

PEACOCK

Multi Function Display System

(Rev. 1.3 ENG)



Index

- 1. Intro 4
 - 1.1 What is Peacock..... 4
 - 1.2 Who is Peacock for 4
 - 1.3 Main features..... 4
 - 1.3.1 Project management 4
 - 1.3.2 Layout management 5
 - 1.4 Minimum system requirements..... 5
- 2. Installation 6
 - 2.1 Setup 6
 - 2.2 Setup for Distributed version 7
- 3. Peacock’s Editor..... 10
 - 3.1 The ToolBar..... 11
 - 3.1.1Home 11
 - 3.1.2 Edit 12
 - 3.1.3 Screen Layout 13
 - 3.1.4 Tools..... 13
 - 3.2 The Toolbox 17
 - 3.2.1 Inserting a plugIn..... 17
- 4. Objects management 18
 - 4.1 Monitors..... 19
 - 4.1.1 Overview..... 19
 - 4.1.2 Properties 20
 - 4.1.3 Alarms Executors / Notifiers 21
 - 4.1.4 Project Panel 22
 - 4.2 Overview..... 22
 - 4.2.2 Common Properties 23
 - 4.2.3 Text Controls 27
 - 4.2.4 Time Controls 34
 - 4.2.5 Image Controls 40
 - 4.2.6 Internet Controls..... 42
 - 4.2.7 Media Controls..... 43
 - 4.2.8 AxelSoftwareMonitors 50
 - 4.3 Components..... 58

Index

4.3.1 Alarms Executors.....	58
4.3.2 Alarms Notifiers.....	61
4.3.3 Components	65
5. HowTo.....	65
5.1 Export / Import of a layout.....	65
5.2 Creating a project.....	68
5.2.1 Creation of a Digital Signage project	71
5.2.2 Creation of a Web TV project.....	77
5.3 Creation of a DMLVideo System Monitoring project	83
5.4 How to create the Streaming of a Peacock project.....	87
5.5 Creating a scheduled Peacock project	89
6. Glossary	90



Via Caduti Di Sabbiuno 6/F
40011 Anzola Emilia - Bologna - Italy
Tel. +39 051 736555 - Fax. +39 051 736170

e-mail: info@axeltechnology.com
web site: www.axeltechnology.com

All rights are reserved. No parts of this manual can be reproduced or spread through photocopies, microfilms or any other media, without the written permission from Axel Technology.

All quoted brands belong to the respective owners.

Intro

1. Intro

Compliments for having acquired Peacock, the multi-function system for the creation and management of video PlugIns.

1.1 What is Peacock

Peacock is an integrated environment for the realization of visualization projects and the management of their projects. The particular organization of the user interface, combined with the flexibility of the program, allow you to obtain fast and good results. Peacock provides a complete audio / video / graphics editor that allows anyone to create their own contents with great speed and flexibility. Moreover, Peacock can run and control its projects with the opportunity to make changes and updates during the execution of the project itself.

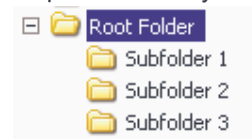
1.2 Who is Peacock for

Peacock finds its ideal application field in: pubs, hospitals, shop-windows, showrooms, schools, public environments, governmental buildings, waiting rooms, supermarkets, fairs, airports, hotels, shops, highways, information services, ships, sport complexes, gyms, security systems, web-tv and in any places where high quality and high reliability dynamic audio-visual information are needed.

1.3 Main features

1.3.1 Project management

The management of a Peacock project and its executable file is based on the concept of path's relativity. It is important that all the external objects, images, and texts that are linked to Peacock's controls are in subfolders of the Root Folder, thus creating a structural stability which also allows the integral displacement of the project (that is the Root Folder) on any other Pc, without changing its functionality.



The screen size (in Design mode) is better to be set according to the proportions of the output (that is the screen where the Preview is created). Even if they have different proportions (e.g.: Design is 800x400, that is 4:3, and Preview is a 16:9 tv), the output is created but the objects inserted in Design will be resized according to the Preview.

For each screen inserted on Peacock's project, it is possible to create an executable file. This file is default created into the Root Folder and has the screen's name. Together with the executable file, all the .dll files that are necessary to the functioning of it are created. If we move the Root Folder, we also need to move its content in order to keep the functionality of the executable file.

For the creation of a project, refer to "5. HOW TO" Chapter

1.3.2 Layout management

The layout management has been conceived to help the work of the user that has to create several projects that are similar as for the tipology of the inserted objects, but different in their content. It is possible to export the layout (therefore, only Location and Size properties) of all the objects on the selected screen.

For the layout management, refer to "5. HOW TO" Chapter



1.4 Minimum system requirements

Peacock needs the following hardware and software requirements:

HARDWARE

- CPU: Intel DualCore
- RAM: 2Gb
- VIDEO: VGA video card

SOFTWARE

- OS: Windows XP SP2
- [Microsoft Framework 3.5](#) installed.

2. Installation

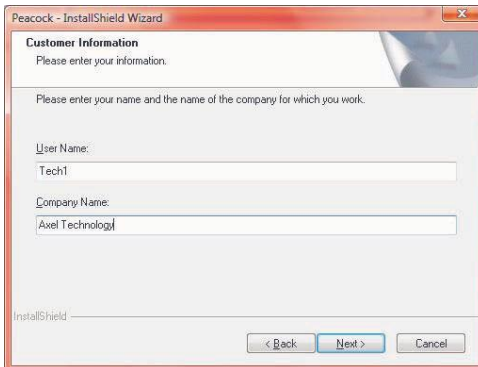
There are two different setup types of Peacock. The first type is for installing the Peacock software, the second type must be used on the Pc where you want to execute the file compiled by Peacock.

2.1 Setup

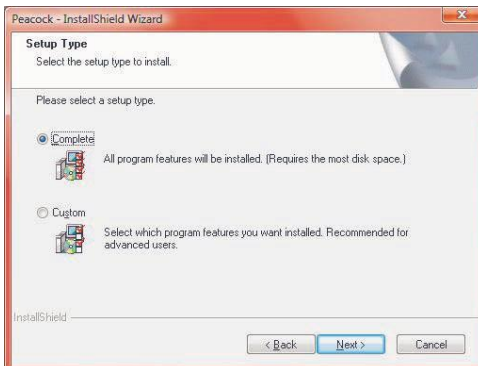
Run the setup file, as it is shown in figure:



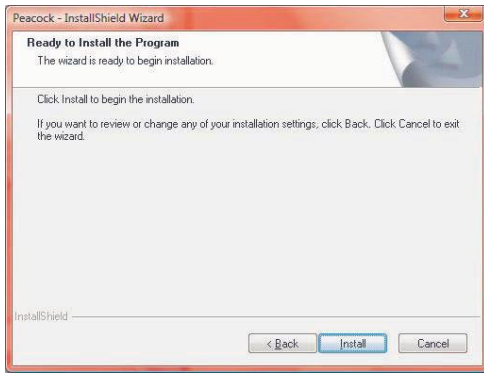
Enter "User Name" and "Company Name" data:



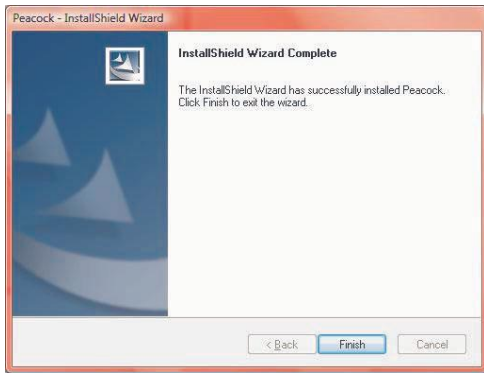
Select the desired setup type:



Click on the “Install” button:



Once the installation is completed, click on the “Finish” button:

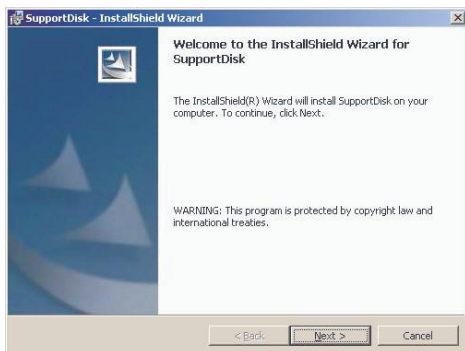


Insert the hardware key

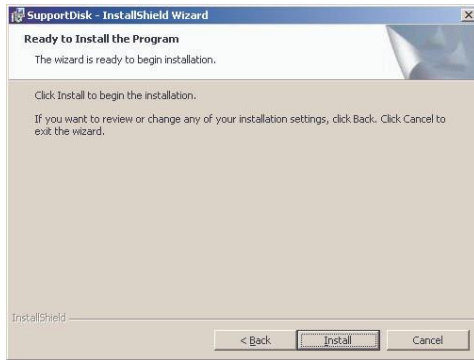
2.2 Setup for Distributed version

There is also a setup for the Distributed version, its installation is necessary on the Pc where you have to execute the compiled file that has been created with Peacock.

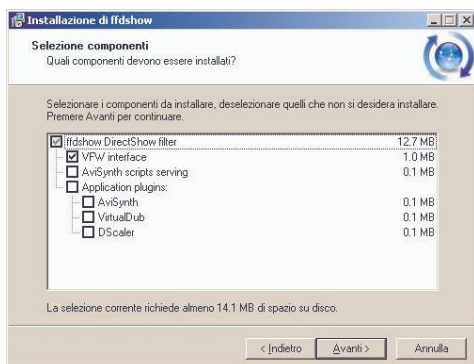
This installation registers and copies all the DLL files and codecs needed for the functioning of the compiled file.

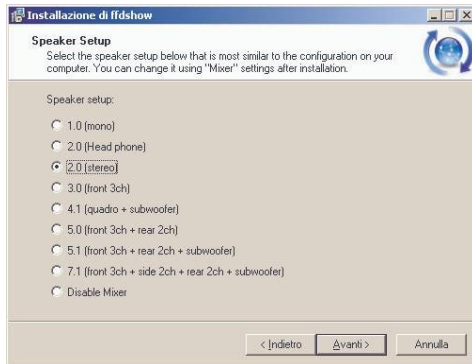
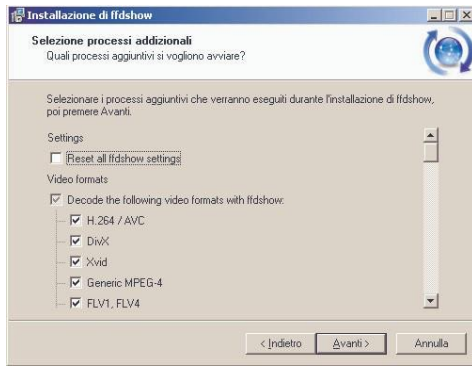


Index



Select the language and continue with the installation:



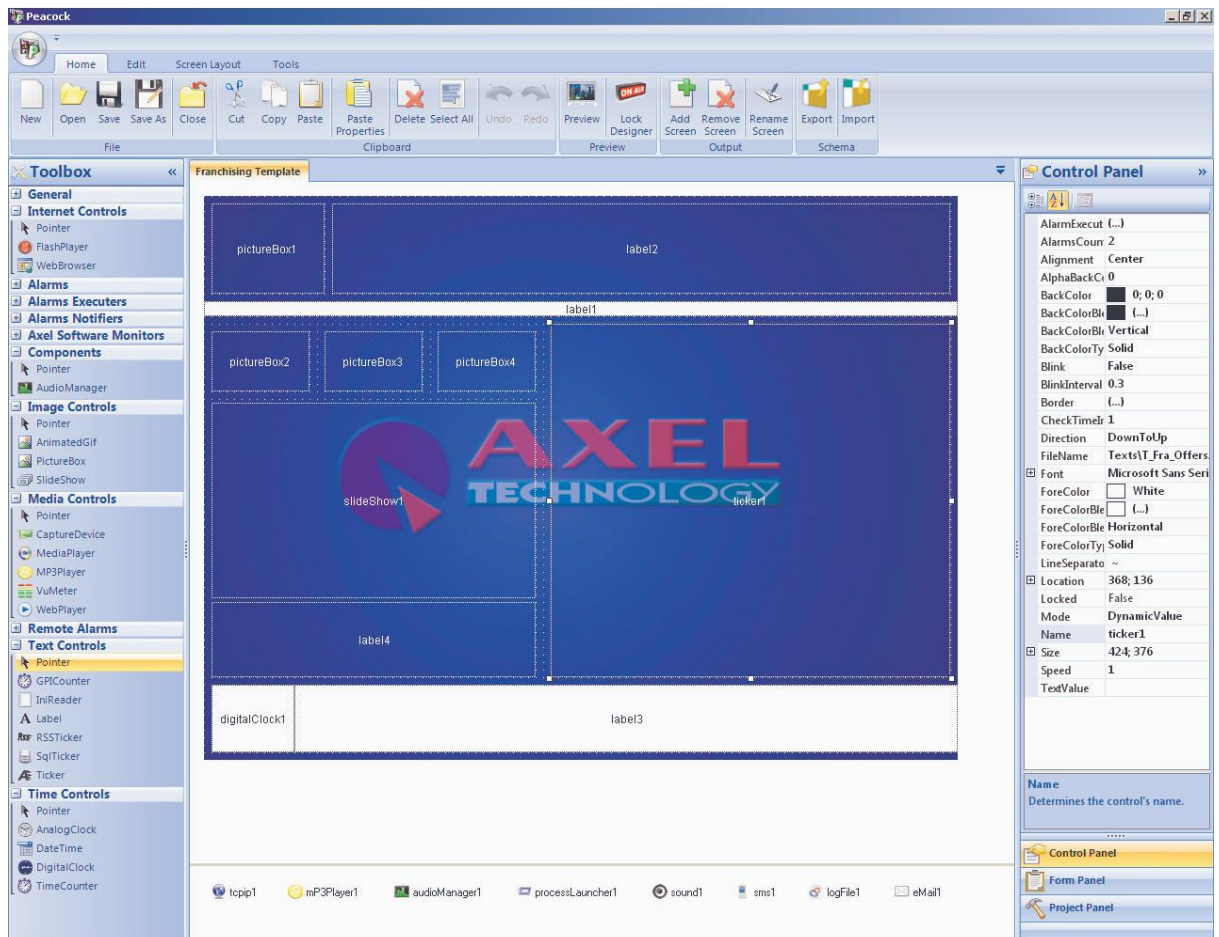


3. Peacock's Editor

One of Peacock's most important features is the ability to work in real time during the whole creation and setting of a project. It means that the program immediately responds to the user's actions, without any waiting times, always ensuring the best quality in final result.

Peacock's interface fits to any video resolution.

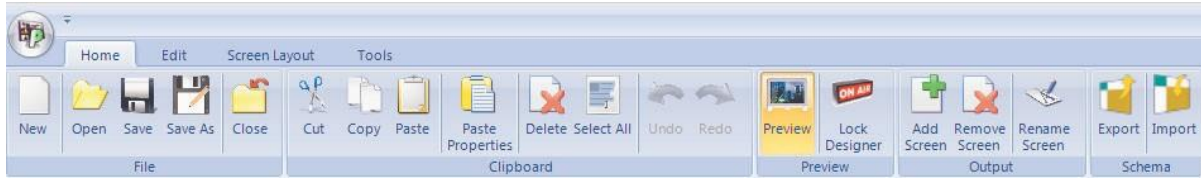
Peacock shows all the available objects, called "PlugIn", inside the **Toolbox** on the left of the screen. All the used plugIns are shown both in the work area and in the **Objects list**; they can be customized through the related panel (**Object properties**) shown under the Objects list.



3.1 The ToolBar

The operations concerning the management of a project are gathered in the toolbar upon the Peacock's editor.

3.1.1 Home



File

- New: it opens a new empty project
- Open: it opens an existing project
- Save: it saves the changes done on the project
- Save as: it saves the changes done on the project in a path and with a name specified by the user
- Close: it closes the current project

Clipboard

- Cut, Copy, Paste: operations on the selected object
- Paste Properties: it copies, in the selected object, the properties of the object previously stored in the clipboard
- Delete: it deletes the selected object
- Select All: it selects all the objects into the project

Preview

- Preview: it enables/disables the preview window of the project
- Lock Designer: it locks/unlocks the editing mode of the project

Output

- Add Screen: it adds a new work area and the related preview window
- Remove Screen: it removes the selected work area and the related preview window
- Rename Screen: it allows to rename the selected work area

Schema

- Export: it allows to export the layout of the selected screen.
- Import: it allows to import on a screen a layout that has been previously exported.

3.1.2 Edit



Control Properties

This area collects a set of tools for the formatting of the selected control.

Text Properties

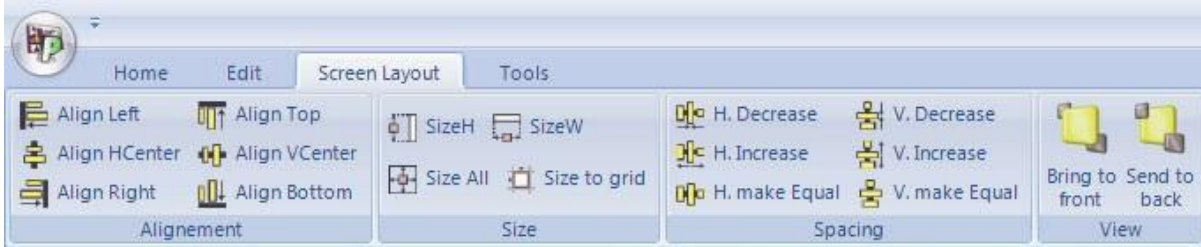
This area collects a set of tools for the formatting of the text that is associated to the selected control.

Action

This area contains all the tools that directly interacts with the selected controls:

- Show / Hide Status: it shows, if it exists, the status form of the selected control.

3.1.3 Screen Layout



Alignment

This area collects the tools for managing the alignment among the selected items

Size

This area collects the tools for managing the relative scaling among the selected items and with the grid of the work area.

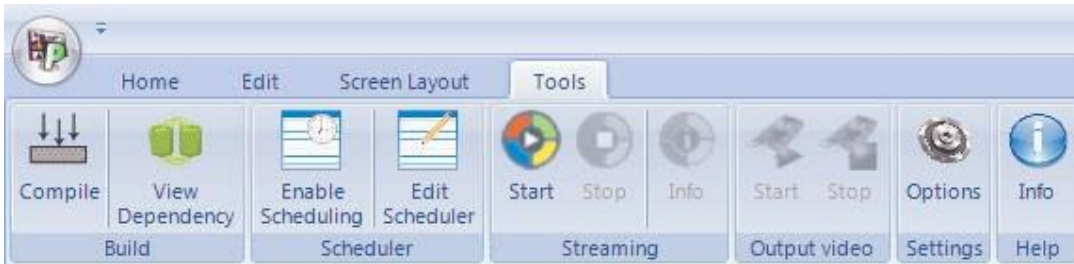
Spacing

This area collects the tools for managing the spacing among the selected elements

View

- Lock controls: it allows to lock/unlock the option for editing all the items of the project
- Bring to front: it moves the selected item on the foreground
- Send to back: it moves the selected item on the background

3.1.4 Tools



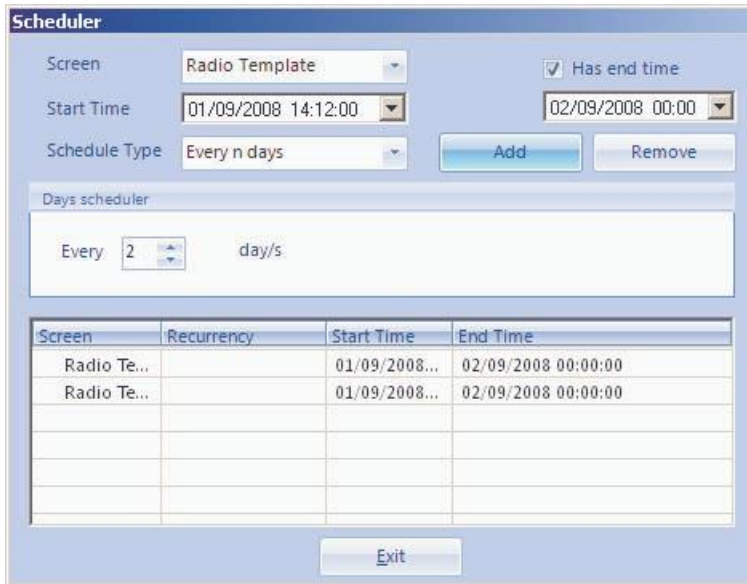
Build

- Compile: it allows to create an executable file of the project, that can be run without the software.
- View Dependency: it shows what are the files and libraries that are needed for running the project.

Scheduler

- Enable Scheduling: it enables the scheduler.
- Edit Scheduler: it shows the general settings of the scheduler

All the settings of the Edit Scheduler window are listed and explained below:



- Screen: it determines on which output the scheduler is set.
- Start Time: it determines the start day and hour for the scheduler
- Has end time: if enabled, it determines the end day and hour for the scheduler
- Schedule Type: it determines the frequency of the scheduler on the selected screen.
- Add: it adds the parameters set on the scheduler list.

Streaming

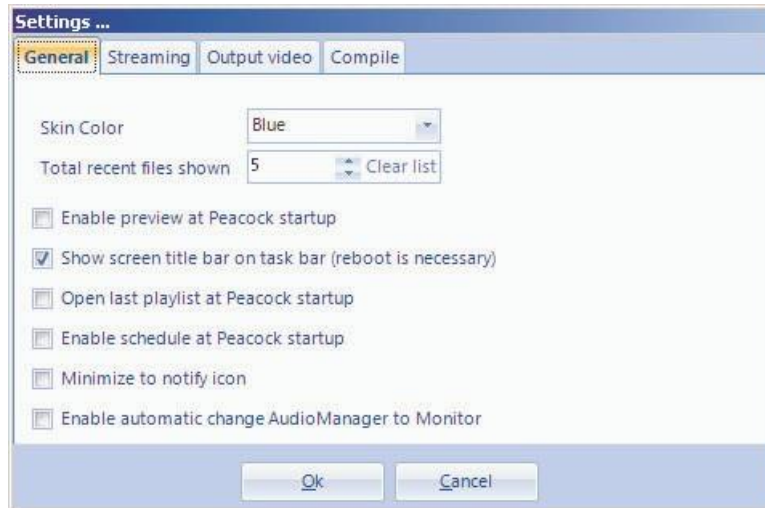
- Start: it starts the streaming of the selected monitor.
- Stop: it stops the streaming of the selected monitor.
- Info: it shows the general settings of the streaming.

Settings

- Options: it shows the general settings of the program.

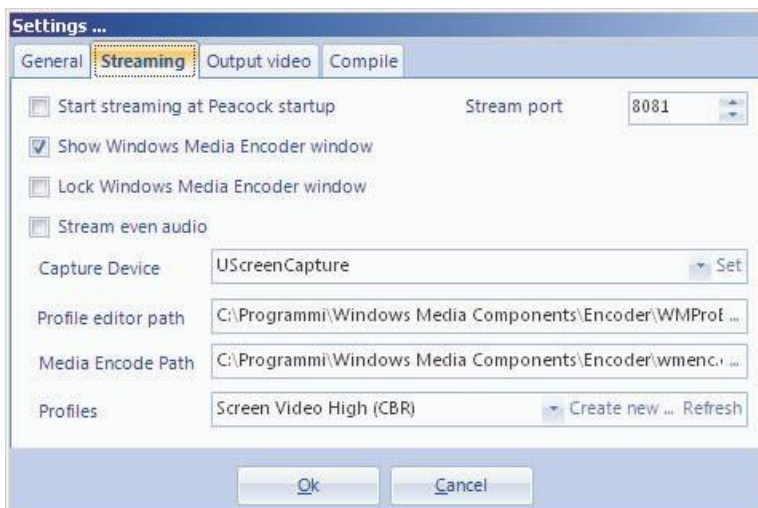
The settings of the Options window are listed and explained below:

 **General**



- Skin Color: it determines the style of Peacock.
- Total recent files shown: it sets the maximum number of projects that can be saved on the recent playlist. The “Clear list” button clears the recent playlist.
- Show screen title bar on task bar: it minimizes the program at startup on the taskbar.
- Open last playlist at Peacock startup: it opens the last Peacock project in the recent playlist.
- Enable schedule at Peacock startup: it enables the scheduling at the program’s startup.
- Minimize to notify icon: reduce il programma all’avvio come icona sulla barra delle applicazioni.

 **Streaming**

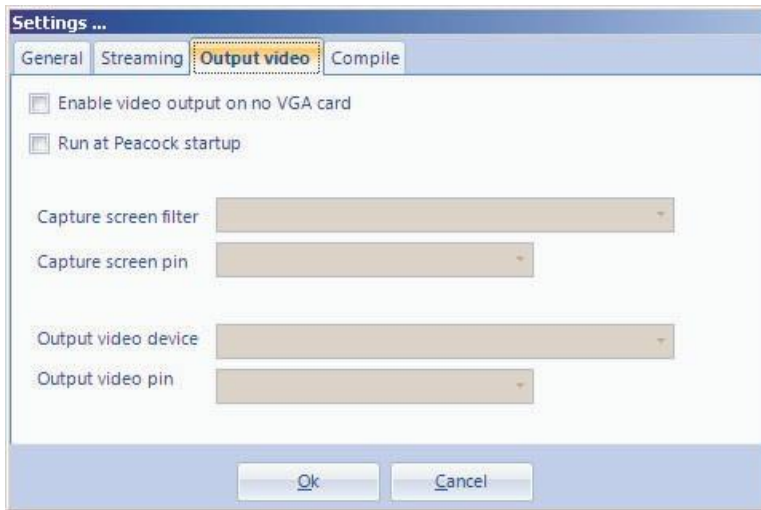


- Start streaming at Peacock startup: it allows to enable the streaming at Peacock’s startup.
- Show Windows Media Encoder window: it allows to show the window of Windows Media Encoder during the streaming.
- Lock Windows Media Encoder window: it allows to show and disable the window of Windows Media Encoder during the streaming.

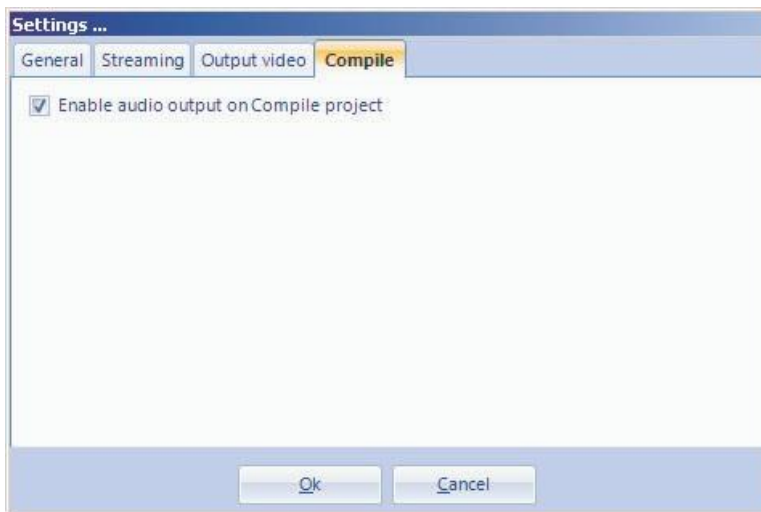
Index

- Stream even audio: it allows to stream the audio any time you enable the streaming.
- Stream port: it sets the TCP/IP output port of the streaming.
- Capture Device: it sets the card to be selected for the streaming output. Use the “Set” button to set the card.
- Profile editor path: it determines the path of Windows Media Profile Editor to be used for the streaming.
- Media Econde Path: it determines the path of Windows Media Encoder to be used for the streaming.
- Profiles: it allows to choose a profile of WME, previously saved through the “Create new...” button, to be used for the streaming. Use the “Refresh” button to refresh the list of profiles.

 Output video



Compile



- Enable audio output on Compile project: it enables the audio output of the project on the compiled file.

3.2 The Toolbox




Every project is composed by one or more plugIns, each one with its own life and features. Peacock shows the plugIns that are loaded at startup into the **Toolbox** placed on the left of the screen.

The plugIns of each project are gathered into the **Objects list** that allows to select and manage them.

3.2.1 Inserting a plugIn

To insert a new plugIn, you need to select it from the Toolbox and drag it into the work area.

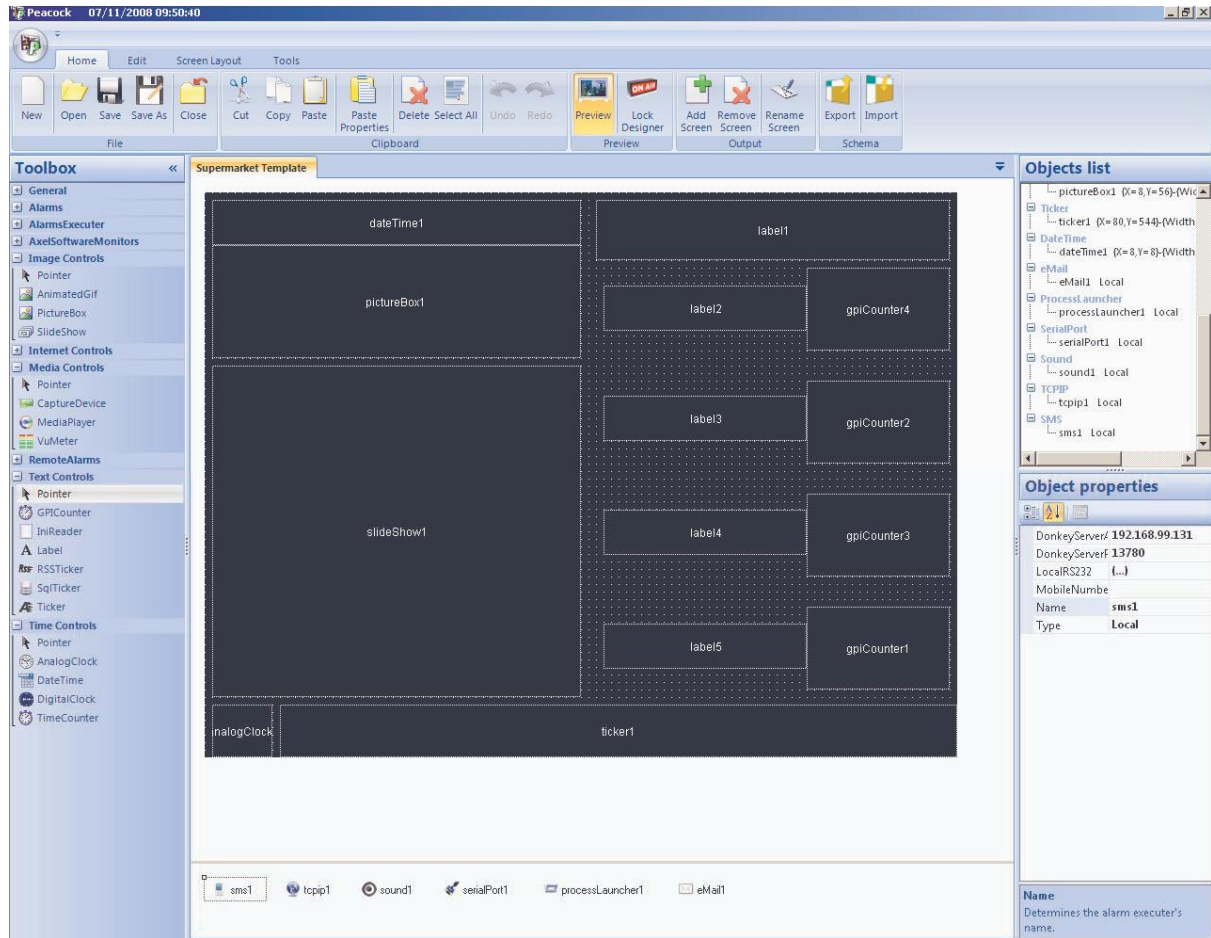
Once you have dragged the item, it will appear in the Objects list too.

 **Note:** every plugIn that has been created all over again by Peacock has default settings that allow its immediate use.

4. Objects management

Peacock's **Objects** are divided into 3 macro categories:

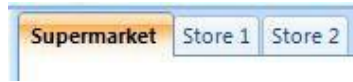
1. **Monitors:** In this category you can manage the work areas that are the project's layouts.
2. **Controls:** This category collects, in subcategories, all the items for creating the layouts.
3. **Components:** This category collects all the items working in background into the project.



4.1 Monitors

4.1.1 Overview

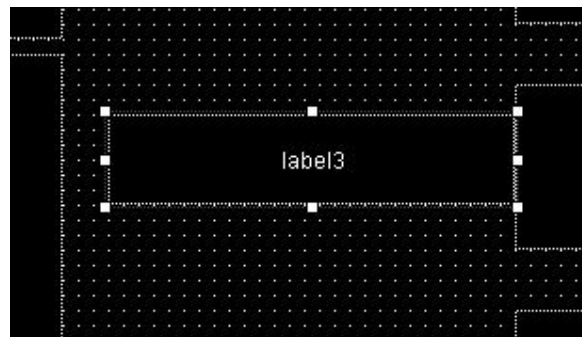
It is possible to add several **Monitors** for each project. They define the work area for the layout.



You can modify the name for each Monitor through the related “Rename Monitor” button (see “The Toolbox” paragraph) or by simply double clicking on the panel’s TAB.



The work area has been designed to make the operations of editing and placing on the objects extremely easy and fast.



This panel allows to select every object through a simple mouse click, showing the related controls that allow its resize. To move an item inside the work area, drag it with the mouse or use the arrow keys of the keyboard.

Keys	“Position” change
<i>Left</i>	Move left
<i>Right</i>	Move right
<i>Up</i>	Move up
<i>Down</i>	Move down
<i>Ctrl+Right</i>	Move right (higher precision)
<i>Ctrl+Left</i>	Move left (higher precision)
<i>Ctrl+Up</i>	Move up (higher precision)
<i>Ctrl+Down</i>	Move down (higher precision)

To resize the selected object, drag the little squares appearing onto the sides and corners of the selection. It is also possible to use the arrow keys together with the Shift key: in this case, the position of the upper left corner will not change.

Keys	“Position” change
Shift+ Left	Decrease width
Shift+ Right	Increase width
Shift+ Up	Decrease height
Shift+ Down	Increase height

4.1.2 Properties

Each panel that we add to the project is provided with several features and properties, that can be displayed and edited through the **Object properties** mask when we select the desired panel.

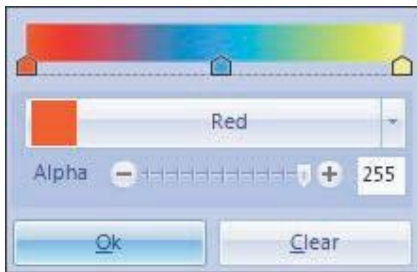
Below, a list of these properties in alphabetical order:

BackColor:

It sets the background color for the selected Monitor.

BackColorBlend:

It sets the mix of colors for the background of the selected Monitor.



BackColorBlendDirection:

It sets the direction of the mix of colors for the background of the selected Monitor.

BackColorType:

If it is set on Solid, the BackColor property is taken into consideration, otherwise, if it is set on Blend, the BackColorBlend property for the background of the selected Monitor is taken into consideration.

BackgroundImage:

It sets the background image for the selected Monitor.

DrawGrid:

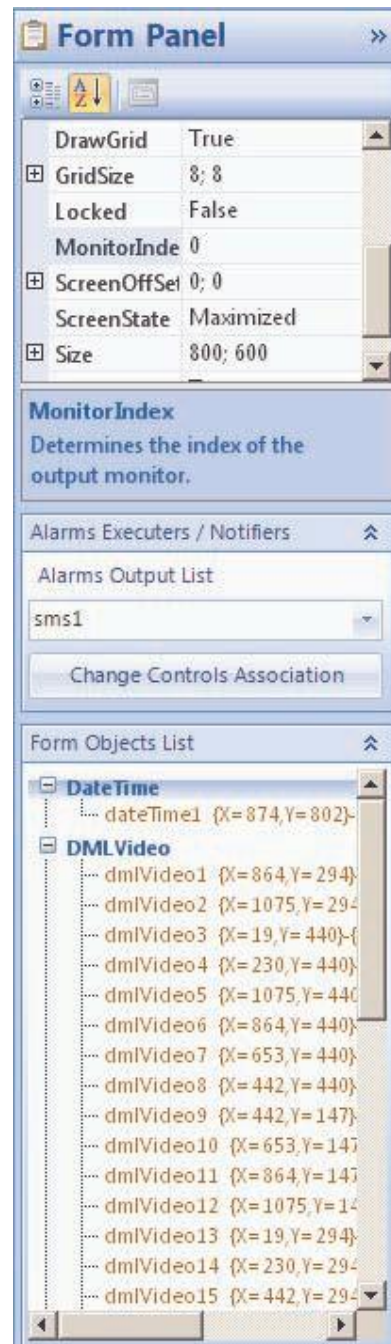
If it is set on True, it determines the grid display on Monitor.

GridSize:

It sets the cells size of the grid (if it is visible) on the selected Monitor.

Locked:

If it is set on True, it determines the objects lock on Monitor. They can not be neither resized or moved.



MonitorIndex:

It sets the output screen for the selected Monitor.

ScreenOffset:

It sets the shift of the output monitor from 0;0 coordinates of the screen. This property takes effect only if ScreenState is in Normal mode.

ScreenState:

If it is set on Normal, the output's size is the one of the Size property. Otherwise, if it is set on Maximized, the form fills the whole output screen.

Size:

It sets the Monitor size.

SnapToGrid

:

If it is set on True, it automatically align the added objects to the Monitor's grid.

4.1.3 Alarms Executors / Notifiers

In every screen we can associate to each Alarm Executor or Alarm Notifier a list of controls, that are on the form, to be associated. In this way, if such controls will get an error, the Alarm Executor or the Alarm Notifier will generate an alarm signal.



4.1.4 Project Panel

In the Project Panel you will find those controls regarding the general project. In this panel it is possible to manage what follows:

1. Audio: it is possible to determine the output audio of the complete project by choosing the desired screen among all the screens added to the project. Obviously, in order to get an output Audio, it is necessary that at least one Audio Manager component has been inserted on a screen. Moreover, with the "Show Audio Object Status" button it is possible to display the status of the Audio control or of the selected Audio component.



2. Alarm Management: It is possible to enable or disable the alarms throughout the project, by the related checkbox. Besides, with the "Show Alarm Controls Status" button it is possible to display the status of the selected Alarm control.



4.2 Controls

4.2.1 Overview

"Controls" objects are divided into the following subcategories:

- Text Controls
- Time Controls
- Image Controls
- Media Controls
- Internet Controls
- Alarms
- RemoteAlarms
- AxelSoftwareMonitors

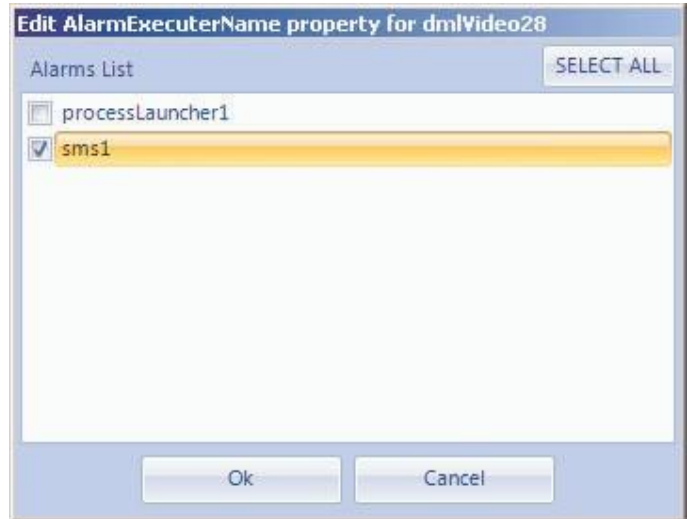
4.2.2 Common Properties

Every item that is inserted into the project is provided with several features and properties that can be displayed and edited through the **Object properties** panel when the desired object is selected.



Note: Usually, if there is the Style property, all the properties related to the ForeColor are not present, and vice versa.

Below, you find a list of these properties in alphabetical order:



AlarmExecuterName

e:

It sets the list of the AlarmsExecuter Components that are into the project and are added to the selected control.

AlarmsCountToRais

e:

It sets the number of times the Control must reach before before launching any alarm through one or more AlarmsExecuter Components linked to the Control itself.

AlphaBackColor

:

It sets the transparency on the background color of the Control.

BackColor

:

It sets the background color of the Control.

BackColorBlen

d:

It sets the background colors mix for the Control.

BackColorBlendDirection:

It sets the direction of the background colors mix for the Control.

BackColorType:

If it is set on Solid, the BackColor property is taken into account; otherwise, if it is set on Blend, the BackColorBlend property is taken into account for the Control background.

Blink:

If it is set on True, the Control will blink with an interval set by the BlinkInterval property.

BlinkInterval:

It sets the Control's blink interval (if the Blink property is set on True).

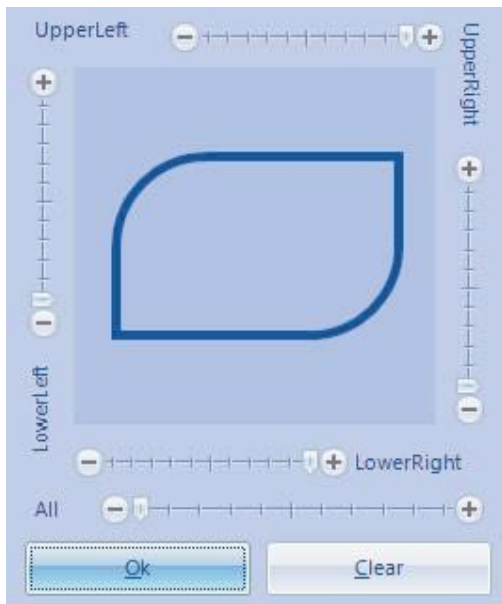
Border:

BlendColors: it sets the background colors mix.



Direction: it sets the direction of the color.

Corners: It sets the corners of the edges in percentage related to the length of the bottom side (values from 0 to 50).



Size: it sets the width of the edge.

Style: it sets the style for the edge's line.

Visible: if it is set on True, the edge is visible.



Index

CheckTimeInMinute (in all categories except **Alarms**, **RemoteAlarms** and **AxelSoftwareMonitors**):

It sets the timer (in minutes) that is used to control the properties liable to changes (such as, for example, the Filename property).

CheckTimeInSecond (only in controls of **Alarms**, **RemoteAlarms** and **AxelSoftwareMonitors** categories):

It sets the timer (in seconds) that is used to control the alarm status.

Font:

It sets the font for the Control.

ForeColor:

It sets the font color for the Control.

ForeColorBlend:

It sets the font color mix for the Control.



ForeColorBlendDirection:

It sets the direction of the font color mix for the Control.

ForeColorType:

If it is set on Solid, the ForeColor property is taken into account; otherwise, if it is set on Blend, the ForeColorBlend property is taken into account.

Location:

It sets the position of the object into the Monitor.

Name:

It sets the Control name.

Size:

It sets the Control size.

Visible:

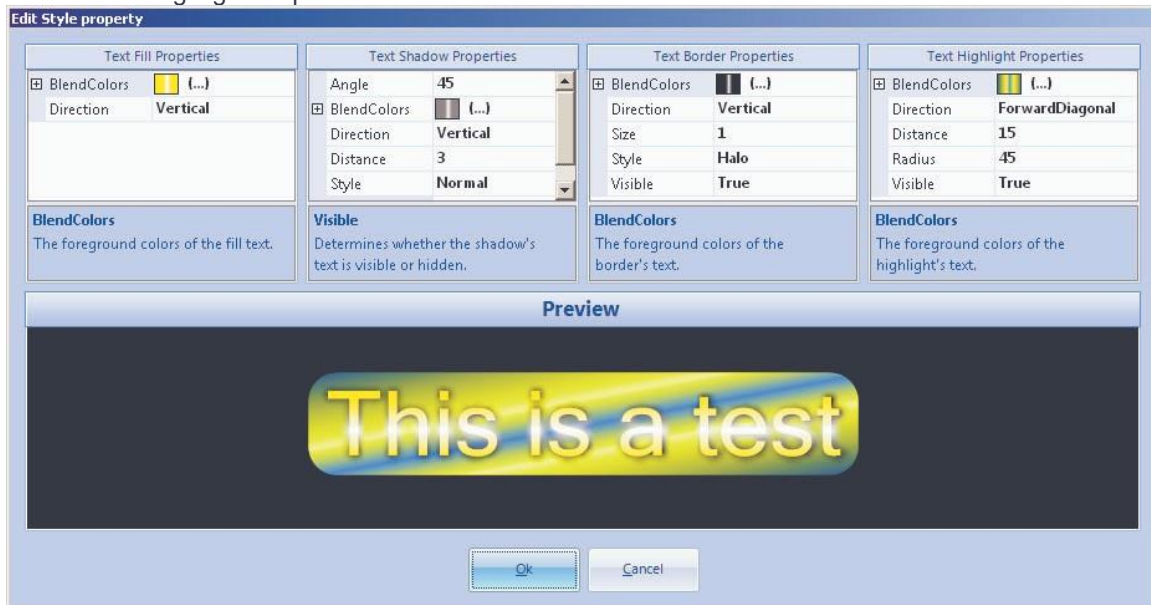
If it is set on True, the Control is visible.

Many Controls within the above mentioned categories have some complex properties in common. Below, such properties are listed:

Style:

It sets the style for the text displayed on the Control. It is possible to set the following parameters:

1. Fill Text Properties
2. Text Shadow Properties
3. Text Border Properties
4. Text Highlight Properties



- *Fill Text Properties*

It sets the colors for the text filling, and their direction:

- BlendColors: it sets the mix of colors for the text filling.
- Direction: it sets the colors' direction.

- *Text Shadow Properties*

It sets the text shadow:

- Angle: it sets the shadow slope related to the text.
- BlendColors: it sets the mix of colors for the text shadow.
- Direction: it sets the colors' direction.
- Distance: it sets the distance between the shadow and the text.
- Style: it sets the shadow style:
 - Normal: it sets the classic style, without any effect.
 - Halo: it sets the style with halo.
- Visible: if it is set on True, the shadow will be displayed.

- *Text Border Properties*

It sets the text border:

- BlendColors: it sets the mix of colors for the text border.
- Direction: it sets the colors direction.
- Size: it sets the size for the text border.
- Style: it sets the style for the text border:
 - Normal: it sets the classic style, without any effect.
 - Halo: it sets the style with halo.
- Visible: if it is set on True, the border is displayed.

It sets the text highlight:

- BlendColors: it sets the mix of colors for the text highlight.
- Direction: it sets the colors direction.
- Distance: it sets the distance between the highlight and the text.
- Radius: it sets the angle radius for the border of the text highlight.
- Visible: if it is set on True, the border is displayed.

4.2.3 Text Controls



4.2.3.1 Overview

The objects of the Text Controls category are the following:

1. **GPICounter**
2. **IniReader**
3. **Label**
4. **RSSTicker**
5. **SqlTicker**
6. **Ticker**

All of these controls determine the display of a text; the text source can be external (a file, a GPI card, etc...) or an inscription, directly inserted by the user, inside a property of the Control.

The Text Control Tickers (Ticker, SqlTicker e RSSTicker), unlike the other Text Controls, are not provided with the Style property and the color can be set according to the following properties:

ForeColor:

It sets the color for the text body.

ForeColorBlend:

It sets the mix of colors for the text.

ForeColorBlendDirection:

it sets the direction of the mix of colors for the text.

ForeColorType:

It shows whether to use the mix of colors or the color that is set in the ForeColor property.

The above mentioned controls will be explained below.

4.2.3.2 GPICounter

The GPICounter control shows a numeric value that is the result of the inputs sent by a GPI card. The displayed value may be increased, decreased and reset to a minimum value.

Below you will find the list of the properties for this control, in alphabetical order:

Align:

It sets the text alignment inside the Control.

Angle:

It sets the bend angle of the text inside the Control.

AutosizeFont:

If it is set on True, the Control's font is resized according to the dimension of the Control itself.

DonkeyServerAddress:

It sets the host of the pc running the Donkey program that controls the GPI card and returns the input for the Control.

DonkeyServerPort:

It sets the port of the pc running the Donkey program that controls the GPI card and returns the input for the Control.

Font:

It sets the font for the text body.

GPICountDown:

Bit: it sets the bit of the port for the GPI card (a value from 0 to 7).

Port: it sets the port of the GPI card (A,B or C).

It sets the listening value on the GPI card for the count down of the value displayed on the Control.

GPICountReset:

Bit: it sets the bit of the port for the GPI card (a value from 0 to 7).

Port: it sets the port of the GPI card (A,B or C).

It sets the listening value on the GPI card for the reset of the value displayed on the Control.

GPICountUP:

Bit: it sets the bit of the port for the GPI card (a value from 0 to 7).

Port: it sets the port of the GPI card (A,B or C).

It sets the listening value on the GPI card for the count up of the value displayed on the Control.

Max:

It sets the maximum value the Control counter can reach.

Min:

It sets the minimum value the Control counter starts from.

Style:

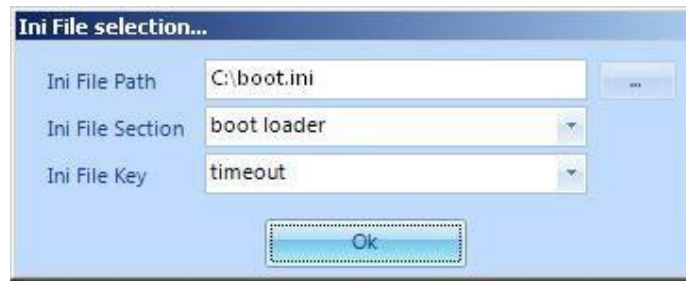
Refer to Common Properties.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
Align	MiddleCenter
AlphaBackColor	255
Angle	0
AutosizeFont	True
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
DonkeyServerAddress	192.168.99.131
DonkeyServerPort	13780
Font	Microsoft Sans
GPICountDown	0, 0
GPICountReset	0, 0
GPICountUp	0, 0
Location	72; 24
Locked	False
Max	100
Min	0
Name	gpiCounter1
Size	200; 50
Style	(...)

AlarmExecuterName
Determines the alarms executer names.

4.2.3.3 IniReader

the IniReader control shows a value extracted from a key of a INI file. To get this value, you must set on the IniFile property the INI file path, the value of the section and of the desired key, as follows:



Below, the list of the properties for this control, in alphabetical order:

Align:

It sets the text alignment inside the Control.

Angle:

It sets the blend angle of the text inside the Control.

AutosizeFont:

If it is set on True, the font of the Control is resized according to the size of the Control itself.

IniFile:

It sets the path for the INI file from which you want to get the value of the required key.

Font:

It sets the font of the text body.

Style:

Refer to the Common Properties.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
Align	MiddleCenter
AlphaBackColor	255
Angle	0
AutosizeFont	True
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
CheckTimeInMinute	1
Font	Microsoft Sans
IniFile	(...)
Location	288; 24
Locked	False
Name	iniReader1
Size	200; 100
Style	(...)
AlarmExecuterName Determines the alarms executer names.	

4.2.3.4 Label

The Label control shows a text that can be taken from an external file or directly set into the TextValue property.

Below, the list of the properties for this control, in alphabetical order:

Align:

It sets the text alignment inside the Control.

Angle:

It sets the blend angle of the text inside the Control.

AutosizeFont:

If it is set on True, the font of the Control is resized according to the size of the Control itself.

FileName:

It sets the path of the file you get the text to display from, only when the Mode property is set on DynamicValue.

Font:

It sets the font of the text body.

LineSeparator:

It sets the text to be used for separating the lines when you want to display that text on a single line.

Mode:

If it is set on StaticValue, the text inserted on the TextValue property is taken into account; otherwise, if it is set on DynamicValue, the text extracted from the file set on the FileName property is taken into account.

Multiline:

If it is set on True, the text is displayed on multiple lines, otherwise it is displayed on a single line.

Style:

Refer to the Common Properties.

TextValue:

It sets the text to be displayed when the Mode property is set on StaticValue.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
Align	MiddleCenter
AlphaBackColor	255
Angle	0
AutosizeFont	True
BackColor	0; 0; 0

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
Align	MiddleCenter
AlphaBackColor	255
Angle	0
AutosizeFont	True
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
CheckTimeInMinute	1
Font	Microsoft Sans
IniFile	(...)
Location	288; 24
Locked	False
Name	iniReader1
Size	200; 100
Style	(...)

AlarmExecuterName
Determines the alarms executer names.

4.2.3.5 RSSTicker

The RSSTicker shows a scrolling text produced from a RSS link.

Below, the list of the properties for this control, in alphabetical order:

Font:

It sets the font of the text body.

LineSeparator:

It sets the text to be used for separating the lines.

Random:

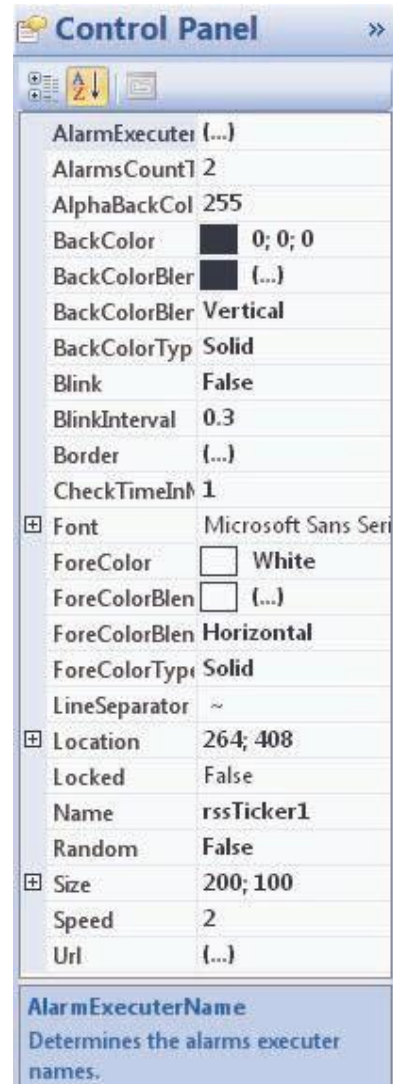
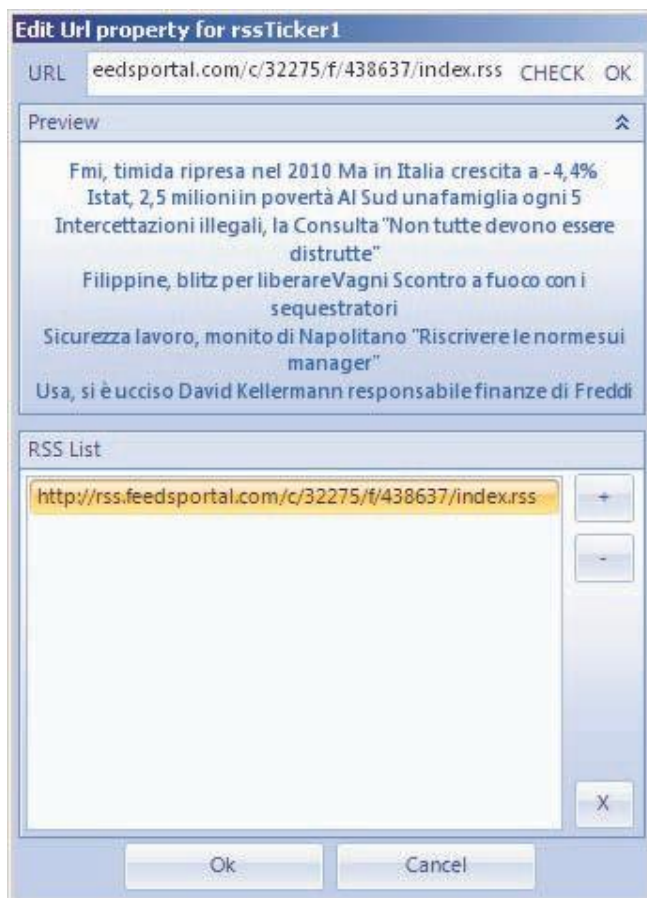
If it is set on True, the URL paths are produced randomly.

Speed:

It sets the speed of the scrolling text. The value is set from 1 to 5.

Url:

It sets a of internet path list of RSS objects from where you want to get the text to be displayed.



4.2.3.6 SQLTicker

The SQLTicker control shows a scrolling text derived from the rows of a table of a database by specifying the column. To get the text, you must set the SQLDatabase property:

- **Server Type:** database type to be set (SqlServer, MySql)
- **Server Name:** name of the server where the database is installed. It is possible to use the Load button that automatically loads the list of servers (of the specified type) that are on the net.
- **Server Port:** Port of the server you want to connect to.
- **Username:** Username to be used for the server access.
- **Password:** Password to be used for the server access.
- **Database Name:** Name of the database. It is possible to use the Load button that automatically loads the list of the databases that are on the selected server.
- **Table Name:** Name of the database table. It is possible to use the Load button that automatically loads the list of the tables that are on the selected database.
- **Column Name (DATABASE):** Name of the table's column. It is possible to use the Load button that automatically loads the list of the columns that are on the selected table.
- **Column Name (SORT):** Name of the sort column. It is possible to use the Load button that automatically loads the list of the columns that are on the selected table.
- **Order By:** Sorting method (ASC o DESC).

To check whether the entered data is correct, you can test the connection using the "Test Connection" button. In case of error, the following message could appear:



Otherwise, the following message will appear:



Below, the list of the properties for this control, in alphabetical order:

Alignment:

It sets the horizontal alignment of the text when the Direction property is set on DownToUp or UpToDown.

Direction:

It sets the scrolling text direction. The available values are: LeftToRight, RightToLeft, DownToUp or UpToDown.

Font:

It sets the font of the text body.

LineSeparator:

It sets the text to be used for separating the lines.

SQLDatabase:

It sets the connection to the database from where you get the scrolling text to be displayed.

Speed:

It sets the scrolling text speed. The value is from 1 to 5.

TimerValue:

It sets the scrolling step.

Object properties >>

AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
Alignment	Center
AlphaBackColor	255
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
CheckTimeInMinute	1
Direction	RightToLeft
Font	Microsoft Sans Seri
ForeColor	White
ForeColorBlend	(...)
ForeColorBlendDirection	Vertical
ForeColorType	Solid
LineSeparator	~
Location	80; 416
Locked	False
Name	sqlTicker1
Size	200; 100
Speed	2
SQLDatabase	(...)
TimerValue	33

AlarmExecuterName
Determines the alarms executer names.

4.2.3.7 Ticker

The Ticker control shows a scrolling text taken from an external file or from an inscription inserted by the user directly on the TextValue property.

Below, the list of the properties for this control, in alphabetical order:

Alignment:

It sets the horizontal alignment of the text when the Direction property is set on DownToUp or UpToDown

Direction:

It sets the scrolling text direction. The available values are: LeftToRight, RightToLeft, DownToUp or UpToDown.

FileName:

It sets the path of the file you get the scrolling text to display from, only when the Mode property is set on DynamicValue.

Font:

It sets the font of the text body.

LineSeparator:

It sets the text to be used for separating the lines.

Mode:

If it is set on StaticValue, the text inserted into the TextValue property is taken into account; otherwise, if it is set on DynamicValue the text extracted from the file set into the FileName property.

Speed:

It sets the scrolling text speed. The value is from 1 to 5.

TextValue:

It sets the text to be displayed when the Mode property is set on StaticValue.

TimerValue:

It sets the scrolling step.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
Alignment	Center
AlphaBackColor	255
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
CheckTimeInMinute	1
Direction	RightToLeft
FileName	
Font	Microsoft Sans Seri
ForeColor	White
ForeColorBlend	(...)
ForeColorBlendDirection	Vertical
ForeColorType	Solid
LineSeparator	~
Location	408; 264
Locked	False
Mode	StaticValue
Name	ticker2
Size	200; 100
Speed	2
TextValue	
TimerValue	33

AlarmExecuterName
Determines the alarms executer names.

4.2.4 Time Controls



4.2.4.1 Overview

The objects in the Time Controls category are the following:

1. AnalogClock
2. DateTime
3. DigitalClock
4. TimeCounter

All of these controls determine the displaying of a date, a clock or a timer.

4.2.4.2 Common Properties

Every item inserted into the project is provided with several features and properties, each one can be displayed and edited through the **Object properties** panel when the desired object is selected.

TimeZoneInformation:

It sets the current hour in the time zone selected from the list below:



4.2.4.3 AnalogClock

The AnalogClock control shows a time through an analog clock.



Below, the list of the properties for this control, in alphabetical order:

SpringEffect:

If it is set on True, the spring effect is activated on the clock seconds hand.

TimeZoneInformation:

Refer to Common Properties.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
AlphaBackColor	0
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
Location	496; 80
Locked	False
Name	analogClock1
Size	130; 130
SpringEffect	True
TimeZoneInformation	(GMT + 1.00 h) An

AlarmExecuterName
Determines the alarms executer names.

The DateTime control shows the current date and time in various formats. In the following examples two different views are shown:

12:09:52

mercoledì 15 ottobre 2008 12:09:52

Below, the list of the properties for this control, in alphabetical order:

Align:

It sets the alignment of the text inside the Control.

Angle:

It sets the blend angle of the text inside the Control.

AutosizeFont:

If it is set on True, the font of the Control is resized according to the size of the Control itself.

DateTimeFormat:

It sets the date view. Several default format types are suggested, and the opportunity to enter a custom format is left to the user.

Font:

It sets the font of the text body.

MillisecondVisible:

If it is set on True, the date's millisecond decimals are displayed.

Style:

Refer to the Common Properties.

TimeZoneInformation:

Refer to the Common Properties.

Properties Object	
AlarmExecuterName	(...)
AlarmsCountToRaiseAlarm	2
Align	MiddleCenter
AlphaBackColor	255
Angle	0
AutosizeFont	True
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
DateTimeFormat	ddd MMM yy
Font	Microsoft San
Location	216; 8
Locked	False
MillisecondsVisible	False
Name	dateTime1
Size	384; 72
Style	(...)
TimeZoneInformation	(GMT + 1.00 h

AlarmExecuterName
Determines the alarms executer names.

4.2.4.5 DigitalClock

The DigitalClock control shows a time through a digital clock.



Below, the list of the properties for this control, in alphabetical order:

ColorFiveSeconds:

It sets the color for the five seconds dots on the clock.

ColorSeconds:

It sets the color for the seconds dots on the clock.

ColorTextSeconds:

It sets the text color for the seconds on the clock.

ColorTextTime:

It sets the time text color (seconds excluded) on the clock.

Font:

It sets the font of the text body.

ImagePath:

It sets the path for the clock image.

ImageVisible:

If it is set on True, the clock image (if it exists) is displayed.

TimeZoneInformation:

Refer to the Common Properties.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
AlphaBackColor	0
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
ColorFiveSeconds	<input type="checkbox"/> White
ColorSeconds	<input checked="" type="checkbox"/> Yellow
ColorTextSeconds	<input type="checkbox"/> White
ColorTextTime	<input type="checkbox"/> White
Font	DS-Digital; 10pt; s
ImagePath	
ImageVisible	True
Location	144; 304
Locked	False
Name	digitalClock1
Size	130; 130
TimeZoneInformation	(GMT + 1.00 h) An

AlarmExecuterName
Determines the alarms executer names.

4.2.4.6 TimeCounter

The TimeCounter control shows a counter that is activated via input by a GPI card. The displayed value can be a countdown or a countup, according to the Direction property setting.

Below, the list of the properties for this control, in alphabetical order:

Align:

It sets the alignment of the text inside the Control.

Angle:

It sets the blend angle of the text inside the Control.

AutosizeFont:

If it is set on True, the font of the Control is resized according to the size of the Control itself.

Direction:

It sets the counter way. The possible values are Down (countdown) or Up (countup).

DonkeyServerAddress:

It sets the host of the pc running the Donkey program that controls the GPI card and returns the input for the Control.

DonkeyServerPort:

It sets the port of the pc running the Donkey program that controls the GPI card and returns the input for the Control.

Font:

It sets the font of the text body.

GPISet:

Bit: it sets the bit of the port for the GPI card (a value from 0 to 7)

Port: it sets the port for the GPI card (A,B or C)

It sets the listening value on the GPI card for the counter reset.

GPISet:

Bit: it sets the bit of the port for the GPI card (a value from 0 to 7)

Port: it sets the port for the GPI card (A,B or C)

It sets the listening value on the GPI card for the counter start.

GPISet:

Bit: it sets the bit of the port for the GPI card (a value from 0 to 7)

Port: it sets the port for the GPI card (A,B or C)

It sets the listening value on the GPI card for the counter stop.

StartTime:

It sets the counter start value.

Style:

Refer to the Common Properties.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
Align	MiddleCenter
AlphaBackColor	255
Angle	0
AutosizeFont	True
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
Direction	Down
DonkeyServerAddress	192.168.99.131
DonkeyServerPort	13780
Font	Microsoft Sans
GPISet	0, 0
GPISet	0, 0
GPISet	0, 0
Location	368; 192
Locked	False
Name	timeCounter1
Size	200; 50
StartTime	00:10:00
Style	(...)

AlarmExecuterName
Determines the alarms executer names.

4.2.5 Image Controls



4.2.5.1 Overview

The objects in the Image Controls category are the following:

1. **AnimatedGif**
2. **PictureBox**
3. **SlideShow**

All of these controls determine the displaying of one ore more images into the Screen.



Note: The file extensions recognized by Peacock are as follows:

- bmp, dib
- JPG, JPEG, JPE JFIF
- GIF
- TIFF, TIF
- PNG

Below, the above mentioned controls will be explained:

4.2.5.2 Ani11ud£dGif

! naledGif cmtrd slol\IS en image file IMth GIF

FileName:

It sets the path for the GIF image to be dspayed.

Object properties

@!-fA	
AlarmExecuterName	
AlarmsCountToRaiseAiarm	2
AlphaBackColor	255
BackColor	— 0;0;0
BackColorBiend	— ◀ ..1
BackColorBiendDirection	Vertic.1l
BackColorType	Solid
Blink	False
BlinlInterval	0.3
Border	(..1
CheckTimeInMinute	1
FileName	
lB Location	368;48
Locked	False
Name	mi
lB Size	200; 1 00
<p>AlarmExecuterName Determines the alarms executer names.</p>	

4.2.5.3 PictureBox

The PictureBox control shows an image file: BMP, JPEG, GIF, TIFF, ecc...

Below, the list of the properties for this control, in alphabetical order:

FileName:

It sets the path for the GIF image to be displayed.

StretchImage:

If it is set on True, the image is resized to the size of the Control containing it, otherwise it keeps the original size.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
AlphaBackColor	255
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
CheckTimeInMinute	1
FileName	
Location	400; 184
Locked	False
Name	pictureBox1
Size	216; 64
StretchImage	False
AlarmExecuterName	
Determines the alarms executer names.	

4.2.5.4 SlideShow

The SlideShow control shows a list of images in a folder that can be set in the Folder property. These images are reproduced in a slideshow, and the playback speed can be set.

Below, the list of the properties for this control, in alphabetical order:

Folder:

It sets the path of the folder containing the images to be played on the slide.

PauseTime:

It sets the pause time between the images playback.

Random:

If it is set on True, the images will be played randomly.

StretchImage:

If it is set on True, the image is resized to the size of the Control containing it, otherwise it keeps the original size.

Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
AlphaBackColor	255
BackColor	255; 255; 255
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
Border	(...)
CheckTimeInMinute	1
Folder	slide
Location	688; 472
Locked	False
Name	slideShow2
PauseTime	5
Random	False
Size	104; 120
Stretch	True

AlarmExecuterName
Determines the alarms executer names.

4.2.6 Internet Controls



4.2.6.1 Overview

The objects in the Internet Controls category are the following:

1. **FlashPlayer**
2. **WebBrowser**

All of these controls determine the displaying of a web page or of a Flash or Flash compatible video project.

Below, the above mentioned controls will be explained:

4.2.6.2 FlashPlayer

The FlashPlayer control shows an object that is compatible with Flash. This object can have the source on internet or on a local file.

FileName:

It sets the path or the internet URL address of the object that is compatible with Flash.

4.2.6.3 WebBrowser

The WebBrowser control shows a web page like it was an image on Monitor.

Below, the list of the properties for this control, in alphabetical order:

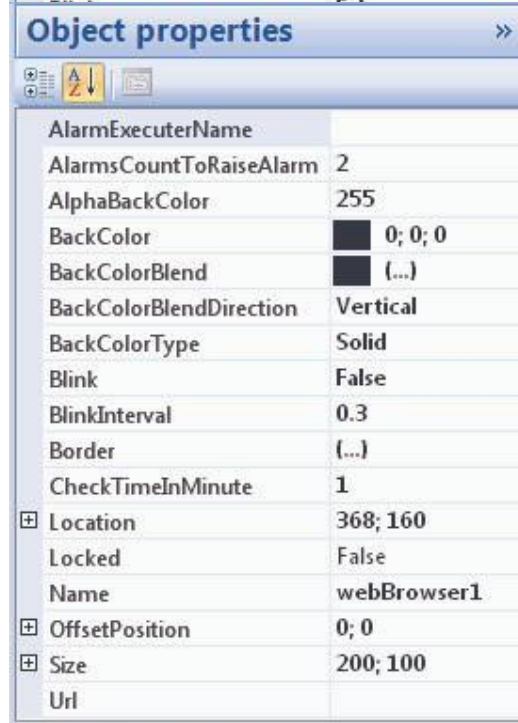
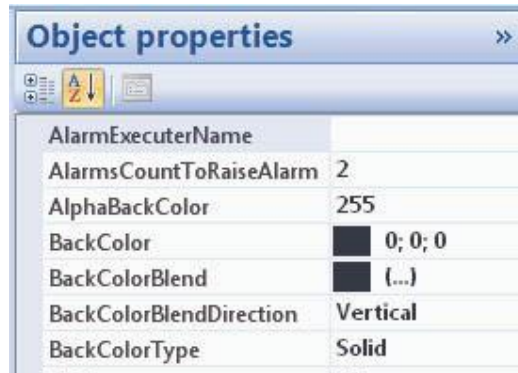
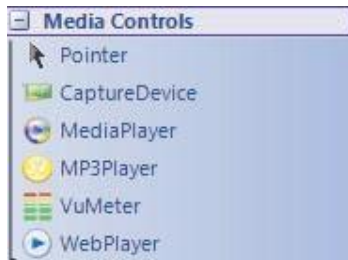
OffsetPosition:

It sets the coordinates for starting to capture the web page as an image for the control.

Url:

It sets the path for the HTML (or IE compatible) file, or the URL address of the web page that you want to display as an image.

4.2.7 Media Controls




4.2.7.1 Overview

The objects in the Media Controls category are the following:

1. **CaptureDevice**
2. **MediaPlayer**
3. **MP3Player**
4. **VuMeter**
5. **WebPlayer**

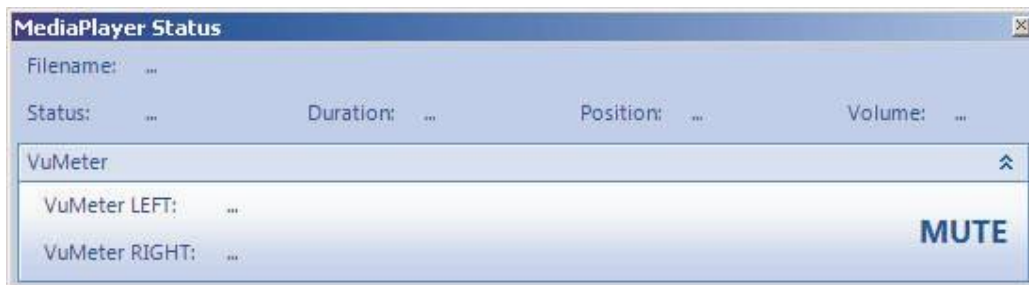
All of these controls determine the displaying of a video stream and of an audio signal on the Monitor.

 **Note:** Some video file extensions cannot be displayed because they are not compatible with the Peacock. For a proper video playback, it is always advisable to install the Peacock packet for the Distributed version (rif. 2.2. Below, you will find the extensions of the files that are not compatible with the system:

- MOV: the MOV files are displayed only if created with Audio Source AAC and Video Source AVC and renamed or exported in .AVI extension (it requires the installation of the Peacock packet for the Distributed version).

Except VuMeter, the other objects create, if not already present, the AudioManager component (refer to 4.3.2.2 section) that determines what are the Media Controls to be enabled the audio to. All the other objects are in Mute mode.

To display the form Status of Media Control you can use the following combination of hot keys: **Ctrl + I**

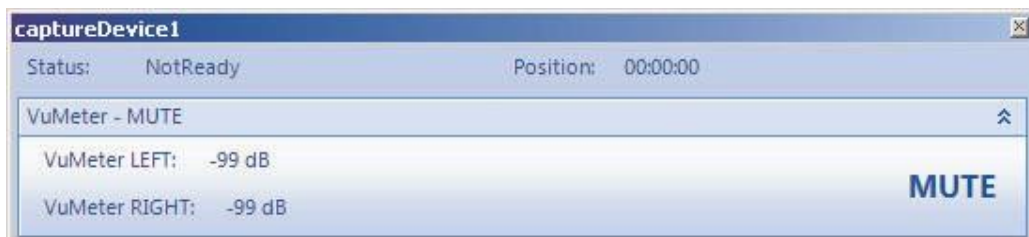


The above mentioned controls will be explained below:

4.2.7.2 CaptureDevice

The CaptureDevice control plays an audio-video stream that is compatible with ffmpeg. This object has a device card as a source.

To display the status of this control you can open the form status through the “Show/Hide Status” button (refer to 3.1.2 section) as shown in figure:



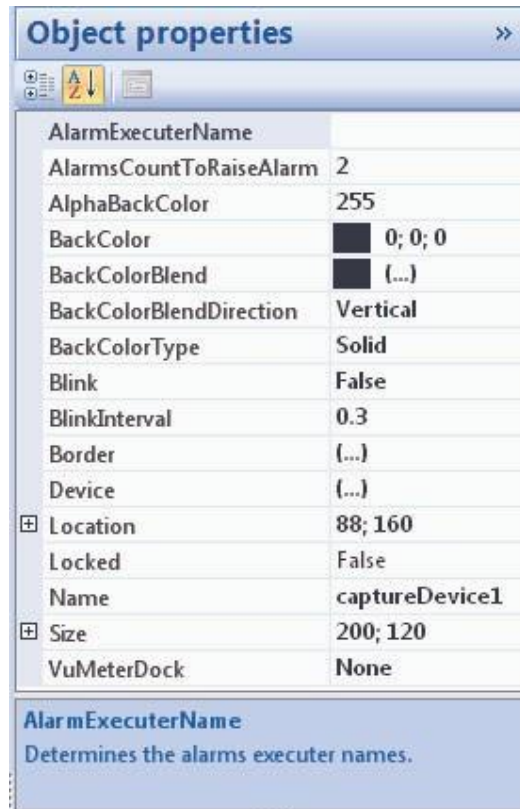
Below, the list of the properties for this control, in alphabetical order:

Device:

It sets a DirectShow compatible capture device through the related selection form.

VuMeterDock:

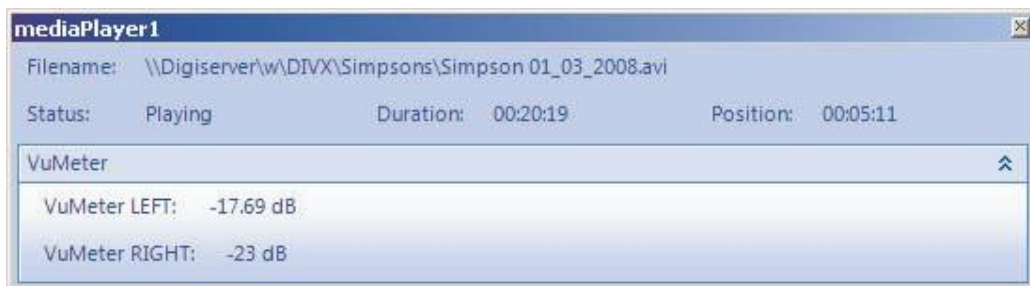
It sets the position of the VuMeter on the control. If this property has None as a value, the VuMeter is not visible.



4.2.7.3 MediaPlayer

The MediaPlayer control plays an audio-video stream that is compatible with ffdshow. This object has a folder as a source.

To display the status of this control you can open the form status through the "Show/Hide Status" button (refer to 3.1.2 section) as shown in figure:



Below, the list of the properties for this control, in alphabetical order:

Folder:

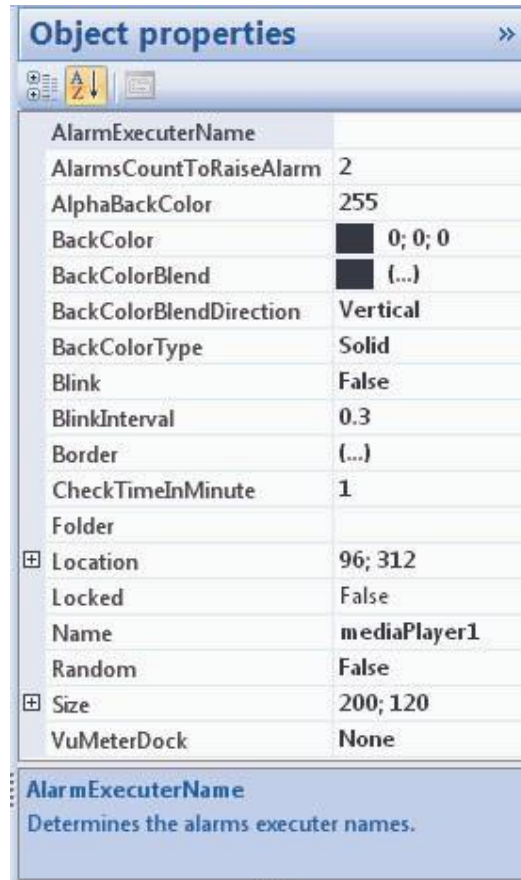
It sets the path of the folder containing the video files that will be played on Media Player.

Random:

If it is set on True, the video files will be played randomly.

VuMeterDock:

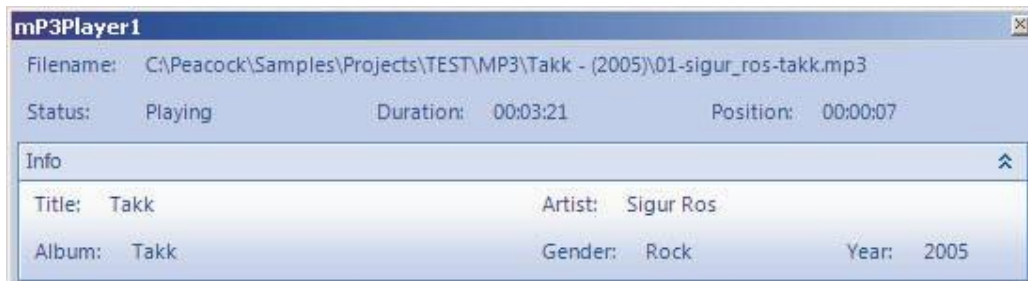
It sets the position of the VuMeter on the control. If this property has None as a value, the VuMeter is not visible.



4.2.7.4 MP3Player

The MP3Player control creates a playlist of MP3 files.

To display the status of this control you can open the form status through the “Show/Hide Status” button (refer to 3.1.2 section) as shown in figure:



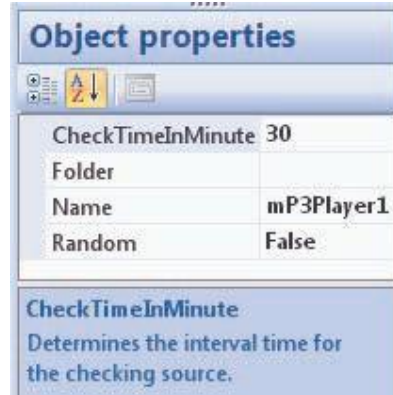
Below, the list of the properties for this control, in alphabetical order:

Folder:

It sets the path of the folder containing the MP3 files that will be played on the MP3Player control.

Random:

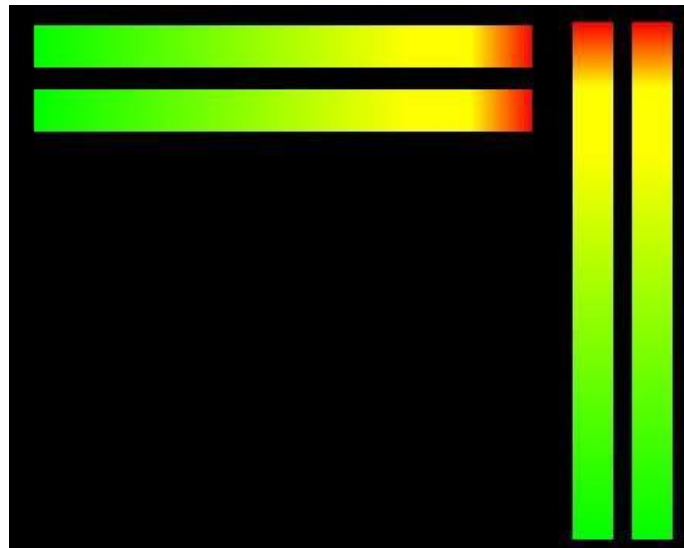
If it is set on True, the MP3 files will be played randomly.



4.2.7.5 VuMeter

The VuMeter control shows the audio level of a DirectShow compatible capture device.

In the following example, two VuMeter objects are displayed (one horizontal and one vertical):



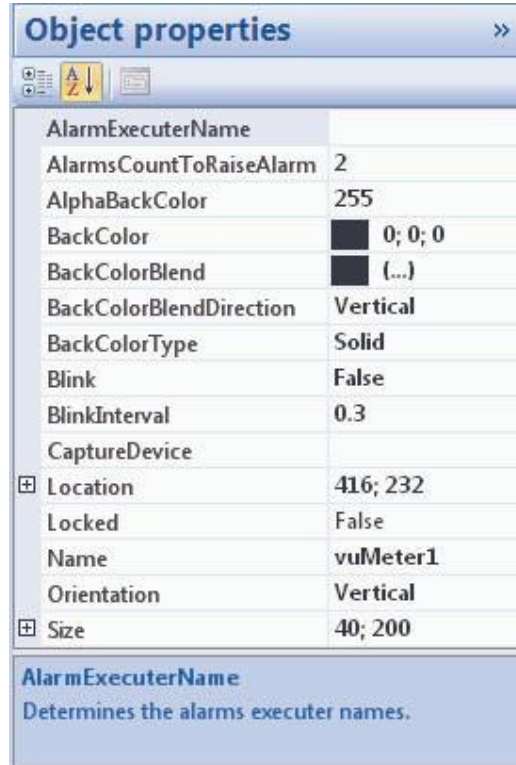
Below, the list of the properties for this control, in alphabetical order:

CaptureDevice:

It sets a DirectShow compatible capture device from the list on the properties.

Orientation:

If it is set on Vertical, the VuMeter is oriented vertically; If it is set on Horizontal, il VuMeter is oriented horizontally.



Object properties	
AlarmExecuterName	
AlarmsCountToRaiseAlarm	2
AlphaBackColor	255
BackColor	0; 0; 0
BackColorBlend	(...)
BackColorBlendDirection	Vertical
BackColorType	Solid
Blink	False
BlinkInterval	0.3
CaptureDevice	
Location	416; 232
Locked	False
Name	vuMeter1
Orientation	Vertical
Size	40; 200

AlarmExecuterName
Determines the alarms executer names.

4.2.7.6 WebPlayer

The WebPiayer cmtrd shoi'IS en aJdio-IAdeo stream that is based on Windoi'IS MediaPlayer object.

URL

It sets the URL path for the aJdio/IAdeo stream to be played through the cmtrd.

Object properties



AlarmExecuterName
AlarmsCountToRaiseAalarm 2

AlphaBackColor BackColor BackColorBiend
BackColorBiendDirection BackColorType
Blink
BlinkInterval
Border
1B Location Locked Name
1B Size
URL
AlarmExecuterName

Index
255
0;0;0
0..1
Vertic.11 Solid F.11se
0..3
(..1
240;480
False webPl.tyer1
200; 1 00

Determines the alarms executer names.

4.2.8 AxelSoftwareMonitors



4.2.8.1 Overview

The objects in the AxelSoftwareMonitors category are the following:

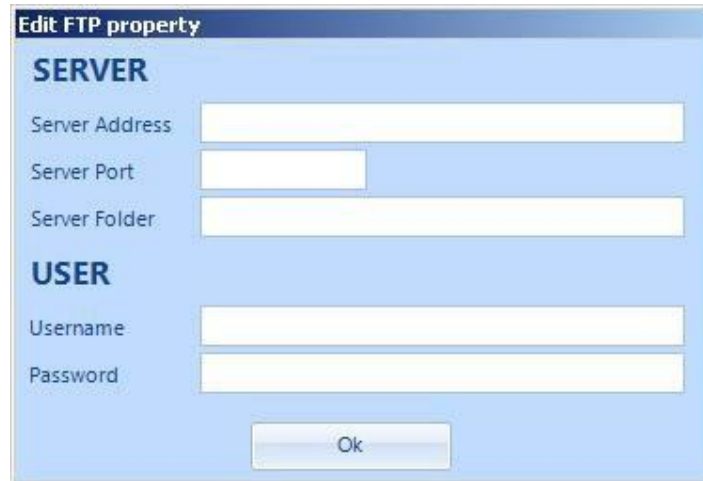
1. DJPro
2. DMLServer
3. DMLVideo
4. MultiPing
5. Storage
6. XPlayout

All of these controls determine the monitoring of a server or or of a program that is external to the Peacock.

4.2.8.2 Common Properties

FTP:

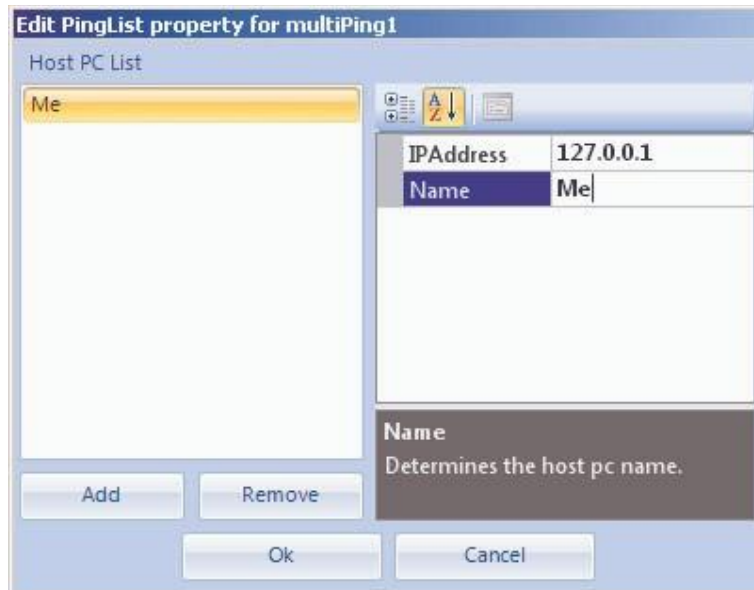
It sets the parameters for the FTP connection:



- Server Address: it sets the address for the FTP server.
- Server Port: it sets the port for the FTP server.
- Server Folder: it sets the desired folder on the FTP server.
- Username: it sets the username for the access on the server.
- Password: sets the password for the access on the server.

PingList:

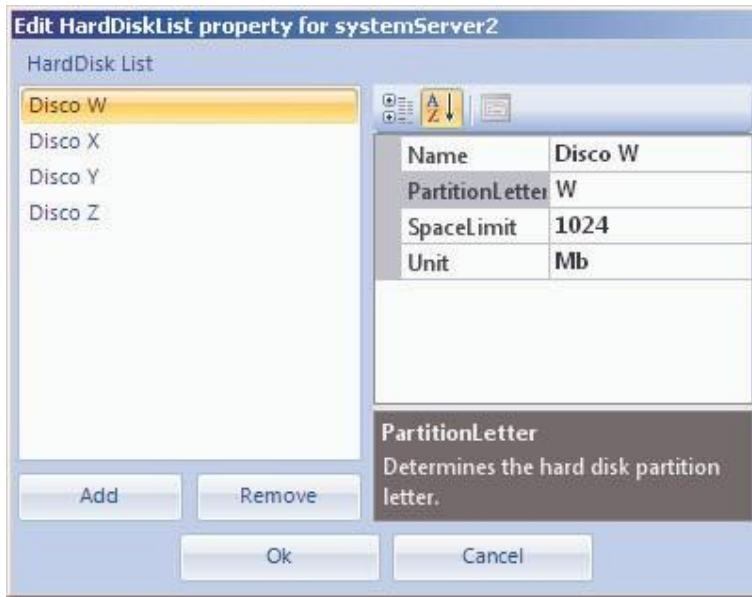
It sets the parameters for the host list to check through ping:



- IPAddress: it sets the IP address to check through ping.
- Name: it sets the name to be shown on the list.

HardDiskList:

It sets the parameters for the list of the shared pc partitions:




- Name: it sets the name to be shown on the list.
- PartitionLetter: it sets the letter for the shared partition.
- SpaceLimit: it sets the space limit beyond which the Hard Disk is in error.
- Unit: it sets the unit of measure referring to the size set in the SpaceLimit property.

Folders:

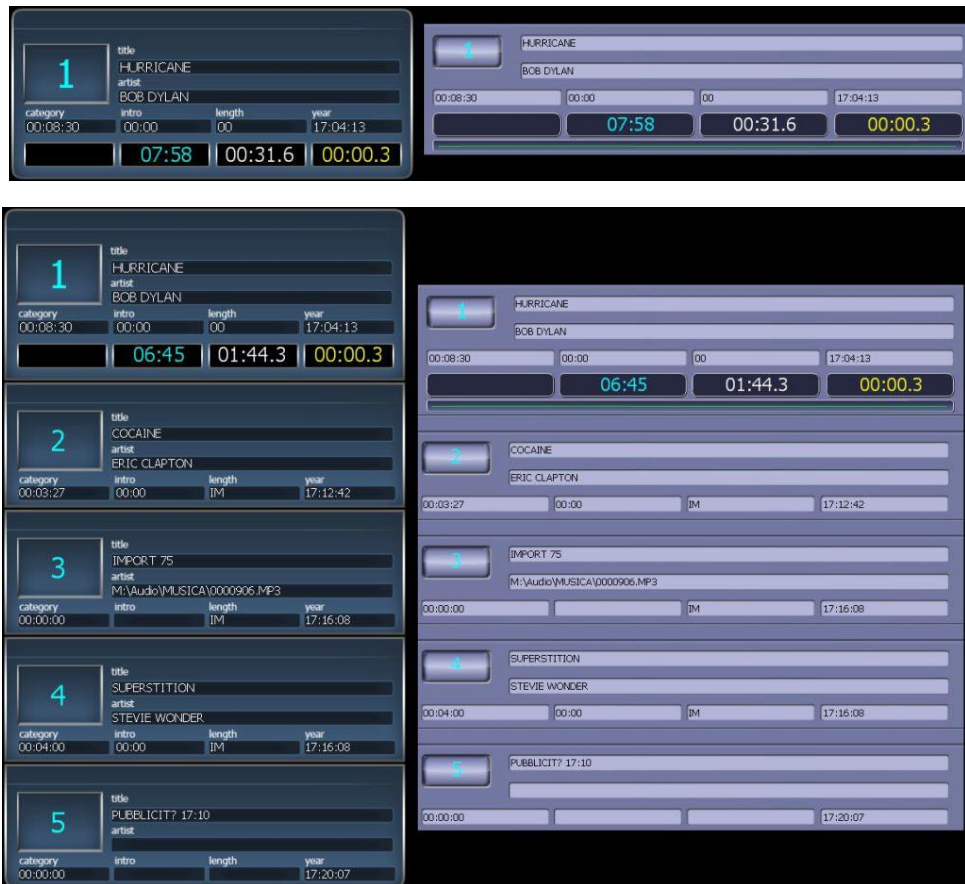
It sets the parameters for the folder list to be checked on the remote pc:



You can enter the network path through the related button  or manually. Then it is possible to determine the folders sorting through the PLUS and MINUS buttons.

The above mentioned controls will be explained below:

The DJPro control monitors the status of DJ Pro External Terminal. In the following images this control is displayed in OnAir mode (both Peacock and Classic) and in Complete mode (both Peacock and Classic):



Below, the list of the properties for this control, in alphabetical order:

Server:

It sets the host name for the server where the DJPro Terminal is running.

ServerPort:

It sets the port for the server where the DJPro Terminal is running.

ViewMode:

It sets the view mode for the control, it can be Complete (that is the list of the 5 songs into the playlist) or OnAir (that is only the first song).

ViewStyle:

It sets the graphic style for the control that can be Peacock or Classic.

Object properties

AlarmExecuter	
Blink	False
BlinkInterval	0.3
CheckTimeInS	0.8
Location	24; 48
Locked	False
Name	djPro1
Server	
ServerPort	0
Size	240; 404
ViewMode	Complete
ViewStyle	Peacock

AlarmExecuterName
Determines the alarms executer names.

4.2.8.4 DMLServer

The DMLServer control monitors the status of a server through the Ping control, the date control, the control of the available dimension on the server partitions and the control of date-time of the most recent file contained into the folders set by the user.

In the following example this control is displayed in RealTime:



Below, the list of the properties for this control, in alphabetical order:

AlarmDMLServer:

It sets, for each server alarm, the enabling for the error notification and the number of times after which the alarm is started:

- General: referred to the failed connection to the server, or to any other errors.
- Ping.
- Time + TimeDiff: time in seconds after which the alarm is reported.
- HardDisks.
- Folder + TimeDiff: time in hours after which the alarm is reported.



Folders:

It sets the list of the server folders where you must control the time of the most recent file contained in them.

HardDiskList:

It sets the server partitions and the minimum free space limit beyond which the alarm will run. You can add a maximum of four partitions.

Labelcolor:

It sets the labels color for the control.

PeacockSpyPort:

It sets the port for the server where the Peacock Spy is running, which returns the status of partitions and of the date. The Peacock Spy must be run on the server you want to monitor.

PeacockSpyUsed:

it determines whether to use or not the Peacock Spy.

RemotePcName:

It sets the host name of the server you want to monitor.

Titlecolor:

It sets the title color for the control.



The DMLVideo control monitors the Axel DML status.

In the following example this control is displayed in OnlyImage and Complete modes:



Below, the list of the properties for this control, in alphabetical order:

AlarmDML:

It sets, for each DML alarm, the enabling to error notification and the number of times beyond which the alarm will be run:

- General: Referred to the failed connection to the DML program or to any other general errors.
- Status.
- Video.
- Audio.
- Time + TimeDiff: time in seconds after which the alarm is reported.
- IDR.
- ADR.
- Drop Frames.



Connection:

If it is set on LAN, the connection type is via LAN, otherwise if it is set on FTP, the connection type is via FTP.

FTP:

(refer to 4.2.8.2 section).

ShowChannelName:

If it is set on True, it displays the control title.

StatusFolder:

It sets the path on the folder containing the Status.txt file where the data associated to the control alarms are written.

ViewMode:

It sets the view mode of the control, it can be Complete (that is with image and alarms) or OnlyImage (that is only with image).

Titlecolor:

It sets the title color for the control.



4.2.8.6 MultiPing

The MultiPing control monitors the status of a maximum of 10 hosts printers, pc, servers, routers, ecc...) through the Ping control.

In the following example this control is displayed in RealTime:



Below, the list of the properties for this control, in alphabetical order:

AlarmMultiPing:

It sets, for each server alarm, the enable to error notification and the number of times after which the alarm is started:

- Ping.



Caption:

It sets the control title.

Labelcolor:

It sets the labels color for the control.

PingList:

It sets the IP address for the hosts to be controlled through ping.

Titlecolor:

It sets the title color for the control.

It sets the port of the server where the Peacock Spy that returns the status of partitions and the date is running. Peacock Spy must run on the server you want to monitor.

PeacockSpyUsed:

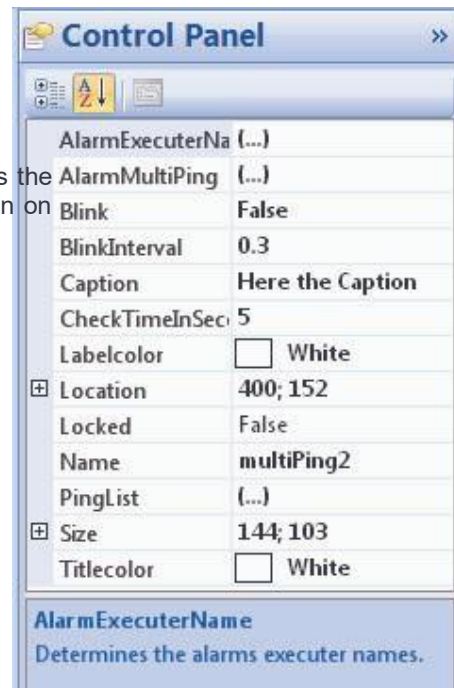
It determines whether to use the Peacock Spy or not.

RemotePcName:

It sets the host name for the server you want to monitor.

Titlecolor:

It sets the title color for the control.



4.2.8.8 XPlayout

the Xplayout control monitors the status of Axel Xplayout.

In the following example this control is displayed in RealTime:



Below, the list of the properties for this control, in alphabetical order:

AlarmXPlayout:

It sets, for each XPlayout alarm, the enable to error notification and the number of times after which the alarm is started:

- General: referred to the failed connection to XPlayout, or to any other errors.
- Status.
- IP Address.
- Current File.
- Next File.
- Delay.
- Position.
- Remain.



Connection:

If it is set on LAN, the connection type is via LAN, otherwise if it is set on FTP, the connection type is via FTP.

FTP:

(refer to 4.2.8.2 section).

Labelcolor:

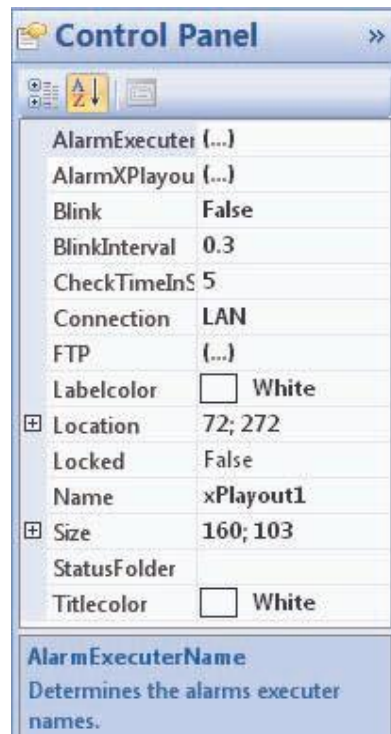
It sets the character color for the control (red is not available).

StatusFolder:

It sets the path to the folder containing the XtvStatus.txt file where the data associated to the control alarms are written.

Titlecolor:

It sets the title color for the control.



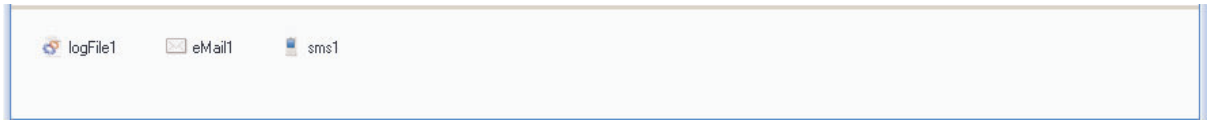
4.3 Components

the Components objects are divided into the following subcategories:

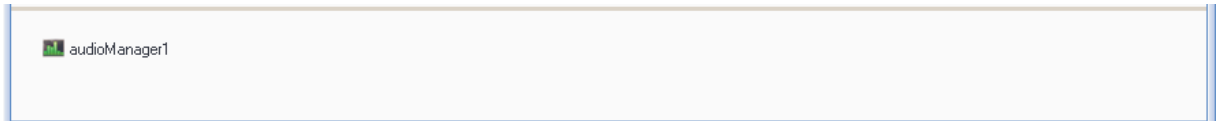
□ AlarmsExecuter



□ AlarmsNotifiers



□ Components



Unlike the Controls, the Components are objects working in background. They are associated to Controls or Monitors.

Moreover, the Components are displayed on the bottom of the Monitor and not inside, because they are not displayed in RealTime.

Every Monitor can contain several Components which, in turn, can be associated to all the Controls inserted into the concerned Monitor.

4.3.1 Alarms Executors



4.3.1.1 Overview

The objects in the Alarms Executors category are the following:

1. **ProcessLauncher**
2. **SerialPort**
3. **Sound**
4. **TCPIP**

All of these components determine an alarm run if the control that is associated to them will change its own status by generating an error.

These components are associated to controls by a common property, **AlarmExecuterName** (see section 4.2.2), and you can associate several AlarmsExecuters to each control.

Usually, the Controls that are associated to these components are in the **AxelSoftwareMonirtos**, **Alarms** and **RemoteAlarms** subcategories.

The alarm can be run locally or remotely by the Donkey program (its parameters can be set for each Alarms Executer).

4.3.1.2 Common Properties

Each item in the project is provided with several features and properties, all viewable and editable through the **Object properties** panel when you select the desired object.

Below in alphabetical order, a list of this common properties:

DonkeyServerAddress:

It sets the IP address or the Host name of the server where the Donkey program is running.

DonkeyServerPort:

It sets the port of the server where the Donkey program is running.

Name:

It sets the component's name.

4.3.1.3 ProcessLauncher

the ProcessLauncher component, if used, creates a new process according to the statements set on the ProcessParameters and ProcessPath properties. This control works only in remote through Donkey:

Below, the list of the properties for this component, in alphabetical order:

DonkeyServerAddress:

(refer to section 4.3.1.2).

DonkeyServerPort:

(refer to section 4.3.1.2).

Name:

(refer to section 4.3.1.2).

ProcessIsClosable:

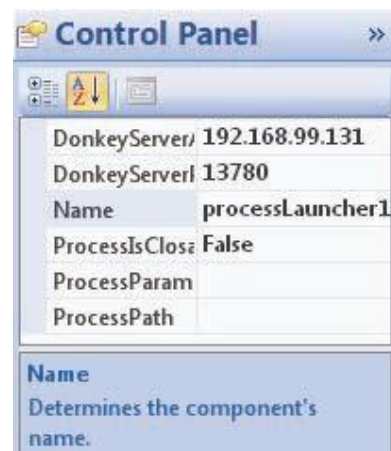
If it is set on True, the process, once the associated control is no longer in error, is closed; otherwise, if it is set on False, the process remains open regardless of the control status.

ProcessParameters:

It sets any parameters to be associated to the process.

ProcessPath:

It sets the path of the process you want to run. It is also possible to choose a non executable file: Windows will automatically load up the file according to its specifics.



4.3.1.4 SerialPort

The SerialPort component, if used, sends a serial comand, that can be set through theMessage property, to a serial port that can be local or remote. This control works only locally. To set the local port, you need to set the LocalRS232 property:



Below, the list of the properties for this component, in alphabetical order:

DonkeyPortNumber:

It sets the remote serial port. Specify the number of COM (es: 1 is for COM1).

DonkeyServerAddress:

(refer to section 4.3.1.2).

DonkeyServerPort:

(refer to section 4.3.1.2).

LocalRS232:

It sets the properties for the local serial port.

Message:

It sets the message to be sent to the serial port.

Name:

(refer to section 4.3.1.2).



4.3.1.5 Sound

The Sound component, if used, plays a sound through wav file. When you use this component you cannot run the remote alarm, but only the local one. This control works only locally.

Below, the list of the properties for this component, in alphabetical order:

LoopMode:

If it is set on True, the sound is played in loop.

Name:

(refer to section 4.3.1.2).

SoundPath:

It set the absolute path of the wav file that you want to use as an alarm.

TimeOut:

It sets the time length for the sound to be played.



4.3.1.6 TCP IP

The TCP/IP component, if used, sends a serial comand, that can be set on the theMessage property, through TCP/IP protocol to the specified address. This control works only in remote through Donkey.

Below, the list of the properties for this component, in alphabetical order:

DonkeyServerAddress:

(refer to section 4.3.1.2).

DonkeyServerPort:

(refer to section 4.3.1.2).

IPAddress:

It sets the IP address to send the alarm message to.

Message:

It sets the message to be sent to TCP/IP.

Name:

(refer to section 4.3.1.2).

Port:

It sets the port to send the alarm message to.



4.3.2 Alarms Notifiers



4.3.2.1 Overview

The objects in the AlarmsExecuter category are the following:

1. **LogFile**
2. **eMail**
3. **SMS**

All of these components will run an alarm if the associated component will change its status, causing an error.

These components are associated to the controls through the **AlarmExecuterName** common property (see section 4.2.2); it is possible to associate several AlarmsExecuters for each control.

Usually the Controls that are associated to these components are shown in the **AxelSoftwareMonirtos**, **Alarms** and **RemoteAlarms** subcategories.

The alarm can be run locally or remotely through the Donkey program (its parameters can be set for each AlarmsExecuter).

4.3.2.2 Common Properties

Every item inserted into the project is provided with several features and properties that can be displayed and edited through the **Object properties** panel when you select the desired object.

Below, a list of these common properties, in alphabetical order:

DonkeyServerAddress:

It sets the IP address or the Host name of the server where the Donkey program is running.

DonkeyServerPort:

It sets the porta of the server where the Donkey program is running.

Name:

It sets the name of the component.

4.3.2.3 LogFile

The LogFile component, if used, creates or modifies a user defined log file. This control works only locally.

Below, the list of the properties for this component, in alphabetical order:

DonkeyServerAddress:

(refer to section 4.3.1.2).

DonkeyServerPort:

(refer to section 4.3.1.2).

Filename:

It sets the name of the file (without path) that you want to create for the error log.

Name:

(refer to section 4.3.1.2).



Index
4.3.2.4 eMail

The eMail component, if used, sends an email according to the settings in the Email property. This control works only in remote through Donkey:



- **From:** the email sender.
- **To:** the email recipients. You have to list the recipients by separating them with a new line.
- **SMTP Host:** SMTP Host of the mail server that you want to use for sending emails.
- **SMTP Port:** SMTP port number of the mail server that you want to use for sending emails.
- **Username:** Username to be used for accessing the server.
- **Password:** Password to be used for accessing the server.

Below, the list of the properties for this component, in alphabetical order:

DonkeyServerAddress:
(refer to section 4.3.1.2).

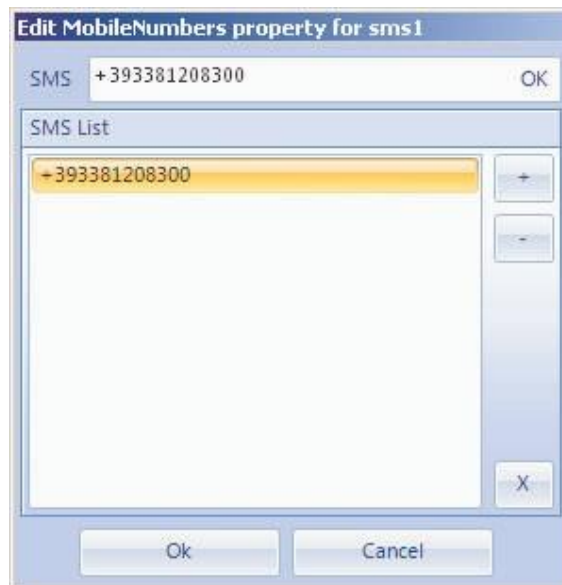
DonkeyServerPort:
(refer to section 4.3.1.2).

Email:
It sets the parameters needed for the email send.

Name:
(refer to section 4.3.1.2).



the SMS component, if used, sends a SMS through serial port to all the mobile numbers that are set in the MobileNumbers property. This control works only in remote through Donkey:



Below, the list of the properties for this component, in alphabetical order:

DonkeyServerAddress:
(refer to section 4.3.1.2).

DonkeyServerPort:
(refer to section 4.3.1.2).

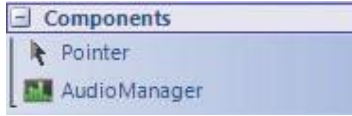
LocalRS232:
It sets the properties for the local serial port.

MobileNumbers:
It sets the list of mobile numbers you want to send the alarm to.

Name:
(refer to section 4.3.1.2).



4.3.3 Components



4.3.3.1 Overview

The objects in the Components category are the following:

1. AudioManager

All of these components determine the management of the controls in the Peacock's project.

Usually the Controls that are associated to these components are shown in the **MediaControls** subcategory.

4.3.3.2 AudioManager

The AudioManager component is automatically inserted, if not already present, when entering an object that is in Media Controls.

Below, the list of the properties for this component, in alphabetical order:

ActiveControl:

It sets the Media Controls control that must be audio-enabled. Consequently, all other controls have the sound turned off.

Name:

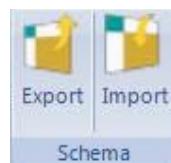
(refer to section 4.3.1.2).



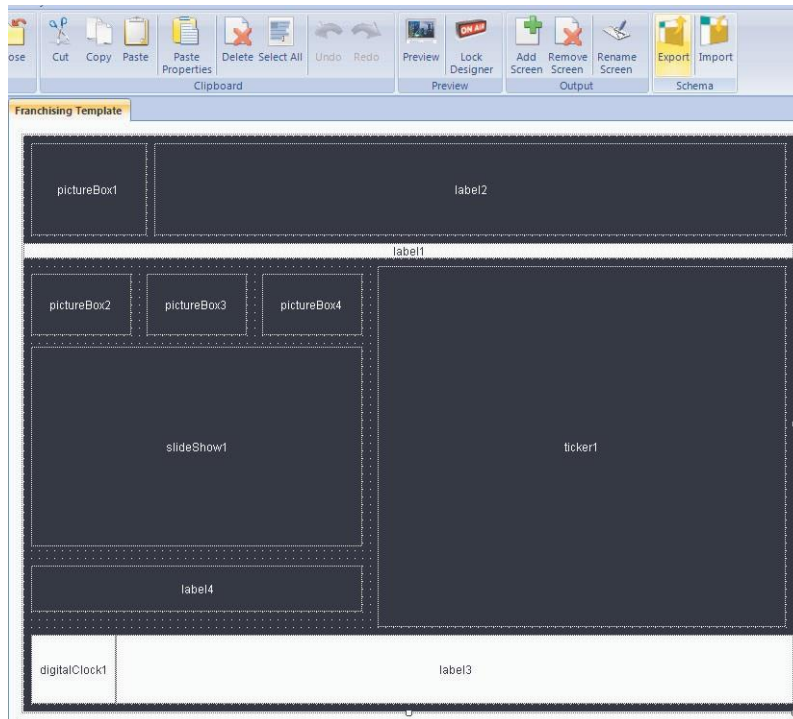
5. HowTo

5.1 Export / Import of a layout

You can create and load the layouts through the Export and Import buttons that are in the Schema section Peacock Toolbar:



After creating a layout on a monitor, click on the Export button:

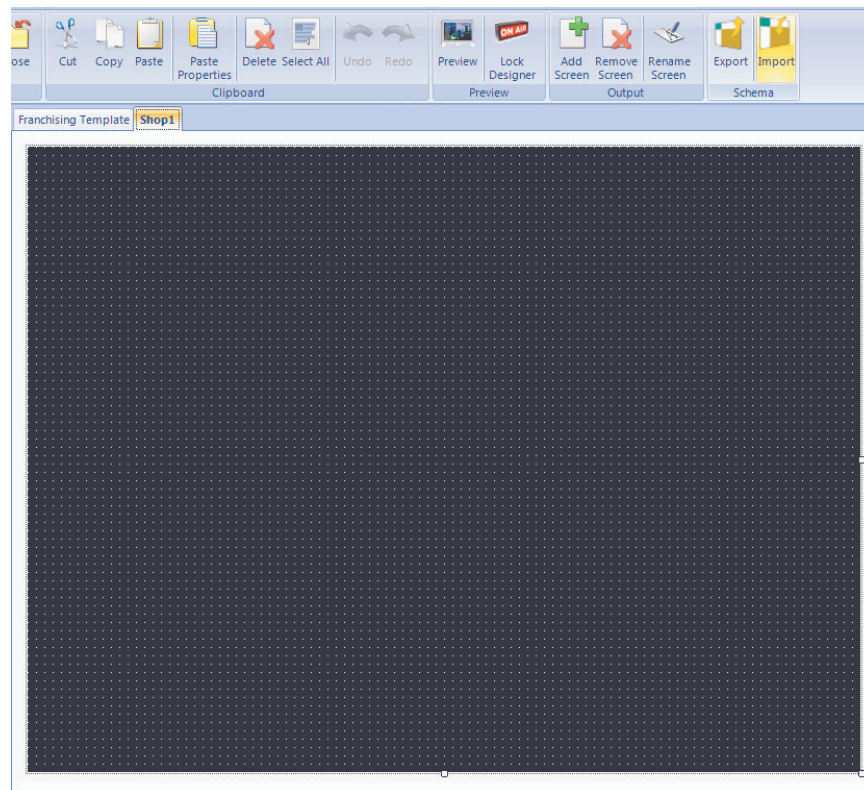


The following window will appear:

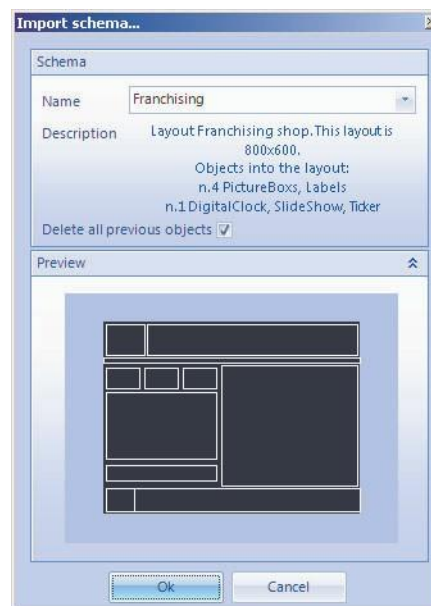


Index

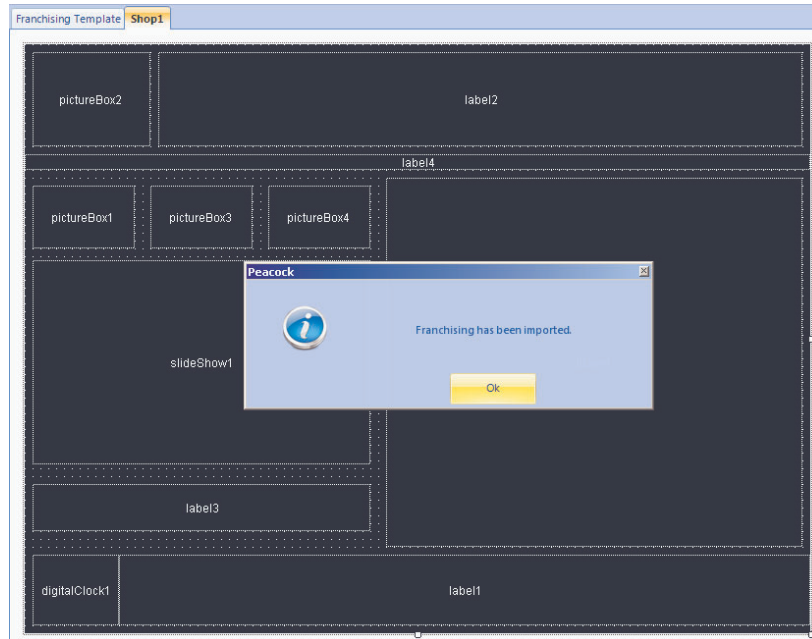
You can determine the name and a short description of the Layout you want to export. Then, add a new screen and name it “Shop1”:



Click on the Import button and select the Schema that you just exported. Note that the description and a preview for the layout of the selected schema are reported:



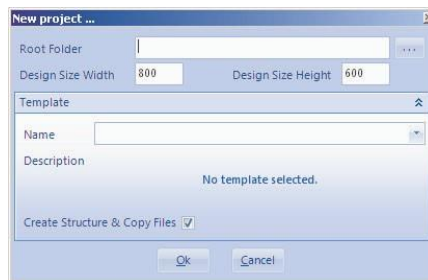
Note the “Delete all previous objects” checkbox that determines the deletion of all the possible objects on the monitor you want to export the schema on. Otherwise, if you uncheck it, the objects on the monitor are added to the ones of the schema.



Remember that only the Location and Size properties of all the objects in the user defined layouts are exported. Consequently, any inserted texts or linked images are exported because they are not relevant for the layout.

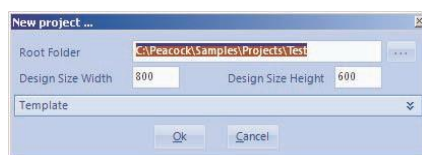
5.2 Creating a project

The creation of a project in Peacock is simple and intuitive. You have to choose the Root Folder, that is the path where you want to save the project you are creating; the Design Monitor size (height and width), keeping in mind that the project will be resized according to the resolution of the monitor that will be used for the Preview; possibly, choose a Template, that is the opening of a Peacock's default project containing some controls that are already set. Below, the mask for the creation of a new project:

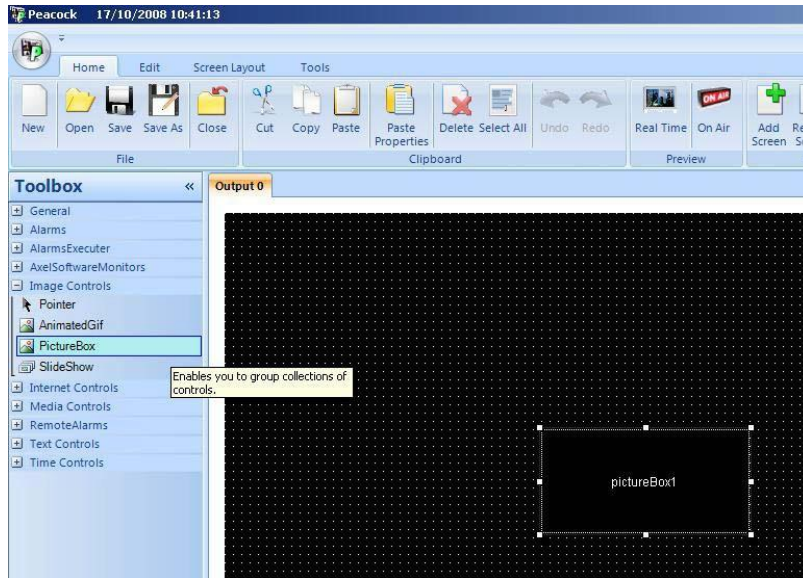


The project's creation and the related compile of the executable file are based on the concept of path relativity. It is important to remember that all the external objects, images and texts linked to Peacock's controls should be stored in subfolders of the Root Folder, thus creating a structural stability that allows the movement of the whole project to any other pc without altering its functionality

For example, select "C:\Peacock\Samples\Projects\Test" as Root Folder and click on Ok:

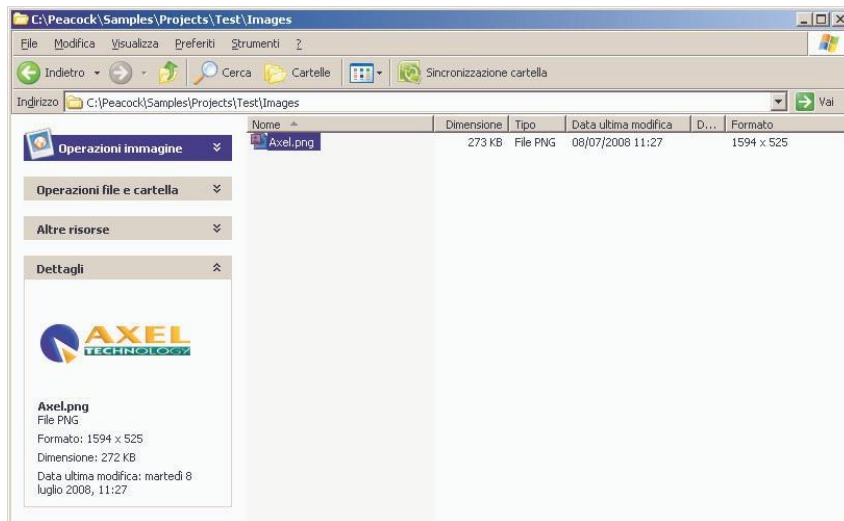


Add a PictureBox control into the Monitor, as it is shown in figure:



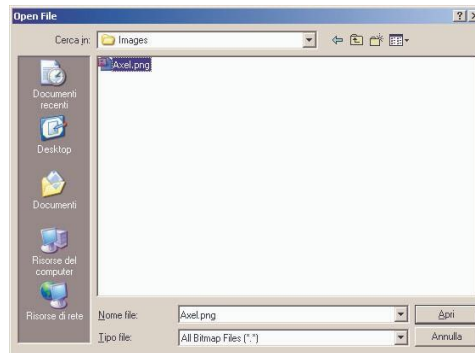
Create a Images subfolder into the Root Folder.

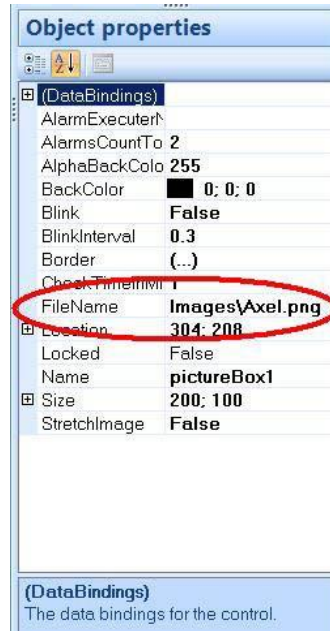
Now, copy a picture (e.g.: "Axel.png") from the pc and paste it into the subfolder you have just created ("C:\Peacock\Samples\Projects\Test\Images"):



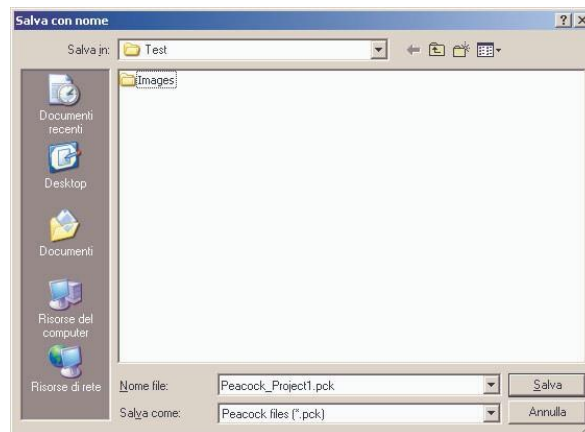
Go back to the project and set the FileName property of the pictureBox1 control with the related path of "Axel.png" file. It is possible to get help through the related dialog box ("File Open"):

Note that, on the FileName property of the PictureBox, automatically appears the relative path ("Images\Axel.png"):





Finally, save the project in the Root Folder previously set, using the “Save” button (note that the path suggested by the dialog box is the Root Folder):

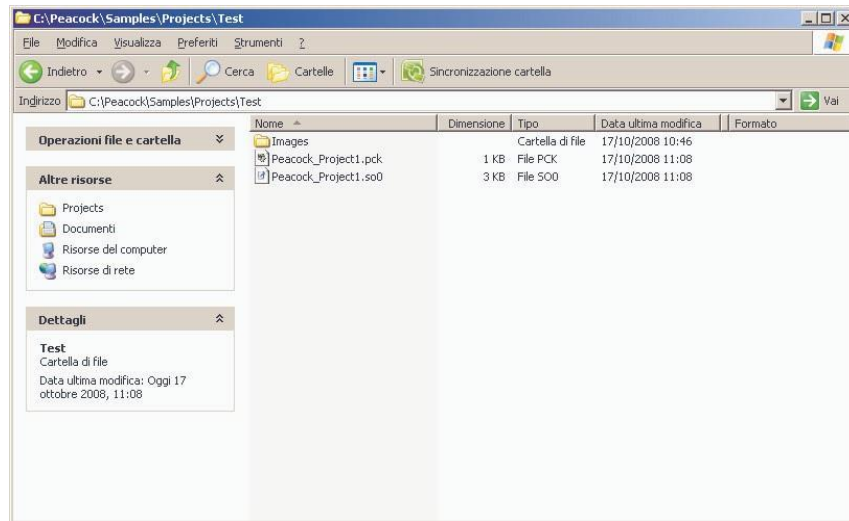


If there were no errors, the project is saved correctly:



It is important to remember that, following this procedure, the relativity of the project has been kept as to the Root Folder, so if you move the Root Folder it can be used in any pc path.

For example, after closing the current project on Peacock by the “Close” button, return to “C:\Peacock\Samples\Projects\Test” folder and you should find a situation like the following:



Move the “C:\Peacock\Samples\Projects\Test” folder onto the Desktop, thus changing the absolute path of the folder itself in “C:\Documents and settings\name.surname\Desktop\Test”.

Return to Peacock and click on the “Open” button, select the new path of the project (“C:\Documents and settings\nome.cognome\Desktop\Test”) and open the .pck file.

Enabling the Preview, you can see that the PictureBox object still has the previously set image.

5.2.1 Creation of a Digital Signage project

5.2.1.1 Aim

Assume to create a project that shows through a monitor the promotions of some products of a company.

5.2.1.2 What you need

For this kind of project it is necessary to have an image of the company logo, several pictures of the promoted products, a text-like file with a list of the promotional offers and a rather winning background image.

5.2.1.3 Making of

Create a new project by the "Open" button, choose the Root Folder and the size of the Design Monitors according to the output monitor which you assumed to use.

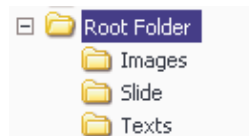


Note: in the following example, let's assume we created a project with Design Monitor size 800x600.

Copy all the mentioned files into separated subfolders in the **Root Folder**, as follows:

1. The company Logo image and the background image must be copied into the **Images** folder
2. The pictures of the promoted products must be copied into the **Slide** folder
3. The text-like file must be copied into the **Texts** folder

Then we will have the following structure:



Return to Peacock and change the Monitor's name ("Output 0") into the name of the shop you are creating the project for:



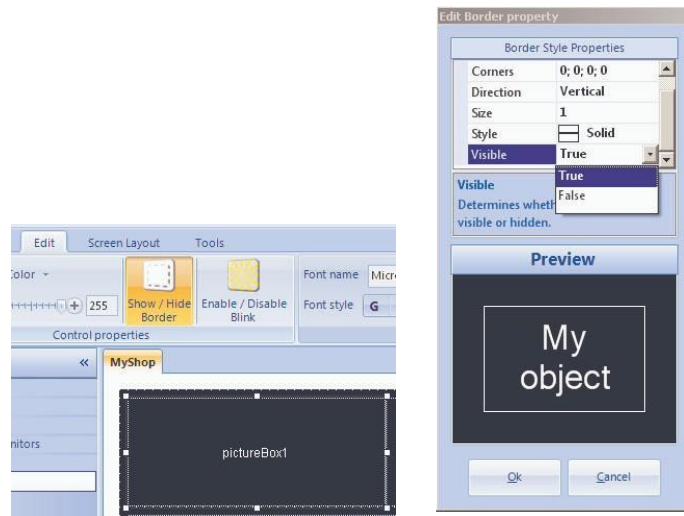
Add the following items to the Design Monitor, in the following positions and with the following dimensions:

1. **PictureBox** from "Image Controls". Set the **Location** property at 8; 8 and the **Size** property at 248; 104.
2. **Label** from "Text Controls". Set the **Location** property at 256; 8 and the **Size** property at 536; 104.
3. **SlideShow** from "Image Controls". Set the **Location** property at 8; 120 and the **Size** property at 784; 424.
4. **Ticker** from "Text Controls". Set the **Location** property at 0; 552 and the **Size** property at 800; 48.

The Design Monitor should appear as follows:



Select the **PictureBox** object and make the control's border invisible through the Visible parameter of the **Border** property, or by unchecking the **"Show/Hide Border"** button from the Edit menu, like in the following figures:



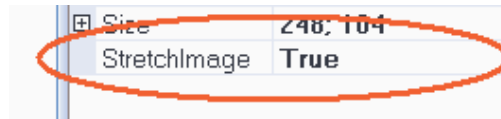
Set the full transparency on the control by setting the **AlphaBackColor** property to 0, or by using the slide object from the **"Control Properties"** group in the Edit menu, like in the following figures:



Then select the company Logo image through the editor of the **FileName** property.



Finally, set the **StretchImage** property on True.

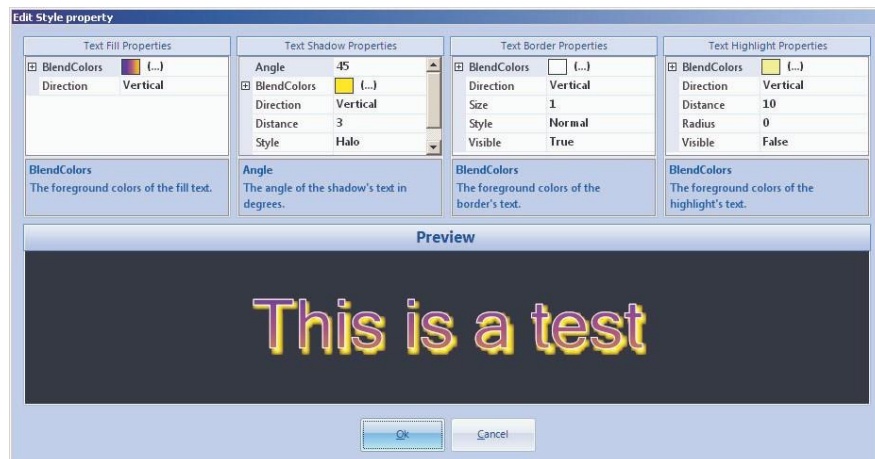


Select the **Label** object and make the control's border invisible through the Visible parameter of the **Border** property, or by unchecking the "Show/Hide Border" button from the Edit menu, like in the previous images. Modify the **Font** property using an "impressive" font type such as, for example, "Viner Hand ITC". Note that, if the **AutosizeFont** property is set on True, it is not important to determine the Font size because it is reset according to the control's height and width.

Set the full transparency on the control by setting the **AlphaBackColor** property to 0, or by using the slide object from the "Control Properties" group in the Edit menu, like in the following figures:



Insert the text (e.g.: "My Shop") in the **TextValue** property and modify the Style property, for example, as follows:



The result will be the following:

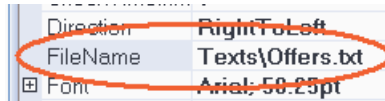


Index

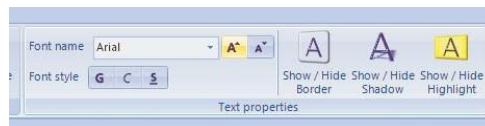
Select the **Slide** object and load the pictures of the promoted products byt setting the **Folder** property:



Select the **Ticker** object and make the control's border invisible through the Visible parameter of the **Border** property, or by unchecking the **Show/Hide Border** button from the Edit menu, like in the previous images. Then change the **BackColor** and **ForeColor** respectively with the Blue and Orange color. Select the text-like file in the FileName property, as in figure:



Increase the **Font** size up to about 50.25pt by the related property or using the button in the **Text Properties** group in the Edit menu, as in figure:



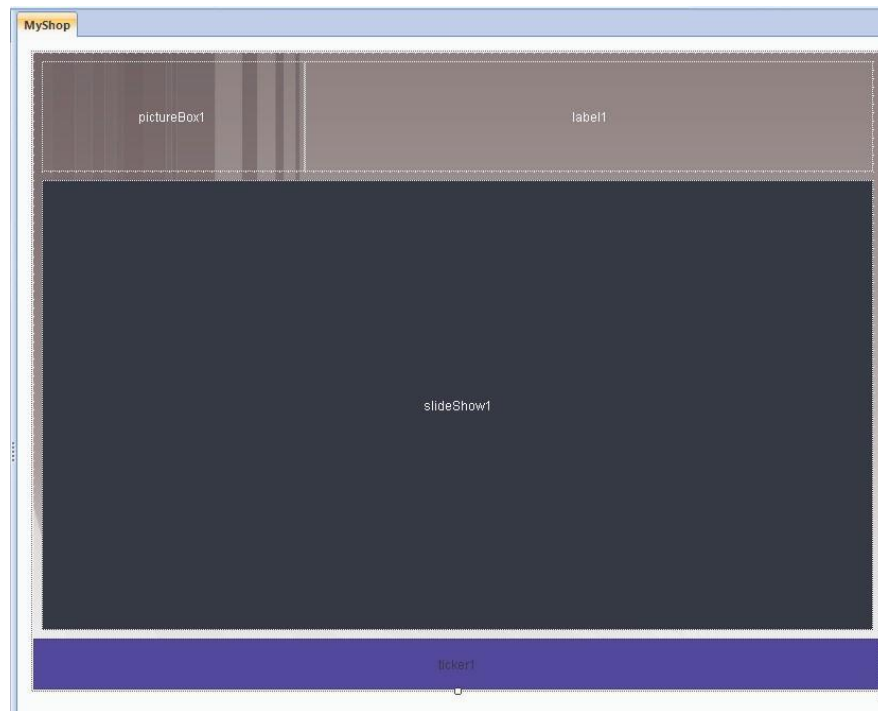
Finally, select the **Form** and set the Monitor background through the editor of the **BackgroundImage** property:



Index

The final result will be the following: In

Design:



In Preview:



5.2.2 Creation of a Web TV project

5.2.2.1 Aim

Assume to create a project that displays a WebTV.

5.2.2.2 What you need

For this kind of project it is necessary to have an image of the WebTV logo, some video clips to be played or, if you don't have them, a link to a WebTV video stream, a MySQL or SQLServer database containing, for example, SMS that you want to show through a ticker and a rather winning background image.

5.2.2.3 Making of

Create a new project by the "Open" button, choose the Root Folder and the size of the Design Monitors according to the output monitor which you assumed to use.



Note: in the following example, let's assume we created a project with Design Monitor size 800x600.

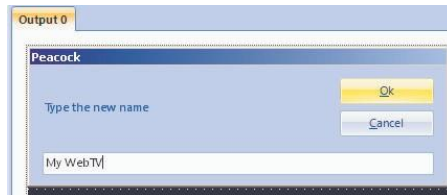
Copy all the mentioned files into separated subfolders in the **Root Folder**, as follows:

1. The company Logo image and the background image must be copied in the **Images** folder
2. The video clips of the WebTV sono must be copied in the **Videos** folder

Then we will have the following structure:



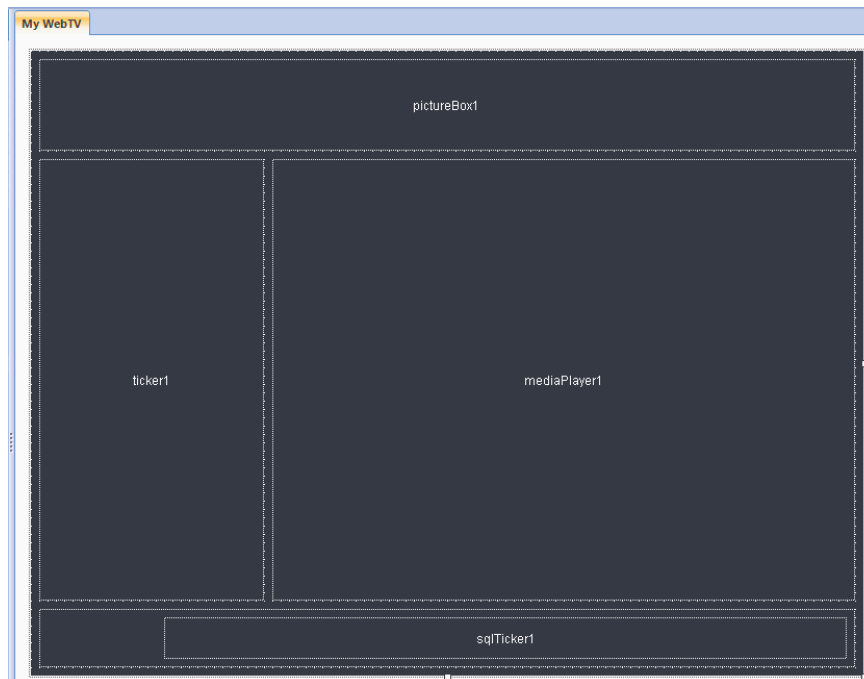
Return to Peacock and change the Monitor's name ("Output 0") into the name of the WebTV:



Add the following items to the Design Monitor, in the following positions and with the following dimensions:

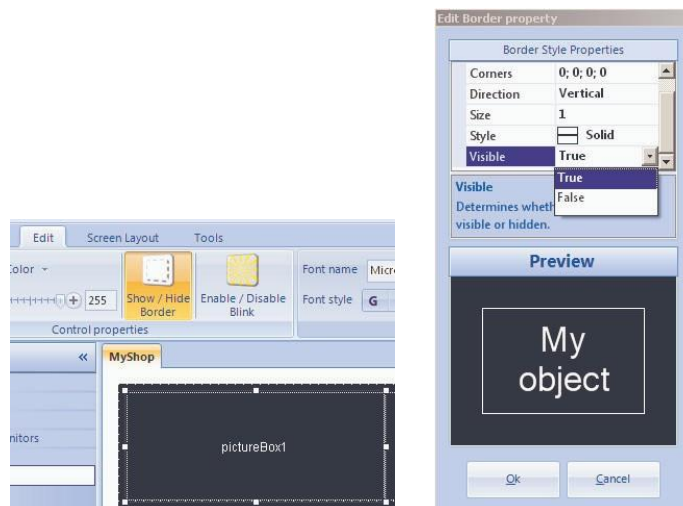
1. **PictureBox** from "Image Controls". Set the **Location** property at 8; 8 and the **Size** property at 784; 88.
2. **Label** from "Text Controls". Set the **Location** property at 8; 536 and the **Size** property at 784; 56.
3. **MediaPlayer** from "Media Controls". Set the **Location** property at 232; 104 and the **Size** property at 560; 424.
4. **Ticker** from "Text Controls". Set the **Location** property at 8; 104 and the **Size** property at 216; 424.
5. **SqlTicker** from "Text Controls". Set the **Location** property at 128; 544 and the **Size** property at 656; 40.

The Design Monitor should appear as follows:



Note: When you insert the MediaPlayer control, an AudioManager component automatically appears on screen. Such object can be left with its default settings, but if you want to listen to the MediaPlayer's audio, you need to set the ActiveControl property of the AudioManager object on "mediaPlayer1".

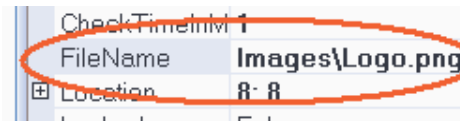
Select the **PictureBox** object and make the control's border invisible through the Visible parameter of the **Border** property, or by unchecking the **Show/Hide Border** button from the Edit menu, like in the following figures:



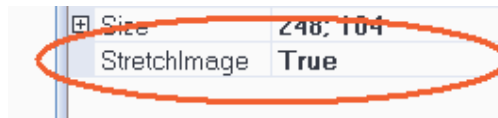
Set the full transparency on the control by setting the **AlphaBackColor** property to 0, or by using the slide object from the “**Control Properties**” group in the Edit menu, like in the following figures:



Then select the WebTV Logo image through the editor of the **FileName** property.



Finally, set the **StretchImage** property on True.



Select the **Label** object and make the control’s border invisible through the Visible parameter of the **Border** property, as in figure:



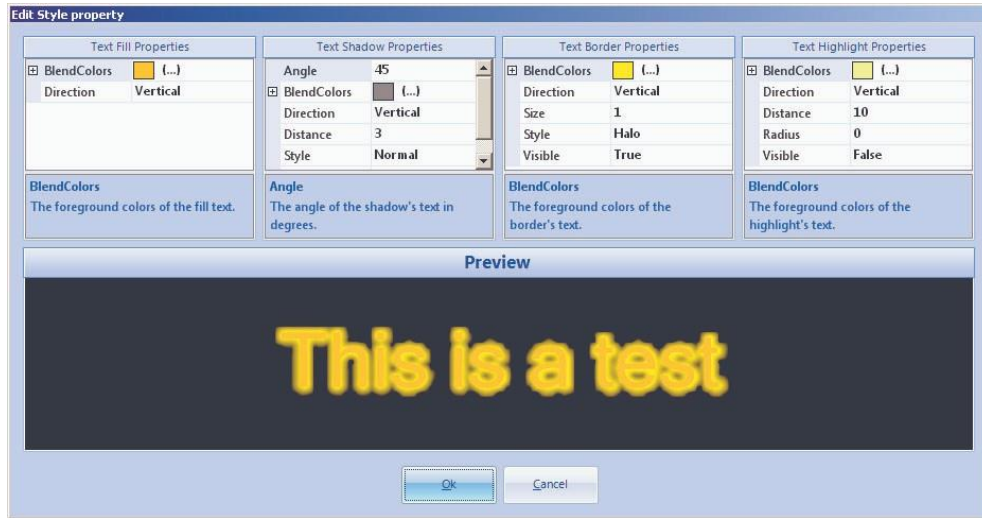
Set the full transparency on the control by setting the **AlphaBackColor** property to 0, or by using the slide object from the “**Control Properties**” group in the Edit menu, like in the following figures:



Modify the **Align** property in order to align the text of the label:



Insert the text (e.g.: “SMS box”) in the **TextValue** property and then and the modify the **Style** property, for example, as follows:



The result will be the following:

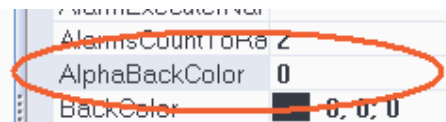
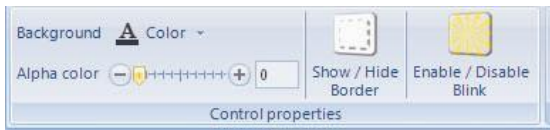


Select the **MediaPlayer** object and, if you are provided with some video clips, set the folder containing them in the **Folder** property:

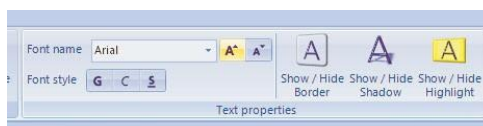


Otherwise, insert the URL of the stream to be played in the **StreamURL** property, remembering to change the **SourceType** property from “Folder” to “URL”.

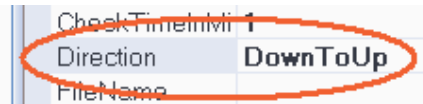
Select the **Ticker** object and make the control's border invisible through the Visible parameter of the **Border** property, or by unchecking the “**Show/Hide Border**” button from the Edit menu, like in the previous images. Set the full transparency on the control by setting the **AlphaBackColor** property to 0, or by using the slide object from the “**Control Properties**” group in the Edit menu, like in the following figures:



Set the **ForeColor** with the Gold color and increase the **Font** size up to about 23.25pt by the related property or using the button in the “**Text Properties**” group in the Edit menu, as in figure:



Set the **Direction** property on DownToUp and insert a text in the **TextValue** property (e.g.: "It's not that I'm afraid to die, I just don't want to be there when it happens". Woody Allen)

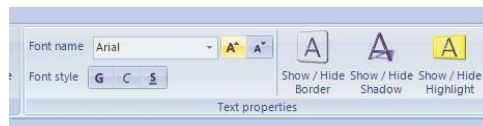


Select the **SqlTicker** object and make the control's border invisible through the Visible parameter of the **Border** property, or by unchecking the "Show/Hide Border" button from the Edit menu, like in the previous images.

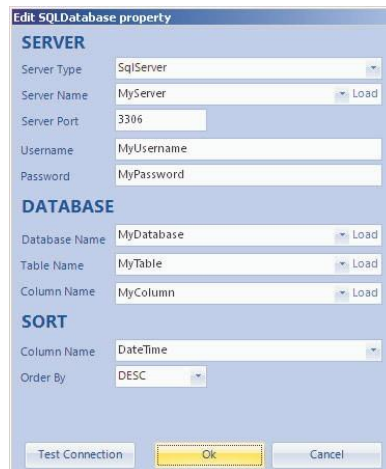
Set the full transparency on the control by setting the **AlphaBackColor** property to 0, or by using the slide object from the "Control Properties" group in the Edit menu, like in the following figures:



Set the **ForeColor** with the Gold color and increase the **Font** size up to about 23.25pt by the related property or using the button in the "Text Properties" group in the Edit menu, as in figure:



Set the **SQLDatabase** property with the related Editor, as in figure:



If you want you can moderate the speed by increasing the value in the **TimeValue** property from 17 to 51.

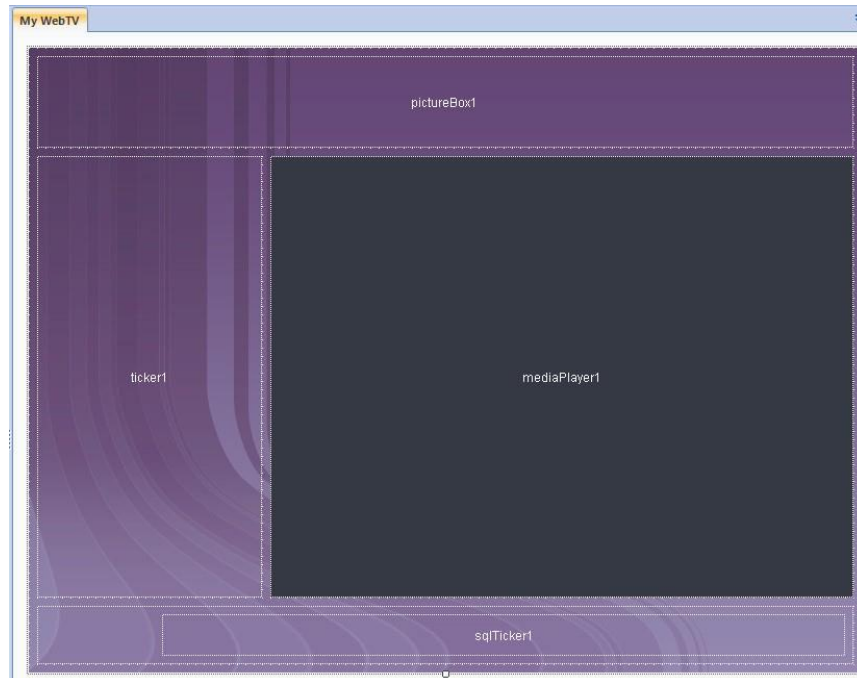
Finally, select the **Form** and set the Monitor background through the editor of the **BackgroundImage** property:



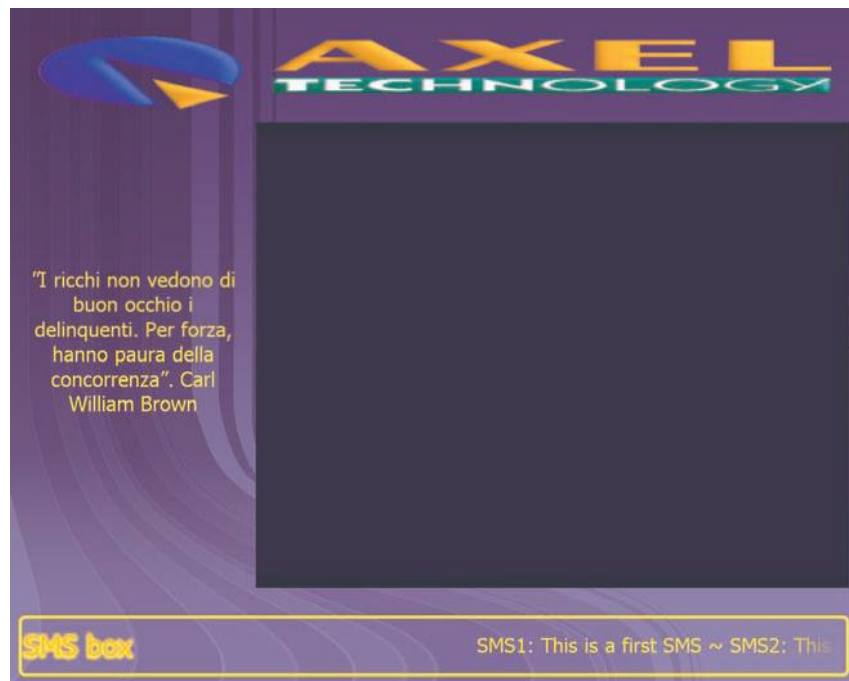
Index


The final result will be the following:

In Design:



In Preview:



 **Note:** now it is also possible to stream the output you generated. Refer to section 5.3.

5.3 Creation of a DMLVideo System Monitoring project

5.3.1.1 Aim

Assume to create a tv monitoring system that uses, for an amount of 16 channels, Axel DMLVideo software. To have a real-time reporting of any errors reported by DMLVideo, the Peacock provides a variety of alarm executors as well as a graphical display of the DMLVideo software.

5.3.1.2 What you need

For this kind of project you must have installed the DML program on any PC on the network, a SMTP server and a GPRS card installed for the SMS send and the Donkey program run on a given server.

5.3.1.3 Making of

Create a new project by the "Open" button, choose the Root Folder and the size of the Design Monitors according to the output monitor which you assumed to use.



Note: in the following example, let's assume we created a project with Design Monitor size 800x600.

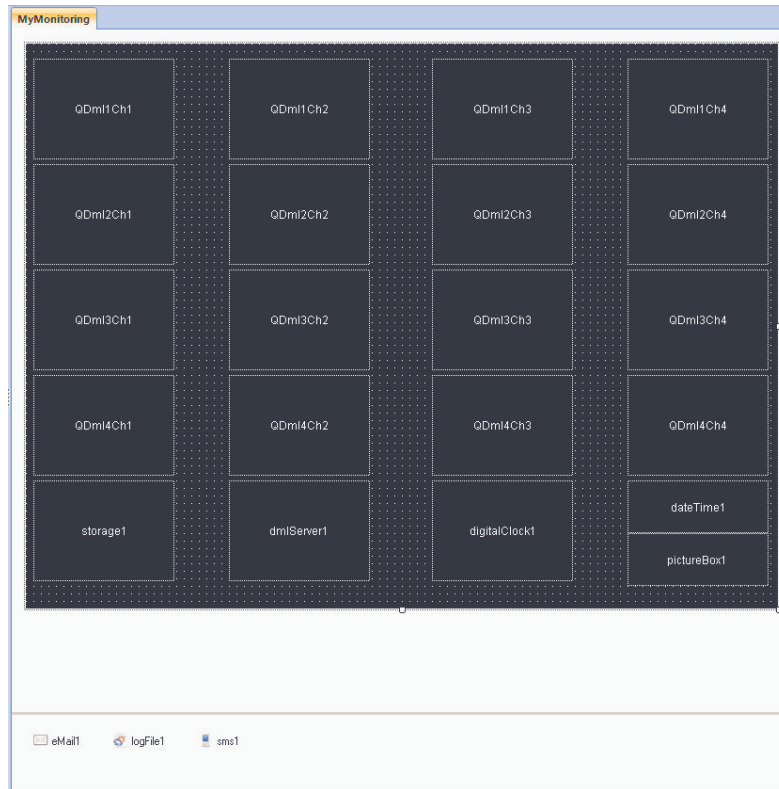
Return to Peacock and change the Monitor's name ("Output 0") into "MyMonitoring":



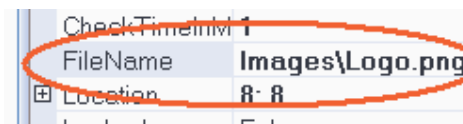
Add the following items to the Design Monitor, in the following positions and with the following dimensions:

1. **DigitalClock** from "Time Controls". Set the **Location** property at 432; 464 and the **Size** property at 150; 107.
2. **DateTime** from "Time Controls". Set the **Location** property at 640; 464 and the **Size** property at 150; 56.
3. **PictureBox** from "Image Controls". Settare la proprietà **Location** a 640; 528 and the **Size** property at 150; 56.
4. **DateTime** from "Time Controls". Set the **Location** property at 592; 456 and the **Size** property at 368; 50.
5. **N. 16 DMLVideo** from "AxelSoftwareMonitors". Set the **Location** property since they are spaced as shown below, and the **Size** property at 150; 107.
6. **Storage** from "AxelSoftwareMonitors". Set the **Location** property at 8; 464 and the **Size** property at 150; 107.
7. **DMLServer** from "AxelSoftwareMonitors". Set the **Location** property at 216; 464 and the **Size** property at 150; 107
8. **SMS** from "Alarms Notifiers".
9. **eMail** from "Alarms Notifiers".
10. **LogFile** from "Alarms Notifiers".

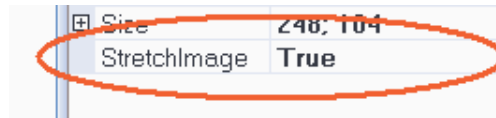
The Design Monitor should appear as follows:



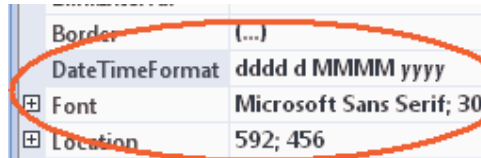
Select the **PictureBox** object and insert the company Logo image through the editor of the **FileName** property.



Finally, set the **StretchImage** property on True.



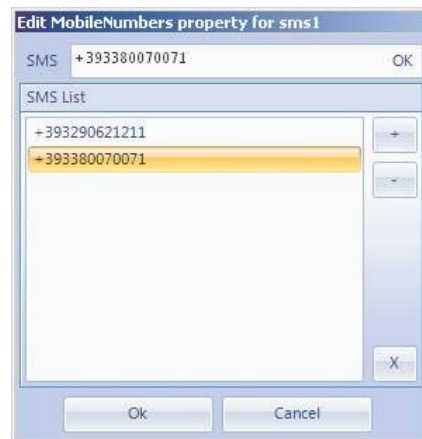
Select the **DateTime** object, set the **AutoSizeFont** property on False and set the **DateTimeFormat** and **Font** properties as in the following figure:



Set the previously inserted Alarms Notifiers with the settings that are necessary to make them be operative. Begin with the **remote** ones:


Select the **eMail** component and set the IP address and the port of the server where the Donkey is running, by setting **DonkeyServerAddress** and **DonkeyServerPort** properties.

Select the **SMS** component and set the IP address and the port of the server where the Donkey is running, by setting **DonkeyServerAddress** and **DonkeyServerPort** properties. Finally, set the **MobileNumbers** property with the list of mobile numbers to which you want to send the notification message, as in figure:



Now it's turn of the **local** Alarms Notifier:

Select the LogFile component and set the name (without path) of the log file you want to create/edit using the **Filename** property.

 **Note:** the GPRS card and the SMTP mail server must be installed on the server where the Donkey is running.

Now it is possible to set the paths needed for the AxelSoftwareMonitors controls to monitor the DMLVideo and XPlayout programs, and the whole server.

Select **DMLVideo** and set the **StatusFolder** property with the path on the StatusAndLogs folder containing the Status.txt file.

Select **DMLServer** and set the **RemotePcName** property with the name of the remote server where the Donkey is running, to monitor the partitions (8 maximum) of the server itself. Then, set the partition list through the **HardDiskList** property. Finally, set the folders of the server that you want to monitor.

Select **Storage** and set the **RemotePcName** property with the name of the remote server where the Donkey is running, to monitor the partitions (8 maximum) of the server itself. Then, set the partition list through the **HardDiskList** property.

Finally, for each AxelSoftwaremonitors control, it is necessary to associate one or more AlarmsExecuter or AlarmNotifiers components so that, in case of an error, the notifications will be sent. To do this, there are two ways:

1. Select one by one all the **DMLVideo** controls and the **Server** control, then set the **AlarmExecuterName** property for each one as follows:



2. Use the Form's panel by selecting one by one all the Alarms Executors or Notifiers from the Alarms Executors / Notifiers curtain menu as in figure:



Assign to each of them the controls in that form by clicking the "Change Controls Associations" button as shown in figure:



The final result will be similar to the following:



5.4 How to create the Streaming of a Peacock project

5.4.1.1 Aim

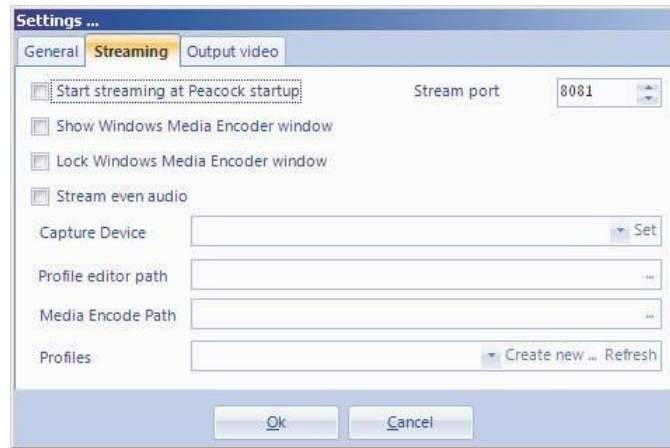
Assume to create a WebTV and you want to send on web the stream of a project created with the Peacock.


5.4.1.2 What you need

The Peacock uses the UScreenCapture streaming filter. If you own a program for streaming, it is advisable to set the UScreenCapture filter on this program. If, instead, you don't have any program for streaming, you can install the free software Windows Media Encoder from Peacock's installation CD.

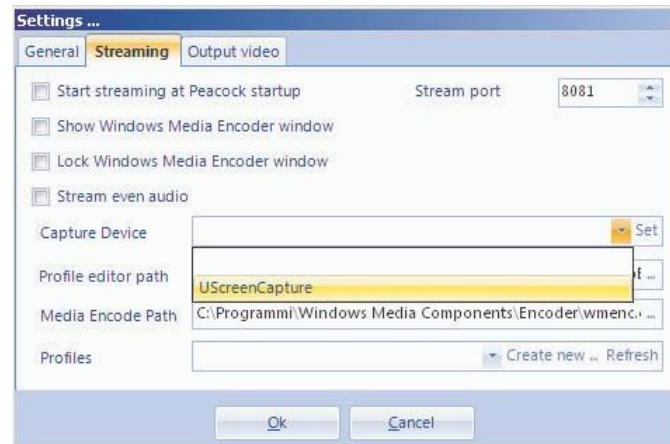
5.4.1.3 Making of

Create a new project and, from the Tools menu, click on Options. Then select the Streaming tag as shown in figure:



 **Note:** For the settings' use refer to section 3.1.3.

Set the paths of Windows Media Encoder and Window Profile Encode and select the UScreenCapture Capture Device as shown:

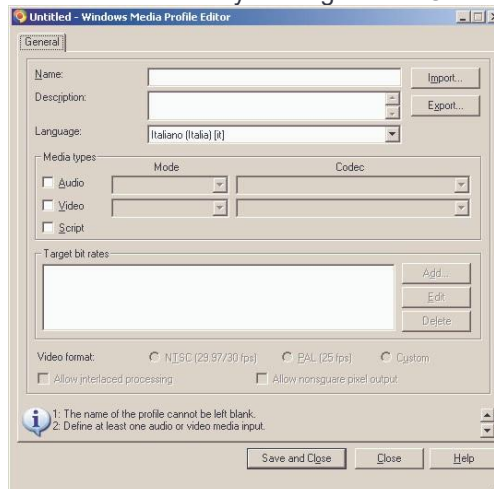


Clicking on the Set button, once the Capture Device is given, the following window appears:



In this window you can set the output monitor and choose on what portion of the monitor the stream will run: either on the whole monitor (check "Capture full screen") or on one region only, by setting the coordinates of the point on the upper left and on the bottom right to determine the stream area.

Then choose an existing profile or save a new one by clicking on the "Create new ..." button as shown:



Note that, after saving the settings, the Start button for the Streaming is enabled. So, the program is ready to start the stream according to the settings previously set.

Click on Start and start the stream through the Peacock. To stop the stream, click on Stop.

5.5 Creating a scheduled Peacock project

5.5.1.1 Aim

Assume to create a project with several outputs and that you want to display them on the same screen, following a timetable for each one. To do this, you must set the Scheduler and enable it.

5.5.1.2 Making of

Create a new project with 2 screens and open the Edit Scheduler window from the Tools menu. At this point it is necessary to determine the frequency of both the first and second selected screen. Once they have been added to the list, remember to enable the scheduler.

6. Glossary

A

- **Alarms**: Category containing all the controls of individual local alarm.
- **AlarmExecuter**: Category containing all components required for the notifications in the event a control generates an alarm.
- **AlarmExecuterName**: Control property associated with one or more AlarmExecuter components.
- **AlarmsCountToRaiseAlarm**: Control property that determines the number beyond which the alarm is started. The alarm is run only if the AlarmExecuterName property is not empty.
- **Align**: Control property that determines the text alignment.
- **AlphaBackColor**: Control property that determines the transparency (0 to 255) for the control's back color.
- **Angle**: Control property that determines the bend angle for the control's text.
- **AutoSizeFont**: Control property that determines the automatic resizing of the text according to the size of the control.
- **AxelSoftwareMonitors**: Category containing all the monitoring controls for the Axel programs.

B

- **BackColor**: Control property that determines il colore dello sfondo del control o del monitor.
- **BackColorBlend**: Control property that determines the color mix for the control background or monitor background.
- **BackColorBlendDirection**: Control property that determines the color mix direction for the control background or monitor background.
- **BackColorType**: Control property that determines whether to consider the BackColorBlend property or the BackColor property.
- **BackgroundImage**: Monitor property that determines the path of an image used as a background for the monitor itself.
- **BlendColors**: Subproperty of Style and Border properties that determines the color mix to be applied to the Text Border or Style.
- **BlendTransparentVisible**: Control property that determines if the transparency visualization at the text's beginning and end is enabled or not.
- **Blink**: Control property that determines if the control is blinking or not.
- **BlinkInterval**: Control property that determines the interval (in milliseconds) of the control's flashing.
- **Border**: Control property that determines the border's properties.
- **Border Text**: Control property that determines the text border's properties.

C

- **CheckTimeInMinute**: Control property that determines how often (in minutes) the control must self-check its own dynamic properties.
- **CheckTimeInSecond**: Control property that determines how often (in seconds) the control must self-check its own dynamic properties.
- **Compile**: Tool that allows you to create the exe file for the selected monitor. This file is created into the Root Folder.
- **Component**: Object linked to one or more controls, working in background. It is inserted in the project's Monitors.
- **Components**: Category containing all the components that are useful for managing the controls of Peacock's project.
- **Connection**: Control property that determines the type of connection to be used.
- **Control**: Object to be displayed on the output. It is inserted in the project's Monitors.
- **Corners**: Subproperty of Border property that determines the borders angle in percentual, related to the length of the control's underside.

D

- **DateFormat**: Control property that determines the date format to display.

- **Design:** Peacock's design mode.
- **Device:** Control property that determines the DirectShow compatible device to display.
- **Direction:** Control property that determines the direction of the text to display.
- **Donkey:** Program used as a remote request manager for Peacock.
- **DonkeyServerAddress:** IP address for the Donkey server.
- **DonkeyServerPort:** Donkey server port.
- **DrawGrid:** Monitor property that determines whether to display the grid or not.

E

- **Email:** Component property that determines the settings for the email to send in case of an alarm.
- **Export:** Function that imports a monitor's layout from a schema managed by Peacock.

F

- **FileName:** Control property that determines the path of the file you want to display.
- **Fill Text:** Control property that determines the text's filling properties.
- **Folder:** Control property that determines the folder that must display its content.
- **Font:** Control property that determines the font for the text to display.
- **ForeColor:** Control property that determines the color for the text to display.
- **ForeColorBlend:** Control property that determines the color mix for the text to display.
- **ForeColorBlendDirection:** Control property that determines the color mix direction for the text to display.
- **ForeColorType:** Control property that determines whether to take into account the ForeColorBlend property or the ForeColor property.
- **FTP:** Control property that determines the FTP connection properties.

G

- **GPICountDown:** Control property that determines the GPI settings for the count down.
- **GPICountReset:** Control property that determines the GPI settings for the count down reset.
- **GPICountUp:** Control property that determines the GPI settings for the count up.
- **GPIReset:** Control property that determines the GPI settings for the count up reset.
- **GPIStart:** Control property that determines the GPI settings for the count start.
- **GPIStop:** Control property that determines the GPI settings for the count stop.
- **GridSize:** Control property that determines the grid size.

H

- **Highlight Text:** Control property that determines the text's highlighting properties.

I

- **Image Controls:** Category containing all the image displaying controls.
- **Import:** Function that exports a monitor's layout on a schema managed by Peacock.
- **IniFile:** Control property that determines the path, section and key of the INI file.
- **Internet Controls:** Category containing all the controls for displaying Internet pages and Flash compatible objects.
- **IPAddress:** Component property that determines the IP address to use for running the message in case of an alarm.

J

K

L

- **Labelcolor:** Control property that determines the color for the text to display.
- **LineSeparator:** Control property that determines the character used to separate the text lines.
- **Location:** Control property that determines the control's position into the monitor containing it.

- **LocalRS232**: Component property that determines the settings for the local serial port to use for sending the message in case of an alarm.
- **Locked**: Control property that determines whether the control can be resized or blocked.
- **Loop Mode**: Control property that determines if the loop mode is enabled on the control.

M

- **Max**: Control property that determines the maximum value to be achieved.
- **Media Controls**: Category containing all the controls for audio and video stream displaying.
- **Message**: Control property that determines the message to send in case of an alarm.
- **Min**: Control property that determines the minimum value to be achieved.
- **MobileNumbers**: Component property that determines the list of mobile numbers to send the alarm to.
- **Mode**: Control property that determines how to choose the text to be displayed. It can be Static or Dynamic.
- **Monitor**: Project's Output. Every project can have several Monitors, each of them is associated to an output (that can be set in the MonitorIndex property).
- **MonitorIndex**: Monitor property that determines what is the output to use.
- **Multiline**: Control property that determines whether the text must be displayed on more than one line or not.
- **Mute**: Control property that determines whether the volume level is enabled or not.

N

- **Name**: Control, component and monitor property that determines their name.

O

- **OffsetPosition**: Control property that determines the coordinates of the web page from which to start the visualization.
- **On Air**: If enabled, all the controls inside the monitors cannot be resized.
- **Options**: Peacock's options.
- **Orientation**: Control property that determines the control's orientation.

P

- **PauseTime** Control property that determines the pause time between the displaying of two images.
- **Port**: Component property that determines the port to use for sending the message in case of an alarm.
- **Preview**: Peacock's project preview.
- **ProcessIsClosable**: Component property that determines if the process can be closed when the alarm is ceased.
- **ProcessParameters**: Component property that determines the parameters for the process to run in case of an alarm.
- **ProcessPath**: Component property that determines the path for the process to run in case of an alarm.

Q

R

- **Real Time**: Peacock's project preview.
- **Recent Documents**: Peacock's projects recently saved.
- **RemoteAlarms**: Category containing all the controls of individual remote alarm.
- **Root Folder**: Peacock's project default path.

S

- **ScreenOffSet**: Monitor property that determines, if ScreenState is not Maximized, the movement of

the output window from coordinates 0;0.

- **ScreenState**: Monitor property that determines the output window mode. It can be Normal (it can be set in Options) or Maximized (it fills the whole output monitor).
- **Schema**: Monitor layout of a Peacock's project.
- **Server**: Control property that determines the server host.
- **ServerPort**: Control property that determines the server port.
- **Shadow Text**: Control property that determines the text shadow properties.
- **ShowChannelName**: Control property that determines the controls title displaying.
- **Size**: Control or monitor property that determines their size.
- **SnapToGrid**: Monitor property that determines whether the controls inserted on it are automatically aligned to the grid or not.
- **SoundPath**: Component property that determines the path for the WAV file to use for running the alarm.
- **Speed**: Control property that determines the text scrolling speed.
- **SQLDatabase**: Control property that determines the properties of the database to connect to.
- **StartTime**: Control property that determines the initial time to be displayed.
- **StatusFolder**: Control property that determines the path of the folder containing the status file.
- **StreamURL**: Control property that determines the URL of the stream to display on the control.
- **StretchImage**: Control property that determines whether the image is resized according to the control or not.
- **Style**: Control property that determines the text properties.

T

- **Template**: Peacock project already set, it can be selected during the creation of a new project.
- **Text Controls**: Category containing all the controls of text visualization.
- **TextValue**: Control property that determines the value of the text to display.
- **Time Controls**: Category containing all the controls of date and time visualization.
- **TimeFormat**: Control property that determines the format of the time to display.
- **TimeOut**: Component property that determines the time beyond which the alarm is disabled.
- **TimeZoneInformation**: Control property that determines the time zone to set on the control's date/time.
- **TimerValue**: Control property that determines the text's scrolling step.
- **Titlecolor**: Control property that determines the color of the title to display.
- **Type**: Component property that determines whether the alarm message is run locally or via Donkey.

U

- **URL**: Control property that determines the URL path.

V

- **ViewMode**: Control property that determines the view mode for the control.
- **Volume**: Control or component property that determines the Audio level.
- **VuMeterDock**: Control property that determines the position, if displayed, of the VuMeter in the Media control.

W

X

Y

Z