Device-Independence with UIML
(User Interface Markup Language)

Marc Abrams
marc@harmonia.com

uiml.org

W3C Workshop on Web Device Independent Authoring
October 2000
Quotes from Position Papers

• New XML language should
  – Allow author-once-deploy many scenarios
  – Achieve clean separation between behavior, content, presentation [Ruud Siebelink]

• Issue is *interaction* not *presentation* [Paul Smethers, WAP]

• Ideal solution: write *well-formed code* once [Jansen]

• Semantics [meaning of Web content] must be made clear at primary design level [William Loughborough]

• Must adapt to new devices not envisioned [Ralph Case, Stephane Maes]
Where UIML Fits In*

Application database

Content (XML, audio,…)

Device Adaption

UIML

Using CC/PP

XHTML  VoiceXML  XHTML  WML

* Diagram from Dave Raggett’s talk
UIML... One Part of a Solution

- One canonical representation of UI for any device, language, OS, UI-metaphor
- 3+ years in development at Harmonia, Center for HCI at Virginia Tech
- Tools downloaded in 40+ countries
- Can be compiled to lots of things
- Anyone can freely implement UIML
- Objective is open standard
Problem with Existing Approaches

• Suggested way to annotate existing markup:

```html
<card>
  <select class="DISPLAY_SMALL">
  ...
</select>
</card>
```
Key Concept:
UIML is a “Meta” Language

• XML
  – Doesn’t define tags (<P>, …)
  – Must add doc type definition to make it useful
  – No need to change XML as new tag sets invented

• UIML
  – Doesn’t define tool-kit specific tags (<Menu>, …)
  – Uses a few powerful tags (<part>, <property>, …)
  – Must add toolkit vocabulary to make it useful
  – No need to change UIML as new devices invented
UIML Model
Underlying principle of single authoring is MVC [to separate content/structure] [Ralph Case, Stephane Maes]

6-way separation of UI description (vs. 3-way MVC)
What parts comprise the UI & what’s their relationship?
UIML Skeleton – Part 2

```xml
<?xml version="1.0" ... ?>
<uiml version="2.0">
  <interface>
    <structure>...</structure>
    <style>...</style>
  </interface>
</uiml>
```

What presentation style for each part?
What content for each part? (text, sounds, image, ...)

<?xml version="1.0" ... ?>
<uiml version="2.0">
  <interface>
    <structure>...</structure>
    <style>...</style>
    <content>...</content>
  </interface>
</uiml>
<xml version="1.0" ... />
<uiml version="2.0">

<interface>
  <structure>...</structure>
  <style>...</style>
  <content>...</content>
  <behavior>...</behavior>
</interface>

</uiml>
UIML Skeleton – Part 5

```xml
<?xml version="1.0" ... ?>
<uiml version="2.0">
  <interface>
    <structure>...</structure>
    <style>...</style>
    <content>...</content>
    <behavior>...</behavior>
  </interface>
  <peers>...</peers>
</uiml>
```

How to connect to outside world?
(business logic, UI toolkit object)
<peers> Maps Classes to Targets

<d-class name="JButton" ... maps-to="javax.swing.JButton"> ... </d-class>

Versus

<d-class name="JButton" ... maps-to="html:input"> ... </d-class>

This part is written once, like a device driver for an OS. Events and calls to outside world handled similarly.
**NxM Problem (Old Way)**

- App composed on
  - $M$ “pages”
  - accessed via $N$ devices
  - requires $N \times M$ authoring steps

[Ralph Case, Stephane Maes]
**NxM Problem (New Way)**

- App composed on
  - \( M \) “pages”
  - accessed via \( N \) devices
  - requires \( NxM \) authoring steps

[Ralph Case, Stephane Maes]

Reduces to \( N+M \)
UIML Permits Development Continuum

- Use device-specific vocabulary: `<part Class="JPopupMenu">` versus `<part Class="Select">`
- Use generic vocabulary: `<part Class="Menu">`

Richest user experience vs. Fastest to build
UIML Permits Families of UIs
Another Perspective...

- Machine language
- Assembly language
- "High-level" language (C++, Java)
- Scripting languages
- Device-dependent markup
- Device-independent markup
Still...

UIML is Not a Silver Bullet...

Some open problems:

• Aid/enforce accessibility guidelines [Jon Wu]

• Support auto adaptation/personalization [Ruud Siebelink]

• Reorganizing UI:
  – Many apps will need to be re-designed entirely [Guido Grassel]
  – 1 page in a desktop Web browser might be split into 2 screens for TV [Peter Ferne]
For More Info

Visit uiml.org

Upcoming: European Workshop on UIML – January 2001 in Paris