

Making Ideas a Reality



MDA entzaubert – mit Ameos

Dr. Peter Dencker
Klaus Wachsmuth



Agenda

- Aonix – wer ist das?
- Wie kam es (Aonix, OMG) zu MDA?
- Wozu ist MDA nützlich?
- Wo stößt MDA heute (noch) an Grenzen?
- Wie ist der Entwicklungsprozess mit MDA?
- Wie funktioniert MDA mit einem konkreten Modellierungswerkzeug Ameos/UML unter Verwendung von UML 2.0 Profilen?
- ...
- Wann gibt's den Apéro?



Aonix Strategie

- Software-Werkzeug-Anbieter
 - für's Modellieren, Programmieren, Testen, ...
 - mit starker Dienstleistungskomponente
- Unser Motto:

Qualität und Produktivität
in der Software-Entwicklung



Aonix Historie

'80

'85

'90

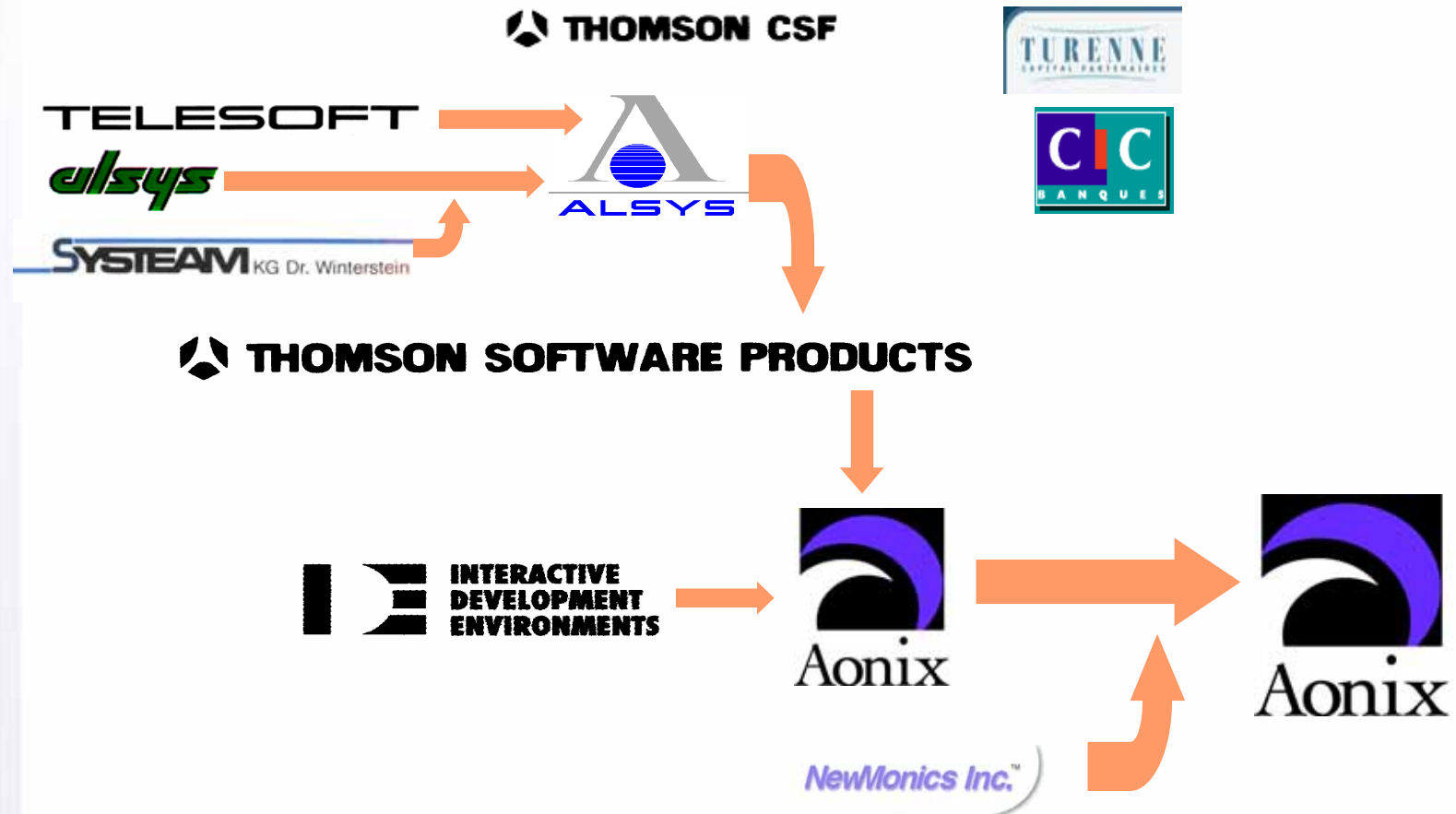
'92

'94

'96

'03

'04





Aonix Märkte

- Branchen
 - Automobilindustrie, Telekommunikation
 - Luft- und Raumfahrt, Militärtechnik, Transport
 - Energie, Medizintechnik, Industrieautomation
 - Banken, Versicherungen
- Anwendungsgebiete
 - Fortschrittliche & komplexe Systeme
 - Realzeit- und eingebettete Systeme
 - Unternehmenskritische Systeme
 - Sicherheitskritische Systeme

- Mitglied bei der OMG (seit 1992)
 - UML/MDA
 - Profile: EJB, Realtime, System Engineering
- Leitung GI Fachgruppe Ada
 - Mitgliedschaft GI FB Sicherheit - Schutz und Zuverlässigkeit
- Mitgliedschaft VDE/GMA FA 5.13
„Zuverlässige Software in automatisierten Systemen“
- Marketing Leitung J-Consortium
- EU Projekt HIDOORS
 - High Integrity Distributed Object Oriented Realtime Systems
 - Realtime Java
 - Realtime UML
- Project HAWK
 - Common RT Java Standard von Sun und J-Consortium



- **Ameos/UML**
Das Modellierungswerkzeug der nächsten Generation



Software through pictures



- **Software through Pictures (StP)**
Familie von Modellierungswerkzeugen
 - /UML für objektorientierte Analyse und Design
 - /ACD Template gesteuerte Codegenerierung
 - /SE für strukturierte Analyse und Design



- **ObjectAda**
Entwicklungsumgebung für Ada 95



- **Raven**
zertifizierbarer Laufzeitkernel für sicherheitskritische Echtzeitanwendungen nach DO178B, IEC 61508 u.a.

Produktfamilien II

PERC

- **PERC**
Embedded Java VM für zuverlässige Echtzeitanwendungen mit umfassender Entwicklungsumgebung



- **TeleUSE**
Generator für graphische Benutzeroberflächen auf der Basis von Motif



- **VectorCAST**
Modultest und Coverage Analysewerkzeug mit hohem Automatisierungsgrad für die Sprachen Ada 83, Ada 95, C und C++



StP/Ameos MDA Kundenauswahl

- Astrium (Simulation, System Engineering)
- Bizerba (Fleischerwaagen)
- BMW (Corporate Data Model)
- Bosch (Variantenreiche Embedded C Software, Getriebe, Airbags)
- FZ Ka (Thermische Abfallverwertung)
- HIMA (Sicherheitskritische SPSen)
- Kordoba (Unternehmensweites Datenmodell, Cobol)
- Nord-Micro (Kabinendrucksystem, MISRA C)
- Siemens PSE (Reisereservierungssystem)
- Swisslog (Logistiksysteme)

- Supply Chain Management System 1999
 - Zühlke Engineering empfahl Aonix ACD Ansatz
- Konfigurierbarer AutomationManager
 - a set of real-time control systems that provides all the components of warehouse logistical automation.
- „A **three** man month deployment now takes just **one** month. We could not have done this without Aonix ACD (MDA)“.
- Nachklapp:
 - OO Datenbank Architektur wurde 2002 in 3 Monaten nach Oracle umgeschrieben.



Agenda

- Aonix – wer ist das?
- **Wie kam es (Aonix, OMG) zu MDA?**
- Wozu ist MDA nützlich?
- Wo stößt MDA heute (noch) an Grenzen?
- Wie ist der Entwicklungsprozess mit MDA?
- Wie funktioniert MDA mit einem konkreten Modellierungswerkzeug Ameos/UML unter Verwendung von UML 2.0 Profilen?
- ...
- Wann gibt's den Apéro?

- Projekt 1997:
 - EuroSTART Reisereservierungs- und Buchungssystem
 - 3-Schichten-Architektur
 - Präsentation, Geschäftslogik, Datenhaltung
 - Verteilte Anwendung mit CORBA
 - 1 Mio Zeilen Quellcode
 - 70 Projektmitarbeiter
- Schlüsselkriterien:
 - Anpaßbarkeit von StP an Entwicklungsprozeß
 - Integrationsfähigkeit mit SNIFF+ und CM Werkzeug
 - IDL Code Generierung



Ökonomie der Templates

$$\text{Produktivitäts-Gewinn} = \frac{\text{LOC generierter Code} * 1x}{\text{LOC Templates} * 3x}$$

Annahme: der Aufwand Templates zu schreiben ist 3 mal so groß, wie für Zielcode

EuroSTART Projekt:

$$\frac{400\,000 \text{ LOC}}{12\,000 \text{ LOC} * 3} = 11$$

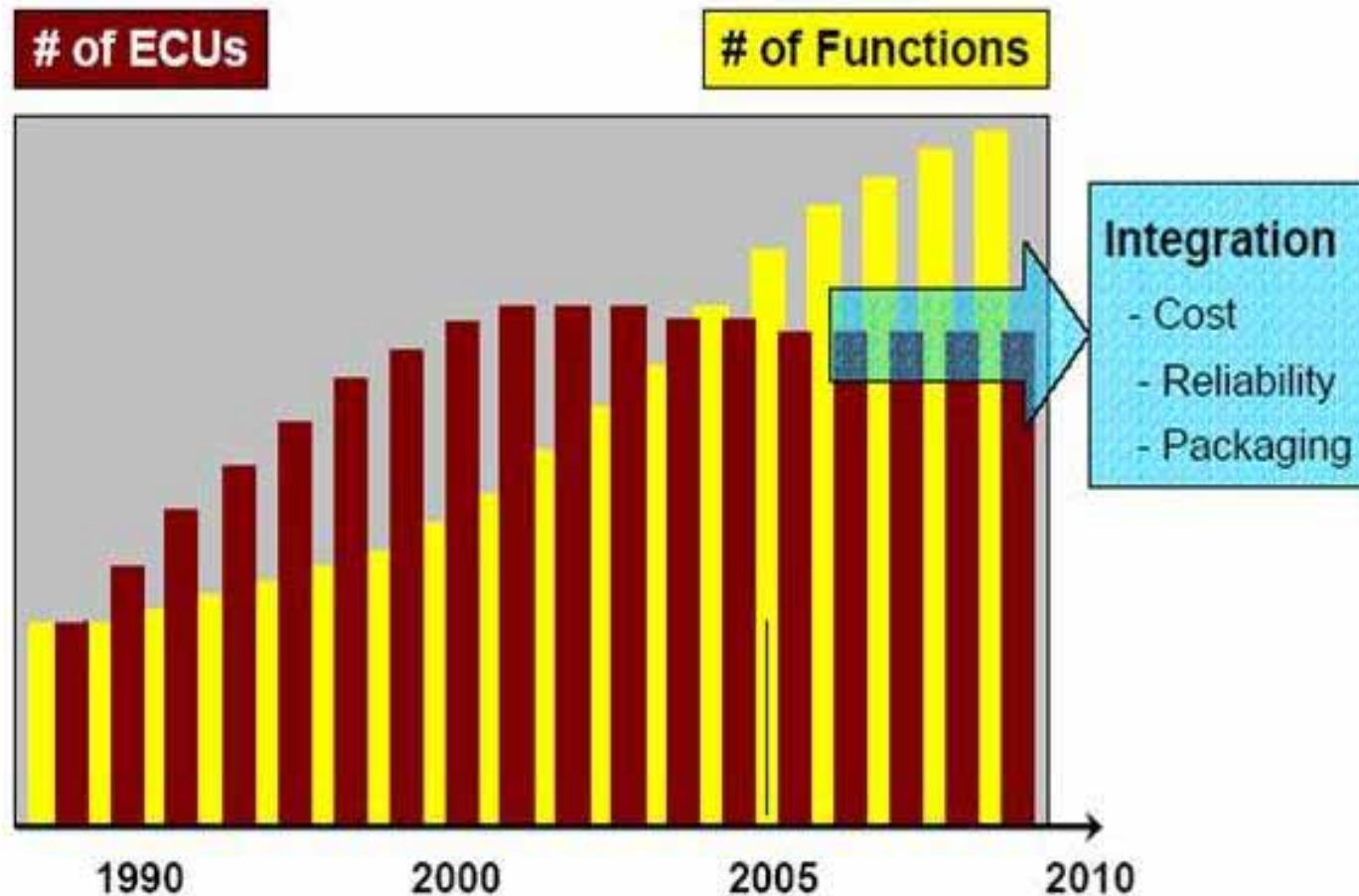
Das EuroSTART Projekt war **11 mal** so produktiv durch das Generieren von 400 000 LOC, anstatt sie manuell zu implementieren!

Status der SW Entwicklung

- Anteil der Software wächst in allen Bereichen
- Anforderungen an SW Systeme wachsen
 - Benutzerfreundlichkeit
 - Hohe Verfügbarkeit
 - Gutes Zeitverhalten
- Mix aus unterschiedlichen Technologien
- Höhere Komplexität der Anwendungen

Komplexitätsentwicklung bei MBtech

Geschäftsfeld Elektrik/Elektronik



Wandel der Systemlandschaft

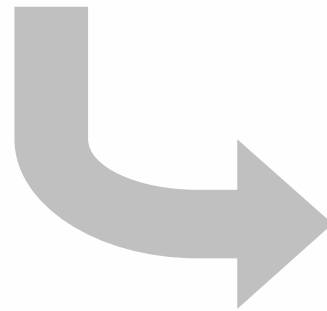
Früher

- Einzelplatzlösungen
- Isolierte Anwendungen
- Kooperation über
 - Dateien
 - Datenbanken



Jetzt zunehmend

- Softwareintegration
- Verschränkte Planung
 - Strategische Planung
 - Operative Planung
 - Aktualdisposition



Status der SW Entwicklung

- Anteil der Software wächst in allen Bereichen
- Anforderungen an SW Systeme wachsen
 - Benutzerfreundlichkeit
 - Hohe Verfügbarkeit
 - Gutes Zeitverhalten
- Mix aus unterschiedlichen Technologien
- Höhere Komplexität der Anwendungen
- ➔ Bändigung der Komplexität durch höhere Abstraktion und der Technologien durch Expertenwissen

Status der SW Entwicklung

- Anteil der Software wächst in allen Bereichen
 - Anforderungen an SW Systeme wachsen
 - Benutzerfreundlichkeit
 - Hohe Verfügbarkeit
 - Gutes Zeitverhalten
 - Mix aus unterschiedlichen Technologien
 - Höhere Komplexität der Anwendungen
- ➡ Abstraktionsgewinn durch die Verwendung von Modellen
- Vervielfältigen von Expertenwissen durch „Klonen“

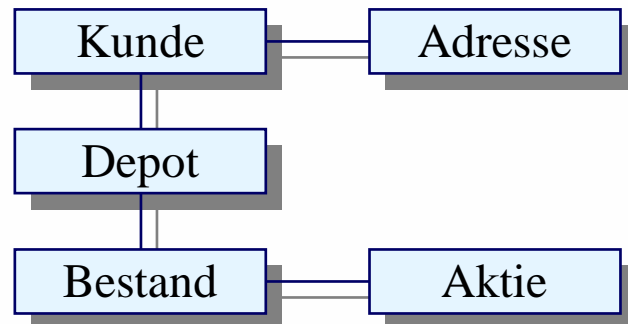
Ziel ist ein lauffähiges System

- Es muß Code erzeugt werden
 - Ada, Java, C++, C, ..
 - Es muß getestet werden
 - Testfälle, Test Scripte, ...
 - Es muß dokumentiert werden
 - HTML, XML, RTF, ...
 - Es muß verstanden werden
 - Graphische Darstellung
 - Wesentliches von Details trennen
- ➔ **Umsetzung** der Modelle in die Implementierung
Konsistenz der verschiedenen Entwicklungsstufen

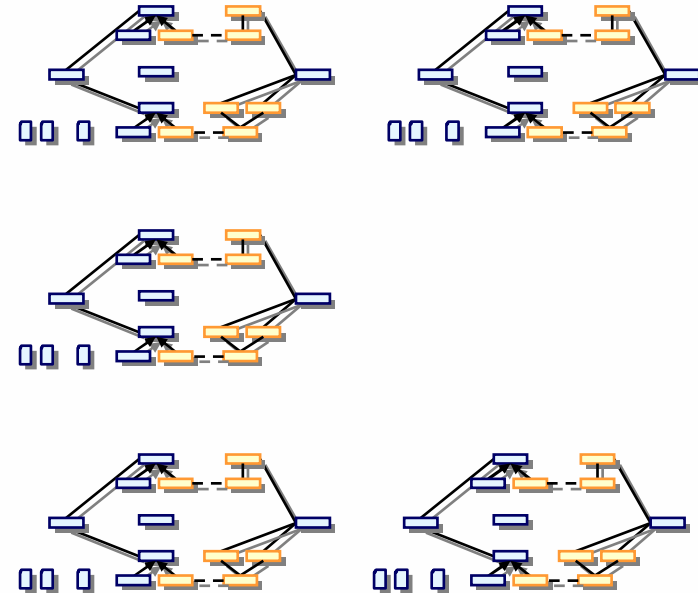
Problem beim Einsatz von UML Tools

- **Im Design wachsen Modelle oft dramatisch an**
 - Aus einem Analyse Objekt werden mehrere Design Objekte
 - Architektur/Technologie führt zu redundanten Informationen
- **Zwei Optionen für die Lösung**
 - Klares, übersichtliches Design, keine Code Generierung möglich
 - Detailliertes, unübersichtliches Design, Code Generierung möglich

Bessere Modellierung, Konzentration auf das Wesentliche



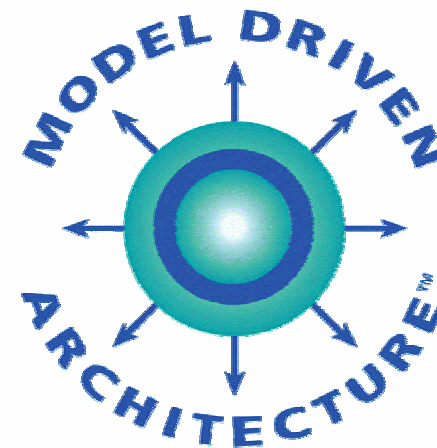
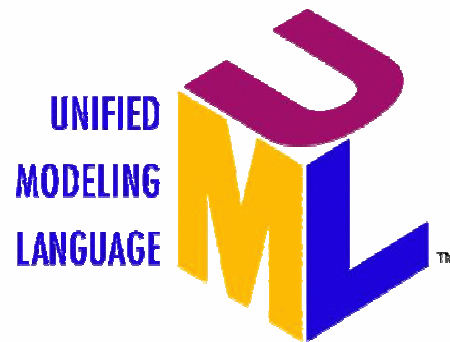
Fachlich Entwerfen



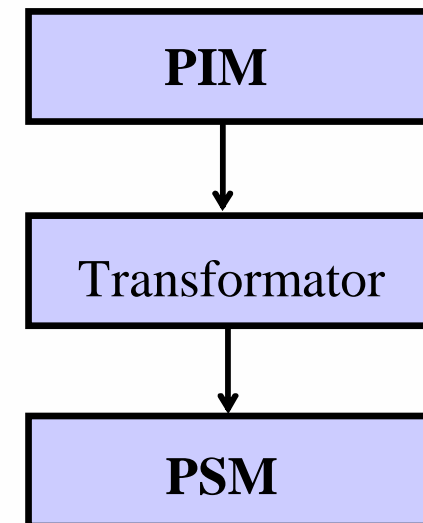
Details (weitestgehend) generieren

▷ Transformation von Modellen

<<UML + MDA>>

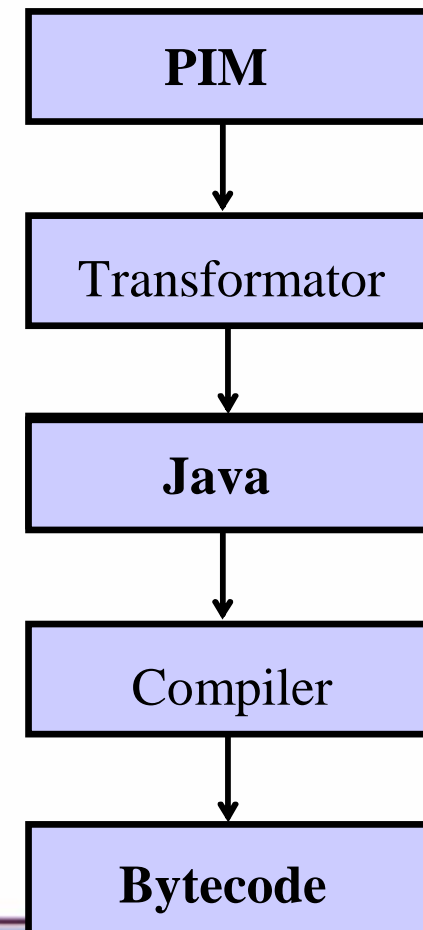


- Modelle nehmen **zentrale** Stellung im Entwicklungsprozess ein
- Erstellen von *Platform Independant Model's* (PIM)
 - Verwendung von UML Notation
 - Fokus liegt auf der Anwendungsdomäne
- Abbilden auf *Platform Specific Model's* (PSM)
 - Erweiterung des PIM durch die technische Infrastruktur
 - Durch Transformatoren
 - Maximizing automation of the mapping step is the goal (MDA Paper, OMG)
 - Von Hand
- Source Code ist ein PSM



UML und MDA

- Modelle nehmen **zentrale** Stellung im Entwicklungsprozess ein
- Erstellen von *Platform Independant Model's* (PIM)
 - Verwendung von UML Notation
 - Fokus liegt auf der Anwendungsdomäne
- Abbilden auf *Platform Specific Model's* (PSM)
 - Erweiterung des PIM durch die technische Infrastruktur
 - Durch Transformatoren
 - Maximizing automation of the mapping step is the goal (MDA Paper, OMG)
 - Von Hand
- Source Code ist ein PSM



MDA mit Profilen ist UML nach Maß

- Semantik der Modells ist abhängig von der Aufgabenstellung
 - Kaufmännische Applikationen
 - Embedded Systeme
 - Systems Engineering
- Beschreibung spezieller Sachverhalte
 - Ohne die Notation zu verändern
 - über Stereotypen, Constraints und Tagged Values



Verwendung von UML Profilen mit



- Definiert einfache Entwurfselemente und komplexe Entwurfsmuster
- Definiert Abbildungsvorschriften von PIM nach PSM

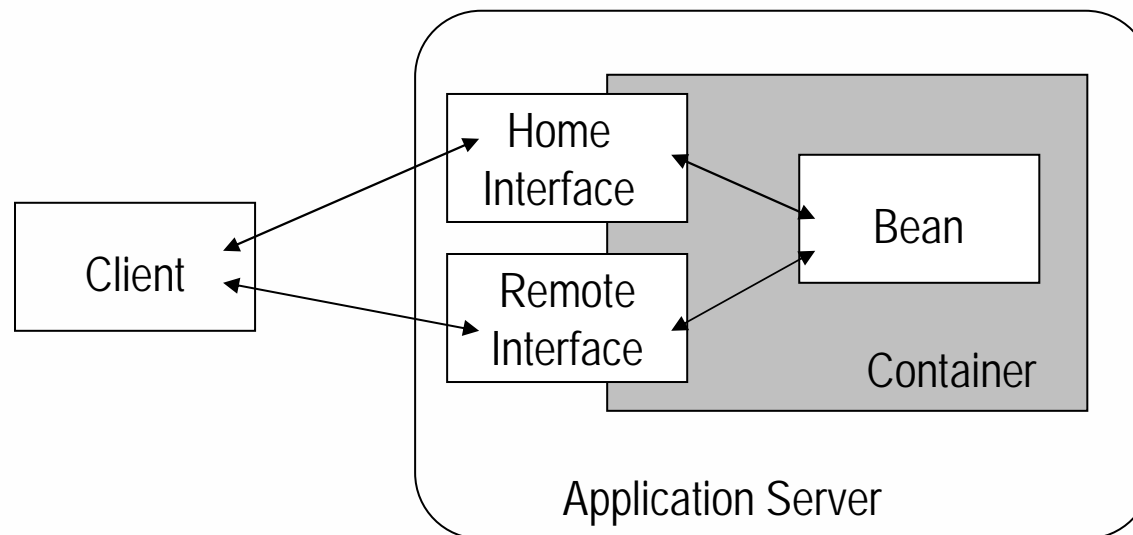


Agenda

- Aonix – wer ist das?
- Wie kam es (Aonix, OMG) zu MDA?
- **Wozu ist MDA nützlich?**
- Wo stößt MDA heute (noch) an Grenzen?
- Wie ist der Entwicklungsprozess mit MDA?
- Wie funktioniert MDA mit einem konkreten Modellierungswerkzeug Ameos/UML unter Verwendung von UML 2.0 Profilen?
- ...
- Wann gibt's den Apéro?

Beispiel EJB

- Business Objekt lebt in einem Container
- Definierte Schnittstellen nach außen

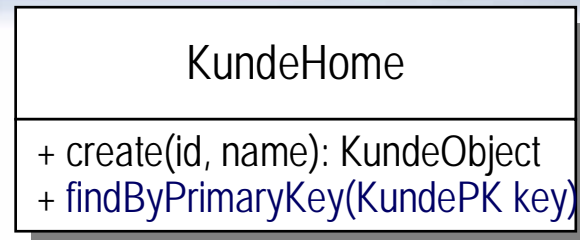




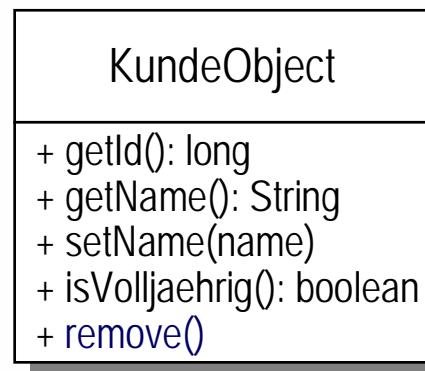
Entwicklung des EJB Modells



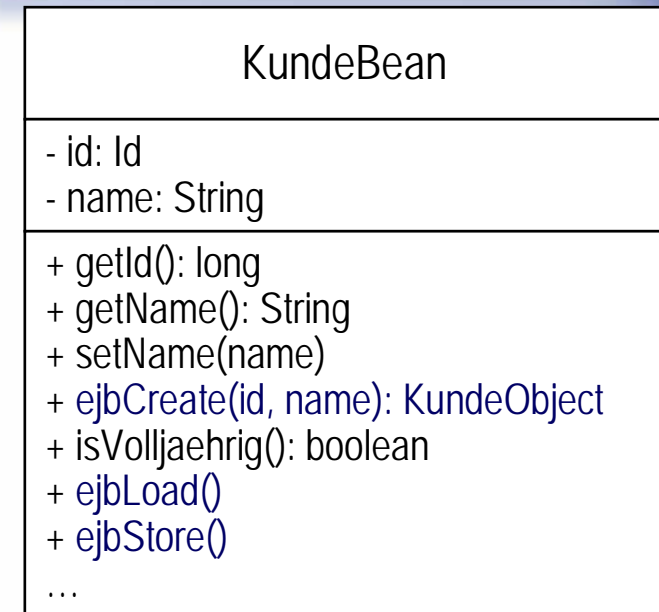
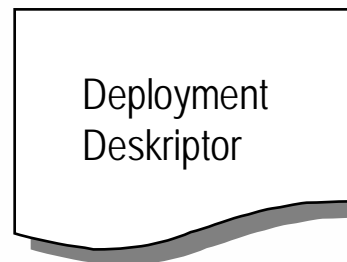
Fachliches Modell



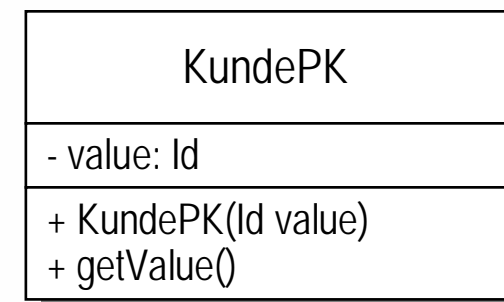
Home Interface



Remote Interface

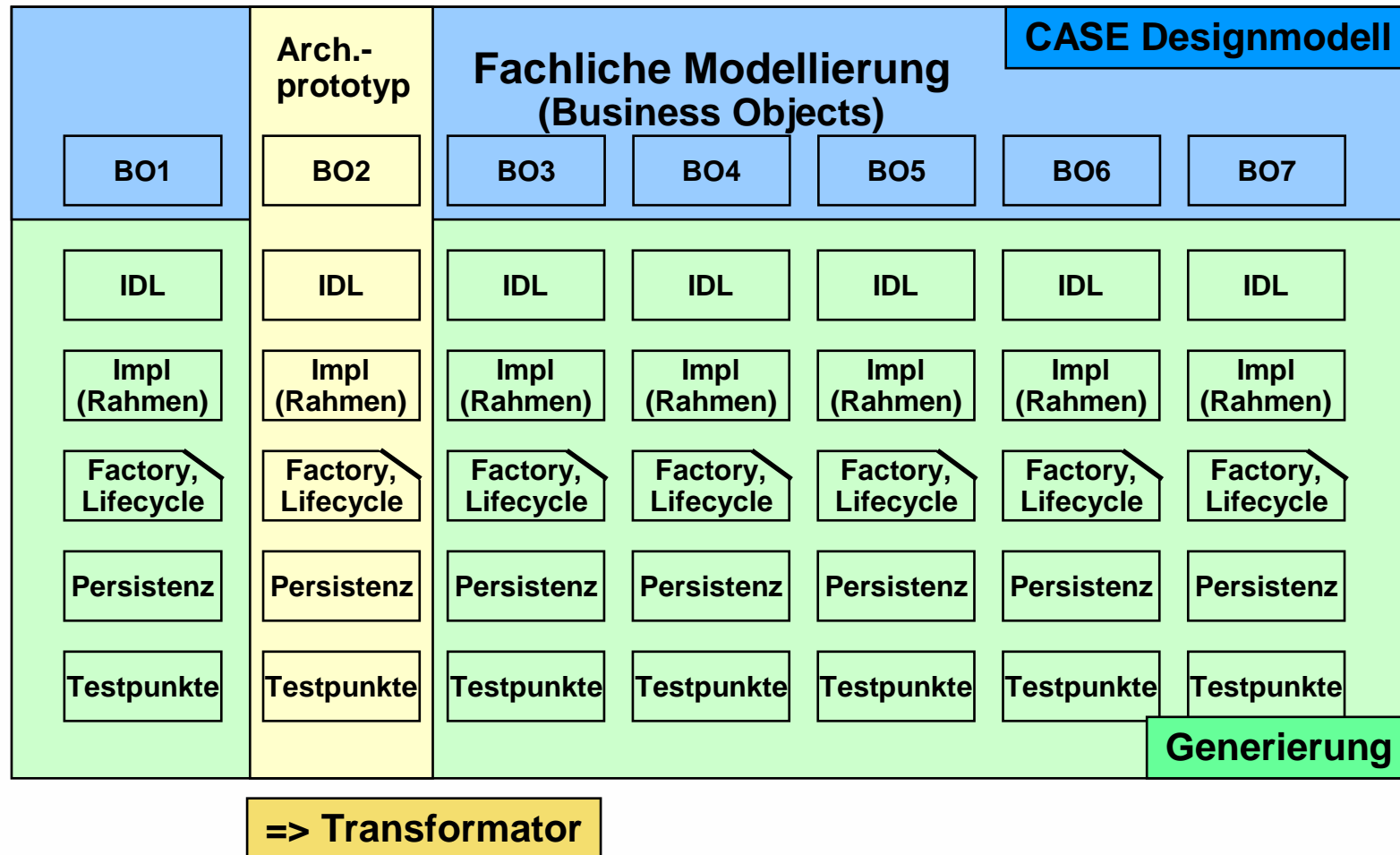


Entity Bean



Primary Key Class

Automatisiertes 'Copy&Paste'

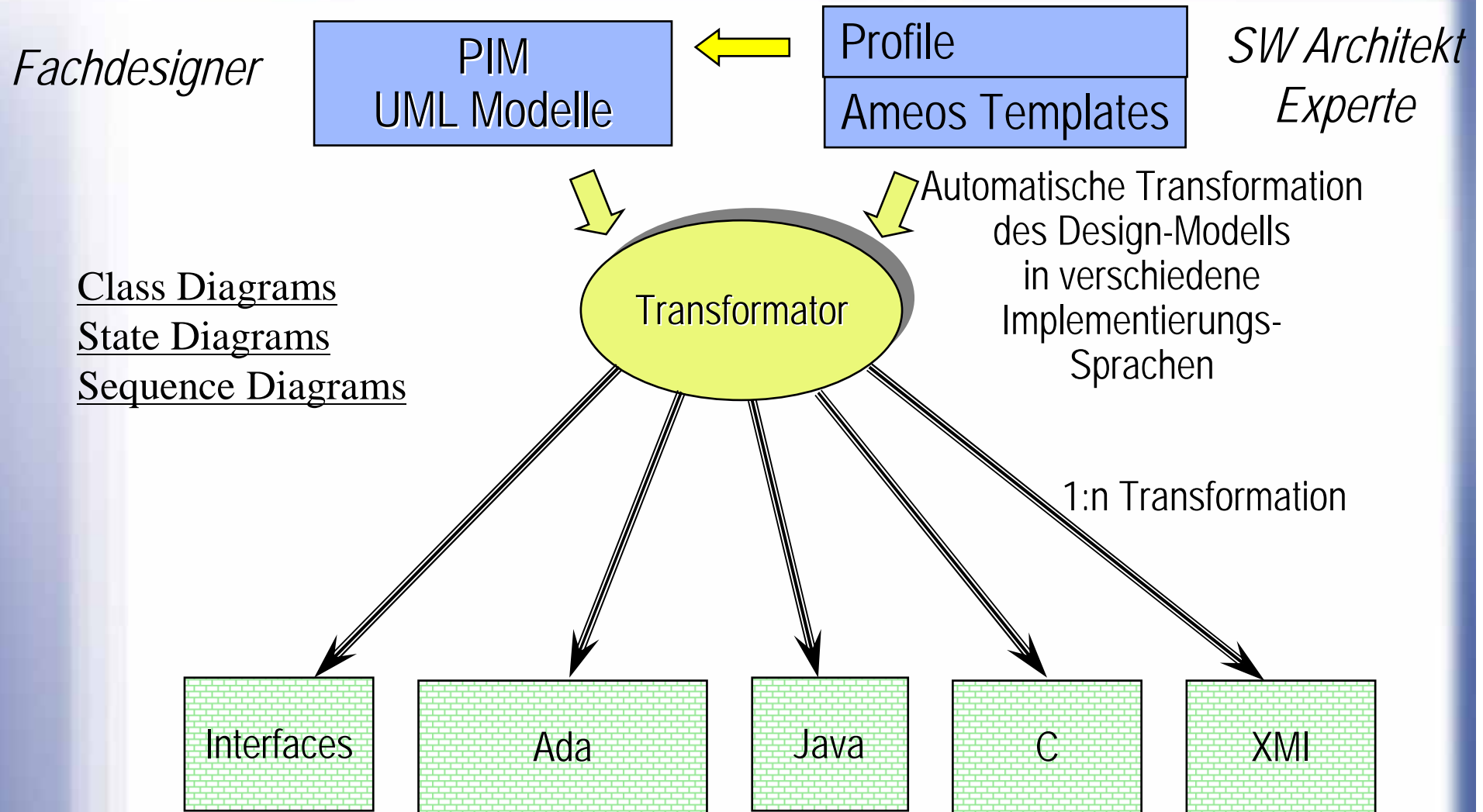




Agenda

- Aonix – wer ist das?
- Wie kam es (Aonix, OMG) zu MDA?
- Wozu ist MDA nützlich?
- **Wo stößt MDA heute (noch) an Grenzen?**
- **Wie ist der Entwicklungsprozess mit MDA?**
- **Wie funktioniert MDA mit einem konkreten Modellierungswerkzeug Ameos/UML unter Verwendung von UML 2.0 Profilen?**
- ...
- Wann gibt's den Apéro?

MDA mit Ameos/ACD



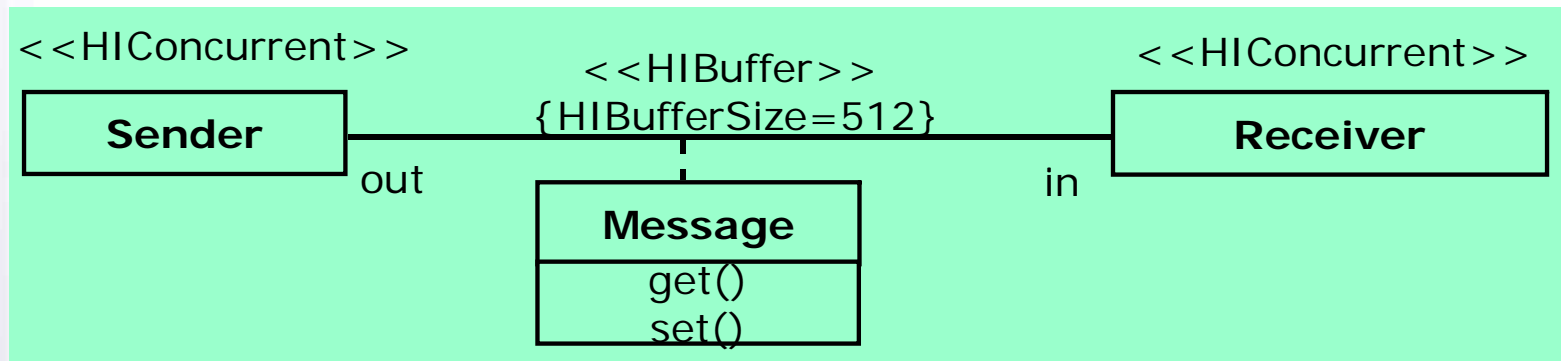


Agenda

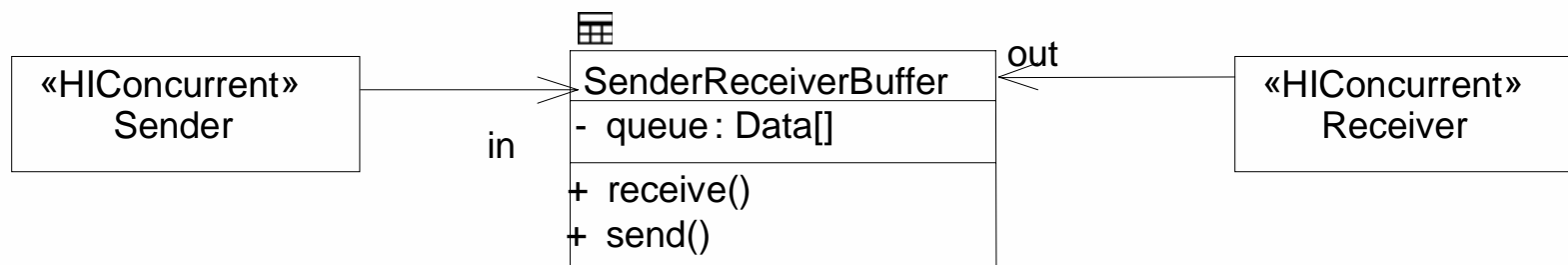
- Aonix – wer ist das?
- Wie kam es (Aonix, OMG) zu MDA?
- Wozu ist MDA nützlich?
- Wo stößt MDA heute (noch) an Grenzen?
- Wie ist der Entwicklungsprozess mit MDA?
- **Wie funktioniert MDA mit einem konkreten Modellierungswerkzeug Ameos/UML unter Verwendung von UML 2.0 Profilen?**
- ...
- Wann gibt's endlich den Apéro?

HIDOORS Communication Pattern

Abbildung einer High Level Modellbeziehung in einfachere Assoziationen (Modell Transformation => MDA)



Codegenerierung





Generated Code für Messagebuffer

```
import com.aonix.hidoors.tools.ApexBuffer;
import javax.realtime.PriorityCeilingEmulation;
public class SenderReceiverBuffer {
private static int priorityCeiling = 12;
private ApexBuffer queue = new ApexBuffer(512,
    new PriorityCeilingEmulation(priorityCeiling));

public Message receive() {
    return (Message) queue.get();
}
public void send (Message element) {
    queue.put(element);
}
```



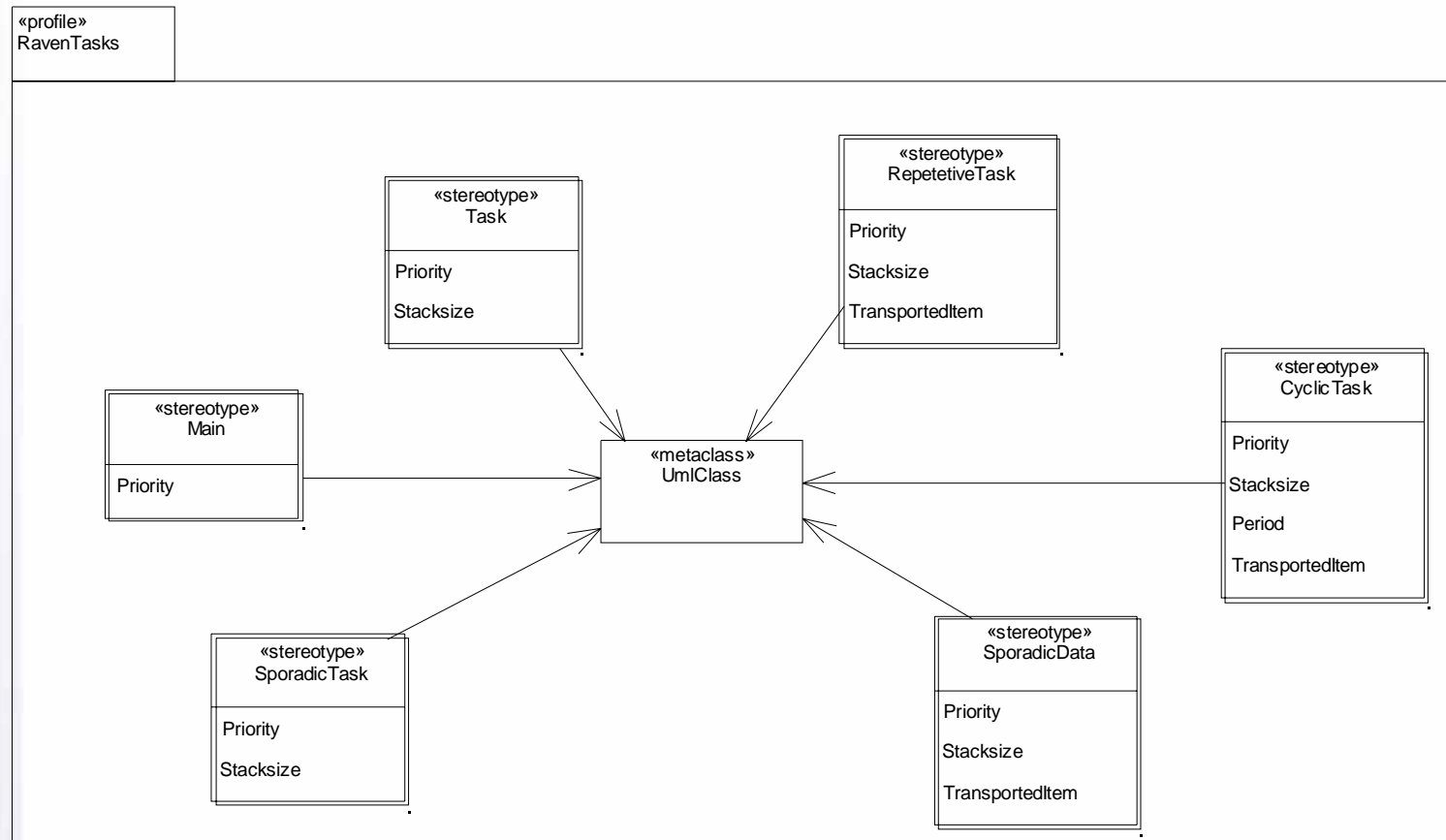
Template für MessageBuffer

```
template BufferAssoc (MAssociation)
[loop (MAssociation->AssociationStart->MClass As SenderCl)]
  [loop (MAssociation->AssociationEnd->MClass As ReceiverCl)]
import com.aonix.hidoors.tools.ApexBuffer;
import javax.realtime.PriorityCeilingEmulation;
public class [SenderCl.name][ReceiverCl.name]Buffer {
private static int priorityCeiling = 12;
  [loop (MAssociation->MAssociatedClass As MessageCl)]
private ApexBuffer queue = new ApexBuffer([getSize([MessageCl]],
  new PriorityCeilingEmulation(priorityCeiling));

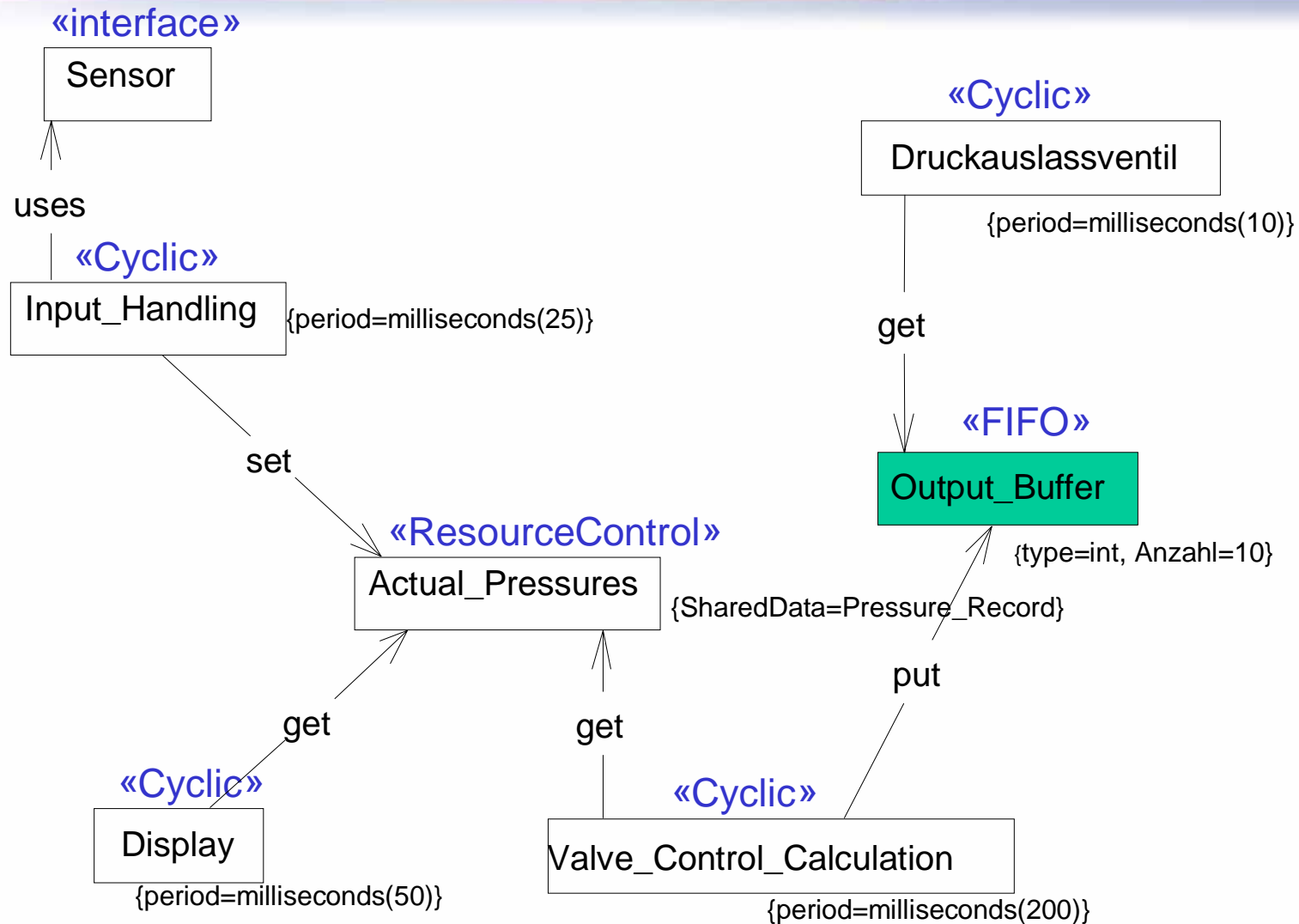
public [MessageCl.name] receive() {
  return ([MessageCl.name] ) queue.get();
}
public void send ([MessageCl.name] element) {
  queue.put(element);
}
[end loop end loop end loop]
end template
```



UML Profil Editor: RavenTasks



Beispiel „Control System“ mit Raven Profil





Der generierte C Code für OutputBuffer

```
#define number_of_elems 10
int buffer[number_of_elems];
int next_write = 0; int next_read = 0; int elems_in_buffer = 0;

int Output_Buffer_Set_Next_Item (int Item) {
    if (elems_in_buffer > number_of_elems) {
        buffer [next_write] = Item;
        next_write = (next_write++ % number_of_elems);
        elems_in_buffer++;
        return 1;
    } else return 0;
};

int Output_Buffer_Get_Next_Item (int *Item) {
    if (elems_in_buffer > 0) {
        *Item = buffer [next_read];
        next_read = (next_read++ % number_of_elems);
        elems_in_buffer--;
        return 1;
    } else return 0;
};
```



Das Template für Stereotyp <<FIFO>>

```
template FIFO_BufferC(MClass)
#define number_of_elems [getNrOfElements([MClass])
    [getType([MClass])] buffer\[number_of_elems];
    int next_write = 0; int next_read = 0; int elems_in_buffer = 0;

int [ClassName([MClass])]._Set_Next_Item ([getType([MClass]) Item) {
    if (elems_in_buffer < number_of_elems) {
        buffer [next_write] = Item;
        next_write = (next_write++ % number_of_elems);
        elems_in_buffer++;
        return 1;
    } else return 0;
};

int [ClassName([MClass])]._Get_Next_Item ([getType([MClass]) *Item) {
    if (elems_in_buffer > 0) {
        *Item = buffer [next_read];
        next_read = (next_read++ % number_of_elems);
        elems_in_buffer--;
        return 1;
    } else return 0;
};

end template
```



Templategesteuerter Codegenerator vs herkömmlicher prozeduraler Codegenerator

- Vergleich C++ Generator (LOC)

