Designing Usable Applications based on Web Services

Fabio Paternò, Carmen Santoro, Lucio Davide Spano
Human Interfaces in Information Systems Laboratory
Pisa, Italy

http://giove.isti.cnr.it
ISTI-C.N.R.
Goals and Motivations

- Model-based user interface generation for multi-device environments
  - exploiting XML logical descriptions and associated transformations for the target devices and implementation languages.
- Web services are increasingly used to support remote access to application functionalities
  - “a software system designed to support interoperable machine-to-machine interaction over a network”
  - Often described using WSDL (Web Services Description Language) files, which are XML-based descriptions as well.
- The home is becoming more and more populated by interactive devices
  - remote access to their state in order to query or modify it.
- Industrial Collaboration with Almaviva Company / ServFace EU Project
Problems with Mobile Access
Design of Multi-Device Interfaces: Approaches

- **Manual solutions,**
  - Expensive
- **Transcoders,**
  - Low cost/low usability
- **Style sheets,**
  - Partial solution
- **Information Visualization**
  - Require heavy computations
- **Model-based Approaches**
  - Need for a trade-off between abstraction and control
ServFace EU Project

ServFace 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)

Consortium: SAP Research, CNR-ISTI, Lyria, Univ. Dresden, Univ. Manchester
Abstraction Levels and Transformations

Reverse

Task and Object

Abstract Interface

Concrete Interface

Forward
Related Work

- Most model-based approaches have not addressed Web-service based applications.
- PUC supports only access to one appliance at any time.
- Work on user interfaces for web services but without using model-based approaches at Dresden and Yonsei Universities.
- Work by Vermeulen et al. (EIS07) on extending Web services with OWL-S combined with task and layout model.
The First Case Study

Universal Access to Interactive Home

Interoperability Layer

User Interfaces

Web Services

UPnP
Konnex
BTicino

Air conditioning
Lights
Rolling shutters & Windows
Domestic Appliances
Video recorder
The Home Control Environment Architecture

- Home Server Publishing
- Web Services methods
- Server Manager
- Abstract Interface Generator
- Concrete Interface Generator
- User Interface Generator
- User Interface Generation
- User Selection
- Web Interface for Platform Selection
- Desktop Web Interface
- PDA Application Interface
- Concrete Patterns
The Desktop Interface
The Mobile Interface
MARI A XML Requirements
(MARI A - Model-based Authoring envirOnment for Interactive Applications)

- XML-based Languages with Schemas
- Support for Abstract Data Types
- Able to generate user interfaces including complex Javascipts and Ajax scripts
- More engineered and powerful language (e.g. Pacman)
## Mappings WSDL/AUI/CUI

<table>
<thead>
<tr>
<th>Data type</th>
<th>Abstract UI</th>
<th>Concrete UI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numerical</td>
<td>Numerical edit</td>
<td>Numerical edit</td>
</tr>
<tr>
<td>Numerical restricted into a range</td>
<td>Numerical edit in range</td>
<td>Numerical edit in range</td>
</tr>
<tr>
<td>Boolean</td>
<td>Single choice</td>
<td>Radio button</td>
</tr>
<tr>
<td>Enumeration with low cardinality</td>
<td>Single choice</td>
<td>Drop down list</td>
</tr>
<tr>
<td>Enumeration with high cardinality</td>
<td>Single choice</td>
<td>List box (single choice)</td>
</tr>
<tr>
<td>Array data type with low cardinality</td>
<td>Multiple choice</td>
<td>Check box</td>
</tr>
<tr>
<td>Array data type with high cardinality</td>
<td>Multiple choice</td>
<td>List box (multiple choice)</td>
</tr>
<tr>
<td>Text</td>
<td>Text edit</td>
<td>Text edit</td>
</tr>
<tr>
<td>Unknown type</td>
<td>Object edit</td>
<td>Object edit</td>
</tr>
</tbody>
</table>
MARI A Tool Requirements

(MARI A - Model-based Authoring Environment for Interactive Applications)

- 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.
Automatic vs Semiautomatic Development
Automatic generation

- + More Efficient
- + More Consistent
- - Less Control
- - Less Usability

More effective when the application domain is well-known
The Proposed 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.
- Inclusion of Usability Guidelines in Authoring and UI Generation
Conclusions and Future Work

- Automatic vs Semi-Automatic vs Manual Design
- Future work will be dedicated to
  - Improving tool support for the identified methodology
  - Relations between BPMN/BPEL and CTT/MARIA
  - Application to other case studies
- W3C Meeting in Cannes on October 21 on Model-Based User Interfaces and the Potential Role of Standards