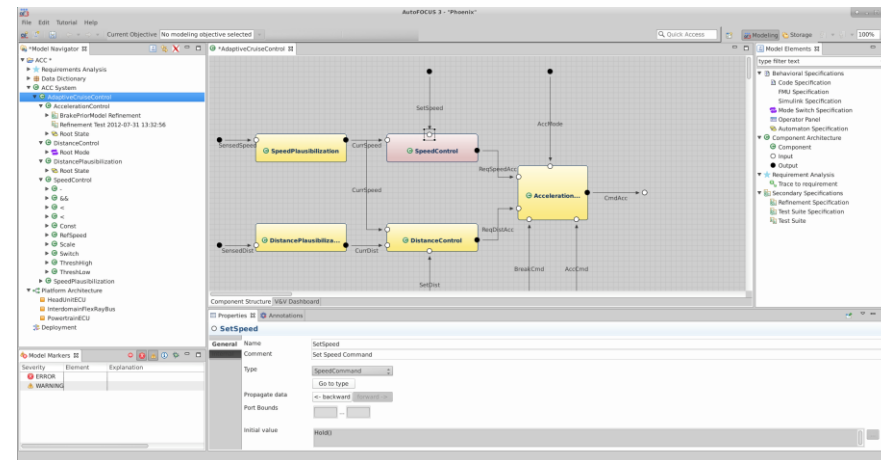
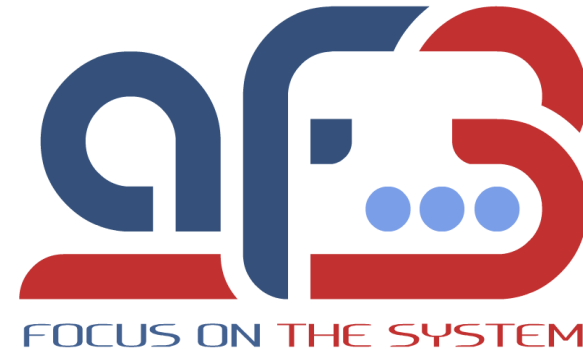




Tutorialmodus in AutoFOCUS3 & Integration einer SPES Technologie in die SPEDiT Toolkette

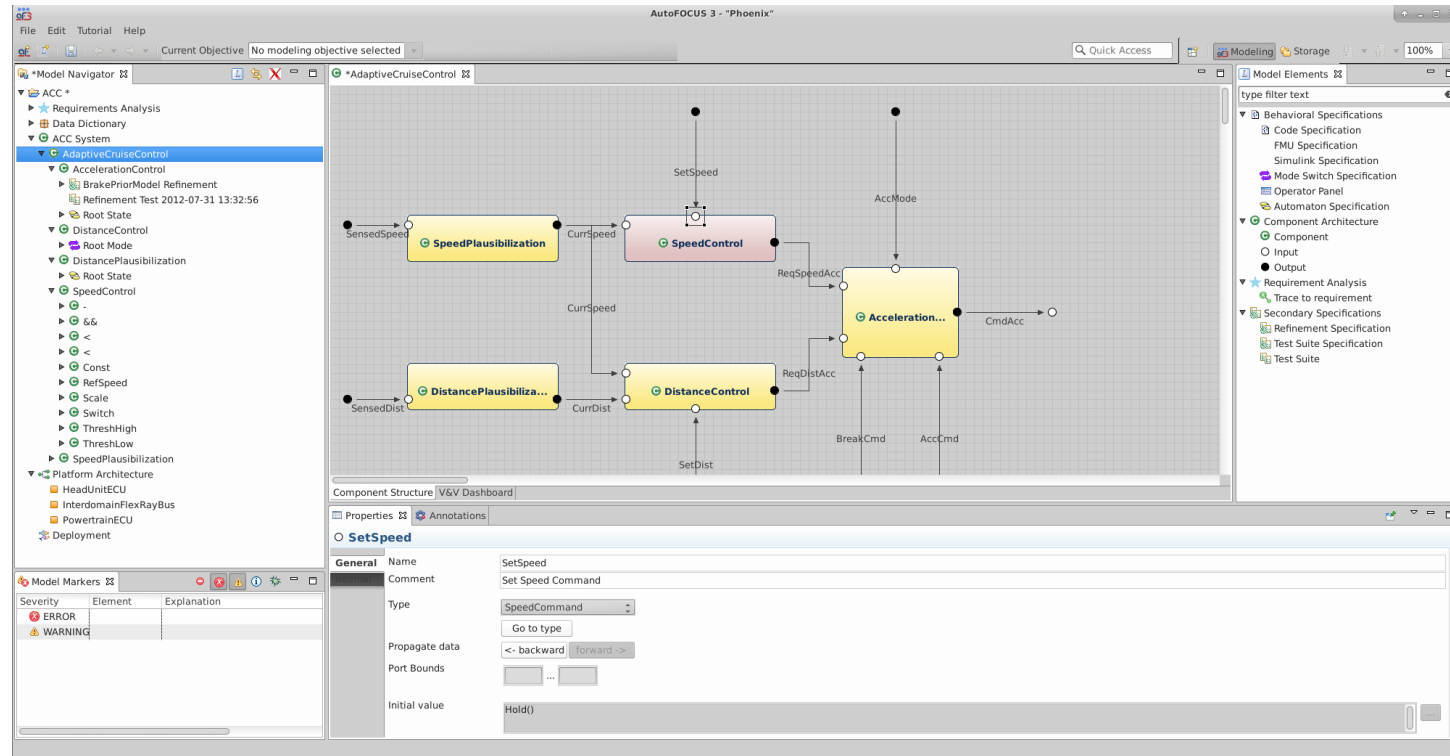
J. Eder, S. Voss (fortiss)

- Tool zur modelbasierten Entwicklung von verteilten, reaktiven, eingebetteten Systemen
- Modelle in allen Entwicklungsphasen z.B.
 - Anforderungs-Analyse,
 - Design logischer Architekturen,
 - Design technischer Architekturen
- Methoden zur formalen Analyse und Synthese von Modellen

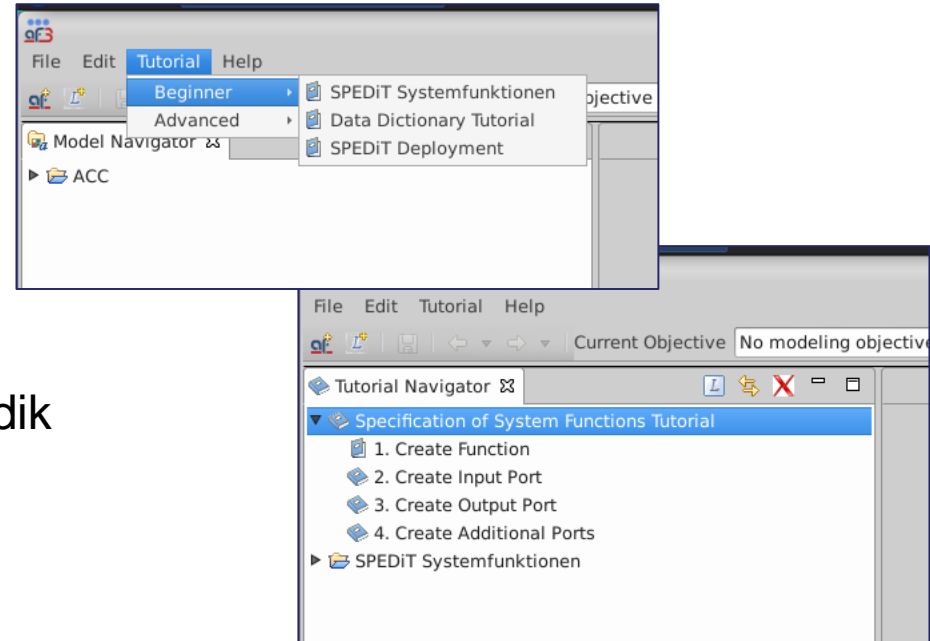


Tutorialmodus AF3 – Warum?

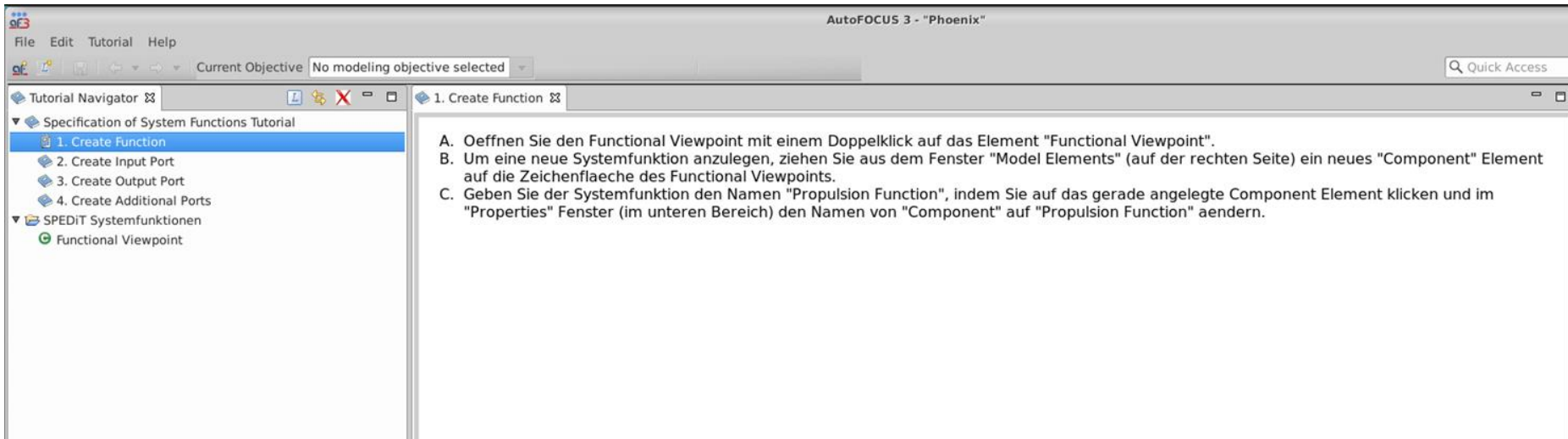
- Orientierung an Problemen die ein Nutzer bei anfänglicher Benutzung eines Tools hat



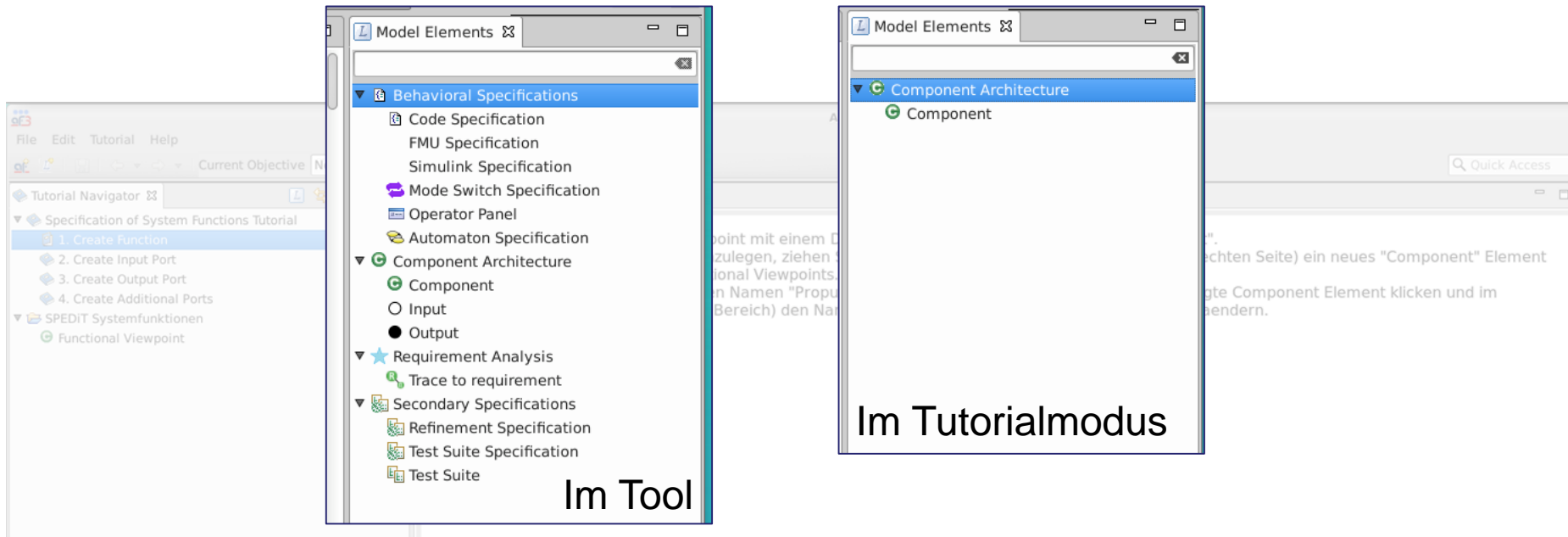
- Orientierung an Problemen die ein Nutzer bei anfänglicher Benutzung eines Tools hat
- Sehr starke Einschränkung der Benutzungsmöglichkeiten (z.B. Reduzierung der angebotenen Modellierungselemente)
- Fokus auf die zu erlernende Methodik
- Führen des Nutzers durch die einzelnen Schritte
- Entwicklung eines eigenen Modellierungsmodus



- Erklärung jedes Modellierungsschrittes

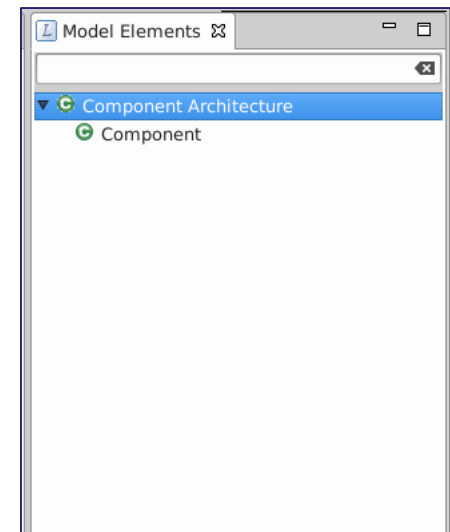
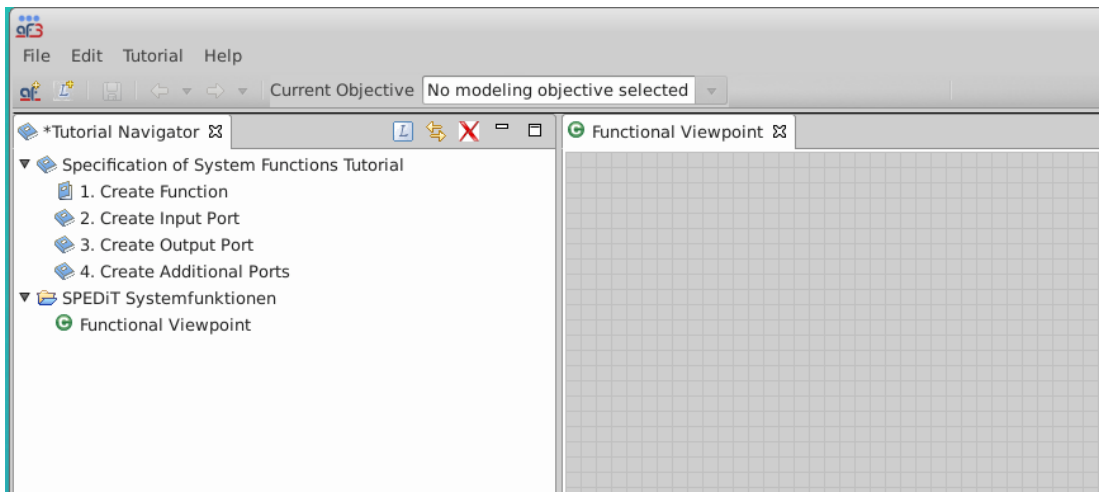
The screenshot shows the AutoFOCUS 3 software interface. The title bar reads "AutoFOCUS 3 - 'Phoenix'". The menu bar includes "File", "Edit", "Tutorial", and "Help". Below the menu bar is a toolbar with icons for file operations and a "Current Objective" dropdown menu showing "No modeling objective selected". On the right side of the toolbar is a "Quick Access" search box. The main workspace is divided into two panes. The left pane is the "Tutorial Navigator", which shows a tree view under "Specification of System Functions Tutorial". The first item, "1. Create Function", is selected and highlighted in blue. Below it are "2. Create Input Port", "3. Create Output Port", and "4. Create Additional Ports". Underneath these is a section for "SPEDiT Systemfunktionen" with a sub-item "Functional Viewpoint". The right pane is titled "1. Create Function" and contains three numbered instructions: "A. Oeffnen Sie den Functional Viewpoint mit einem Doppelklick auf das Element 'Functional Viewpoint'."; "B. Um eine neue Systemfunktion anzulegen, ziehen Sie aus dem Fenster 'Model Elements' (auf der rechten Seite) ein neues 'Component' Element auf die Zeichenflaeche des Functional Viewpoints."; and "C. Geben Sie der Systemfunktion den Namen 'Propulsion Function', indem Sie auf das gerade angelegte Component Element klicken und im 'Properties' Fenster (im unteren Bereich) den Namen von 'Component' auf 'Propulsion Function' aendern."

- Erklärung jedes Modellierungsschrittes
- Einschränkung der Funktionalität und der Modellierungselemente individuell in jedem einzelnen Modellierungsschritt



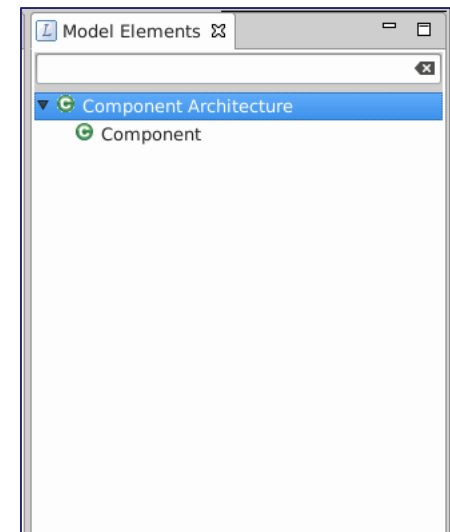
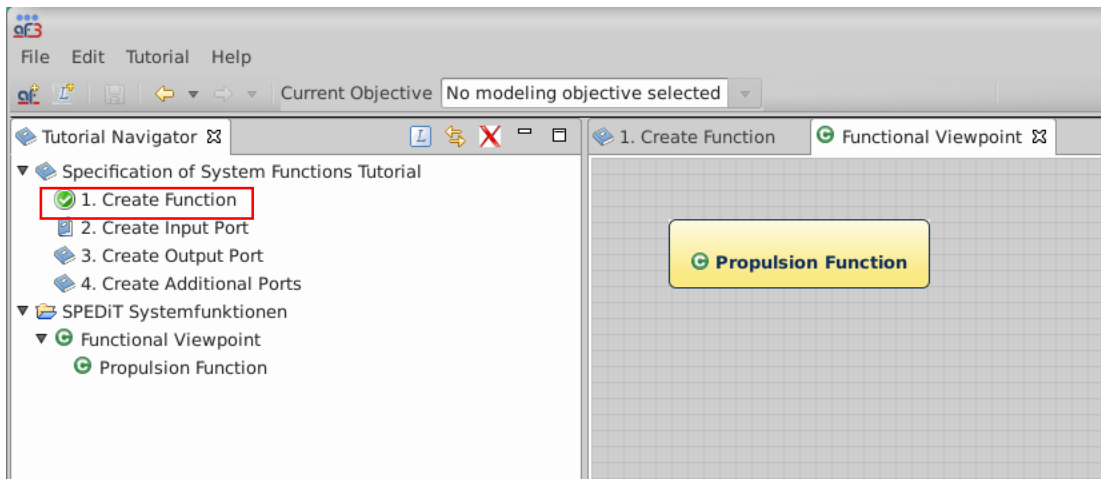
1. Create Function

- A. Öffnen Sie den Functional Viewpoint mit einem Doppelklick auf das Element "Functional Viewpoint".
- B. Um eine neue Systemfunktion anzulegen, ziehen Sie aus dem Fenster "Model Elements" (auf der rechten Seite) ein neues "Component" Element auf die Zeichenfläche des Functional Viewpoints.
- C. Geben Sie der Systemfunktion den Namen "Propulsion Function", indem Sie auf das gerade angelegte Component Element klicken und im "Properties" Fenster (im unteren Bereich) den Namen von "Component" auf "Propulsion Function" ändern.

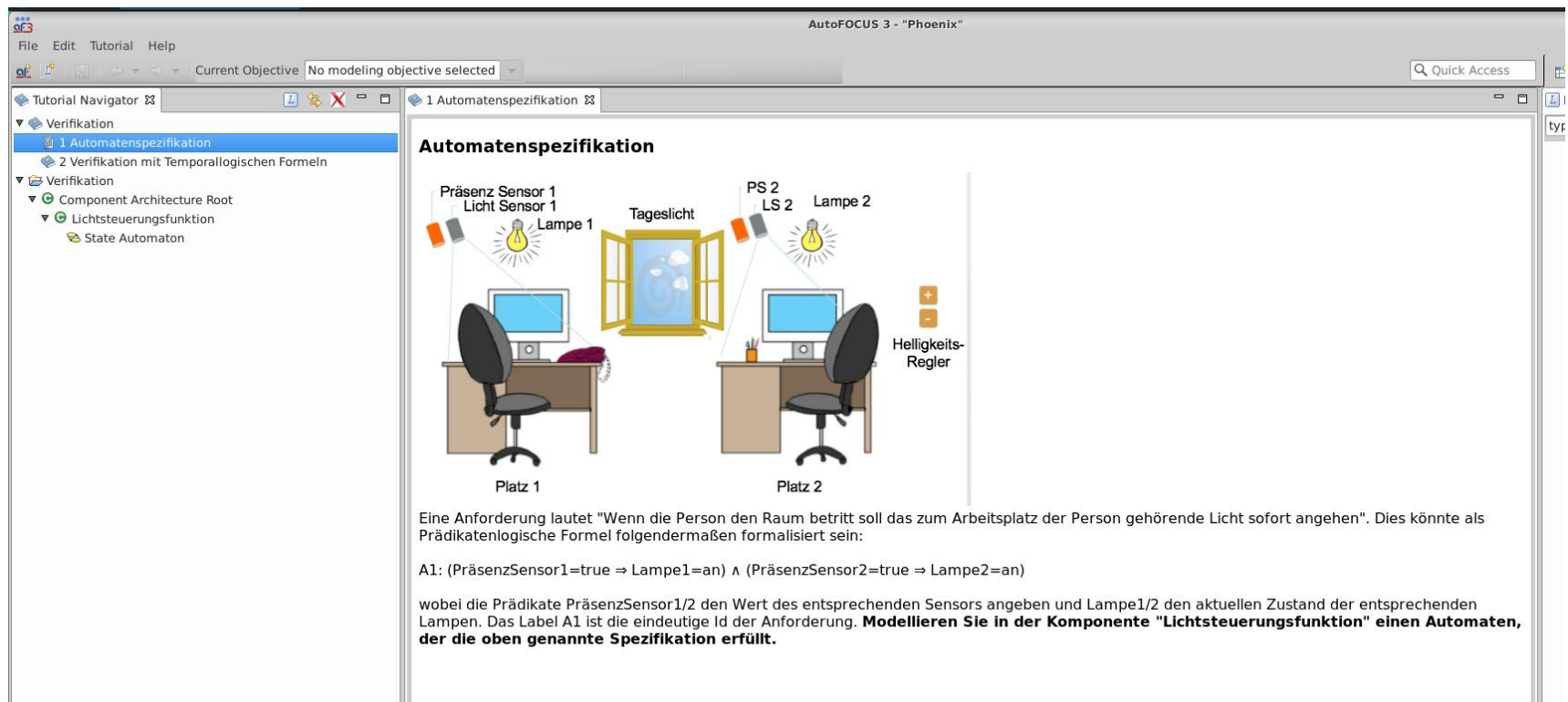
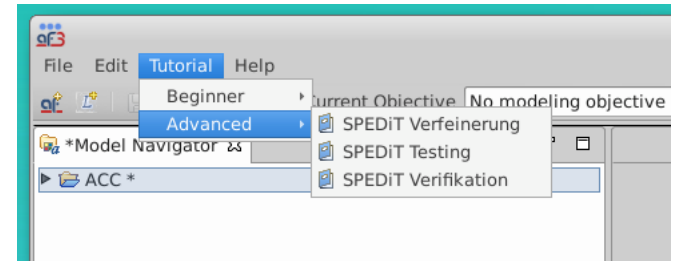


1. Create Function

- A. Öffnen Sie den Functional Viewpoint mit einem Doppelklick auf das Element "Functional Viewpoint".
- B. Um eine neue Systemfunktion anzulegen, ziehen Sie aus dem Fenster "Model Elements" (auf der rechten Seite) ein neues "Component" Element auf die Zeichenfläche des Functional Viewpoints.
- C. Geben Sie der Systemfunktion den Namen "Propulsion Function", indem Sie auf das gerade angelegte Component Element klicken und im "Properties" Fenster (im unteren Bereich) den Namen von "Component" auf "Propulsion Function" ändern.



- Komplexeres Tutorial – Verifikation
- Benutzung eines Model-Checkers zur Überprüfung von Zwischenschritten



Automatenspezifikation

Präsenz Sensor 1
Licht Sensor 1
Lampe 1

Tageslicht

PS 2
LS 2
Lampe 2

Helligkeits-Regler

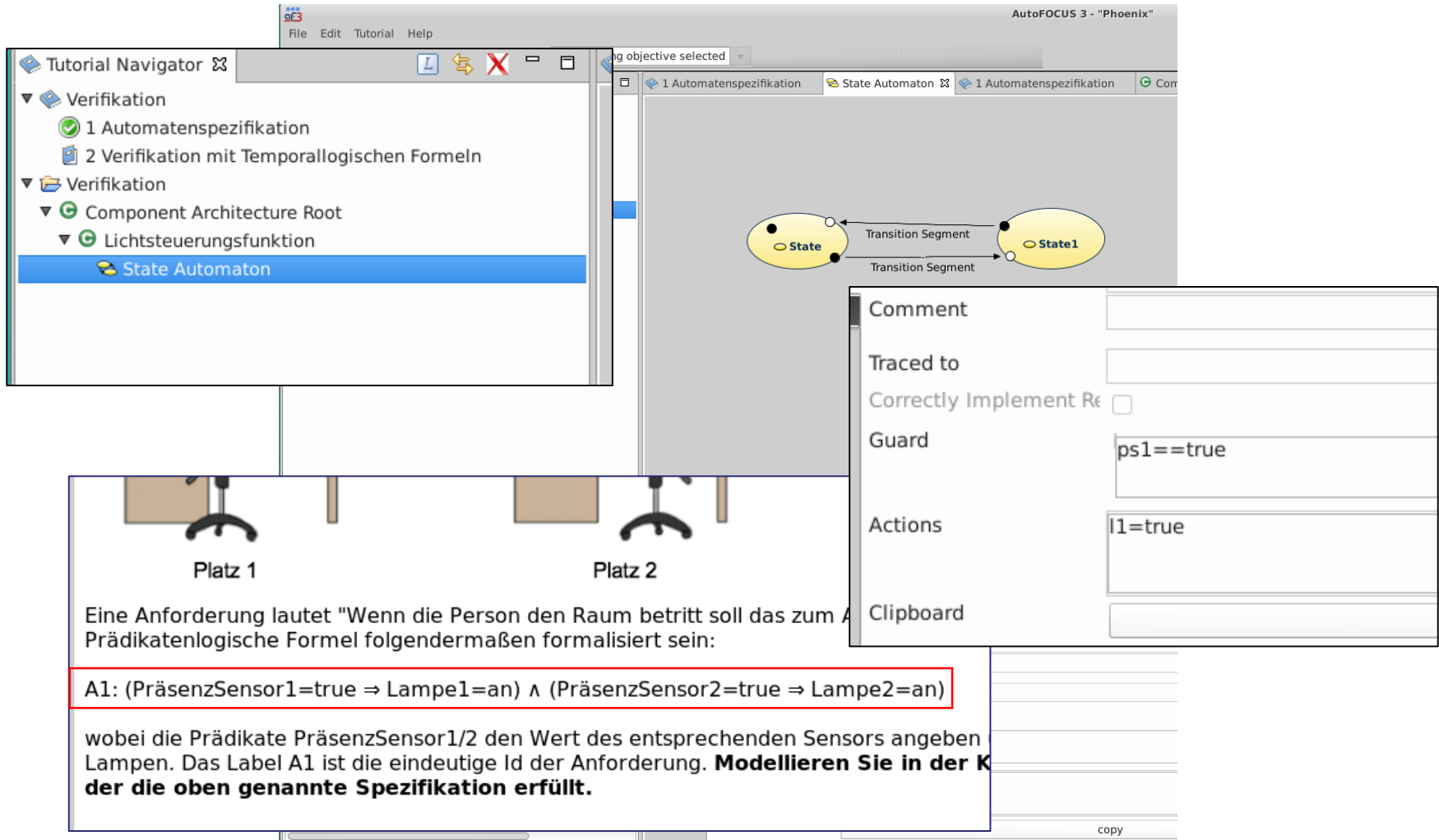
Platz 1

Platz 2

Eine Anforderung lautet "Wenn die Person den Raum betritt soll das zum Arbeitsplatz der Person gehörende Licht sofort angehen". Dies könnte als Prädikatenlogische Formel folgendermaßen formalisiert sein:

$$A1: (\text{PräsenzSensor1}=\text{true} \Rightarrow \text{Lampe1}=\text{an}) \wedge (\text{PräsenzSensor2}=\text{true} \Rightarrow \text{Lampe2}=\text{an})$$

wobei die Prädikate PräsenzSensor1/2 den Wert des entsprechenden Sensors angeben und Lampe1/2 den aktuellen Zustand der entsprechenden Lampen. Das Label A1 ist die eindeutige Id der Anforderung. **Modellieren Sie in der Komponente "Lichtsteuerungsfunktion" einen Automaten, der die oben genannte Spezifikation erfüllt.**



The screenshot displays the AutoFOCUS 3 - "Phoenix" software interface. On the left, the Tutorial Navigator shows a tree structure with "State Automaton" selected. The main workspace shows a state automaton with two states, "State" and "State1", connected by two "Transition Segment" arrows. A context menu is open over the automaton, showing fields for "Comment", "Traced to", "Correctly Implement Re" (with a checkbox), "Guard" (containing "ps1==true"), "Actions" (containing "l1=true"), and "Clipboard". Below the workspace, a diagram shows two desks labeled "Platz 1" and "Platz 2". A text box contains the requirement A1: $(\text{PräsenzSensor1}=\text{true} \Rightarrow \text{Lampe1}=\text{an}) \wedge (\text{PräsenzSensor2}=\text{true} \Rightarrow \text{Lampe2}=\text{an})$, which is highlighted with a red border. Below this, a text box explains the predicates and asks the user to model the requirement.

Eine Anforderung lautet "Wenn die Person den Raum betritt soll das zum A1
Prädikatenlogische Formel folgendermaßen formalisiert sein:

A1: $(\text{PräsenzSensor1}=\text{true} \Rightarrow \text{Lampe1}=\text{an}) \wedge (\text{PräsenzSensor2}=\text{true} \Rightarrow \text{Lampe2}=\text{an})$

wobei die Prädikate PräsenzSensor1/2 den Wert des entsprechenden Sensors angeben
Lampen. Das Label A1 ist die eindeutige Id der Anforderung. **Modellieren Sie in der K
der die oben genannte Spezifikation erfüllt.**

- Nutzung des Tutorialmodus In Bezug auf bestimmte Rollen
- Z.B. „SPEDiT Persona“
 - Modul M01: System Architect / Persona "Sebastian Nitsch"
 - Modul M02: System Requirements Engineering / Persona "Kaiwen Li"
 - Modul M03: Test Engineer / Persona "Cornelius Cover"
 - Modul M04: Software Architect / Persona "Christian Strätz"
 - Modul M05: Software Developer / Persona "Hedwig Hacker"

- Rückspiegelung des Lernfortschritts im Tool in die moodle Lernumgebung



Toolübung in einer Lerneinheit (Block)

60 %

Lernfortschritt im Tool

moodle

Remote-Tool-Server

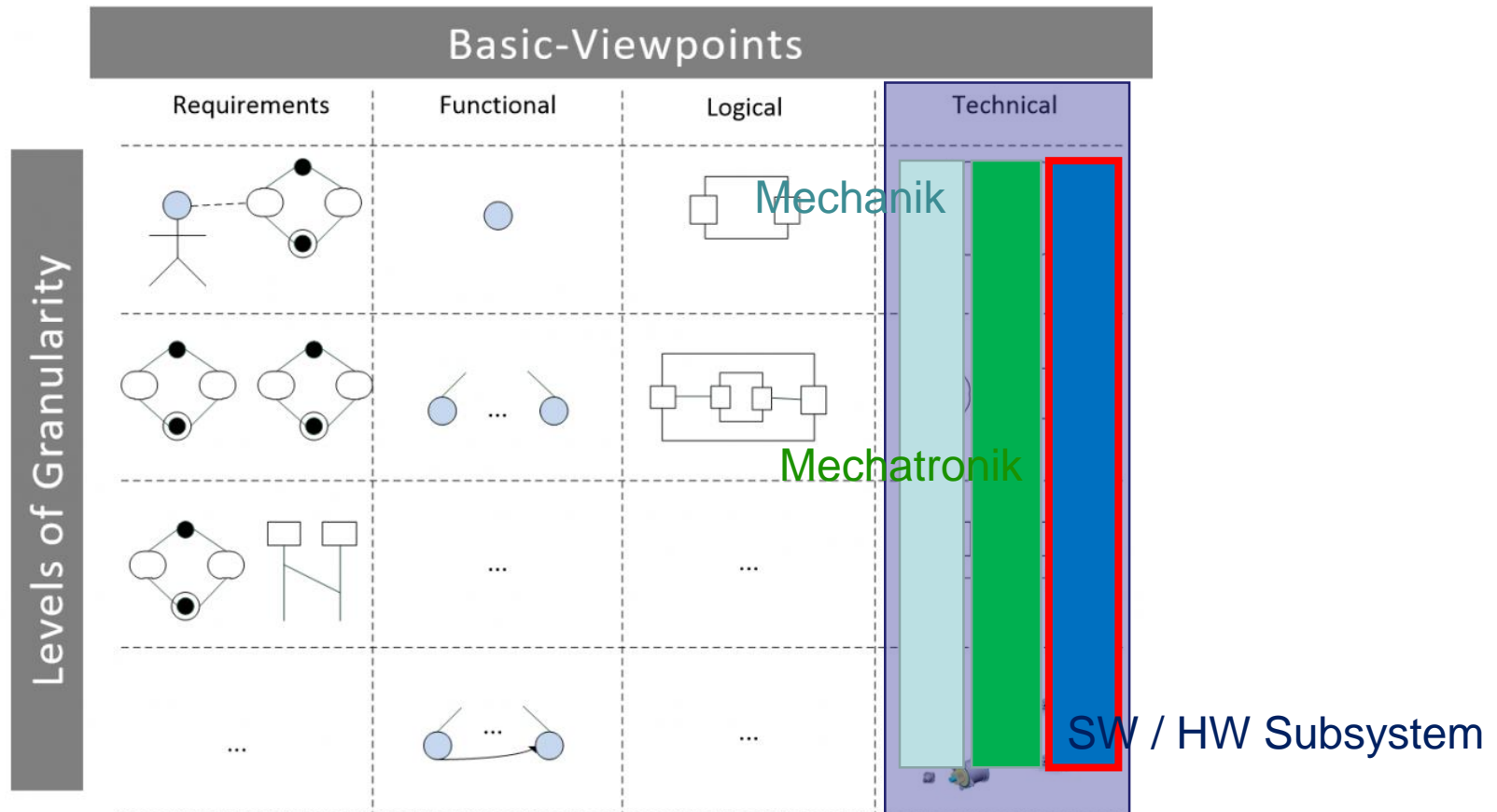
Integration einer SPES Technologie in die SPEDiT Toolkette



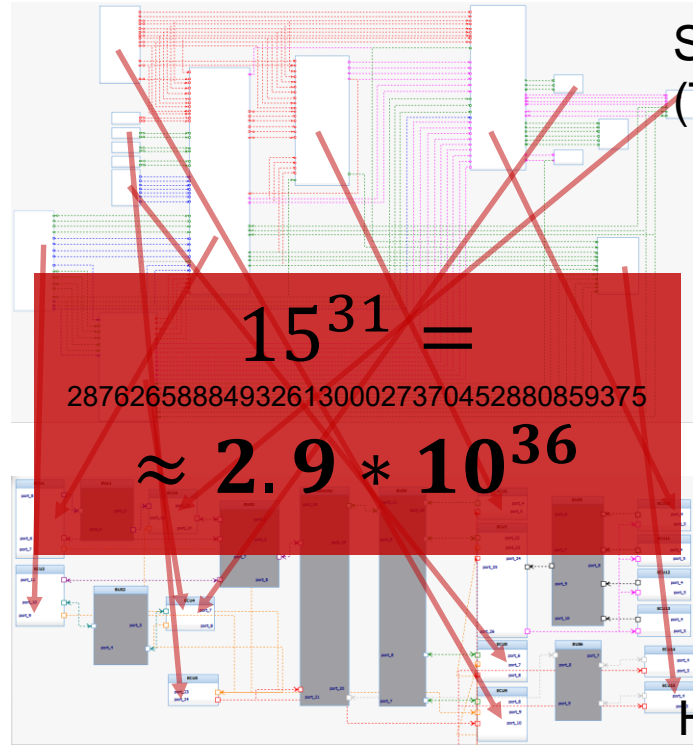
- Nutzung des open-source Tools AutoFOCUS3 um eine in SPES XT entwickelte Automatisierungsmethode im PTC Integrity Modeler zur Verfügung zu Stellen
- Nutzung des SPES-Viewpoint Konzepts als Grundlage für den Datenaustausch zwischen den Tools



- Auf welcher technischen Ausführungseinheit (ECU, Processor, ...) werden welche Software Komponenten ausgeführt?

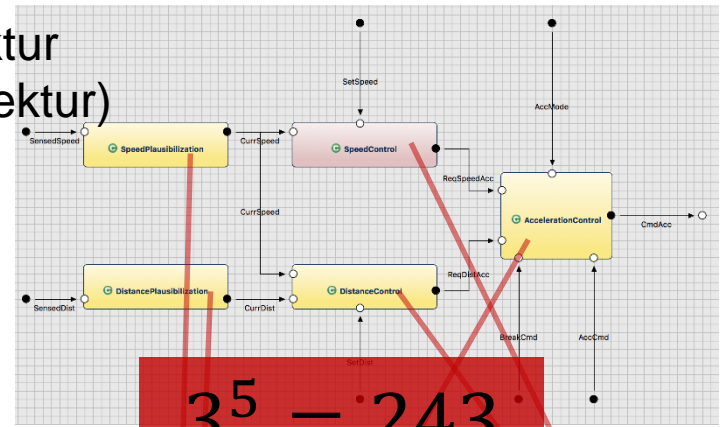


„Zustandsraumexplosion“



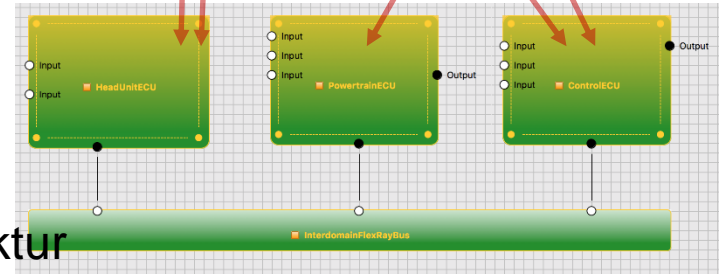
Industrial sized use case

SW Architektur (Task Architektur)



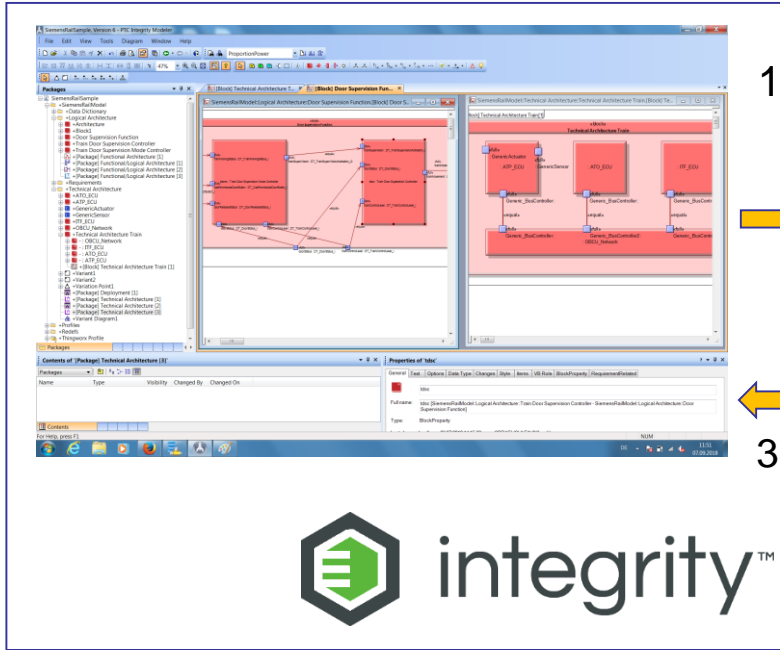
$3^5 = 243$

HW Architektur




Small use case

PTC Integrity Modeler <-> AutoFOCUS3



1. SW+HW Architektur

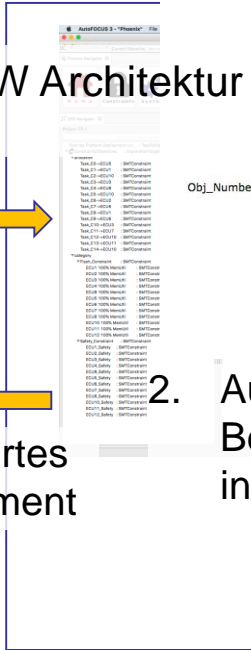
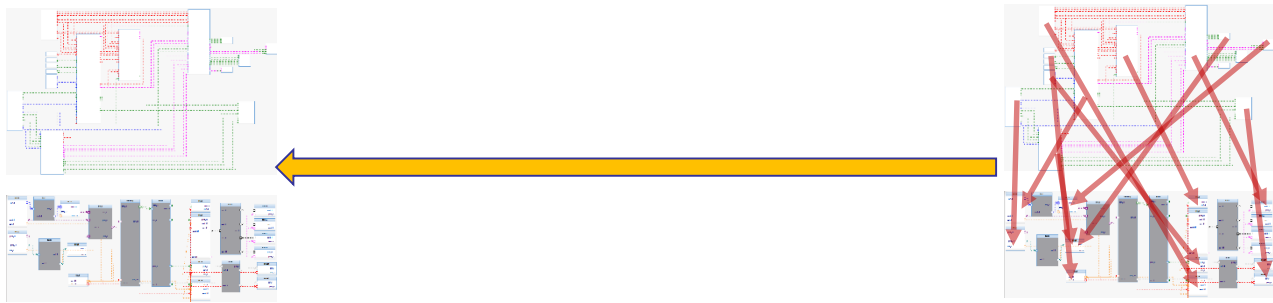
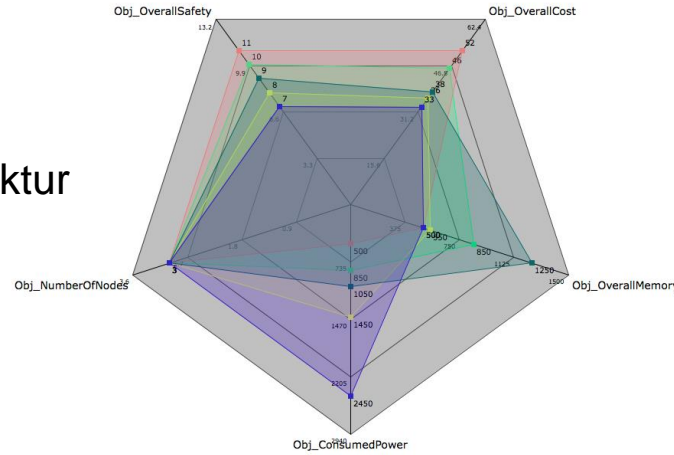
3. Optimiertes Deployment



1. SW+HW Architektur

COM API

2. Automatische Deployment Berechnung und Optimierung in AutoFOCUS3

Fragen?