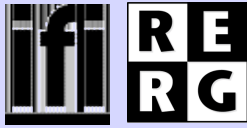


Experimental Validation of Aspect-oriented Language Constructs in ADORA

Silvio Meier

smeier@ifi.unizh.ch



University of Zurich
Department of Informatics
Requirements Engineering Research Group

Schedule

- Motivation for this experiment
- Introduction into Aspect-Oriented Software Development
- Introduction into ADORA
- Aspect-Oriented extension of ADORA
- Weaving semantics
- Demonstration of the tool / Hands-on
- Validation
- Discussion

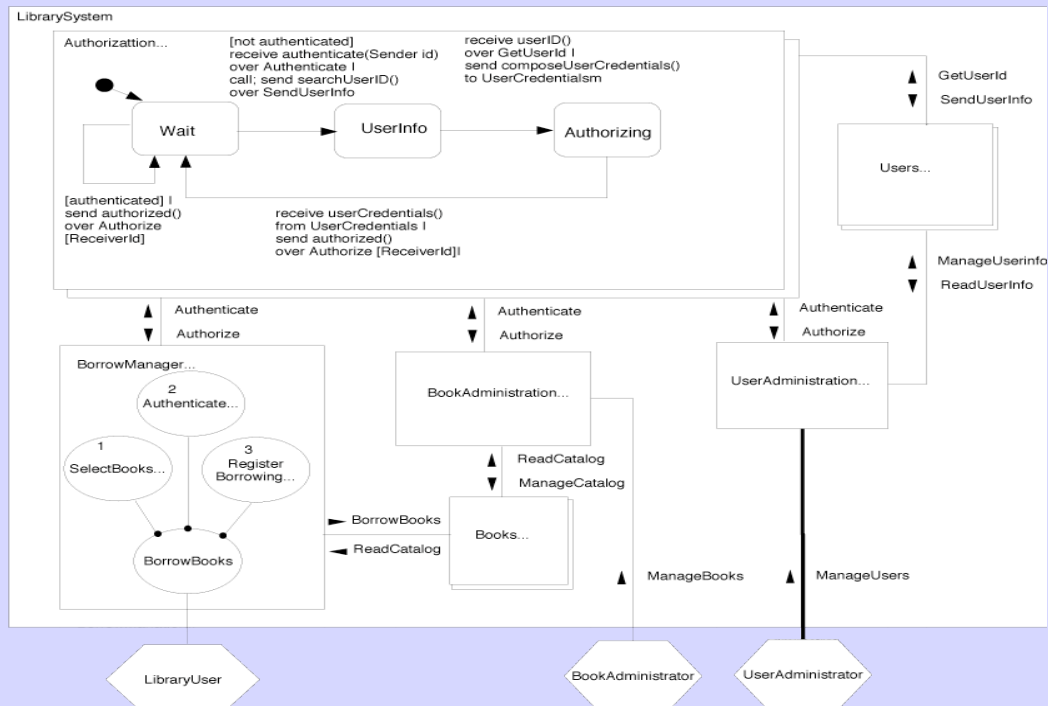
Motivation for this Validation

- Aim: to find the evidence for supporting the hypothesis that ...
 - ... Aspect-oriented requirements models increase the understandability in some situations and in others they do not.
 - ... Thus, switching between the conventional modeling view and ao view must be possible.
- Prerequisites: Repetition and introduction of ADORA, aspect-oriented ADORA and the ADORA tool.

Introduction into ADORA

- **Integrated modeling language** (integrated view and meta model)
- Basic Structure: **Object Model** - Enables integrated modeling
- **View Concept**
 - Basic view: can be combined with additional views.
 - Structural view (Associations)
 - Behavior
 - Context
 - User/Scenario
 - Functional view is not combined with the other views.
 - Partial views are possible, i.e. hiding arbitrary elements of a view.
- **Partial Models**
 - Intentionally incomplete models are marked as such

ADORA Example – A Library System



17. Dez. 2007

Validation of Aspect-Oriented ADORA Language Constructs / (c) Silvio Meier

5

Functional Specification

functional specification

requires LibrarySystem.User.userinfo;

attributes msg : message;

operation authenticate(user : string, pw : string)

statements

send authorized() **to** msg.msgAnswerTarget

end operation authenticate

end functional specification

17. Dez. 2007

Validation of Aspect-Oriented ADORA Language Constructs / (c) Silvio Meier

6

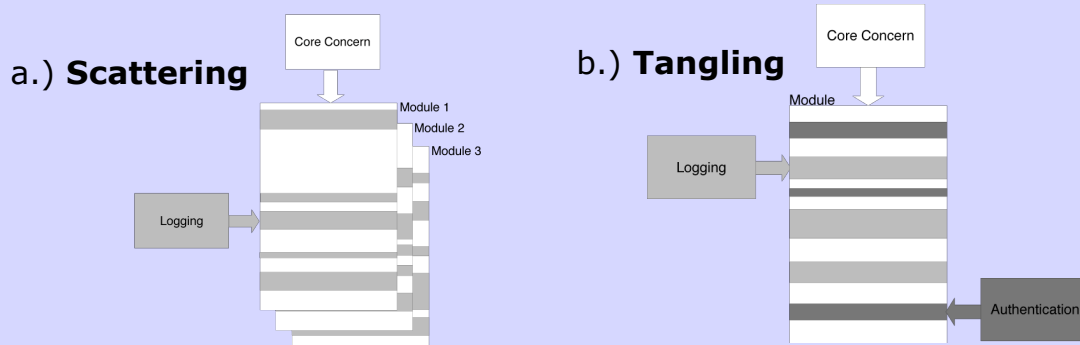
Introduction Aspect-Orientation

- Crosscutting Concerns
- Resulting code is complex (see below)!
- Maintenance: difficult!

```
public void transfer(float amount, BankAccount account) {
    Logger.log("ueberweisen called with" + amount + " " + account);
    if (amount <= 0) {
        RuntimeException ex = new RuntimeException(
            "Programfehler, negativer Betrag");
        Logger.log("Exiting ueberweisen with Exception " + ex);
        throw ex;
    }
    if (account == null) {
        RuntimeException ex = new RuntimeException(
            "Programfehler, Konto == null");
        Logger.log("Exiting ueberweisen with Exception " + ex);
        throw ex;
    }
    sum = sum - amount;
    try {
        account.deposit(amount);
        isPossible();
    } catch (RuntimeException ex) {
        Logger.log("Exiting ueberweisen with Exception " + ex);
        throw ex;
    }
    Logger.log("Exiting ueberweisen normally");
}
```

Aspect-Orientation: Problems

- Problem: One-Dimensional Decomposition
- Results in:
 - **Tangling** and
 - **Scattering** in the resulting code of the system

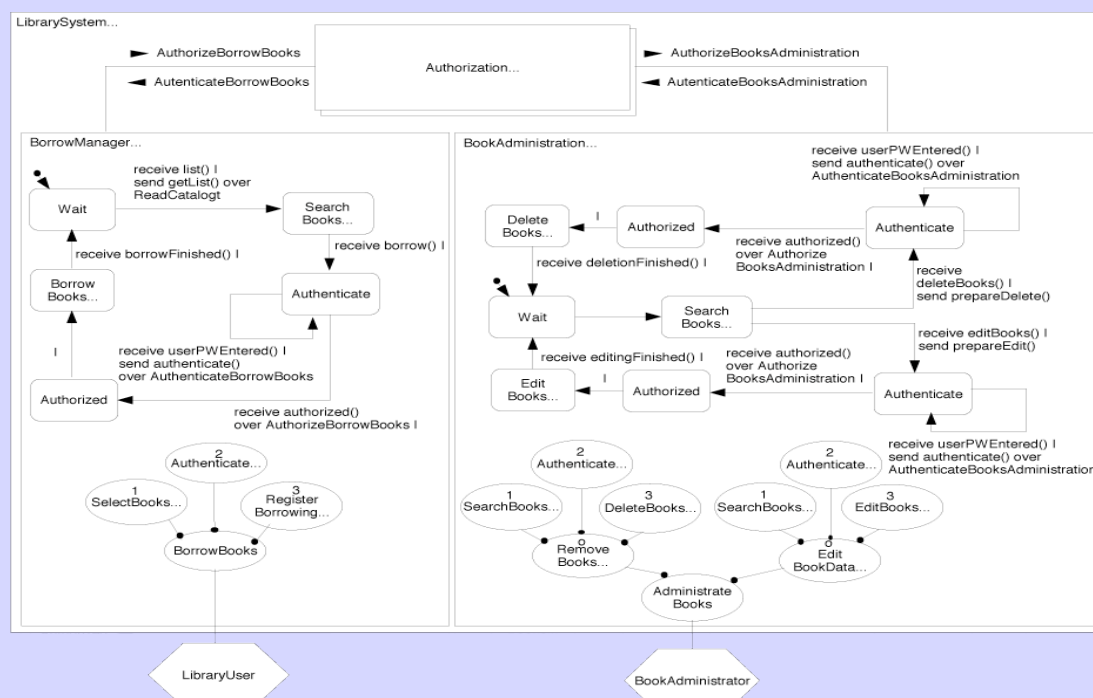


- Crosscutting concerns are e.g. **Logging** and **Security** issues

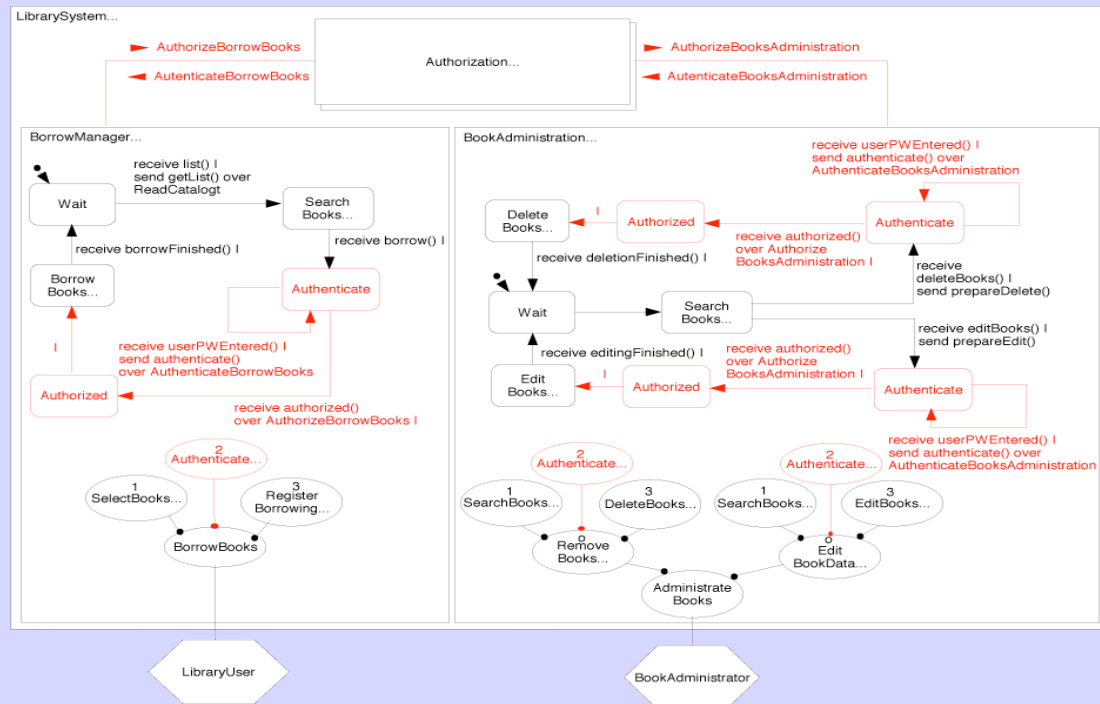
Aspect-Orientation: Addressed Problems

- Aspect-Oriented Programming (AOP):
 - **New modularization dimension: Aspects**
 - Modularized crosscutting concerns at the programming level
- **Crosscutting concerns** occur **already in earlier phases**:
- **AO support for early phases**: Requirements Phase, Architectural Phase
 - An AO modeling language is needed. There are modeling approaches but they do not satisfy completely!
 - Extension of ADORA

Aspect-Orientation: Introduction Example - 1



Aspect-Orientation: Introduction Example - 2

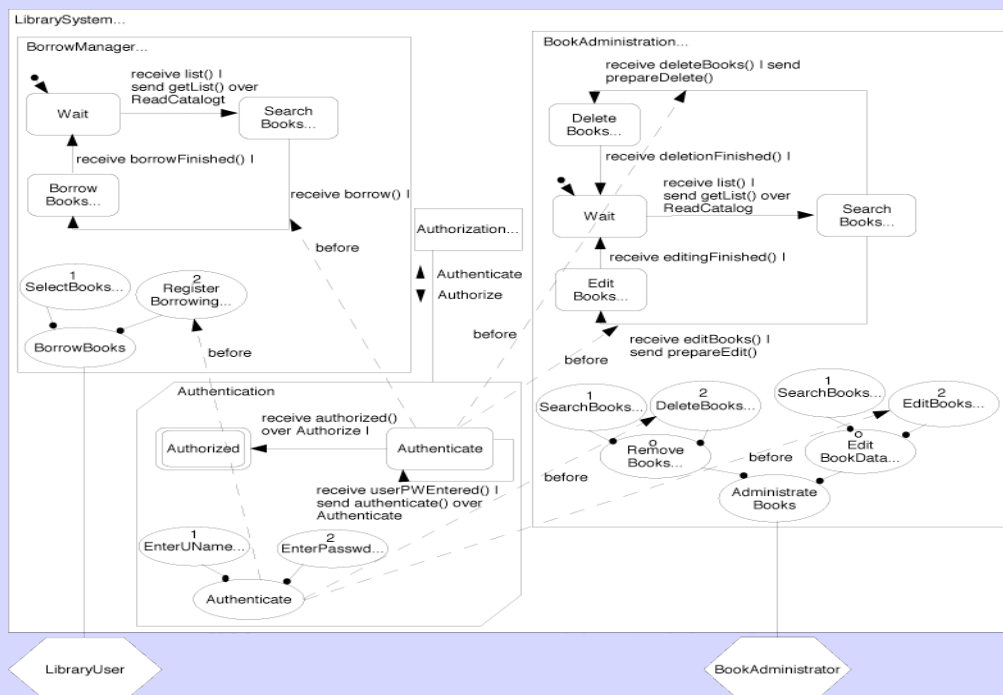


17. Dez. 2007

Validation of Aspect-Oriented ADORA Language Constructs / (c) Silvio Meier

11

Modeling of Crosscutting Concerns in ADORA



17. Dez. 2007

Validation of Aspect-Oriented ADORA Language Constructs / (c) Silvio Meier

12

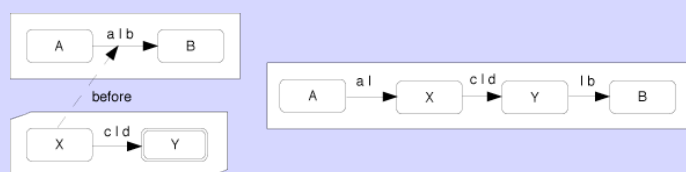
Weaving Semantics

- Hypothesis: *An aspect-oriented view is only useful in some situations, in other situations, the conventional view is better.*
 - It must be possible to switch between the aspect-oriented view and the conventional view
 - => Weaving semantics
- Weaving can be done automatically, according to given rules.
- However, the weaving is in the following experiment simulated.

Statechart Weaving Semantics

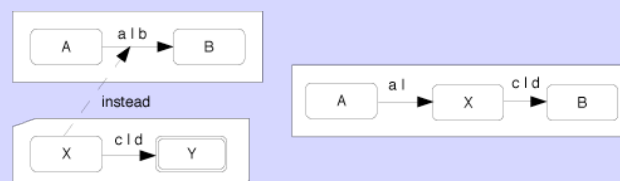
- **Before** Semantics:

- Insert State X
- New artificial State Y
- Split transition $a \mid b$



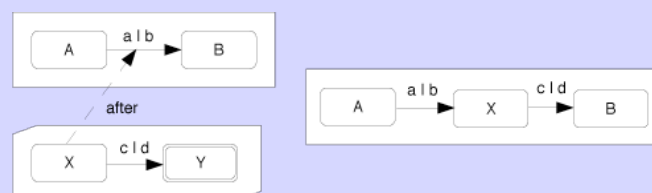
- **Instead** Semantics:

- Discard Action b
- Insert State X



- **After** Semantics:

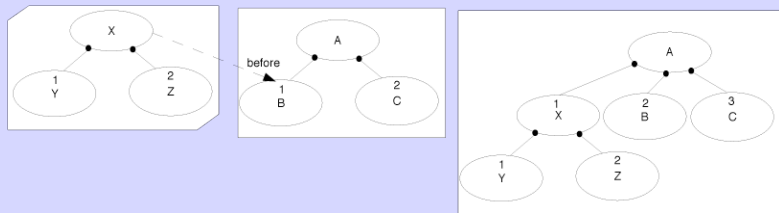
- Insert State X
- Insert transition $c \mid d$ after the transition $a \mid b$



Scenario Weaving (1)

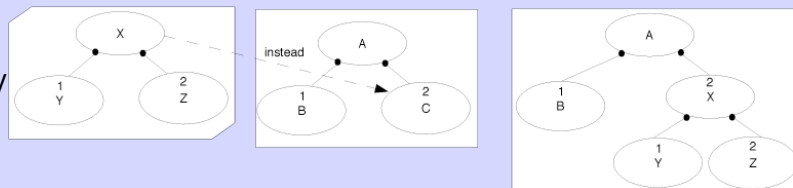
- **Before Semantics:**

- Insert scenario X and its sub nodes before B
- Change sequence



- **Instead Semantics:**

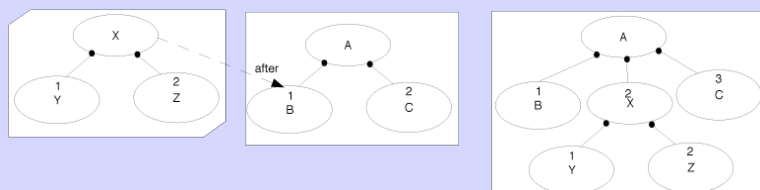
- Remove scenario C by X and its sub nodes.
- X gets the sequence number of C.



Scenario Weaving (2)

- **After Semantics:**

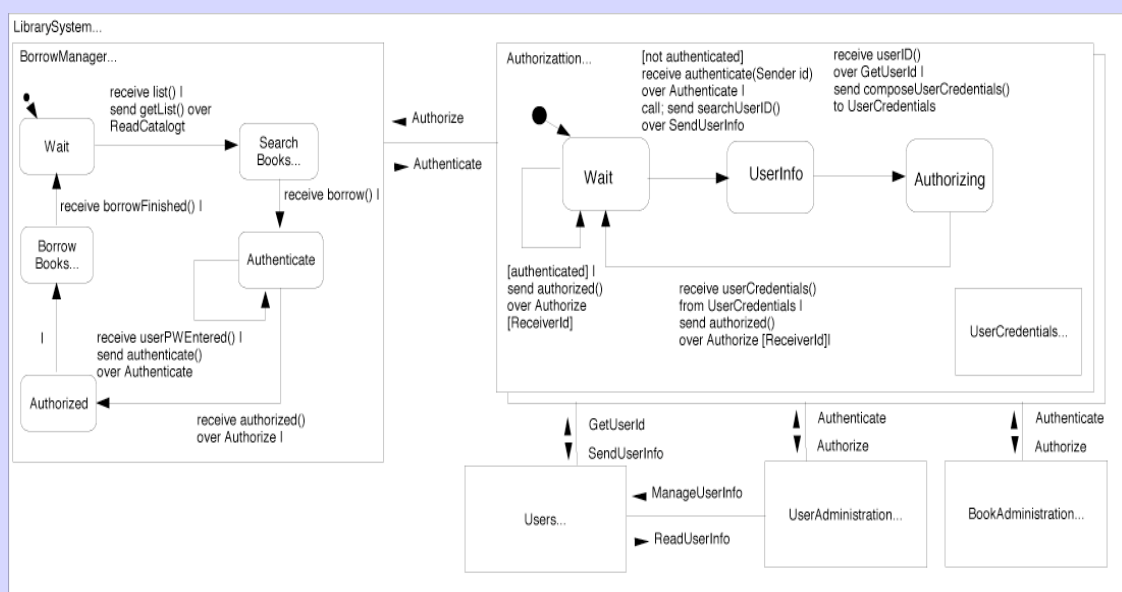
- Remove scenario C by X and its sub nodes.
- X gets the sequence number of C.



Other Elements / Shown in the Hands-on Session

- Decomposition of aspect
 - Server components
 - Embedded components
- Context-maps
 - Decoupling between target context and aspect
 - Definition of proxy variables in the aspect
 - Define a mapping for each proxy variable to the context of the target

The Woven Model



Validation (1)

- First: practical part
 - Tool Introduction, playing around.
 - Language examples and brief hands-on.
 - Short introduction into the test environment.
- Experiment
 - **Two groups**
 - **Two case studies** A and B with aspect-oriented and conventional models
 - Each group will answer questions for the case studies, where
 - Group 1 answers questions for the aspect-oriented model of A and questions for the conventional model of B
 - Group 2 answers questions for the conventional model of A and questions for the aspect-oriented model of B
 - Models are shown in the editor
 - You have to answer questions (Multiple Choice) about the shown models

Validation (2)

- Anonymously gathering of the following data:
 - ... your answers and
 - ... the time you had to answer a question
- Rules:
 - Try to give the right answer
 - Answer as fast as possible
 - Don't change the mode how you are looking up information in the model
 - Don't use the zoom tool. However you may use the scaling slider

Questions

