

HSPLIT and VSPLIT

HSPLIT or VSPLIT allows to define a container area that is subdivided into two split cells. Between the split cells there is a border. By dragging and dropping the border, you can change the size of the split cells. Each split cell itself is a container that can be used just as normal.

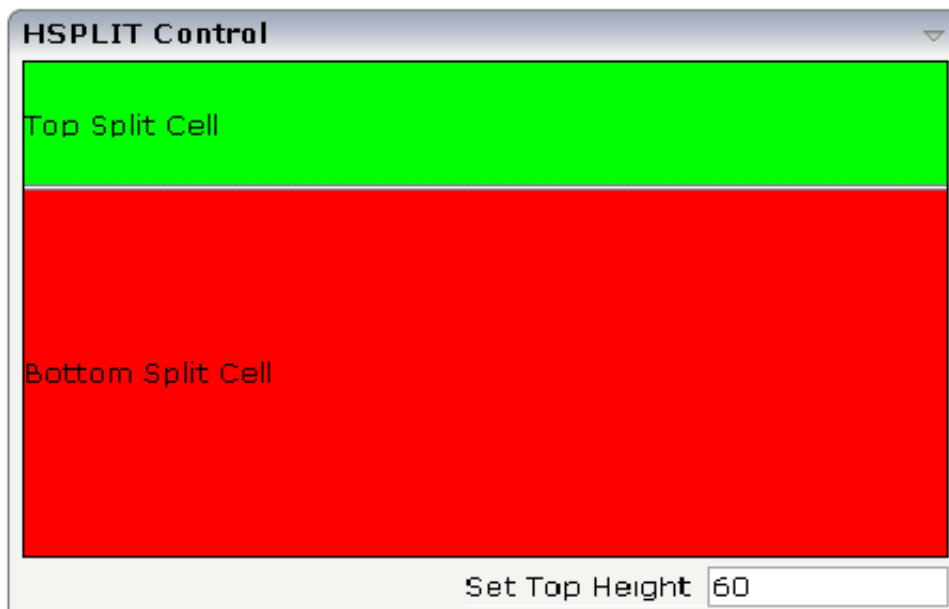
While an HSPLIT control subdivides an area into two split cells by a horizontal line, VSPLIT uses a vertical line.

The following topics are covered below:

- Example for HSPLIT
 - Example for VSPLIT
 - HSPLIT Properties
 - VSPLIT Properties
 - SPLITCELL Properties
 - Defining the Split Size
-

Example for HSPLIT

The following example shows the usage of the HSPLIT control:



The split area is divided into two cells: a green cell and a red cell. In addition, there is a line at the bottom in which you can provide the split factor.

The XML layout definition is:

```
<rowarea name="HSPLIT Control" height="100%">
  <hsplit height="100%" heighttopprop="heighttop" hsplitstyle="border:1 solid #000000">
    <splitcell takefullheight="true" cellstyle="background-color: #00FF00">
      <tr height="100%">
        <label name="Top Split Cell" asplaintext="true">
          </label>
        </tr>
      </splitcell>
      <splitcell takefullheight="true" cellstyle="background-color: #FF0000">
        <tr height="100%">
          <label name="Bottom Split Cell" asplaintext="true">
            </label>
          </tr>
        </splitcell>
      </hsplit>
      <vdist>
      </vdist>
      <itr>
        <hdist width="100%">
        </hdist>
        <label name="Set Top Height" width="100">
        </label>
        <field valueprop="heighttop" width="100" flush="server" validation="[0-9%]+"
          validationuserhint="100, 200, 500, 30%, 50%">
        </field>
      </itr>
    </rowarea>
```

You see that the vertical split area consists of

- one VSPLIT definition, and
- two SPLITCELL definitions.

It is not allowed to have more than two split cells inside one HSPLIT container.

The sizing of the split cells can be done by using a property that is referenced by the HSPLIT property `heighttopprop`. The property must return either a percentage value or a pixel value. When the user changes the size by moving the line between the split cells, then the current new pixel width of the left split cell is written back into the property.

Example for VSPLIT

The VSPLIT control is defined in the same way as the HSPLIT control - but now transferred to vertical dimension. It looks like:



The VSPLIT part of the XML layout definition is:

```
<vsplit height="200" widthleftprop="widthleft" vsplitstyle="border: 1 solid #000000">
  <splitcell takefullheight="true" cellstyle="background-color:#00FF00">
    <itr>
      <label name="Left Split Cell" asplaintext="true">
        </label>
      </itr>
    </splitcell>
    <splitcell takefullheight="true" cellstyle="background-color: #FF0000">
      <itr>
        <label name="Right Split Cell" asplaintext="true">
          </label>
        </itr>
      </splitcell>
    </vsplit>
```

HSPLIT Properties

Basic

<p>height</p>	<p>Height of the control.</p> <p>There are three possibilities to define the height:</p> <p>(A) You do not define a height at all. As consequence the control will be rendered with its default height. If the control is a container control (containing) other controls then the height of the control will follow the height of its content.</p> <p>(B) Pixel sizing: just input a number value (e.g. "20").</p> <p>(C) Percentage sizing: input a percentage value (e.g. "50%"). Pay attention: percentage sizing will only bring up correct results if the parent element of the control properly defines a height this control can reference. If you specify this control to have a height of 50% then the parent element (e.g. an ITR-row) may itself define a height of "100% ". If the parent element does not specify a width then the rendering result may not represent what you expect.</p>	<p>Optional</p>	<p>100</p> <p>150</p> <p>200</p> <p>250</p> <p>300</p> <p>250</p> <p>400</p> <p>50%</p> <p>100%</p>
<p>heighttop</p>	<p>Definition of the initial height of the top split area.</p> <p>The height either is a pixel value ("100") or a percentage value ("50% ").</p> <p>You can also define the height dynamically by your adapter - see documentation for HEIGHTTOPPROP property.</p>	<p>Optional</p>	<p>1</p> <p>2</p> <p>3</p> <p>int-value</p>
<p>heighttopprop</p>	<p><code>\$en/popupwizard/njx_hsplitt_attr_heighttopprop\$</code></p>	<p>Optional</p>	
<p>hsplitstyle</p>	<p>CSS style definition that is directly passed into this control.</p> <p>With the style you can individually influence the rendering of the control. You can specify any style sheet expressions. Examples are:</p> <p>border: 1px solid #FF0000</p> <p>background-color: #808080</p> <p>You can combine expressions by appending and separating them with a semicolon.</p> <p>Sometimes it is useful to have a look into the generated HTML code in order to know where direct style definitions are applied. Press right mouse-button in your browser and select the "View source" or "View frame's source" function.</p>	<p>Optional</p>	<p>background-color: #FF0000</p> <p>color: #0000FF</p> <p>font-weight: bold</p>

comment	Comment without any effect on rendering and behaviour. The comment is shown in the layout editor's tree view.	Optional	
vscroll	Definition of the vertical scrollbar's appearance. You can define that scrollbars only are shown if the content is exceeding the control's area ("auto"). Or scrollbars can be shown always ("scroll"). Or scrollbars are never shown - and the content is cut ("hidden"). Default is "auto".	Optional	auto scroll hidden

VSPLIT Properties

Basic			
height	Height of the control. There are three possibilities to define the height: (A) You do not define a height at all. As consequence the control will be rendered with its default height. If the control is a container control (containing) other controls then the height of the control will follow the height of its content. (B) Pixel sizing: just input a number value (e.g. "20"). (C) Percentage sizing: input a percentage value (e.g. "50%"). Pay attention: percentage sizing will only bring up correct results if the parent element of the control properly defines a height this control can reference. If you specify this control to have a height of 50% then the parent element (e.g. an ITR-row) may itself define a height of "100%". If the parent element does not specify a width then the rendering result may not represent what you expect.	Optional	100 150 200 250 300 250 400 50% 100%
widthleftprop	\$en/popupwizard/njx_vsplit_attr_widthleftprop\$	Optional	

vsplitstyle	<p>CSS style definition that is directly passed into this control.</p> <p>With the style you can individually influence the rendering of the control. You can specify any style sheet expressions. Examples are:</p> <p>border: 1px solid #FF0000</p> <p>background-color: #808080</p> <p>You can combine expressions by appending and separating them with a semicolon.</p> <p>Sometimes it is useful to have a look into the generated HTML code in order to know where direct style definitions are applied. Press right mouse-button in your browser and select the "View source" or "View frame's source" function.</p>	Optional	<p>background-color: #FF0000</p> <p>color: #0000FF</p> <p>font-weight: bold</p>
comment	<p>Comment without any effect on rendering and behaviour. The comment is shown in the layout editor's tree view.</p>	Optional	
overflow	<p>Definition of the vertical scrollbar's appearance.</p> <p>You can define that the scrollbars only are shown if the content is exceeding the control's area ("auto"). Or scrollbars can be shown always ("scroll"). Or scrollbars are never shown - and the content is cut ("hidden").</p> <p>Default is "auto".</p>	Optional	<p>auto</p> <p>scroll</p> <p>hidden</p>

SPLITCELL Properties

Basic			
takefullheight	<p>Indicates if the content of the control's area gets the full available height.</p> <p>If you use percentage sizing inside the control's area then this property must be switched to 'true'. If you use no explicit vertical sizing at all - or you use vertical pixel sizing for your controls - the property must be switched to 'false'.</p> <p>Background information: container control's internally open up a table in which you place rows (ITR/TR) which then hold controls (e.g. LABEL/FIELD). The table that is opened up normally has no explicit height and grows with its content as consequence. By specifying "takefullheight=true" the table itself is sized to fill the maximum height of the available area.</p>	Optional	true false
cellstyle	<p>CSS style definition that is directly passed into this control.</p> <p>With the style you can individually influence the rendering of the control. You can specify any style sheet expressions. Examples are:</p> <p>border: 1px solid #FF0000</p> <p>background-color: #808080</p> <p>You can combine expressions by appending and separating them with a semicolon.</p> <p>Sometimes it is useful to have a look into the generated HTML code in order to know where direct style definitions are applied. Press right mouse-button in your browser and select the "View source" or "View frame's source" function.</p>	Optional	background-color: #FF0000 color: #0000FF font-weight: bold
comment	<p>Comment without any effect on rendering and behaviour. The comment is shown in the layout editor's tree view.</p>	Optional	

Defining the Split Size

The split size of HSPLIT and VSPLIT can be set in the following ways:

- Fixed definition of initial split size: by using the HSPLIT property `heighttop` and the VSPLIT property `widthleft`, you can preset the size in a "hard way". The value will be used as the initial size.

- By using the HSPLIT property `heighttopprop` and the VSPLIT property `widthleftprop`, the size can be defined by a server side property. Maybe you have some personalization in which the size is kept for every split area - and proposed the next time the user visits the page.