A Graphics Software Architecture for High-End Interactive TV Terminals

Pablo Cesar
Helsinki University of Technology
Telecommunications Software and Multimedia Laboratory
pcesar@tml.hut.fi
Outline

• Introduction
• Definitions
• DTV
• Challenges
• Contribution
• Services
• Conclusions
Introduction (1/2)

• What is this thesis about?
  – About software for digital television receivers
  – About multimedia capabilities in next-generation receivers
  – About visualization of content and interaction

• Why to write it?
  – Bringing multimedia to the mass audience is a challenge
  – As an input to all the standardization efforts
  – New ideas for manufacturers
  – To get a PhD

• How?
  – Work on Otadigi – DTV Broadcast system
  – Development of Ubik – DTV Receiver
Introduction (2/2)

- Understanding and explaining DTV
  - Understanding DTV and its standards
  - Trying to explain it to the general public
- Extending and filling the gaps of those standards
  - Multimedia capabilities
- Providing a reference implementation
  - How to implement the extensions
  - Showing a number of services that can be provided (more attractive environment)
- Proposing specific graphics architecture
  - To understand current needs of media stations
  - To improve previous proposed architectures
Definitions (Multimedia)

"Computer mediated applications that integrate and present different media objects, which are arrange spatially and temporally. Moreover, user interaction can control the behavior of the application."

Outline
Introduction
Definitions
DTV
Challenges
Contribution
Services
Conclusions

Multimedia Objects
Visual Style
Temporal Dimension
Spatial Layout
User Interaction

Application Logic
Definitions
(Multimedia Objects & Visual Style)

• Discrete Media
  – Icons: Semantic images (e.g., stop symbol). Require the user to have previous knowledge
  – Graphics: Computer generated. Can be 2D or 3D graphics depending on the goal
  – Images: Natural source (e.g., photograph)
  – Text: **Size**, font type, **Color**

• Continuous Media
  – Motion pictures (audio + video)
Definitions (Spatial Layout)

- **Absolute**
  - Coordinates relative to origin

- **Directional relations**
  - Define order in space

- **Topological relations**
  - Disjoint, touch, inside of, covered by...

- **Text Flow**
  - One-dimensional flow showed in two-dimensional area
Definitions (Temporal Dimension)

• Temporal Models:
  – Definite: For example, 6 seconds
  – Indefinite: For example, when user clicks
  – Parallel and Sequential relations: For example, start these two videos at this moment or start this video after this other one

• Animation:
  – Mixture of temporal dimension and spatial layout (i.e., position of an object changes in time)
Definitions (User Interaction)

• Different Levels of Interaction (Aleem):
  – Passive: Only visualization
  – Reactive: Limited interaction
  – Proactive: Choose a path or make selections
  – Reciprocal: Corresponds to user authoring of information

• Interaction Models (Boll):
  – Navigational: Choice to decide where to go next
  – Design: User can modify the visual style of the presentation (e.g., colors)
  – Movie: User can control the global time (e.g., VCR capabilities)
Definitions (Development Languages 1/2)

• Requirements
  – Supported media types: Audio, video, text, graphics, and animation
  – Arrangement of the signs: Spatial and temporal
  – Interaction: Passive, reactive, proactive, and reciprocal
  – Difficulty to use (threshold)
  – Expressional power (i.e., ceiling)
  – Safety of distribution
  – Interoperability

• Development Languages
  – Compiled languages: Normally, used for system software (e.g., operating system) and resource demanding services
  – VM languages: A Virtual Machine is an abstraction of the computing environment (e.g., Java)
  – XML-based Languages: Only what has to be done, not how
### Definitions (Development Languages 2/2)

<table>
<thead>
<tr>
<th></th>
<th>Threshold</th>
<th>Ceiling</th>
<th>Interoperability</th>
<th>Safety of Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compiled Languages</td>
<td>+++</td>
<td>+++</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>VM Languages</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>XML-based Languages</td>
<td>+</td>
<td>+</td>
<td>+++</td>
<td>+++</td>
</tr>
</tbody>
</table>

- Compiled Languages: Not so safe to distribute, only for native applications
- VM Languages: In DTV world is called Procedural Environment
- XML-based Languages: In DTV world is called Declarative Environment
DTV (What is it?)

- Audiovisual signal is compressed and transported by digital means

<table>
<thead>
<tr>
<th>Audio</th>
<th>Video</th>
<th>Data</th>
<th>. . .</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>MPEG-2 Stream</td>
</tr>
</tbody>
</table>

- Spectrum efficiency:
  - That’s what we say when we can fit more channels in the same bandwidth
- Each broadcaster decides how to use its own bit-rate:
  - Only television programs
  - Only interactive applications
  - Television program(s) plus interactive application(s)
- Interactive applications as value added services
  - Java applications (procedural)
  - XHTML documents (declarative)
DTV (Requirements)

**Infrastructure**
- **Home**
  - Receive TV: Antenna
  - Watch TV: TV set
  - Tune to channel: Digital Set-top Box
- **Broadcast**
  - A/V stream: MPEG-2 encoder
  - Data Stream: Object Carousel
  - Transmission: Multiplexer, Modulator, Transmitter, Antenna

**Smooth Transition**
- **Horizontal-Market**: Standards agreed by major player
- **Easiness of use**: usability considerations
- **Price**
- **Viewer Expectations**
  - A/V: Better quality
  - Content: More channels
  - Services: Extra services
- **Receiver**
  - A/V Support
  - Application Support
  - Return Channel
DTV (Broadcast Chain)

Outline
- Introduction
- Definitions
- DTV
- Challenges
- Contribution
- Services
- Conclusions

13/28
DTV (Standards)

- Standards = Horizontal-Market
  - Applications and decoders can be developed by different developers and manufacturers
  - Freedom to choose and interoperability assurance
- DVB Standards (Digital Video Broadcasting)
  - DVB is the European initiative in Digital Television field
  - Specifies the transmission in different mediums: DVB-S (Satellite), DVB-C (Cable), DVB-T (Terrestrial)
- MHP (Multimedia Home Platform)
  - Procedural environment (DVB-J) is a set of Java APIs
  - Declarative environment (DVB-HTML) is an XML-based solution
Challenges

• Low market penetration
  – Viewer expectations:
    • More channels
    • Entertainment services
  – Price

• Smooth Transition
  – Interoperable Standards
  – Profiling
    • Different profiles depending on user needs
    • Evolution profiles, so the providers and consumers go step by step

• Worldwide Solutions (GEM)
  – Development of applications by any country
  – That is one of the reasons of Internet success
Contribution (Profiles 1/3)

**Broadcast: Basic**
- **Requirements:**
  - Television usage: sit on the sofa and watch the favorite show at a given time
  - Catch majority of the population
- **Proposed profile:**
  - Audio Visual Content Digitized
  - Examples: DVB-T/C/S (Europe), ISDB (Japan), ATSC (USA)

**Broadcast: Enhanced**
- **Requirements:**
  - Interactive applications with internal logic
  - Procedural environment: Java
- **Proposed profile:**
  - DVB-J (as part of GEM, worldwide accepted version of MHP 1.0)
Contribution (Profiles 2/3)

Interactive: Basic
- Requirements:
  - Browse simple XML documents
  - Limited navigation
  - Images, text
- Proposed profile:
  - XHTML Basic + CSS

Interactive: Internet Access
- Requirements
  - Browse XML Compound Documents (complex)
  - Interactive
  - Temporal Synchronization
  - Video and audio
- Proposed profile:
  - XML Compound Documents (SMIL + XForms)
Interactive: High-End

- Requirements
  - Integration of all kinds of multimedia objects
  - Temporal Dimension

- Proposed profile:
  - DVB-J + OpenGL Java Wrappers
  - JMF media player (all kind of formats)
Contribution (Graphics Architecture 1/2)
Contribution (Graphics Architecture 2/2)

- Lack of pointing device
- One task at the time
- Screen composed of multimedia objects
- Seamless integration of video, 2D and 3D objects
- A layered architecture, so developers can implement at any level

- HAL: renders the final graphics output
- Graphical Context: cross-platform abstraction of the rendering region
- Graphical Environment: means to control different contexts
- GUI Toolkit: "ready-made" user interface widgets
- HLL: to develop simple services
Contribution (Implementation)

<table>
<thead>
<tr>
<th>MHP</th>
<th>High Level Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SMIL</td>
</tr>
<tr>
<td></td>
<td>XForms</td>
</tr>
<tr>
<td></td>
<td>XHTML</td>
</tr>
<tr>
<td>GUI Toolkit</td>
<td>FTV</td>
</tr>
<tr>
<td></td>
<td>JMF</td>
</tr>
<tr>
<td></td>
<td>JOGL</td>
</tr>
<tr>
<td>Graphical Environment</td>
<td>SDLAWT</td>
</tr>
<tr>
<td>Graphical Context</td>
<td>SDL</td>
</tr>
<tr>
<td>Hardware Abstraction Layer</td>
<td>DirectFB OpenGL</td>
</tr>
</tbody>
</table>

Outline
Introduction
Definitions
DTV
Challenges
Contribution
Services
Conclusions
Services (Otadigi 1/3)

Outline
Introduction
Definitions
DTV
Challenges
Contribution
Services
Conclusions

Navigator

Simple Application
Services (Otadigi 2/3)

Red Cross Service
Services (Otadigi 3/3)

SMIL Player

Enhanced TV

Outline
Introduction
Definitions
DTV
Challenges
Contribution
Services
Conclusions
Services (Ubik 1/3)

Example native 3D Graphics
- Some games downloaded from a portal
Services (Ubik 2/3)

- Example Java 3D Commercial
  - Car
  - Interactive
Services (Ubik 3/3)

Example of Super Teletext

Example of E-Learning application
Conclusions

• This thesis tries to understand the situation in Digital Television environment and to explain it
• This thesis tries to improve current standardization activities and provides a reference implementation (services, how-to tutorial)
• This thesis proposes a graphics architecture for media stations

Thanks!
Questions, Comments? Discussion