

# XForms At A Glance

T. V. Raman  
IBM Research  
<tvraman@us.ibm.com>

May 13, 2003

## XForms Implementations

Implementation	Description
X-Smiles	Open Source XML browser from HUT, Finland.
FormsPlayer	XForms plug-in for IE
Novell XForms	Stand-alone XForms Java client
IBM Alphaworks	XForms on the server and client

## XForms Features At A Glance

<b>Declarative</b>	Declarative XForms are easier to maintain.
<b>Strong typing</b>	Submitted data can be checked using off-the-shelf XML tools.
<b>Schema re-use</b>	XForms enables the re-use of business rules.
<b>Schema augmentation</b>	Enables creation of dynamic interaction.
<b>XML submission</b>	Obviates custom server-side logic to marshal data.
<b>Internationalization</b>	Using XML 1.0 makes the submitted data I18N ready.
<b>Accessibility</b>	User interface controls are designed for accessibility.
<b>Device independence</b>	Abstract user interface controls lead to intent-based authoring.
<b>Localization</b>	Labels and help text can be referenced via URIs.
<b>Actions</b>	Declarative actions obviate common uses of scripting.

## XForms Components

Component	Description
Model	<ul style="list-style-type: none"><li>• Encapsulate all data aspects of a form.</li><li>• Uses XML Schema to define constraints.</li><li>• Uses XPath to define model properties.</li><li>• Captures <i>what</i>, <i>how</i> and <i>where</i> to submit.</li></ul>

*continued on next page*

<i>Continued from previous page</i>	
<b>Component</b>	<b>Description</b>
Properties	<ul style="list-style-type: none"> <li>• Model properties capture application constraints.</li> <li>• Enable reactive user interfaces.</li> <li>• Encourage declarative authoring in place of scripts.</li> </ul>
UI Binding	<ul style="list-style-type: none"> <li>• Connects user interface to the model.</li> <li>• Uses XPath to address nodes in the instance.</li> </ul>
Controls	<ul style="list-style-type: none"> <li>• UI controls collect user input.</li> <li>• <i>Bind</i> to underlying model.</li> <li>• Encapsulate all relevant metadata.</li> <li>• Controls are designed for accessibility.</li> <li>• Abstract controls encourage device independence.</li> </ul>
UI	<ul style="list-style-type: none"> <li>• UI constructs aggregate user interface controls.</li> <li>• Encourage intent-based authoring.</li> <li>• Create dynamic user interaction.</li> </ul>
Events	<ul style="list-style-type: none"> <li>• Eventing brings a user interface to life.</li> <li>• XML Events gives access to DOM2 eventing.</li> <li>• Extensible means of attaching dynamic behavior.</li> </ul>

Table 1: XForms at a glance.

## XML Standards

Standard	Role in XForms
XML	<ul style="list-style-type: none"><li>• Encapsulate structured instance data.</li><li>• XML DOM forms underlying model.</li><li>• Serialized instance data is I18N ready.</li></ul>
Namespaces	<ul style="list-style-type: none"><li>• Partition space of element and attribute names.</li><li>• Enable different XML vocabularies to co-exist.</li><li>• Used to create modular XForms applications.</li><li>• Key to integrating XForms into XML host languages.</li></ul>
Schema	<ul style="list-style-type: none"><li>• Enable object-oriented descriptions of XML.</li><li>• Rich set of data types.</li><li>• Enables off-the-shelf validation.</li><li>• Used to capture static constraints in the model.</li></ul>
XPath	<ul style="list-style-type: none"><li>• Locators address portions of an XML instance.</li><li>• Locators used to bind controls to the model.</li><li>• Locators used to bind model properties.</li><li>• XPath is a side-effect free expression language.</li><li>• Express dynamic properties within XForms.</li></ul>
<i>continued on next page</i>	

Continued from previous page	
Standard	Role in XForms
DOM2	<ul style="list-style-type: none"> <li>• Cross-platform API for eventing.</li> <li>• Used to attach <i>behavior</i>.</li> <li>• Enables hosting XForms processors in a browser.</li> </ul>
XML Events	<ul style="list-style-type: none"> <li>• XML access to DOM2 Events.</li> <li>• Defines common XML markup for eventing.</li> <li>• Used to attach event handlers in XForms documents.</li> <li>• Declarative authoring in place of scripting.</li> </ul>

Table 2: XML standards at a glance.

## XForms User Interface Controls

Control	Description
<input>	Generic input control.
<secret>	Password entry
<textarea>	Multi-line text entry
<select>	Select from a set
<select1>	Exclusive select from set
<range>	Pick from range of values
<upload>	Upload data
<trigger>	Activate command
<submit>	Trigger submission

Table 3: XForms controls at a glance.

## XForms User Interface

Construct	Purpose
<b>&lt;group&gt;</b>	<ul style="list-style-type: none"> <li>• Group related controls to ease refactoring.</li> <li>• Enable structured navigation.</li> <li>• Factor common parts of binding expressions.</li> </ul>
<b>&lt;switch&gt;</b>	<ul style="list-style-type: none"> <li>• Enable conditional user interfaces.</li> <li>• Enable interaction-based switching.</li> <li>• Create user interface wizards and multi-page tab dialogs.</li> <li>• Progressively reveal complex user interfaces.</li> </ul>
<b>&lt;repeat&gt;</b>	<ul style="list-style-type: none"> <li>• Iterate over collections.</li> <li>• Create user interfaces that grow or shrink as needed.</li> <li>• Enable creation and maintenance of hierarchical content.</li> </ul>

Table 4: XForms user interface at a glance.

## XForms Model Properties

Property	Description
<i>relevant</i>	<ul style="list-style-type: none"> <li>• Model-based switching for conditional interfaces.</li> <li>• Dynamically revealing appropriate portions of an interface.</li> <li>• enabling controls based on state of instance data.</li> </ul>
<i>continued on next page</i>	

<i>Continued from previous page</i>	
<b>Property</b>	<b>Description</b>
<i>required</i>	<ul style="list-style-type: none"> <li>• Conditionally make fields <i>required</i>.</li> </ul>
<i>readonly</i>	<ul style="list-style-type: none"> <li>• Conditional editing of data.</li> <li>• Dynamic user interaction based on instance values.</li> </ul>
<i>constraint</i>	<ul style="list-style-type: none"> <li>• Refine static schema constraints.</li> <li>• Constrain values based on previous user interaction.</li> <li>• Express cardinality constraints on node-sets.</li> </ul>
<i>calculate</i>	<ul style="list-style-type: none"> <li>• Computational dependency among instance nodes.</li> <li>• Automatically compute fields based on user input.</li> <li>• Enable spread-sheet like functionality within XForms.</li> </ul>
<i>type</i>	<ul style="list-style-type: none"> <li>• Extend schema type definitions.</li> <li>• Useful when schema cannot be modified.</li> </ul>
<i>p3ptype</i>	<ul style="list-style-type: none"> <li>• Hold P3P type information.</li> <li>• Enables users manage their personal data.</li> </ul>

Table 5: XForms model properties at a glance.

## XForms Functions

<b>Function</b>	<b>Arguments</b>	<b>Returns</b>	<b>Description</b>
<i>avg</i>	node-set	number	Average
<i>boolean-from-string</i>	string	boolean	Type conversion
<i>continued on next page</i>			

<i>Continued from previous page</i>			
<b>Function Arguments</b>	<b>Returns</b>	<b>Description</b>	
<i>count-non-empty</i>	node-set	number	count non-empty
<i>days-from-date</i>	string	number	Days in epoch
<i>if</i>	boolean, string, string	string	conditional
<i>index</i>	string	number	Repeat index
<i>instance</i>	string	node-set	Locate instance
<i>max</i>	node-set	number	Maximum
<i>min</i>	node-set	number	Minimum
<i>months</i>	string	number	Months in period
<i>now</i>		string	Current time
<i>property</i>	string	string	Feature value
<i>seconds</i>	string	number	Seconds in period
<i>seconds-from-dateTime</i>	string	number	Seconds in epoch

Table 6: XForms functions at a glance.

## XForms Actions

<b>Action</b>	<b>Description</b>
<b>&lt;setfocus&gt;</b>	Move focus
<b>&lt;setvalue&gt;</b>	Assign value
<b>&lt;load&gt;</b>	Load URI
<b>&lt;send&gt;</b>	Initiate submission
<b>&lt;reset&gt;</b>	Clear form
<b>&lt;message&gt;</b>	Display message
<b>&lt;action&gt;</b>	Group handlers
<b>&lt;dispatch&gt;</b>	Send event
<b>&lt;rebuild&gt;</b>	Rebuild dependencies
<b>&lt;recalculate&gt;</b>	Recompute values
<b>&lt;revalidate&gt;</b>	Revalidate all values
<b>&lt;refresh&gt;</b>	Update user interface
<b>&lt;insert&gt;</b>	Insert node
<b>&lt;delete&gt;</b>	Delete node
<b>&lt;setindex&gt;</b>	Scroll <b>&lt;repeat&gt;</b>

Table 7: XForms actions at a glance.

## XForms Events

Event	Cancel	Bubbles	Target
Initialization Events			
xforms-model-construct	N	Y	⟨model⟩
xforms-model-construct-done	N	Y	⟨model⟩
xforms-ready	N	Y	⟨model⟩
xforms-model-destruct	N	N	⟨model⟩
Processing Events			
xforms-rebuild	Y	Y	⟨model⟩
xforms-recalculate	Y	Y	⟨model⟩
xforms-revalidate	Y	Y	⟨model⟩
xforms-refresh	Y	Y	⟨model⟩
Interaction Events			
xforms-previous	Y	N	⟨control⟩
xforms-next	Y	N	⟨control⟩
xforms-focus	Y	N	⟨control⟩
xforms-help	Y	Y	⟨control⟩
xforms-hint	Y	Y	⟨control⟩
xforms-reset	Y	Y	⟨model⟩
xforms-submit	Y	Y	⟨submission⟩
DOMActivate	Y	Y	⟨control⟩
Notification Events			
DOMFocusIn	N	Y	⟨control⟩
DOMFocusOut	N	Y	⟨control⟩
xforms-value-changing	N	Y	⟨control⟩
xforms-value-changed	N	Y	⟨control⟩
xforms-select	N	Y	⟨item⟩ or ⟨case⟩
xforms-deselect	N	Y	⟨item⟩ or ⟨case⟩
xforms-scroll-first	N	Y	⟨repeat⟩
xforms-scroll-last	N	Y	⟨repeat⟩
xforms-insert	N	Y	⟨instance⟩
xforms-delete	N	Y	⟨instance⟩
xforms-valid	N	Y	⟨control⟩
xforms-invalid	N	Y	⟨control⟩
xforms-in-range	N	Y	⟨control⟩
xforms-out-of-range	N	Y	⟨control⟩
xforms-readonly	N	Y	⟨control⟩
xforms-readwrite	N	Y	⟨control⟩
xforms-required	N	Y	⟨control⟩
xforms-optional	N	Y	⟨control⟩
xforms-enabled	N	Y	⟨control⟩

*continued on next page*

<i>continued from previous page</i>			
<b>Event</b>	<b>Cancel</b>	<b>Bubbles</b>	<b>Target</b>
xforms-disabled	N	Y	<control>
xforms-submit-done	N	Y	<submission>
xforms-submit-error	N	Y	<model>
Error Notifications			
xforms-binding-exception	N	Y	<bind>
xforms-link-exception	N	Y	<model>
xforms-link-error	N	Y	<model>
xforms-compute-exception	N	Y	<model>

Table 8: XForms events at a glance.

## Web Services

<b>XForms</b>	<b>Web Services</b>
<b>model</b>	<ul style="list-style-type: none"> <li>• Collect data as XML.</li> <li>• Serialize request as XML.</li> <li>• Receive response as structured XML.</li> </ul>
<b>Bind</b>	<ul style="list-style-type: none"> <li>• Connect distributed services.</li> <li>• Connect user interfaces to actual data.</li> <li>• Enable aggregation of information services.</li> </ul>
<b>UI</b>	<ul style="list-style-type: none"> <li>• Connect users to their information.</li> <li>• Enable ubiquitous information access.</li> <li>• Enable delivery to multiple devices.</li> <li>• Enable multimodal, mobile Web services.</li> </ul>
<i>continued on next page</i>	

<i>Continued from previous page</i>	
<b>XForms</b>	<b>Web Services</b>
<b>submit</b>	<ul style="list-style-type: none"> <li>• Serialize data as valid SOAP messages.</li> <li>• Smart forms enable automatic data import and export.</li> </ul>

Table 9: Web services at a glance.

## Multimodal Interaction

<b>XForms</b>	<b>Multimodal Interaction</b>
<b>model</b>	<ul style="list-style-type: none"> <li>• Hold interaction state.</li> <li>• Support multi-device access.</li> <li>• Enable synchronized views.</li> </ul>
<b>Bind</b>	<ul style="list-style-type: none"> <li>• Connect multiple interaction modalities.</li> <li>• Connect user interfaces to data.</li> </ul>
<b>UI</b>	<ul style="list-style-type: none"> <li>• Enable late binding of user interaction.</li> <li>• Enable ubiquitous information access.</li> <li>• Enable delivery to multiple devices.</li> <li>• Enable multimodal, mobile user experience.</li> </ul>

Table 10: XForms and multimodal interaction.

## Accessibility

XForms	Accessibility
Model	<ul style="list-style-type: none"><li data-bbox="570 478 964 512">• Factors non-presentational aspects.</li><li data-bbox="570 531 1052 564">• Central point for hooking custom behavior.</li></ul>
<i>continued on next page</i>	

<i>Continued from previous page</i>	
<b>XForms</b>	<b>Accessibility</b>
Controls	<ul style="list-style-type: none"> <li>• Designed for accessibility.</li> <li>• Encapsulate all relevant metadata.</li> <li>• Exhibit predictable behavior.</li> <li>• Enable consistent navigation.</li> </ul>
UI	<ul style="list-style-type: none"> <li>• Aggregation constructs encourage intent-based authoring.</li> <li>• Create content that enables refactoring.</li> <li>• Complex interfaces can be progressively revealed.</li> </ul>
Events	<ul style="list-style-type: none"> <li>• Flexible eventing framework.</li> <li>• Can attach custom behaviors.</li> </ul>

Table 11: XForms accessibility at a glance.