Hasso Plattner Institut

IT Systems Engineering | Universität Potsdam

Model Driven Design of Web Service Operations using Web Engineering Practices

07.01.2010 | EWST Seminar

Model Driven Design of Web Service Operations using Web Engineering Practices

Vicente Pelechano

Marta Ruiz

Presented at Workshop on Emerging Web Services Technology 2006

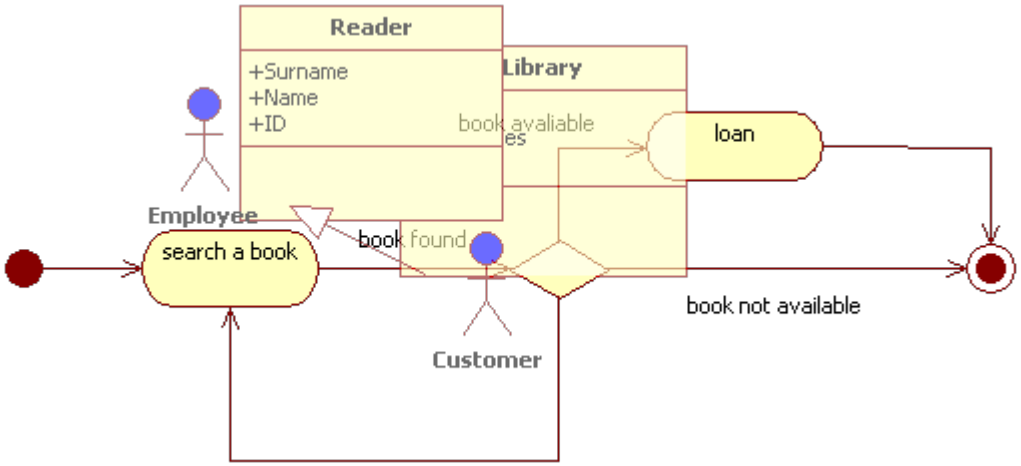
Overview

1. Model Driven Development
   - OO – Method
   - OOWS – Method

2. Web Service Design Process
   - Process Steps
   - Identifying Operations from
     - Requirements / OO Method Models
     - OOWS Method Models

Model Driven Development

- Programming on a higher abstraction level
- Often DSLs are created and used
- Model transformations
- Clear distinction between implementation and functions

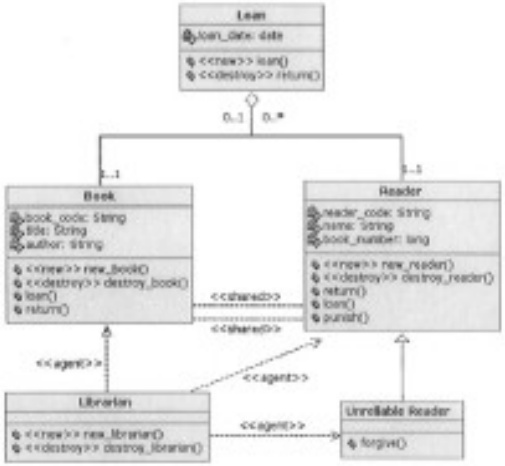


Automatic way

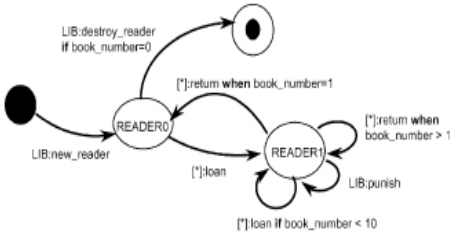
OO - Method

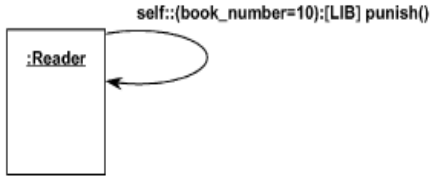
Conceptual Models

Object Model



Dynamic Model





Functional Model

CLASS: Reader ATTRIBUTE: book_number CATEGORY: push-pop

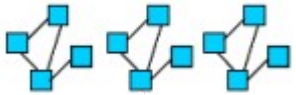
Action	Effect	Action Type
increase	reader:loan()	+1
decrease	reader:return()	-1

OO - Method

High Abstraction

Low Abstraction

Conceptual Models



conceptual construct → formel concept

Formal Specification OASIS

```
      ....
valuation
    [loan()] book_number= book_number + 1;
    [return()] book_number= book_number - 1;
preconditions
    librarian:destroy_reader () if
    book_number = 0 ;
     .....
```

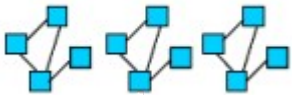
Execution Model

Implementation (n-tier architectur)

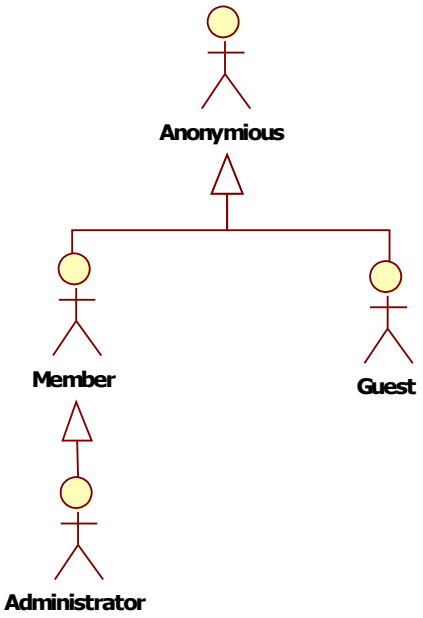
```
public interface ISystem {
  public abstract String[] getList_of_Classes(String agent_Class);
  public abstract String[] getList_of_Services(String server_Class,
  String client_Class);
}

public class System_View implements ISystem {...}
```

OOWS - Method

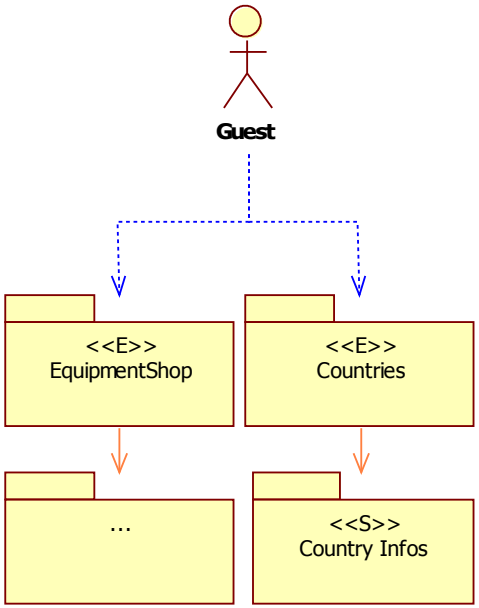
Web Conceptual Models



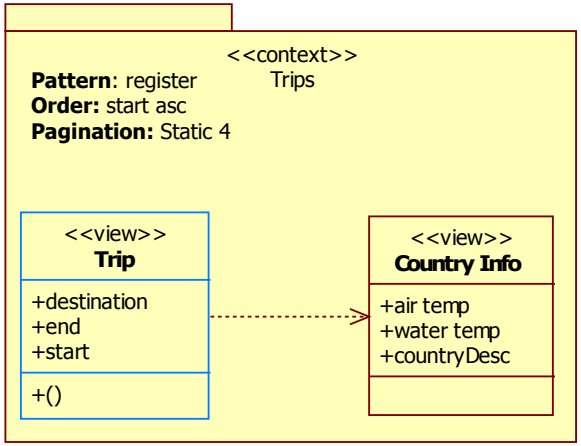
User Model



Navigational Model



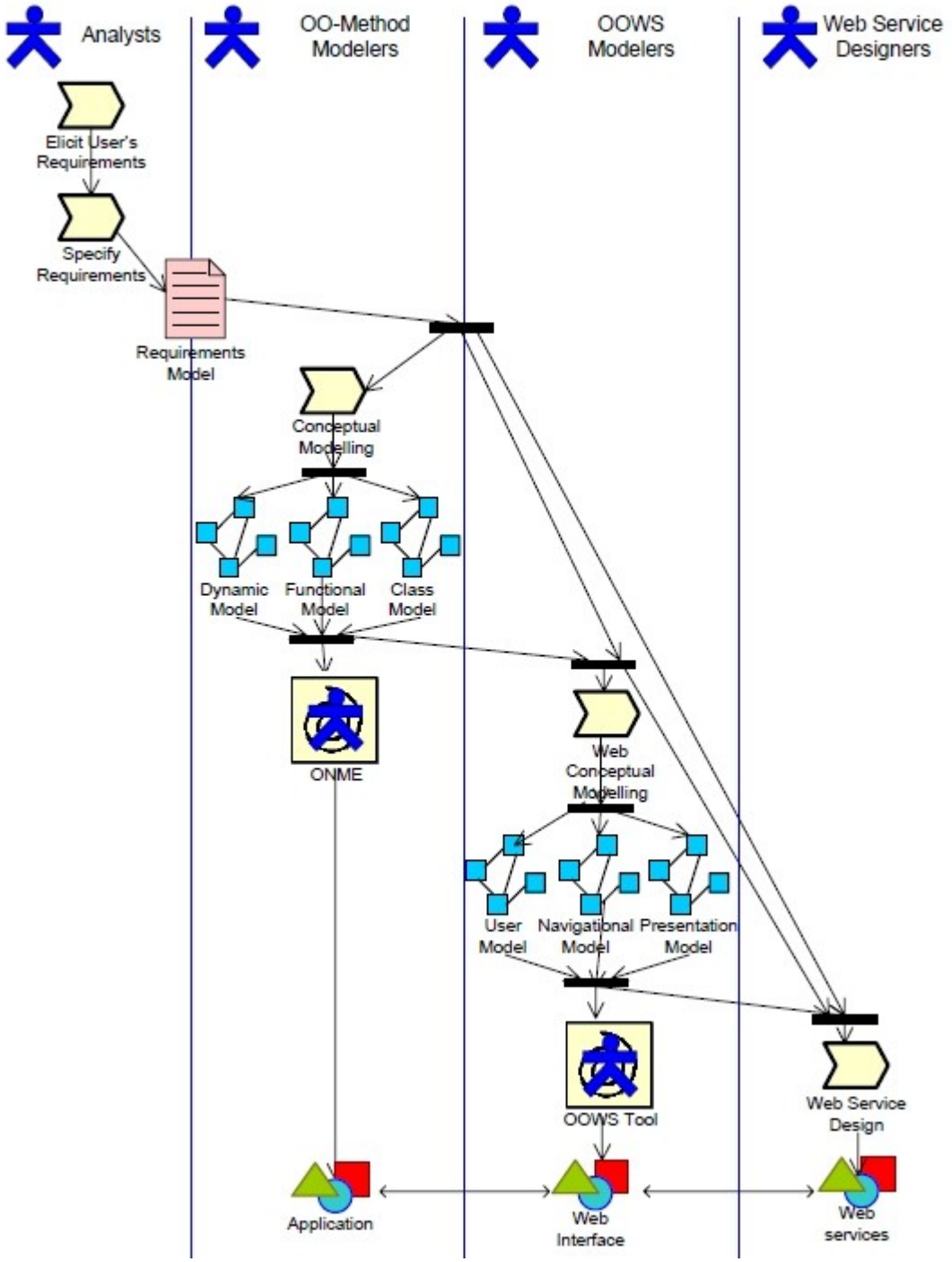
Presentational Model



Overview

1. Model Driven Development
   - OO – Method
   - OOWS – Method

2. Web Service Design Process
   - Process Steps
   - Identifying Operations from
     - Requirements / OO Method Models
     - OOWS Method Models

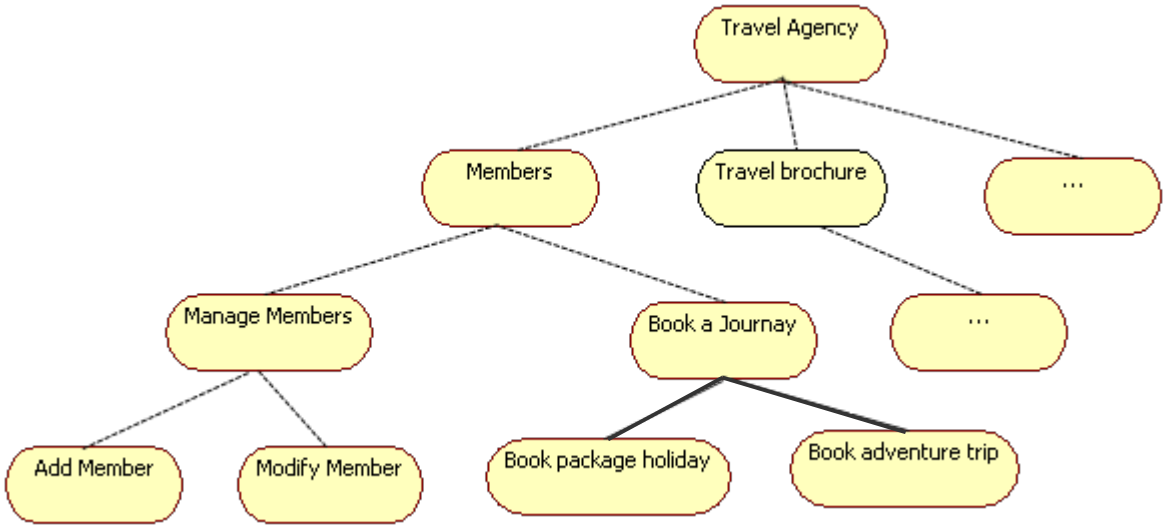

Analysts
OO-Method Modelers
OOWS Modelers
Web Service Designers
Elicit User's Requirements
Specify Requirements
Requirements Model
Conceptual Modelling
Dynamic Model
Functional Model
Class Model
ONME
Web Conceptual Modelling
User Model
Navigational Model
Presentation Model
OOWS Tool
Web Service Design
Application
Web Interface
Web services

Identifying Operations from Requirements / OO Models

- Result of requirements elicitation is a requirements model
- Based on concept of tasks
  - Define task taxonomy for each kind of user (Concur Task Tree)
  - For each task create description of the interaction users require

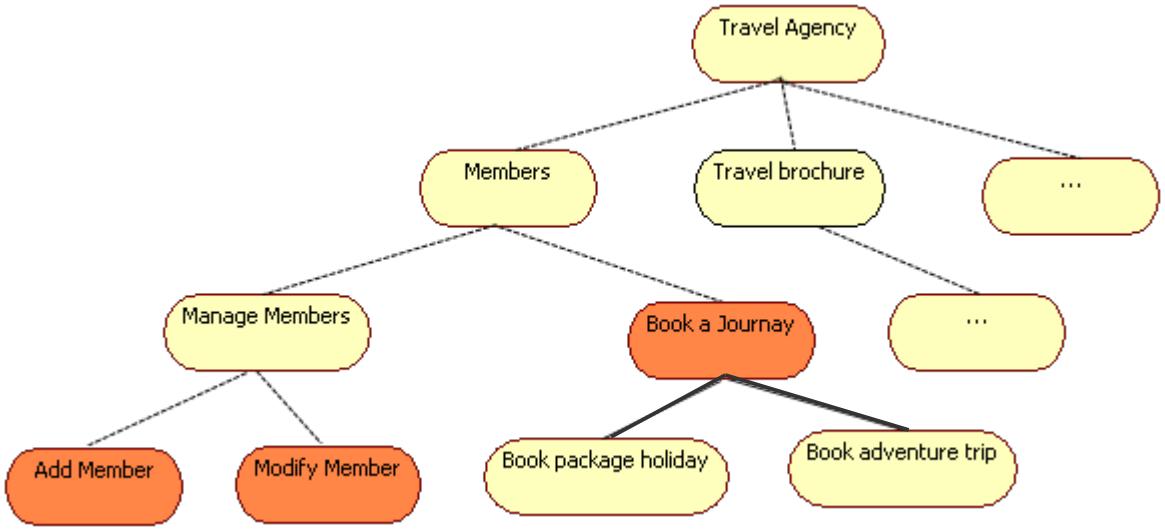
Identifying Operations from Requirements / OO Models

- Result of requirements elicitation is a requirements model
- Based on concept of tasks
  - Define task taxonomy for each kind of user (Concur Task Tree)
  - For each task create description of the interaction users require



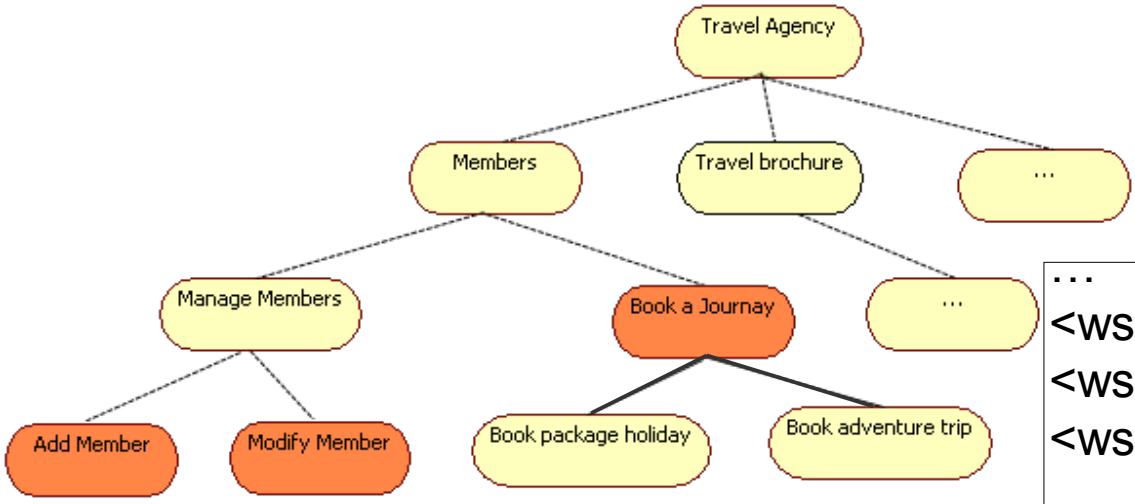
Identifying Operations from Requirements / OO Models

- Result of requirements elicitation is a requirements model
- Based on concept of tasks
  - Define task taxonomy for each kind of user (Concur Task Tree)
  - For each task create description of the interaction users require



Identifying Operations from Requirements / OO Models

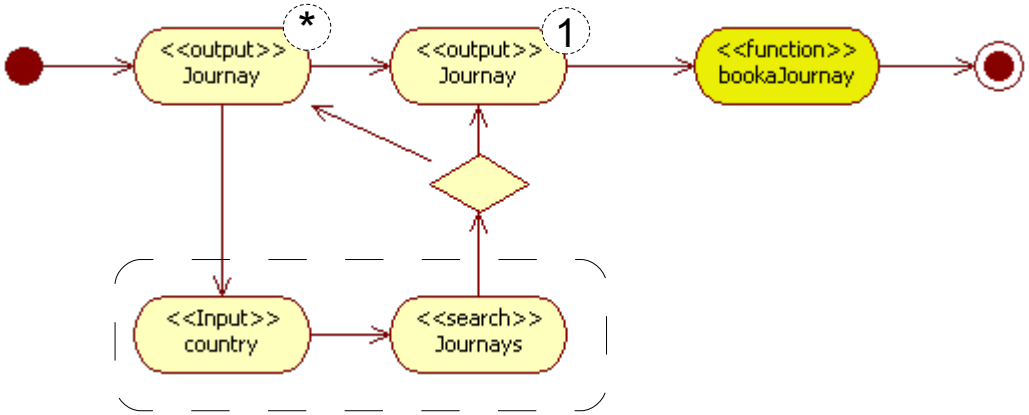
- Result of requirements elicitation is a requirements model
- Based on concept of tasks
  - Define task taxonomy for each kind of user (Concur Task Tree)
  - For each task create description of the interaction users require




```
...
<wsdl:op name="addMember">
<wsdl:op name="modifyMember">
<wsdl:op name="bookAJournay">
...
```

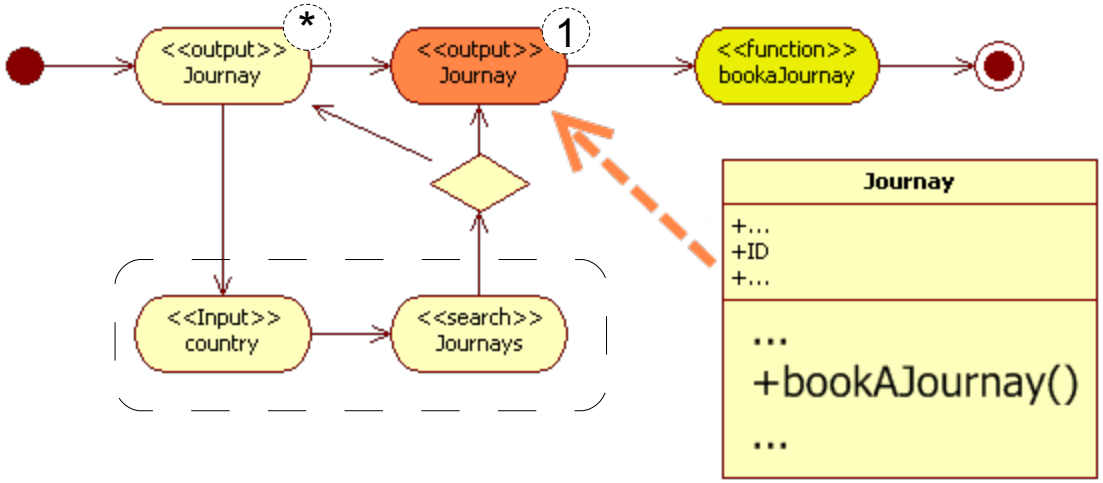
Identifying Operations from Requirements / OO Models

- Result of requirements elicitation is a requirements model
- Based on concept of tasks
  - Define task taxonomy for each kind of user (Concur Task Tree)
  - For each task create description of the interaction users require



Identifying Operations from Requirements / OO Models

- Result of requirements elicitation is a requirements model
- Based on concept of tasks
  - Define task taxonomy for each kind of user (Concur Task Tree)
  - For each task create description of the interaction users require



Overview

- Model Driven Development
  - OO – Method
  - OOWS – Method

2. Web Service Design Process
   - Process Steps
   - Identifying Operations from
     - Requirements / OO Method Models
     - OOWS Method Models

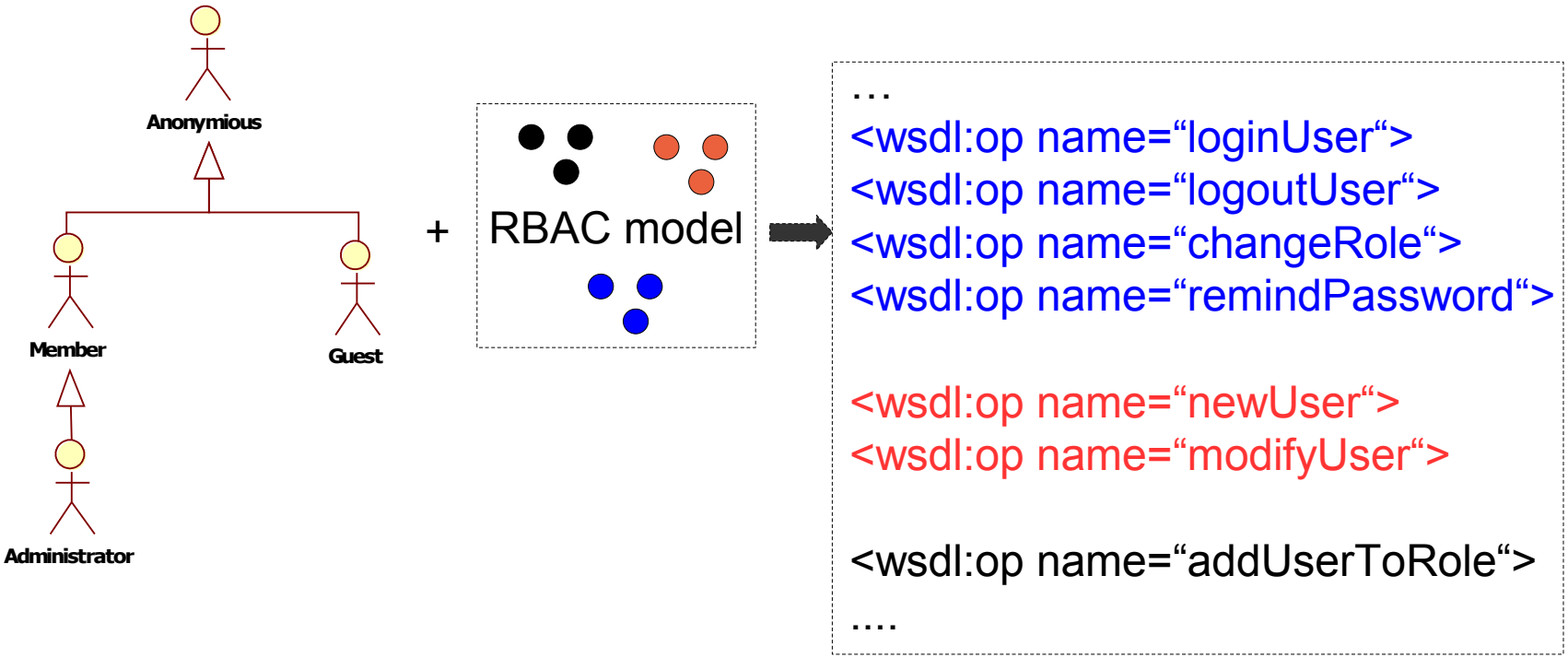
Identifying Operations from OOWS - User Model

Classification of operations into three types

- User identification
  - *LoginUser, logoutUser, obtainRole,changeRole, remindPassword*
- User administration
  - *newUser,modifyUser,deleteUser*
- Management of user's permission and roles
  - *newRole,deleteRole,addUserToRole,removeUserToRol, addPermission,removePermission* (RBAC model)

Identifying Operations from OOWS - User Model

Operations are detected from user diagram and RBAC model



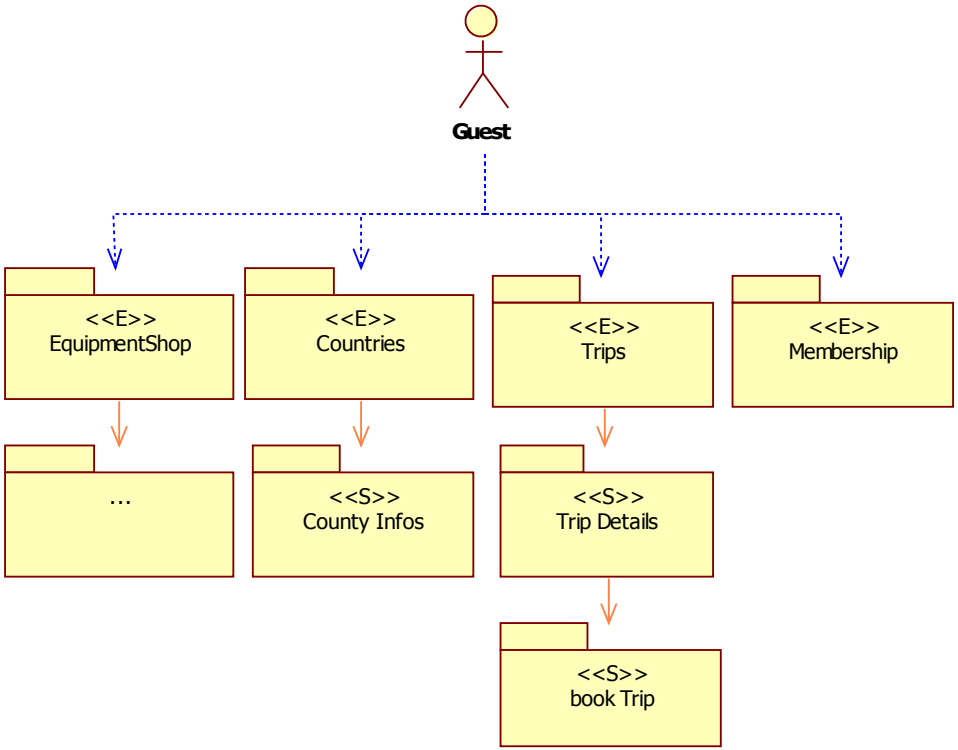
Identifying Operations from OOWS - Navigational Model

- For each kind of user specify system view

- Built in to steps
  1. Define global view ofer navigation – Navigational Map
  2. Create description of each element defined in first step – Navigational Context

Identifying Operations from OOWS - Navigational Model

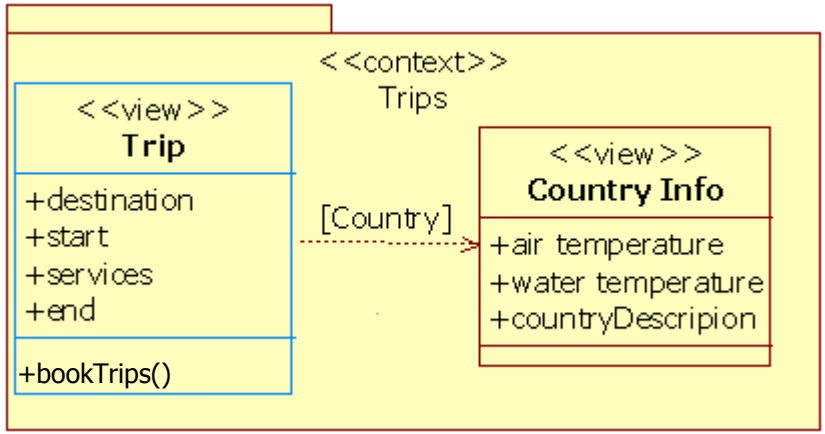
1. Define global view ofer navigation – Navigational Map




```
…
<wsdl:op name="explorationLink">

<wsdl:op name="sequenceLink">
…
```

Identifying Operations from OOWS - Navigational Model

2. Create description of each element – Navigational Context




```
...
<wsdl:op name="retrieveTrip">
<wsdl:op name="getIndexedTrip">
<wsdl:op name="searchTrip">
...
```

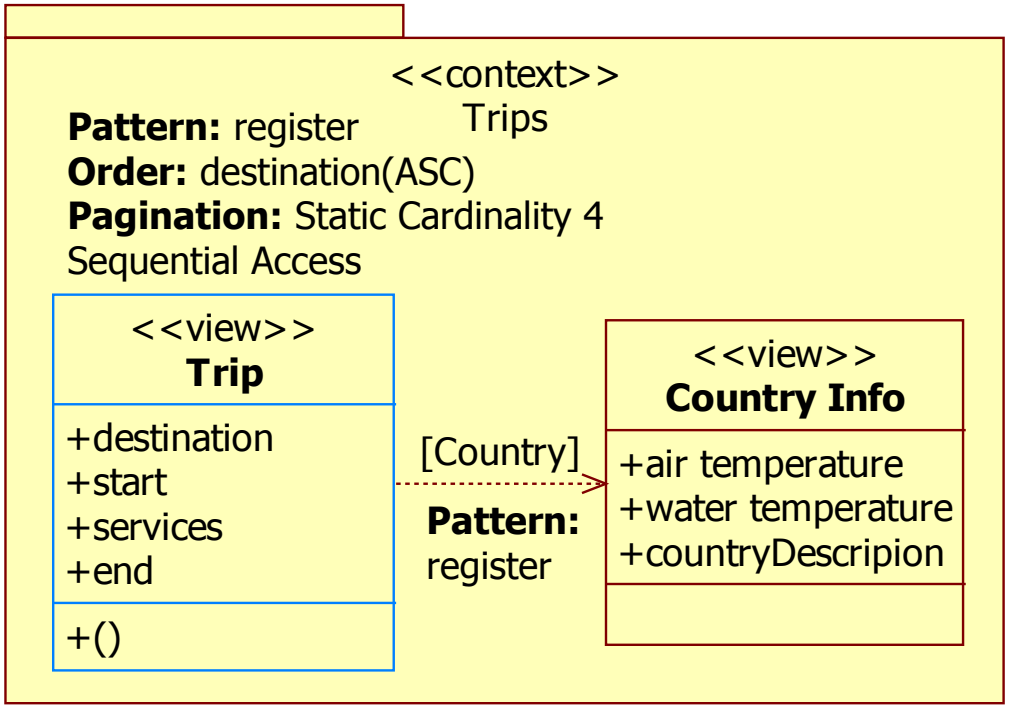
```
ATTRIBUTE INDEX destination_char
ATTRIBUTES destination,start,end
LINK ATTRIBUTE destination
```

```
FILTER TRIP
ATTRIBUTE start
TYPE APPROXIMATE
```

Identifying Operations from OOWS - Presentational Model

- Based On navigational contexts
- Basic patterns for presentation requirements
  - *Information Paging*
    - Scrolling through „logical blocks“ of information
  - *Ordering Criteria*
    - Class population ordering (asc,desc
  - *Information Layout*
    - Register, tabular, master-detail, tree

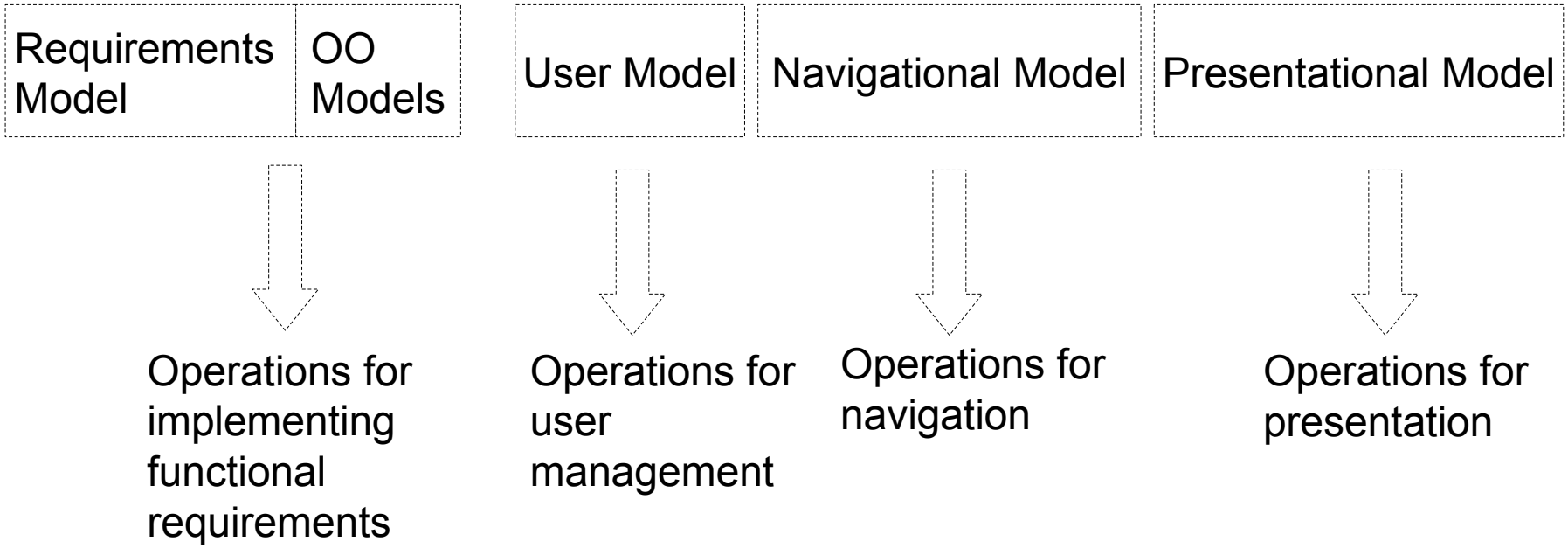
Identifying Operations from OOWS - Presentational Model




```
...
<wsdl:op name="presentationContext">
<wsdl:op name="presentationInfo">
...
```

Summarization

- Methodological guide to develop web service based web applications and identifying operations needed by service to fit in requirements
- Model driven approach based on OO / OOWS – Method



- Tool support to generate WSDL-file automatically

Discussion

- One Web service with a lot of operations for whole web application?

- Idea of a web service and using it in context of supporting navigational and presentational features?