



The new EverQuest UI
About SIDL

Last Updated: June 9, 2002



CONTENTS

CONTENTS.....2

Overview.....3

Tags! <tags>, </tags>, <tags/>!.....3

UI Hierarchy.....3

Types.....4

MinOccurs, MaxOccurs.....6

The List of ElementTypes.....7

The Extensive List of Elements.....9



Overview

The new EverQuest UI uses the XML format for all of its data files. It is a standard practice with using XML to create one file that serves as the definition (or schema) for how all the remaining XML files can be written (<http://www.w3.org/TR/NOTE-xml-schema-req>). In this case, our definition file is, indeed, SIDL.xml.

SIDL stands for **S**UITE Interface **D**efinition **L**anguage. The acronym, SUITE, refers to the name of our UI engine, while "Interface Definition Language" hammers in that this file is the master schema file.

This document lays out the contents of SIDL and explains what each property definition is for. Don't worry if you can't get a full-grasp of everything the first read through -- you will probably use this more as reference material rather than reading material, and SIDL's role will just start to make more and more sense as you continue to make changes to the UI.

Feel free to open up SIDL.xml to see how the file and this document correspond. But always remember, you cannot edit it!

Tags! <tags>, </tags>, <tags/>!

When it comes down to it, the most basic thing a schema does is specify which tags all of the other XML files can use. For example, the line in SIDL.xml:

```
<element name="Text" type="string" />
```

can translate to the following line (taken from EQUI_CharacterCreate.xml)

```
<Text>Welcome to EverQuest</Text>
```

This also lets us specify default values for our UI objects, so when you have in SIDL.xml:

```
<element name="TextOffsetX" type="int">
  <default>0</default>
</element>
```

this means that if you do not specify the tag for this item, it will automatically be included with the value of 0.

UI Hierarchy

Our schema also lets us specify a hierarchy of UI objects. For example, SIDL.xml indicates that a Label is a ScreenPiece. This means that the Label, along with having all of its text info, has all the data of a ScreenPiece as well -- this specifies how and where the object actually appears on the screen.

SIDL accomplishes this by employing three tags of its own:

- **ElementType**
- **element**
- **superType**

At this time, you may want to browse SIDL.xml to see these tags and where they are used.

ElementType - This defines a UI piece. Some examples would be an EditBox, a Slider, a Texture item, and an RGB item. ElementTypes are always a collection of elements, so in essence an ElementType tag is just a top-level name that ties data elements together.

element - This defines a UI property. Height, widths, positions, text values, etc.



About SIDL

superType - This defines that a piece is a specific type of some more generic category. For example, ScreenPieces are anything that can be put on the screen. Controls are anything that can accept user input. A combobox is a specific type of a control that provides a pulldown list.

Any ElementType that has a superType also contains all of its superType's elements.

Types

Every SIDL element also specifies an associated type. An item's type defines what value its tag can accept. Some examples at the end of this section will help clear this up.

To start with, there are three basic types:

- integer
- string
- boolean

Integers specify that the property is a number.

Strings specify that the property is a text value.

Booleans specify that the property can be set to true or false (useful for toggling a feature on or off).

Aside from basic types, any ElementType can also be set here. For example, the FrameTemplate (an ElementType) has a bunch of 2DAnimations (another ElementType) for its border elements. This way, you can create windows with frames that animate.

The final note here is that some types are attached with the text, ":item". For example, you will see "Ui2DAnimation:item" as opposed to just "Ui2DAnimation". This indicates that you create the element type first and then refer to it later by its item name. Otherwise, you would just define the child element type within the parent ElementType (see examples below).

The following examples serve as tutorial purposes only and do not necessarily exist anywhere in the UI files provided:

integer in SIDL

```
<element name="Width" type="int" />
```

can translate to

```
<Width>100</Width>
```

string in SIDL

```
<element name="Text" type="string" />
```

can translate to

```
<Text>Hello</Text>
```

boolean in SIDL

```
<element name="AlignCenter" type="boolean" />
```

can translate to

```
<AlignCenter>true</AlignCenter>
```

ElementType in SIDL



About SIDL

```
<ElementType name="Point">
  <element name="X" type="int" />
  <element name="Y" type="int" />
</ElementType>

<element name="Location" type="Point" />
```

can translate to

```
<Location>
  <X>50</X>
  <Y>100</Y>
</Location>
```

ElementType: item in SIDL

```
<ElementType name="RGB">
  <superType type="Class" />
  <element name="Alpha" type="int" />
  <element name="R" type="int" />
  <element name="G" type="int" />
  <element name="B" type="int" />
</ElementType>

<element name="TextColor" type="RGB:item">

<RGB item="ColorWhite">
  <Alpha>255</Alpha>
  <R>255</R>
  <G>255</G>
  <B>255</B>
</RGB>
```

can translate to

```
<TextColor>ColorWhite</TextColor>
```



MinOccurs, MaxOccurs

In a few items, you will see that the element tag contains minOccurs/maxOccurs values. This indicates that the particular element can occur more than once, so in effect you can have a list of them.

Let us look at (a shortened) definition for a pulldown list:

```
<ElementType name="Combobox">  
  <element name = "Choices" type = "string" minOccurs = "0" maxOccurs = "*" />  
</ElementType>
```

With this, a fair implementation would be:

```
<Combobox>  
  <Choices>This tutorial is working for me</Choices>  
  <Choices>This tutorial is NOT working for me</Choices>  
  <Choices>Please leave me alone</Choices>  
</Combobox>
```

minOccurs specifies the minimum times you have to declare this element, and maxOccurs specifies the maximum number of times. The "*" here means there is no maximum.

In the EverQuest UI, you will notice that we always declare these two values as 0 and *, respectively, for maximum flexibility.



The List of ElementTypes

The following items are defined in SIDL and can be used in all EverQuest UI XML files.

```
Point
Size
Class
├─ RGB
├─ TextureInfo
├─ Frame
├─ Ui2DAnimation
├─ ButtonDrawTemplate
├─ GaugeDrawTemplate
├─ SpellGemDrawTemplate
├─ FrameTemplate
├─ ScrollbarDrawTemplate
├─ WindowDrawTemplate
├─ SliderDrawTemplate
├─ ScreenPiece
│   ├── StaticScreenPiece
│   ├── StaticAnimation
│   ├── StaticText
│   ├── StaticFrame
│   ├── StaticHeader
│   └─ Control
├─ ListboxColumn
│   ├── Listbox
│   ├── Button
│   ├── Gauge
│   ├── SpellGem
│   ├── InvSlot
│   ├── Editbox
│   ├── Slider
│   ├── Label
│   ├── STMLbox
│   ├── Combobox
│   ├── Page
│   ├── TabBox
│   └─ Screen
└─ SuiteDefaults
```



ElementType	Description
Button	A button or a checkbox
ButtonDrawTemplate	The art for a Button
Class	The most basic SUITE piece, contains an unique item name that the UI system can then refer to
ComboBox	A pulldown list
Control	Any UI piece on the screen that accepts user interaction (via keypresses and mouse clicks)
Editbox	An area you can type in
Frame	Data for a single frame within an animation
FrameTemplate	The border art of a window
GaugeDrawTemplate	Art for an EverQuest gauge
Gauge	An EverQuest gauge - the bar for showing health, Mana, experience points, etc.
InvSlot	An item slot for an EverQuest inventory screen
Label	A read-only text area
Listbox	A table, much like this list of descriptions you're looking at now
ListboxColumn	One column in the Listbox
Page	A UI screen that is contained within a tab box and is associated with one of the tabs
Point	An X, Y coordinate
RGB	R,G,B, and alpha information
Screen	A top-level window, which contains a whole bunch of UI pieces within it
ScreenPiece	Any SUITE piece that is meant to be drawn on the screen
ScrollbarDrawTemplate	The art for a scrollbar within a window
Size	Width and height information
Slider	A horizontal bar with a draggable piece used to set some value within a range
SliderDrawTemplate	The art for a slider
SpellGem	An EverQuest spellgem (the tiny icons you click on to cast a spell)
SpellGemDrawTemplate	The art for an EverQuest spellgem's background
StaticAnimation	An animation that you just plaster up on the screen -- it doesn't accept user input
StaticFrame	A window frame that you just plaster up on the screen -- it doesn't accept user input
StaticHeader	Some header text with art that you just plaster up on the screen -- it doesn't accept user input
StaticScreenPiece	Any piece that is drawn on the screen but doesn't accept user input
StaticText	Some text that you just plaster up on the screen -- it doesn't accept user input
STMLbox	A read-only text area that displays STML (a subset of HTML). This allows for color, font size variations, etc.
SuiteDefaults	A collection of UI components that are global to the whole application (mostly mouse cursors)
TabBox	A frame that has tabs along the top -- when you click on a tab, it will display an associated Page
TextureInfo	A UI pieces that has a filename for it's item name, plus the size of the file, used for loading in art from files.
Ui2DAnimation	A collection of Frames that make up an animation
WindowDrawTemplate	The art for a window



The Extensive List of Elements

The following section looks at all ElementTypes in full detail. These items are alphabetized so you can use this as a quick reference.

Button

Parameter	Type	Default	Description
Style_Checkbox	bool		This button acts as a checkbox (does not pop back up on mouse release)
RadioGroup	string		This button is part of a radio group.
Text	string		Text
ButtonDrawTemplate	ButtonDrawTemplate		Template that defines this button's art
SoundPressed	string		Sound to play on button press (currently not implemented)
SoundUp	string		Sound to play on button release (currently not implemented)
SoundFlyby	string		Sound to play on button hover (currently not implemented)
DecalOffset	Point		Offset for this button's decal, if it exists (see ButtonDrawTemplate)
DecalSize	Size		Size to fit this button's decal in, if it exists (see ButtonDrawTemplate)

ButtonDrawTemplate

Parameter	Type	Default	Description
Normal	Ui2DAnimation:itm		Image for a button just sitting around
Pressed	Ui2DAnimation:itm		Image for a button under the oppression of the mouse click
Flyby	Ui2DAnimation:itm		Image for a button with the mouse hovering over it
Disabled	Ui2DAnimation:itm		Image for a button that has been disabled
PressedFlyby	Ui2DAnimation:itm		Image for a depressed button with the mouse hovering over it (used by Checkbox buttons)
NormalDecal	Ui2DAnimation:itm		Image that appears on top of a button
PressedDecal	Ui2DAnimation:itm		Image that appears on top of a pressed button (defaults to NormalDecal if not set)
FlybyDecal	Ui2DAnimation:itm		Image that appears on top of a highlighted button (defaults to NormalDecal if not set)
DisabledDecal	Ui2DAnimation:itm		Image that appears on top of a disabled button (defaults to NormalDecal if not set)
PressedFlybyDecal	Ui2DAnimation:itm		Image that appears on top of a disabled and highlighted button (defaults to NormalDecal if not set)

Class

Parameter	Type	Default	Description
item	string		Name that can be used to refer to this SUITE piece

ComboBox

Parameter	Type	Default	Description
Button	ButtonDrawTemplate:item		Pull-down list button
ListHeight	int		Max height of this window when it is being pulled down
Choices	string[]		String choices to go into this combobox's pulldown list

Control

Parameter	Type	Default	Description
Style_VScroll	boolean	false	↑↓ This control is vertically scrollable
Style_HScroll	boolean	false	←→ This control is horizontally scrollable
Style_Transparent	boolean	false	You can see through this control
Style_Border	boolean		This widget is surrounded by a border
TooltipReference	string		Help text for this control if the user holds the cursor over the item
DrawTemplate	WindowDrawTemplate:item		Template that defines this window's art

Editbox

Parameter	Type	Default	Description
Style_Multiline	boolean		This editbox can contain multiple lines of text



About SIDL

Frame

Parameter	Type	Default	Description
Texture	string		Image texture this frame's image is contained in
Location	Point		Location of this frame's image in the texture
Size	Size		Size of this frame's image
Hotspot	Point		An important reference point. For example, it is used to keep an animation centered if every frame in it is a variable size. This value is also used in cursors.
Duration	int	1000	Milliseconds of life for this frame in an animation cycle
Shading	RGB[]		A layer of shade to apply to the texture
Specular	RGB[]		A layer of specular gloss to apply to the texture

FrameTemplate

Parameter	Type	Default	Description
TopLeft	Ui2DAnimation:itm		Image for this frame's top-left corner
Top	Ui2DAnimation:itm		Image for this frame's top border
TopRight	Ui2DAnimation:itm		Image for this frame's top-right corner
RightTop	Ui2DAnimation:itm		Image for this frame's right-top border
Right	Ui2DAnimation:itm		Image for this frame's right border
RightBottom	Ui2DAnimation:itm		Image for this frame's right-bottom border
BottomRight	Ui2DAnimation:itm		Image for this frame's bottom-right corner
Bottom	Ui2DAnimation:itm		Image for this frame's bottom border
BottomLeft	Ui2DAnimation:itm		Image for this frame's bottom-left corner
LeftTop	Ui2DAnimation:itm		Image for this frame's left-top border
Left	Ui2DAnimation:itm		Image for this frame's left border
LeftBottom	Ui2DAnimation:itm		Image for this frame's left-bottom border
Middle	Ui2DAnimation:itm		Image for this frame's center area
OverlapLeft	int	0	Pixels to let the middle overlap over the left frame
OverlapTop	int	0	Pixels to let the middle overlap over the top frame
OverlapRight	int	0	Pixels to let the middle overlap over the right frame
OverlapBottom	int	0	Pixels to let the middle overlap over the bottom frame

Gauge

Parameter	Type	Default	Description
GaugeDrawTemplate	GaugeDrawTemplate		Template that defines the art for this gauge
EQType	int		Defines what EQ value the gauge displays (HP, Mana, etc.)
FillTint	RGB		Color of the bar that fills in.
DrawLinesFill	boolean		Whether or not to draw the lines filling in
LinesFillTint	RGB		Color of the lines when filling in
TextOffsetX	int	0	X-offset for the text associated with this gauge
TextOffsetY	int	0	Y-offset for the text associated with this gauge
GaugeOffsetX	int	0	X-offset for the gauge itself
GaugeOffsetY	int	16	Y-offset for the gauge itself

GaugeDrawTemplate

Parameter	Type	Default	Description
Background	Ui2DAnimation:itm		Background image for the gauge
Fill	Ui2DAnimation:itm		The bar that fills in on the gauge
Lines	Ui2DAnimation:itm		The hash marks and hi-lites
LinesFill	Ui2DAnimation:itm		The filled in version of the lines

Produced by the EverQuest Live Department



Contents are Proprietary to Sony Online Entertainment and Verant Interactive Inc

About SIDL

EndCapRight	Ui2DAnimation:itm		Right end cap piece
EndCapLeft	Ui2DAnimation:itm		Left end cap piece

InvSlot

Parameter	Type	Default	Description
EQType	int		Inventory slot type (user, trading, merchant, bank, etc.)
Background	Ui2DAnimation:item		Background image for this inventory slot, when empty
ItemOffsetX	int	0	X-offset to apply to a contained item
ItemOffsetY	boolean	0	Y-offset to apply to a contained item

Label

Parameter	Type	Default	Description
NoWrap	boolean	false	Don't allow this label's text to wrap
AlignCenter	boolean	false	Center the text. By default, the text is left-justified
AlignRight	boolean	false	Right-justify the text. If AlignCenter is true, this value is ignored.

Listbox

Parameter	Type	Default	Description
Columns	ListboxColumn[]		Columns that make up this listbox
OwnerDraw	boolean		This object draws its columns itself

ListboxColumn

Parameter	Type	Default	Description
Header	FrameTemplate:item		A special frame for the heading of this list's column
Heading	string		The text for the heading of the list's column
Width	int		Width of this column
Sortable	boolean		Specifies if this list be sortable
DataType	string		Not used

Page

Parameter	Type	Default	Description
TabText	string		Text to attach to the tab that opens this page
TabIcon	Ui2DAnimation:item		Icon to attach to the tab that opens this page. The icon is always drawn left-justified. If any text also exists, it will be drawn to the right of the icon.
Pieces	ScreenPiece:item[]		Children items

Point

Parameter	Type	Default	Description
X	int	0	X-coordinate
Y	int	0	Y-coordinate

RGB

Parameter	Type	Default	Description
Alpha	int	255	Transparency value, 0 - 255
R	int	0	Red value, 0 - 255
G	int	0	Green value, 0 - 255
B	int	0	Blue value, 0 - 255

Screen

Parameter	Type	Default	Description
Style Titlebar	boolean		True if this window has a titlebar
Style Closebox	boolean		True if this window has a close button
Style Minimizebox	boolean		True if this window has a minimize button
Style Sizeable	boolean		True if this window can be resized
Pieces	ScreenPiece:item[]		Children items



ScreenPiece

Parameter	Type	Default	Description
ScreenID	string		An identifier that is unique on the scope of all items being created within a parent control. All parent controls can access any of their children by this ID.
Font	int	3	Font style from 0 - 6 (0 being small, 6 being large)
RelativePosition	boolean	true	Draw this @ (x, y) relative from its parent window's topleft corner
Location	Point		(x, y) coordinates of top-left corner. Ignored if RelativePosition AND Autostretch is true.
Size	Size		(w, h) of item. Ignored if RelativePosition AND Autostretch is true.
AutoStretch	boolean	false	Stretch this window to the borders of its parent. If true, this window will be resized when his parent is resized. If not, all anchor variables (below) are ignored.
TopAnchorToTop	boolean	true	If true, keep the top side of this window a fixed offset away from its parent's top. Else, keep it a fixed offset away from its parent's bottom.
BottomAnchorToTop	boolean	true	If true, keep the bottom side of this window a fixed offset away from its parent's top. Else, keep it a fixed offset away from its parent's bottom.
LeftAnchorToLeft	boolean	true	If true, keep the left side of this window a fixed offset away from its parent's left. Else, keep it a fixed offset away from its parent's right.
RightAnchorToLeft	boolean	true	If true, keep the right side of this window a fixed offset away from its parent's left. Else, keep it a fixed offset away from its parent's right.
TopAnchorOffset	int	0	Used by TopAnchorToTop
BottomAnchorOffset	int	0	Used by BottomAnchorToTop
LeftAnchorOffset	int	0	Used by LeftAnchorToLeft
RightAnchorOffset	int	0	Used by RightAnchorToLeft
Text	string		Main text for this item.
TextColor	RGB		Color of the main text

ScrollbarDrawTemplate

Parameter	Type	Default	Description
UpButton	ButtonDrawTemplate		Template that defines the art for this scrollbar's up button
DownButton	ButtonDrawTemplate		Template that defines the art for this scrollbar's down button
Thumb	FrameTemplate		Template that defines the art for this scrollbar's scroll box
MiddleTextureInfo	string		Filename for an image file whose entirety is the pattern for for the scroll area of the scrollbar
MiddleTint	RGB		Tint to apply to the scroll area texture

Size

Parameter	Type	Default	Description
CX	int		Width value
CY	int		Height value

Slider

Parameter	Type	Default	Description
SliderArt	SliderDrawTemplate:item		Template that defines the art for this slider

SliderDrawTemplate

Parameter	Type	Default	Description
Thumb	ButtonDrawTemplate		Template that defines the art for this slider's thumb (the piece that is scrolled to change the slider's value)
Background	Ui2DAnimation:item		Background image for the slider (that the thumb scrolls along)
EndCapRight	Ui2DAnimation:item		Right end cap piece
EndCapLeft	Ui2DAnimation:item		Left end cap piece

spellGem

Parameter	Type	Default	Description
SpellGemDrawTemplate	SpellGemDrawTemplate		Template that defines the art for this spellgem
SpellIconOffsetX	int	0	Offset of the icon onto this spellgem's background
SpellIconOffsetY	int	0	Offset of the icon onto this spellgem's background

spellGemDrawTemplate

Parameter	Type	Default	Description
-----------	------	---------	-------------

Produced by the EverQuest Live Department



About SIDL

Holder	Ui2DAnimation:item		The image for the spell gem container (when empty)
Background	Ui2DAnimation:item		The background for a spell gem icon
Highlight	Ui2DAnimation:item		A shine to put on the spellgem on mouseover

StaticAnimation

Parameter	Type	Default	Description
Animation	Ui2DAnimation:item		An animation to draw

StaticFrame

Parameter	Type	Default	Description
FrameTemplate	FrameTemplate:item		A frame to draw

StaticHeader

Parameter	Type	Default	Description
FrameTemplate	FrameTemplate:item		A header frame to draw that has some text in it (useful for a menu)
TextReference	string		The text to draw
TextColor	RGB		The text color to draw with

StaticScreenPiece

Parameter	Type	Default	Description
AutoDraw	boolean	true	Have this piece automatically appear (as opposed to having the code programmatically choose to draw it)

StaticText

Parameter	Type	Default	Description
NoWrap	boolean	false	Don't allow this text to wrap if squished
AlignCenter	boolean	false	Center this text
AlignRight	boolean	false	Right justify this text

STMLbox

Parameter	Type	Default	Description
-----------	------	---------	-------------

SuiteDefaults

Parameter	Type	Default	Description
DefaultWindowDrawTemplate	WindowDrawTemplate:item		Store the game's standard window
CursorDefault	Ui2DAnimation:item		Default cursor
CursorResizeNS	Ui2DAnimation:item		Default 渺响 cursor
CursorResizeEW	Ui2DAnimation:item		Default 潞臧 cursor
CursorResizeNESW	Ui2DAnimation:item		Default ㄵ cursor
CursorResizeNWSE	Ui2DAnimation:item		Default ㄴ cursor
CursorDrag	Ui2DAnimation:item		Default cursor when dragging an item

TabBox

Parameter	Type	Default	Description
TabBorderTemplate	FrameTemplate:item		The template that defines the art for a tab
PageBorderTemplate	FrameTemplate:item		The template that defines the art for the main window frame where each tab page goes.
Pages	Page:item[]		A collection of pages (each page gets an associated tab).

TextureInfo

Parameter	Type	Default	Description
Size	Size		The size of this image file

Ui2DAnimation

Parameter	Type	Default	Description
Cycle	boolean		Cycle the animation
Grid	boolean		Set this animation to be a "grid" (used for drag items, etc.)

Produced by the EverQuest Live Department



Contents are Proprietary to Sony Online Entertainment and Verant Interactive Inc

About SIDL

Vertical	boolean		Grid is Vertical instead of horizontal (only used when Grid = true)
CellHeight	boolean	0	Height of each cell in the grid (only used when Grid = true)
CellWidth	boolean	0	Width of each cell in the grid (only used when Grid = true)
Frames	Frame[]		Animation frames

WindowDrawTemplate

Parameter	Type	Default	Description
Background	TextureInfo:item		Background image for this window
VSBTemplate	ScrollbarDrawTemplate		Template that defines this window's vertical scrollbar art
HSBTemplate	ScrollbarDrawTemplate		Template that defines this window's horizontal scrollbar art
CloseBox	ButtonDrawTemplate		Template that defines this window's close button art
MinimizeBox	ButtonDrawTemplate		Template that defines this window's minimize button art
TileBox	ButtonDrawTemplate		Template that defines art to tile all along this window's background
Border	FrameTemplate		Template that defines this window's border art
Titlebar	FrameTemplate		Template that defines this window's title art

