Tools and Methods for the Design of Multi-Device User Interfaces

Fabio Paternò
fabio.paterno@isti.cnr.it
http://giove.isti.cnr.it/~fabio/

ISTI-C.N.R.
Pisa, Italy
Ever-increasing introduction of new types of interactive devices
How to support designers and developers?
How to obtain interfaces able to adapt to multiple devices (any device) while preserving usability?
## Structure of the Tutorial

<table>
<thead>
<tr>
<th>Time</th>
<th>Subject</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.00-9.15</td>
<td>Introduction, Basic Concept, Issues</td>
<td>15’</td>
</tr>
<tr>
<td>9.15-9.30</td>
<td>Model-based design of multi-device interfaces</td>
<td>15’</td>
</tr>
<tr>
<td>9.30-9.45</td>
<td>Task/Platform Taxonomy</td>
<td>15’</td>
</tr>
<tr>
<td>9.45-10.00</td>
<td><em>exercise – Multi-device interface</em></td>
<td>15’</td>
</tr>
<tr>
<td>10.00-10.15</td>
<td>Task Analysis and Modelling</td>
<td>15’</td>
</tr>
<tr>
<td>10.15-10.45</td>
<td>TERESA</td>
<td>30’</td>
</tr>
<tr>
<td>10.45-11.00</td>
<td>Semantic redesign for different interaction platforms</td>
<td>15’</td>
</tr>
<tr>
<td>11.00-11.30</td>
<td><em>Coffee Break</em></td>
<td>30’</td>
</tr>
<tr>
<td>11.30-11.45</td>
<td>Model-based design of multi-modal interfaces</td>
<td>15’</td>
</tr>
<tr>
<td>11.45-12.15</td>
<td>Migratory Interfaces</td>
<td>30’</td>
</tr>
<tr>
<td>12.15-12.30</td>
<td><em>exercise – Migratory interfaces</em></td>
<td>15’</td>
</tr>
<tr>
<td>12.30-12.45</td>
<td>Architectures for Migratory Interfaces</td>
<td>30’</td>
</tr>
<tr>
<td>12.45-13.00</td>
<td>Research agenda &amp; Conclusions</td>
<td>15’</td>
</tr>
</tbody>
</table>
Possible Views of an Interactive Systems

- Task and object – *I want to select a work of art*
- Abstract Interface – *Single selection object with high cardinality*
- Concrete Interface – *List Interaction object with X elements*
- Code – *List object in Java or XHTML or* ....
Significant Models in HCI

Models
- Task & Object Models
- Interactor
- User
- Platform
- Environment

Run time
Adaptivity
- Run-time events
- Reconfiguration

Design time
Multiple-Version

Task and object model
Abstract User Interface
Concrete User Interface
User Interface
Tools in Model-based environments

- Modelling Tools
- Analysis Tools
- Development Tools

Models
- Task & Object Models
- Abstract UI
- Concrete UI
- Context

Designer

Design criteria
Transformations

User Interface
Context of use
Reverse Engineering Tools
Adaptation

- **Adaptability** - Capacity of a UI to adapt its behaviour from an explicit human intervention
- **Adaptivity** - Capacity of a UI to adapt without any explicit human intervention
- What can be adapted? Presentation, Navigation, Content
Context-dependent Adaptation

About the environment:
- Location
- Temperature
- Light level
- Noise level
- ...

About the user:
- Preferences
- Knowledge level
- Goals
- Background
- ...

About the device:
- Screen resolution
- Bandwidth
- Browser capability
- Network
- ...

Tools and Methods for the Design of Multi-Device User Interfaces
Design of Multi-Device Interfaces: Current Practice

- Manual solutions,
  - expensive
- Transcoders,
  - low cost/low usability
- Style sheets,
  - partial solution
State of Art

- Aura project at CMU (adaptation at application level)
- Pebbles project at CMU (limited to appliances control)
- XIML has not public tool support
- UIML does not support high-level task descriptions
  [http://www.uiml.org/](http://www.uiml.org/) (developed by Harmonia and cooperation with Virginia Tech)
XForms

- Apply concepts from model-based design
- Separate presentation from content (form controls markup is separated from data-types and returned values)
- XForms 'native' form controls are device-independent
- Reduce need for scripting through client-side checking
- XML instance is returned allowing strong typing
Motivations for task analysis and modelling

- Main usability principle:
  - *Focus on the users and their tasks*

- Tasks represent the logical activities performed for reaching user goals

- Need for modelling is most acutely felt when the design aims to support system implementation as well
  - Especially for large projects and some application domains
Definitions

- Task - activity that has to be performed to reach a goal
- Goal
  - desired modification of state
  - Attempt to receive state information
- Each task is associated with one goal
- Each goal is associated with one or multiple tasks
- Multiple abstraction levels - Basic task
- Task Analysis
- Task Models
Task - related issues in multi-platform environments

- Platform definition
- Same task on multiple platforms in the same manner
- Same task on multiple platforms but performed in different manner
- Dependencies among tasks performed on different platforms
- Tasks meaningful only on a single platform type
### Examples of Platform-dependent tasks

<table>
<thead>
<tr>
<th>Desktop system</th>
<th>Mobile System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparing prices of flights and making reservations.</td>
<td>Checking status of a particular flight.</td>
</tr>
<tr>
<td>Gathering background on a company, including maps.</td>
<td>Getting driving directions to a company—while on the road.</td>
</tr>
<tr>
<td>Browsing medical information.</td>
<td>Monitoring a medical condition.</td>
</tr>
<tr>
<td>Reading a movie review and/or watching a trailer.</td>
<td>Purchasing a cinema ticket to avoid the line.</td>
</tr>
</tbody>
</table>
Same task on multiple platforms with different user interface objects
Same task on multiple platforms with different task decomposition
Dependencies among tasks performed on different platforms
Exercise
Multi-device Interfaces
TERESA

- Mixed initiative
- Model-based
- XML-based
- Flexible development
- Web-oriented but can be extended to other environments
- Available at http://giove.isti.cnr.it/teresa.html
One Model – Many Interfaces
TERESA Environment for Flexible Development

Task model for envisioned nomadic applications

System task model
- Cellphone (XML)
- Desktop (XML)
- Voice (XML)

Abstract UI
- Cellphone (XML)
- Desktop (XML)
- Voice (XML)

Concrete UI
- Cellphone (XML)
- Desktop (XML)
- Voice (XML)

XHTML MP, WML ...
XHTML, Java ...
VoiceXML, Salt ...

Tools and Methods for the Design of Multi-Device User Interfaces
Two platform-independent languages: task (CTT) and abstract interface

One level (concrete interface) represented through a number of platform dependent languages

Designers aware of the potential platforms (not devices) early on in the design process

Method allows developers to avoid dealing with a plethora of low-level details (transformation from concrete description to implementation is automatic)

Easy to add support for new implementation languages
Another example
Communication-oriented Composition operators

- **Grouping**: a set of elements logically related to each other

- **Ordering**: existing of an order among interactors (i.e. temporal)

- **Relation**: One interactor related to a group of other interactors (i.e. disabling them)

- **Hierarchy**: a logical hierarchy among a set of interactors
Structuring the User Interface

- **Grouping** – Example: Task decomposition -> grouping of corresponding interaction techniques
- **Ordering** – Sequential communicating tasks -> adjacent interaction techniques
- **Relation** – Control tasks (one to many relations)
- **Hierarchy** – Frequent tasks -> More space or larger attributes
The Structure of the Abstract User Interface

- Language platform-independent
- Interactors (selection, navigator, activator, ...)
- Communication-oriented composition operators
- Connections among presentations
The Structure of the Abstract User Interface

User Interface

Presentation 1
- Grouping
- Selection
- Edit
- Navigator

Connection

Presentation 2
- Hierarchy
- Description
- Edit
- Multiple-selection
A nomadic museum application
TERESA support in Development

- Choice of device platform/type
- Settings for general attributes,
- How to implement composition operators
- How to implement interactors
- Summary of design choices and preview
- Recording and reuse of concrete aspects defined
Concrete User Interface

- Defines some concrete aspects of the user interface
- Provides indications for the implementation of abstract interactors

EXAMPLE

**Abstract level**

```
- <interactor id="Go_to_section3">
  <interaction category="interaction">
    <control type="control">
      <navigator object="navigator"/>
      <control>
    </control>
  </interaction>
  </interactor>
```

**Concrete level**

```
- <interactor id="Go_to_section3">
  <interaction>
    <control>
      <navigator/>
      <control>
    </control>
  </interaction>
  </interactor>
```
Example of platform-dependent concrete interactor choice

EXAMPLE:
Single choice abstract interactor

<table>
<thead>
<tr>
<th>Cardinality</th>
<th>Desktop Computers</th>
<th>Mobile Phones</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low cardinality</td>
<td>Radio Button</td>
<td>Radio Button</td>
</tr>
<tr>
<td>Medium cardinality</td>
<td>List Box</td>
<td>Drop Down List</td>
</tr>
<tr>
<td>High cardinality</td>
<td>List with scrollbars</td>
<td>Drop Down List</td>
</tr>
</tbody>
</table>
Example of platform-dependent composition operator implementation

EXAMPLE: Grouping Operator

- Desktop Computers
  - Fieldset
  - Bullet
  - Background Color
  - Column-oriented organization
  - Row-oriented organization

- Mobile Phones
  - Unordered List On Column
  - Fieldset (only for medium-large phones)
Example of TERESA-generated User Interface DEMO
Use of Reverse Engineering

Tasks and Objects
Abstract UI
Concrete UI
Final UI

Redesign

Tasks and Objects
Abstract UI
Concrete UI
Final UI

Transcoding

Platform X
Platform y

Tools and Methods for the Design of Multi-Device User Interfaces
Semantic Redesign

- Use of abstractions for changing the design for a new target platform
- Use of reverse engineering to obtain the abstractions
- Different possible solutions
Semantic Redesign with Forward Engineering

Abstract User Interface

Concrete User Interface

Redesign

Abstract User Interface

Concrete User Interface

Desktop User Interface

Mobile User Interface
Reverse and Forward Engineering

Abstract User Interface

Concrete User Interface

Redesign

Abstract User Interface

Concrete User Interface

Desktop User Interface

Mobile User Interface

Tools and Methods for the Design of Multi-Device User Interfaces
Support for Redesign

- Page splitting based on the composition operators and the number of interactors
- Connections: original ones + those derived from page splitting
- Images: resize depending on target device keeping the same aspect ratio
- Tables for converting terms and labels
From Desktop to Cell-phone

Download Software

Please fill in the form and then select the link that will appear on the bottom.

Name:
LastName:
Organization:
E-Mail:
City:
Country:
Purpose:

Grouping 0

Grouping 1

Grouping 2

Language:
French
English

System:
Win 2000

Submit Cancel
From Desktop to Cell-phone

Download Software

Please fill in the form and then select the link that will appear on the bottom.

Form - Part 1
Form - Part 2

Submit  Cancel

Language:  French  English
System:  Win 2000

Name:  
LastName:  
Organization:  
E-Mail:  
next
home

City:  
Country:  
Purpose:  
\[\square\text{Research Project}\]
\[\square\text{Application Designer}\]
\[\square\text{Teaching}\]
\[\square\text{Development}\]
List Subscription:  
\[\bigcirc\text{Yes}\]
\[\bigcirc\text{No}\]
prev
home
Another example of semantic redesign
Task-based Semantic Redesign

Abstract User Interface

Concrete User Interface

Nomadic Task Model

Redesign

Abstract User Interface

Concrete User Interface

Desktop User Interface

Mobile User Interface

Tools and Methods for the Design of Multi-Device User Interfaces
From Desktop to Cell-phone
Semantic redesign - demo
Vocal Interaction

- Characteristics: linear, not persistent, faster and more natural for some operations
- Provide feedback to check the status of application
- Brief prompts and short lists of options to reduce memory capability
- Management of events (no-input, no-match, help)
Speech implementation of composition operators

- **Grouping:**
  - Insert a sound
  - Insert a pause
  - Use some keywords
  - Use a specific volume of synthesizer voice

- **Ordering**
  - Alphabetical order
  - Use some keywords

- **Relation**
  - Change context (change type of menu)

- **Hierarchy**
  - Increase or decrease the volume of synthesizer voice
Specifying general parameters for all presentations
Vocal Interaction Generation - DEMO
Welcome message
Management of no input event
Provide feedback
Description Object
Composition operators
Generation of Multi-Modal Interfaces

- X+V
  - Supported by OPERA Browser, also for PDAs
- EMMA not supported by any public tool
- SMIL not interaction oriented
- Identification of design criteria for multimodal platforms
**Interactor Interaction**

- **Prompt**: represents the interface output indicating that it is ready to receive an input.
- **Input**: represents how the user can actually provide the input.
- **Feedback**: represents the feedback of the system after the user input.
Multimodal properties

- Complementary
- Redundancy
- Assignment
- Equivalence
Design of Multimodality Support

- Identification of new platforms (multimodal desktop, multimodal PDA, …)
- Design how to support composition operators and interactors
  - multimodal desktop:
    - composition operators -> graphically supported
    - interactors -> graphical prompt, input either graphical or vocal, feedback in both modalities
  - multimodal pda:
    - composition operators -> supported both graphically and vocally
    - interactors -> vocal prompt, input either graphical or vocal, feedback in both modalities
Multimodal desktop

- Composition operators → Graphical Assignment
- Interactors
  - OnlyOutput → Graphical Assignment
  - Interaction
    - Prompt: Graphical Assignment
    - Input: Equivalence
    - Feedback: Graphical Assignment
Multimodal PDA

- Composition operators → Redundancy
- Interactors
  - OnlyOutput → Complementary/Redundancy
- Interaction
  - Prompt: Redudancy
  - Input: Equivalence
  - Feedback: Redudancy
Welcome in the movie description page. In the Robots film a world is populated entirely by robots. This is the story of a young genius, Rodney, who wants to make robots capable....

Would you like to book a ticket or come back to home?