Fabio Paternò

CNR-ISTI

HIIS Laboratory

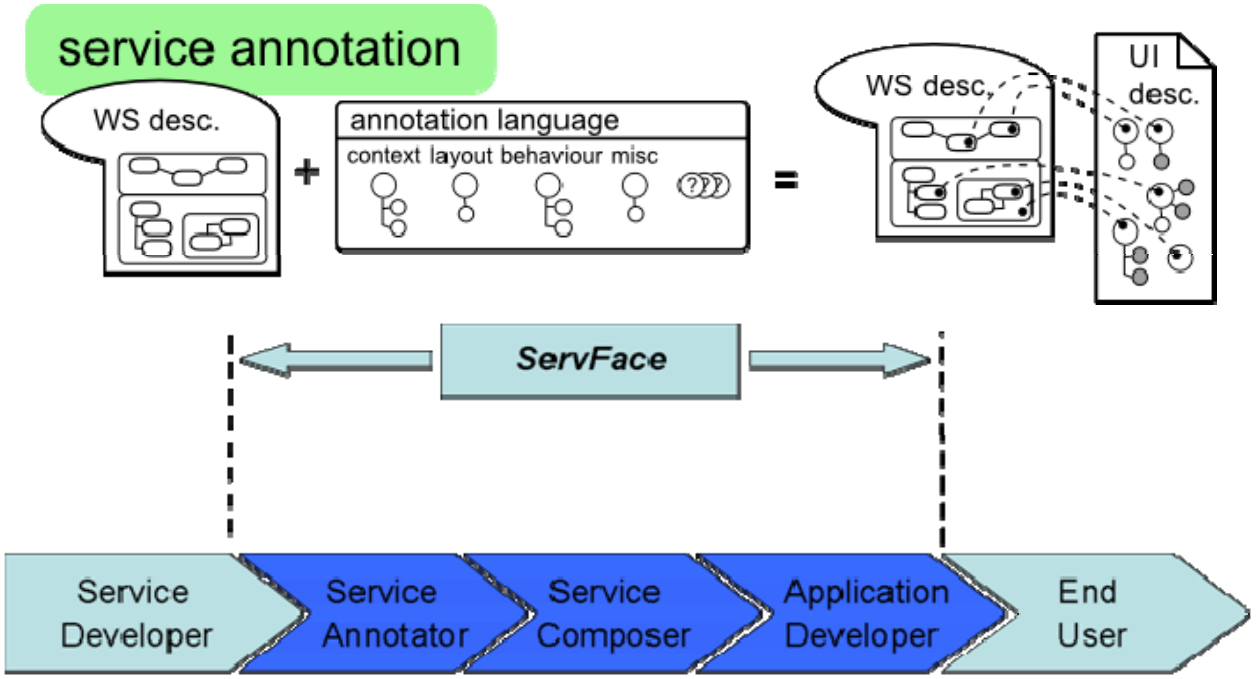
http://giove.isti.cnr.it

Report on Model-based Work in the ServFace and OPEN Project

Active EU Projects

- **ServFace** (http://www.servface.eu/) aims to create a **model-driven service engineering methodology** for
  - the design of user interfaces for applications based on web services (primary goal); and
  - the composition and integration of user interfaces for applications based on web-services (secondary goal)
  - SAP (coordinator), CNR-ISTI, Lyria, Univ. Dresden, Univ. Manchester
- **OPEN** (http://www.ict-open.eu/) aims to deliver seamless and transparent support to users in carrying out their tasks when changing services and/or devices, even in multi-user applications
  - **Migration = Device Change + Adaptation + Continuity**
  - CNR-ISTI (coordinator), SAP, NEC, Vodafone Italy, Univ. Aalborg, Univ. Clausthal, Arcadia

UI Annotations for Web Services

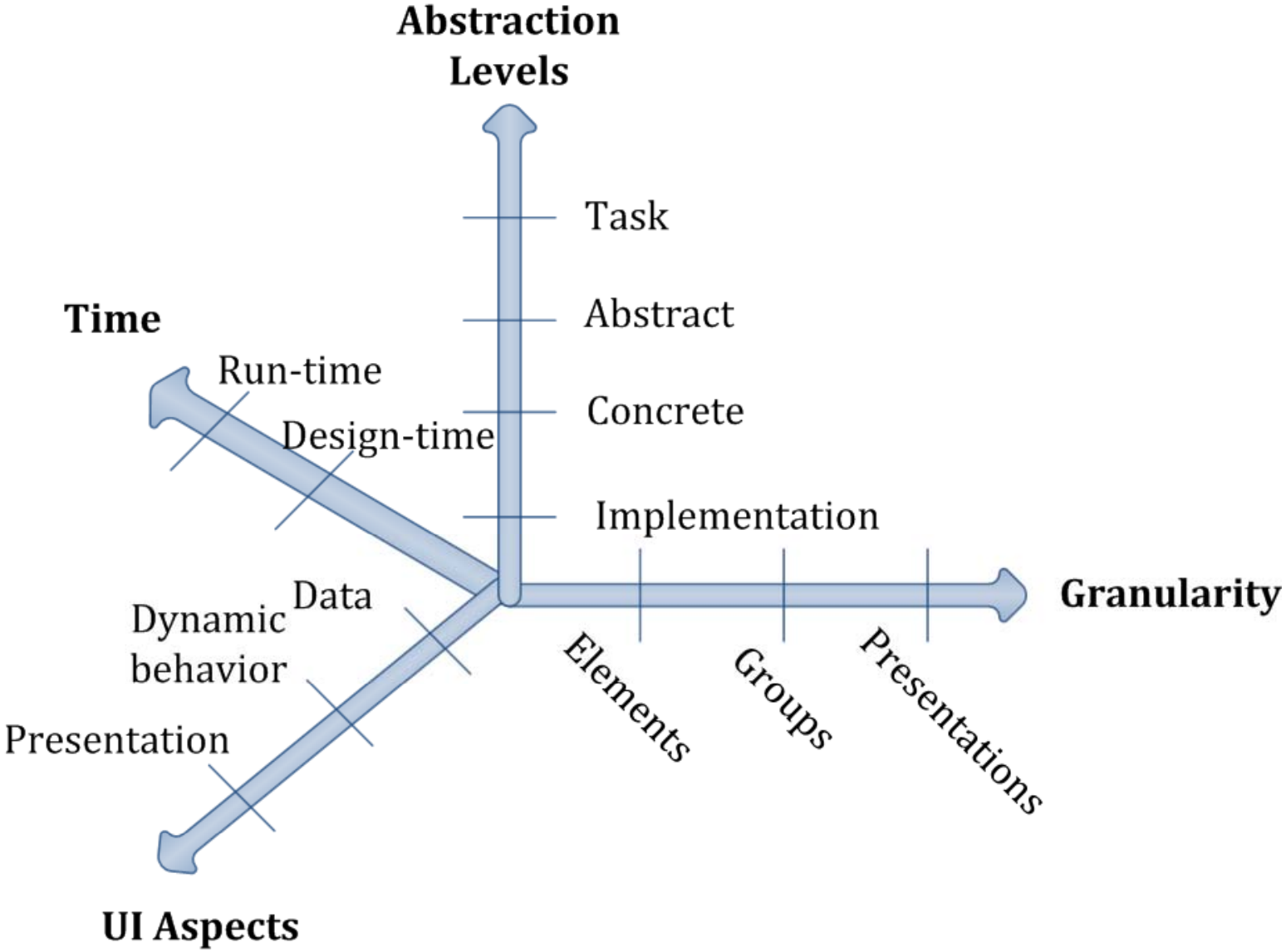


UI Annotations are hints for obtaining better user interfaces

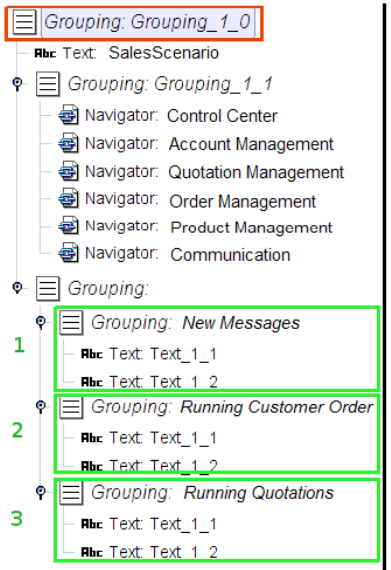
Composing What?

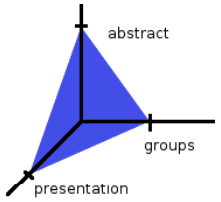
- Business Processes
  - BPMN/ WS-BPEL [Oasis, 2007]
  - BPEL4people [Agrawal et al., 2007 A],
  - WS-Human Task [Agrawal et al., 2007 B]
- User Interfaces
  - the dynamic behaviour, which means the possible sequencing of user actions and system feedback,
  - the presentation aspects (e.g.in graphical user interface we have to indicate the spatial relationships)
  - the content manipulated by the user interface.
- Annotations
  - Partial user interface specifications

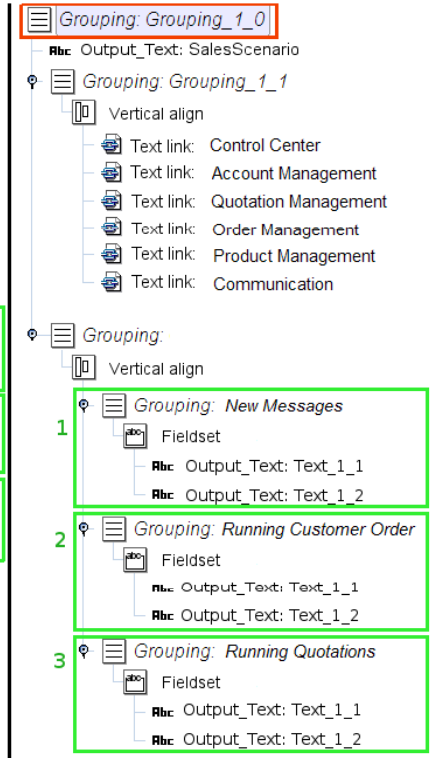
Composing User Interfaces

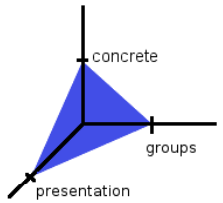


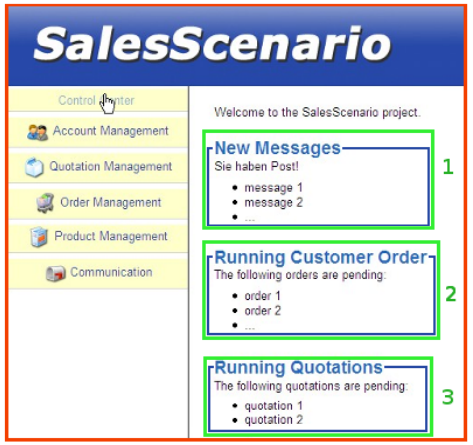
Example

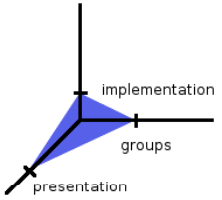

Grouping: Grouping_1_0
Text: SalesScenario
Grouping: Grouping_1_1
Navigator: Control Center
Navigator: Account Management
Navigator: Quotation Management
Navigator: Order Management
Navigator: Product Management
Navigator: Communication
Grouping:
1
Grouping: New Messages
Text: Text_1_1
Text: Text_1_2
2
Grouping: Running Customer Order
Text: Text_1_1
Text: Text_1_2
3
Grouping: Running Quotations
Text: Text_1_1
Text: Text_1_2


abstract
groups
presentation

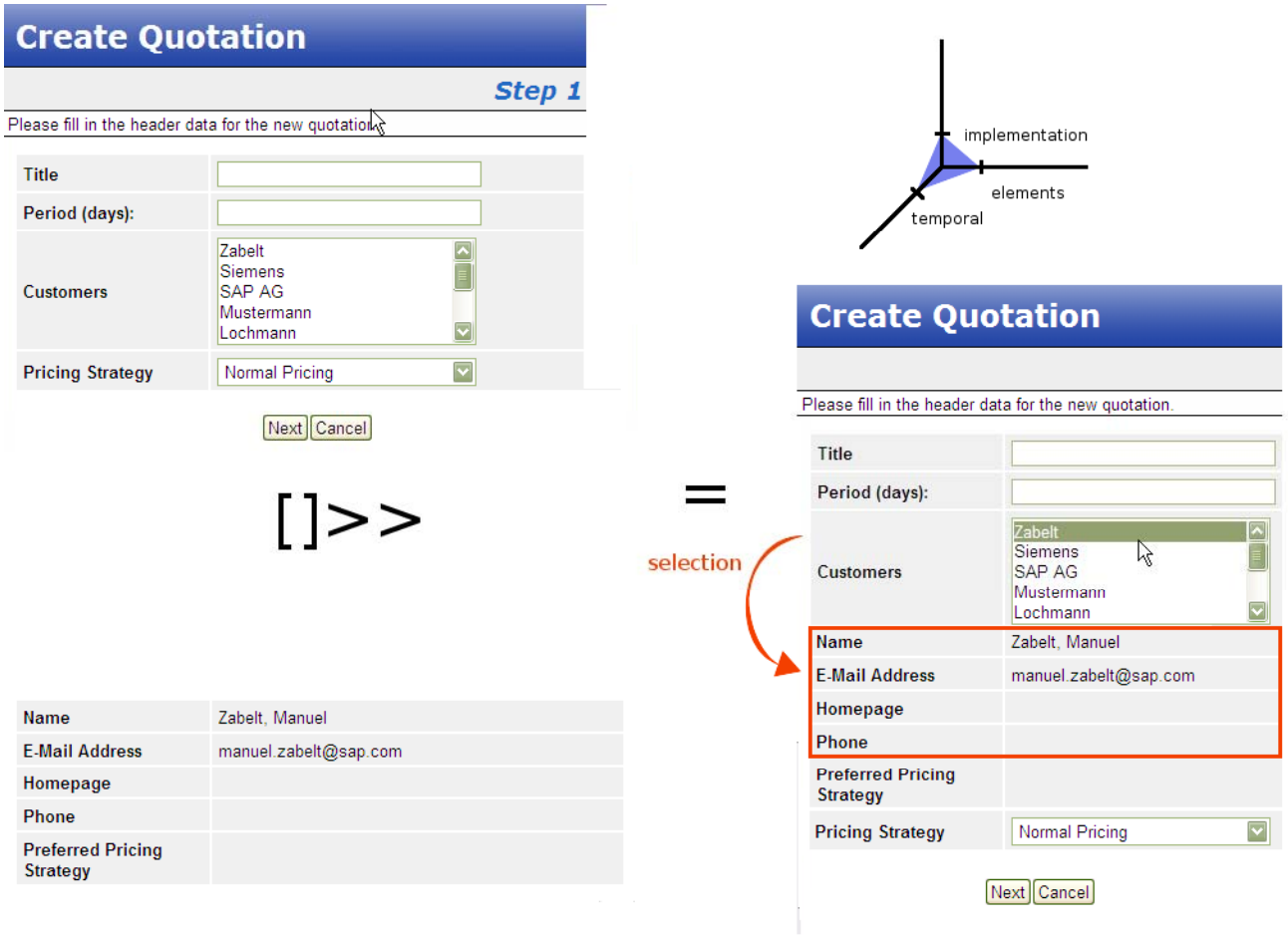

Grouping: Grouping_1_0
Output_Text: SalesScenario
Grouping: Grouping_1_1
Vertical align
Text link: Control Center
Text link: Account Management
Text link: Quotation Management
Text link: Order Management
Text link: Product Management
Text link: Communication
Grouping:
Vertical align
1
Grouping: New Messages
Fieldset
Output_Text: Text_1_1
Output_Text: Text_1_2
2
Grouping: Running Customer Order
Fieldset
Output_Text: Text_1_1
Output_Text: Text_1_2
3
Grouping: Running Quotations
Fieldset
Output_Text: Text_1_1
Output_Text: Text_1_2


concrete
groups
presentation


SalesScenario
Control Center
Account Management
Quotation Management
Order Management
Product Management
Communication
Welcome to the SalesScenario project.
New Messages
Sie haben Post!
message 1
message 2
1
Running Customer Order
The following orders are pending:
order 1
order 2
2
Running Quotations
The following quotations are pending:
quotation 1
quotation 2
3


implementation
groups
presentation

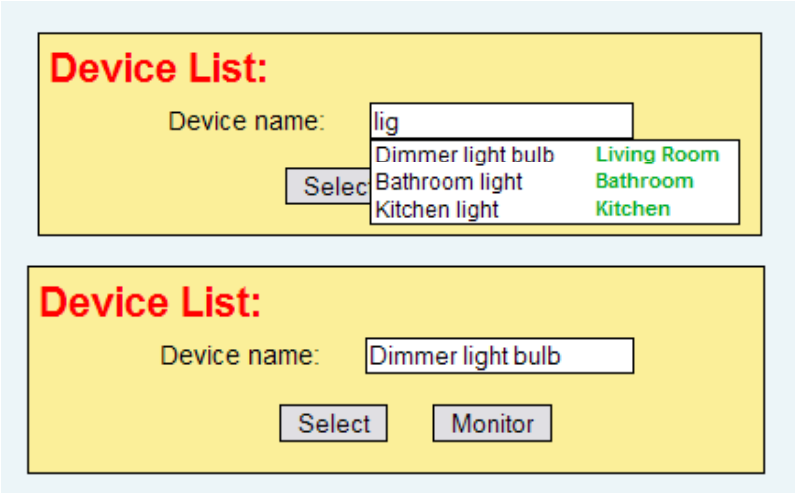
Example


Create Quotation
Step 1
Please fill in the header data for the new quotation.
Title
Period (days):
Customers
Zabelt
Siemens
SAP AG
Mustermann
Lochmann
Pricing Strategy
Normal Pricing
Next
Cancel
[]>>
Name
Zabelt, Manuel
E-Mail Address
manuel.zabelt@sap.com
Homepage
Phone
Preferred Pricing Strategy
=
selection
implementation
elements
temporal
Create Quotation
Please fill in the header data for the new quotation.
Title
Period (days):
Customers
Zabelt
Siemens
SAP AG
Mustermann
Lochmann
Name
Zabelt, Manuel
E-Mail Address
manuel.zabelt@sap.com
Homepage
Phone
Preferred Pricing Strategy
Pricing Strategy
Normal Pricing
Next
Cancel

MARIA XML Requirements

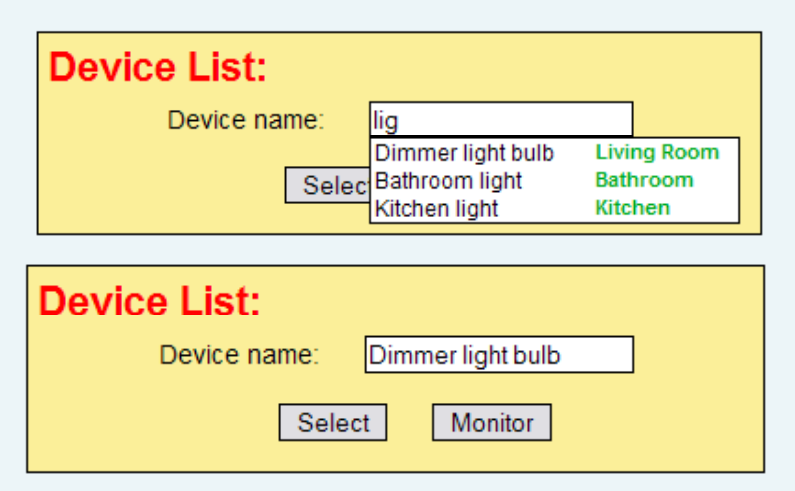
- Support for Abstract Data Types
  - Also useful for specifying the format of input values
- Able to support user interfaces including complex Javascripts and Ajax scripts
  - Continuously updating of fields
- Event model at abstract/concrete levels
  - Property change events / Activation events
- Extended Dialogue Model
- Dynamic set of user interface elements
  - Conditional connections between presentations
  - Possibility to change only a part of a UI

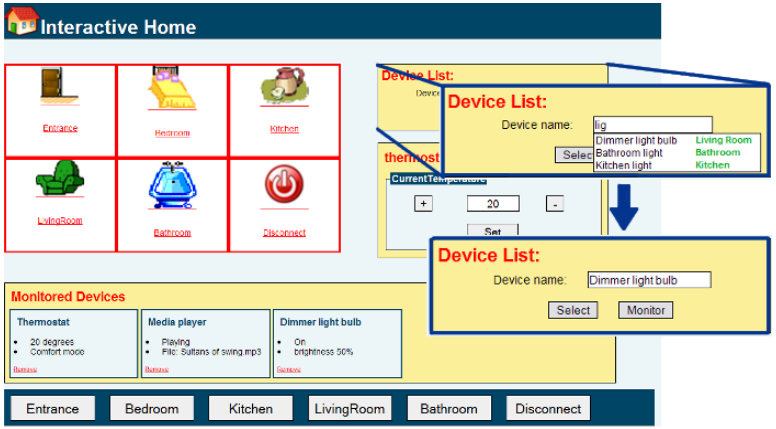
MARIA Example



- When value changed event of the text_edit interactor
- A function is called using an abstract script, which retrives relevant values
- The *hidden* property of the single choice interactor is changed to false
- When the selected element changes, it enables two activators
- The input text of the text_edit is completed with the selected value and the choice hides itself.

MARIA Example





The Proposed Design Approach

- A traditional top-down approach going through the various abstraction layers does not seem particularly effective.
  - Create interactive applications accessing application functionalities developed by others.
- First a bottom-up step in order to analyse the Web services providing functionalities useful for the new application to develop.
  - Analysis of the operations and the data types associated with input and output parameters is carried out in order to associate them with suitable abstract interaction objects
- Task model expressed in ConcurTaskTrees (CTT) for describing the interactive application and how it assumes that tasks are performed.
  - Design based on user requirements
  - Indicate how to compose functionalities implemented in different Web services.
  - Web services are application functionalities, thus they are associated with system tasks.
- Level of granularity to reach in the task decomposition.
  - Associating the system basic tasks to the web services
  - Associate each system basic task with the operations of the web services. Thus, if a Web Service supports three operations, then there would be three basic system tasks.

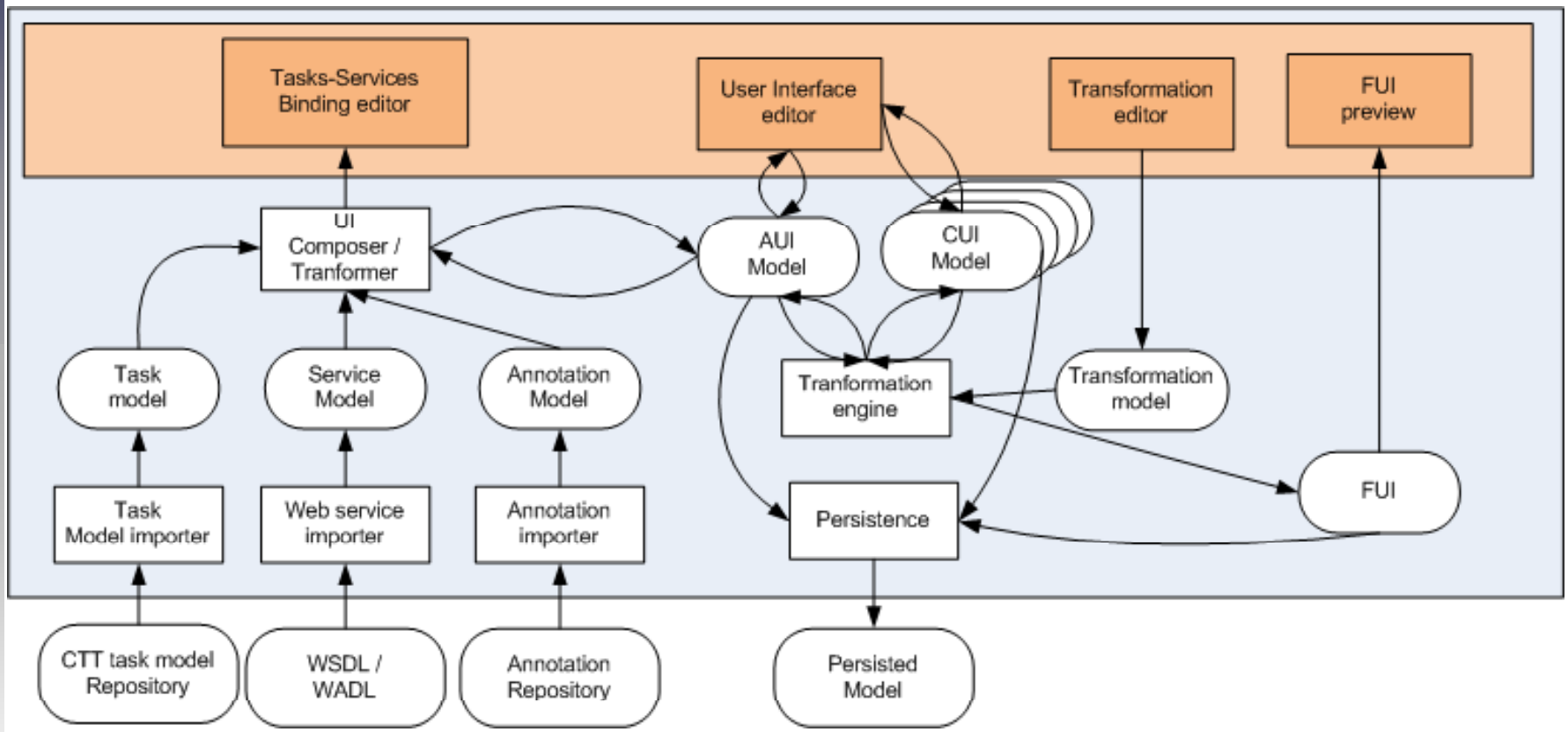
The Proposed Design Approach

- Transformation of Task Model into Abstract and Concrete Descriptions
  - Temporal operators indicate requirements for UI dialogue model
  - Task hierarchy provides information regarding grouping of tasks
  - Type of task indicates type of interactor
- Inclusion of annotations in resulting Abstract/Concrete Specification
- Editing Abstract/Concrete Specification
- User Interface Generation
- Inclusion of Usability Guidelines in Authoring and UI Generation

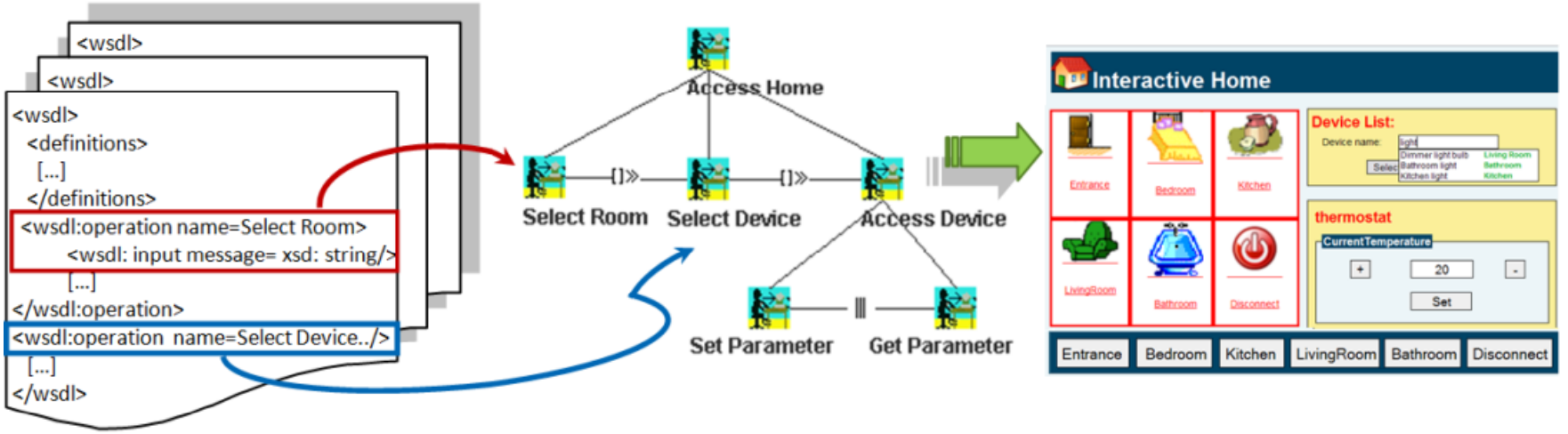
MARIA Tool Requirements

- New Authoring Environment
  - Integrated Support for Web Services
    - Mappings WSDL/LUI
    - Generation/Refinement
  - Not only traditional top-down approaches
  - Transformations not hard-coded but defined externally and performed with XSLT
  - Integration of BPMN/BPEL with Model-based UIs through task models.

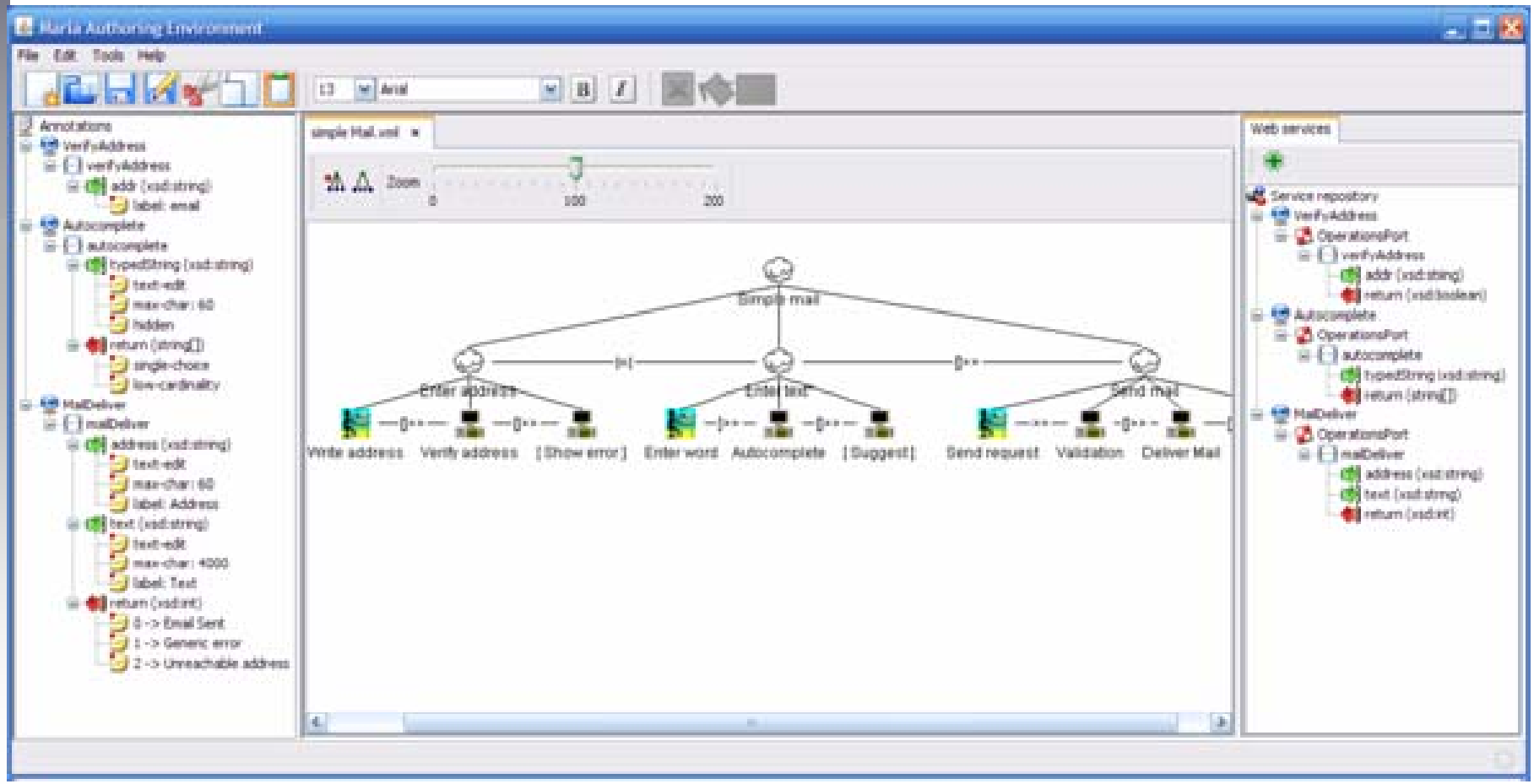
The Environment Architecture



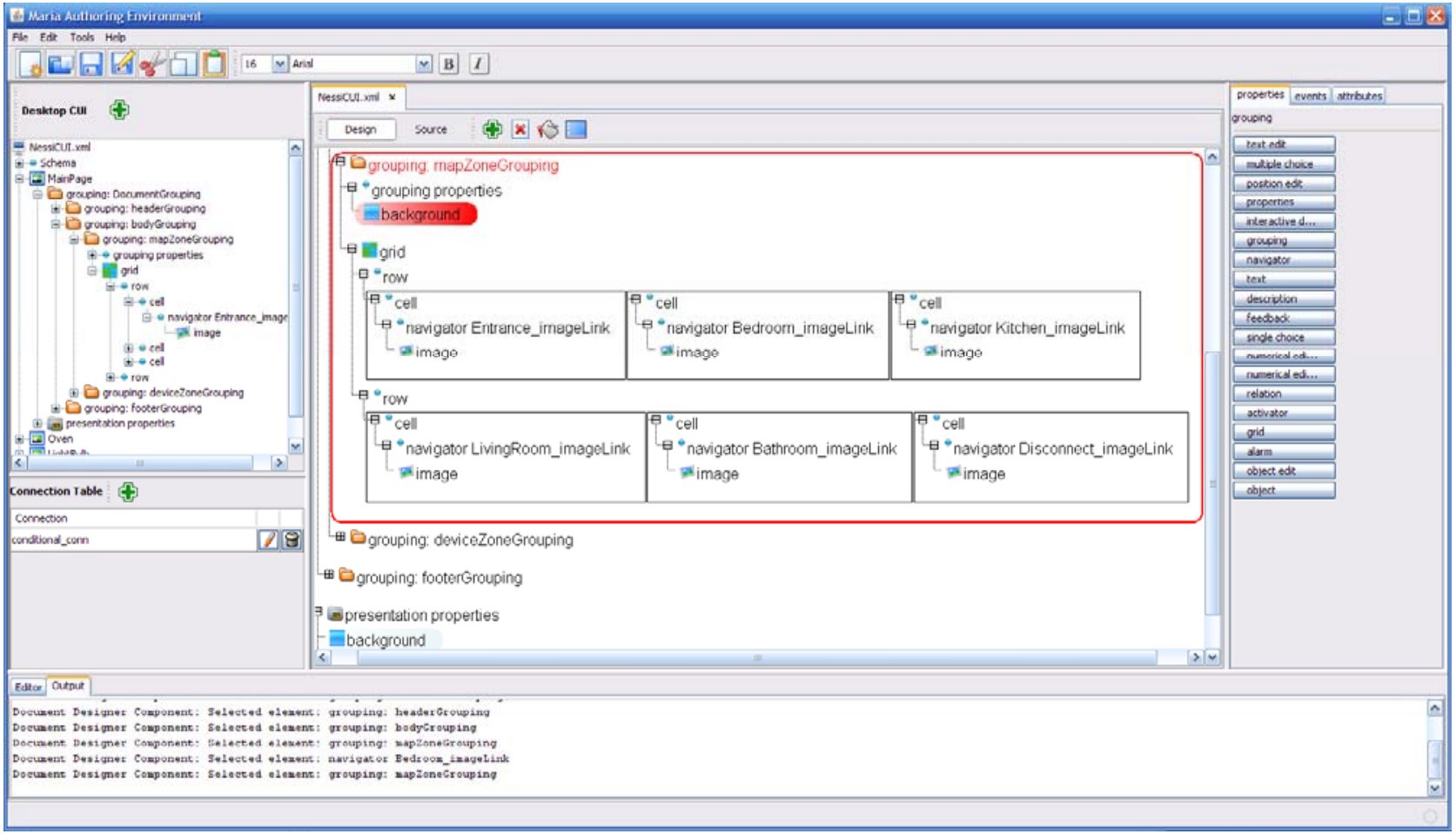
Two Main Phases



The Task-Service Binding Editor



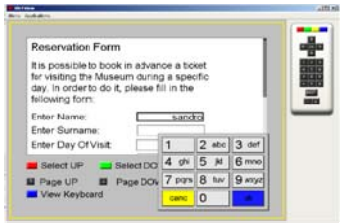
MARIA Editor



Why Migratory Interfaces

- Our life is becoming a multi-device experience
- One of the main source of frustration is that we need to restart for each device change
- Need for continuous access to interactive services across various devices
- Migratory user interfaces can transfer among different devices (from ‘source’ devices to ‘target’ devices), so as to allow the users to continue their tasks
- Application domains such as shopping, bids for auction on line, games, making reservations

Migration Architecture



User interface generation
& activation
(newUrl)

3

Server

1

Migration request
(url,runtime data)

2

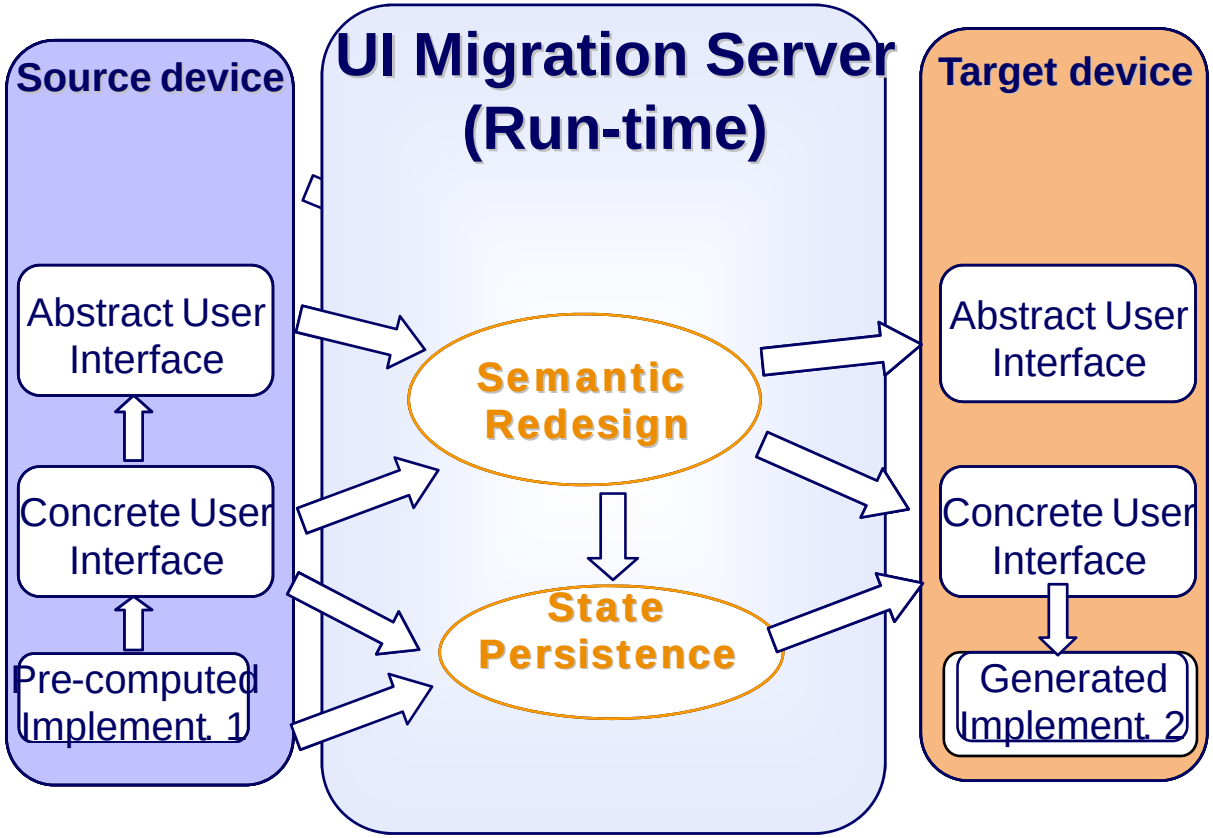
State Mapping and Adaptation

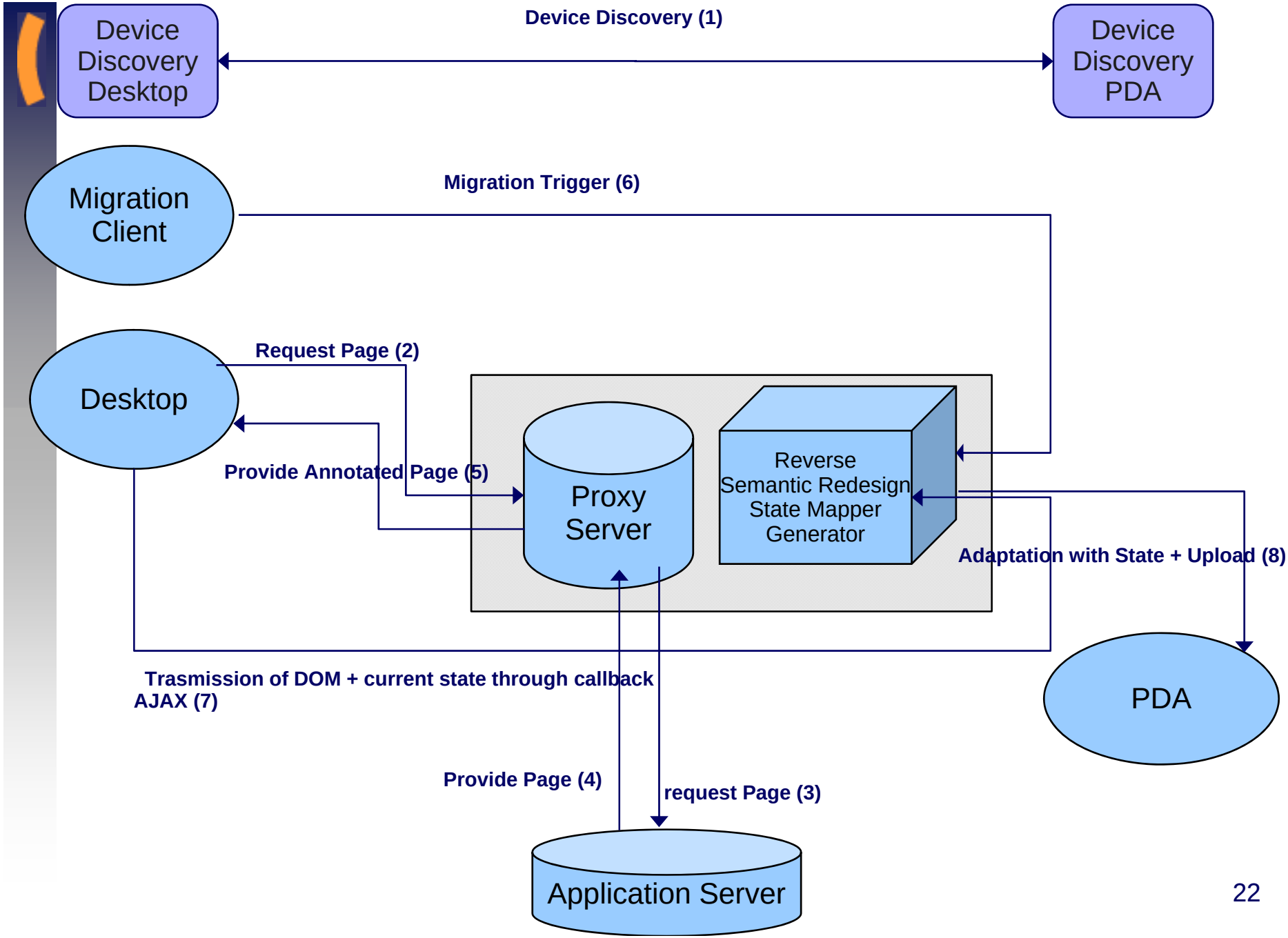
The Migration Process

- **Device Discovery:** Devices notify their presence
- **When to Migrate**: Migration Trigger (User or System or Mixed initiative)
- **Where to Migrate**: Identification of Target and its resources
- **What to Migrate**: User interface and/or Application logic
- **How to Migrate**: Adaptation (depending on the type of target)
- **State Persistence**: Source state extraction and association to target version
- **Activation in the target device:** at the point in which the user left off on the source device

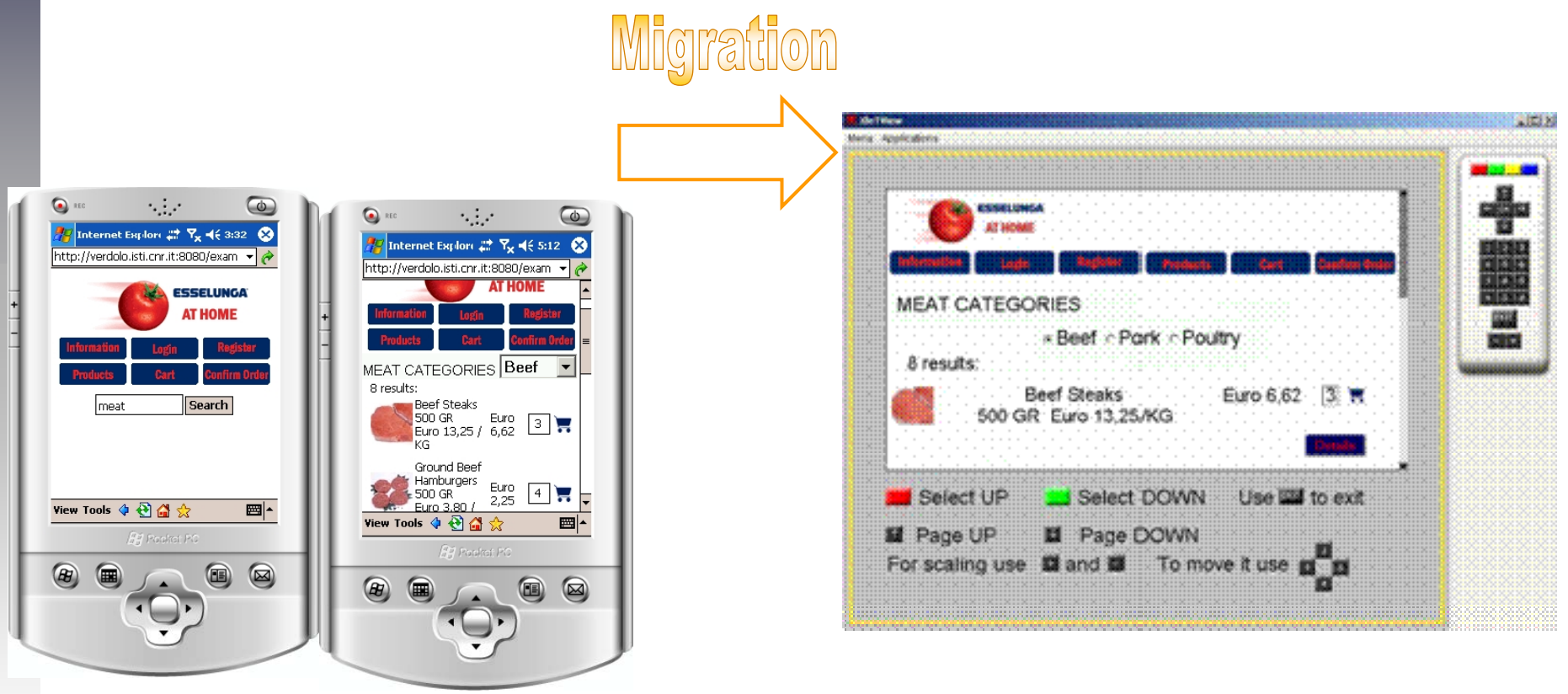
User Interface Migration

- Avoid ad-hoc solution
- Existing Web desktop Applications
- Possibility of targeting a variety of interaction platforms
- Use of Logical Description Languages




Device Discovery Desktop
Device Discovery (1)
Device Discovery PDA
Migration Client
Migration Trigger (6)
Request Page (2)
Desktop
Provide Annotated Page (5)
Proxy Server
Reverse Semantic Redesign State Mapper Generator
Adaptation with State + Upload (8)
Trasmission of DOM + current state through callback AJAX (7)
PDA
Provide Page (4)
request Page (3)
Application Server

Migrating between mobile and Digital Tv



Migrating between desktop and mobile device