POLITECNICO DI MILANO



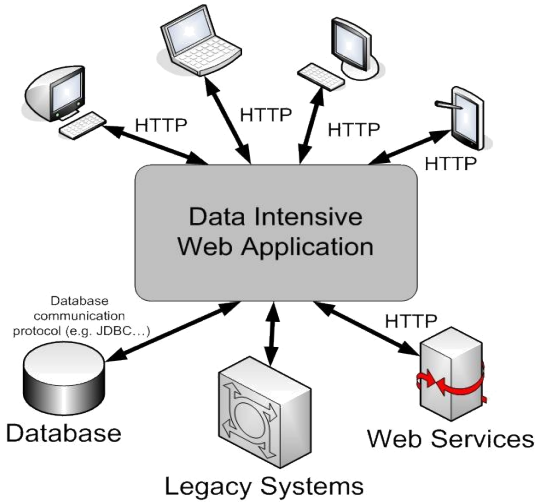


WebML: Model-driven design of Web Applications

Massimo Tisi

Stefano Ceri, Piero Fraternali, Aldo Bongio, Marco Brambilla, Alessandro Bozzon

- **WebML**: **Web M**odeling **L**anguage
  - Structured approach for the design of Data-intensive Web applications
  - Book: “Designing Data-intensive Web Applications”, Ceri et Al., 2002
  - Patent in Europe and U.S.



- WebRatio
  - A commercial tool based on WebML
  - A university spin-off: WebModels

- Example: Acer-EMEA www.acer-euro.com
  - 37 countries
  - 22 multi-lingual applications
  - >600 page templates
  - >3500 queries
  - from design to deployment in 8 weeks

WEB RATIO®
You think You get

Motivations

- Data-intensive Web site design and implementation are complex processes
  - based on methodologies borrowed from different sectors
  - involving different actors (DB, software eng., designers...)
- Complexity of modern Web applications (e.g., multi-device output)
- Always evolving
  - standards
  - best practices
  - architectures

WebML Purpose

- WebML provides a **structured approach** to the design of Data-intensive Web sites
  - navigational interface
  - browsing and management of data
- A set of **domain specific models** helps designers in high-quality Web sites production
- **Separation of concerns** is enforced
  - database design
  - application design
  - business logic development
  - presentation and style design

Advantages of Modeling approach

A rigorous modeling approach:

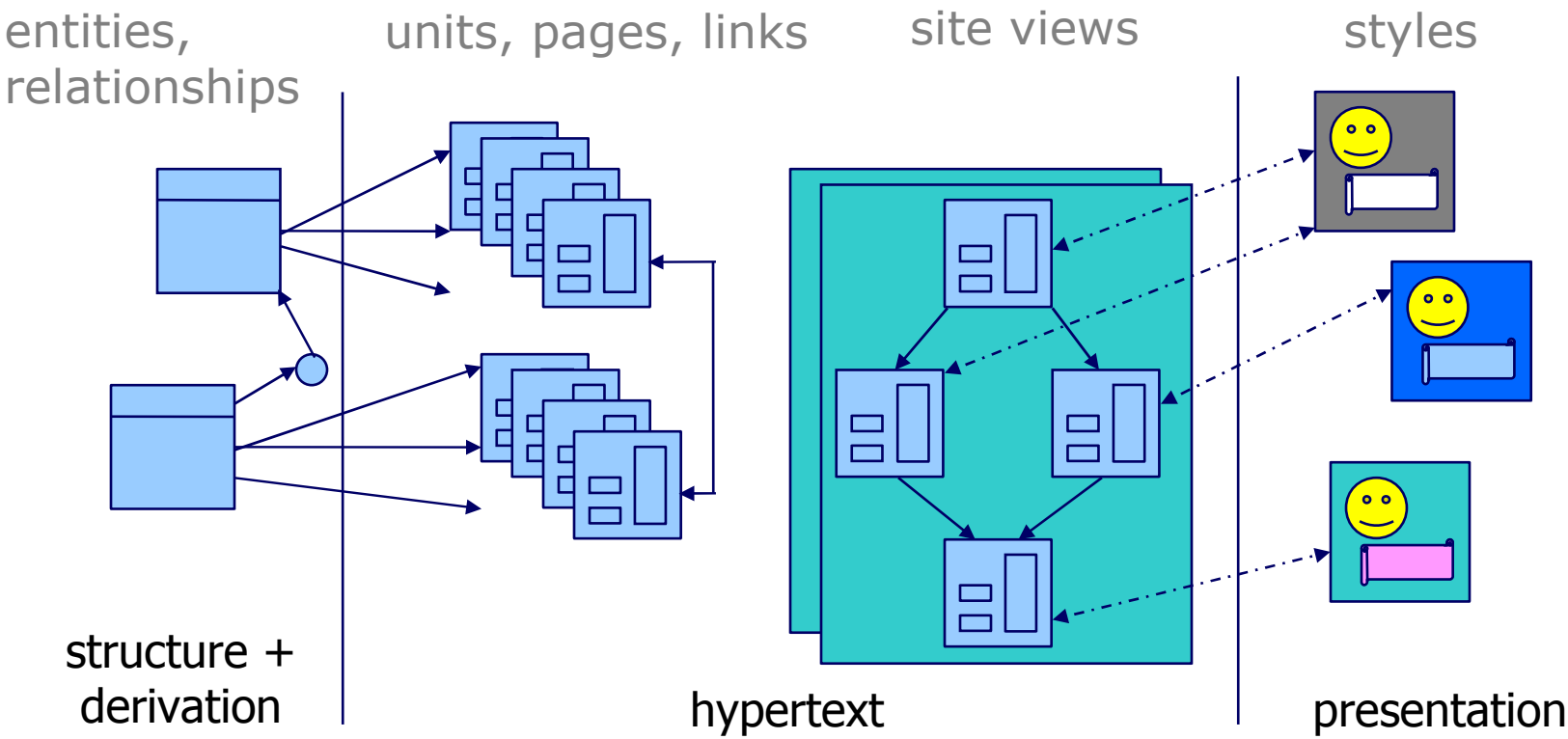
- Can reduce development efforts (cost and time)
- Allows a more structured development process
- Produces more usable and coherent final results
- Design models are self-documenting and always up-to-date projects

Immediate prototyping can be achieved

Requirements for Web modeling

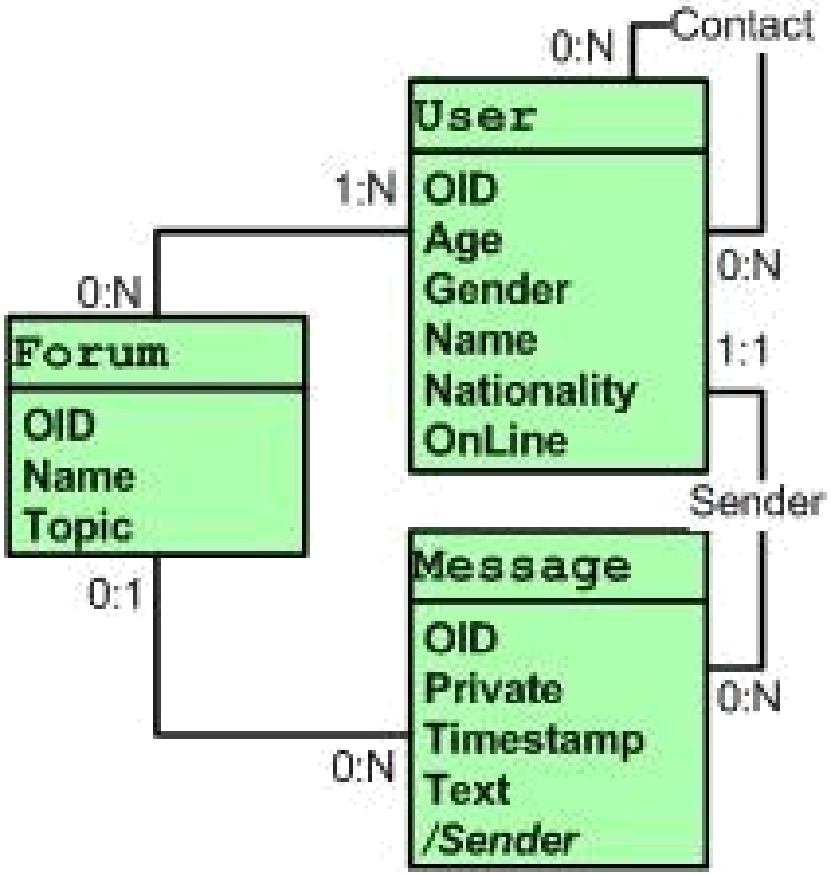
- Expressiveness
  - Real-life cases should be expressible
  - Frequently used design patterns should be captured
- Ease of use
  - Intuitive notation
  - Clear semantics
  - Consistency checks
- Implementability
  - Efficient mapping to physical data structures
  - Flexible code generation from behavioral specifications

Data Intensive Web Applications



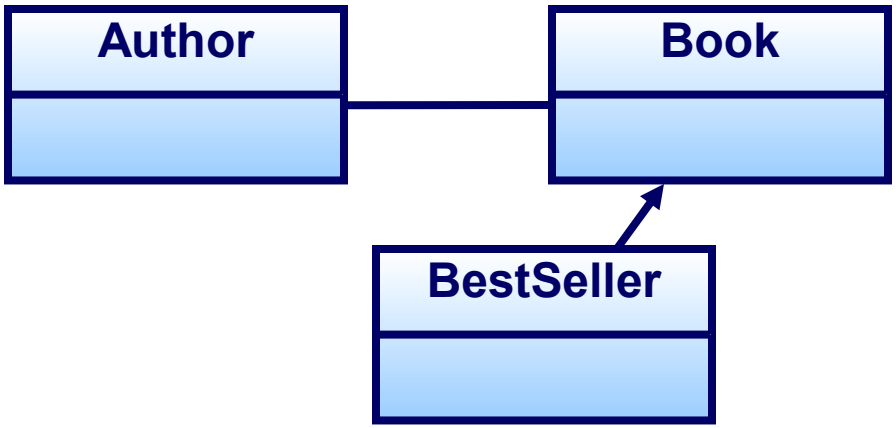
Simplified Entity-Relationship model

- Binary relationships between entities
- IS-A hierarchies
- Simple typed attributes in entities
- Derivation model can be applied for redundant data



Derivation Model

- Redundant data can be easily specified using a WebML-OQL (Object Query Language).
- E.g.:
  - Author.BooksNumber = count(self.Author2Book)
  - BestSeller := Book where Book.Sales > 50,000

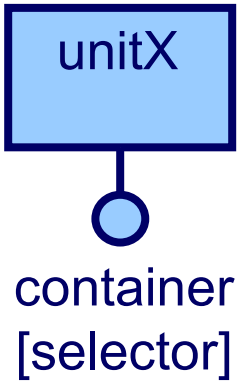


Hypertext Model

- Goals
  - Modelling at a high level:
    - the front-end of a dynamic Web application
    - the interactions with the back end business logic and data
  - Using a simple visual notation
  - Enabling automatic generation of dynamic page templates and business logic java classes

Content Units

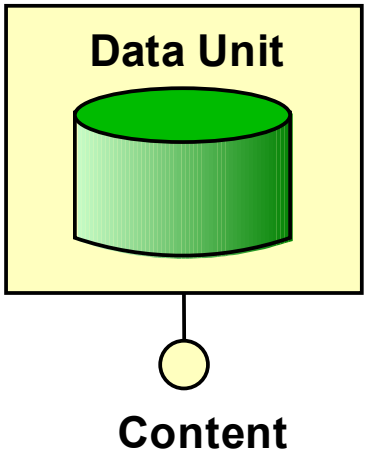
- A WebML unit is the atomic information publishing element



- It is a “view” defined upon a container of objects.
- E.g.:
  - All the instances of an entity
  - Instances of an entity that meet a selection condition called selector

Hypertext: examples of Content Units

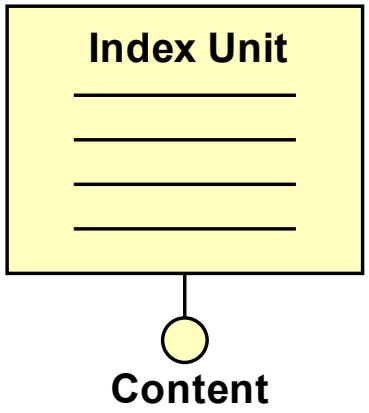
DATAUNIT



To publish information about A SINGLE object (e.g. Forum Message)

Forum Message
Sender:XXX Text:YYY Timestamp:

INDEXUNIT

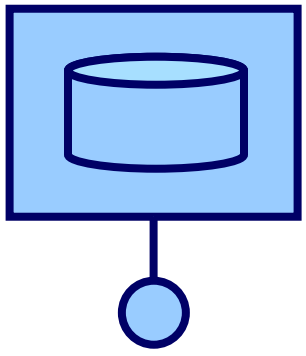


To publish a list of objects (e.g. Forum Messages)

Index of Messages
1. WebML? 2. WebRatio? 3. RIA?

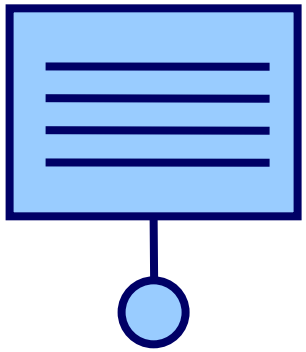
Basic Content Units

DATAUNIT



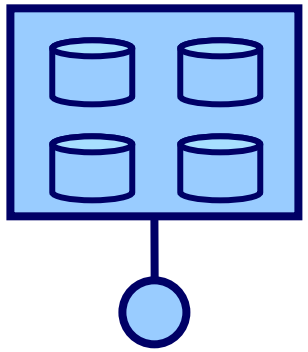
entity
[Selector]

INDEXUNIT



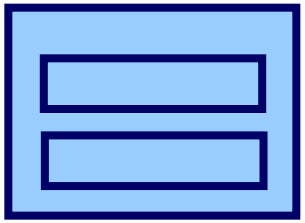
entity
[Selector]

MULTIDATAUNIT

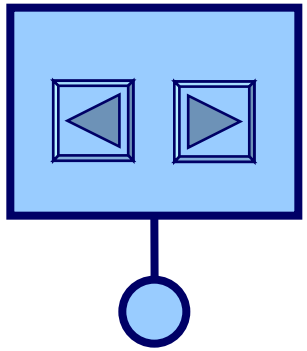


entity
[Selector]

ENTRYUNIT

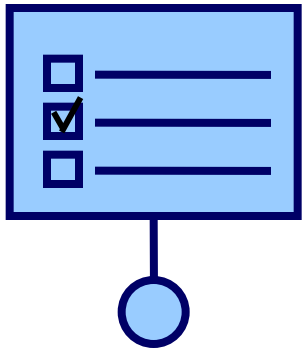


SCROLLERUNIT



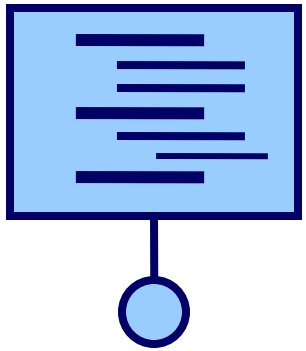
entity
[Selector]

MULTICHOICE



entity
[Selector]

HIERARCHICAL

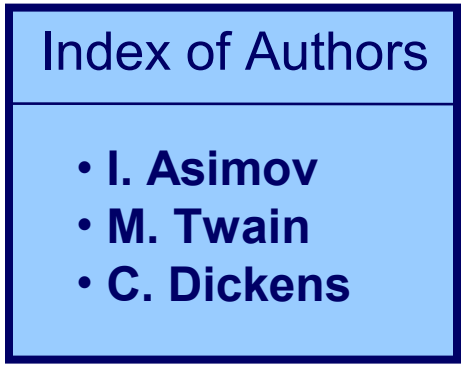


entity
[Selector]

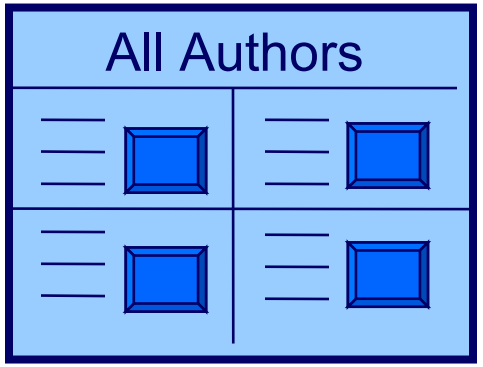
Meaning of Content Units

DATAUNIT

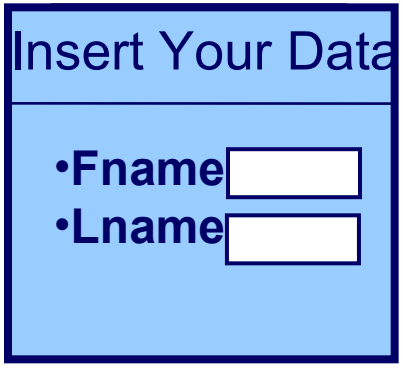
INDEXUNIT



MULTIDATAUNIT



ENTRYUNIT



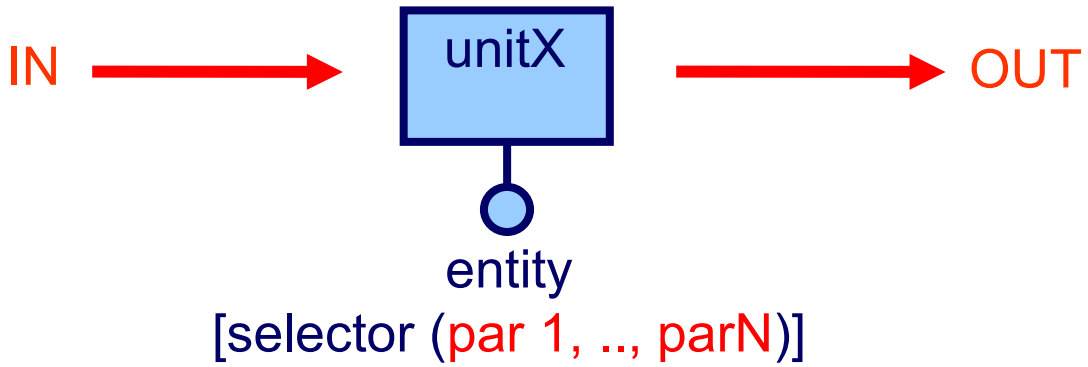
SCROLLERUNIT

MULTICHOICE

HIERARCHICAL

Content units computation

- A unit may need some “context” to be computed
- Each unit exposes input and output parameters

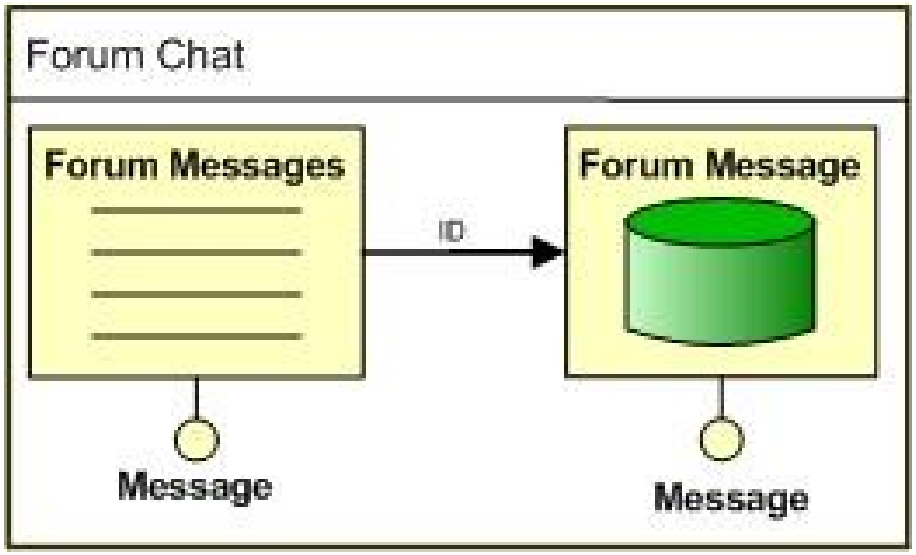


- A content unit is not computed until its needed context is available
  - Parameters pre-defined for the unit +
  - Other parameters required by the selector of the unit
- A content unit is recomputed (and eventually redrawn) each time its context changes
- Output can be used to compute other unit(s)

Hypertext: Pages

- **A Page is a structured container of units**
  - **Possibly structured in and/or sub-pages**
  - **Permits one to cluster related information for more efficient communication**

- E.g.:



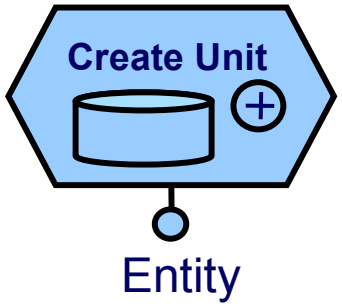
The index of messages and the selected message are shown together in the same page

Operation Unit

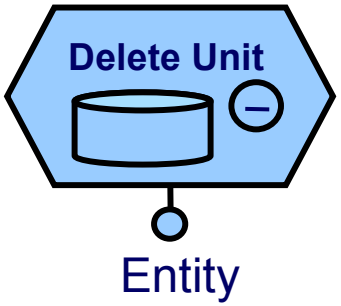
- Models a generic operation
- Built-in operation units:
  - Data manipulation
  - Session context management
  - User authentication
  - ...
- The predefined WebML units can be enriched by adding custom external operations (e.g. SendMail, …)

Data Management Operations

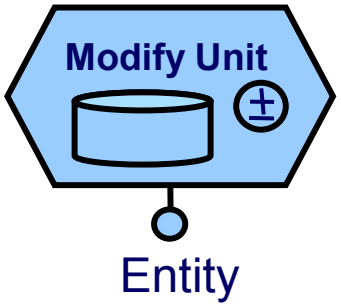
CREATE



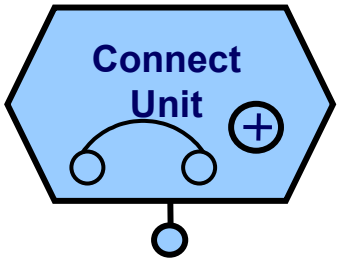
DELETE



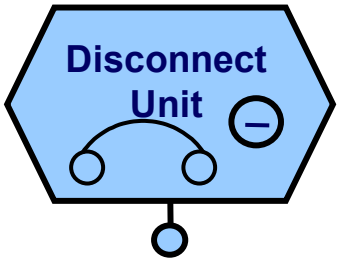
MODIFY



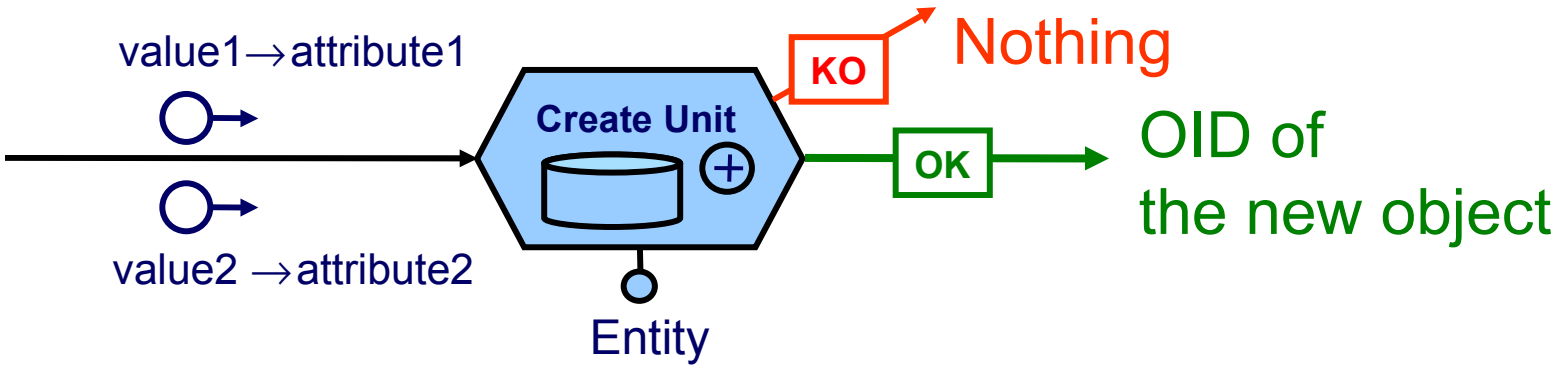
CONNECT



DISCONNECT

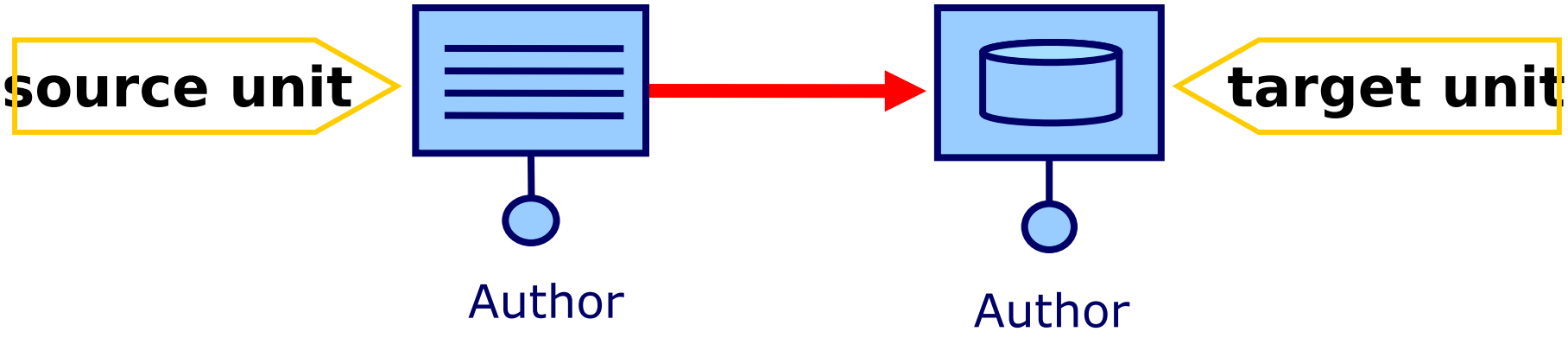


Operation units: computation



- An operation unit is computed each time an incoming link is activated
  - Exception for transport links
- Some operation units can be activated also by other events

(Normal) links



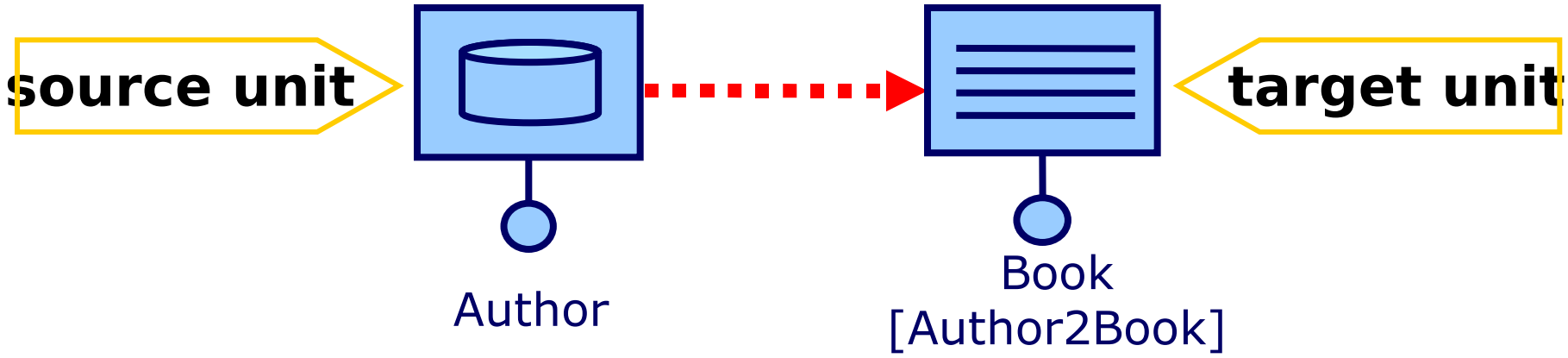
Semantics of links:

- Allowing the user to move from one place to another
  - rendering by means of anchors or submit buttons
- Transporting information from one place to another
  - context propagation by parameters coupling
- Activating an operation

Outgoing links from operation units two labels:

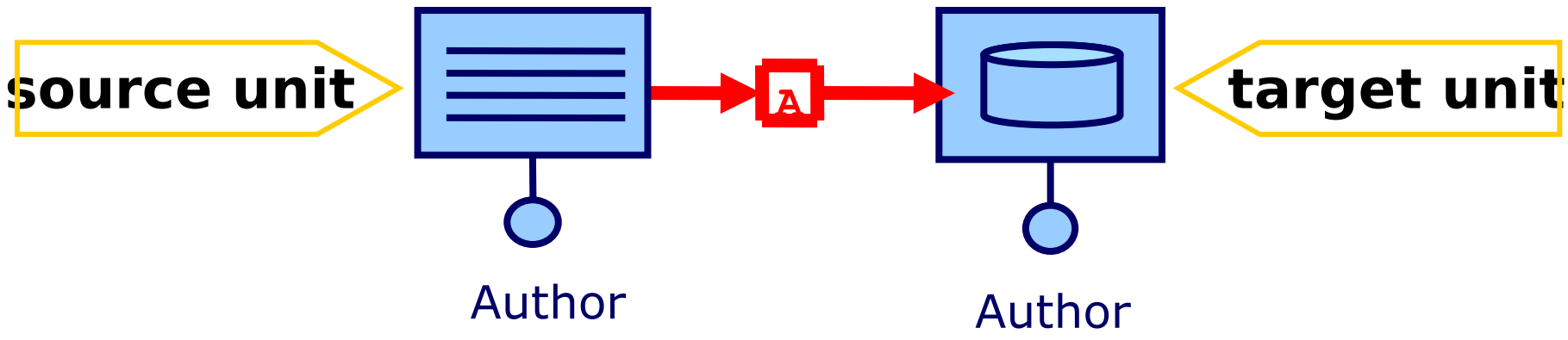
- **OK** link if the operation completes correctly
- **KO** link if the operation fails

Transport Links



- A transport link has a context that the source unit makes available to the target unit immediately after its computation, **without user intervention**
- The user **cannot** change the context and therefore the link is not rendered with an anchor

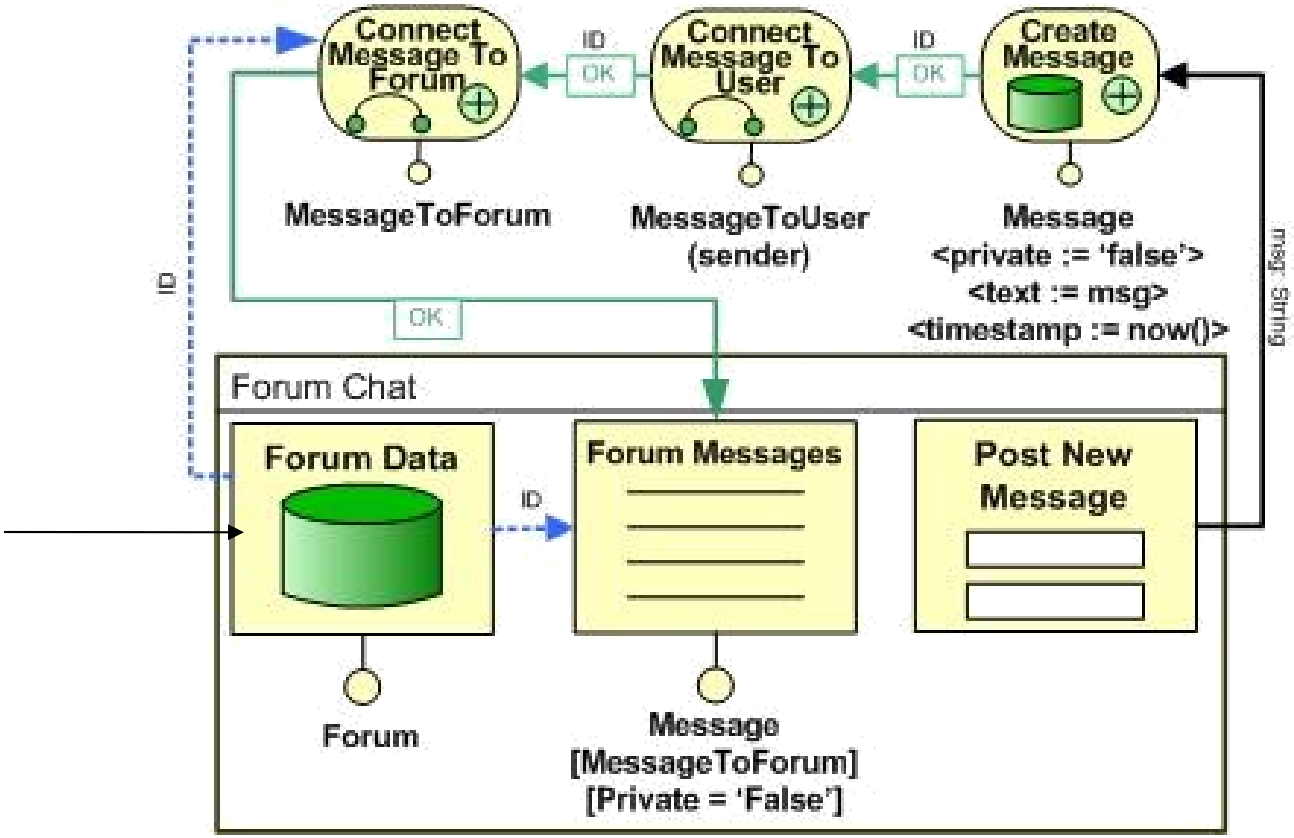
Automatic Links



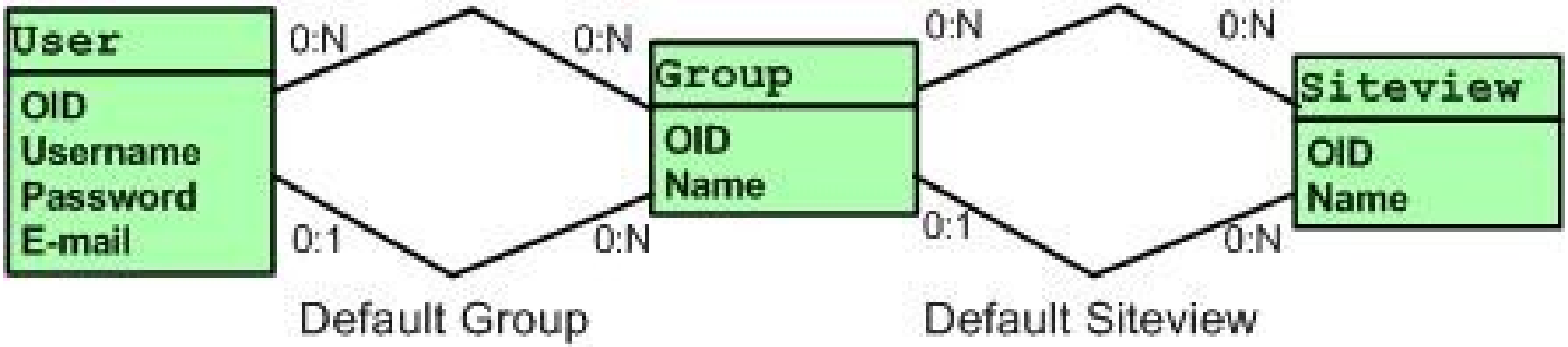
- An automatic link has both the behaviors of a normal link and a transport link
  - makes a context immediately available to the target unit
  - Is rendered and can be selected by the users for subsequent activations

Hypertext: WebML operations

- E.g.: creation of a message



- A Siteview is a set of pages that the user can experience as a whole Web site
- Different site views can be defined for different devices and different groups of users
- Thus, access control and multi-devices delivery is achieved



WebML: Other Constructs

Areas

Transactions

Master Pages

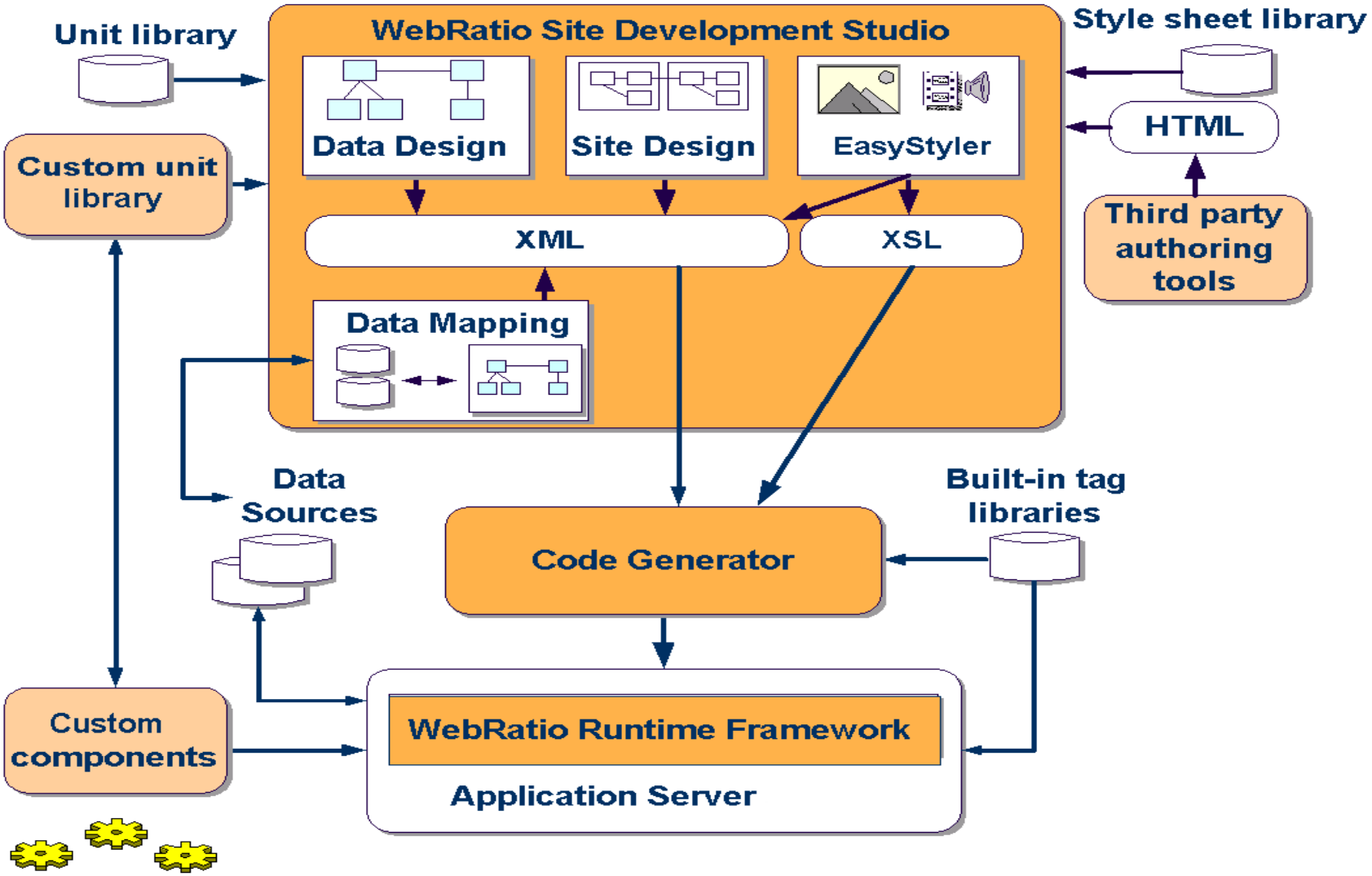
Alternatives

Global Parameters

Functionalities:

- Structure model design
- Data derivation (Derivation Wizard)
- Hypertext model design
- Consistency checks (warnings)
- Structure Mapping onto a datasource
- Units positioning in the pages
  - Grid for main content
  - Unlimited named locations
- Automatic web site generation with presentation styles
  - HTML + Custom Tags + CSS
- Compatibility with best selling tools for presentation editing
  - E.g., WebML extensions for Dreamweaver

WebRatio


Unit library
WebRatio Site Development Studio
Style sheet library
Data Design
Site Design
EasyStyler
HTML
Custom unit library
XML
XSL
Third party authoring tools
Data Mapping
Data Sources
Built-in tag libraries
Code Generator
Custom components
WebRatio Runtime Framework
Application Server

Web Ratio: a short demo

Ongoing work: Event Modeling

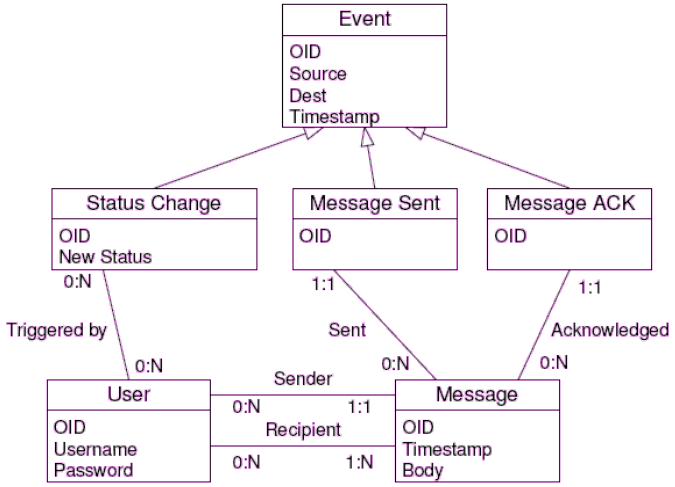
RIAs enforce decoupling of user interaction and browser requests

- Allow asynchronous polling (persistent connection technologies), non-interruptive application interaction
- Enable server-to-client communication (server PUSH)
- Make **event-driven** Web applications reality, e.g.:
  - instant messaging, shared calendars, online auctions

Extended WebML Notation for event modeling

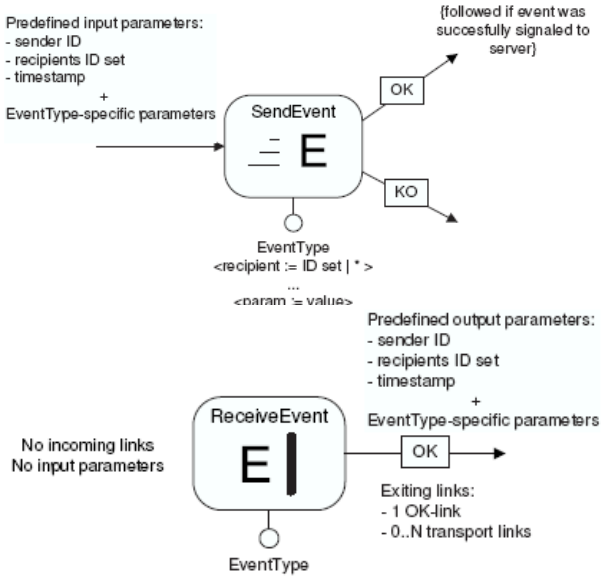
Data Model Extensions

- application-specific event types are represented by adding new entities to the data model
- all event types extend the predefined *Event* entity
- Specific event types can have relationships with application domain entities



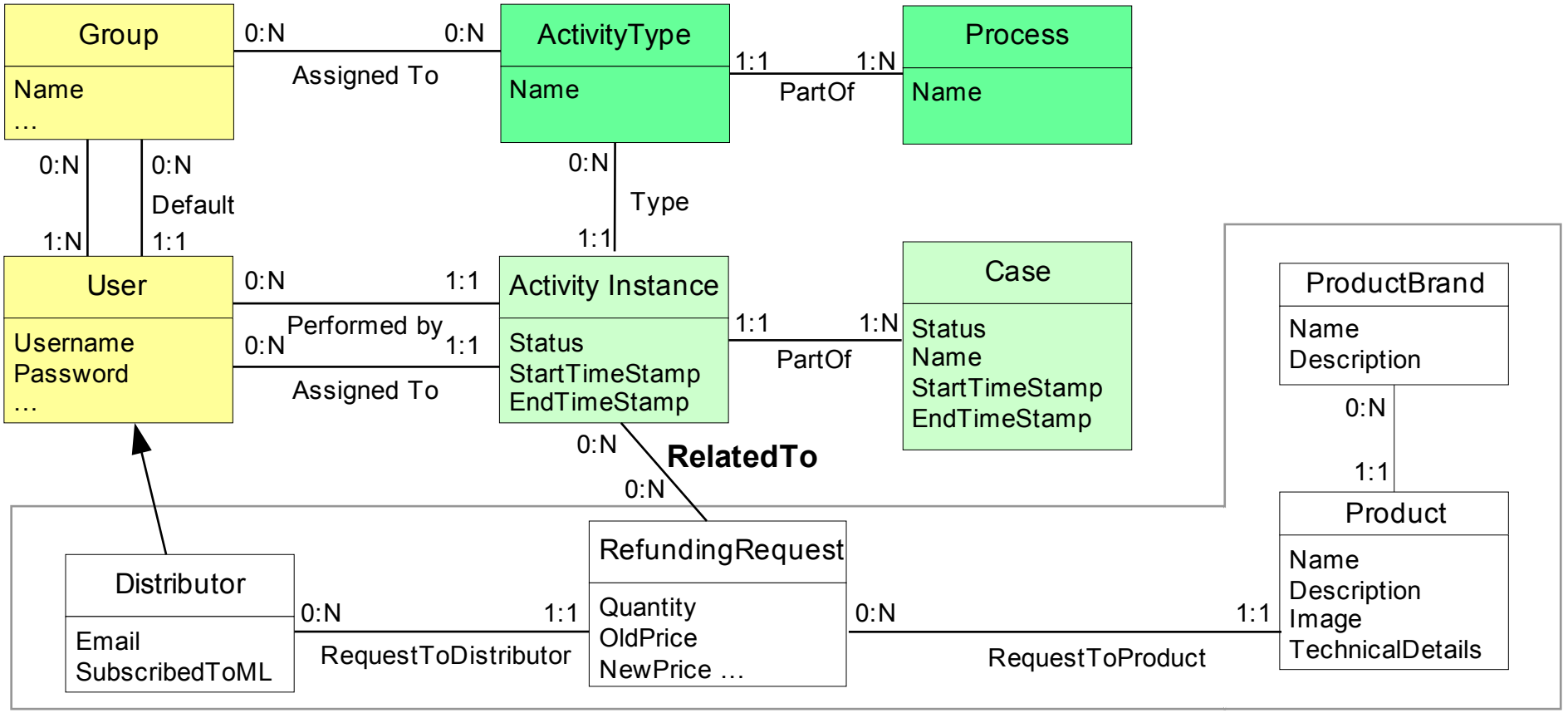
Composition/Navigation Model Extensions

- We extended the WebML hypertext model to support event notifications by means of two WebML operations:
  - *send event:* send an event notification to a (set of) recipient(s);
  - *receive event*: receive the notification and trigger a reaction;



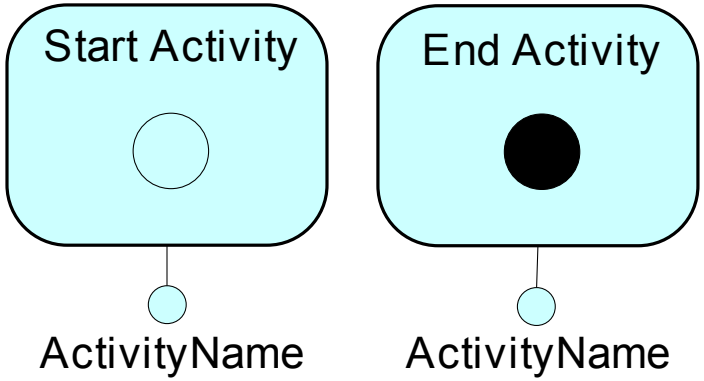
Extension of Data Design to capture processes

- Based on the WfMC concepts

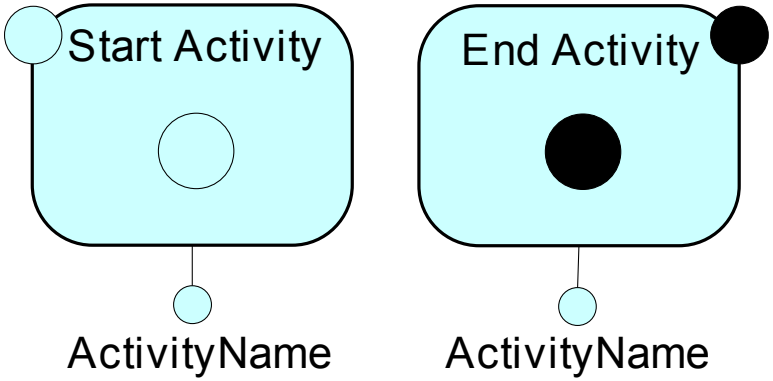


Extension of Hypertext Design to capture processes

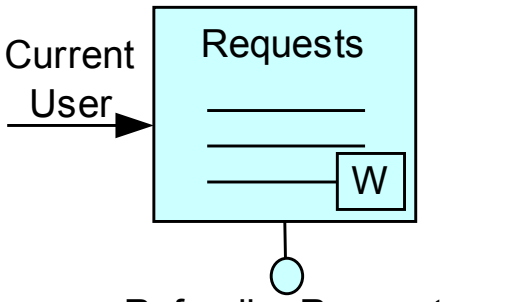
Activity delimiters



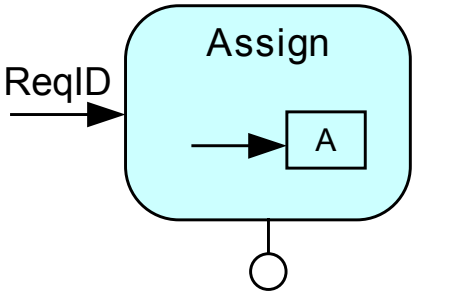
Process delimiters



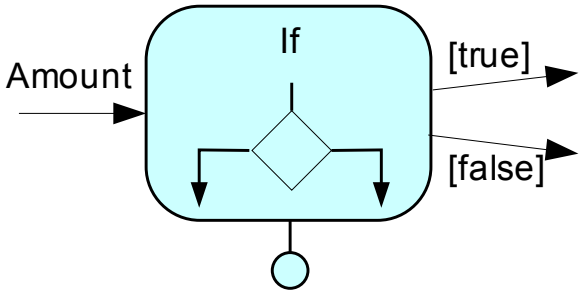
WF indexes



Assignements



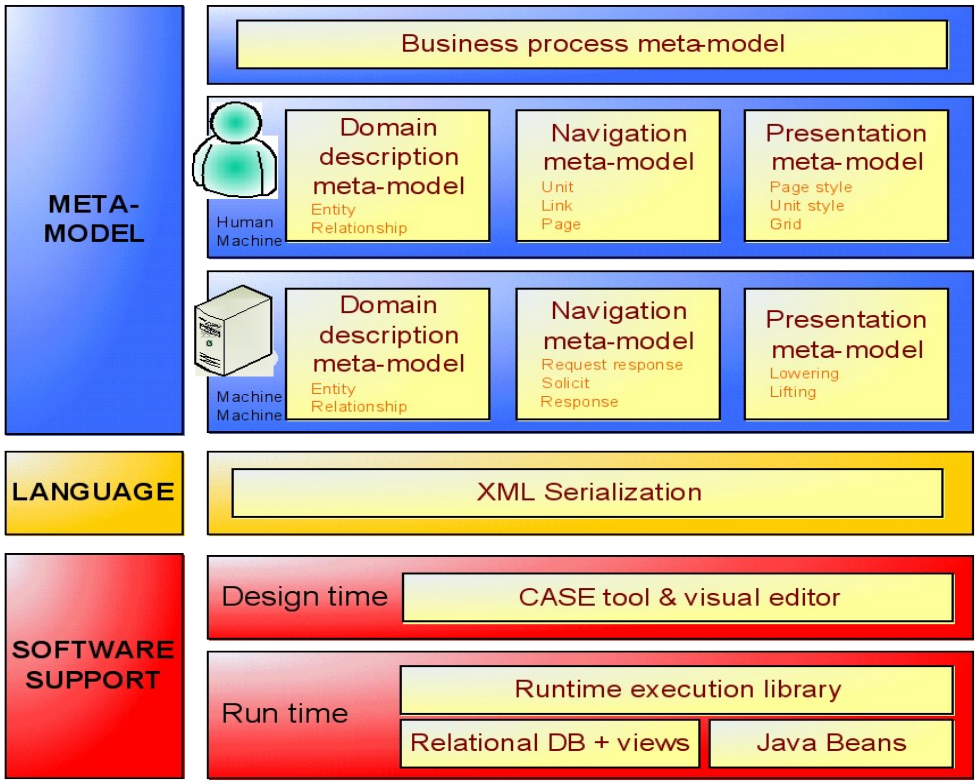
Conditions



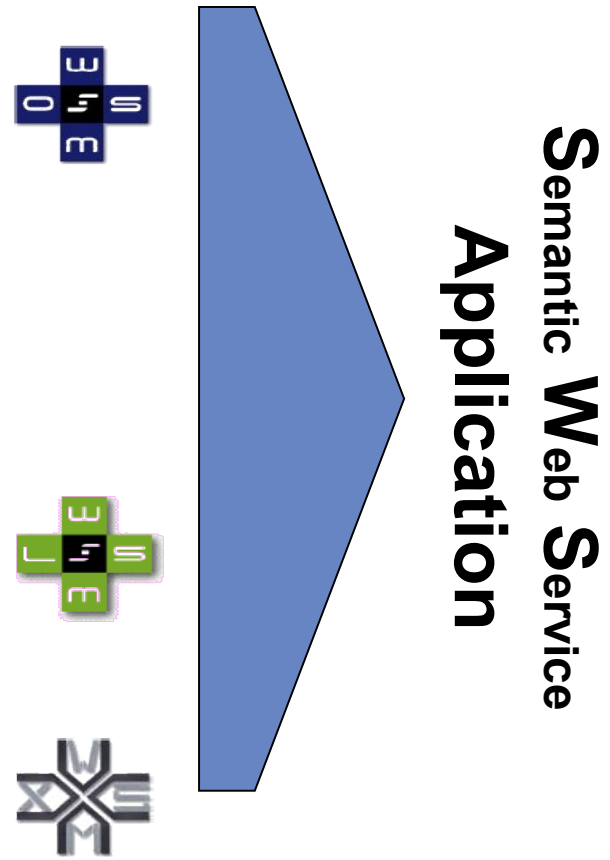
- Semantic Web Services (SWS) have a great potential
  - easy web service discovery
  - automatic web service integration
  - easy interoperability
  - …
- Till now SWS are rarely used in practice
  - annotations are an extra cost
- Software Engineering (SE) tools and methodologies can push the use of SWS
  - model driven development techniques can be improved to include annotation and generate Semantic Web Services

Business process modeling notations

WebML



SWS



The most complete solution presented at phase-II of SWS Challenge '06

- Integration with other modelling languages / models, tools (esp. UML, WFMC, BPEL, MDWEnet)
  - Production of a (part of) WebML model from other diagrams
  - Reverse transformation
  - Manage correspondences and check consistency
- Current approach:
  - Ecore metamodel
  - ATL transformations
  - ATL HOTs

Testing, Debugging, Metrics

- Code generation testing:
  - Transformation for test set generation
  - Coverage metrics
- Debugging
  - Adding traceability to the code-generation transformation
  - Adding a debugging environment to WebRatio
- Metrics
  - Transformation for functional size metrics

Thanks