Partition zur Erfassung der zugrundeliegenden Infrastrukturelemente

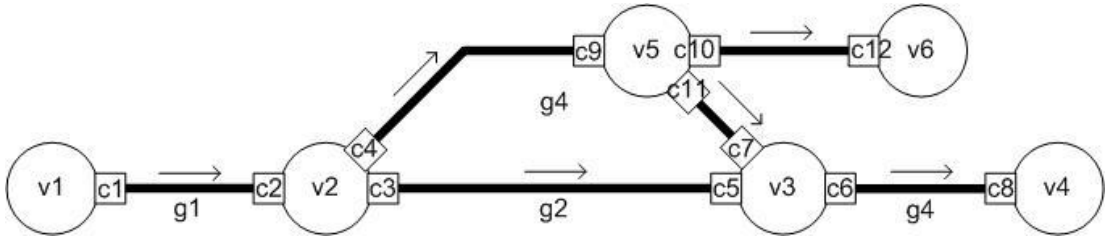
```
package estwr.lindaunis.topology

graph:
          // Knoten-Kantenmodell des Gleisgraphen

topography:
          // Punktförmige Objekte (Gleispunkte) auf Gleiskanten

area:
          // Bereiche, beschrieben durch Gleispunkten und
          // Gleisknoten

control:
          // Formulierung von Stellbedingungen durch Bereiche,
          // Gleispunkte und Gleisknoten (Fahrstrassen)
```

MENGES
KoSSE

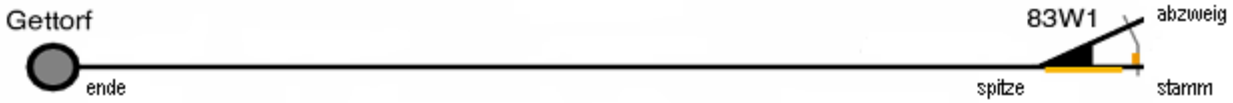
funkwerk
information technologies

Der Gleisgraph als erweitertes Knoten-Kanten-Modell






```
graph:
        einbruchsknoten Gettorf
        weichenknoten 83W1 abzweigend links
                connect spitze with Gettorf.ende
        [...]
```

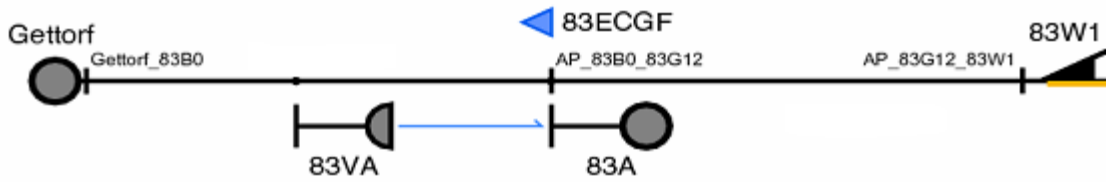


MENGES KoSSE

funkwerk information technologies

Verortung von punktförmigen Elementen auf dem Gleisgraphen

```
topography:
        edge from Gettorf.ende {
                achszaehlpunkt Gettorf_83B0
                ks-vorsignal 83VA for 83A
                achszaehlpunkt AP_83B0_83G12
                fzpunkt 83ECGF wirkrichtung umgekehrt
                ks-hauptsignal 83A
                achszaehlpunkt AP_83G12_83W1
        }
        […]
```

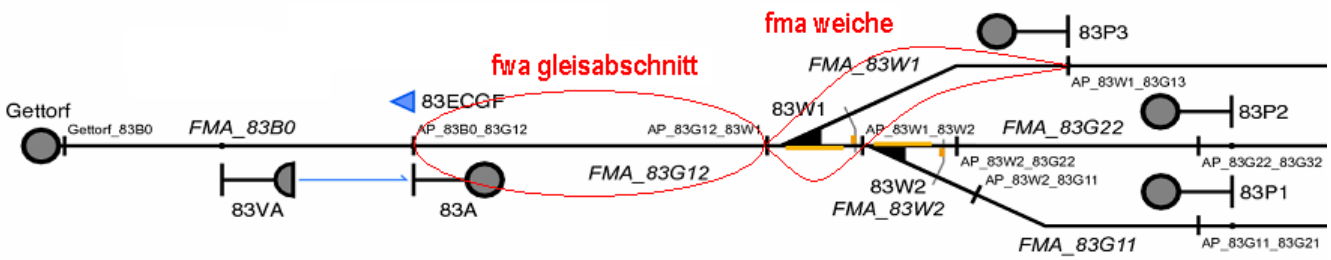


MENGES KoSSE

funkwerk information technologies

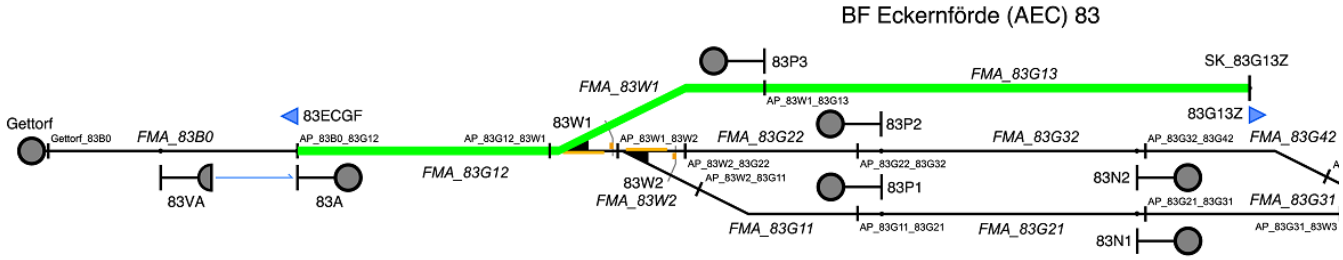
Aufbau von Bereichen über bereits definierte Teilstrukturen

```
area:
        fma weiche FMA_83W1 bounded by AP_83G12_83W1,
                AP_83W1_83G13, AP_83W1_83W2 with 83W1
        fma gleisabschnitt FMA_83G12 bounded by AP_83B0_83G12,
                AP_83G12_83W1
        bedienzentrale BEDZ_AEC
        lokalstellwerk AEC with
                bedienzentralen BEDZ_AEC
                weichen FMA_83W1, FMA_83W2, […]
                gleisabschnitte FMA_83B0, FMA_83B1, […]
                fahrstrassen 83A_83G13Z, 83A_83N1, […]
        […]
```



Stellbedingungen als Beziehungen zwischen Fahrwegelementen und Fahrstraßen

```
control:
        zugstrasse einfahrt 83A_83G13Z from 83A to 83G13Z
                fahrweg FMA_83G12,FMA_83W1,FMA_83G13
                        with 83W1.lage=links
                flankenschutz
                        for 83W1 by 83P1 83P2
        […]
```



Elemente der Partition der Stellwerkelementtypen (Interlocking Types)

```
package estwr.lindaunis.stellwerkelementtypen


// Zustandsmengen (state sets)
// Aufzählungstypen (enumeration types)
// Kommandosätze (command sets)
// Kommunikationseinheiten (communication types)
// Stellwerkelemente (interlocking elements, interfaces)
```

MENGES
KoSSE

funkwerk
information technologies

Beispielstruktur für Weiche (generische Applikation)

```
interlocking element GaWeiche extends Besetztelement implements
Bedienbar, Beanspruchbar
{
        statevars
                auffahrung: Auffahrung;
                lage: Weichenlage;
        properties
                boolean endlageUeberwachung;
                boolean verschlossen;
        predicates
                istAufgefahren() = auffahrung != nein
                istStellbar() =
                        !istAufgefahren()
                        && zustandsdatenVorhanden
                        && !verschlossen;
}
```

Beispielstruktur einer Fahrstraße

```
interlocking element Fahrstrasse implements Bedienbar {
    statevars
        status: (
                grundzustand, zulassungspruefung,
                reserviert, beansprucht, ueberwachung
        );
    properties
        list<Beanspruchbar> fahrweg;
        list<Lagebedingung> fahrwegBedingungen;
    predicates
        hatSolllage(GaWeiche weiche) =
            exists (bedingung in fahrwegBedingungen) {
                bedingung.weiche == weiche
                && bedingung.lage == weiche.lage
            };
}
```

MENGES KoSSE

funkwerk information technologies

Aufbau der Logik-Partition

```
package estwr.lindaunis.logic

// Beschreibung von Zustandsübergängen (state machines)
// Regelblöcke (rule blocks)
```

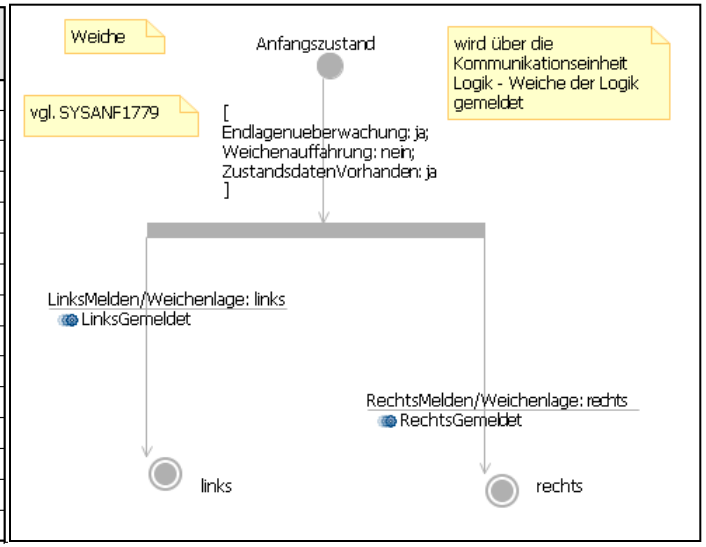
MENGES
KoSSE

funkwerk
information technologies

Regelblöcke als Verhaltensbeschreibung für Wenn-Dann-Bedingungen (Bsp. Lage-Meldung)

```
rule block lageMelden references GaWeiche () {
  rule graph
  --> [zustandsdatenVorhanden && endlageUeberwachung &&
         !istAufgefahren()] {
        --> [antriebMeldung.li] { action lage = links; }
        --> [antriebMeldung.re] { action lage = rechts; }
    }
}
```

Information		Wert für erlaubten Startzustand	für Zustandsübergangsbeschreibung relevante Werte	Bemerkung
a)	Weichenlage	egal	re; li	
b)	Überwachung der Endlage	ja		
c)	Weichenauffahrung	nein		
d)	Freimeldung	egal		
e)	Freimeldung Profilraum	egal		
f)	Umstellsperre	egal		
g)	Sperrung der Weichenlaufkette	egal		
h)	Befahrbarkeitssperre	egal		
i)	Solllage vorhanden	egal		
j)	Verschluss	egal		
k)	Folgeabhängigkeit wirksam	egal		
l)	Reservierung	egal		
m)	Beanspruchung	egal		
n)	Flankenschutz gesichert	egal		
o)	aktuelle Zustandsdaten	ja		

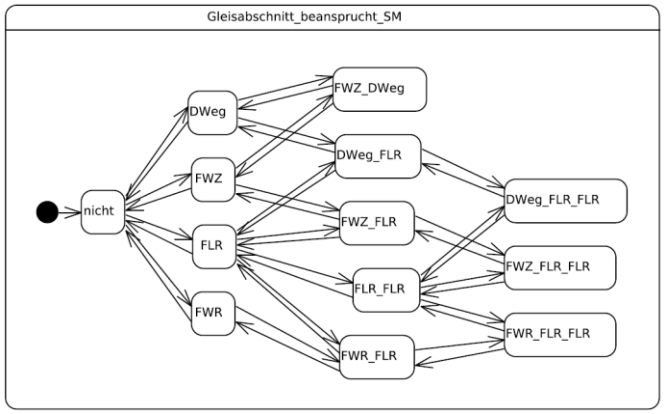


Résumée und Ausblick

funkwerk
information technologies

Nutzung neuer Sprachen und Modelle für eine zielgerichtete und optimierte Entwicklung

- Modelltransformation für durchgängige und effiziente Entwicklung (Codegenerierung mit hohem Automatisierungsgrad)
- Verbesserung der Kommunikation mit den Domänenexperten durch Verwendung intuitiver und domänenspezifischer Sprachelemente
- Erkennung und Abstraktion von Entwicklungsmustern mittels neuer Darstellungs- und Eingabeformen (paralleles Arbeiten in Text und Grafik)
- stärkere Wiederverwendung von Ausdrücken und Strukturen (Vererbung, Interfaces, Aspektorientierung, Bildung und Zusammenfassung von aussagenlogischen und prädikatenlogischen Ausdrücken)



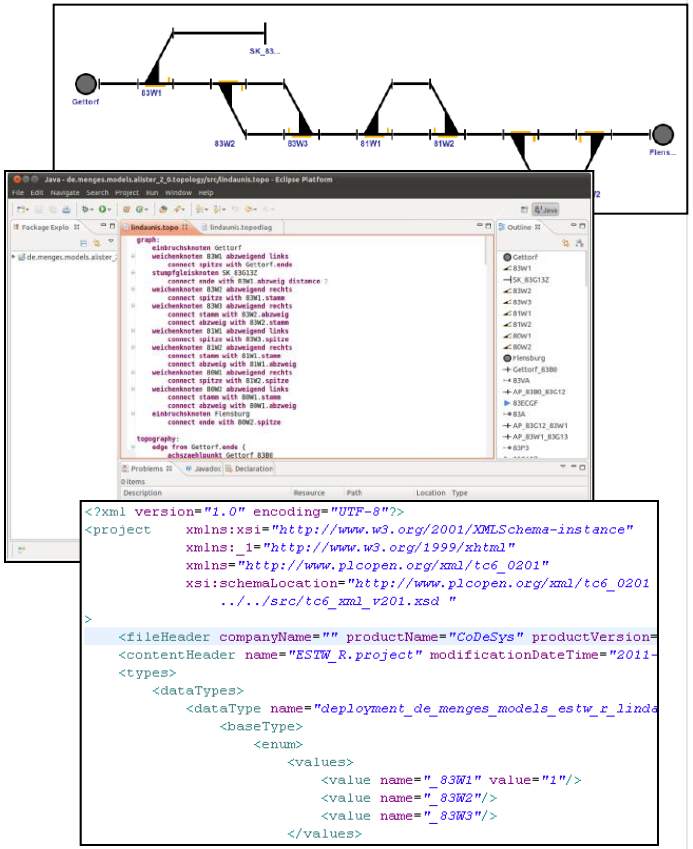

```
predicates
  hatSolllage(GaWeiche weiche) =
    exists (bedingung in fahrwegBedingungen) {
        bedingung.weiche == weiche
      && bedingung.lage == weiche.lage
```

MENGES
KoSSE

funkwerk
information technologies

Entwicklung einer speziell für die ESTW-Domäne optimierten Werkzeugfamilie

- Verwendung moderner und offener Entwicklungsplattformen anstelle proprietärer Herstellertools (Eclipse Modeling Project, Xtext)
- Integration neuester wissenschaftlicher Ansätze (z. B. KIELER-Framework für Layout und Pragmatik)
- Wahl marktgängiger und breit unterstützter Zielplattformen (CoDeSys, PLCopen) als Laborumgebung
- lückenlose Entwicklung von einer formalen Spezifikation bis zum generierten Quellcode der Zielplattform

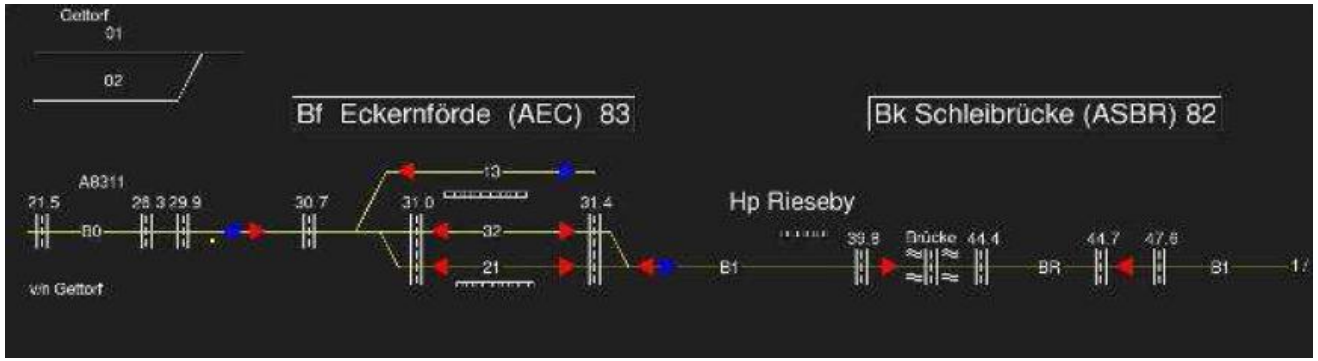


MENGES
KoSSE

funkwerk
information technologies

Die Frage nach dem Benefit für den Kunden

- dramatische Reduzierung der Entwicklungszeiten infolge des Rationalisierungsschubs in ESTW-Projekten, die MDSE nutzen
- klare Planungsaussagen und stabile Terminketten
- Transparenz und Eindeutigkeit durch frühzeitige Formalisierung der Spezifikation
- insgesamt deutliche Einsparungen bei Entwicklungsaufwänden und Qualitätssicherung
- Verbesserung der Kostensituation bei Anbietern und nachhaltige Senkung der Beschaffungskosten auf Seiten der Infrastrukturbetreiber



MENGES
KoSSE

funkwerk
information technologies

Perspektiven für die Zukunft

- Schaffung von Schnittstellen zur Projektierung (automatische Integration von Beständen / Planungen aus Planungssystemen in die ESTW-Entwicklung)
- automatische Generierung von Testfällen auf Basis der MENGES-Spezifikation (inkl. Fault Injection)
- Simulation und Analyse auf Modellebene
- Verbesserung der Mensch-Maschine-Schnittstelle in der Entwicklung
- Instrumentierung / Profiling und Rückfluss der Ergebnisse in die Spezifikation und Modellierung

MENGES
KoSSE

funkwerk
information technologies

Vielen Dank für Ihre Aufmerksamkeit!

MENGES
KoSSE

funkwerk
information technologies