Modellgetriebene Softwareentwicklung für Robotiksysteme

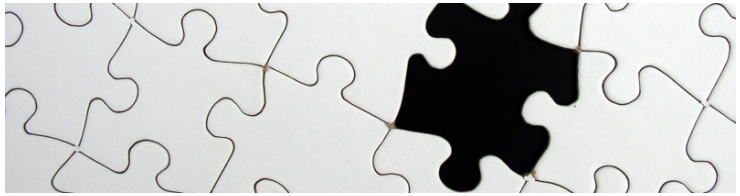
Andreas Steck, Dennis Stampfer und Christian Schlegel

Fakultät Informatik
ZAFH Servicerobotik
Hochschule Ulm

http://smart-robotics.sourceforge.net/
http://www.zafh-servicerobotik.de/ULM/index.php

Model Driven Software Development
Introduction and Motivation

What is this talk about ?



- not just another software framework
- not just another middleware wrapper

➔ we have plenty of those ...

But

- separation of robotics knowledge from short-cycled implementational technologies
- providing sophisticated and optimized software structures to robotics developers not requiring them to become a software expert

How to achieve this ?

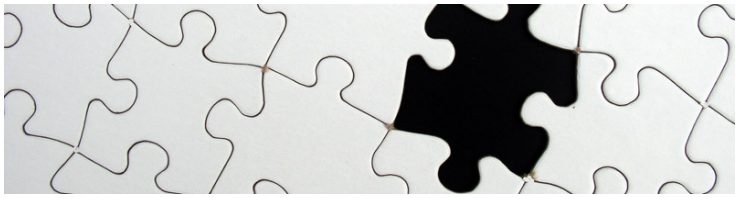
- make the step from code-driven to model-driven designs
- using common open source tools for robotics !

Modellgetriebene Softwareentwicklung
Einleitung und Motivation

Warum ist die modellgetriebene Softwareentwicklung wichtig für die Robotik?

- wegkommen von den “von Hand” programmierten Systemen
- entwerfen von Systemen aus Standardkomponenten mit explizierten Eigenschaften und Parametern
- wiederverwenden und modifizieren der Lösungen auf Modellebene
- von den Erfahrungen der Softwareingenieure profitieren; Transformationen (M2M / M2T), optimierte Frameworks, effiziente Strukturen, ...
- analysieren und verifizieren von Parametern
- berücksichtigen von ressource awareness

- ... und viele weitere

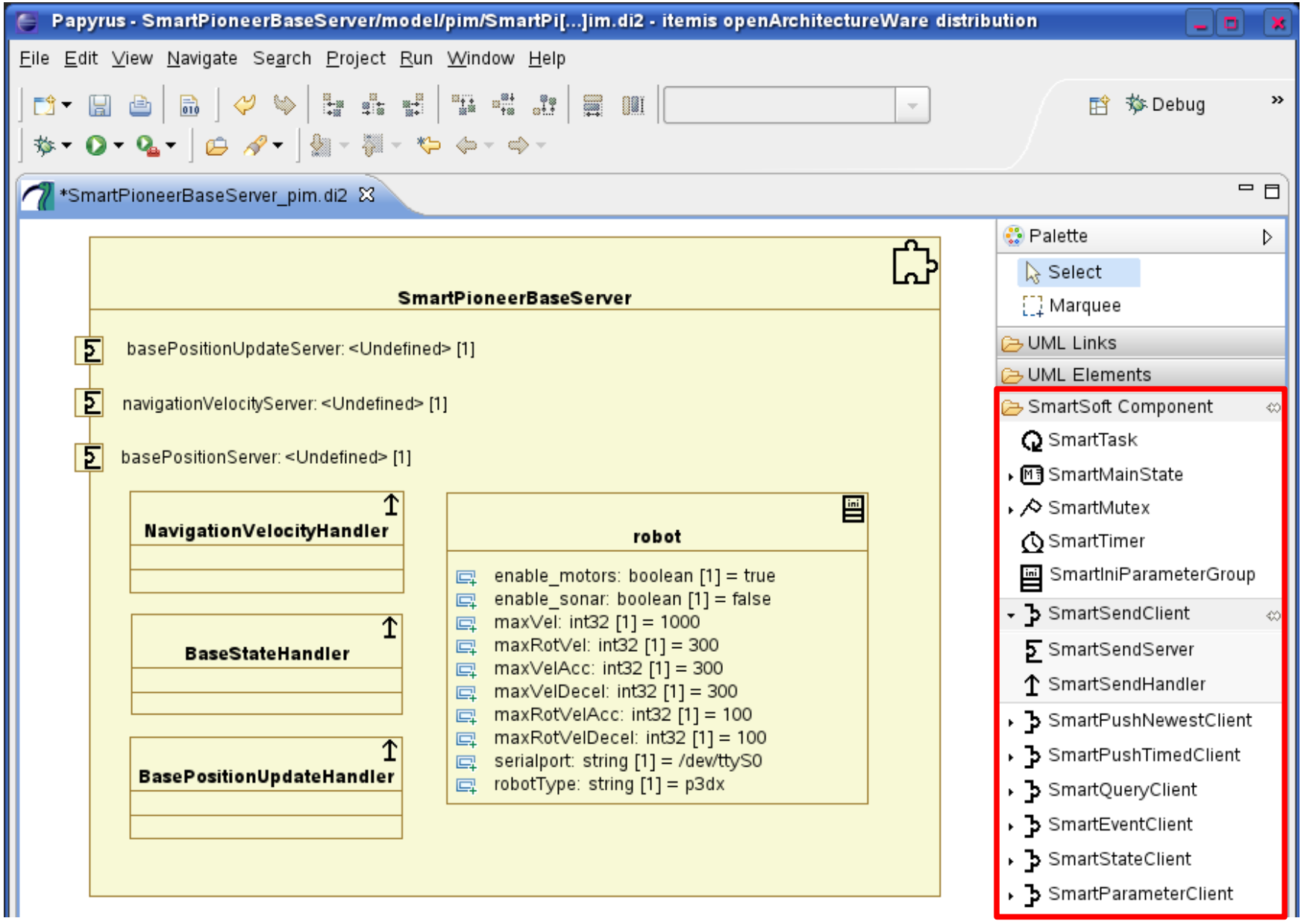


Der Schritt zur modellgetriebenen Softwareentwicklung ist maßgeblich zur Erschließung breiter Anwendungspotentiale für (Service)-Roboter

service robotik

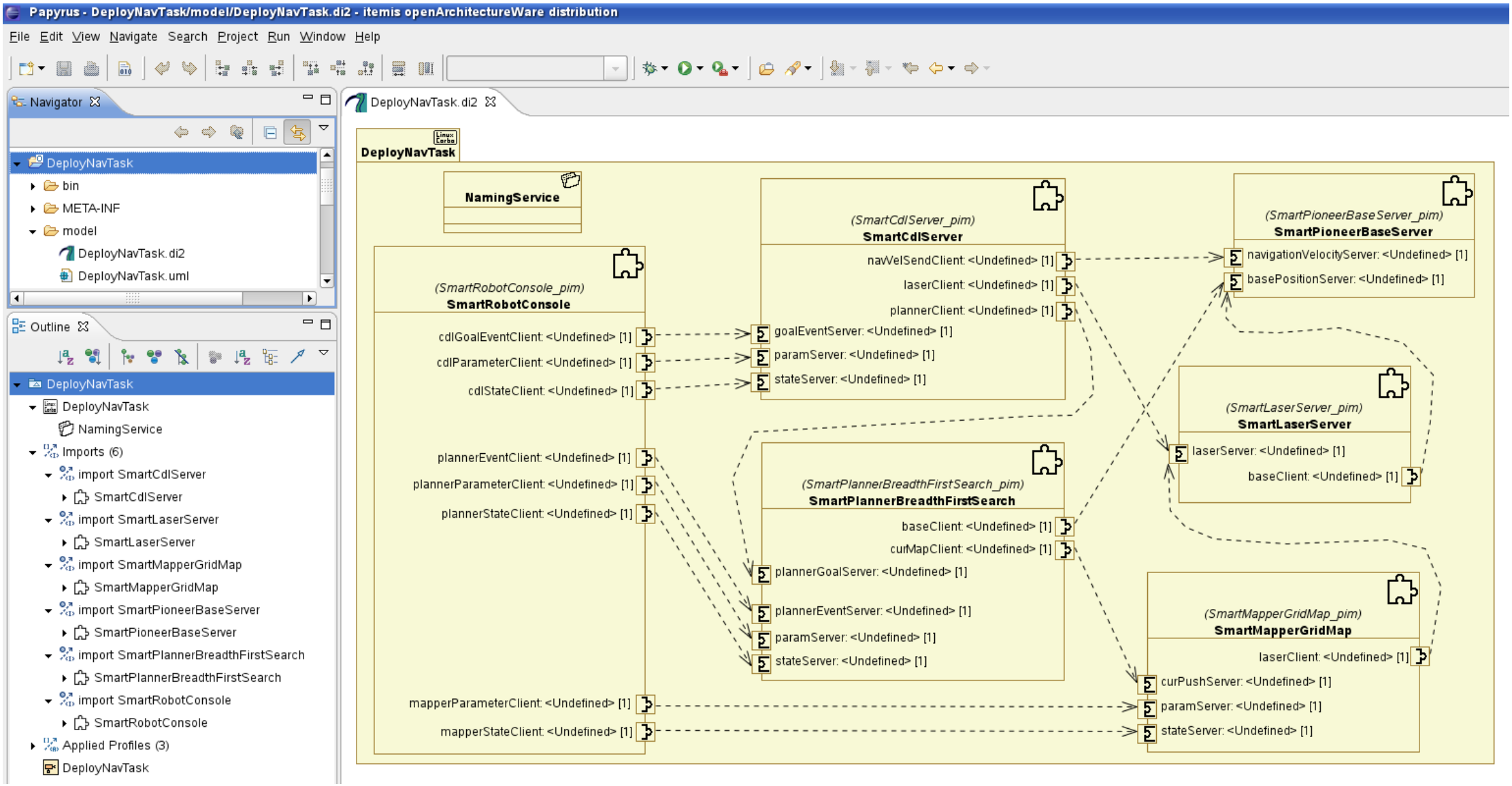
Modellgetriebene Softwareentwicklung
Entwicklung von Komponenten

Pioneer P3DX

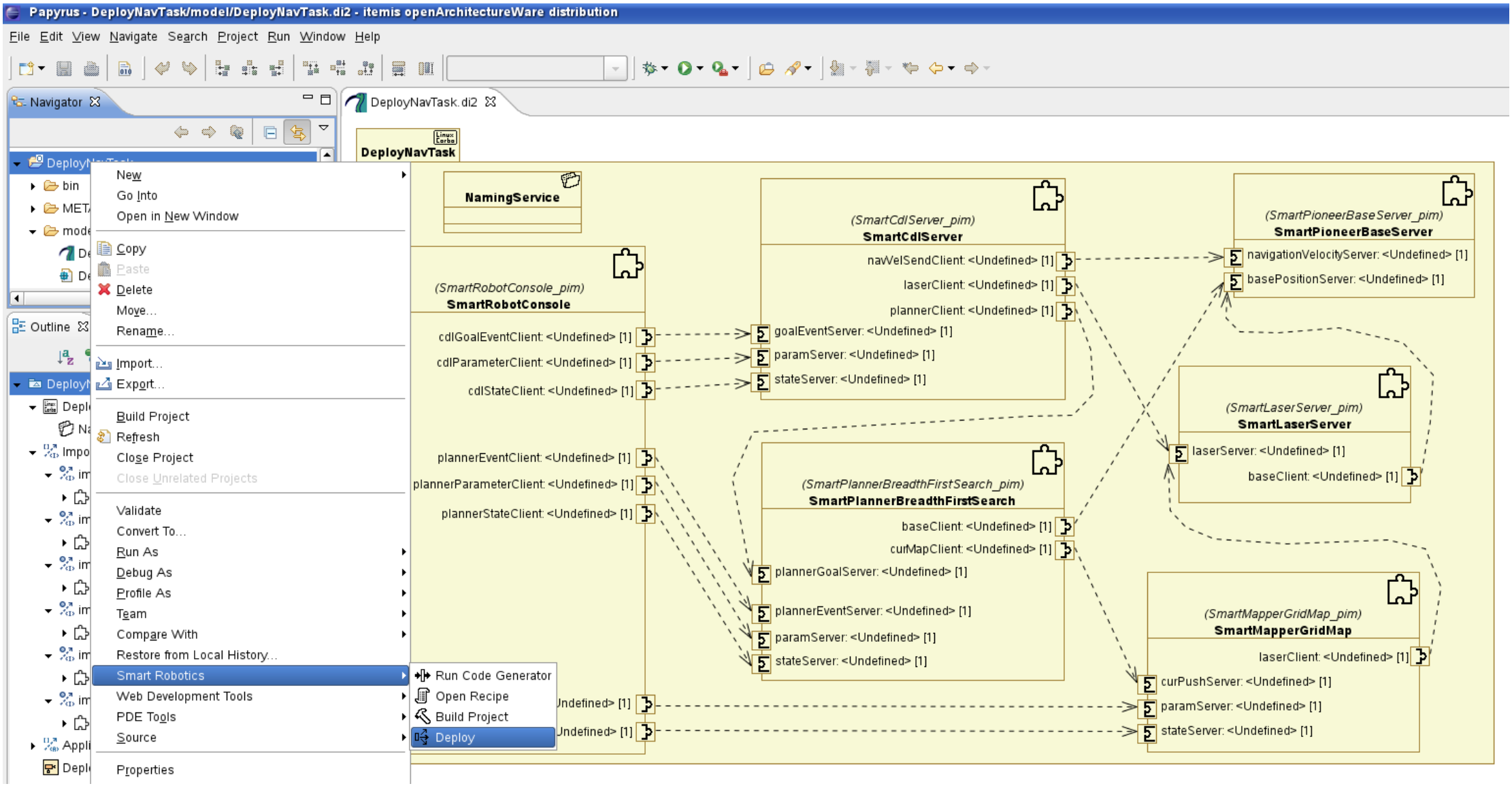


service robotik

Modellgetriebene Softwareentwicklung
Deployment von Komponenten



Modellgetriebene Softwareentwicklung
Deployment von Komponenten



Modellgetriebene Softwareentwicklung
Einleitung und Motivation

Wir haben einige ausschlaggebende Entscheidungen getroffen:

- Granularität, wie ein System aus Komponenten zusammengesetzt wird
  - lose gekoppelte Komponenten mit lokalen Verantwortlichkeiten
  - Services werden zur Verfügung gestellt / benötigt
- Streng festgelegte Interaktionsmuster zwischen den Komponenten
  - präzise definierte Semantik der Interaktionen
  - diese sind als Richtlinien zu verstehen (auf jede Middleware abbildbar)

  ➔*unabhängig von der Implementierungstechnologie*
- minimales Komponentenmodell zur Systemintegration
  - dynamic wiring des Datenflusses zwischen den Komponenten
  - interner Zustandsautomat um Komponenten in unterschiedliche Betriebsmodi zu bringen
  - param-/ config-Port um Komponenten von außen zu orchestrieren

  ➔*sicherstellen der Zusammensetzbarkeit/ Systemintegration*
- unabhängig von der Ausführungsumgebung
  - Tasks (periodic, non-periodic, hard real-time, no realtime), Synchronisation

  ➔*kann wiederum auf verschiedene Betriebssysteme abgebildet werden*

Modellgetriebene Softwareentwicklung
Das SmartSoft Framework

SmartSoft:

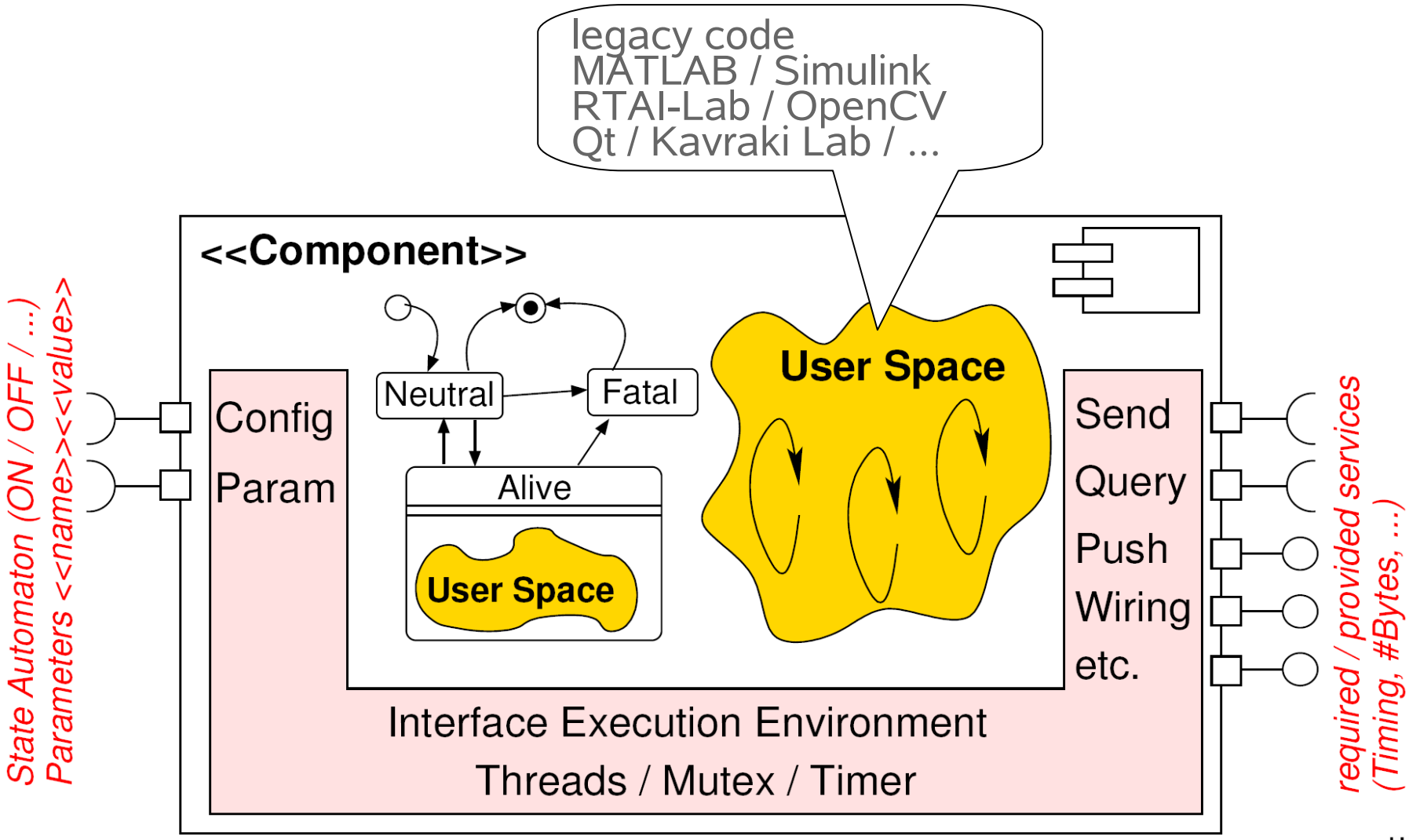
- die Idee
  - wie Systeme aus Komponenten zusammengesetzt werden
  - wie die Komponentenhülle aussieht
  - wie Komponenten miteinander interagieren
- die konkreten Implementierungen, basierend auf:
  - CORBA (ACE/TAO) => CorbaSmartSoft
  - ACE (ohne CORBA / TAO)
  - ...

Die Auswahl der Muster ist ausreichend, da sowohl Request/Response Interaktionen als auch asynchrone Mitteilungen und Push Services zur Verfügung stehen.

Die SmartSoft Interaktionsmuster

send	one-way communication
query	two-way request/response
push newest	1-to-n distribution
push timed	1-to-n distribution
event	asynchronous conditioned notification
wiring	dynamic component wiring

Modellgetriebene Softwareentwicklung
Das SmartSoft Framework



Hochschule Ulm

Modellgetriebene Softwareentwicklung
SmartSoft modellgetrieben

PIM

SmartMARS – Metamodel

(**M**odeling and **A**nalysis of **R**obotics **S**ystems)

- UML2-Profil
- plattformunabhängige Stereotypen
  - SmartComponent
  - SmartTask
  - SmartMutex
  - SmartQueryServer
  - SmartEventClient
  - ...

M2M
oAW
xTend

PSM

CorbaSmartSoft
CORBA-basierte Implementierung von SmartSoft

AceSmartSoft
ACE-basierte Implementierung von SmartSoft

Microsoft Robotic Studio
MSRS-basierte Implementierung

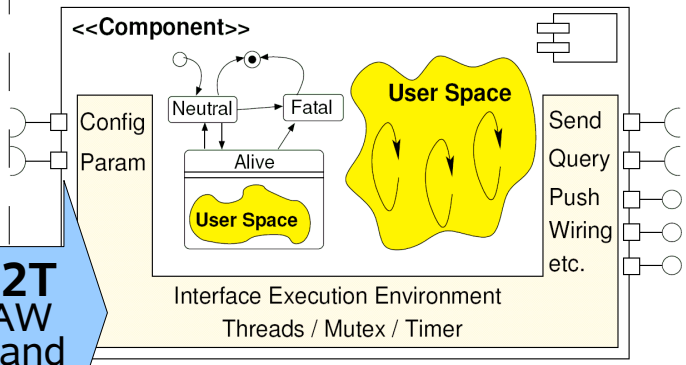
...
jede andere Middleware

- UML2-Profil
- plattformspezifische Stereotypen

dies wird von Middlewareexperten zur Verfügung gestellt

M2T
oAW
xPand

PSI



Im Userspace können beliebige Libraries verwendet werden

Der Userspace muss nicht angepasst werden, wenn sich die Plattform ändert

Es wird ausschließlich die Komponentenhülle erzeugt

Hochschule Ulm

Modellgetriebene Softwareentwicklung
SmartSoft modellgetrieben

PIM

SmartMARS – Metamodel

(**M**odeling and **A**nalysis of **R**obotics **S**ystems)

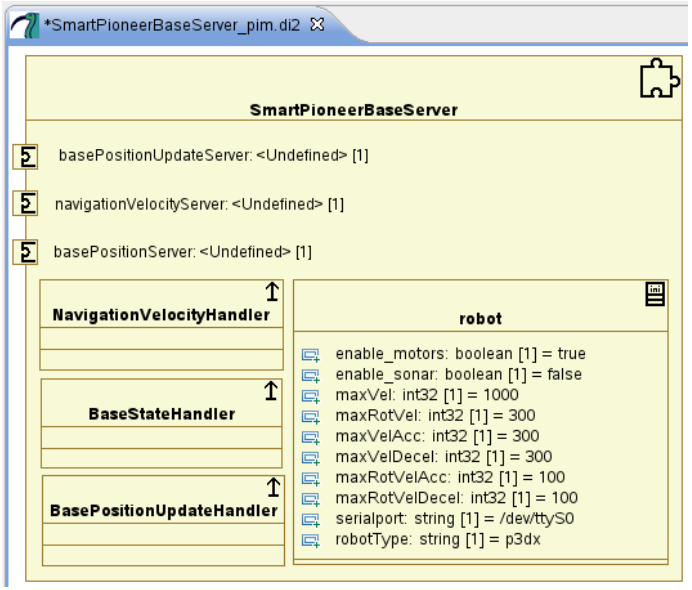
M2M
oAW
xTend

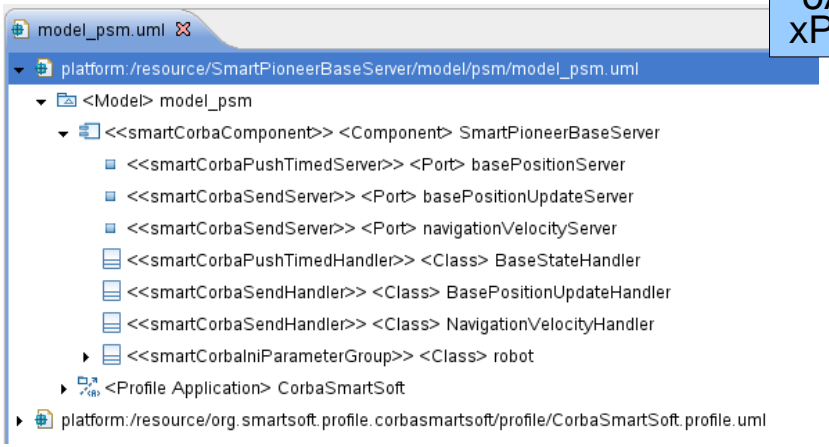
PSM

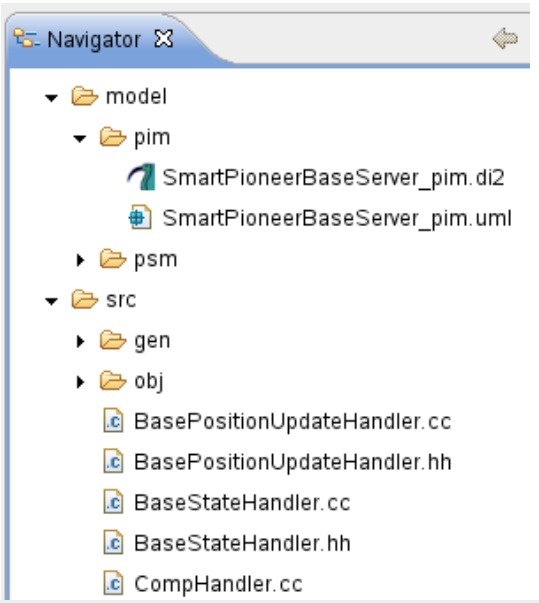
CorbaSmartSoft
CORBA-basierte Implementierung von SmartSoft

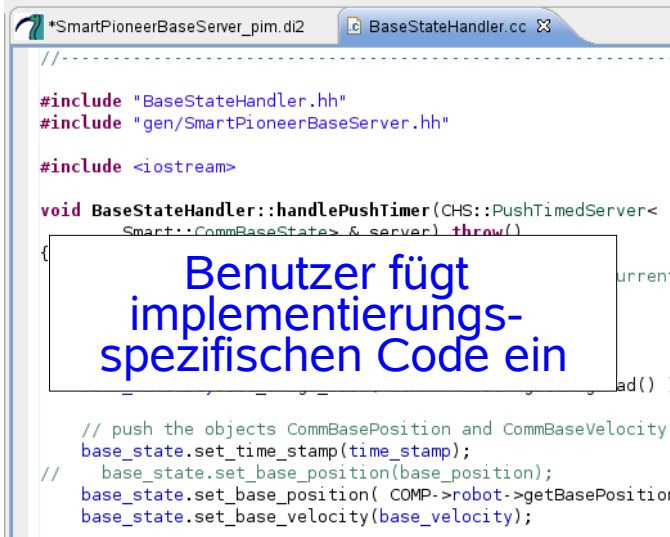
M2T
oAW
xPand

PSI









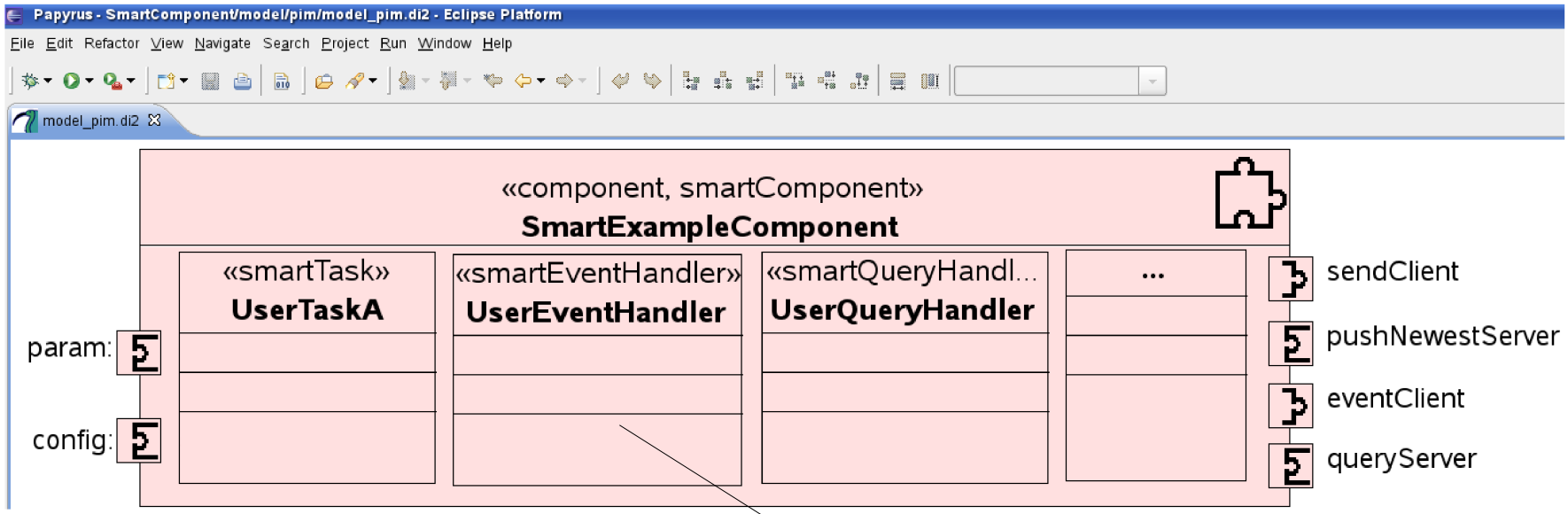
Benutzer erstellt PIM

PSM wird vom Workflow generiert

Benutzer fügt implementierungs-spezifischen Code ein

Model Driven Software Development Workflow Example (User View)

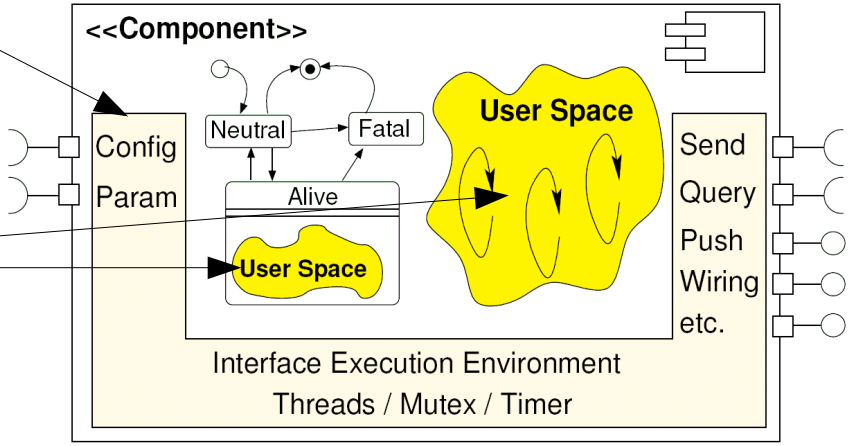
PIM



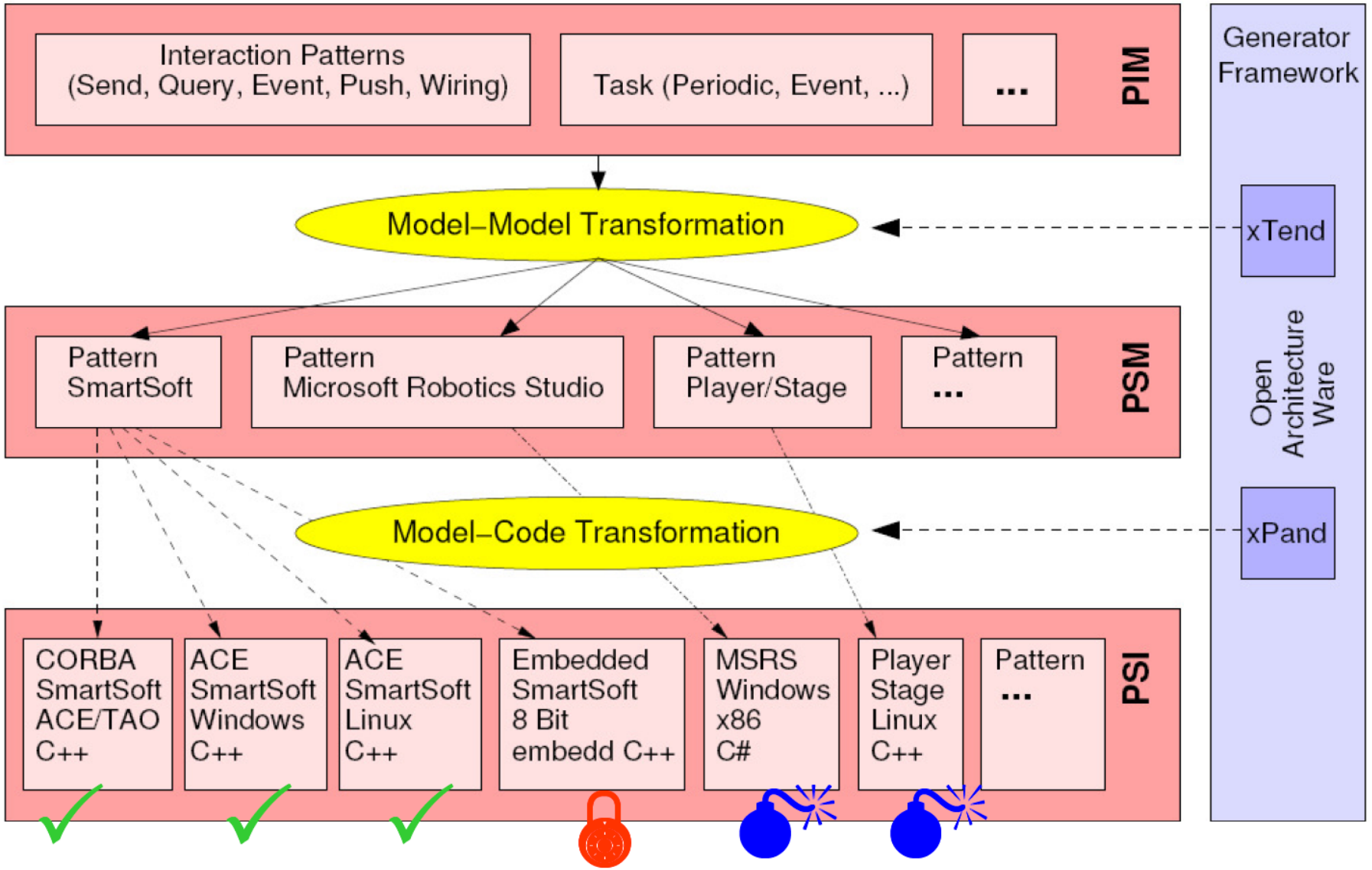
verification (e.g. QoS)+ transformation

executable component

User Code
MATLAB / Simulink
RTAI-Lab
OpenCV / Qt / Kavraki-Lab

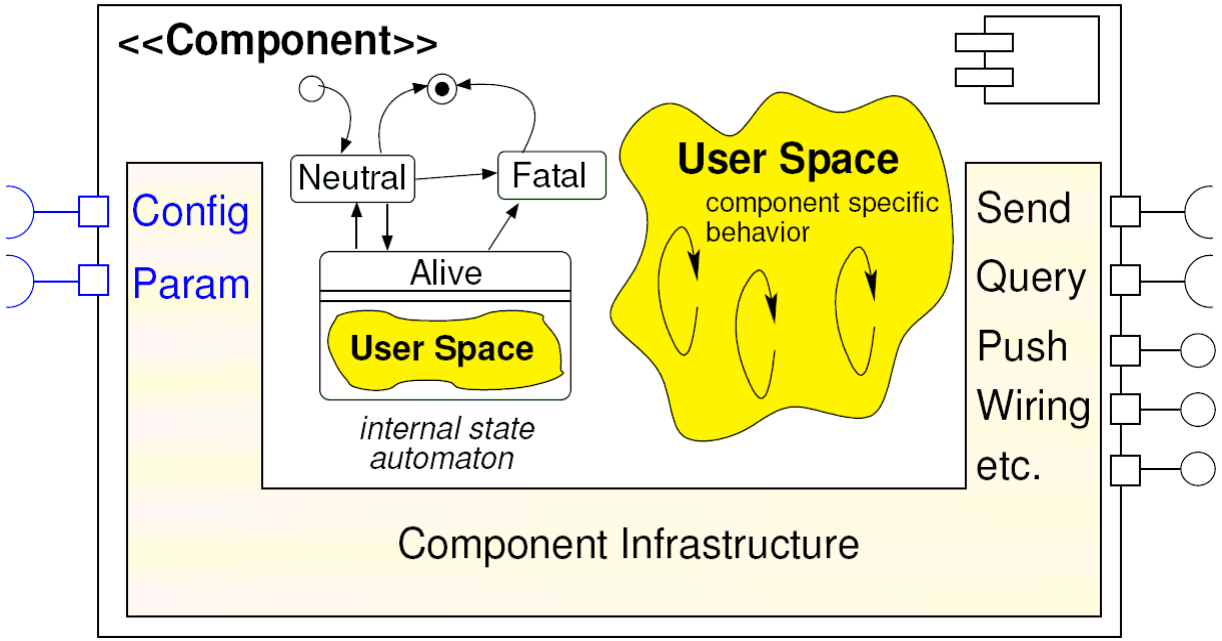


Modellgetriebene Softwareentwicklung
SmartSoft modellgetrieben



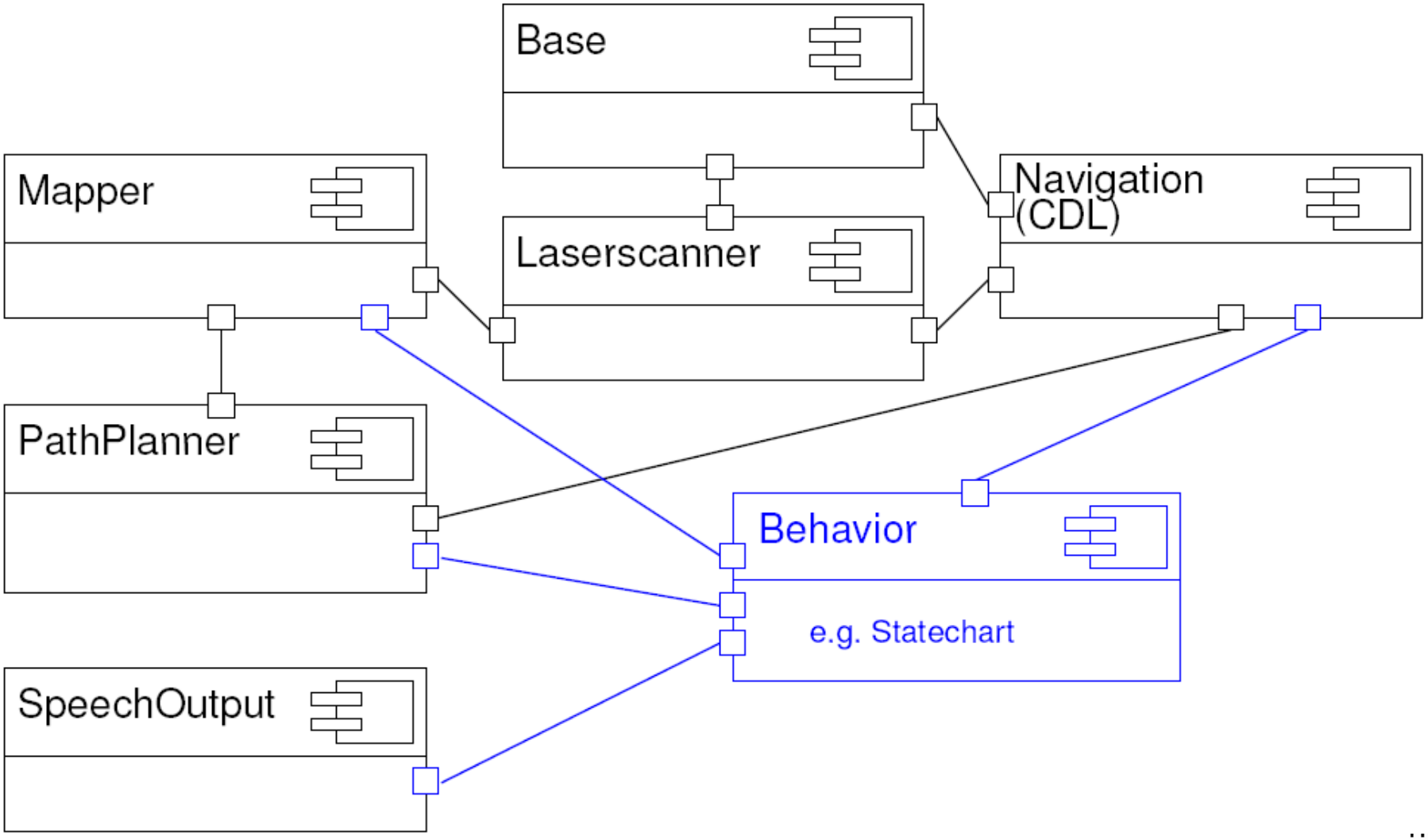
Modellgetriebene Softwareentwicklung
SmartSoft modellgetrieben

vereinheitlichter Komponentenzugriff von außen



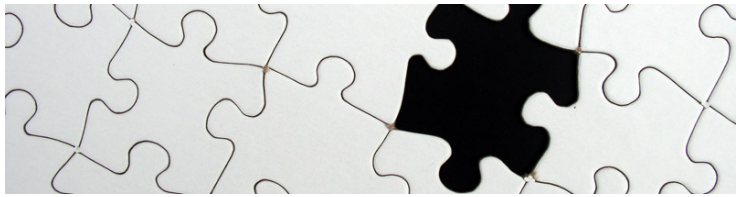
- **`Config:`** setzen des internen Zustands der Komponente
  - navigation: **`neutral; moverobot`**
- **`Param:`** senden von Parametern
  - navigation: **`(TRANSVEL(0)(500))`**
  - pathPlanner: **`(SETDESTINATIONCIRCLE(xPos)(yPos)(dist))`**

Modellgetriebene Softwareentwicklung
SmartSoft modellgetrieben



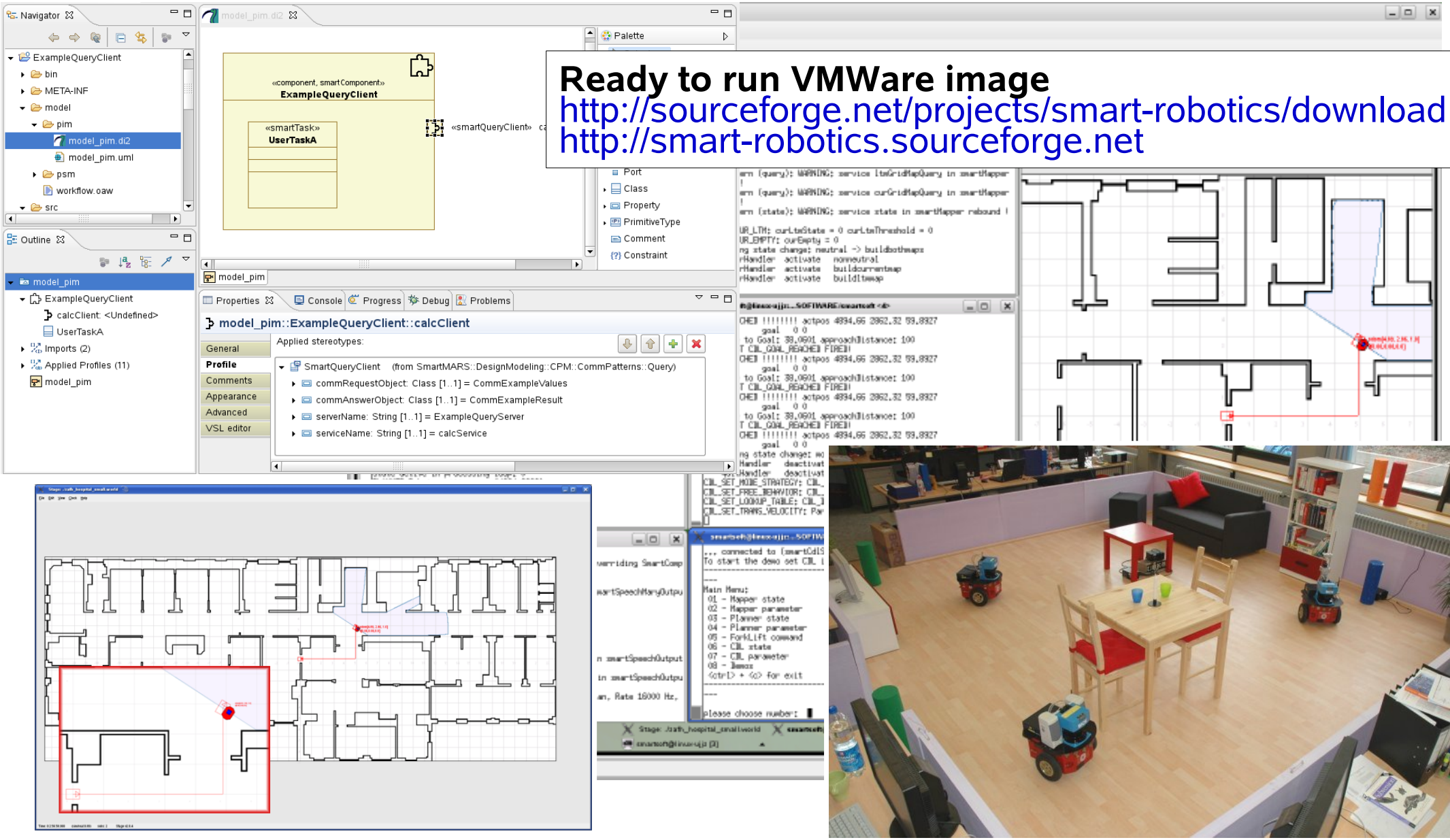
Model Driven Software Development
Zusammenfassung und Ausblick

- Toolchain basiert auf openArchitectureWare
  - http://www.openarchitectureware.org/
  - UML-Tool: PapyrusUML
  - in Eclipse integriert



- MDSD Toolchain
  - PIM: SmartMARS Robotik Profil (Modeling and Analysis of Robotics Systems)
  - PSM: SmartSoft mit verschiedenen Technologien umgesetzt, jedoch mit derselben Semantik!
    - kann an weitere PSM's angepasst werden
- SmartSoft [LGPL]
  - http://smart-robotics.sourceforge.net/
  - CORBA (ACE/TAO) basierte SmartSoft Version
    - frei auf Sourceforge verfügbar
  - ACE (ohne CORBA) basierte SmartSoft Version
    - demnächst auf Sourceforge verfügbar [Linux, Windows, ...]
  - oAW Toolchain für SmartSoft
    - frei auf Sourceforge verfügbar

Model Driven Software Development
Zusammenfassung und Ausblick



Ready to run VMWare image
http://sourceforge.net/projects/smart-robotics/download
http://smart-robotics.sourceforge.net