Model-Driven Web Engineering

UWE Approach

Nora Koch

Web Engineering Group
Ludwig-Maximilian-Universität München (LMU)
Germany

Madrid, 28.05.2008
Web Engineering Group of LMU

- **Current staff**
  - Alexander Knapp and Nora Koch
  - Gefei Zhang (PhD student)
  - Christian Kroiß and Marianne Busch (students)
  - Martin Wirsing (head of department)

- **Projects**
  - UWE
  - MAEWA
  - SENSORIA

- **Cooperations**
  - Universities of Alicante, Extremadura, Málaga, Sevilla (ES), Milano (IT), Viena & Linz (AT), Lingby (DK), LaPlata (AR)
  - Industry: FAST/Cirquent, S.Co LifeScience

- **Other activities**
  - ICWE conferences since 2003
  - MDWE workshops since 2005
  - MDWEnet initiative since 2006
Web Engineering

- Software Engineering for the Web domain
- Concept introduced by San Murugesan in 1998

Definition
- Web engineering is the application of a **systematic and quantifiable approach** to cost-effective requirements analysis, design, implementation, testing, operation, and maintenance of **high-quality Web software**.
- Web engineering is also the **scientific discipline** concerned with the study of these approaches.

*Gerti Kappel, Birgit Pröll, Siegfried Reich, Werner Retschitzegger*  
*Web Engineering* (2006)

- Established as own discipline
- Model-Driven Web Engineering relevant topic as many Web engineering approaches follow a **model-driven development approach**
Outline

- Web Software and Model-Driven Development
- UWE Approach
  - Modelling Language
  - Metamodel
  - Development Process
  - Tool Support
- Outlook
Web Software

Web information system (WIS)

- software system based on client/server technology provides information through a user interface (Web browser). Pages belong to a particular domain name or subdomain on the World Wide Web
  - Web site, homepage, Web portal

- Web application
  - software system that provides Web specific resources such as content and services through a Web browser

- Web service
  - software system designed to support interoperable machine to machine interaction over a network. Web services are frequently just application programming interfaces (API) that can be accessed over a network, such as the Internet, and executed on a remote system hosting the requested services

\(^1\text{W3C, www.w3.org}\)
Categories of WIS and Web Applications

- Document based
- Transactional
- Interactive
- Workflow-based
- Collaborative
- Ubiquitous
- Semantic web
- Portal-oriented

Web Specifics in Engineering

- **Hypermedia paradigm**
  - nodes & links
  - text & multimedia

- **Omnipresent due to the nature of the Web**
  - global and permanent availability
  - comfortable and unified access
  - distributed information / services

- **Dynamic development**
  - incremental number of Web pages
  - continuous improvement of existing Web applications (content, links, layout)
  - offer of new services
  - adaptation required by new Web technologies

- **Management aspects**
  - multidisciplinary development team
  - inhomogeneous and immature technical infrastructure
  - short product lifecycles → short development cycles

Specific engineering methods for the Web domain
Model-Driven Development

- MDD approaches based on
  - models, metamodels and model transformations
- MDD approaches require languages for
  - specification of models
    - UML, BPMN, …
  - description of metamodels
    - UML, MOF, OCL, …
  - definition of model transformations
    - Java
    - Graph transformations
    - ATL, QVT, …
- Model-Driven Architecture (MDA)
  - computational independent model (CIM)
  - platform independent model (PIM)
  - platform specific model (PSM)
Model Transformations

- Goal is automatic translation between source and target models
- Translation performed by a transformation engine that executes transformation rules
- Set of rules
  - seen as a model
  - based on a transformation metamodel
- Metamodels are based on a metametamodel
- MDA model transformations
  - CIM2PIM
  - PIM2PIM
  - PIM2PSM

Model transformation pattern (J. Bézivin, 2004)
### Web Engineering Methods

- **HDM / HDM-lite**: Hypertext Design Model
- **Hera**: Navigational Development Technique
- **NDT**: Hypertext Modeling Method (HM³)
- **MIDAS**: Service-Oriented Development Method (SOD-M)
- **OO-H**: Object-Oriented Hypermedia Method
- **OOHDM**: Object-Oriented Hypermedia Design Method
- **OOWS**: Object-Oriented Web Solution
- **RMM**: Relationship Management Methodology
- **UWE**: UML-based Web Engineering
- **W2000**: 
- **WAE / WAE2**: Web Application Extension
- **WebML**: Web Modeling Language
- **WebSA**: Web Software Architecture
- **WSDM**: Web Site Design Method

**Based on**
- **ER**: Entity Relationship Model
- **OMT**: Object Modeling Technique
- **UML**: Unified Modeling Language
Several methods propose building models
  - Hera, MIDAS, OOHDM, OO-H, UWE, WebML, …
  - separation of concerns
    - content
    - navigation
    - presentation
    - business processes
    - adaptation, …
  - similar Web specific modeling elements
  - different notations

Some methods define metamodels for modelling languages

Few approaches address model transformations
  - MIDAS, OOWS, UWE, WebSA, …

Goal of almost all
  - Platform specific models in a late development stage
Main characteristic is the use of UML for all models
- “pure” UML whenever possible
- UML extension for Web specific features: UML profile

Use of OMG standards, such as UML, MDA, MOF, OCL, XMI, …

Focuses on systematisation and automatic generation

UWE comprises
- a modelling language for the graphical representation of models of Web applications
- a metamodel for UWE modelling elements
- a development process
- tools supporting semi-automatic generation
  - MagicUWE & ArgoUWE editors for the design
  - set of ATL transformations
    - model-to-model and model-to-code transformations
    - set of plugins integrated in eclipse environment
Dimensions of Web Modelling

- **Levels**
  - Presentation
  - Hypertext
  - Content
  - Structure
  - Behaviour

- **Adaptivity**
  - Levels

- **Phases**
  - Requirements Analysis
  - Design
  - Implementation

**Aspects**

**Modelling process**

- Information-driven ("content first")
- Presentation-driven ("layout first")
- Functionality-driven ("business processes first")

*Source: Kappel et al. Web Engineering, d-punkt (2003)*
Why UML?

- UML is a graphical language for specifying, constructing and documenting software artifacts
- UML is a de facto industry standard and an OMG standard
- UML includes
  - notation
  - diagram types
  - Object Constraints Language (OCL)
  - metamodel
  - well-formedness rules
- UML does not provide a development process

How expressive is UML for the development of Web applications?
- UML does not include specific Web model elements
- UML defines extension mechanisms → UML profiles
UML Extensions

- **Light weight extension**
  - called a **UML profile**
  - based on extension mechanisms provided by UML
  - defines stereotypes for new metaclasses
    - domain specific: EJB «bean», «session», «entity», …
    - defined in the UML: «metaclass», «trace», «file», …
  - tagged values for metaattributes
  - OCL constraints for invariants, pre- and postconditions
  - CASE tool support by UML tools

- **Heavy weight extension**
  - different notation
  - other diagram types not defined in the UML
  - need of proprietary CASE tool
Analysis and Design Models in UWE

- Analysis models of a Web application
  - functional requirements are specified by
    - uses cases
    - workflows
  - data (content) requirements are specified by
    - domain models

- Design models of a Web application
  - information aspects
    - content model
  - hypertext structure and navigation functionality
    - navigation model
  - layout schema
    - presentation model
  - functionality
    - process model
    - adaptivity model
Example: Simple Music Portal

- **Inspired by** [www.mp3.com](http://www.mp3.com)
  - offers albums for downloading
  - contains information about albums, songs and artists
  - this information is available for free
  - registered users can download them
  - for downloading they need to have enough credit on their prepaid account
  - accounts are rechargeable
Modelling Requirements with UWE

- Graphical visualization by UML use case diagram
  - to model required functionality
  - distinguishes between navigation and process use cases
- Web specific model elements (not yet implemented in MagicDraw)
  - «navigation» use cases for browsing tasks
    - browse use cases
    - view use cases
    - search use cases
  - «web process» use cases for other tasks
    - other use cases
Content Modelling

- Representation of domain information
  - persistent data

- Modelling technique
  - UML class diagram
  - plain UML
  - no additional semantics required

UML class diagram:

- **Album**
  - title : String
  - price : float
  - recorded : Date
  - cover : String
  - description : String
  - downloadURL : String

- **Genre**
  - name : String

- **Track**
  - number : int
  - length : Integer

- **Song**
  - title : String

- **Performer**
  - description : String

- **Group**
  - name : String
  - formed : Date [0..1]
  - disbanded : Date [0..1]

- **Artist**
  - lastName : String
  - firstName : String
  - born : Date
  - died : Date [0..1]
User Model

- Representation of session specific information
  - allows for customization
- Represented as UML class diagram
  - “normal” UML classes
  - Visit object for each session
    - «visitClass»
Navigation Modelling

- **Goals**
  - to represent nodes and links of the hypertext structure
  - to design navigation paths
  - to avoid disorientation and cognitive overload

- **Navigation model**
  - represented by a UML class diagram
  - uses specific modelling elements for Web concepts

- **Basic elements to model the core hypertext structure**
  - «navigation class» specifies a hypertext node visited by a user through browsing (related to a content class)
  - «navigation link» specifies a hyperlink used to access the target navigation object from the source navigation object
Access Primitives

- Systematic enhancement of the navigation structure model by
  - «index» for all navigation links which have multiplicity > 1 at the directed association end
  - «menu» for all navigation classes with more than 1 outgoing association

- Design decision to include
  - «guidedTour» instead of index
  - «query» for selection of instances of a navigation class
  - tagged value `home` to indicate starting point of the application (node without ingoing links)
  - tagged value `landmark` to indicate that a node is reachable from everywhere (all other nodes include a link to the landmark node)

- Shortcuts for more complex constructs (if represented in UML without extension)
**Navigation Model Elements: Menu**

- **Menus** are used to structure the outgoing links from a node
  - usually associated to a navigation class by composition
  - consists of a set of links to heterogeneous elements, such as indexes, guided tours, queries, instances of navigation classes or other menus
  - UML stereotype: «menu»

- Semantics of menu
Navigation Model
(excerpt)
Modelling Processes in UWE

- **Navigation model of a Web application**
  - represents the static information structure accessible to a user of the system
  - specifies browsing (navigation) functionality

- **Process model**
  - represent the dynamic aspects of a Web application
  - specifies functionality, such as transactions and complex workflows of activities

- Process modelling consists of
  - definition of process classes (for non-navigation use cases)
  - integration of these process classes in the navigation model
  - description of the behaviour through a process flow
    - represented as UML activity diagram
Process Elements

- **Process class** represents the process through which the user will be guided in the Web application
  - for complex process that require more than a single class, an additional process model is built
  - UML stereotype: «process class»

- **Process link** is used to model the association between a «navigation class» and a «process class»
  - indicates entry points and exit points of processes within the navigation structure
  - UML stereotype: «process link»
Modelling the Process Flow

- The behavior of a Web process is defined by the process flow model
  - represented by UML activity diagram
  - result of the refinement of the activity diagram drawn for requirements specification
  - «process class» stereotype
  - optional use of nested activity diagrams

- Process flow consists of
  - flow of execution represented by activity nodes connected by activity edges
  - control nodes that provide flow-of-control constructs, such as decisions and synchronization
  - object nodes that represent data flowing along object flow edges or pins associated to the actions
  - in UML2 the semantic of activities is based on control and data token flows, similar to Petri nets
Process Model: Login
Modelling Presentation

- Representation of layout for the underlying navigation and process models
  - is an abstract presentation
  - concrete presentation requires specification of additional physical properties of the layout
    - colour, position, …
    - # of columns in table, type of menu, …

- Presentation classes represent Web pages or part of pages
  - composition of user interface elements
  - hierarchical composition of presentation elements

- UML class diagram for the structure of the presentation
  - using UML container notation

- UML interaction diagram (sequence diagrams)
  - used for modelling behaviour of presentation classes (classical UML)

- Alternative: development of a prototype
Presentation Model Elements

- Structural presentation elements
  - «presentationClass»/ «presentationGroup»: container of user interface elements representing a logic unit of presentation associated to a navigation class or process class
  - «presentationPage»: presentation class at highest level
  - «presentationAlternatives»: container for presentation classes which are not shown simultaneously

- User interface elements
  - «anchor»
  - «button»
  - «text»
  - «image»
  - «textInput»
  - «selection»
Adaptive Web Applications

- Adaptation/Customization for
  - user properties: knowledge, tasks, preferences, interests
  - context properties: location (place and time) & platform (HW, SW, network)

- Update of a user model / context model
  - observation of the user behaviour or environment by the system

- Techniques for adaptation
  - **content** adaptation
    - inserting and removing text/multimedia features
    - content variants
  - **navigation** adaptation
    - link ordering
    - link annotation
    - link hiding
    - link generation
  - **presentation** adaptation
    - modality adaptation (audio or text)
    - language selection
    - layout variants (resizing of fonts, images, changing colours)
Modelling Adaptivity

- UWE uses a technique called Aspect-Oriented Modelling (AOM)
- Identification of
  - «pointcut» (including conditions)
  - «advice»
- Weaving the result into the web application based on
  - current state of the user model
  - information provided by link traversal
- Example: links only visible for registered users to
  - BuyAlbum
UWE Metamodel

- UWE Metamodel is defined as a conservative extension of UML 2.0
  - model elements of the UML metamodel are not modified
  - all new elements are related by inheritance to at least one model element of the UML
  - use of OCL to specify additional semantics of the new elements
  - so-called light-weighed extension of UML
UWE Metamodel Characteristics

- UWE metamodel
  - reflects separation of concerns in the structure of Core
  - shows cross-cutting aspect of adaptation
- UWE metamodel is profileable
  - mapping to a UML profile is possible
- UWE metamodel is MOF compatible
  - uses XML metadata interchange format (XMI)
UWE Metamodel: Navigation

(see stereotypes used in the example)
UWE Profile: Navigation

- UML stereotypes for Web specific concepts used for the specification of the hypertext structure
- Extends relationships
- UML metaclasses

Diagram:

- «profile» Navigation
- «metaclass» Class
- «metaclass» Property
- «metaclass» Association
- «stereotype» Node
- «stereotype» NavigationProperty
- «stereotype» Link
- «stereotype» ProcessClass
- «stereotype» NavigationClass
- «stereotype» Access Primitives
- «stereotype» ProcessLink
- «stereotype» Menu
- «stereotype» GuidedTour
- Dependencies: Dependency
- «stereotype» Primitive2Property
- «stereotype» NavigationClass2Menu
Model-Driven Process of UWE

- Graphical representation of the process
  - process as UML activity diagram
  - model transformations as stereotyped UML actions
  - models as UML object flow states
  - implicit initial and final state

- Types of models in UWE
  - requirements model (CIM)
  - functional models (PIM)
    - content model
    - navigation model
    - ...
  - architecture models (PIM)
  - integration models (PIM)
  - models for J2EE, .Struts (PSM)
UWE Development Process
Requirements to Functional Models

- Requirements Model
  - «CIM2PIM» Requirements2Content
  - «CIM2PIM» Requirements2Navigation
  - «CIM2PIM» Requirements2Process

- Content Model
  - «PIM2PIM» Content2Navigation

- Navigation Model
  - «PIM2PIM» Navigation2Content
  - «PIM2PIM» Navigation2Process
  - «PIM2PIM» Navigation2Navigation

- Process Model
  - «PIM2PIM» Process2Navigation

Web Software & MDD
- UWE Approach
- Modelling Language
- Metamodel
- Development Process
- Tool Support
Construction of Functional Models

- UWE metamodel and UWE profile
  - navigation elements: *navigation class*, *navigation link*, *index*, …
  - presentation elements: *presentation class*, *anchor*, *image*, …

- Case tool ArgoUWE/MagicUWE
  - extension of ArgoUML/MagicDraw
  - provides UWE Profile
  - supports (semi-)automatic execution of transformations
Transformation Content to Navigation

- **Content2Navigation**
  - generates navigation classes from content classes
  - adds a navigation links based on associations of the content model

- **Marking elements**
  - identification of classes of the content model that are relevant for the navigation view
  - task performed by designer

- **Implementation**
  - ArgoUWE/MagicUWE plugin implemented in Java
  - ATL (ATLAS Transformation Language)
Refinement of Navigation Model

- Improvement based on patterns
  - index for associations with multiplicity greater than one at the directed association end
  - menu for navigation classes with multiple outgoing associations

- Implementation
  - Java in ArgoUWE/MagicUWE
  - ATL
Integration with Architecture Models

- **Web Software Architecture (WebSA) approach***
  - domain specific language for modelling architectural views of Web applications
    - subsystem model
    - configuration model
    - integration model
  - UML profile of architectural modelling elements
    - Web component
    - Web port
    - Web connector
    - server page,
    - etc.
  - model transformations written in QVT-P

Generating “Big Picture” Model

- Generation of an integrated functional model (“big picture”)
  - transformation target UML state machine for integration of content, navigation and process models
  - graph transformation language
  - tool: Attributed Graph Grammar System (AGG)
  - validation of correctness by model checking
Capture navigation nodes as states (with parameters for data)

Example: music portal: transformation for navigation node “song”
Big Picture: Transformation of Business Process
Model Validation

- Model transformations for Web applications based on UWE and its UML-based metamodel
- Graph transformations into integrating UML state machine
- Model validation by model checking using Hugo/RT (http://www.pst.ifi.lmu.de/projekte/hugo)
- Automation of transformation process using Attribute Graph Grammars (AGG)
Generation of Web Applications

- UWE uses a transformational approach
  - to generate data model and presentation layer
  - based on content, navigation structure and presentation models
  - transformation rules from UWE content model to Java beans
  - transformation rules from UWE presentation model to Java Server Pages (JSPs)
- UWE uses an interpretational approach
  - using a virtual machine
  - to interpret the process model (activity diagrams)
  - configuration data for the virtual machine is generated from process and navigation model
- Implemented so far
  - using the Spring framework
  - transformations defined in ATLAS Transformation Language (ATL)
Model to Code Transformations
### Classification of UWE Model Transformations

<table>
<thead>
<tr>
<th>Transformation</th>
<th>Characteristics</th>
<th>Type</th>
<th>Complexity</th>
<th>Marks</th>
<th>Execution</th>
<th>Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Req2Content</td>
<td></td>
<td>CIM to PIM</td>
<td>simple</td>
<td>WebRE profile</td>
<td>automatic</td>
<td>QVT</td>
</tr>
<tr>
<td>Req2Architecture</td>
<td></td>
<td>CIM to PIM</td>
<td>simple</td>
<td>-</td>
<td>manual</td>
<td>-</td>
</tr>
<tr>
<td>Content2Navigation</td>
<td></td>
<td>PIM to PIM</td>
<td>simple</td>
<td>navigation relevance</td>
<td>semi-automatic</td>
<td>Java, ATL</td>
</tr>
<tr>
<td>NavigationRefinement</td>
<td></td>
<td>PIM to PIM</td>
<td>simple</td>
<td>UWE profile &amp; patterns</td>
<td>automatic</td>
<td>Java</td>
</tr>
<tr>
<td>Req2Navigation</td>
<td></td>
<td>CIM to PIM</td>
<td>merge</td>
<td>WebRE profile</td>
<td>automatic</td>
<td>QVT</td>
</tr>
<tr>
<td>Navigation2Presentation</td>
<td></td>
<td>PIM to PIM</td>
<td>simple</td>
<td>UWE profile</td>
<td>automatic</td>
<td>Java, ATL</td>
</tr>
<tr>
<td>StyleAdjustment</td>
<td></td>
<td>PIM to PIM</td>
<td>merge</td>
<td>style guide</td>
<td>automatic</td>
<td>Java</td>
</tr>
<tr>
<td>Functional2BigPicture</td>
<td></td>
<td>PIM to PIM</td>
<td>merge</td>
<td>patterns</td>
<td>automatic</td>
<td>graph transformations</td>
</tr>
<tr>
<td>Functional&amp;Architecture2Integration</td>
<td></td>
<td>PIM to PIM</td>
<td>merge</td>
<td>UWE &amp; WebSA profile</td>
<td>automatic</td>
<td>QVT-P</td>
</tr>
<tr>
<td>Integration2J2EE</td>
<td></td>
<td>PIM to PSM</td>
<td>merge</td>
<td>patterns</td>
<td>automatic</td>
<td>QVT-P, ATL</td>
</tr>
</tbody>
</table>
Tool Support for UWE

- **Goal**
  - support of **UWE notation** for design of Web applications
  - separation of concerns (navigation, process, presentation, …)
  - implementation of **UWE development process** allowing for semi-automatic model generation
  - support of **model validation** checking models consistency
  - code generation
ArgoUWE & MagicUWE

- Extension of ArgoUML
  - advantages
    - open source UML CASE tool
    - easy extension of design critics feature for checking model consistency
    - XMI output as basis for code generation
  - disadvantages
    - still based on UML 1.x
    - inherited usability problems
  - Plugin for ArgoUML 0.16

- Extension of MagicDraw
  - advantages
    - based on UML 2
    - UML profile definition support
    - XMI output
  - disadvantages
    - commercial tool
    - distribution of UML profile
  - Plugin for MagicDraw 15.0

- Plugins for other CASE tools
  - Rational Software Modeller
  - interest in further extensions supporting UWE notation and model transformations
MagicUWE

Web Software & MDD
UWE Approach
Modelling Language
Metamodel
Development Process
Tool Support

Madrid – 2008 – Nora Koch
Development Environment

- Eclipse based
  - PIM2PIM transformations (ATL)
  - PIM2PSM & PSM2Code (ATL)
  - written in ATL
- Spring Framework
- Java Server Pages
## Evolution of UWE

<table>
<thead>
<tr>
<th></th>
<th>Modelling Language</th>
<th>Metamodel</th>
<th>Development Process</th>
<th>Tool support</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Current Status</strong></td>
<td>UML profile for content, navigation, process, abstract presentation, adaptation</td>
<td>Conservative extension of UML metamodel (light-weight) and profileable</td>
<td>Different languages for model transformations</td>
<td>CASE tool ArgoUWE, Eclipse plugins Spring JSP</td>
</tr>
<tr>
<td><strong>Ongoing Work</strong></td>
<td>Extension for concrete presentation, requirements and services</td>
<td>Evolving to include modelling elements for concrete presentation and services</td>
<td>Model transformation language ATL</td>
<td>Plugin for MagicDraw Eclipse plugins JSF</td>
</tr>
<tr>
<td><strong>Future Work</strong></td>
<td>Extension for Web 2.0</td>
<td>Evolving to include modelling elements for Web 2.0</td>
<td>Model transformation language QVT (if standardized)</td>
<td>Plugins for other CASE tools Editors for development environment</td>
</tr>
</tbody>
</table>
Literature

- **Web Engineering: Modelling and Implementing Web Applications**

- **Web Engineering: Systematic Development of Web Applications**
  Gerti Kappel, Birgid Pröll, Siegfried Reich, Werner Retschitzegger (eds.)
  dpunkt-verlag (German version), 2003,
  John Wiley & Sons (English version), 2006.

- **Model-Driven Generation of Web Applications in UWE**
  Andreas Kraus, Alexander Knapp and Nora Koch
  3rd International Workshop on Model-Driven Web Engineering (MDWE 2007), Como, Italy

- **Metamodelling the Requirements of Web Systems**
  María José Escalona and Nora Koch
  2nd International Conference on Web Information Systems and Technologies (WebIST’06), Setubal,

- **Modelling Adaptivity with Aspects**
  Hubert Baumeister, Alexander Knapp, Nora Koch and Gefei Zhang
  5th International Conference on Web Engineering (ICWE 2005), Sydney, Australia, LNCS 3579, 406-416, 2005.
Muchas gracias!

Nora Koch
www.pst.ifi.lmu.de/people/staff/koch
nora.koch (at) pst.ifi.lmu.de

UWE
www.pst.ifi.lmu.de/projekte/uwe
uwe (at) pst.ifi.lmu.de

MDWEnet
www.pst.ifi.lmu.de/projekte/mdwenet
mdwenet (at) pst.ifi.lmu.de