

Designing Adaptive Web Applications

Peter Dolog

dolog@cs.aau.dk

CS Department

Information Systems



Outline

Motivations and Applications

Adaptive Web Application Design

- Information Access Applications
- Service Based Applications

Further Challenges

Outline

Motivations and Applications

Adaptive Web Application Design

- Information Access Applications
- Service Based Applications

Further Challenges

Adaptation/Customization

- Customization by humans (designers)
- Dynamic adaptation by a system itself
- Adaptation is about decision on which information resource or function variant to provide or recommend access to,
- We need a knowledge to decide about appropriate information or service configuration in a certain processing step (user or other):
 - Resource and information access environment
 - Application domain
 - User/Context
 - And their configuration – variants and their meaningful combinations for certain purposes

The UML-Guide

Adresse <http://localhost:8080/uml/course.jsp> Wechseln zu Links >>

Navigation Map

- Start
- Language Basics
- Object Basics and Simple Data Objects
- Classes and Inheritance
- Interfaces and Packages
- Overview
- Object Oriented Programming Concepts
 - Language Basics
 - StartOO
 - What Is an Object**
 - What Is a Message/next
 - What Is a Message
 - What Is a Class
 - Relations to Code
 - Questions
 - EndOO
 - Common Problems
 - End

:.Overview :.Lecture Modules

Previous Next

LECTURE MODULE: What is an Object?

Objects are key to understanding object-oriented technology. Software objects are modeled according to a state and a behavior. A software object maintains its state in one or more variable.

A variable is an item of data named by an identifier. A software object implements its behavior with methods . A method is a function (subroutine) associated with an object.

The following illustration is a common visual representation of a software object:

Methods (behavior) Variables (state)

DEFINITIONS:

Fertig Lokales Intranet

Exercises

- Exercise1: [Object definition](#)

Examples

- Example1
- Example2

Personal Learning Assistant (WWW2004)

The screenshots illustrate the workflow of the ELENA Personal Learning Assistant:

- Initial Search Page:** The user selects a user profile (e.g., 'Michael') and enters a concept name (e.g., 'intelli').
- Concept Selection:** The user chooses one or more concepts from an ontology, such as 'Distributed artificial intelligence'.
- Query Results:** The system displays a table of personalized recommendations based on the selected concepts.

Preco	Reco	Title	Description	Concepts
■	■	Aufgaben zum Thema Intelligente Agenten	Aufgaben, um den Stoff des Moduls zu vertiefen	Intelligent Agents
■	■	Einige Fragen zum Thema Intelligente Agenten	Fragen, die Ihnen helfen sollen, den Stoff besser zu verstehen	Intelligent Agents
■	■	Vorlesung Künstliche Intelligenz WS 2002 - Stichworte zum Thema Umgebungen	Was stellen die verschiedenen Grundtypen Intelligenter Agenten vor und ihre prinzipielle Programmierung	Intelligent Agents
■	■	Weiterführende Materialien	Eine Sammlung von weiterführenden Inks zum Thema Künstliche Intelligenz und Intelligente Agenten	Special-purpose; Intelligent Agents

Personal Reader (AH2004)

The screenshot displays the 'Personal Reader' web application. The main interface is divided into a sidebar and a main content area. The sidebar contains navigation sections: 'PERSONAL READER' (Peter Dolog), a search box, 'Title' (listing 'Java Tutorial'), 'Generalizations', 'Details' (with checkboxes for topics like 'Object Basics and Simpl...', 'Object-Oriented Program...', 'Learning the Java Lecture', 'Classes and Inheritance', 'Interfaces and Packages', and 'Custom Packages and The...'), 'Summaries', 'Exercises', and 'Again'. The main content area shows 'The Java Tutorial' with a search bar and a list of trail types: 'Basics | GUIs | Specialize'. Below this, there are sections for 'Trails Covering the Basics:' and 'Trail on Constructing GUIs:'. An inset window titled 'ELENA Personal Learning Assistant' is overlaid on the main content. It features the ELENA logo, the text 'Personal Learning Assistant for SMART SPACE FOR LEARNING' by Peter Dolog & Michael Sestak, and the Information Society Technologies logo. The inset window shows a 'Personalized Search Service' interface with a search bar, a 'Query results:' section, and a table of results.

Preco	Reco	Title	Description	Concepts
	•	JLayeredPane (Java 2 Platform SE v1.4.2)	?	http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Container; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#JComponent; int; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Integer; Numbers
	•	LongBuffer (Java 2 Platform SE v1.4.2)	?	http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Array; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Long; byte; if; new Operator; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#this; http://webbase.learninglab.uni-

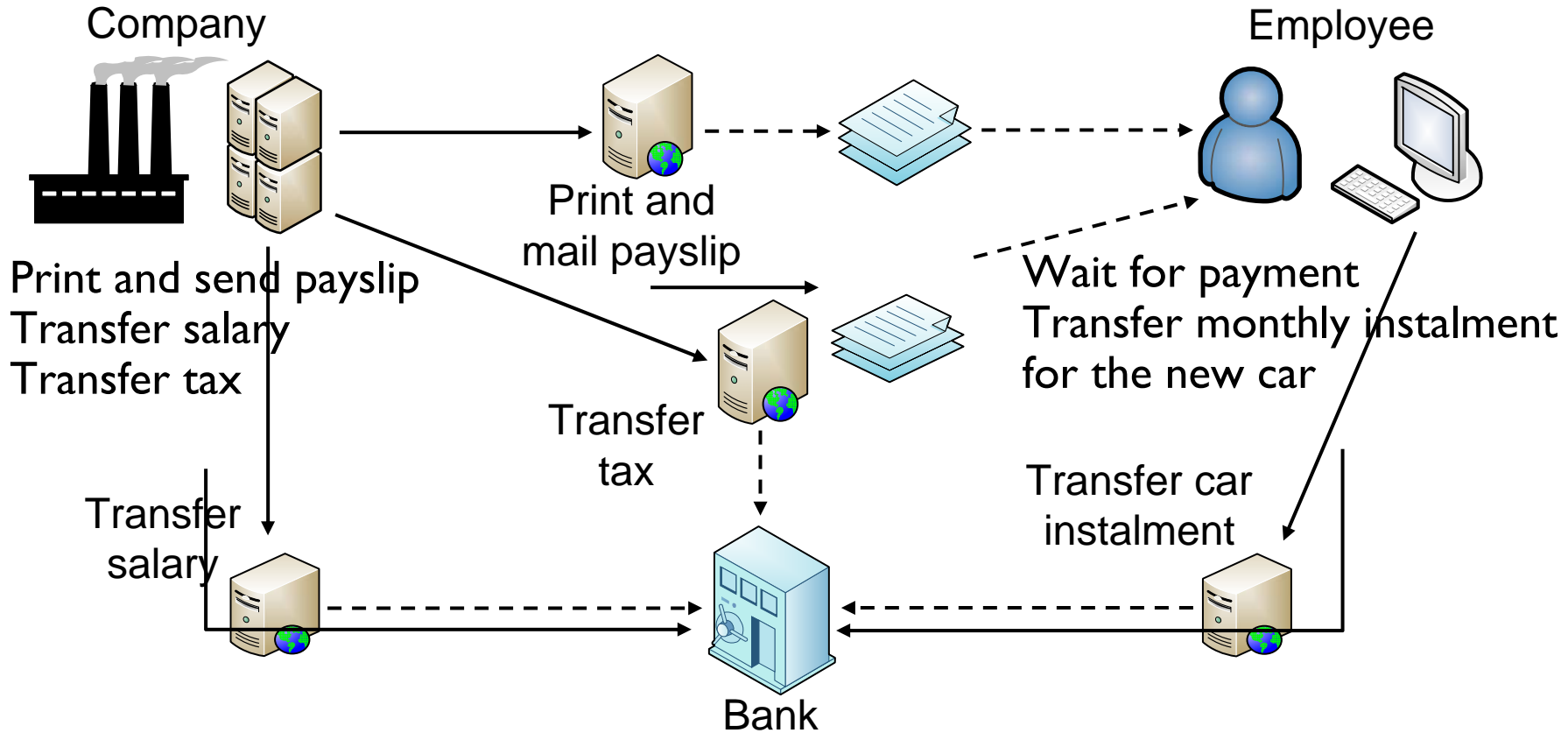
http://www.hcd-online.com (ACMTOIT 2008)

Search results are currently collected. Next update in 2 seconds.

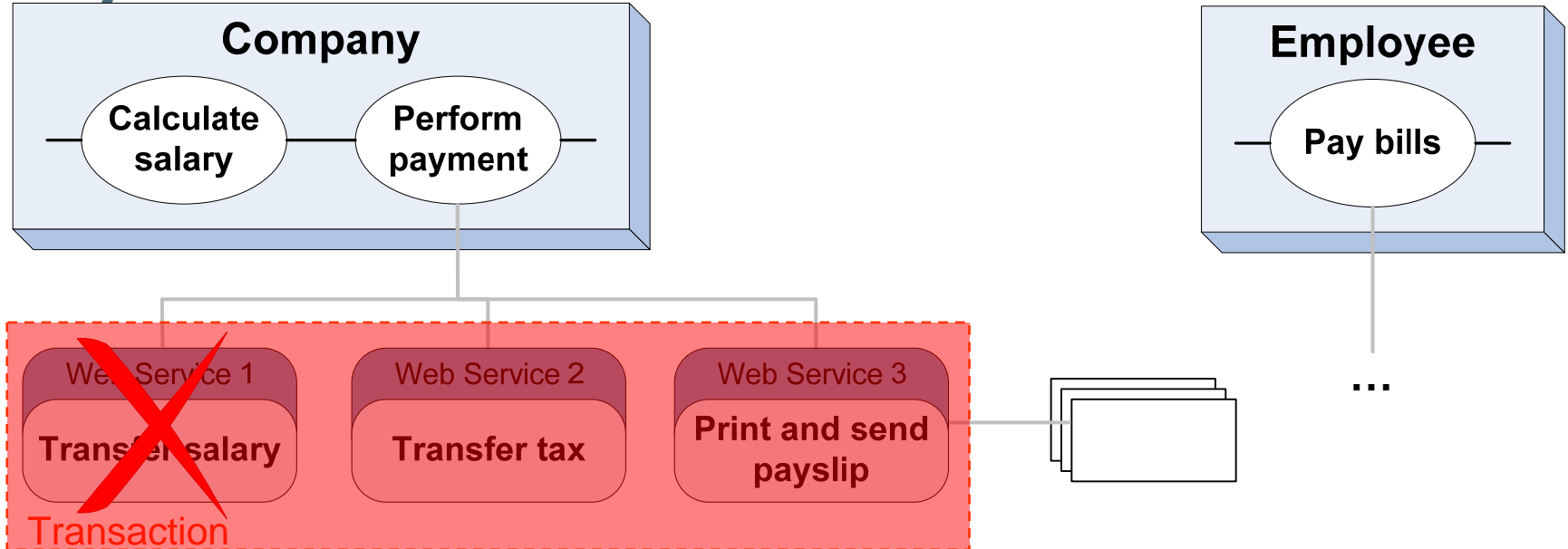
Currently used search-term(s): **economy**

Select	Relevance	Title / Description	Remote service status:		
<input type="radio"/>	100. (1)	Change-Management und Innovation Die Zukunft (die sogenannte □Next Economy□) ist eine Innovations□Economy. Nur wer sich in Zukunft auf Ver□nderungen einstellt, wer hohe Innovationsraten verwirklichen kann, wird am Markt bestehen. Ein ...	Executive Academy (WBZ): ✓	LASON: ✗	EduSource: ✓
<input type="radio"/>	2.70 (2)	The Experience Economy: Work Is Theatre & Every Business a Stage Availability: Usually ships in 24 hours	EduNext-UPM: ✗	Edutella: ✗	Seminarshop.com: ✓
<input type="radio"/>	2.70 (3)	Book of Common Prayer (1979, Personal Size Economy, Black) Availability: Usually ships in 24 hours	CLIX: ✗	ULI: ✗	Knowledgebay: ✗
<input type="radio"/>	2.70 (4)	Illicit : How Smugglers, Traffickers and Copycats are Hijacking the Global Economy Availability: Usually ships in 24 hours	Metzingen VHS-Kursdatenbank: ✓	Amazon: ✓	bfi-vienna: ✗
<input type="radio"/>	2.70 (5)	Basic Economics: A Citizens Guide to the Economy, Revised and Expanded Availability: Usually ships in 24 hours	Yes	Amazon	English
<input type="radio"/>	2.70 (6)	Twilight in the Desert: The Coming Saudi Oil Shock and the World Economy Availability: Usually ships in 24 hours	Yes	Amazon	English
<input type="radio"/>	2.70 (7)	The Macro Economy Today with DiscoverEcon with Solman Videos Availability: Usually ships in 2 to 5 weeks	Yes	Amazon	English
<input type="radio"/>	2.36 (8)	The Travels of a T-Shirt in the Global Economy : An Economist Examines the Markets, Power, and Politics of World Trade Availability: Usually ships in 24 hours	Yes	Amazon	English
<input type="radio"/>	2.02 (9)	How We Compete : What Companies Around the World Are Doing to Make it in Today's Global Economy Availability: Usually ships in 24 hours	Yes	Amazon	English
<input type="radio"/>	2.02 (10)	The Wal-Mart Effect : How the World's Most Powerful Company Really Works--and How It's Transforming the American Economy Availability: Usually ships in 24 hours	Yes	Amazon	English
<input type="radio"/>	1.88 (11)	3043 Advanced Novell Network Administration NetWare 6.5 NetWare 6.x wurde speziell auf die Bed□rfnisse der heutigen Net Economy zugeschnitten. NetWare 6.x kann in bestehenden Netzwerken eingesetzt werden, um diese in ein einziges, alles umfassendes Netz - ...	Yes	Seminarshop.com	German
<input type="radio"/>	0.0 (12)	China Reise Taijiqian Qi Gong 21 Tage China Reise - Taijiqian / Qi Gong Unterricht an der Sportuniversit□t Peking, Rundreise zu faszinierenden St□tten der chinesischen Kultur	Yes	Seminarshop.com	German
<input type="radio"/>	0.0 (13)	Szenario-Technik Jeder Unternehmer, jede F□hrungskraft ist t□glich immer wieder neu mit der Frage konfrontiert□- Wie k□nnte mein Unternehmen/mein Bereich in ca. 1-3 Jahren aussehen?- Welche Anforderungen k□nnte der Ku ...	Yes	Seminarshop.com	German
<input type="radio"/>	0.0 (14)	The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations	Yes	Amazon	English

Payroll Scenario (ICWE2007)



Payroll Scenario



A service fails due to an internal error.

The error can traditionally be compensated by aborting the complete transaction (all or nothing semantics).

Adaptation as replacement: a different service exists that can perform the same operations, so that the failed one can be replaced

General Problems to Deal with

What to Adapt

According to what to Adapt

How to adapt

How to Manage the Adaptation at a Product Level

Outline

Motivations and Applications

Web Application Design

- Information Access Applications
- Service Based Applications

Further Challenges

Work Products of Web Application Design

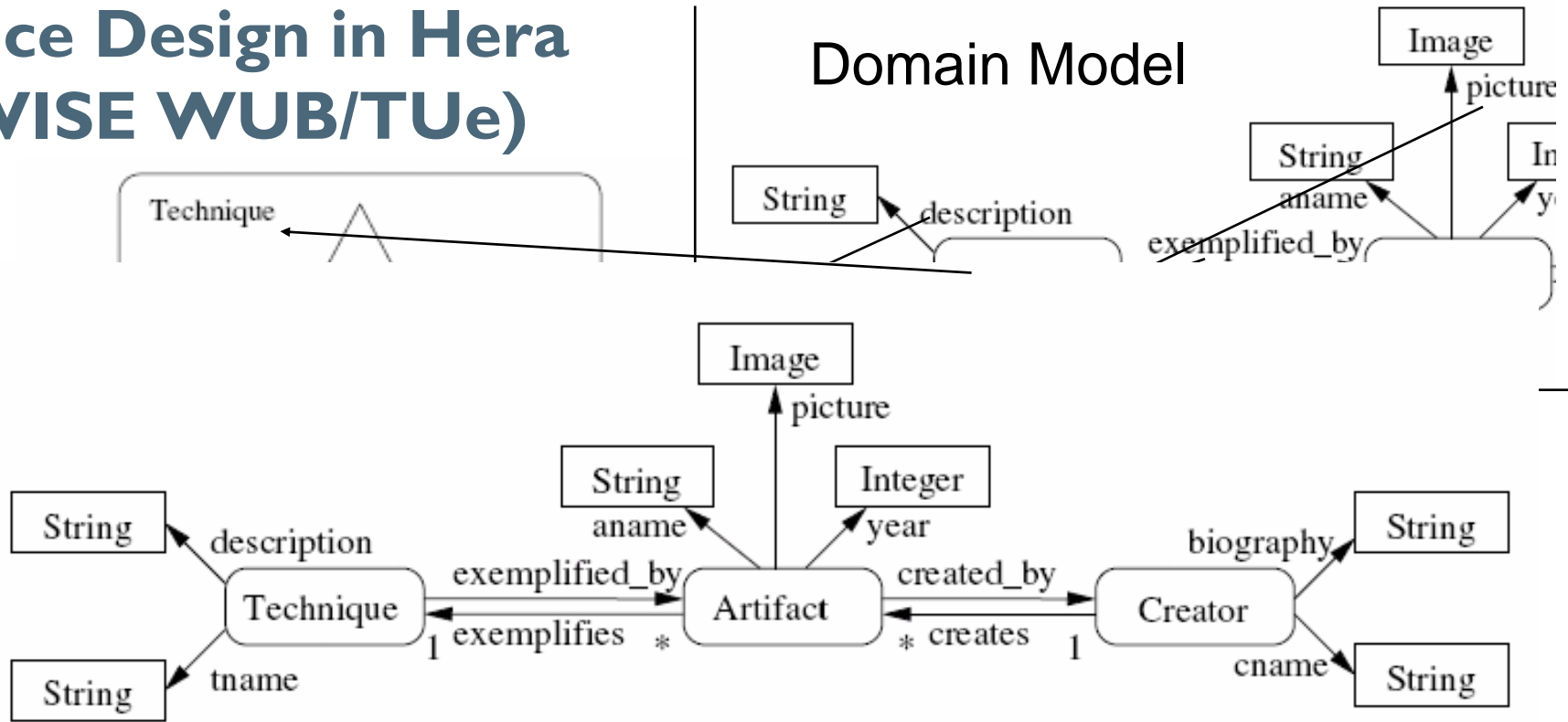
Content -> Application Domain

Navigation -> Information Context

Presentation -> Placement and Appearance

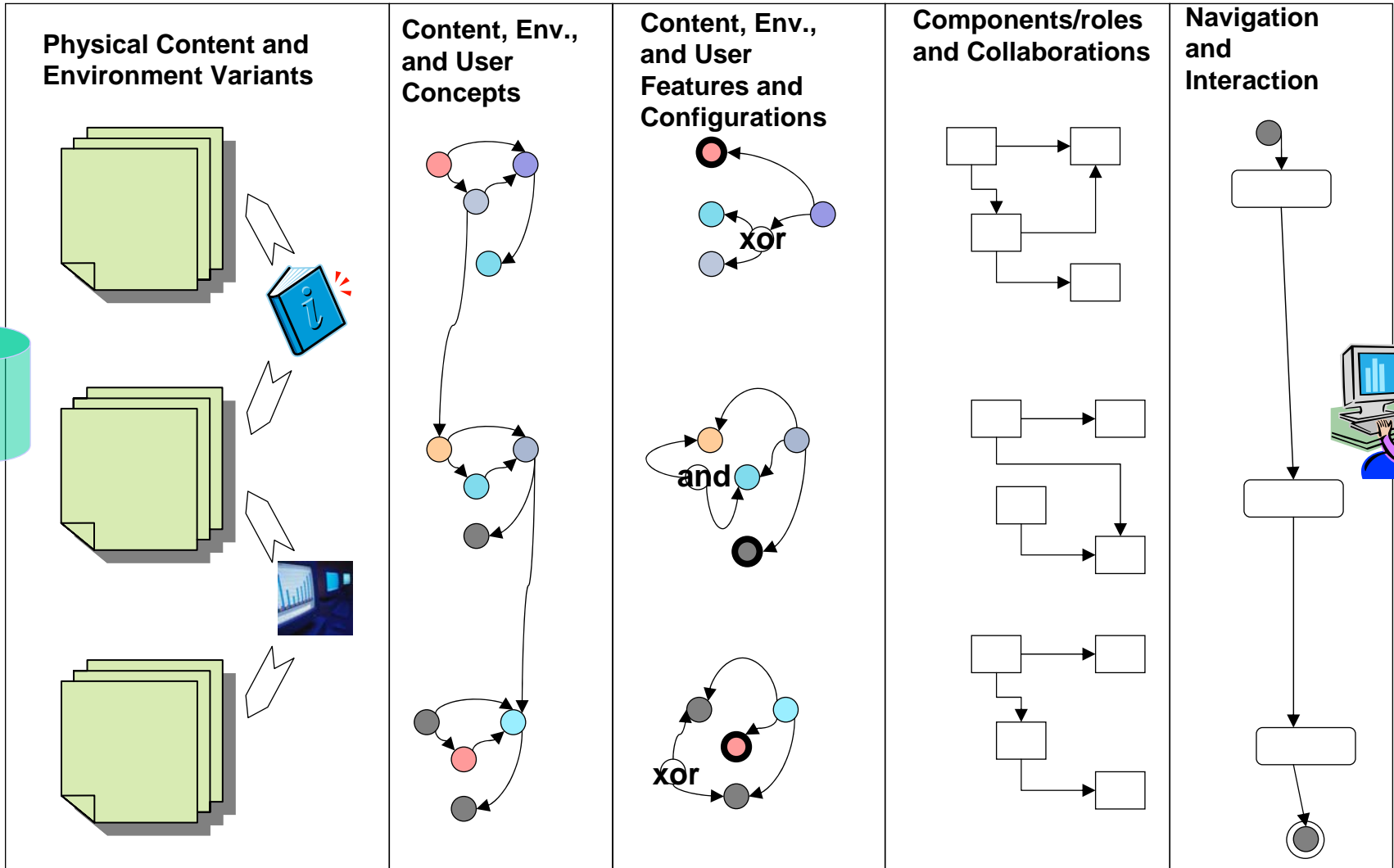
Slice Design in Hera (WISE WUB/TUe)

Domain Model



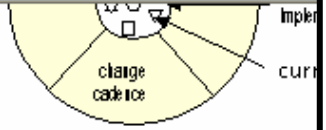
- property
- aggregation (with CM property name)
- condition

Layers of Abstraction



What Is a Class? - Microsoft Internet Explorer - [Offlinebetrieb]

Adresse <http://java.sun.com/docs/books/tutorial/java>

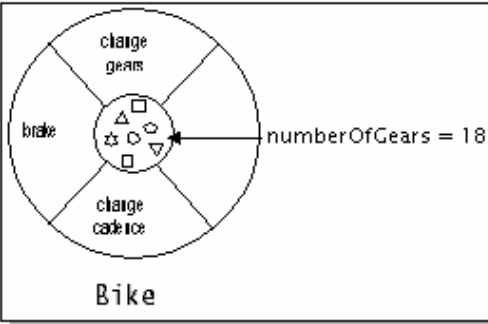


MyBike

Application Domain Concepts:

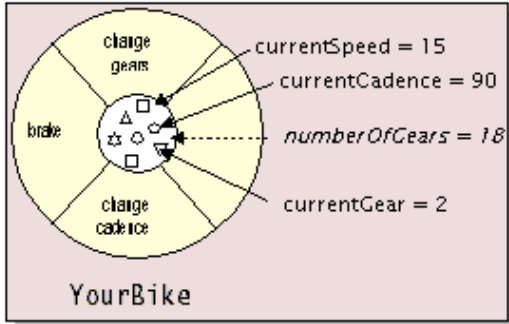
- Class – the main goal,
- Class Variables,
- Class Methods

In addition to instance variables, classes can define **class variables**. A class variable contains information that is shared by all instances of the class. For example, suppose that all bikes had the same number of gears. In this case, defining an instance variable to hold the number of gears is inefficient, because each instance would have its own copy of the variable, but the value would be the same for every instance. In such situations, you can define a class variable that contains the number of gears. All instances share this variable. If one object changes the variable, it changes for all other objects of that type. A class can also declare **class methods**. You can invoke a class method directly from the class, whereas you must invoke instance methods on a particular instance.



Bike

Class



YourBike

Instance of a Class

Fertig Internet

Building an Application: Part 3: Receiving Input with Text Fields and Check Boxes, and Event Ha - Microsoft Internet Explorer

Adresse <http://java.sun.com/developer/onlineTraining/new>

Contents
[BACK](#) << [Constructor Completion](#) | Instance and Class Members

Why didn't you have to create an instance of the method?

If a field or method is `static`, you do not need a method, followed by the dot operator and the name of the class.

Instance and Class Members

When you created the `name` and `street` variables in order to use these variables, you created instances of the `Diver` class, `street`.

Sometimes you don't want each instance to use a class variable, or rather a `static` variable.

- To force an object to share data rather than having its own copy. For instance, a counter might be shared by several class instances. Every time a new instance of a class is initialized, the counter increments by 1, sharing that data with each instance of the class. If you incorrectly used an instance variable instead, then each new class instance might reset the counter variable to 0, count the initialization as 1, and always give the incorrect count as 1, instead of however many initializations there really are.
- To initialize data before or as an object is created.

```
private String name = "";
private String id = "";
static String co = "Sun Microsystems";
```

Employee Instance **Employee Instance**

```
name = "Dana Nourie";
id = "555";
```

```
name = "Jenny Pratt";
id = "55A";
```

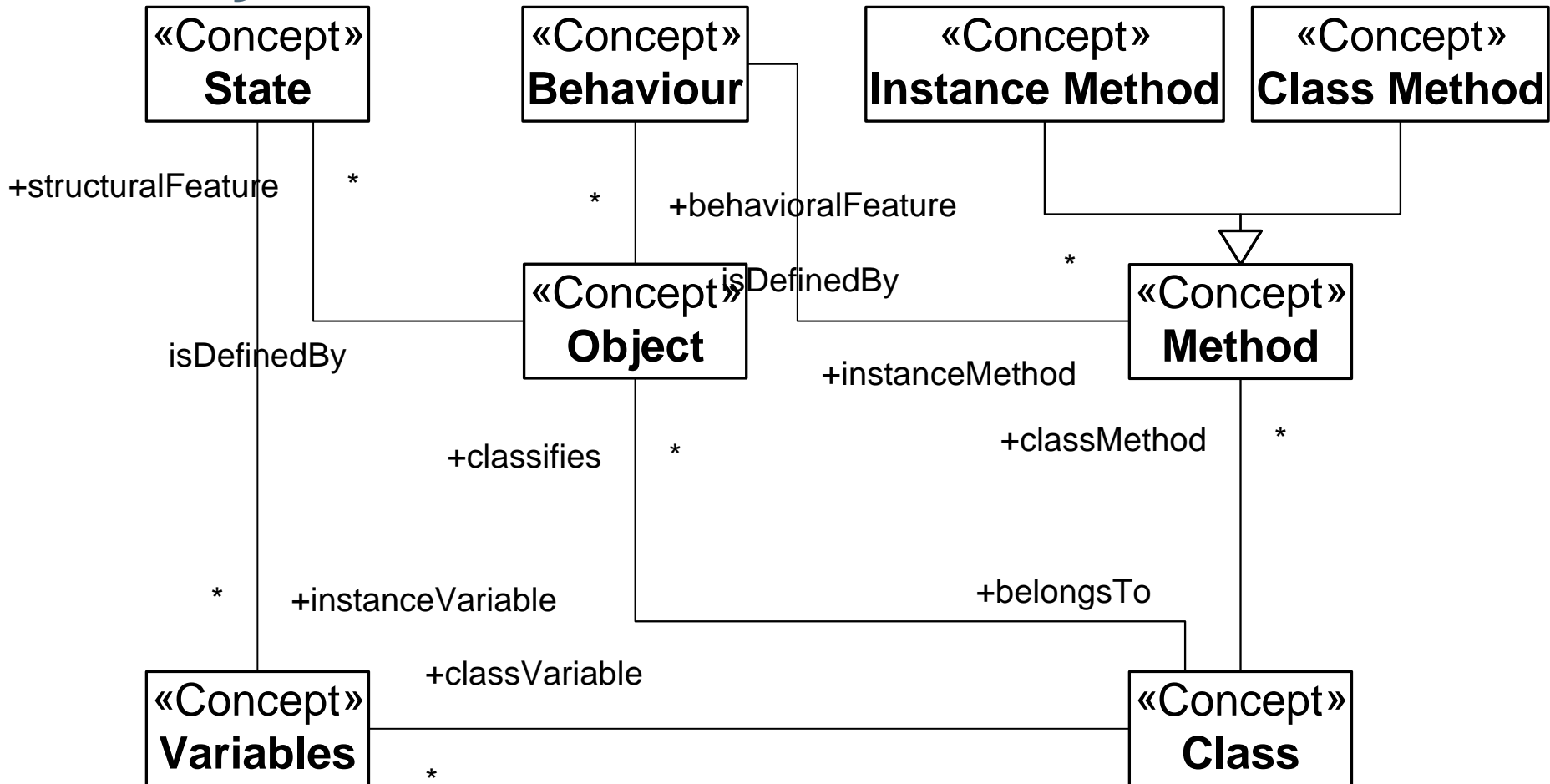
Sun Microsystems

Internet

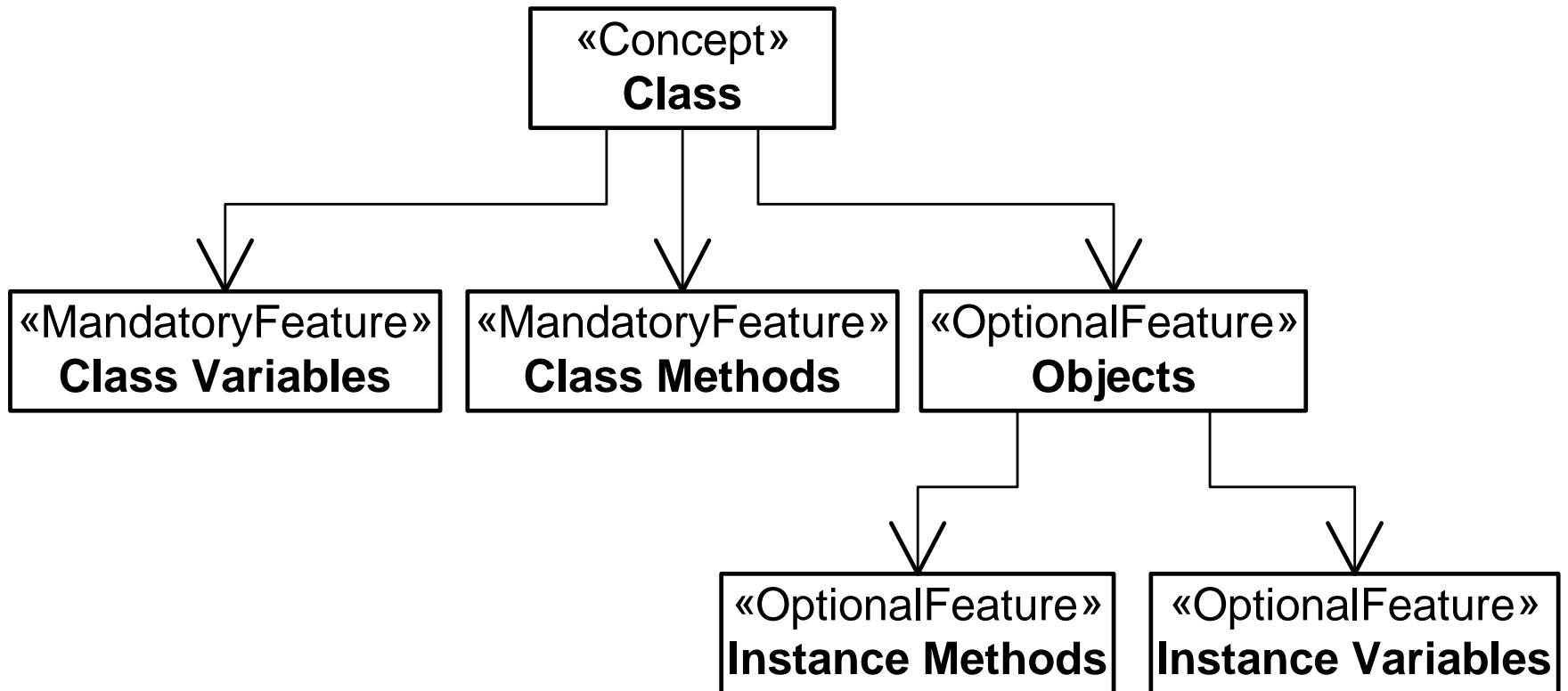
Application Domain Concepts:

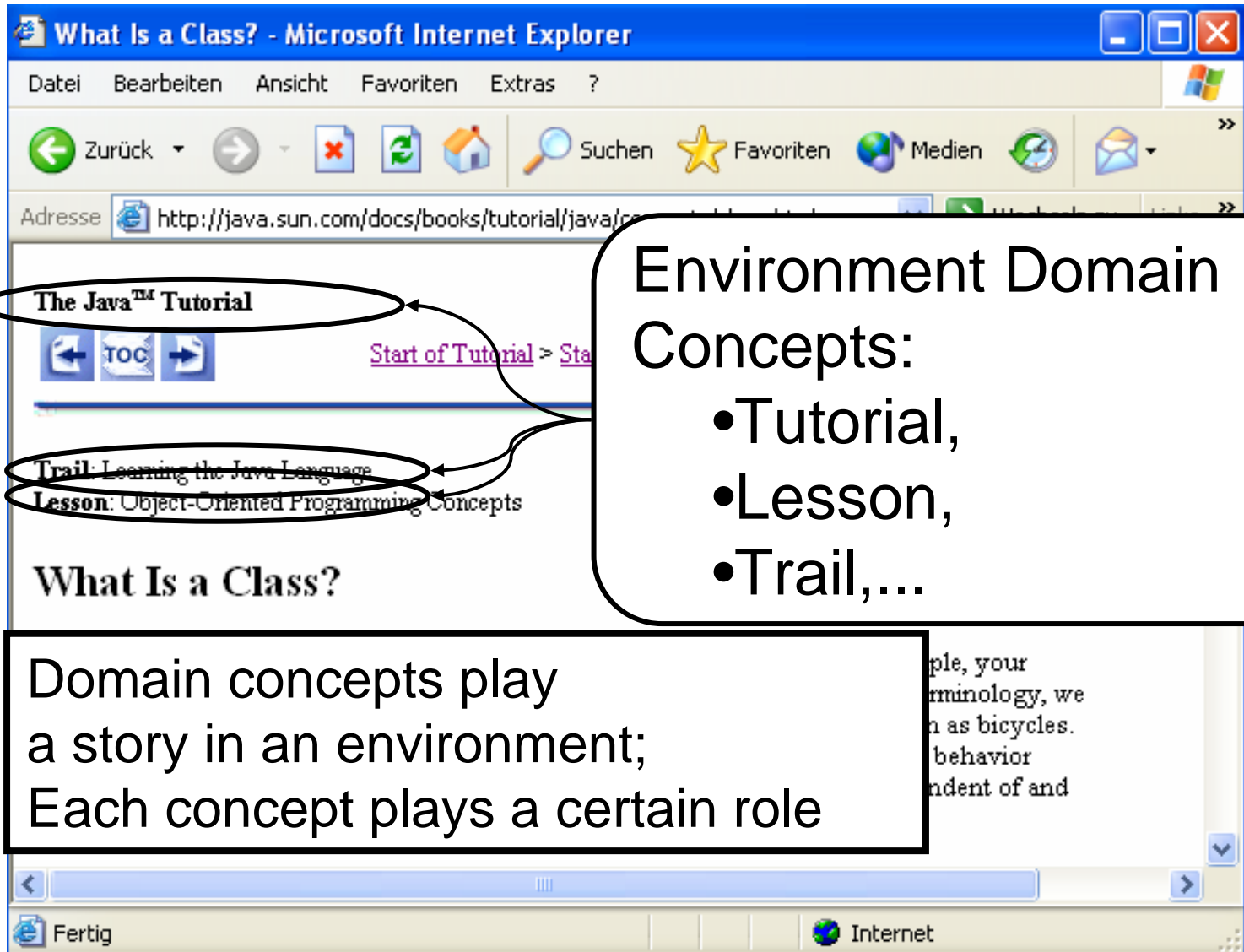
- Common:
 - Class
 - Static Variables
- Variable:
 - Instance
- Dependency

An Excerpt of Domain Conceptual Model for OOP/Java



An excerpt of *Class* feature model





Environment Domain Concepts:

- Tutorial,
- Lesson,
- Trail,...

Domain concepts play a story in an environment; Each concept plays a certain role

Script

discuss

Module

Basis-inform-ationen

einige Fragen

weiter-führende Materialien

Aufgaben

Einführung KI und Intelligente Agenten

Syntax und Semantik

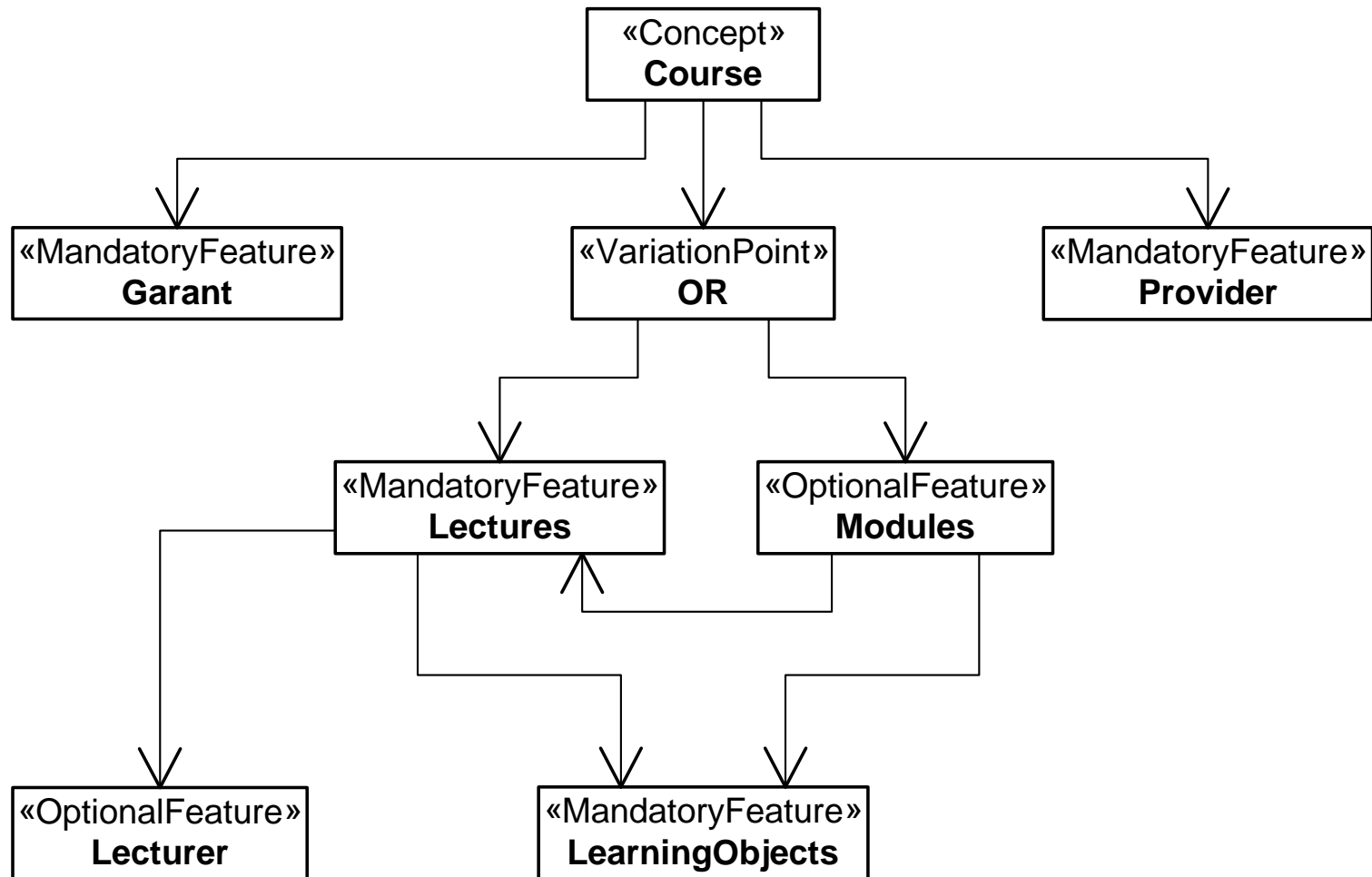
Environment Domain Concepts:

- Common:
 - Course,
 - Module,
 - Script
- Variable:
 - Exercise,
 - Discussion

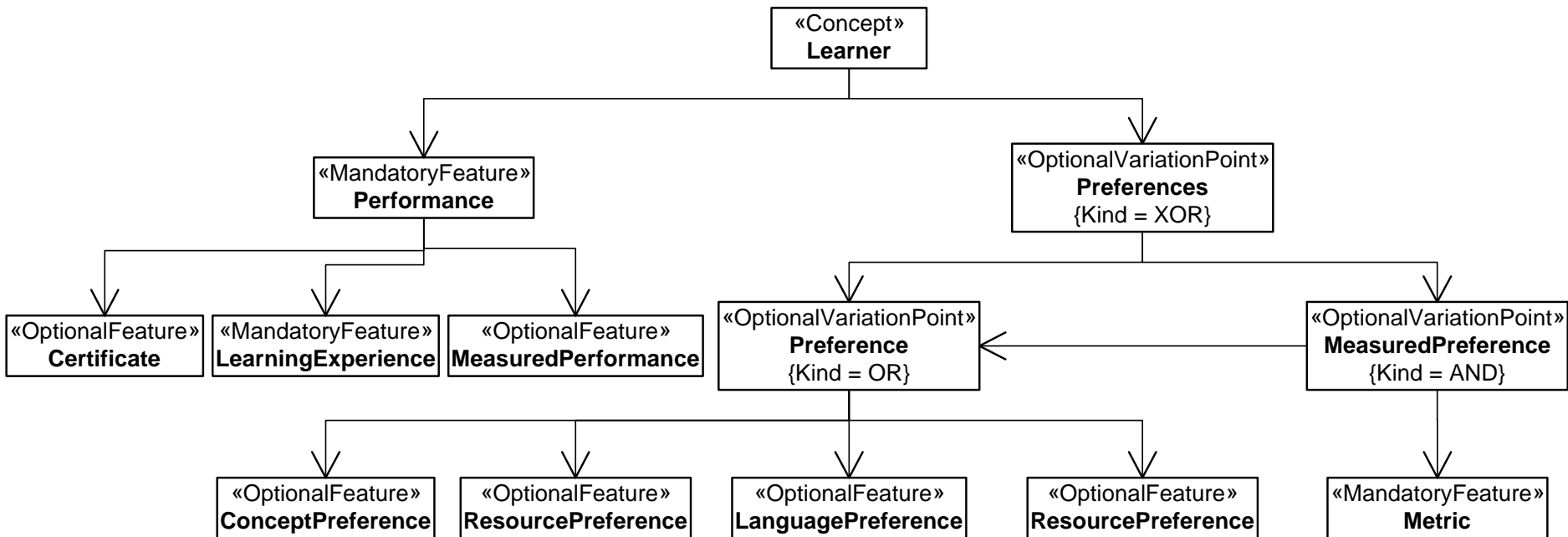
• "Is it an Agent, or just a Program?"

• [Agents Group](#)

An Excerpt of Course Feature Model

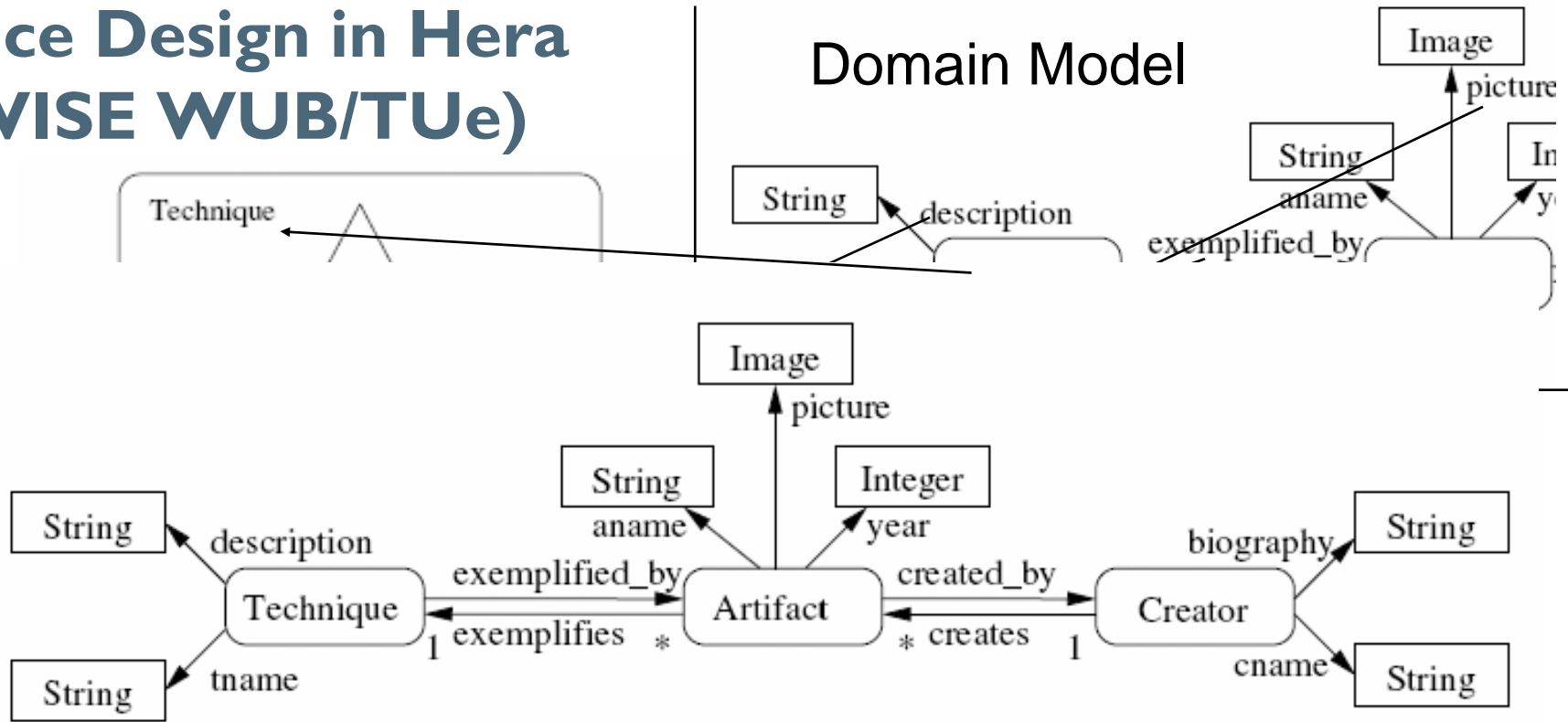


User Feature Model



Slice Design in Hera (WISE WUB/TUe)

Domain Model



- property
- aggregation (with CM property name)
- ⋯ condition

Variability at Run-Time

Variability at the design time – subgraph of the feature model

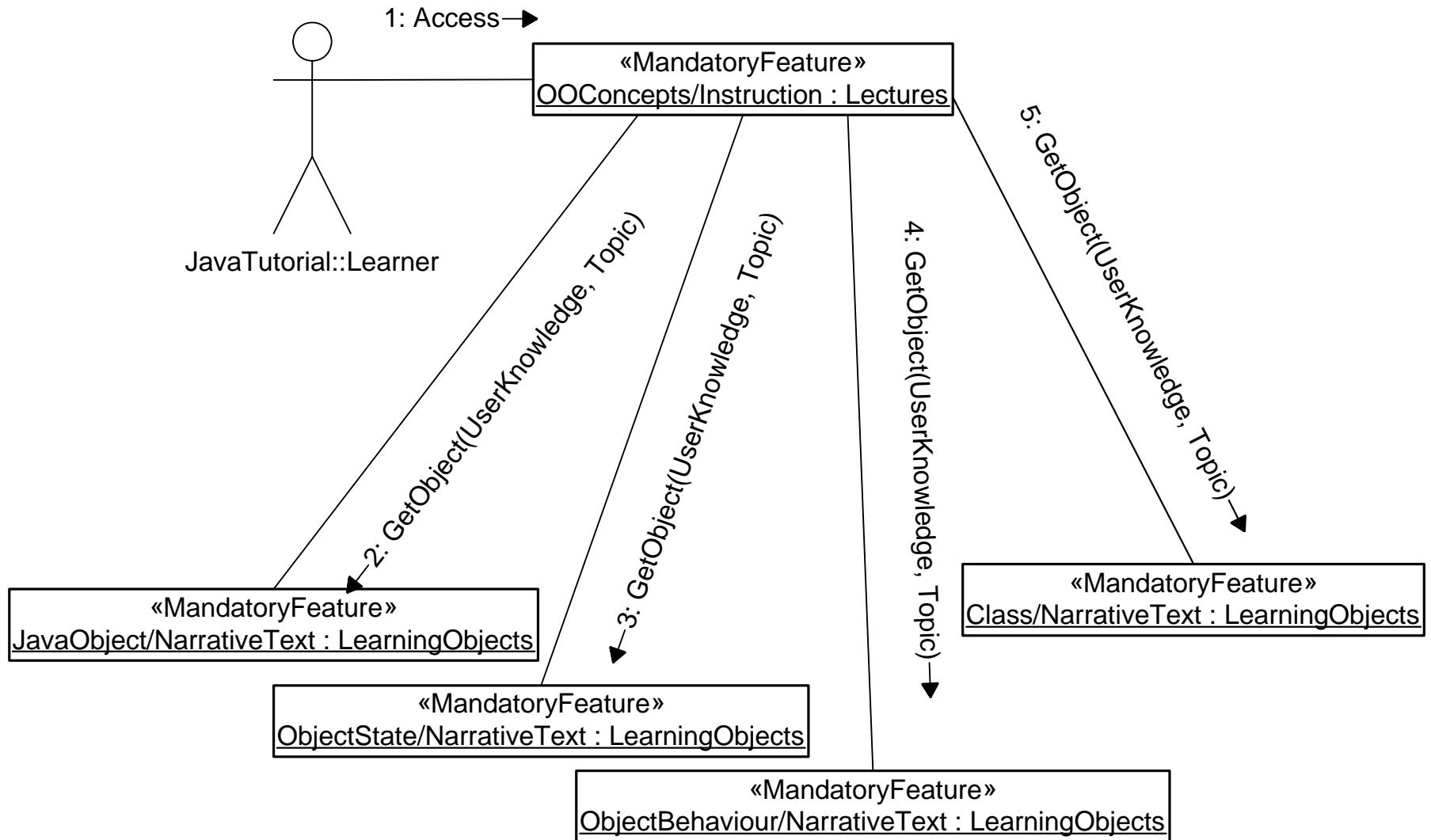
Run-Time – selecting target information according to features which are evolving dynamically (e.g. a user profile)

Such selection contributes to the evolution of the user profile as well

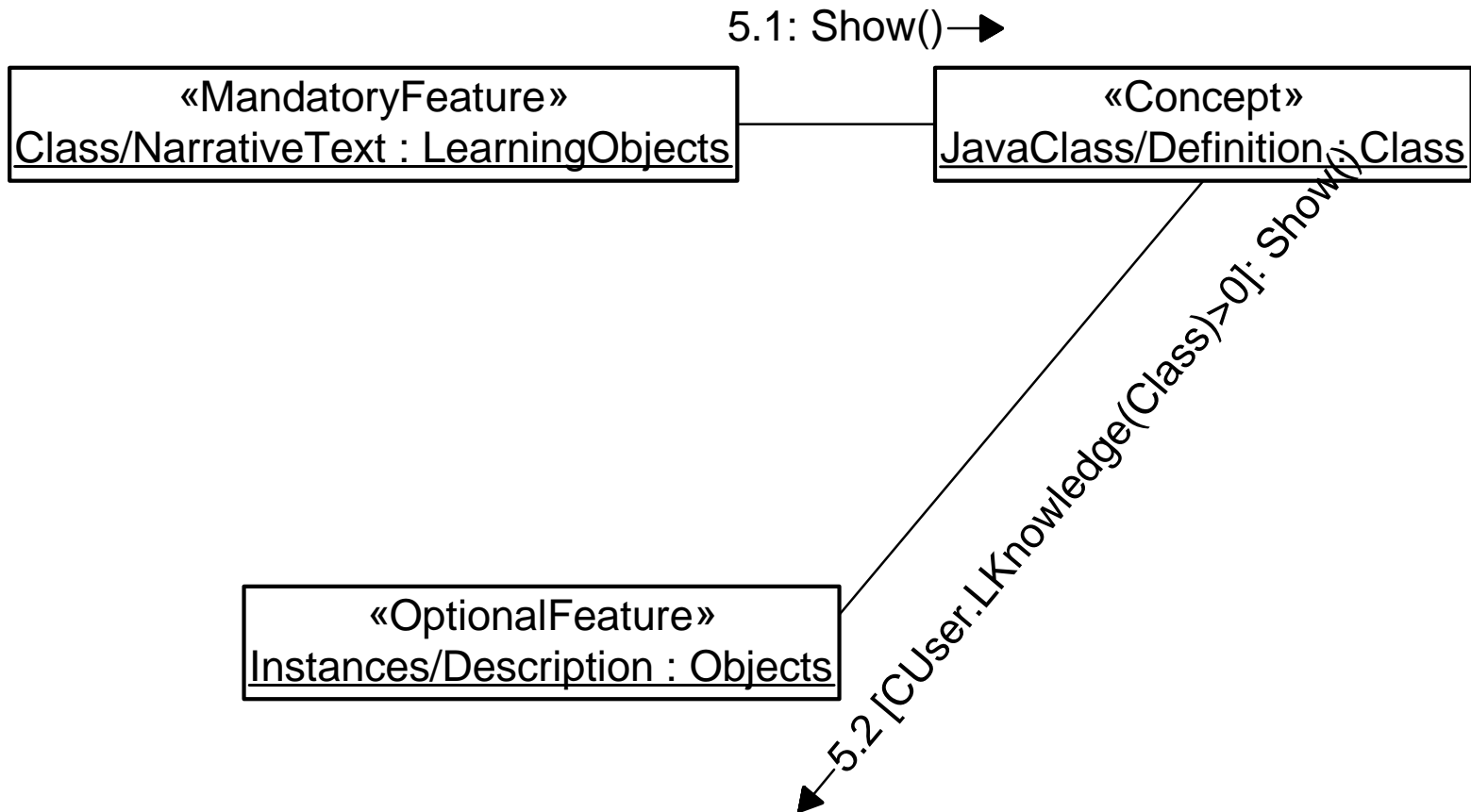
Path through information can be altered according to the profile as well

Static structure model is not enough => behavioural models, e.g. State diagrams which provide a user interaction and reactivity point of view to the specifications

Collaboration Models



Collaboration Models



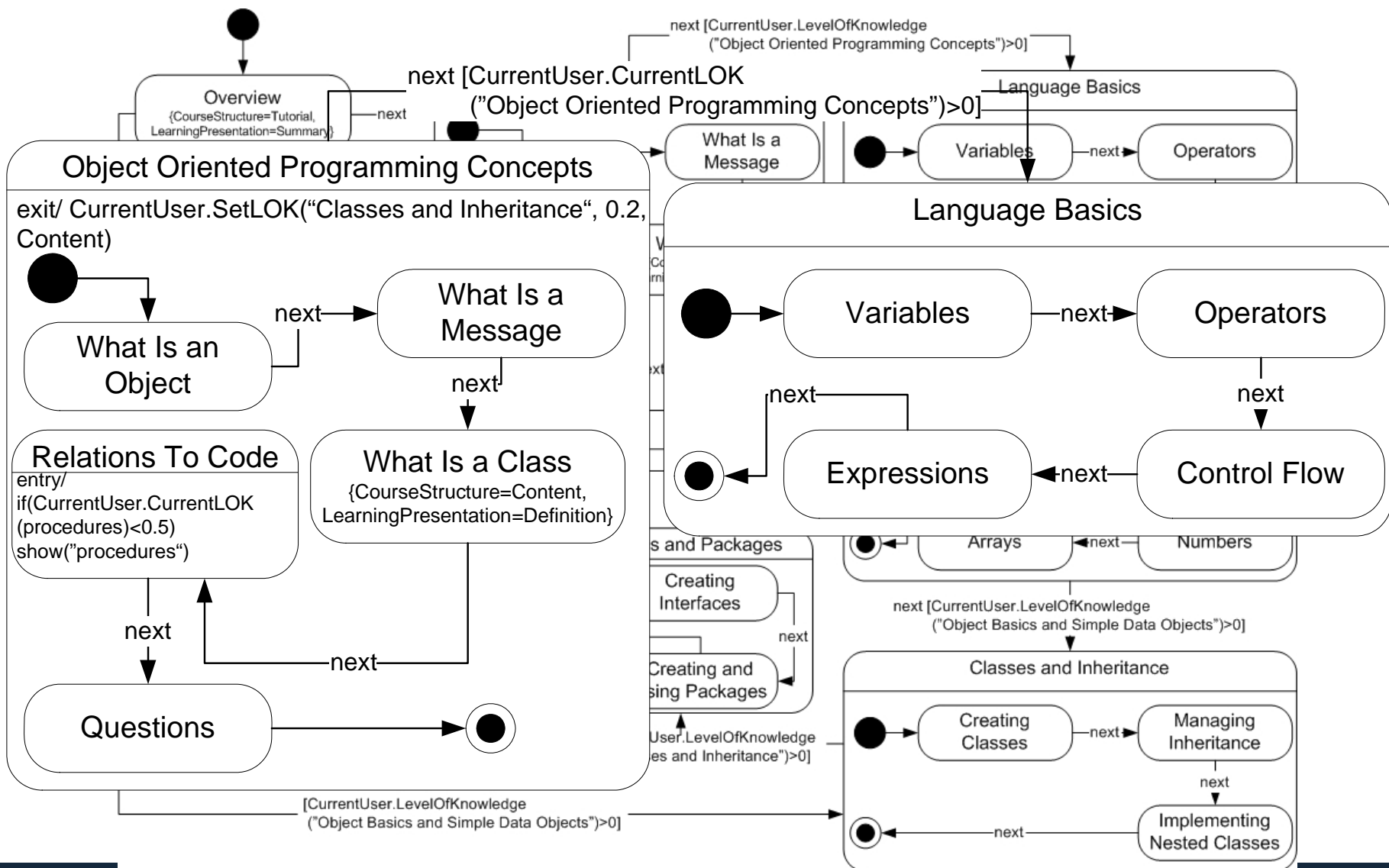
State Diagrams: Adaptive Navigation

State Diagram as a model of a trail through an information space
(states are hypertext nodes, transitions are possible links)

Event-Condition-Action on states and transitions for constraints
on possible links and nodes

User model features in conditions

Actions for user model run-time updates and additional
computations needed for rendering or processing



The UML-Guide

Adresse <http://localhost:8080/uml/course.jsp> Wechseln zu Links >>

Navigation Map

- Start
- Language Basics
- Object Basics and Simple Data Objects
- Classes and Inheritance
- Interfaces and Packages
- Overview
- Object Oriented Programming Concepts
 - Language Basics
 - StartOO
 - What Is an Object**
 - What Is a Message/next
 - What Is a Message
 - What Is a Class
 - Relations to Code
 - Questions
 - EndOO
 - Common Problems
 - End

:.Overview :.Lecture Modules

Previous Next

LECTURE MODULE: What is an Object?

Objects are key to understanding object-oriented technology. Software objects are modeled according to a state and a behavior. A software object maintains its state in one or more variable.

A variable is an item of data named by an identifier. A software object implements its behavior with methods . A method is a function (subroutine) associated with an object.

The following illustration is a common visual representation of a software object:

Methods (behavior)

Variables (state)

DEFINITIONS:

Fertig Lokales Intranet

The screenshot shows three overlapping browser windows. On the left is the 'PERSONAL READER' for Peter Dolog, featuring a search input field and a list of document titles. In the center is 'The Java Tutorial' page from developers.sun.com. On the right is the 'ELENA: PLA - Personalized Search Service' window, which displays search results in a table. A black arrow originates from the search button in the 'PERSONAL READER' sidebar and points to the search input field in the 'ELENA' window.

PERSONAL READER
Peter Dolog

search

Title
Java Tutorial
[x] The Java Tutorial (...)
[DONE] [AGREE]

Generalizations

Details

- [x] Object Basics and Simpl...
- [x] Object-Oriented Program...
- [] Learning the Java Lecture
- [] Classes and Inheritance
- [] Interfaces and Packages
- [x] Custom Pathways and The...

Summaries

Exercises

Again

logout

developers.sun.com

Java

Developers Home > Products & Technolog...

Tutorials & Code Camps
The Java Tutorial

The Java™ Tuto

A practical guide for programmers
with hundreds of complete, working e...

Search

Trail types: Basics | GUIs | Specialize...

Trails Covering the Basics:

- Your First Cup of Java: Detailed instructions to help you run your program: UNIX, Microsoft Windows
- Getting Started
- Learning the Java Language
- Essential Java Classes
- User Interfaces that Swing: A Quick Guide
- Writing Applets

Trail on Constructing GUIs:

Published in

ELENA: PLA - Personalized Search Service - Microsoft Internet Explorer

ELENA Personal Learning Assistant
for SMART SPACE FOR LEARNING
Peter Dolog & Michael Sestak
Information Society Technologies

Personalized Search Service

USER:
default

Query results:

Preco	Reco	Title	Description	Concepts
	[x]	JLayeredPane (Java 2 Platform SE v1.4.2)	?	http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Container; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#JComponent; int; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Integer; Numbers
	[x]	LongBuffer (Java 2 Platform SE v1.4.2)	?	http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Array; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#Long; byte; if; new Operator; http://webbase.learninglab.uni-hannover.de:9000/pla/ACM_java.rdf#this; http://webbase.learninglab.uni-

Advantages

- Designer
 - Separation of domains with possibility to connect them adaptively
 - State diagrams and collaboration models reflect user interaction better
 - Semiautomatic generation
- Application
 - Models used for search, provision, and integration purposes
- Process
 - Separation can contribute to easier distribution of work

Outline

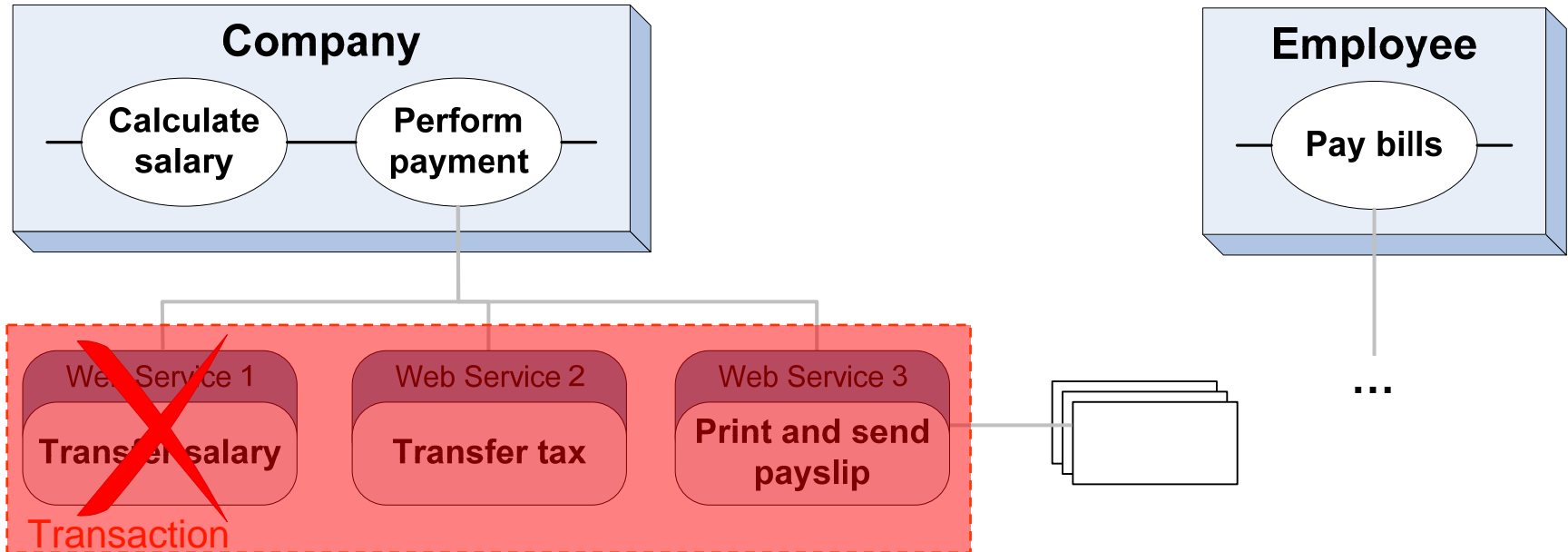
Motivations and Applications

Adaptive Web Application Design

- Information Access Applications
- Service Based Applications

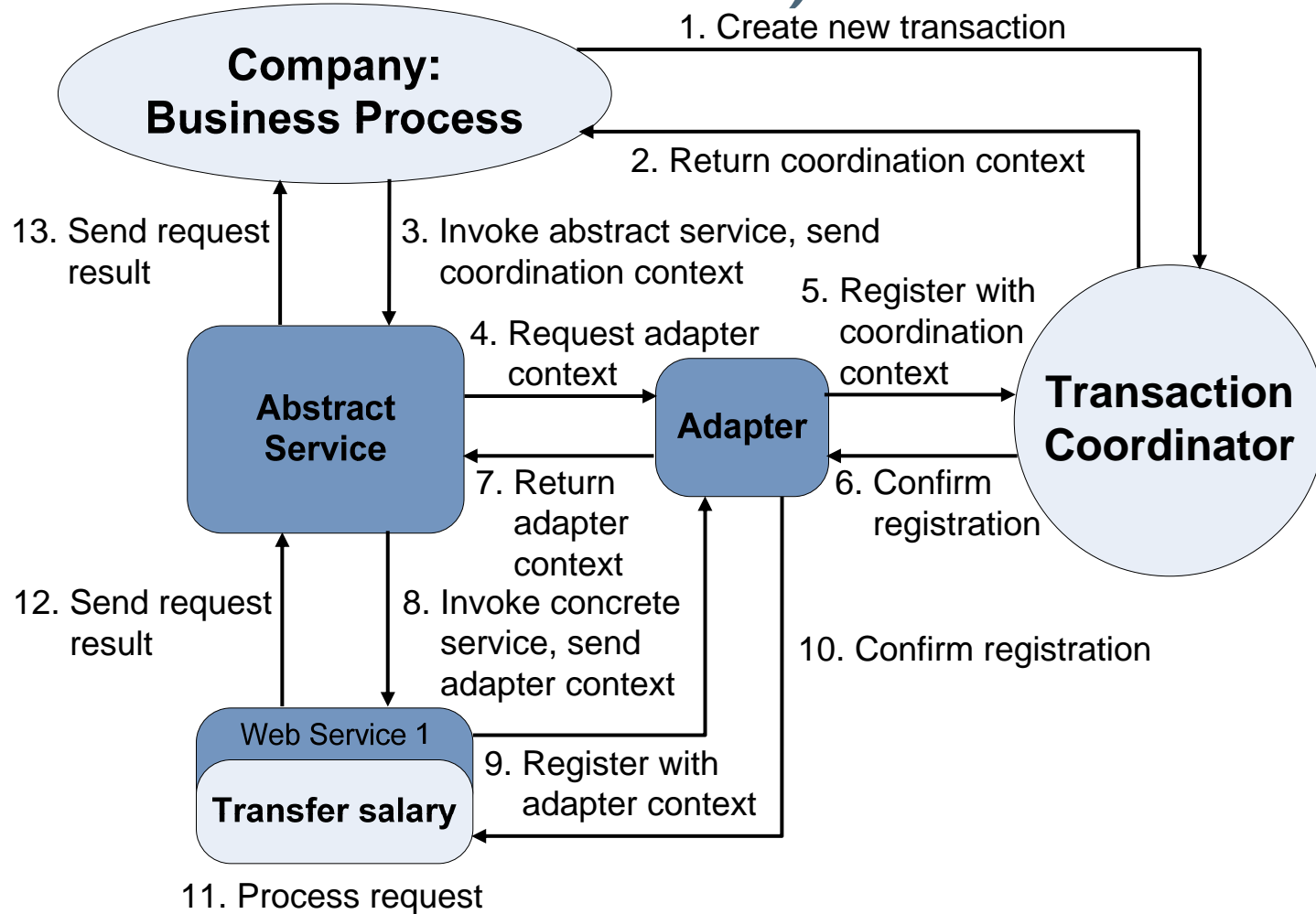
Further Challenges

Payroll Scenario

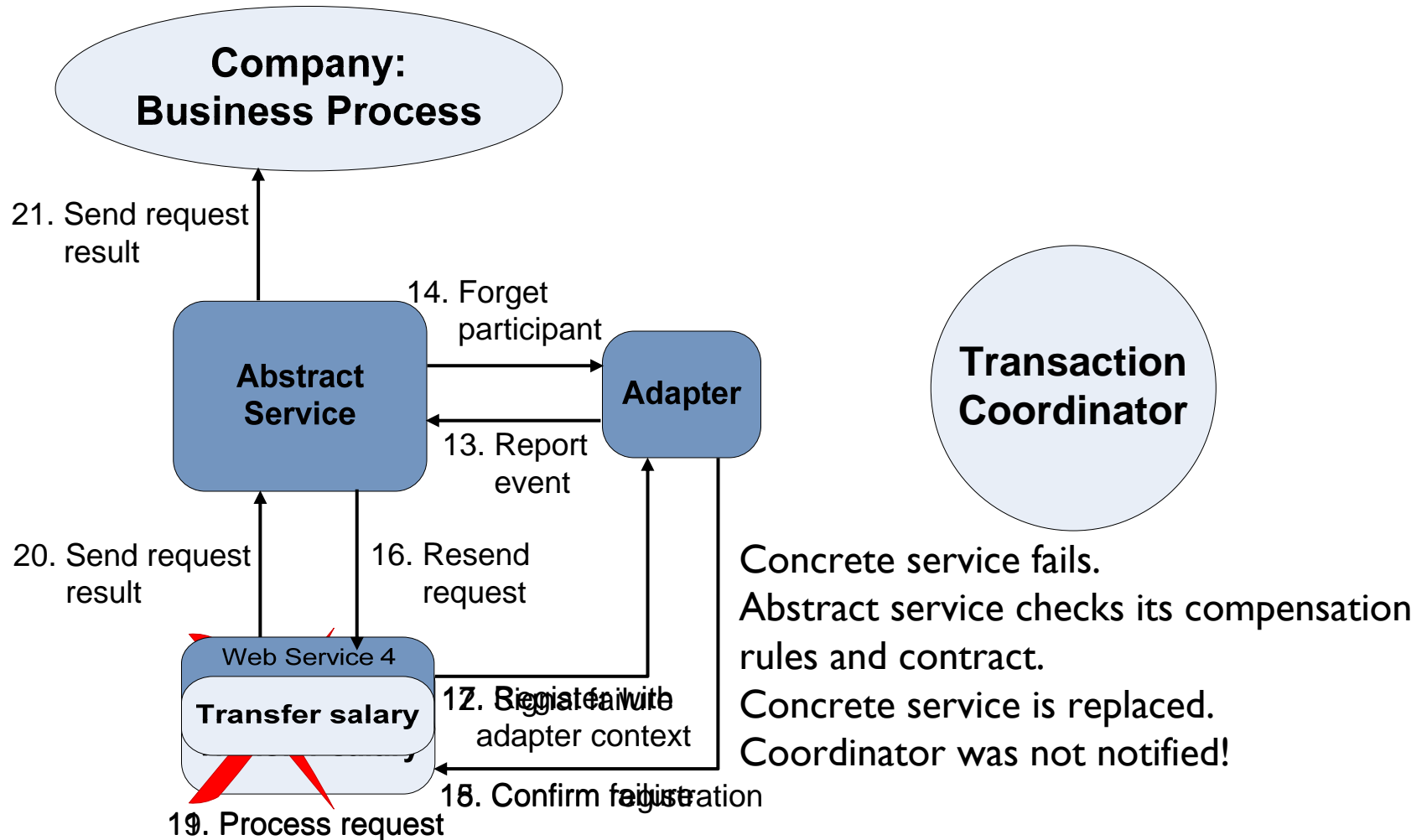


A service fails due to an internal error.
 The error can only be compensated by aborting the complete transaction.
 Why should the transaction be aborted, if a different service exists that can perform the same operations?

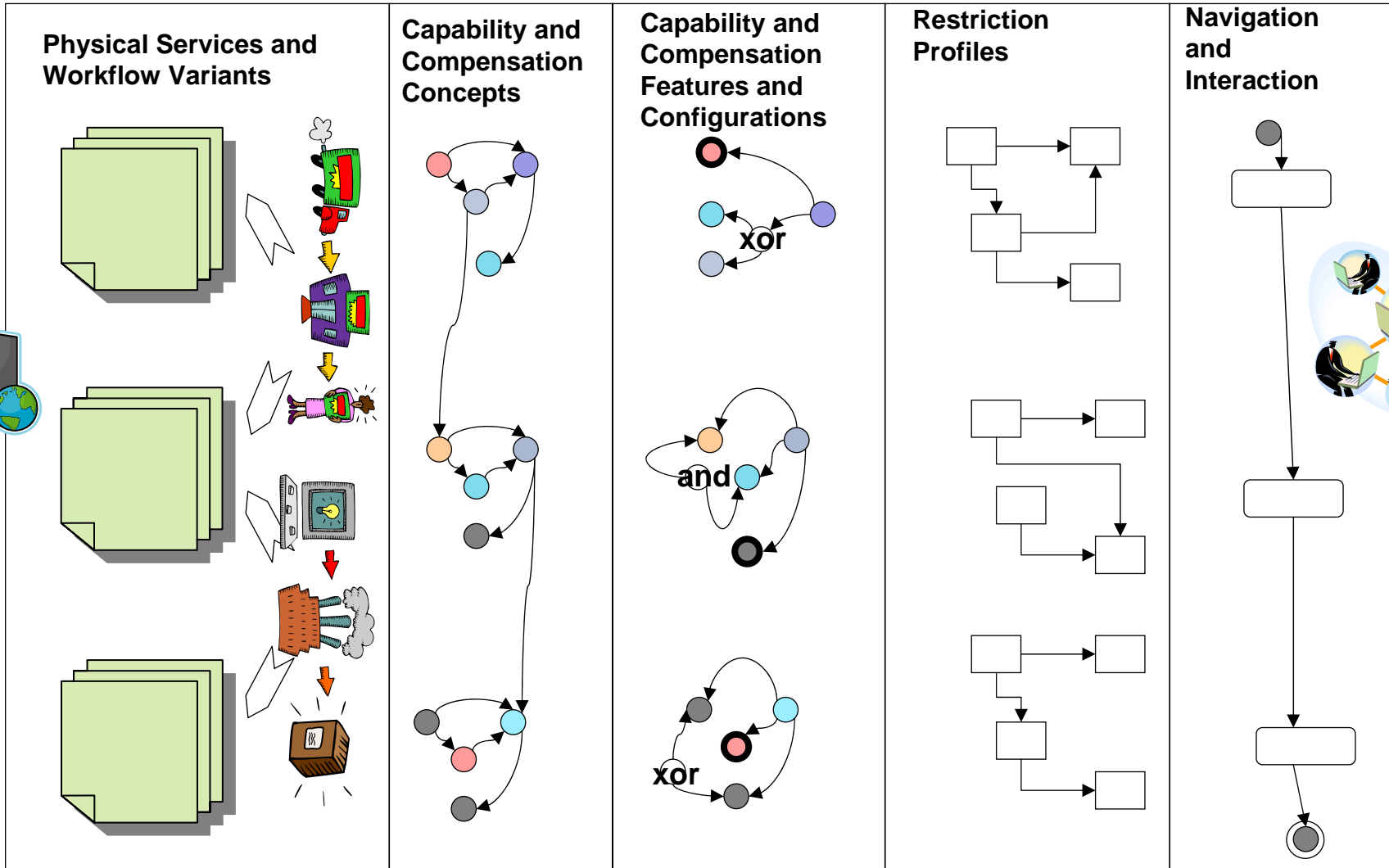
Extended Transaction Coordination Structure (ICWE2007/ACMTWEB 2008)



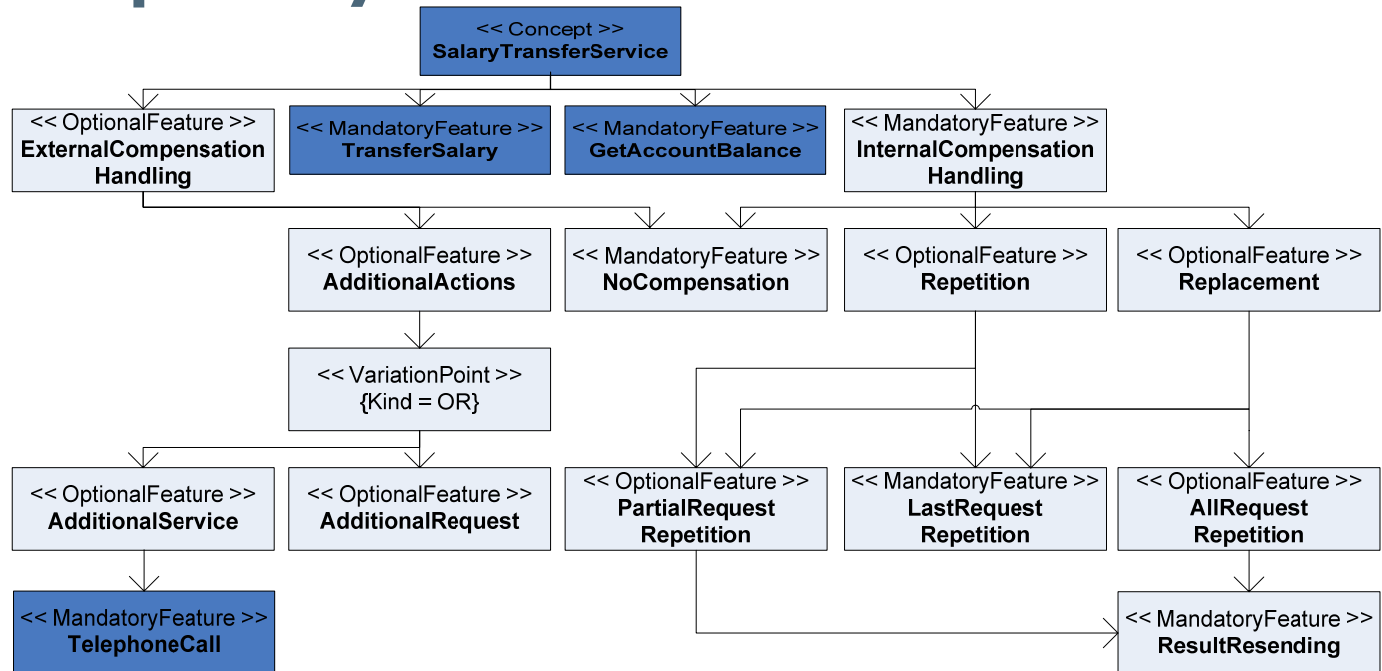
Internal Compensation Handling – Replacement



Layers of Abstraction Revisited



Example: Capability Feature Model



Consists of:

- *functionality feature model*
- *compensation feature model*

The compensation feature model can contain custom features.

Requirements and Restrictions Feature Models

Similarly, client requirements for functionality and compensations can be stated in feature models

Matchmaking with capability profile creates restriction profile

Restriction profile specifies a compensation strategy for abstract service

Outline

Motivations and Applications

Adaptive Web Application Design

- Information Access Applications
- Service Based Applications

Further Challenges

Rich Internet Applications



Apartment Room Arrangements

The screenshot displays a web application interface for configuring apartment room arrangements. At the top, three sofa options are presented:

- KARLANDA stol**: kr 2.599,00
- KARLANDA sofa 3**: kr 3.599,00
- KARLANDA sovesofa 2**: kr 6.499,00

The main interface features a floor plan labeled "Stueplan" with rooms: Stue, Entré, Køkken, Soveværelse, and Bad. A furniture catalog on the right lists various items such as Stole, Børde, Sofæer, Senge, Reoler, Skabe, Tv mv., Vask, Toilet, Køkken, Lamper, Diverse, Planter, and Biler. A "Skalering af objekt" section shows dimensions for a "lænestol" (Length: 260 cm, Width: 110 cm). Annotations include "Recommended StoreItems" pointing to the sofa options, "StoreItems" pointing to the catalog, and "RoomItems/Room Configuration" pointing to the floor plan.

Social Web Applications



FP7/ICT STREP **Kowi** Knowledge in a Wiki

The screenshot shows a Wiki page for 'Bilberry'. On the left is a navigation menu with items like 'User', 'Main Page', 'Help Contents', 'Recent Changes', 'New Page', 'Create Template', 'Upload File', 'Manage Roles', 'Manage Account Sets', 'Manage Users', 'Manage Roles', and 'Page Namespaces'. A sidebar on the right contains a 'Please select a type' dropdown menu with options like 'Type for Bilberry', 'BioSpecies', 'cc:Agent', 'cc:Licence', 'cc:Permission', 'cc:Prohibition', 'cc:Requirement', 'cc:Work', 'dcterms:Box', 'dcterms:DCMC', 'dcterms:DateScheme', 'dcterms:FormScheme', 'dcterms:IMF', 'dcterms:ISO3166', 'dcterms:ISO639-2', 'dcterms:IdentifierScheme', 'dcterms:LCC', 'dcterms:LCSH', 'dcterms:LanguageScheme', 'dcterms:Mesh', 'dcterms:NLM', 'dcterms:Period', 'dcterms:Point', 'dcterms:RFC1746', 'dcterms:RFC3066', 'dcterms:RelationScheme', 'dcterms:SourceScheme', 'dcterms:SpacialScheme', 'dcterms:SubjectScheme', 'dcterms:TGN', 'dcterms:TemporalScheme', 'dcterms:TypeScheme', 'dcterms:UDC', 'dcterms:URI', 'dcterms:WIDDTF', 'exif:ID', and 'exif:Attribute'. The main content area includes a 'References' section with a list of links and a 'Please select a type' dropdown menu. The page title is 'Bilberry' and the URL is 'http://192.168.243.130:8080/index.jsp?title=Bilberry'.

Lines of Code
 Number of Defects per Module
 Number of Defects per Project
 Unresolved Issues
 Changes in Processes



Composition Models



Workflows vs. Middleware

Compensations and adaptations can be specified at the design level in workflows

Compensations and adaptations can be encoded in an intelligent middleware

How to combine them

How to compose them

How to ensure consistency

...

References

- P. Dolog, B. Simon, T. Klobucar, W. Nejdl: Personalizing Access to Learning Networks. ACM TOIT 8(2), January 2008
- S. Ceri, P. Dolog, M. Matera, W. Nejdl: Adding Client Side Adaptation to the Conceptual Design of eLearning Applications. JWE 4(1), 2005
- M. Schäfer, P. Dolog, W. Nejdl: An Environment for Flexible Advanced Compensations of Web Service Transactions. To appear in ACM TWEB, Special Issue on SOA.
- P. Dolog, W. Nejdl: Using UML and XMI for generating adaptive navigation sequences in web-based systems. UML 2003.
- P. Dolog, W. Nejdl: Using UML-based feature models and UML collaboration diagrams to information modelling for web-based applications. UML 2004.
- P. Dolog, J. Stage: Designing interaction spaces for rich internet applications with uml. ICWE 2007.
- P. Dolog, N. Henze, M. Sintek, W. Nejdl: Personalization in distributed e-learning environments. WWW2004
- P. Dolog, N. Henze, M. Sintek, W. Nejdl: The personal reader: Personalizing and enriching learning resource using semantic web technologies. AH2004.
- P. Dolog. Engineering Adaptive Web Applications. Doctoral Dissertation, Leibniz Universität Hannover. March 2006 (summa cum laude)

Thanks!!! Questions?

dolog@cs.aau.dk

<http://www.cs.aau.dk/~dolog>

