Basic

```xml
<?xml version='1.0' ?>
<toolbar>
<ImageButton id="b1" src="some1.gif" />                     image button
<ImageTextButton id="b2" src="some2.gif">                   image button with text
  Text part here
</ImageTextButton>
<LabelButton id="label1">                                   just a label
  Label here
</LabelButton>
<divider id="d1" />                                         divider
<ImageTextButtonXP id="b3" src="some3.gif">                 XP like image button with
                                                            text
  Text part here
</ImageTextButtonXP>
<TwoStateButton id="b4" src="some4.gif" pressedState="true">  two state button
  Text part here
</TwoStateButton>
<SelectButton id="b5" valueList="1,2,3" displayList="One,Two,Three"
/>                                                          selectbox
</toolbar>
```

Full

`<?xml version="1.0" encoding="iso-8859-1"?>`	encoding can be important
`<toolbar toolbarAlign="left" absolutePosition="yes" left="100" top="100" name="Demo menu" width="600px" height="20px" globalTextCss="someCss2" globalCss="someCss4">`	most probably you will never need one from those attributes
`<ImageButton id="b1" src="some1.gif" width="20px" height="20px" mouseover="alter.gif" className="css1" tooltip="some text here" disabled="true" disableImage="alter2.gif"/>`	image button
`<ImageTextButton id="b2" src="some2.gif" width="20px" height="20px" mouseover="alter.gif" className="css1" textClassName="css2" tooltip="some text here" disabled="true" disableImage="alter2.gif">`	image button with text
`  Text part here`	
`</ImageTextButton>`	
`<LabelButton id="label1" width="20px" >`	just a label
`  Label here`	
`</LabelButton>`	
`<divider id="d1" />`	divider
`<ImageTextButtonXP id="b3" src="some3.gif" width="20px" height="20px" mouseover="alter.gif" className="css1" textClassName="css2" tooltip="some text here" disabled="true" disableImage="alter2.gif">`	XP like image button with text
`  Text part here`	
`</ImageTextButtonXP>`	
`<TwoStateButton id="b4" src="some4.gif" pressedState="true" width="20px" height="20px" className="css1" tooltip="some text here" disabled="true" disableImage="alter2.gif">`	two state button
`  Text part here`	
`</TwoStateButton>`	
`<SelectButton id="b5" valueList="1,2,3" displayList="One,Two,Three" width="20px" height="20px" className="css1" disabled="true" />`	selectbox
`</toolbar>`	

ImageButton

tag represent button on toolbar - image button
can contain multiple attributes, but only few of them really necessary, most are optional

className	css class which will be applied to button
disableImage	in disabled state button change its css class , so it possible to define custom disabled effect, if it is not enough, it possible to define one more image which will be used in disabled state
disabled	if attribute exists, button will be rendered in disabled state (can be enabled later, by js API)
height	height of button (in pixels)
id	id of toolbar button must be unique per toolbar
mouseover	optional url, to image, which will be shown instead of default one, when mouse moved over button
src	url to image used as toolbar button
tooltip	tooltip of button
width	width of button (in pixels)

ImageTextButton

tag represent button on toolbar - image with text button - text right to image
can contain multiple attributes, but only few of them really necessary, most are optional
value of tag will be used as text part of button

className	css class which will be applied to button
disableImage	in disabled state button change its css class , so it possible to define custom disabled effect, if it is not enough, it possible to define one more image which will be used in disabled state
disabled	if attribute exists, button will be rendered in disabled state (can be enabled later, by js API)
height	height of button (in pixels)
id	id of toolbar button must be unique per toolbar
mouseover	optional url, to image, which will be shown instead of default one, when mouse moved over button

src	url to image used as toolbar button	
textClassName	css class which will be applied to text part of button	
tooltip	tooltip of button	
width	width of button (in pixels)	

ImageTextButtonXP tag represent button on toolbar - image with text button - text on the bottom of button
can contain multiple attributes, but only few of them really necessary, most are optional
value of rag will be used as text part of button

className	css class which will be applied to button
disableImage	in disabled state button change its css class , so it possible to define custom disabled effect, if it is not enough, it possible to define one more image which will be used in disabled state
disabled	if attribute exists, button will be rendered in disabled state (can be enabled later, by js API)
height	height of button (in pixels)
id	id of toolbar button must be unique per toolbar
mouseover	optional url, to image, which will be shown instead of default one, when mouse moved over button
src	url to image used as toolbar button
textClassName	css class which will be applied to text part of button
tooltip	tooltip of button
width	width of button (in pixels)

LabelButton tag represent button on toolbar - label
value of tag will be used as text part of button

id	id of toolbar button must be unique per toolbar
width	width of button (in pixels)

SelectButton	tag represent button on toolbar - selectbox can contain multiple attributes, but only few of them really necessary, most are optional value of tag will be used as text part of button

className	css class which will be applied to button
disabled	if attribute exists, button will be rendered in disabled state (can be enabled later, by js API)
displayList	comma separated list of labels, not HTML allowed
height	height of button (in pixels)
id	id of toolbar button must be unique per toolbar
valueList	comma separated list of values
width	width of button (in pixels)

TwoStateButton	tag represent button on toolbar - two state button can contain multiple attributes, but only few of them really necessary, most are optional value of tag will be used as text part of button

className	css class which will be applied to button
disableImage	in disabled state button change its css class , so it possible to define custom disabled effect, if it is not enough, it possible to define one more image which will be used in disabled state
disabled	if attribute exists, button will be rendered in disabled state (can be enabled later, by js API)
height	height of button (in pixels)
id	id of toolbar button must be unique per toolbar
mouseover	optional url, to image, which will be shown instead of default one, when mouse moved over button
pressedState	allows to set current state of button • true - button pressed • false - button not pressed

	src	url to image used as toolbar button
	textClassName	css class which will be applied to text part of button
	tooltip	tooltip of button
	width	width of button (in pixels)

---

divider toolbar divider

	id	item identificator, not really necessary , but can be used for manipulation by JS API

---

toolbar just top element of XML, can contain a LOT of attributes
but in real life you most probably will not use any of it

	absolutePosition	if this attribute present , toolbar position will be changed to absolute (in 99.9% of cases you will NOT need it)
	globalCss	css which will be applied to toolbar items
	globalTextCss	css class which will be applied to text inside toolbar items
	height	height of toolbar (optional, by default value from js command used)
	left	if absolute position enabled, this attribute can specify left coordinate of toolbar
	name	optional text on toolbar
	toolbarAlign	align of toolbar items, inside toolbar

- right

- left

	top	if absolute position enabled, this attribute can specify top coordinate of toolbar
	width	width of toolbar (optional, by default value from js command used)