Filling the gap between users and objects: a multichannel interactive environment

Davide Carboni, Gavino Paddeu, Stefano Sanna, Andrea Piras
{dcarboni, gavino, gerda, piras}@crs4.it

CRS4 – Center for Advanced Studies, Research and Development in Sardinia

WOA 2003 – Villasimius (Cagliari)
Environment in evolution

- Mobile computing
- Pervasive computing
- Domotics
- Wearable technologies
Problem

☐ An application is available only on a kind of devices.

☐ Often developers have to re-implement it in order to reach other devices.

■ Expensive tasks in terms of money and time.
The simplest approach

- It is not necessary to re-implement the logic of the application but only to change its user interfaces.
A solution: model-based user interfaces

- A model is usually defined with a tool that generates a task model.
- The task model is translated in text document written following an ad-hoc declarative language.
- A processor parses the text and composes the user interface.
Another solution: based on Model-View-Control

- Decompose the application logic in a set of models that are strictly bound to the data.
- For each model, the designer can provide one or more user interfaces, the View-Control subsystem.
Our approach

- Starting from the MVC.
- The model is a set of objects.
- Each model is analyzed for rendering the user interface according to the features of the target device.
- MORE – Multichannel Object Renderer
Model translation

☐ Create a container for the actual channel.
☐ Detect fields and methods of the object model.
☐ For each field:
  ■ search the coupled widget
  ■ if it exists, add it to container
  ■ else add a link
☐ For each method:
  ■ add widget to container
Categorization of devices

- Fat Client: the device can download and execute mobile code (desktop, notebooks, tablet-PCs).
- Thin Client: the device can execute local code and can download/upload data (PDA, set-top box, programmable phone).
- Web-like Client: the device is equipped with 3rd party browser.
MORE architecture
An object model: personal agenda
Personal agenda on fat clients
User interface generation on fat clients

- The output of the model translation is the user interface.
- The actions performed on the user interface are directly forwarded to the object model.
User interface composition on thin clients

- The model translation is performed on server side and produces an XML document.
- On the device, a small application is installed. It receives the XML document, parses it and generates the user interface.
- Three implementations of MORE: iTV, PDA and smartphone.
Personal agenda on thin clients
User interface on web-like clients

- As for thin clients, the model translation produces an XML document.
- A XSLT processor takes the XML document and a selected set of XSL style sheets and returns the mark-up document for the user browser.
- The processing is server side.
- Four implementations of MORE: “high-end” HTML, “low-end” HTML, WML and VoiceXML.
Personal agenda on “high-end” HTML browser

User login

user name

andrea

login

Close User Session

Personal Agenda: andrea

meetings

Jones Tennis

View Delete

user status

Known

add meeting

Close User Session

> User login

Tennis

confirmed
Yes [ ] No [ ]

date
2003-06-23 10:44 (yyyy-mm-dd hh:mm)

name
Tennis

time
15:00

Close User Session

> User login > Personal Agenda: andrea

WOA 2003 – Villasimius (Cagliari)
Personal agenda on WML browser
Pros

- The object models are ready-to-use for each category of devices with an existing implementation of MORE.
- Easy to extend as new devices categories emerge.
- Fast prototyping.
Cons

- Limited customization of user interface.
- Difficult access to peripherals connected to the user device.
Thanks for your attention

Andrea Piras
piras@crs4.it