Opendiem Training

Exercise 5

Opendiem-TRN-0005

Revision	5.0.0		
Status	Initials	Date	Comment
Released	RAC	11/18/2010	Updated format and content

Exercise 5 – Advanced Opendiem Designer, Part I

Introduction

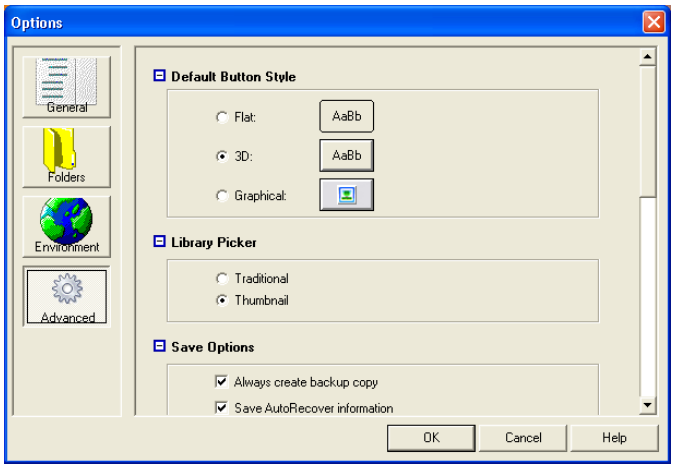
This Exercise covers some Opendiem advanced topics such as graphical buttons, animated graphics, Smart Components and Tab controls.

Objective

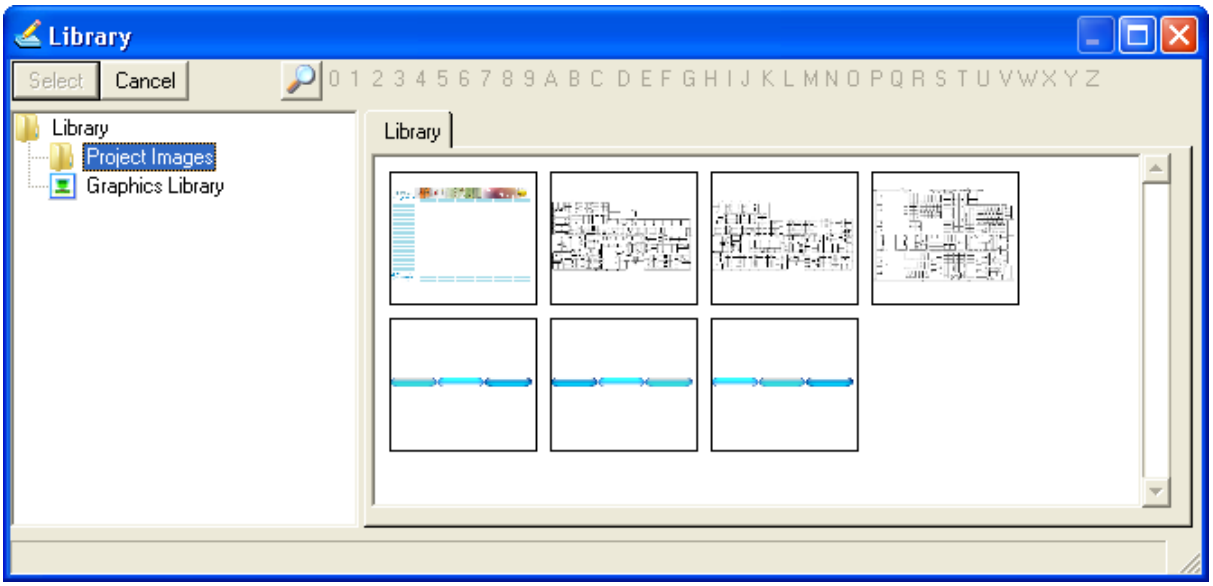
In this exercise you will use Opendiem Designer to add graphical buttons, and animated graphics to your web pages. We will also cover the use of the Opendiem System driver and introduce SmartComponents. A later topic introduces the Opendiem Tab control, which allows objects on different tabs to be made visible and invisible instantly.

Graphical Buttons

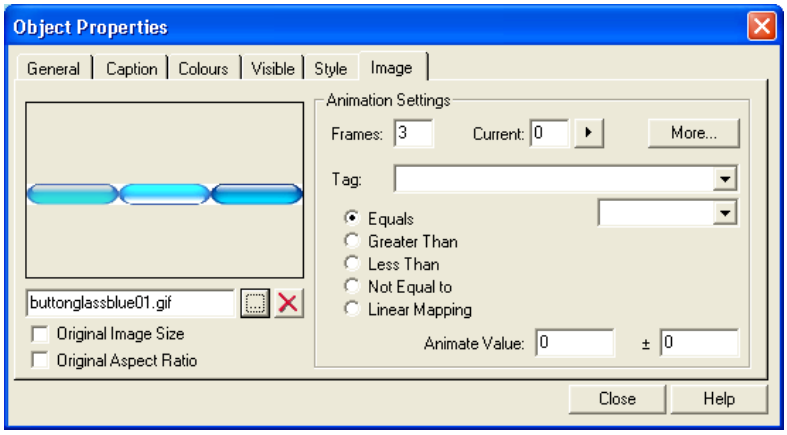
In previous exercises we added button objects to the screen. Opendiem has three styles of buttons, Flat, 3D and graphical. The default button style is selected in **Tools | Options… | Advanced**.



Add a button to the design screen and from the **Object Properties** dialogue select the **Image** tab and click on the ellipses (**…**) the library viewer will appear as shown below. The library viewer allows you to organize your graphics objects on a hierarchical fashion and provides a thumbnail viewer to enable rapid selection of the object required.



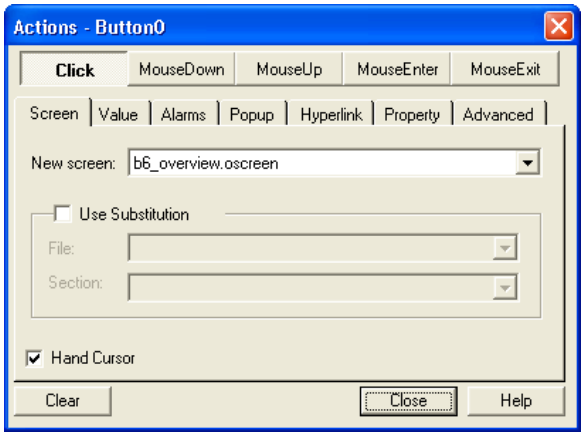
Select the glass button style as shown below and select **Original Image Size**. The button consists of 3 frames, one each for normal, mouse over and mouse down. In the **Frames** box enter '3' and click **Apply**, the graphical button will appear on the Design Screen



Select the **Caption** tab and enter the button text.

Tip: Double click on the button text in the Design Screen to edit the button text in place or F2.

The final step is to define a click action, select 'screen2.oscreen' as the new screen to load in the click actions.



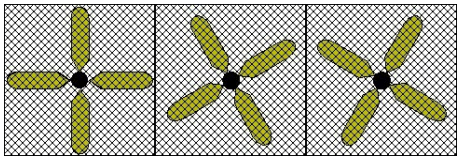
Test the graphical button in the browser.

Animated Images

In Exercise 4 we added a static graphic image to the Design Screen, in this example we will add graphics images which simulate animation dependant on Tag values. As an aid to speeding the development process we will use the Opendiem System driver to produce synthetic values which we will use to verify the animation effects.

First we will look at the details on animated objects and follow up with an example.

To allow Opendiem to animate an image, the image file must contain all the individual frames for the animation in the form of a horizontal 'Film Strip'. The image below illustrates a typical image that Opendiem can animate. The image is defined as having three frames, so Opendiem will only show one-third of the image. As the animation continues, a different frame of the image is shown to give the illusion of a rotating fan. For the animation to be effective it is important that the number of frames divides exactly into the image width in pixels with no remainder. E.g. for the image below the total number of pixels in the width could be 120 which is exactly 40 pixels per frame.

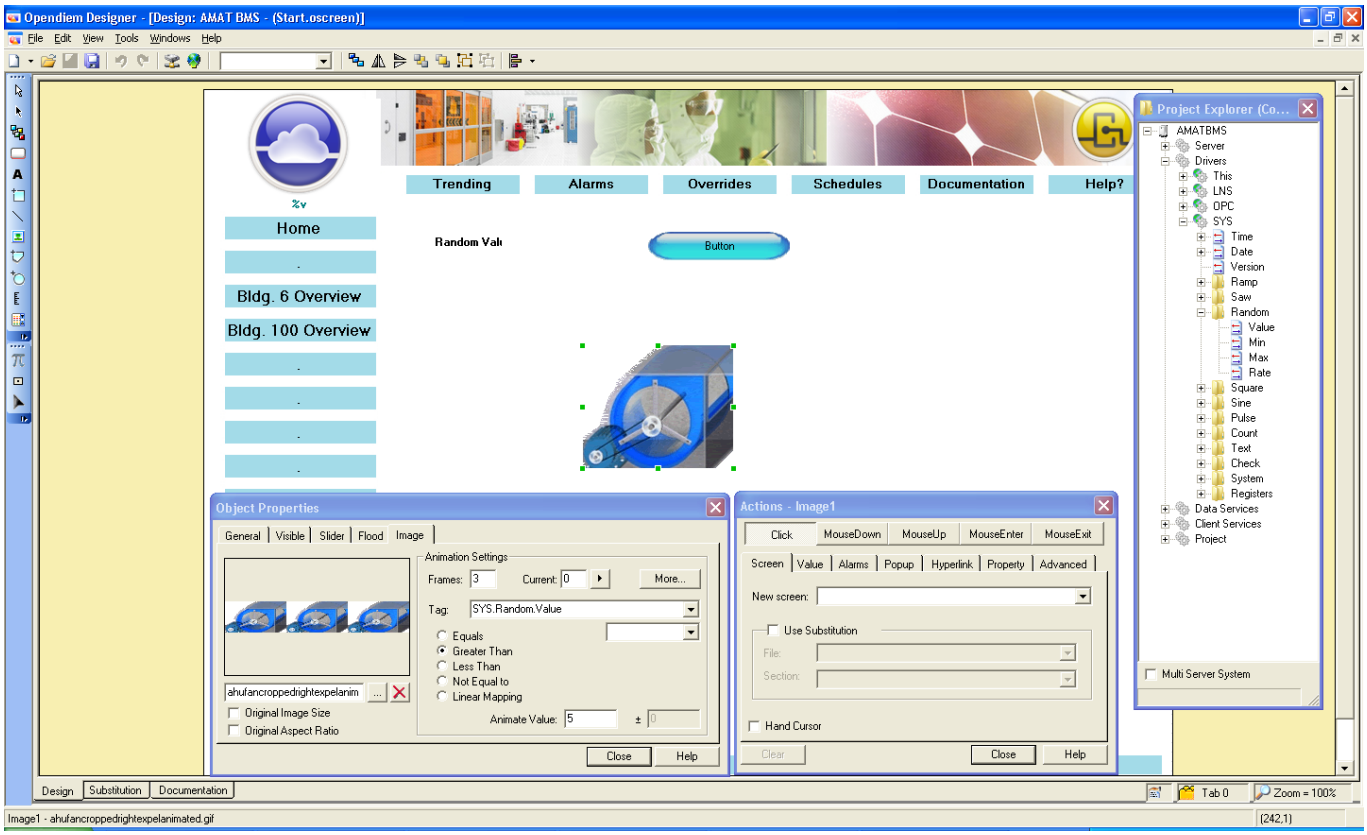


To Create an Image Placeholder	Click on the Picture tool within the Toolbox and draw the required sized placeholder.
To Select an Image	From within the Image Tab, click on the browse button  . This displays the images dialog box containing the default Opendiem Images folder …Opendiem\…\website\images. If the required image has already been copied into this folder then select it and click **Open**. Otherwise, click on the search button  and locate the image. Clicking **Open** will copy the selected image file into the Opendiem images directory. Select the file from the images dialog box and click **OK**
To Delete an Image	To delete the image out of the preview box click on  .

Original Image Size	Displays the image at its original size.
Original Aspect Ratio	Returns the selected image to its original aspect ratio of width to height.
To Animate an Image	1. Select a suitable image to animate, for example the fan. 2. Enter the number of Frames to be animated, which for this fan is 3. 3. Enter the Animate Value, which is the value at which you want the object to animate. This value is provided by the variable referenced in the tag box. 4. Enter a variable reference into the tag box and click ok

Loading the image.

Place an image area on the start.oscreen screen, the size is unimportant. Browse for a file called '`au_fan_anim_3.jpg`'. Select original image size and in the 'Animate' section specify 3 frames. Click on **Apply**.

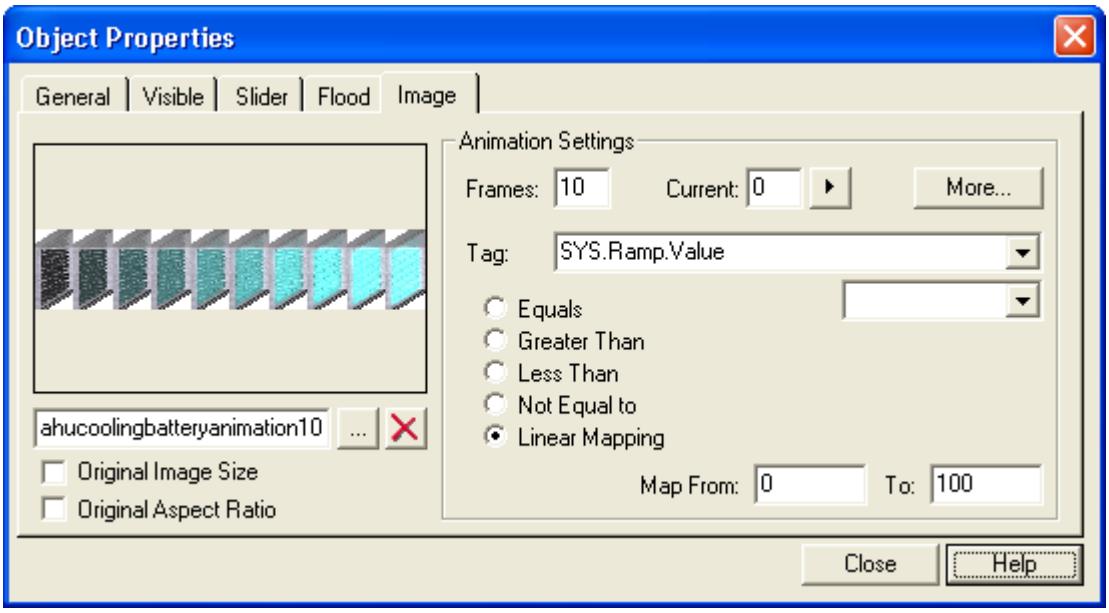


Click on the **Current** button to cycle through the frames and see how the animation will appear. Drag the **Sys.Random.Value** from the Project Explorer onto the Tag and specify the animate value. For this example the fan is required to animate whenever the Sys.Random.Value is greater than 5. Save the screen and validate your results in a Web browser.

Animated sequences.

Another type of animation sequence allows different frames within a 'Filmstrip' to be displayed depending on the Tag value. This feature is useful for displaying position images or level images based on a Tag; for example to display a cooling coil in different stages according to cooling level we might use the following.

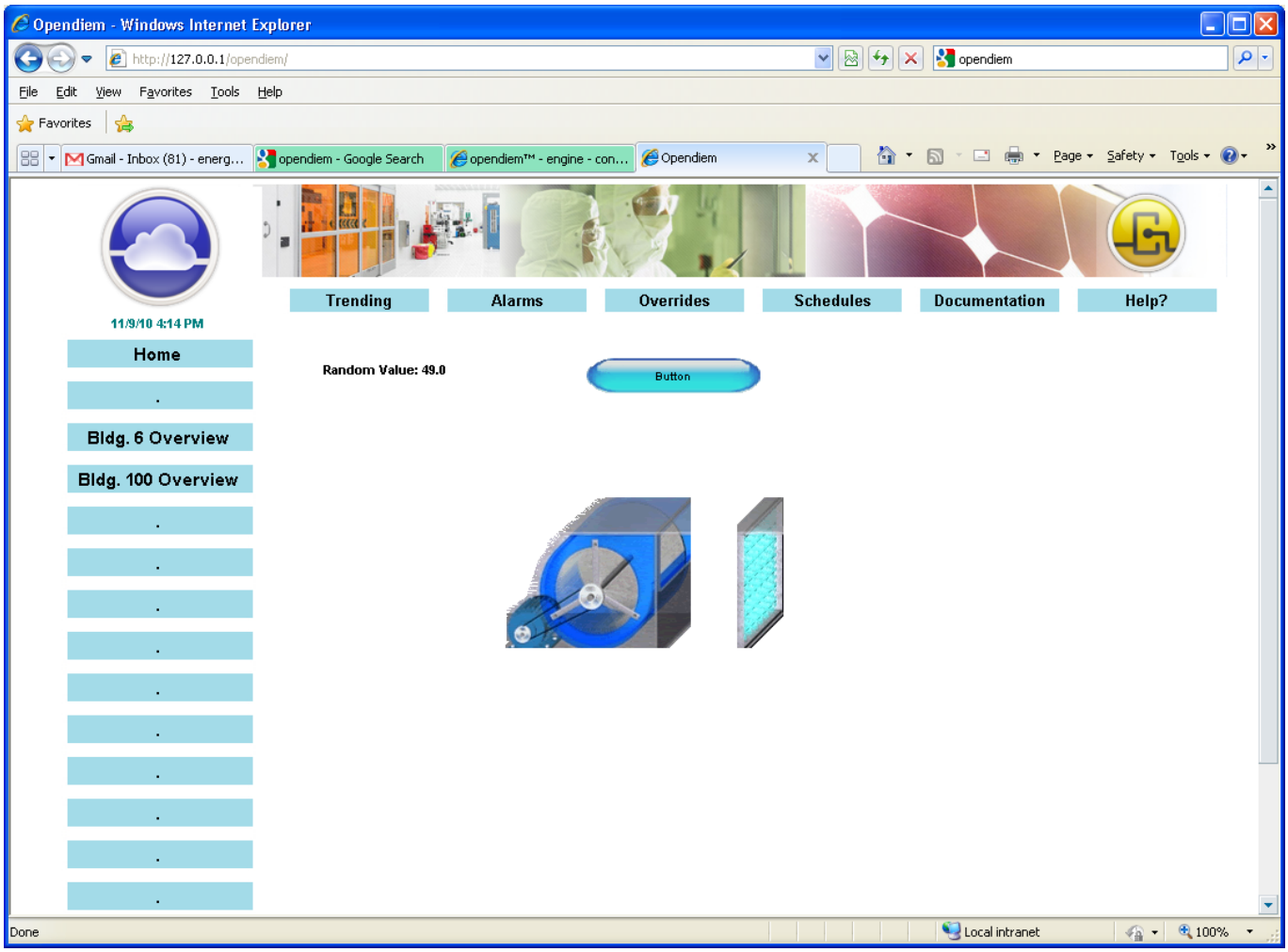
Place an image area on the start.oscreen screen, the size is unimportant. Browse for a file called `ahu_cooling_battery_animation_10frames'. Select original image size and in the **Animate** section specify 10 frames. Click on **Apply** and adjust the position of the Object Properties dialogue so that you can see the image on the Design Screen, see below.



Click on the **Current** button to cycle through the frames and see how the animation will appear. Expand the **Drivers | SYS | Ramp** tree in the Project explorer and drag the **SYS.Ramp.Value** onto the Tag. For the animate value select a **Linear Mapping** from 0 to 100.

The SYS driver is useful for validating animation sequences as it can generate data over an entire range without requiring actual network data which can be time consuming to generate. The SYS driver contains many system generated values including ramp, saw, random, sine, square; each with their own rate, min and max limits.

The default rate is one update per second (60) for this example, create a spin button that allows you to change the rate between 60 and 240 in steps of 10. Save your work and check the results in the browser, your screen should be similar to the one below.

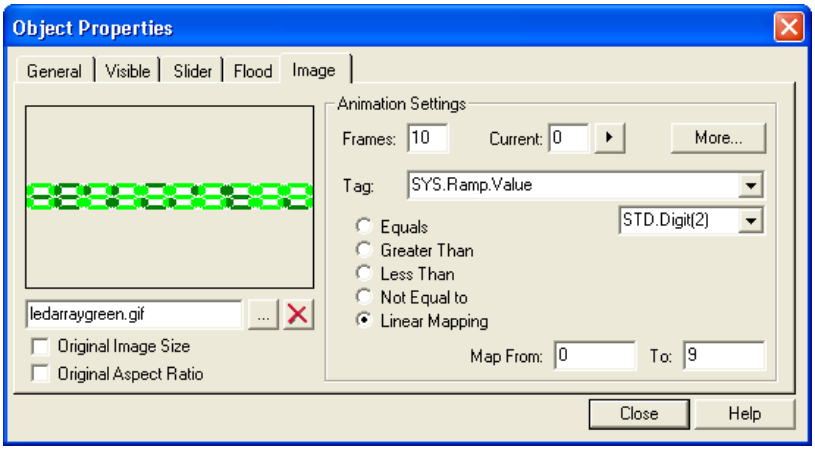


Applying Data Processing to Animations.

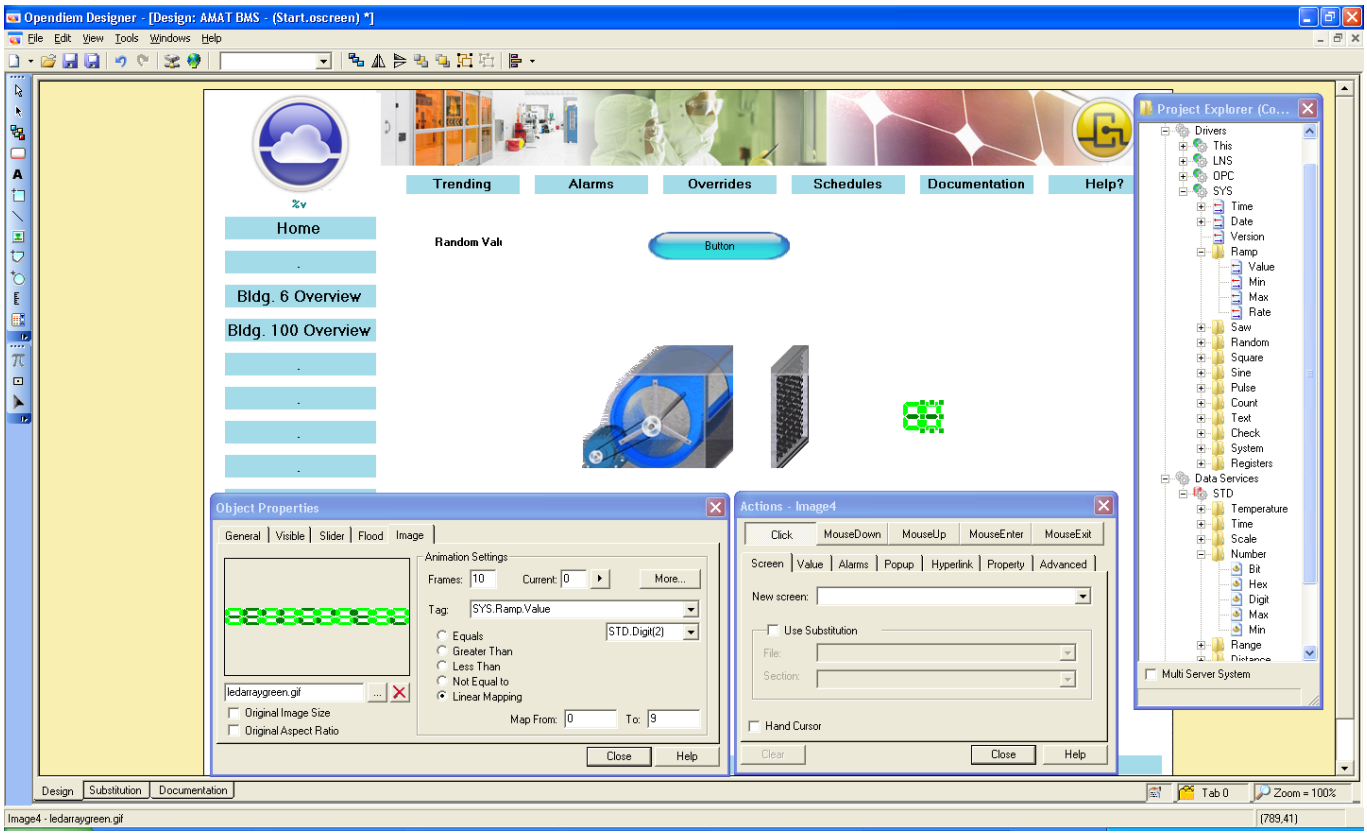
The following example shows the use of the Opendiem Standard functions to apply data processing to an animation. Standard functions are covered in greater detail in a later exercise but are shown here in relation to graphics.

Place an image area on the start.oscreen screen, the size is unimportant. Browse for a file called '`LEDArrayGreen.gif`'. This is a filmstrip of a seven segment LED display. In the **Animate** section of the Object Properties specify 10 frames. Click on **Apply.** Expand the **Drivers | SYS | Ramp** tree in the Project explorer and drag the **SYS.Ramp.Value** onto the Tag. For the animate value select a **Linear Mapping** from 0 to 9 as shown below. Expand the **Data Services | STD | Number** tree in the Project explorer and drag the **Digit item** onto the Tag modifier as shown

below. The box will show '`STD.Digit(Digit)`' Change the Digit in the parentheses to '`1`' to indicate that digit 1 of the number should be extracted from the value. Click **OK**.



Duplicate the display and position the graphic to the left of the first graphic as shown below. Edit the properties and change the digit number to 2 ie '`STD.Digit(2)`'. Place a dark panel behind the LED's to improve the contrast then preview the screen in the browser.



SmartComponents

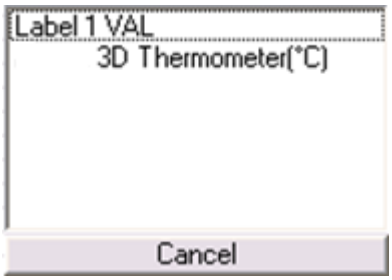
SmartComponents are pre-defined objects which can have intelligent associations with the data and is a powerful feature of Opendiem that compliments the shapes and allows complex screens to be quickly and easily designed.

SmartComponents are linked rather than added to the screen and loaded dynamically when the screen is being viewed. If changes are made to a SmartComponent on the disk then these changes will be reflected on all screens that use that object.

The power behind the SmartComponent is the ability to give it intelligence. A SmartComponent can be associated either with a single variable or a set of variables, such as different variables on a single device or even different variables on multiple nodes on a sub system.

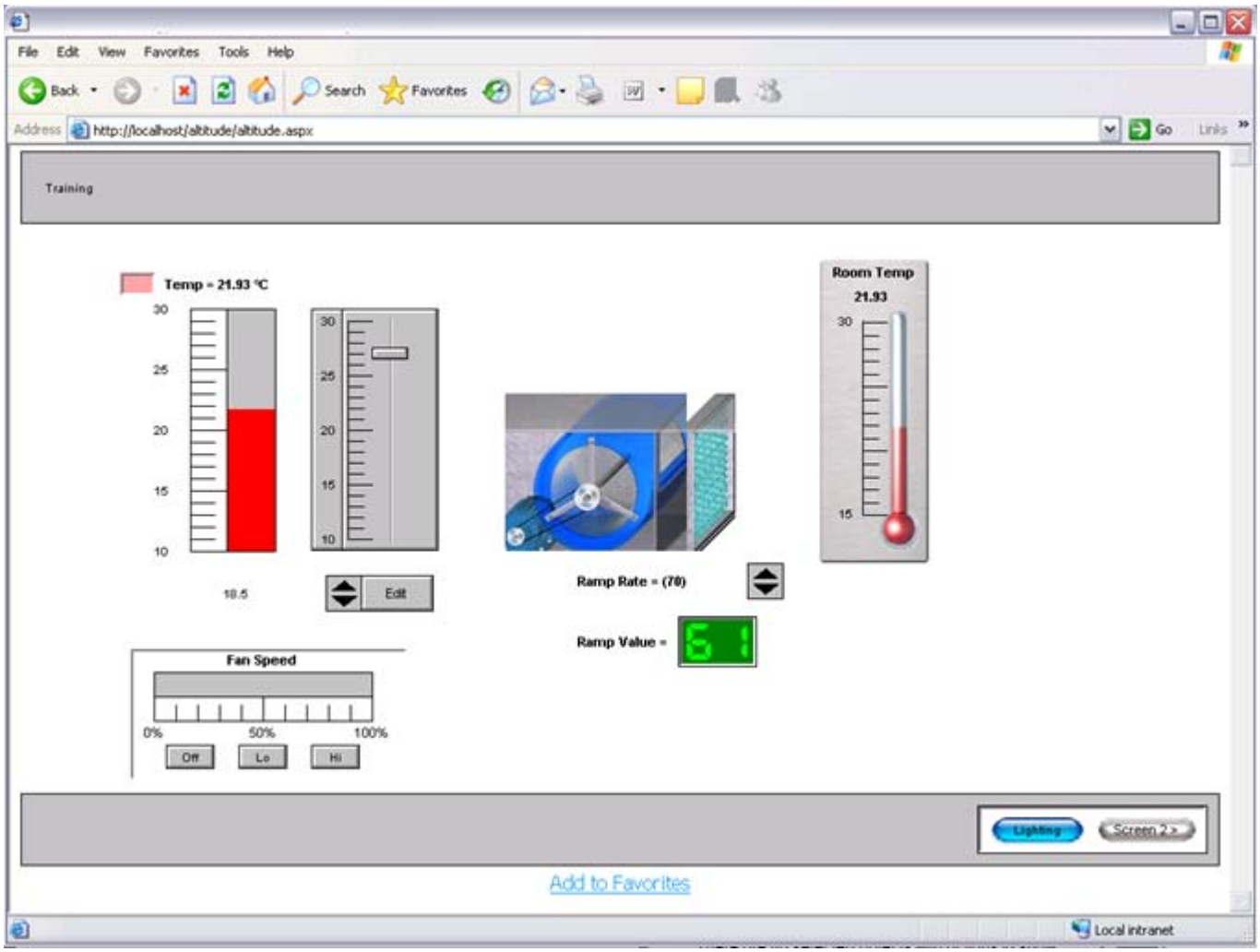
SmartComponents can be created by dragging and dropping a variable with an association from the project browser. For example, a LonWorks® NV of type SNVT_temp_p can be associated with a SmartComponent representing a thermometer.
SmartComponents have custom properties and in the example above dragging the variable SNVT_temp_p onto the design screen can produce a fully configured thermometer with the custom properties automatically completed.
The LonMark® SNVT type SNVT_temp_p is associated with a thermometer SmartComponent. The following exercise demonstrates dragging the temperature variable onto the design screen to instantly produce a configured thermometer complete with variable reading and a flood fill of a GIF image proportional to the variable value.

From the Project Explorer drag the variable nvoRoomTemp onto the Design screen without holding the 'Shift' key. A pop up list of associated SmartComponents will appear as shown below, select '3D Thermometer(°C)'.



A Temperature Gauge SmartComponent will appear on the screen. Select the Object properties for the SmartComponent and select the **Custom** tab; here you can Change exposed properties of the SmartComponent, in this case the Header Title.

View your results in the browser; the screen should be similar to the one below.



Advanced SmartComponents; Reducing Engineering Effort

The above exercises illustrate simple objects such as a thermometer & dial, however SmartComponents can be created that can represent entire subsystems such as an Air Handling Unit or a Conveyor Line complete with all variable values such as temperature, pressure, valve position, motor speed, damper position etc, together with the associated graphics. This powerful feature can dramatically reduce the amount of time and effort required to produce complex mimics.

SmartComponents can be converted back to objects by right clicking on the SmartComponent and selecting **Object | Convert to Objects**. This facility is useful where a SmartComponent Exists that closely matches the requirements but small changes need to be made to meet the exact requirements of a particular project.

Tab Controls.

Tab controls consist of a series of tabs similar to those found in popular spreadsheet programs such as Excel. Tabs in Opendiem work in a similar way to layers in that objects can be placed on different Tabs and made visible or invisible based on the currently selected Tab. Objects on Tab 0 are always visible regardless of the currently selected Tab therefore the Tab control should always be placed on Tab 0.

In this exercise we will change some components on the 'start.oscreen' page from Tab 0 (the default) to Tab 1 and then control the Tab visibility using a Tab Control SmartComponent.

First, add the tab Control SmartComponent to Tab 0. From the toolbox select the Smart Component, if the Tab Strip Smart Component is not in the list select '`more ->`' and browse to select '`SSTab.csc`'. Place an area on the screen big enough for two buttons. In the Object Properties dialogue select 2 Tabs from the drop down list. Enter text against Tab 1 e.g. Fan & Battery and against Tab 2 enter No Fan & Battery.

Now we will assign the Fan animation & the Cooling battery to Tab 1. Drag a selection box around the components. Click right on the group just selected and in the **General** tab of the Properties change the **Tab** to 1; click **OK**. The group selected is now on Tab 1 and will only be visible when Tab 1 is active. Send the screen to the browser and experiment making the Tab Active & Inactive by clicking on the two Tab buttons.

The benefit of the Tab Control is that all items on a screen are always loaded and the data associated with the screens is placed on the Opendiem Scan List therefore actions on the Tab control are instantaneous.

End of Exercise 5

In this exercise we have experimented with advanced features of Designer such as animated images, SmartComponents, the SYS driver, STD service and Tab controls.

Notes:

Version 5.00 November 2010

Building Clouds
3229 Whipple Road
Union City, CA 94587

Email: support@buildingclouds.com
http://www.buildingclouds.com