



Implementatie van de UIML standaard in
UIML.NET

Ingo Berben

Promotor: dr. Kris Luyten
Co-promotor: Prof. dr. Karin Coninx
Begeleider: Jo Vermeulen

2006 - 2007

Samenvatting

Inhoudsopgave

1	Inleiding	1
2	UIDL	1
2.1	Inleiding	1
2.2	HLUIDL	1
2.3	XForms	2
2.4	GladeXML	3
2.5	XAML	3
3	UIML - User Interface Markup Language	3
3.1	Inleiding	3
3.2	Doel	3
3.3	Elementen	4
3.3.1	Templates	4
3.3.2	Interface	4
3.3.3	Vocubulaire	4
3.3.4	Logic	4
3.4	Architectuur	4
3.5	Voorbeelden	4
4	UIML.NET	4
4.1	Inleiding	4
4.2	De 3 W's	4
5	Condities	4
5.1	Inleiding	4
5.2	Opbouw	4
5.3	Waarom zijn er condities nodig?	4

Lijst van figuren

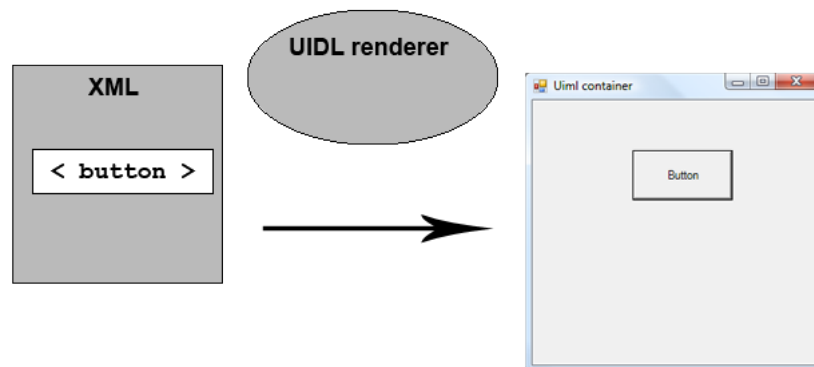
1	XML mapping van <i>button</i> op <i>System.Windows.Forms.Button</i>	1
2	Resultaat van listing 1	2

1 Inleiding

2 UIDL

2.1 Inleiding

Tegenwoordig bestaan er verschillende talen om user interfaces (UI's) te maken. Zo heb je System Windows Forms, Java Swing, GTK, QT, enzovoorts. Maar deze talen hebben allemaal hun eigen code structuren, benamingen, . . . Sinds de opkomst van XML ¹ en mobiele toestellen is de nood gekomen om een meer algemene methode voor het maken van UI's. UIDL of *User Interface Description Language* wordt gebruikt om een UI te beschrijven in bijvoorbeeld XML. Iedere XML tag wordt gemapt op een UI element (zie figuur 1).



Figuur 1: XML mapping van *button* op *System.Windows.Forms.Button*

2.2 HLUIDL

HLUIDL of *High-level user interface description language* gaat nog een stap verder. Zoals de naam het al zegt is dit een taal op een hoger niveau dan UIDL. Met hoger niveau wordt bedoeld dat deze taal nog abstracter is dan

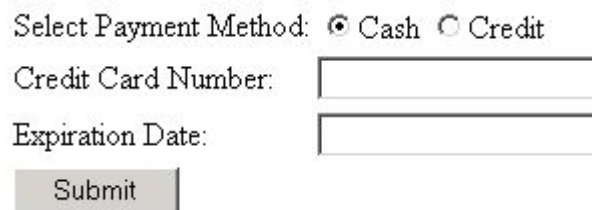
¹eXtensible Markup Language

UIDL. Via één document worden widgets gemaakt op verschillende talen (bv System.Windows.Forms, Gtk, Qt). Dit hangt af van de parameters die werden meegegeven. Deze extra laag zorgt ervoor dat HLUIDL platform-onafhankelijk en toolkit onafhankelijk wordt. Maar hier blijft het niet bij. Met de kennis van enkel de syntax van een HLUIDL kan je programma's ontwikkelen die voldoen aan volgende doelen:

- Overdraagbaar
- Herbruikbaar
- Device-onafhankelijke rendering
- Uitbreidbaar
- Flexible
- Automatische generatie van UI code

2.3 XForms

XForms is een XML formaat dat origineel bedoeld was om om HTML ² en XHTML ³ te vervangen. Het kan echter ook op een standalone manier gebruik worden om UI's te beschrijven. In tegenstelling tot HTML hebben de ontwerpers van XForms zich gebaseerd op de Model-View-Controller aanpak. De data wordt gescheiden van de opmaak van de widgets. Door deze scheiding wordt het mogelijk om formulieren gemaakt met XForms te hergebruiken omdat deze niet afhankelijk zijn van de pagina waarop ze gebruikt worden. Bovendien is XForms niet langer restrictief tot XHTML, het kan geïntegreerd worden in iedere markup language. Een voorbeeld van XForms is te vinden in listing 1, het resultaat in figuur 2.



Select Payment Method: Cash Credit

Credit Card Number:

Expiration Date:

Figuur 2: Resultaat van listing 1

²HyperText Markup Language

³eXtensible HyperText Markup Language

Listing 1: XForms Card Voorbeeld

```
1 <xforms:model>
2   <xforms:instance>
3     <ecommerce xmlns="">
4       <method/>
5       <number/>
6       <expiry/>
7     </ecommerce>
8   </xforms:instance>
9   <xforms:submission action="http://example.com/submit"
10  method="post" id="submit" includenamespaces="" />
11 </xforms:model>
```

2.4 GladeXML

2.5 XAML

3 UIML - User Interface Markup Language

3.1 Inleiding

UI's of gebruikersinterface worden meer en meer ontwikkeld door de opkomst van nieuwere soft- en hardware technologieën. Dit is goed nieuws voor de gebruiker, deze heeft nu meer keuze. Voor de ontwikkelaar van een UI is dit echter geen lachertje, want er doen zich immers verschillende problemen voor. Hij moet verschillende talen aanleren, enkele voorbeelden zijn Wireless Markup Language (WML) dat gebruikt wordt voor mobiele telefoons en aanverwante apparaten die gebruikmaken van het WAP⁴ protocol, C++ met specifieke structuren voor PalmOS of WindowsCE, enzovoorts. Al deze talen hebben het probleem dat ze evolueren over tijd. Functienamen veranderen, sommige worden 'deprecated' en verdwijnen en nieuwe functies en klassen komen erbij. Het is moeilijk voor de ontwikkelaar om deze veranderingen continue te blijven volgen. Sommige talen veranderen zo snel, dat het niet mogelijk is om de code telkens opnieuw aan te passen aan de nieuwere versie. Een volgend probleem dat zich stelt, heeft te maken met de verschillende stukken code die de ontwikkelaar heeft gemaakt voor de verschillende talen. Deze moeten immers consistent blijven.

3.2 Doel

UIML of *User Interface Description Language* is een platformonafhankelijke taal die toelaat om UI's te beschrijven in XML. Het kan gezien worden

⁴Wireless Application Protocol

als een meta-taal. Specieke tags gelijk <window> of <button> zijn niet aanwezig, maar er wordt gebruik gemaakt van generische tags gelijk <part> en <property>.

3.3 Elementen

3.3.1 Templates

3.3.2 Interface

3.3.3 Vocabulaire

3.3.4 Logic

3.4 Architectuur

3.5 Voorbeelden

4 UIML.NET

4.1 Inleiding

4.2 De 3 W's

5 Conditie

5.1 Inleiding

5.2 Opbouw

5.3 Waarom zijn er condities nodig?