Chapter 7. Requirements Engineering for Web Applications

Reyes Grangel Seguer

EI1030. Analysis of Inf. Systems / EI1032. Software Analysis / MT1046. Integrated Inf. Systems
Computer Engineering / Computational Mathematics
Dept. of Computer Systems and Languages

March 20, 2019

Outline

Outline

Main References

- Paper: [escalona 2004]
- Slides: [koch 2007]
- Paper: [Jeary 2009]

Differences between Applications and Web Applications [escalona 2004]

- Different kinds of stakeholders participate in the development process
- The main features of these systems are
  - Navigational structure
  - User interface
  - Personalisation capability

Differences between Applications and Web Applications [koch 2007]

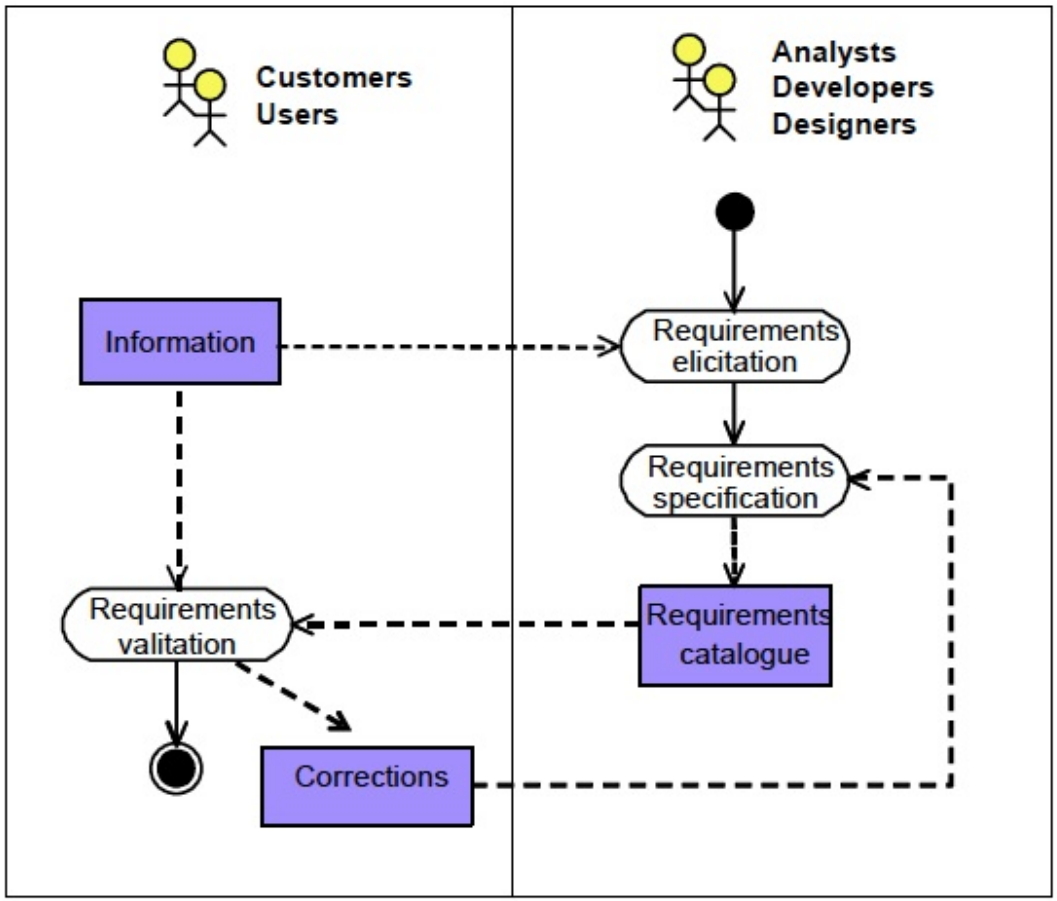
- User Domain
  - user divorced from development
  - no traditional entry or exit point
  - technology more visible to the user
  - high reliance on user interface
  - non-functional requirements primacy
  - volatility in user requirements
- Developer Domain
  - multidisciplinary teams
  - aesthetic and cognitive differences
  - developer inexperience
  - uncertainty
  - rapidly changing technology
  - lack of useful methods
- Environment
  - tight linkage between business architecture and the technical design
  - impact of legacy systems
  - aggressive release demands
  - immaturity of Web development techniques
  - development changes the business model
  - highly competitive, market environment
  - fine grained evolution and maintenance

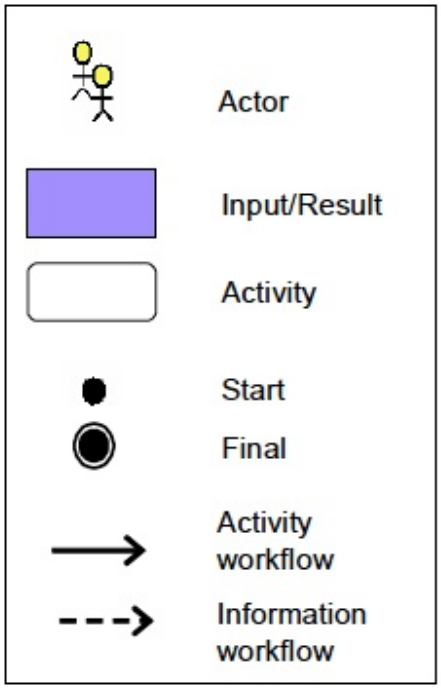
S. Jeary & K. Phalp 2004

Classification of Requirements for Web applications

- **Functional**
  - Data requirements
  - Interface requirement
  - Navigational requirements
  - Personalisation requirements
  - Transactional requirements
- **Quality Requirements, better quality**
  - Availability
  - Performance
  - Security of transactions
  - Reliability
  - Usability
    - Understandability
    - Learnability
  - Changeability
  - Portability
  - Ethic issues

Requirements Engineering Process





Orientation of Web Methodologies

- **Process-oriented**: the approach clearly describes the steps to follow (+), the process without details (o), or does not indicate any process at all (-)
- **Technique-oriented**: the approach clearly depicts the techniques and the way to apply them (+), it enumerates the techniques to apply (o), or it does neither propose any concrete technique nor references any general techniques (-)
- **Product-oriented**: the approach clearly describes the structure of the product to be produced (+), it describes the product without detailing its structure (o), or it does not give any indication about the resulting product (-)

Outline

Current Approaches

- **WSDM: Web Site Design Method**
- SOHDM: Scenario-based Object-Oriented Hypermedia Design Methodology
- RNA: Relationship-Navigational Analysis
- HFPM: Hypermedia Flexible Process Modeling
- **OOHDM: Object Oriented Hypermedia Design Model**
- **UWE: UML-based Web Engineering**
- W2000
- **WebML: Web Modeling Language**
- NDT - Navigational Development Techniques
- Design-driven Requirements Elicitation
- ...

Main Features

- User-centered approach for the development of Web sites
- Models the application based on the information requirements of the users' groups

Phases

- **User modeling**: User groups
  - Users' classification: conceptual maps of roles and activities for representing relationships between stakeholders and the business process
  - Users' group description: information, functional and security requirements for each user's plus a data dictionary
- Conceptual design: class diagram for static model and navigational model for possibilities of navigation
- Implementation design
- Implementation

Main Features

- Method for the development of Web applications
- Mainly technique-oriented
- Proprietary notation for UIDs visual representation
- Tool: OOHDM-Web

Phases

- Phases
  - Conceptual model: class diagram (static aspect)
  - Navigational model: navigation class diagram and a navigation structure diagram
  - Abstract interface model: special technique named ADVs
  - Implementation
- Improvement: User Interaction Diagrams (UIDs)
  - Based on use cases (standard UML)
  - Refinement of use cases building UIDs
  - Interaction between users and system without considering specific aspects of the interface
  - Entry and exit points to UIDs
  - User input data
  - Communication to other UIDs

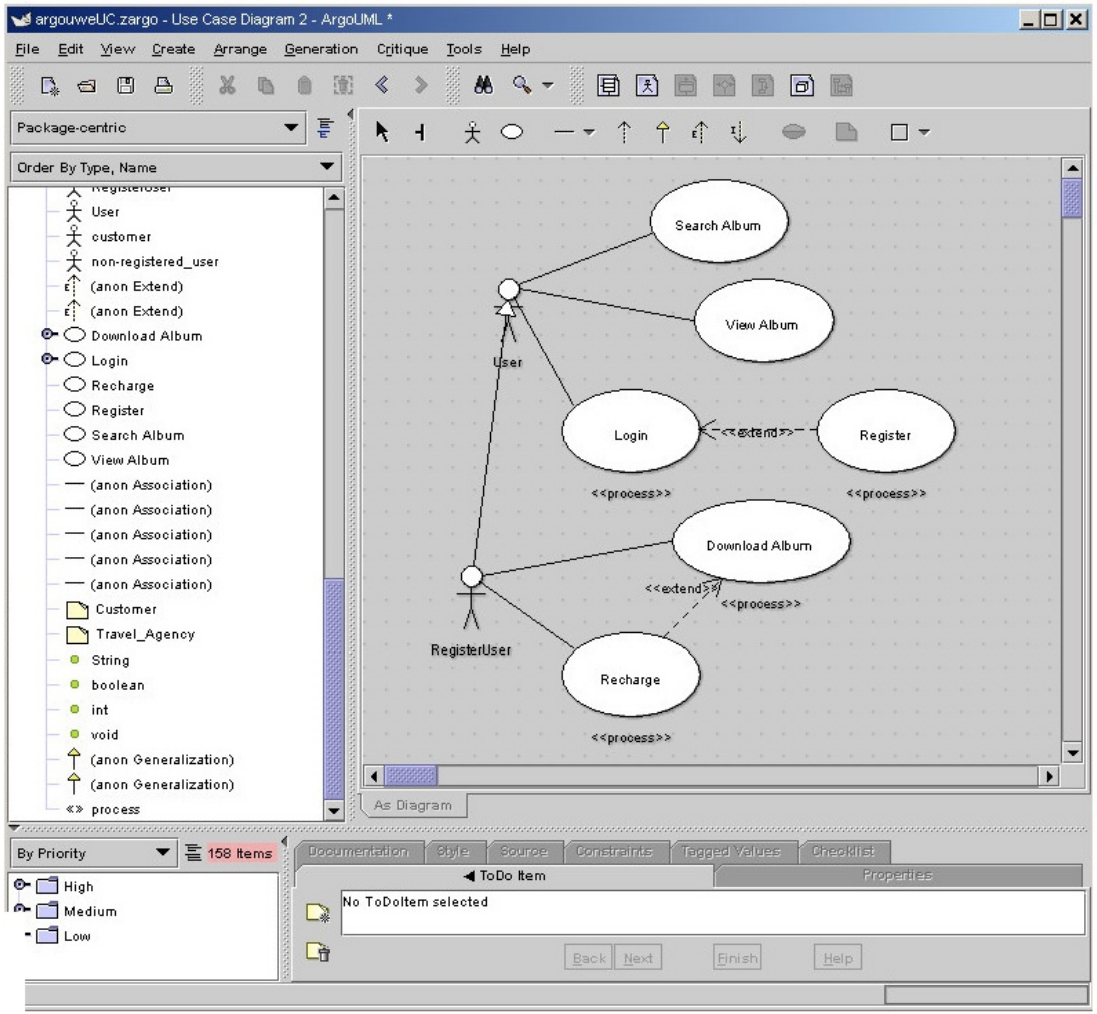
Main Features

- Websites: http://uwe.pst.ifi.lmu.de/index.html
- Methodological approach for the development of Web applications based on the Unified Process
- Semi-formal graphical specification
- Uses extension mechanisms provided by the UML stereotypes
  - Navigation with use case
  - OCL constraints
- Method specification: MOF Metamodel
- Advantage: use of OMG standards (UML, MOF, OCL, etc.)
- Basis for Design: transformations to design models
- Tool: MagicUWE (http://uwe.pst.ifi.lmu.de/toolMagicUWE.html)

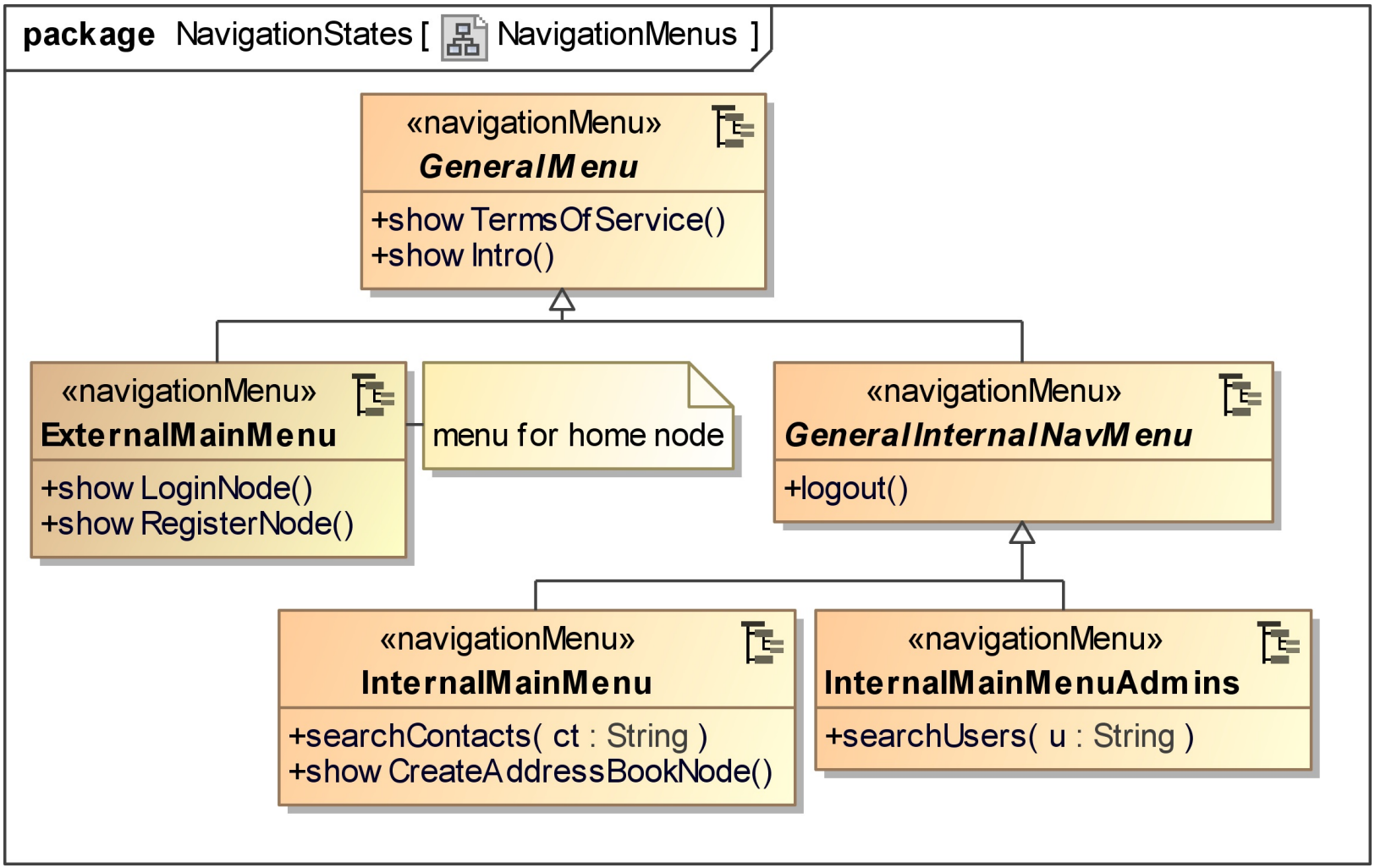
Phases

- Elicitation: interviews, questionnaires and checklists
- Specification: use cases, scenarios and glossaries
- Validation: walk-through, audits and prototypes

Tool (not longer maintained)



Example


package NavigationStates [NavigationMenus]
«navigationMenu»
GeneralMenu
+showTermsOfService()
+showIntro()
«navigationMenu»
ExternalMainMenu
+showLoginNode()
+showRegisterNode()
menu for home node
«navigationMenu»
GeneralInternalNavMenu
+logout()
«navigationMenu»
InternalMainMenu
+searchContacts(ct : String)
+showCreateAddressBookNode()
«navigationMenu»
InternalMainMenuAdmins
+searchUsers(u : String)

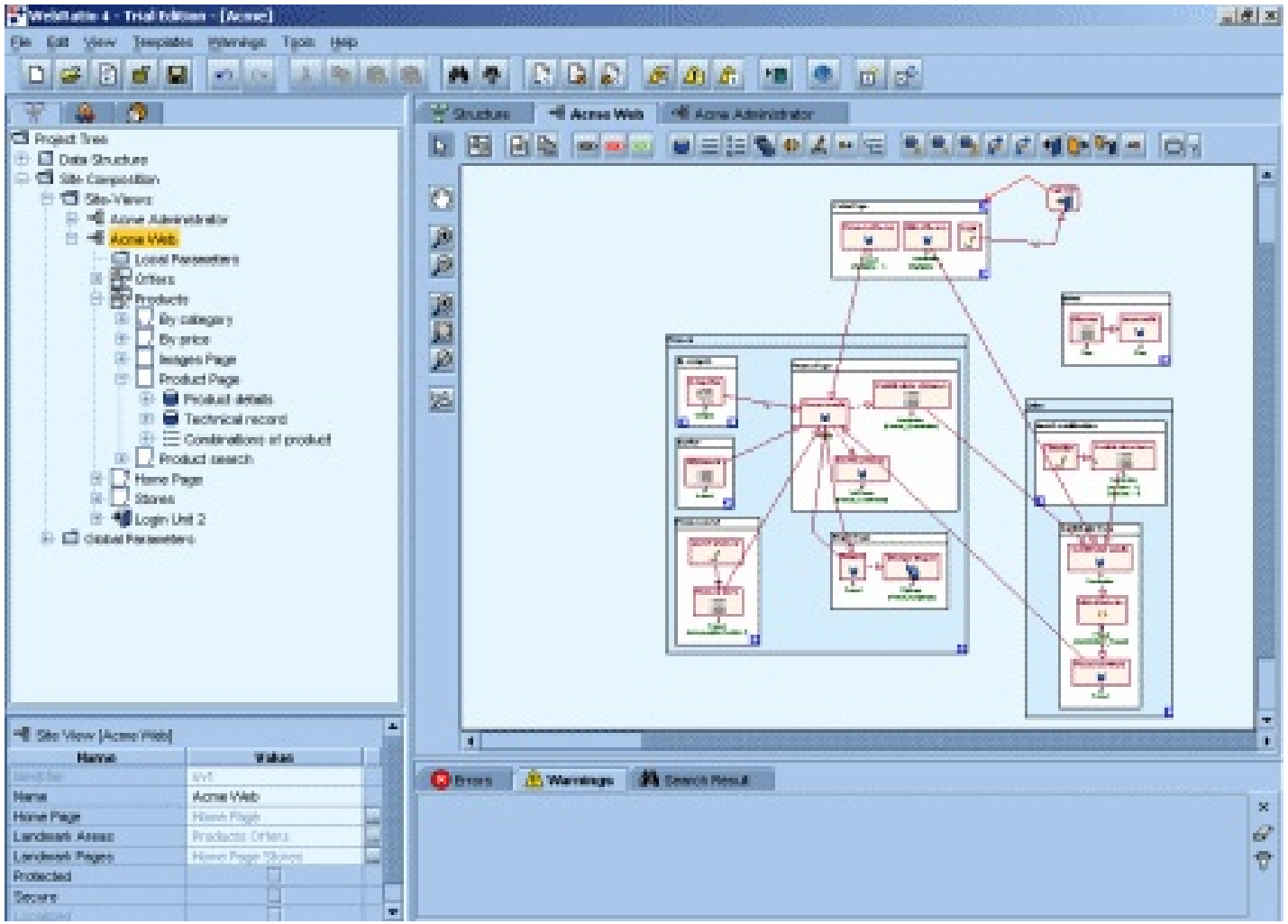
Main Features

- Websites: `http://webml.org/webml/page1.do`
- High-level specification language for hypermedia applications
- Follows the style of both, Entity-Relationship and UML offering a proprietary notation and a graphical representation using the UML syntax
- Tool: WebRatio (`http://www.webratio.com/portal/content/en/home`)

Phases

- Requirements collection: interviewing and analysis of documentation
- Requirements specification (requirements analysis): use case specification supplemented with a semi-structured textual description, use of activity diagrams for complex use cases, template based description and mock-ups (sketches) for site view, and acceptance tests to check non-functional requirements.
- Data design
- Hypertext design

Tool



Outline

By Types of Requirements

	Data Req.	User Interface Req.	Navigational Req.	Adaptive Req.	Transactional Req.	Non-Functional Req.
WSDM	✓			✓		✓
SOHDM	✓	✓			✓	
RNA	✓	✓	✓		✓	
HFPM	✓	✓	✓			✓
OOHDM	✓	✓	✓			
UWE	✓	✓	✓	✓		✓
W2000			✓	✓	✓	
WebML	✓	✓		✓		✓
NDT	✓	✓	✓	✓	✓	✓
DDDP	✓	✓	✓	✓	✓	✓

By Requirements Engineering Process

		WSDM	SOHDM	RNA	HFPM	OOHDM	UWE	W2000	WEbML	NDT	DDDP
Capture	Interviewing	✓		✓			✓		✓	✓	✓
	JAD									✓	
	Brainstorming									✓	
	Concept Mapping	Role-Activity									
	Use Cases Modeling					✓					
	Questionnaire/Checklist						✓				
	Sketching & Storyboarding								✓		
	Other Techniques		DFD						Document analysis		
Definition	Natural Language	✓		✓	✓				✓		
	Glossaries				✓		✓			✓	
	Templates/Patterns								✓	✓	
	Scenarios		SAC				✓				
	Use Cases Analysis				✓	✓	✓	✓	✓	✓	
	Formal Language										
	Prototyping										✓
	Other techniques		Event List		Interface Sketches	UIDs				BNL Phrases	
Validation	Review/Walk-through						✓			✓	
	Audit						✓				
	Matrix of trazability									✓	
	Prototyping				✓		✓				✓
	Other techniques								Acceptance Tests		

By Orientation

	Process-oriented	Technique-oriented	Product-oriented
WSDM	O	-	-
SOHDM	-	+	-
RNA	+	-	-
HFPM	+	O	+
OOHDM	O	+	-
UWE	+	O	O
W2000	O	O	-
WebML	O	O	+
NDT	O	+	+
DDDP	+	O	-