

Device Independent User Interfaces in XML

Roland A. Merrick
Roland_Merrick@uk.ibm.com

BeCHI
2001-05-09

The good old days...

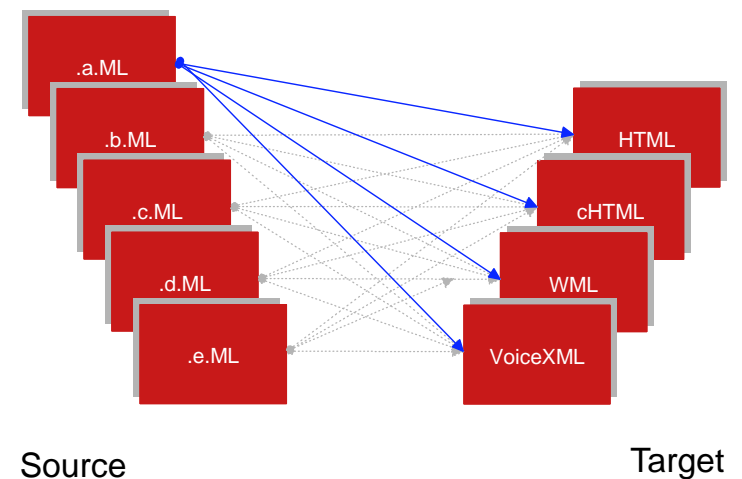
Cross platform applications have been defined as those which run on a varied collection of Operating systems (same is true for client or server applications). Java was introduced with its WORA promise. However, most client systems were very similar, relatively large color screens, keyboards, mice etc..

The advent of Pervasive Computing has introduced a broad spectrum of devices and form factors on which applications may need to run.

Different Form Factors

	PC	Palm	Telephone
	1024*768, 16bit color display	160*160, 2bit gray	Speaker
	Keyboard, mouse	Stylus	Keypad, microphone
Choose title from: Mr., Mrs., Miss, Ms.	A set of 4 radio buttons.	4 entries in a "Push Button"	"Speak" options, press #1 for Mr. press #2 for Mrs. press #3 for Miss press #4 for Ms.

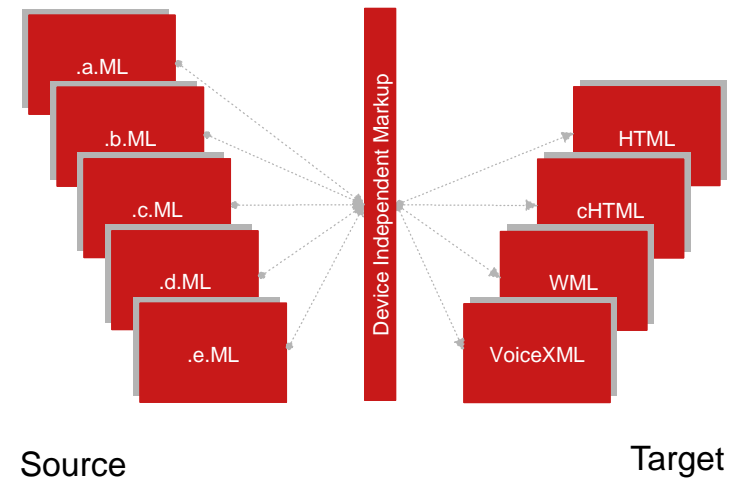
XSLT explosion!



XSLT explosion!

Number of Style Sheets = Source * Target

XSLT tamed!



XSLT tamed!

Number of Style Sheets = Source + Target

Objectives

- keep "intent" separate from rendering
 - specify the purpose, or intent, of an interaction rather than its appearance
- users may use multiple form factors
 - single intent should run on many devices
 - predictable, repeatable representation for a construct

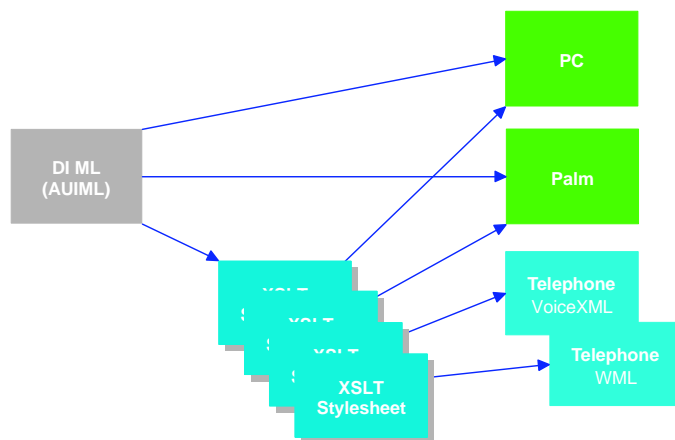
Objectives (continued)

- simplify development
 - increase speed and accuracy
 - reduce/change skills required
 - ideally support by tools, but human readable and comprehensible without any tools
 - declarative *i.e.* what not how

AUIML vocabulary

In 1998 **IBM** undertook an Advanced Technology project to develop a Device Independent Markup Language in XML. This project went through a number of phases but ended up with an XML Vocabulary called **Abstract User Interface Markup Language (AUIML)**.

Some possible transformations



AUIML Element types

- Simple Data types
 - semantics & representation
 - DATE, TIME, NUMBER, STRING, BOOLEAN
 - IMAGE, AUDIO
- Structural Elements
 - CHOICE
 - GROUP
 - TABLE
 - TREE
- Actions
 - ACTION, ACTION-GROUP

Common metadata elements

<CAPTION>

This is the most concise item of descriptive information. In a Graphical User Interface this would typically be text which is rendered as a field prompt or caption when the parent element is displayed.

<HINT>

Provides a short description of the purpose for this data item. It should require little, if any, explicit action from the user to cause it to be rendered. With a visual rendering this would typically be rendered in the style of Hover help. With an audio system, an unexpectedly long pause could be the trigger.

<HELP>

Provides a longer detailed description of the data item, typically a number of sentences. This would not normally be rendered unless a user explicitly requests it.

AUIML example - part 1

```
<DATA-GROUP NAME="MySample">
  <CAPTION>
    <META-TEXT>Questionnaire</META-TEXT>
    <META-IMAGE SRC="Ques.gif" />
  </CAPTION>
  <GROUP NAME="Person">
    <CAPTION>Who are you?</CAPTION>
    <HINT>This section collects information about your identity. It would be a good idea if you filled in this page.</HINT>
    <GROUP NAME="PersonName">
      <CAPTION>Person Details</CAPTION>
      <HINT>Specify the person's title, first and last names, and middle initial.
        If you choose a title of MRS you will be asked for a maiden name.</HINT>
      <CHOICE NAME="PersonTitle" SELECTION-POLICY="SINGLE">
        <CAPTION>Title</CAPTION>
        <HINT>This is a set of valid titles for a person.</HINT>
        <STRING NAME="Mr">
          <CAPTION><META-TEXT><MN>M</MN>r.</META-TEXT></CAPTION>
        </STRING>
        <STRING NAME="Mrs">
          <CAPTION><META-TEXT>M<MN>r</MN>s.</META-TEXT></CAPTION>
        </STRING>
        <STRING NAME="Miss">
          <CAPTION><META-TEXT>M<MN>i</MN>ss.</META-TEXT></CAPTION>
        </STRING>
        <STRING NAME="Ms">
          <CAPTION><META-TEXT>M<MN>s</MN>.</META-TEXT></CAPTION>
        </STRING>
      </CHOICE>
    </GROUP>
  </GROUP>
</DATA-GROUP>
```

AUIML example - part 2

```
<STRING NAME="FirstName" OPTIMUM-LENGTH="12">
  <CAPTION>First Name</CAPTION>
  <HINT>The person's first, or given, name</HINT>
  <VALUE>Roland</VALUE>
</STRING>
<STRING NAME="Initial" MAX-LENGTH="1">
  <CAPTION>Initial</CAPTION>
  <HINT>The person's middle initial</HINT>
  <VALUE>A</VALUE>
</STRING>
<STRING NAME="LastName" MANDATORY="TRUE" OPTIMUM-LENGTH="12">
  <CAPTION>Last Name</CAPTION>
  <HINT>The person's last, or family, name</HINT>
  <VALUE>Merrick</VALUE>
</STRING>
<GROUP NAME="mn" ITEM-NAME="Mrs" TEST="SELECTED" CONDITION="TRUE">
  <CAPTION>Maiden Name</CAPTION>
  <STRING NAME="MaidenName" OPTIMUM-LENGTH="12">
    <CAPTION>Maiden Name</CAPTION>
    <HINT>Your Maiden</HINT>
  </STRING>
</GROUP>
</GROUP>
</DATA-GROUP>
```

AUIML transcoded to Dynamic HTML

The screenshot shows a web form titled "Questionnaire" with the following content:

Questionnaire

Who are you?
This section collects information about your identity. It would be a good idea if you filled in this page.

Person Details
Specify the person's title, first and last names, and middle initial. If you choose a title of MRS you will be asked for a maiden name.

Title: Mr. Mrs. Miss Ms.

First Name:

Middle Name:

Last Name:

Buttons: [Back] [Next] [Print] [Cancel] [Help]

AUIML custom renderer using Swing



AUIML custom renderer on Palm



Native Java renderer interpreting an AUIML document.

XForms Architecture



XForms example - part 1

```
<xform>
  <model name="MySample">
    <group name="Person">
      <group name="PersonName">
        <string name="PersonTitle" enum="closed">
          <value>Mr</value> <value>Mrs</value> <value>Miss</value> <value>Ms</value>
        </string>
        <string name="FirstName" />
        <string name="Initial" maxLength="1" />
        <string name="LastName" required="true" />
        <string name="MaidenName" relevant="equals(value(PersonTitle), 'Mrs')" />
      </group>
    </group>
  </model>
  <instance>
    <Person>
      <PersonName>
        <PersonTitle>Mr</PersonTitle>
        <FirstName>Roland</FirstName>
        <Initial>A</Initial>
        <LastName>Merrick</LastName>
      </PersonName>
    </Person>
  </instance>
</xform>
```

XForms example - part 2

```
<group>
  <caption>Who are you?</caption>
  <help>This section collects information about your identity. It would be a good idea if you filled in this page.</help>
  <group>
    <caption>Person Details</caption>
    <help>Specify the person's title, first and last names, and middle initial.
      If you choose a title of MRS you will be asked for a maiden name.</help>
    <textbox ref="Person/PersonTitle">
      <caption>Title</caption><help>This is a set of valid titles for a person.</help>
    </textbox>
    <textbox ref="Person/FirstName" style="width:12">
      <caption>First Name</caption><help>The person's first, or given, name.</help>
    </textbox>
    <textbox ref="Person/Initial">
      <caption>Initial</caption><help>The person's middle initial.</help>
    </textbox>
    <textbox ref="Person/LastName" style="width:12">
      <caption>Last Name</caption><help>The person's last, or family, name.</help>
    </textbox>
    <textbox ref="Person/MaidenName" style="width:12">
      <caption>Maiden Name</caption>
    </textbox>
  </group>
</group>
```

Demos

- Dynamic HTML
- Java swing
- Palm

References

www.ibm.com/ibm/easy/eou_ext.nsf/Publish/809
math.uma.pt/tupis00/submissions/azevedoroberts/azevedoroberts.html
www.w3.org/2000/09/Papers/Overview.html
www.w3.org/2000/10/DIAWorkshop/statements.html
www.w3.org/2001/di/
www.w3.org/MarkUp/Forms/
[www.w3.org/TR/xforms/]

Roland_Merrick@uk.ibm.com