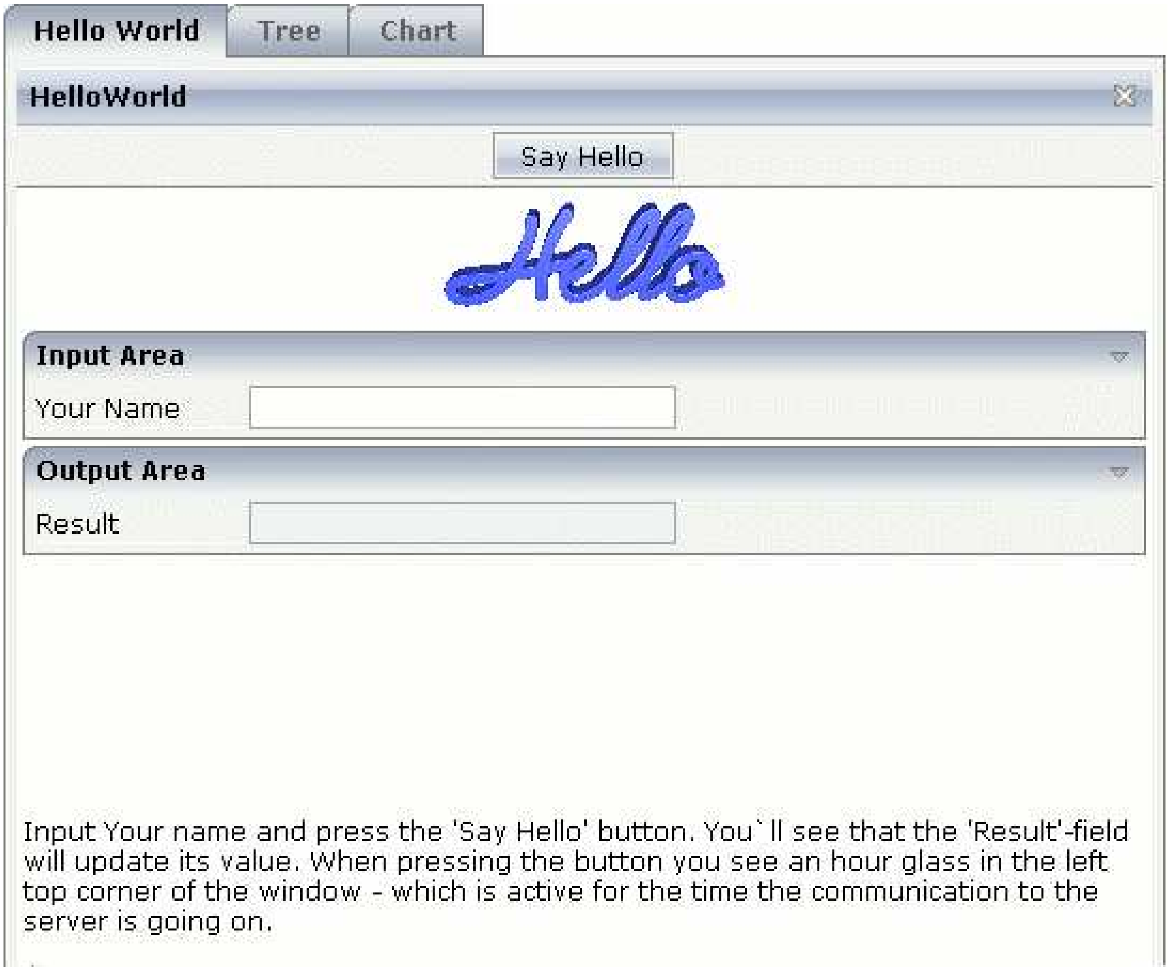
ROWTABSUBPAGES Control

The ROWTABSUBPAGES control allows to switch between several Application Designer pages using tabs. The displayed number of tabs and names are derived dynamically from its adapter properties.

Optionally, the ROWTABSUBPAGES control may contain exactly one STRAIGHTTABPAGE as a subnode. STRAIGHTTABPAGE must be the first tab. This allows for combining ROWTABAREA behavior with ROWTABSUBPAGES behavior. Having a STRAIGHTTABPAGE as the first tab improves the loading behavior of ROWTABSUBPAGES. For an example, see the **80_straighttabpage** layout in the **cisdemos** project.



Input Your name and press the 'Say Hello' button. You`ll see that the 'Result'-field will update its value. When pressing the button you see an hour glass in the left top corner of the window - which is active for the time the communication to the server is going on.

The XML definition is:

```
<pagebody>
    <rowtabsubpages pagesprop="tabpages" height="600" borderwidth="0">
    </rowtabsubpages>
</pagebody>
```

The ROWTABSUBPAGES control references a property `tabPages` which is provided by the adapter class of the page. The height can be specified depending on the whole page’s height or can be fix. The page style can be manipulated directly.

The corresponding adapter source is:

```
// This class is a generated one.

import com.softwareag.cis.server.Adapter;
import com.softwareag.cis.server.util.TABSUBPAGESInfo;

public class RowTabSubPageAdapter
    extends Adapter
{
    // property >tabpages<
    TABSUBPAGESInfo m_tabPages = new TABSUBPAGESInfo();
    public TABSUBPAGESInfo getTabpages() { return m_tabPages; }

    /** initialisation - called when creating this instance*/
    public void init()
    {
        m_tabPages.addItem("Hello&nbsp;World","HelloWorld.html");
        m_tabPages.addItem("Tree","trees_01.html");
        //m_tabPages.addItem("Dynamic&nbsp;Combo", "HelloWorld.html");
        m_tabPages.addItem("Chart", "HelloWorld.html");
    }
}
```

The property `tabPages` is of type `com.softwareag.cis.server.util.TABSUBPAGESInfo`. There are methods for adding and removing items from the `tabPages` object. See the JavaDoc documentation. The number of items can be changed at any time.

This chapter covers the following topics:

- Properties
- Performance Considerations

Properties

Basic			
pagesprop	Name of adapter property representing the control on server side. The property must be of type "TABSUBPAGESInfo". View the Java API Documentation for further information.	Obligatory	
triggerserver	Flag indicating whether the adapter should be triggered if the user switches between pages. If set to true, method trigger() inside the TABSUBPAGESInfo object is called - before switching the page. Therefore the adapter can abort a page switch - maybe a user has to enter some data first on the current page before switching to another one.	Optional	true false

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 percantage 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%
scrollable	If set to "true" then small icons will appear on the right border of the control. If the size of the "tabs" is too big and some tabs are cut as consequence then you can use these icons for scrolling left and right.	Optional	true false
textid	Multi language dependent text that is displayed inside the control. The "textid" is translated into a corresponding string at runtime. Do not specify a "name" inside the control if specifying a "textid".	Optional	
fastbufferswitch	If this property is switched to "true" (default is "false") then the contained subpages are buffered in a way that switching between tabs is not done by loading a new page but by just switching the visibility of pages. Please pay attention to that switching between pages in this case does not reload the page content from the server when switching! In order to enable fast switching you have to set the framebuffersize in cisconfig (n +1), n being the number of tabs to switch.	Optional	true false
comment	Comment without any effect on rendering and behaviour. The comment is shown in the layout editor’s tree view.	Optional	
Appearance			
borderwidth	Border width (in pixels) of the sub-page that is contained inside this control. Define "0" to avoid rendering any border.	Optional	1 2 3 int-value

leftindent	Inserts a horizontal distance left of the first "tab" and shifts the "tabs" to the right as consequence. The value you may define represents the number of pixels that are inserted.	Optional	1 2 3 int-value
paddingleft	Number of pixels which you want to keep as margin between the tab control’s left border and the inner sub page. Default is 5 pixel.	Optional	1 2 3 int-value
paddingtop	Number of pixels which you want to keep as margin between the upper tab row and the inner sub page. Default is 5 pixel.	Optional	1 2 3 int-value
paddingright	Number of pixels which you want to keep as margin between the tab control’s right border and the inner sub page. Default is 5 pixel.	Optional	1 2 3 int-value
paddingbottom	Number of pixels which you want to keep as margin between the bottom of the tab control and the inner sub page. Default is 5 pixel.	Optional	1 2 3 int-value

pagestyle	CSS style definition that is directly passed into this control. With the style you can individually influence the rendering of the control. You can specify any style sheet expressions. Examples are: border: 1px solid #FF0000 background-color: #808080 You can combine expressions by appending and separating them with a semicolon. 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.	Optional	
Miscellaneous			
testtoolid	Use this attribute to assign a fixed control identifier that can be later on used within your test tool in order to do the object identification	Optional	

Performance Considerations

Many users like the subdivision of pages into "tabs". Application Designer offers several controls for this - let us compare the ROWTABSUBPAGES control described in this section with the ROWTABAREA control described in the *Layout Elements* documentation.

The ROWTABAREA control has certain content areas (TABPAGEs) and always makes one of them visible. This means: the page has much more HTML code and controls then are visible. The size of the page is important for the performance of the page in the browser: the bigger the size, the longer it takes the browser to render a page (also if it is already cached).

The ROWTABSUBPAGES control offers a subpage in which you can place contained pages.

Now imagine that you have 500 fields to be displayed inside "tabs": in this case, it is more performant to build one "mother page" containing the ROWTABSUBPAGES control and to have five "detail pages", one for each "tab", than having one big page with all 500 fields, arranged by a ROWTABAREA control.

In the demo workplace, there is an example in which you can "feel" the difference - please have a look!