

Stardock

Impression
c r e a t o r

User's Guide

Version 1.0



Contents

1 What is ImpressionCreator?	4
2 ImpressionCreator Basics	5
2.1 WHAT IS AN OBJECT?	5
2.2 HOW DO WE MAKE A NEW OBJECT?	5
2.3 HOW DO WE MOVE THE OBJECT?	6
2.4 HOW DO WE CHANGE THE IMAGE?	6
<i>Image objects</i>	6
<i>Text Objects</i>	8
2.5 THE BASICS OF LOADING, SAVING AND DELETING	8
3 Making the object do something	9
<i>Layer</i>	9
<i>Shortcut</i>	9
<i>URL</i>	10
<i>Object Controller</i>	10
<i>Widget</i>	10
3.2 HOW DO WE SAVE THE OBJECT?	11
<i>Saving the workspace</i>	11
<i>Saving selected objects</i>	11
<i>Exporting for distribution</i>	11
3.3 HOW DO WE LOAD THE OBJECT AGAIN TO EDIT IT?	13
<i>Loading a workspace</i>	13
<i>Loading selected objects</i>	13
4 Enhancing your objects	14
4.1 NAMING YOUR OBJECT	14
4.2 HOW DO WE SCALE THE OBJECT?	16
<i>Scaling images</i>	16
4.3 GROUPING AND HOW NAMING IS INVOLVED	17
<i>Groups</i>	17
<i>Parents and Children</i>	17



4.4	OBJECT RELATIONS	19
	<i>Visibility</i>	19
	<i>Z-Order</i>	19
	<i>Movement</i>	20
	<i>Activation</i>	20
	<i>Start with</i>	20
	<i>Default Cursor</i>	20
	<i>Popups</i>	20
4.5	TRANSPARENCY	21
4.6	ROLLOVER, AWAY, DOWN STATES	21
4.7	MESSAGES.....	23
4.8	COPYING OBJECTS.....	24
5	Advanced Object Effects	25
5.1	ADDING ANIMATION	25
	<i>Static Animations</i>	25
	<i>Image animations</i>	26
5.2	SHADOWS/GLOWS.....	27
5.3	COLOR CHANGING.....	27
5.4	SOUND.....	28
5.5	ADDING TOOLTIPS	28
5.6	ADDING CUSTOM FILES	28
6	Tutorial.....	29
6.1	GETTING STARTED.....	29
6.2	CREATING THE INSTALLER BACKGROUND.....	30
6.3	CREATING THE CLOSE BUTTON.....	31
6.4	CREATING A WEB LINK.....	32
6.5	CREATING THE BASIC BUTTONS	33
6.6	CREATING SOME EFFECTS FOR THE BUTTONS	34
6.7	ADDING FUNCTIONALITY TO THE BUTTONS.....	37
6.8	FINISHING TOUCH.....	37
6.9	EXPORTING THE APPLICATION	38



1

What is ImpressionCreator?

Having created your application masterpiece, you want to ensure that that when people come to install it they are immediately impressed and excited about using your application.

The purpose of ImpressionCreator is to allow you to create an application that introduces users to your software in a visually attractive and functional way. This application would typically be the sort of thing that is “auto run” when the user inserts the CD containing your application into their PC, though it can be distributed in a range of ways.

By way of example, the application could install your program; provide access to information about the application in documents, link to your website and so on.

This guide is designed to ease you through the process of creating that impression. First you will be introduced to some basic concepts and then we will look at creating the application as you want it. The last stage is showing you how to create the application itself ready for use.

Finally, we will walk you through an example of creating an application. You can tweak this tutorial to meet your own needs, and use this and the other samples as demonstrations of just some of the things you can do with ImpressionCreator.

ImpressionCreator is a version of DesktopX Pro. In this documentation we focus on the basic features relevant to the creation of installer applications. If you wish to explore further, particularly with a view to scripting, you can access the full DesktopX documentation here: www.desktopx.net/resources.asp. If you wish to create applications containing ActiveX controls you need to upgrade to ImpressionCreator Enterprise. Please contact Stardock for more information

Have fun – let's go create a good impression!



2

ImpressionCreator Basics

2.1 What is an object?

ImpressionCreator works by allowing you to create a range of “objects”. For each of these “objects” you can set their appearance and also define the purpose that they serve. If you imagine a regular application there are several “objects” in it, backgrounds, text, buttons and so on. ImpressionCreator works with “objects” just like these but provides you with additional flexibility in terms of how they look and what they do. Everything you create using ImpressionCreator will be an “object”. Sometimes they do things on their own, sometimes in conjunction with other “objects”, but whatever their function they are an “object”.

2.2 How do we make a new object?

Before we start you need to know how to create objects. You’ll be pleased to know is that the actual creation of a basic object is very simple.

All you need to do is click on the “New Object” button on your ImpressionCreator toolbar.

You will then see a basic object appear on the desktop ready for you to customize. The “Object Properties” dialog will also be displayed to allow you to immediately start configuring your object.

At other times you can display the object’s properties dialog by right clicking the object and selection “Properties” from the menu that appears.





2.3 *How do we move the object?*

To move the object, simply drag the object to the position you want it. Another useful thing to remember is that when an object is selected you can move it by using the cursor keys to fine tune its position. In order to do this, hold down the CTRL key and click the object(s) you wish to move to select. Keep the CTRL key held down and use the cursor keys to move the object. If you hold down the SHIFT key at the same time then the object will move by 10 pixels at a time in the direction specified.

2.4 *How do we change the image?*

There are two basic types of objects, "Image" objects and "Text" objects. Now that you have a default object we will now make it look more interesting.

You have the "Object Properties" dialog in front of you. Click on the "States" tab, which is where you can define the appearance of your object. There are a series of tabs at the bottom of this dialog for this purpose. You will see that the "Appearance" tab is selected by default, and that the object is an "Image" and the file selected is "<Default>".



Image objects

If you want a graphical object, you can choose an image already in the workspace by selecting it from the drop-down list where it says "<Default>", or you can "Browse" to find a new image on your computer. Images of various types are supported (BMP, ICO, PNG, TGA and JPG). If you select a file and then click "Apply" you will immediately see the object change to become the image you specified.

When applications save images they are saved as rectangular images, but obviously you don't want to be restricted to having rectangular objects. Both the PNG and TGA image formats allow you to include transparency information with the file, which means that although the saved image may be a rectangle, the actual image that you see on your screen can be any shape you want, because the transparent areas are not shown on screen.



This image on the left is a PNG image created in Macromedia Fireworks with a checkerboard effect showing the transparency.



When this image is used in ImpressionCreator all transparency is preserved and the background shows through as you can see on the right.

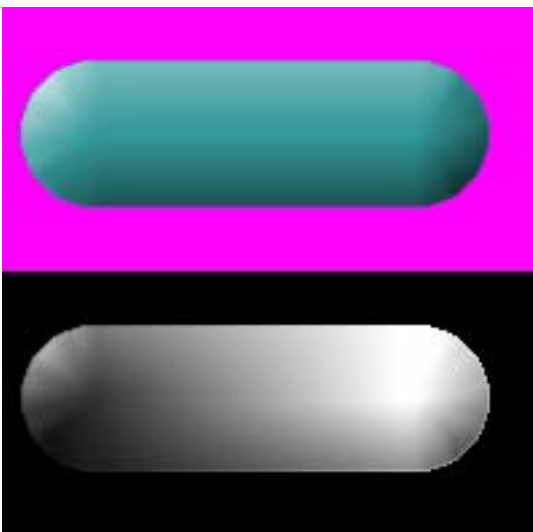


ImpressionCreator even allows this sort of effect in image formats that don't support transparency such as BMP. In these image formats, ImpressionCreator will interpret the color referred to as "Magic pink" to be transparent. "Magic pink" is pure magenta with an RGB (Red, Green, Blue) value of 255, 0, 255.



Although using this method you can't have a graduated transparency as you find in the shadow of the above PNG example, you can make a close approximation. Note, that you need hard crisp edges because if you "anti-alias" to soften the object edges, your object will have a pink glow as these colors are not pure "magic pink".

One final advanced tip for you. If you're forced to use "magic pink" for transparency in BMP images there is a final technique you can use to achieve graduated transparency like PNG files can provide.



What you need to do is double the height of your source image and in the lower half use grayscales to indicate the level of transparency. This is best explained by looking at the image on the left.

The top half represents the basic image, and in the lower half black represents total transparency, and the closer you get to white the more the object will appear.

If you use this graphic normally you will see the grayscale part, so to indicate that you are providing transparency information within the image, you need to go to the "Transparency" tab and check the box that says "The object's transparency is set by the graphic used".



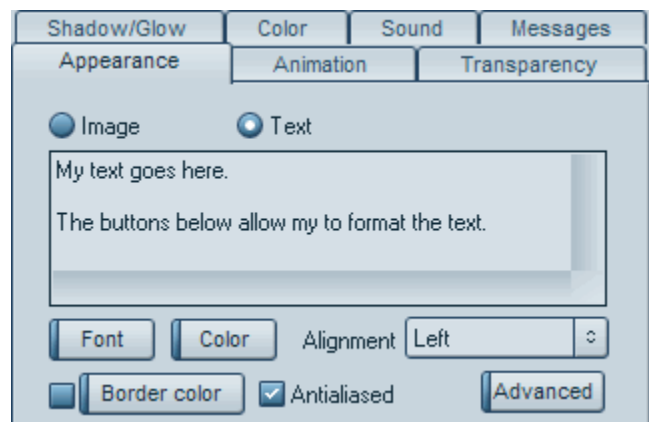
Doing this will cause the image to appear like the one on the left. This effect is very powerful if you are forced to use bitmap images, but PNG and TGA files will allow you to achieve the best effects.

Text Objects

If you choose a “Text” object type then the configuration options changes to a more simple set.

You can enter the text you want in the text area, and set the text alignment in the dropdown “Alignment” list. To further customize your text, you can specify the font (including style and size) and color in which the text will appear. Antialiasing text will make it easier to read.

If you want, you can specify that the text has a border by clicking the box next to the “Border color” button and selecting a border color by clicking the button. By having a colored border and setting the text color to “Magic pink” which is treated as transparent you can create text which is purely an outline.



2.5 The basics of loading, saving and deleting

These items will be discussed in more detail later, but very quickly, here is how you do these things. Your workspace (all your ImpressionCreator objects) is saved automatically as you work, but you can explicitly save it by clicking the “Save Workspace” button on your ImpressionCreator Toolbar. You can then specify the location where you would like to save your workspace.

To load a saved workspace you should click the “Load Workspace” button on the toolbar.

If you have created an object that you no longer want, simply right click the object and select “Delete object(s)”.



3

Making the object do something

Each object you create has a “Type” – that is, the function it performs. By default, objects are “Layers” that sit on the screen and look nice, but don’t actually do anything. If you are going to create an installer you really need it to do more than this so we need to change the “Type” to something more useful.

To change the object type, click the “Change” button next to where it says “Type of object” in the “Object properties” dialogue. This will present you with a list of the other basic object types available.

Once a type is chosen, the activity associated with that object type will occur when an object is clicked, though the object’s “Start with” parameter can be changed to alter when this occurs.

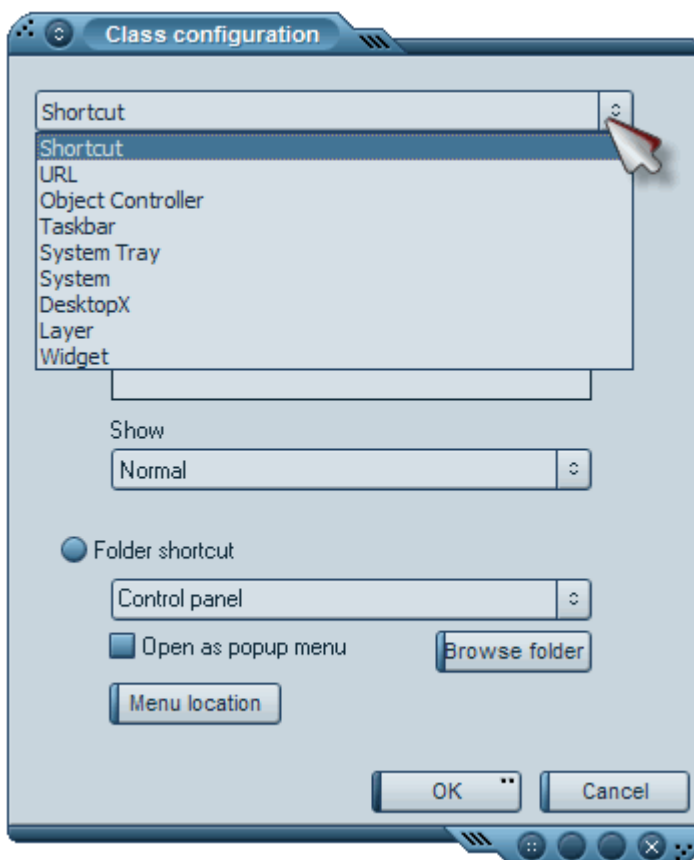
We will go through each of these items in turn and explain how to use them. I encourage you to try them out as you go and experiment with the functionality on offer. This is the best way to learn.

Layer

The default type of object created is a “Layer”. A layer object doesn’t interact with the user in any way. There are several reasons why you may want objects of this type. It may be a text label providing information to the user, or it may just be a graphic enhancing the visual experience for the user.

Shortcut

A “Shortcut” allows the object to link to a file or a folder on your machine or a file which has been included as part of your application. If you





choose a file (e.g. Notepad.exe) you can add arguments to that link. For example, if the file shortcut was Notepad.exe, you could add the argument “d:\whatsnew.txt” which would launch Notepad and then open the file “mydoc.txt”. You can also specify how that application will appear when launched (e.g. maximized).

In addition to the specific paths that you can enter, ImpressionCreator provides several variables that can be used to point to specific directories.

`%exedir%` - returns the executable directory of the application you create. This is useful if running from a CD as you know where additional files reside relative to the location of the executable.

`%objectdir%` - returns the object data directory (where custom files are stored).

So, for example, assuming the installer is on a CD, if you wished to link to a PDF in the “docs” directory, you could link to this “`%exedir%\docs\mydocs.pdf`”. These variables should be used in Shortcuts, not as arguments.

If you choose a “Folder shortcut”, you can choose a system folder such as the Control Panel, or you can browse for any folder on your computer. For example many install CDs have an option to “browse this CD”. You could use a folder shortcut in association and set the target as the `%exedir%` variable as shown above to do this assuming your installer is in the root directory of your CD.

URL

A “URL” object type is similar to the shortcut but it is dedicated to launching web pages in your browser, so all you need to do is enter the web address (e.g. `www.mycompany.com`).

Object Controller

Objects can interact with other objects and this is a way to carry out some of the most common of these interactions.

An individual object or groups of objects can be defined as “popups”, which means that they can be shown on screen only when they are required. We will define how to do this later, but at the moment it is enough to know that the first 2 options here to show or hide the object.

Obviously these commands mean nothing without specifying which object is the target of this action. You do this by selecting the object from the “Target object” dropdown list.

Widget

This type of object is useful because it allows you to replicate the functionality available in the exported application's menus – to close the application, minimize it, or display the About box.



3.2 How do we save the object?

OK, earlier you learnt the basics of creating and saving objects, but now you need to know how to save and export your creation properly.

The first thing to consider is that there are two fundamental ways to save your work. The first allows you to continue to develop your work, and the second will export your creation for distribution to users.

If you plan to continue work you can choose to either save the workspace (all your objects), or just selected objects that you can load again into another workspace.

Saving the workspace

To save the entire workspace, simply right click the ImpressionCreator icon. There are two options. ImpressionCreator automatically saves work as you go along so if you close ImpressionCreator and reopen it, you can continue from where you left off, but the “Save Workspace” option just reconfirms this. You can also choose to “Save Workspace As ...”. This will prompt you for a location to save the workspace you are working on.

Saving selected objects

The first thing to do is select the object(s) you want to export. You can do this on the screen by dragging a rectangle with your cursor. A selection area will be highlighted and those objects selected will take on a “blue tint”.

You can also do this by selecting the objects in the Object Navigator. To access the Object Navigator, click the “List Objects” button on the ImpressionCreator Toolbar. From the Object Navigator you can click, SHIFT-Click and CTRL-Click objects on the list to select them.

Once you've selected the objects click “Export” on the ImpressionCreator toolbar which will provide you with a range of options.

Choose to “Export as a packaged object” to export just those objects. On the left there is an option to export just the selected objects, or you can also export “Related” objects, but we will cover these later so you can ignore them for now. You can then specify where to save the file.

Exporting for distribution

On the Export dialog, there is a third option is the one you require for exporting your creation.

You will be asked to complete some information which are related to the object or are displayed in the “About” box:

Application name: Specify the name for your application here





Company: Place your name here, either yourself as the author or your company

Version: You can specify the application version here

URL: You can provide a website address that the user can link to from the 'About' box

Application icon: Optionally you can check the box and specify a ".ICO" file that will be used for the application. If an icon is not specified then the default ImpressionCreator icon will be used.

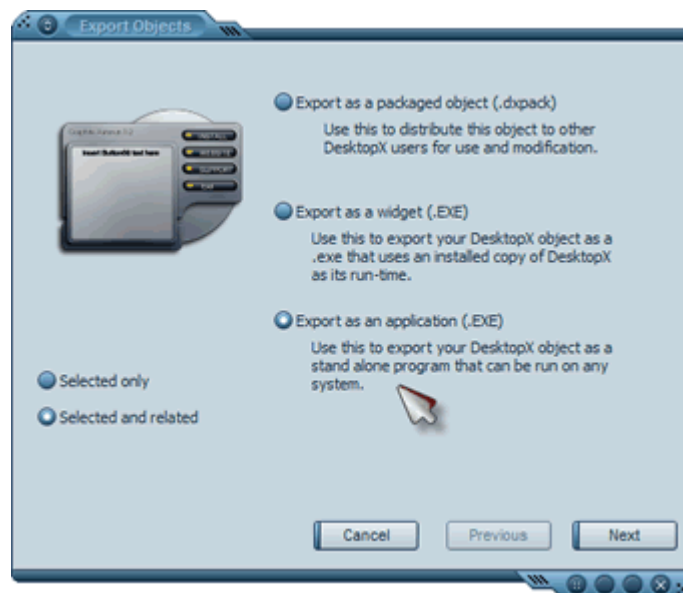
Run Type: You can specify how you want the application to run. Main applications will probably want to run as "Taskbar items" and utilities run from the "System Tray".

Multiple instances: You should leave this unchecked as you don't want to allow more than one copy of the application to run at any time.

On the next page you have options to export the file in two ways. The first way creates a single executable file is created which can be run. This is the option you need to select.

The final item on this page allows to specify a text file that will be accessible from the "About" box. This can be used to communicate information about the application or a licensing agreement for the applications use.

Once you have specified your choices here, the next and final page asks you to confirm your agreement to the distribution of the executable files, and then you can specify where you want to store your application/files.





3.3 How do we load the object again to edit it?

Obviously you won't necessarily complete all your work in one session so you need to be able to load your work again once you have saved it. As mentioned above, your work is saved as you go along, so if you close ImpressionCreator and then reopen it you can continue on. However there are other options available to you.

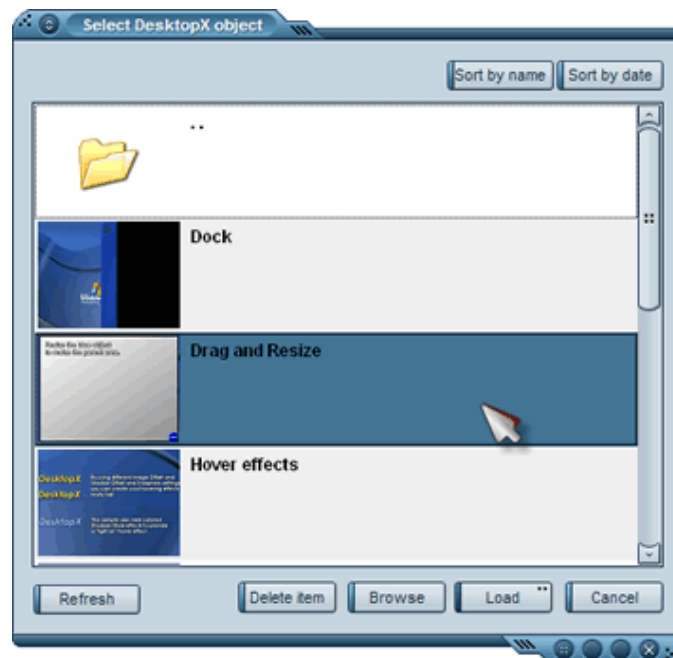
Loading a workspace

To load an entire workspace, simply click the "Load Workspace" button on the ImpressionCreator Toolbar. You can then select from Workspaces already in your "Workspaces" folder, or browse to find one that you have stored in another location. When you select one and click "Load", this will replace the workspace you are currently working on so you will be asked if you want to save your current workspace before you load the new one.

Once you have made your choice as to whether to save the existing workspace or not the new workspace will be loaded ready for you to work on it.

Loading selected objects

Rather than replacing your current work, if you have previously saved selected objects as described above, you can load these and incorporate them into your current workspace. To do this, click the "Load Objects" button on the ImpressionCreator toolbar. Then select or browse to the objects you want and then click "Load". The objects will then be imported into your workspace.





4

Enhancing your objects

4.1 *Naming your object*

The first and most important thing to do is to select the “Summary” tab of the Object Properties and then set the “Object ID”.

This is a unique identifier for the object. It is crucial to assign a unique name to the object for a range of reasons. Any time you deal with object interaction, all types of which we will come onto later you need to identify the object you want to interact with so it must be identified using the “Object ID”.

Even before you get onto object interaction you will quickly understand the need for this when you try to use the “Object Navigator”. This is a tool that allows you to see all the objects within your workspace, along with their properties.

To open the Object Navigator, click “List Objects” on the ImpressionCreator Toolbar.

The Object Navigator will show you all your objects, relationships they have with other objects, position and size, z-order and images used. It is not designed to tell you everything, but rather a summary reference that you can use to identify and tweak objects. There are certain elements of an object's properties that you can edit directly via the Object Navigator for speed; the object position, width, height and comments.

You can select objects using this interface by clicking on them. Use the SHIFT and CTRL keys to select multiple objects. You can also right click an object to access its properties, just as you would by right clicking the actual object on screen.

The row selections on the Object Navigator are synchronized with the actual object selections on screen. This means that at any time you can clearly see which object you are working on.

Imagine the chaos if you didn't use the Object ID property and all the objects were called <unassigned>!



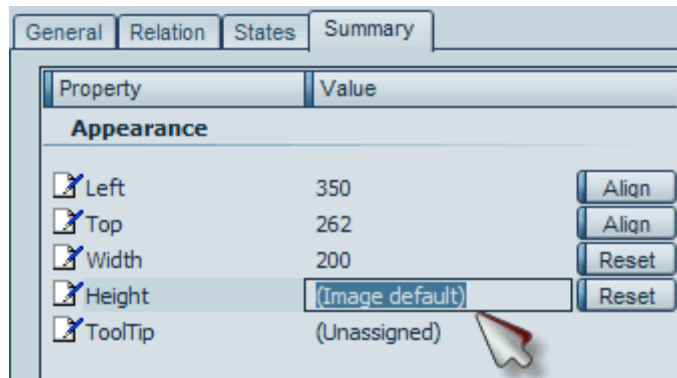
Object ID	Parent ID	Group	Widget	Class	Script	X	Y
Blue_Star_BG		Full Widget	Blue_Star1	Layer		336	180
ButtonBall7	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		353	167
Close_Button	ButtonBall7 (Cont)	Full Widget	Blue_Star1	Layer		5	5
ButtonBall4	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		453	480
Small_Logo6	ButtonBall4 (Cont)	Full Widget	Blue_Star1	Layer		4	4
ButtonBall3	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		454	391
Small_Logo5	ButtonBall3 (Cont)	Full Widget	Blue_Star1	Layer		4	4
Button2	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		302	297
Text2	Button2 (Cont)	Full Widget	Blue_Star1	Layer		60	25
Small_Logo2	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		457	309
ButtonBall2	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		453	305
Small_Logo3	ButtonBall2 (Cont)	Full Widget	Blue_Star1	Layer		4	4
Button3	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		297	384
Text3	Button3 (Cont)	Full Widget	Blue_Star1	Layer		61	23
Button4	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		255	471
Text4	Button4 (Cont)	Full Widget	Blue_Star1	Layer		32	24
Button1	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		313	211
Text1	Button1 (Cont)	Full Widget	Blue_Star1	Layer		48	23
ButtonBall5	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		78	55
Blue_Star_Logo	ButtonBall5 (Cont)	Full Widget	Blue_Star1	Layer		31	32
ButtonBall6	Blue_Star_BG (Cont)	Full Widget	Blue_Star1	Layer		178	370



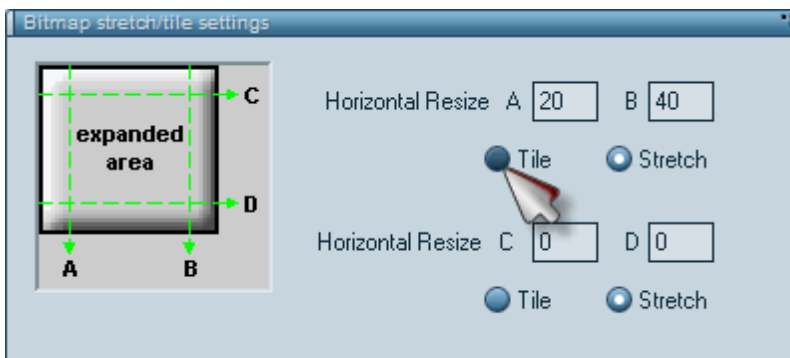
4.2 How do we scale the object?

Scaling images

Once you have selected the image file, the object will automatically be sized to the size of the source image, but you can distort the image by entering a width and height for the object. To do this, open the “Object Properties” for the object and adjust the dialog and adjust the “Width” and “Height” boxes on the “Summary” tab.



ImpressionCreator is not content with just allowing you to size objects in such a brutal fashion, so provides an “Advanced” button on the “States” ... “Appearance” tab to allow you to resize the image more intelligently.



Clicking this button brings up a new dialog.

This allows you to specify different scaling horizontally and vertically. At each edge you can specify a number of pixels that you do not want to stretch and then you can choose to either Tile (repeat) or Stretch the area in between.

For example, if you have an image that is 40 pixels by 30 pixels and you want the outer 5 pixels at each edge to remain unstretched, then the corresponding values A-D would be 5, 35, 5, and 25. You could then choose to either tile or stretch the area in the middle.



4.3 Grouping and how naming is involved

When developing you will soon realize that many of the best objects are created by several objects working together. For example, consider a simple “button with text on it” made up of a single image. Whilst this may be a good object, it’s not very flexible. For example, if you want another similar button you have to create an entire new graphic.

Now, consider creating that button from a background image and a separate text object. All of a sudden it is much more flexible as the text element can easily be changed and all the above problems are gone.

One problem that remains however is the fact that you need to ensure that these objects work together as a single unit. Groups, Parents and Children make this very easy.

If you want several objects to work together is to assign them to the same “Group”.

Groups

An object’s Group is defined on its “Summary” tab. Any predefined groups are available in a combo box so you can allocate an object to that, or you can simply type in a group name to create a new group. Alternatively you can select one of more objects, right-click and select “Group” from where you can get the same options.

Once a group exists, objects within that group share some basic functionality.

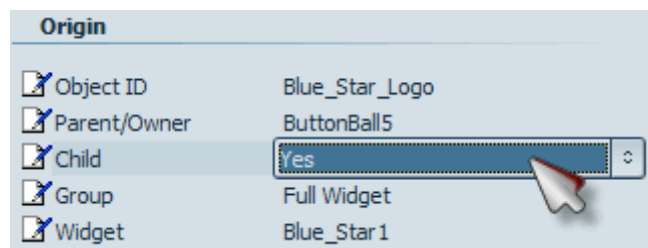
All objects in a group move together, so as you drag one, all other objects in the group move at the same time. Note that you can hold down CTRL when dragging to just move that single object.

Also, as you will see shortly, popups work well with groups. All objects in a group popup or hide together.

Note that you can remove an object or objects from a group by clearing the content of the “Group” field in its properties.

Parents and Children

“Parent / Child” relationships are slightly more intimate in that an object is more directly affected by changes made to its Parent/Owner. Again, in the same “Summary” section there is a “Parent/Owner” drop-down list where all the other objects in the workspace are listed. Simply select one of them and your current object becomes a child of that object.



If a Parent is hidden or moved then the same is applied to objects that it “owns”. Owned objects are also shown higher in the z-order than (above) its Parent. You might initially think that this is



excessive when Groups work in almost the similar way. Consider however, that this saves so creating many named groups to manage where they are unnecessary, it provides increased control over the z-order and also that it provides increased flexibility to control visibility within a group without affecting the whole group.

One added option you have for an object that is owned is to make it a “Child”. This means two things. The first is that its position is defined relative to the Parent object not the screen. This makes positioning within the parent easier. Also, the object is only visible while it is in the confines of the Parent object. Note that this is based on the actual rectangular size of the object, not defined by the visible area.

“Children” also make ImpressionCreator more efficient and therefore should be used wherever possible. When we discussed the creation of images earlier, we mentioned that you may want to split a button into two objects, the image and text; so that you could use the base image for other buttons. This is a prime example of where you may want to use this by making the text a child of the button image.



4.4 Object relations

You've already learnt about how the objects you create can interact with the user, and will now learn how they interact with each other.

ImpressionCreator allows you to tweak how they interact with users, but also how they interact with the desktop and other applications on screen.

To access these settings, open up the "Object Properties" dialogue and then click the "Relation" tab.

Visibility

The first thing you can specify is whether the object is visible or not. You may wonder why you would want to hide objects. There are two main reasons. The first is that the object may be doing work in the background such as loading a webpage which you don't want visible. Alternatively, you may only want the object to be visible at certain times which are controlled by popups which we will come onto later.

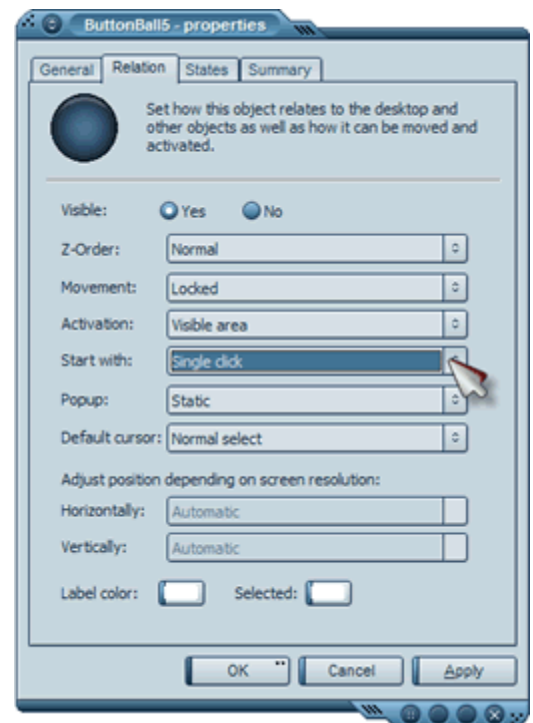
Z-Order

If you consider a desktop, then all objects on there have an "x" and "y" position which defines where they are on the screen. What you might not consciously have thought about is their "Z-Order". This defines which object or application appears "on top" of the other when they overlap. Objects with a higher Z-Order appear on top of those with a lower Z-Order.

You have three Z-Order options for your objects. The first is "Desktop". This means that running applications will always appear above your objects. They will open on the desktop. Those interacted with most recently (clicked/moved), will be on top of other objects.

A "Normal" Z-Order means like the object will behave like a mini-application. Objects with a "Normal" Z-Order appear above or below applications and other "Normal" Z-Order objects depending on which were interacted with most recently.

An "Always on Top" Z-Order means that this object will always be visible above all applications and objects. It functions in the same way as the "Always on top" setting of the default Windows taskbar.





Movement

“Movement” can be defined as “Normal” or “Locked”. If it is “Normal” then the user can drag the object around the screen. If “Locked”, this is not possible. You will typically want locked objects in your application, except for maybe the background.

Activation

“Activation” defines how the user can interact with the object. If the setting is “Default” then it simply follows the user’s settings, but you have three other options. “Rectangular” activation is based on the size of the source graphic and ignores any transparency in the object, whereas “Visible Area” means that only those areas of the object visible on screen can be interacted with. The final option “None” means that the user cannot interact with this object and any attempts to do so are ignored. This is usually most important for cosmetic objects on the desktop.

Start with

“Start with” defines the way in which a user interacts with an object. It is not always the case that you want a user to have to click an object to activate it. In addition to the standard “Default” option – which responds to a “Single click” as defined in the ImpressionCreator settings “Defaults” tab - you have four more ways in which a user can interact with the object. Sometimes for example you may simply want the user to move over an object for something to occur.

Default Cursor

This option allows you to specify which cursor will be displayed when the mouse moves over the object. This is a very useful way to indicate to the user the effect of their interaction with the object, for example if you want a user to drag and move the object rather than clicking it then it would make sense to use the “Move” cursor.

Popups

OK, now onto “Popups”. These are a really cool feature in ImpressionCreator that we briefly mentioned in the Object Types section.

Earlier in this section we discussed visibility. You don’t always want objects to be visible on the screen, and in some cases you may want them to be “Popups” whose appearance on screen is controlled by other objects that have the “Object Controller” type.

If the object is referenced by an object controller then it will function as a popup depending on the value set here. If the object is not used as a popup in this way, then this value is simple ignored.

You should note that when an object is called as a popup then all children of that object or objects in the same group will also inherit the popup functionality of that object. This is because popups are just really a shortcut for the visibility setting. By this, I mean that popups are essentially a dynamic



way of setting the visibility of the object. Visibility is a property that is shared by group members and children inherit it from their parent.

Here are the popup types.

Static popups - Normally, an Object Controller will toggle the appearance of a popup (open it or close it depending on whether it already is visible). Once these “Static” popups are opened they must have an Object Controller that explicitly closes them. Think of them like a normal program. Clicking a link to a program will open it, but clicking the link again will not close the application. A separate “Object Controller”, the close button, is required to do this.

Toggle popups - Are the simplest types. If the popup is hidden when you click the Object Controller it will be shown. If the popup is visible when you click the Object Controller it will be hidden.

Menu popups - These popups once displayed, will only remain until the user interacts with the desktop in anyway. They have this name because they function just like menus; unless you click them immediately to carry out the function they are there to perform then they disappear.

Volatile popups - Once a “Volatile” popup is shown, it will remain until either the controller is toggled or another popup is opened.

Volatile (No toggle) popups - This is just like the previous version, except that toggling the Object Controller is disabled like in a “Static” popup, so the only way to make it disappear is to activate another popup.

4.5 *Transparency*

The “Transparency” tab on the “Object Properties” dialog “States” tab allows you to make the whole object partially transparent (in addition to any transparency specified in the image).

By default you have a “No transparency” option, or you can specify the percent opacity of the image. In the example shown the object is be 40% visible (60% transparent).



The final option allows you to vary transparency as the object animates that will be discussed in the Advanced Object Effects section.

4.6 *Rollover, Away, Down states*

You now know the basics of configuring a single state object. A multi-state object allows you to have the object react differently under different circumstances, and is easy to achieve.

To add a new state, simply click the “Add” button on the “States” tab. You will then be presented by a list of the common states not yet used from which you can select:



Mouse over - occurs when the user moves the cursor over the active part of the object (as defined by the 'Activation' option on the "Relation" tab

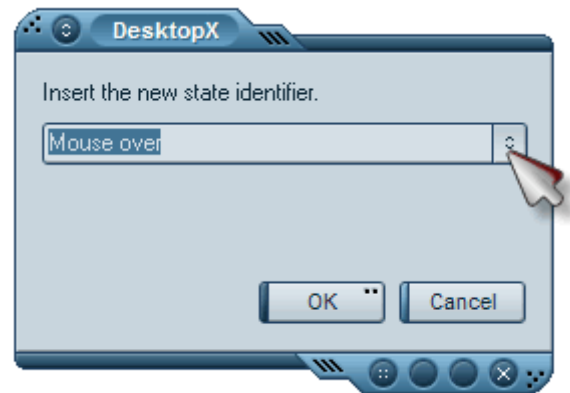
Mouse down - occurs when the user presses down the left mouse button over the object

Mouse up - occurs when the left mouse button is released whilst over the object

Show - occurs when the object appears on screen

Hide - occurs when the object is removed from the screen

Command executed - occurs when the object is activated as defined by the "Starts with" option on the "Relation" tab



You can then configure this state just as you did for the first one. One thing to remember is that all the images used must be exactly the same size.

The other buttons in this section should be fairly clear as to what they do. The "Remove" button removes a created state though you obviously need to have one state for an object.

Once you have spent time setting up the first state, the likelihood is that other states will be refined versions of that state, so for speed you can use the "Copy From" button to copy the settings from another created state to the currently selected one.

The "Reset" button clears all modifications in the currently selected state so you can start again.

When you create a state the drop-down list shows you the states that exist by default in ImpressionCreator, but you aren't just limited to these. In the drop-down list box you can just type any name in and a "Custom State" of that name will be created which can be configured just like any other. Though these states are not triggered by user interaction they can be activated by Messages we will come onto later and are very useful for that purpose.

The one final thing to define in a multiple state object is the "Default Appearance". On the "General" tab you can specify the image to be used when the object first loads. This will not change until the object is interacted with in some way. Note that this image must be the same size as all the other state images.



4.7 Messages

Sending messages in ImpressionCreator is a really simple concept to understand and simple to achieve, but it is a very powerful function.

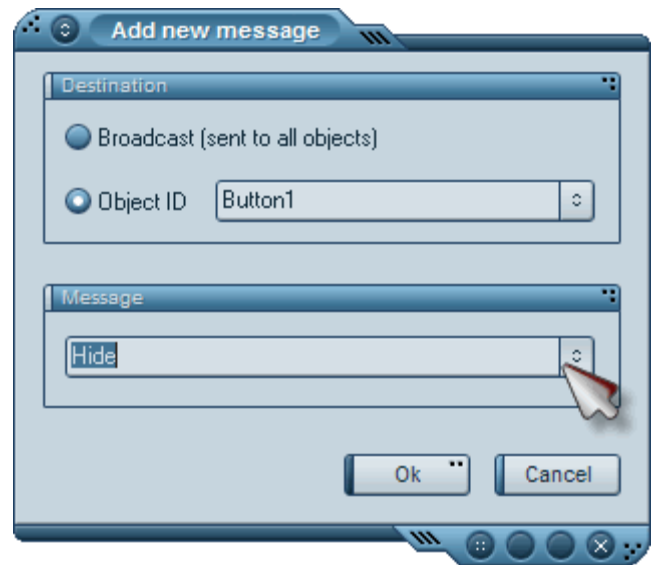
In summary, sending messages means that when a user interacts with one object, cause one or more different objects to react. Consider a light in your room. When you flick the switch it changes state, but the light bulb also changes state. When you flick the switch, a message is sent to the light bulb which causes it to change state.

Sending messages is very simple. Simply, go to the “States” tab of the object you want to send then message from, select the state which you want to occur in order for the message to be sent (e.g. Command executed) and then click the “Messages” tab.

Once you have done this click the “Add” button to pop up the messages dialogue.

There are two things to define in this dialogue; the “Destination” and the “Message” to send.

The “Destination” is the object to which you want to send the message. For example in the example above the object to which we would be sending the message is the Light Bulb.



You have two options here. The first is to send the message to every other object in the workspace. You should use this option very carefully as obviously in a large workspace this could cause several objects to change state at once and you may end up with a massive cascade of state changes, or even a loop you can't get out of if objects send messages to each other.

The more likely option is that you will be sending a message to a specific object, in which case you should select that option and choose the object from the list.

The second thing to define is the message itself. The drop down list will show a list of the default messages from which you can select. Alternatively, if you know of a custom message that has been set up for the destination object you can send that message by typing in the name of that message.

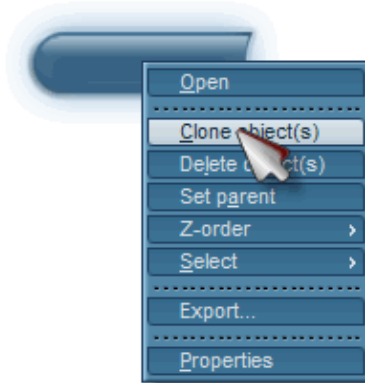
You can add as many messages as you want and an object can send different messages in different states - e.g. it may send one message when the mouse is over an object, and a different one when the mouse moves away from the object.

That's it - sending messages is that simple and the results can be really impressive. The first thing you may want to use this for is to change some text when the user moves the mouse over a button to explain the purpose of that option. We'll see this used effectively in the Tutorial.



4.8 Copying objects

As you develop, you will often find that you want several similar objects such as buttons. Rather than creating an entirely new object you can quickly copy it using the “Clone” option. To “Clone” an object, right click it and select “Clone object(s)”. This will duplicate all selected objects.





5

Advanced Object Effects

5.1 Adding animation

Whilst everything you learnt in the previous section was great, there's nothing that catches the eye like movement, so ImpressionCreator makes it easy for you to add animation to your objects. The "animation" tab is where you can define how your object animates.

The first thing to explain is that there are the two different types of animation that ImpressionCreator can do.

A "Static animation" is the most simple animation option. This simply changes the level of transparency rather than the image itself. You may for example want a button to fade in as the user moves their mouse over it and fade out as the mouse moves away.

Slightly more complex is a "standard" animation with different images over time.

Static Animations

We will start with a "Static animation". To start with, go to the "States" ... "Animation" tab and set a number of frames. A frame is an individual part of an animation, so the higher the number of states the more different "images" an animation has. Once you set this value higher than "1", the "Interval" is enabled. This is how long each part of the animation lasts in milliseconds. The higher the value the longer the animation takes. For example, a 5 frame image with an interval of 50ms will take 250ms (1/4 of a second). You also need to check the "Static animation" checkbox so that ImpressionCreator knows not to expect an animated image, just to vary transparency.

If you check the "Loop" box then the animation will be repeated as long as the object is in that state. Normally if a state change is triggered then the animation will play through to the end before changing state, but if you check the "Interruptible" box then to state will change immediately without playing to the end.

After that you simply need to set the direction in which the animation will fun by selecting one of the 4 options available, such as "Forwards".

If you are looping the animation it will probably make sense to choose one of the bottom pair as this will lead to a nice smooth pulsing transparency effect.



The final thing that you need to do is specify the change in transparency. This is where you get to use the third option on the “Transparency” tab. Here you specify how the transparency varies in one cycle of the animation. This will then be impacted by the animation style you chose and whether you chose to “Loop” the animation.

Try experimenting with these options and see how they change the appearance of the object.

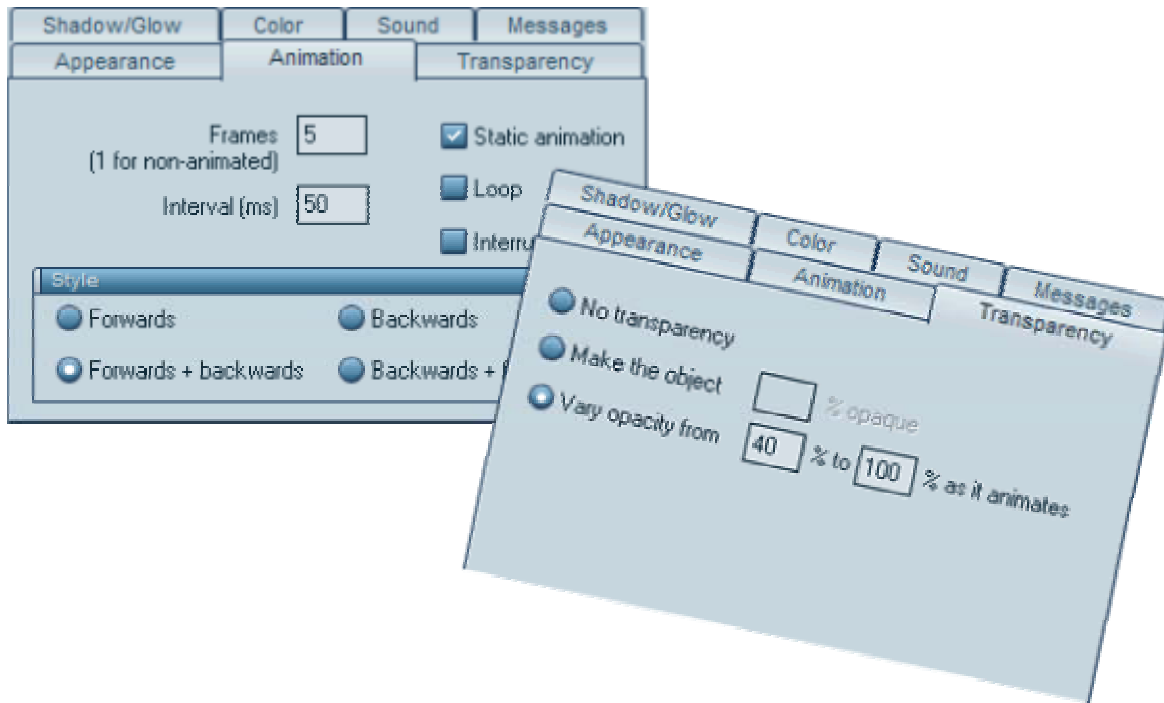


Image animations

We will now look at actually animating the image itself, rather than just animating the transparency. Though I said that this is more complicated, it still isn't actually that difficult. The change you need to make is to combine all the frames of the animation into your source image.

What you need to do is place the frames of the animation side by side from left to right.



The above is the source image for a six frame animation of the ImpressionCreator logo being worked on in Fireworks. There are six small images next to each other in one large image.

Just select this image, make sure that “Static animation” is deselected and that the number of frames specified is the same as the sections of your source image. Then if you run the above animation forwards then the logo will appear to rotate.



Once you have done this then you are free to change the other animation settings as you were shown for the “Static animation”. This includes transparency so you can change both the image and the transparency over the course of an animation.

5.2 Shadows/Glows

The “Shadow/Glow” tab on the “States” tab of the “Object Properties” dialog allows you to add a shadow or glow to the object. If you check the “Enabled” box then you can set parameters to get the effect you want.

If you want a “glow” you should set both X and Y offsets to 0, but if you want a shadow change the offsets to represent the number of pixels distance away from the object you want the shadow to appear.

The “Sharpness” is how fuzzy the shadow or glow is, and the “Darkness” specifies how solid the shadow or glow should be.

The final thing you can edit is the color of your shadow or glow. This is simply a case of choosing a color from the color picker that appears when you press the “Change color” button.



5.3 Color changing

One of the coolest features of ImpressionCreator is the ability to change the color of things without replacing the image. In the “Color” tab you can adjust the objects hue (color) by selecting a value between 1 and 255. Experiment until you find a shade you like! Setting the value to 0 will set the object color to that of its original image. You cannot recolor a greyscale (black and white) object.

You can further fine tune the color by tweaking its brightness and contrast across the specified ranges. All the below images are recolored clones of the left object.

You saw earlier how to create additional states and here there is a button here to quickly apply the change you have made to all different states of the image.





5.4 Sound

The final tab that we will cover in this section is the “Sound” tab. Here you can specify that a sound is played when the state occurs simply by selecting one that’s already being used, or browsing to one on your hard drive. You may want to do this for example when the user clicks a button.

There is also a range of options for looping the sound. You can choose to play the sound once, repeat it for as long as that state is occurring, or synchronize with the object’s animation if it exists.

5.5 Adding Tooltips

If you look at the “Summary” tab of the object, you will see a range of other parameters that you can change.

Here, you can tweak the objects position and size as well as assigning a “Tooltip” message that appears when you hover over the object.



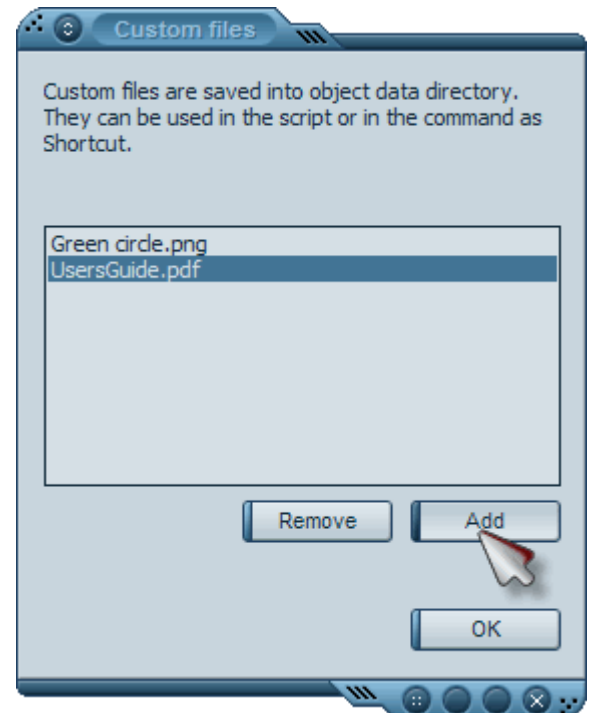
5.6 Adding Custom Files

In the “Summary” tab, there is a button labeled “Custom Files”. This allows you to attach files to objects just as you would do to email that you send. When a package containing Custom files is loaded they are automatically extracted to a temporary directory so they are accessible from there.

The first reason that you may wish to do this is to attach fonts to a package. If you have text objects that use a font that you are not sure a user has on their machine then you can attach them as a Custom File. ImpressionCreator automatically installs any fonts that included with a pack, so then you text objects will appear as intended.

Note that you have the file author’s permission or that the file is not subject to any copyright or licensing agreement which does not permit distribution.

There are many other potential uses for Custom Files. For example, if you attach your brochure as a PDF you could create an object that links to it using the shortcut “%objectdir%\Brochure.pdf”. The options are unlimited.





6

Tutorial

In this section we will go through the creation of an installer step by step to get you started, and then in the following section we will show you a few extra techniques you may find useful.

All the source images you need for this section are in the “/Samples/Tutorial” directory where you installed ImpressionCreator.

6.1 Getting Started

To start from a clean slate, right click the ImpressionCreator icon in the taskbar and select “New Workspace”. This will reset the project and give you a blank desktop. Below is in image of what we are going to create.



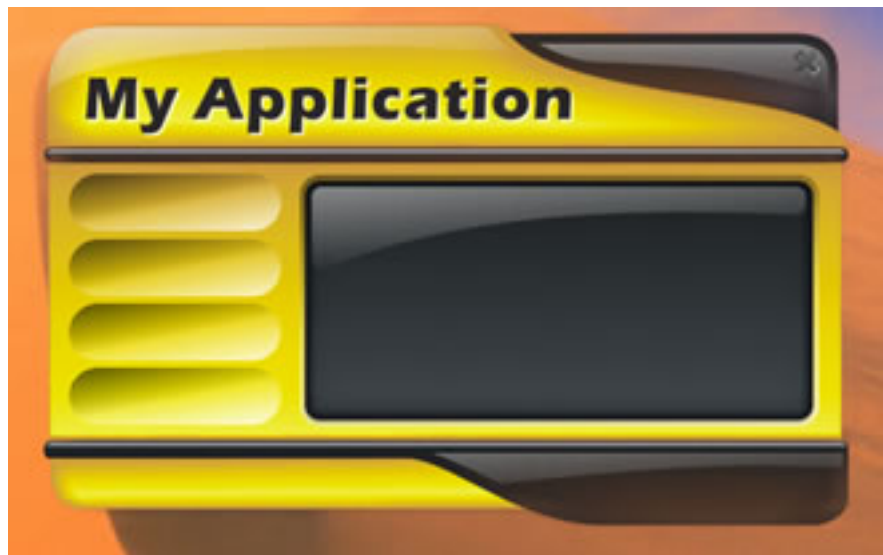
This is a fancy shaped installer with a shadow. A close button appears in the top right corner when you move over it. The text at the bottom is a web link and the 4 buttons all change color when you move the mouse over them and change the context of the content in the main area.

This may sound like a lot, but don't worry, we're going to do things one step at a time!



6.2 *Creating the installer background*

- 1) Before you get started, click the “New Workspace” button on the ImpressionCreator Toolbar to ensure you are starting from a blank canvas. If you have previously done any work that you want to save, do so when prompted.
- 2) On the ImpressionCreator Toolbar click “New Object”. A default object will be created on the desktop, and the Properties dialog for that object will appear.
- 3) Go to the Summary tab and set the “ObjectID” to “Background”. All other objects will be children of this object so we need to be able to refer to it.
- 4) On the “States” tab, go to the “Appearance” sub-tab. By default the object will be defined as an image which is great, but we now need to “Browse” to find the image we want to use. Select the “SplashBackground.png” file and click “Open”. Click “Apply”, and you will see that your default object now looks like an installer as shown below.



- 5) Lets add a bit of a special effect; a shadow to make the installer look cooler. Go to the “States” tab, and select the “Shadow/Glow” tab that is on there. Click “Enable” to specify that you want a shadow. You can then experiment with the parameters to get a shadow you like. Because the object is quite large I’m going to increase to two Offsets to 10. You can click Apply to see the effect. I’m quite happy with that, but you can experiment with the options until you get something you’re happy with and click “OK”.
- 6) Finally, we should move the object to the correct position, so, drag your object to the centre of the screen where you would want it to appear when loaded, and then, open its properties again and click on the “Relation” tab. At the bottom of the tab you should change both the position adjustments to be “Center”. This means that if the user of your application is



running at a different resolution the installer will stall appear at the center of the screen. Once you have done this, click “OK”.

6.3 *Creating the Close button*

You will note that the background has a placeholder for a close button in the top right corner. Here we are going to add a button that fades in as you move the mouse over it and fades away as you move the mouse away.

- 1) Create a new object and give it the name “CloseButton”.
- 2) Set its parent as Background and make it a Child. You will probably get a warning that it will be set outside the parent and therefore not be shown. That’s because the current position is outside the object. Just click “OK” to accept this for now.
- 3) If you see that the position has negative values, change them to positive ones (e.g. 100,100) so that the object appears within the background object. In the future when you create a new object, if you drag it over the background before making it a child you will not get this problem.
- 4) On the “States ... Appearance” tab “Browse” to select the “SplashClose.png” file and Apply the changes. You can now drag the object to its correct position within the close placeholder on the background. You can always fine tune the position in the “Summary” tab. Coordinates of 561,21 should be right.
- 5) We need to set the function of the button, so in the “General” tab, change the type to “Widget” ... “Close”. Click “OK” to accept this.
- 6) On the “Relation” tab set the “Movement” to “Locked” so people can’t drag the button and set the “Activation to “Visible area”. This will mean that the button action and effects we are about to create only occur when the cursor is precisely over the “x” of the button. You should also set “Start with” to be “Single click” to ensure it functions like a regular button rather than requiring a double click to work.
- 7) We want an effect where the image fades in from 0% opacity to 100% opacity when the mouse is over it and the reverse when away. We have a “Mouse away” state so let’s go to the “Animation” tab and set the effect. Let’s say we want 5 graduations of transparency so enter “5” in the frames box. Increase the interval to 100 milliseconds. 5 frames at 100 milliseconds per frame means that the entire animation will take half a second which sounds OK. Now, set the “Static animation” flag so ImpressionCreator knows that we are just changing transparency and not the image itself.
- 8) Next, move onto the “Transparency” tab and check the third option to vary transparency as it animates. Enter “0” in the first box and “100” in the second. This defines the range that the transparency will move through. Remember that this is the Mouse away state so we actually



want to go from 100% to 0% opacity. To do this return to the “Animation” tab and set the state to “Backwards”. You may wonder why we didn’t just set the range from 100 to 0 instead, but this will become clear in the next step.

- 9) Now we need to create the “Mouse over” effect. To do this, click the “Add” state button. On the follow dialog “Mouse over” will already be selected in the list so just click “OK”. This state is basically the same as the “Mouse away” state apart from the direction of animation, so rather than defining everything again we can just click the “Copy From” button and click the option to copy from the “Mouse away” state. Now, all we have to do is set the Style to “Forwards” on the “Animation” tab.
- 10) Click OK, and move your cursor over and away from the placeholder and you should see a red cross appear as you do so. The effect of clicking this will not be clear until you create your application, but it will close the application down.



6.4 *Creating a web link*

Here we will create a text object that changes color on mouse over and when clicked launches a webpage.

- 1) Create a new object and call it “WebLink”. Again, make it a child of the “Background”.
- 2) Open the Properties, and go to the “Appearance” tab. Rather than being an “Image” change its type to “Text”. Type the title for the link such as “www.stardock.com” and set a font and color. I’m using “Century Gothic” Bold, Size 14, and keeping the color as black.
- 3) Click “Apply” and drag the text object into position on the yellow area at the bottom left of the background.
- 4) Create a new “Mouse over” state and copy it’s settings from the “Mouse away” state just as you did before. Now, you can change the color in the “Mouse over” state to be something different so that the user gets feedback on their action. I’m going to use a dark red.
- 5) On the “Relation” tab again, lock “Movement”, and also set “Start with” to be “Single click”. This time the “Activation to “Rectangular area”. Because text has intricate shapes if you use a visible area you will get a constant flickering between the two states. A rectangular activation works much better.
- 6) OK, we’re pretty much there but I can’t help wanting to add another flashy effect. It would look cool if there was a glow around the text when the mouse was over it. To do this, go to



the “Shadow/Glow” tab in the object’s “Appearance” section. Make sure the “Mouse over” state is selected and then enable the effect by checking the box. Then set the offsets to “0” to create a glow rather than a shadow. I want a bright glow rather than a dark one so I’m setting the color to white using the “Change color” button. Click “Apply” and then move your mouse over the link to see how it’s working. Maybe it’s a bit subtle so let’s put the “Darkness” up to 255 and the sharpness to 10. Click “Apply” and see what you think. I like it so I think we’re done for this section, so click “OK”.



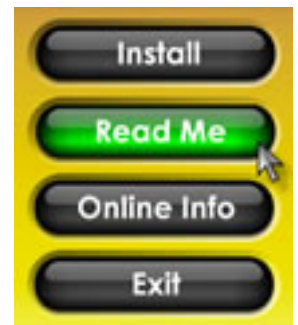
6.5 *Creating the basic buttons*

On the left of the screen we want 4 buttons for the various user options. The plan is that they will all be black with white text, but then on “Mouse over” they will change color. We will make the first three turn green and the fourth one, which will be to exit the application, will turn red.

- 1) Create a new object, call it “Button1” and set its appearance to use the “SplashButtonBlack.png” file.
- 2) Create a “Mouse over” state and apply the “SplashButtonGreen.png” file to that state. You now have a basic button.
- 3) Move it into position in the top recessed area on the left of the background. You now can set some of the parameters we previously did. Make it a child of the “Background” object and “Lock” its movement and set it to “Start with” a “Single Click” on the “Visible area”. Click “OK”.
- 4) What we need now is a text label for the button. Create a new object (“Text1”) and make it a text object rather than an image object. In this example let’s make the text “Install”. To be consistent I’m going to use a Century Gothic font, bold in size 14. I’m also making the font white and centrally aligning it. “Apply” these changes.
- 5) Move the text so that it is centered on the button you just created and make it a child of that object (“Button1”). On the “Summary” tab next to the “Left” setting there is an “Align” button. Set that to “Center”. This ensures that if you were to change the text it would be centralized on the button.
- 6) Now, “Lock” the movement of the object and set its “Activation” to “None”. This is because we don’t want clicking on the text to actually do anything and anytime the user clicks the text it should be ignored so that the button underneath is clicked. Click “OK”.



- 7) We now need three more buttons. Rather than going through the above steps again, there is a shortcut. If you right click the button with your mouse you can choose to “Clone object(s)”. Click this and then you will be prompted to “Clone group and children objects”. We want to clone the text as well, so click “Yes” to do this. Your new button will be created on to of the original. Normally you could drag this to its new position, but we have locked its movement. There is a way to do this however. Hold down the “CTRL” key and then drag the object into position over the second recessed area. Repeat this two more times until you have 4 buttons.
- 8) Now what's required is to change the text on the buttons. Again, with there being no activation you need to hold down “CTRL” whilst right clicking to edit its properties. You can then change the text to, for example, “Read Me”, “Online Info” and “Exit”.
- 9) The final thing we said was that the bottom button was going to be red on “Mouse over”. We could create a new image, but ImpressionCreator creates an alternative. If you select the “Mouse over” state or the “Button4” object and then the “Color” tab. Here we can do a “Hue Shift”, which adjusts the color of the source image. Try alternative values and click “Apply” (not “Apply to all states” – because we don't want to colorize the “Mouse away” state). For example, 80 will turn it Blue. We want Red, and a value of around 160 gives us the color we need. Click “OK” and the button basics are done.



6.6 *Creating some effects for the buttons*

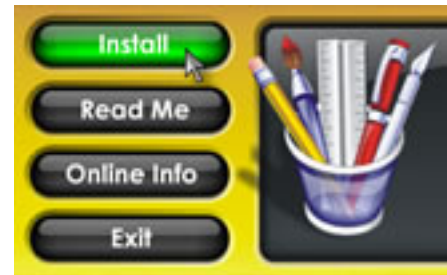
In this section we are going to add some effects so that when the cursor is moved over a button, an appropriate image and some explanatory text is shown.

- 1) Create a new object for the image and call it “SplashImage”. The name here is going to be important so double check it.
- 2) The way we are going to create our effect is through the use of “Custom States”. First of all, select the “Mouse away” state, and click the “Remove” button. Acknowledge the confirmation, and then click “Add” to create a new state. We need a state where no icon is visible to represent when the mouse is not over any buttons. Overwrite the “Mouse away” text with “Blank” and click “OK”. You will see that a new state has been added to the list of States. Set the image to “SplashImgInstall.png”. It may seem weird to do this but it will be explained in the next step.
- 3) You can now move the image into position at the left side of the main black area. Make this object a child of the Background, lock its position and “Activation” to “None”. Now, create a state called “Install” and copy the “Blank” state. Now for the explanation of why we have an image in the “Blank” state. All images for different states have to be exactly the same size. We could create a totally transparent image and use that but instead we can just use the



image we have and set the opacity to “0” so it can’t be seen. Therefore, ensure the “Blank” state is selected, go to the “Transparency” tab and “Make the object 0% opaque”. Click “OK”. The object won’t be visible, but that’s why we moved it into position earlier.

- 4) OK, so now the image is ready for interaction with the button, but now we need to edit the button to change the appearance of this image, so open up the properties of the “Button1” object. Go to the “Messages” tab of the “States” section.
- 5) Messages allow you to trigger a state in another object when a state change occurs in the source object. What we want is to set the “SplashImage” state to “Install” when the mouse is over the button, and to “Blank” when the mouse moves away from the button so let’s do that now. Ensure the “Mouse over” state is selected, and click “Add” in the “Messages” section. You want to sent a message to one object, so check “Object ID” rather than “Broadcast” and choose the “SplashImage” object from the drop down list. Then, from the “Message” drop down list, type “Install”. Click “OK” and the message will appear in the list of messages for that state.
- 6) Now, select the “Mouse away” state and send the “Blank” message to the “SplashImage” object just as you did above.
- 7) Click “OK” and try moving the mouse over and away from the “Install” button. The image should appear and disappear.



- 8) Now we want to repeat this process for the other buttons. It’s now difficult to select the image on screen because it isn’t visible so we’ll use another way. Right click the ImpressionCreator icon in the system tray and select “List objects” to display the “Object Navigator”. Then double click on “SplashImage” from the list. The object properties will appear. You can now create the states “ReadMe”, “OnlineInfo” and “Exit” using the appropriate images “SplashImgReadMe”, “SplashImgOnlineInfo” and “SplashImgExit”.
- 9) You can now add messages to the “Mouse over” and “Mouse away” states of the other three buttons. For example – the “Mouse over” message for “Button2” will be to send the “ReadMe” message to the “SplashImage” object, and the “Mouse away” message to “SplashImage” will be “Blank”.
- 10) We also want a “Text” object to provide additional feedback. This will have exactly the same states as the image, so we can recreate this quickly by creating a clone of the “SplashImage” object. To do this, in the “Object Navigator” right click the “SplashImage object. You will get a list of options just like right clicking the object itself. This object will now appear in the



“Object Navigator”. Double click it to open its properties, and change its name to “SplashText”. You now need to go through the objects and add text to each state.

- 11) Select each state in turn, and on the “Appearance” tab, change the type to “Text”, set the “Alignment” to “Center”, set the font (e.g. Century Gothic, bold, 12 point, white) and enter some appropriate text remembering to use line breaks as appropriate. For example:

Blank: xxxxx

Install: “Click here to”, BREAK, “install My Application”

ReadMe: “Click here to”, BREAK, “read the latest”, BREAK, “release notes”

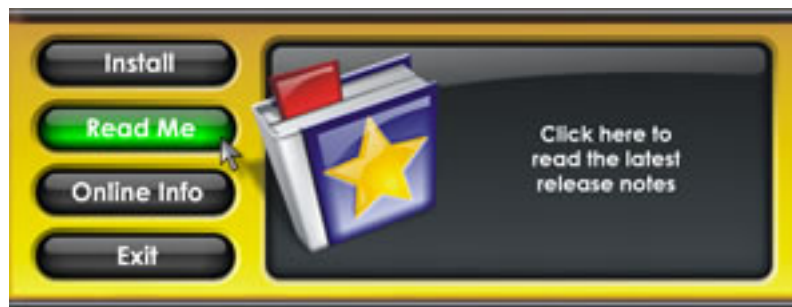
OnlineInfo: “Click here to”, BREAK, “visit our website”

Exit: “Click here to”, BREAK, “close the installer”

Note that rather than redefining the font every time you could copy parameters from the “Blank” state once you have defined that.

Also, ensure that there is no transparency used so the object is visible. Click “OK” once you’re done.

- 12) Now you need to add some more messages to the buttons. Go through each button in turn, adding messages to send a “Blank” message on “Mouse over” and a message the same as we did for the image on “Mouse over”. This is where using the same state names helps make things clearer. Once you have done this, Click “Apply” then you can position the text to the right of where the image appears. This is why we added some dummy text to the blank state. Also, go to the summary tab, and you can click both the x and y “Align” buttons to center the text on the coordinates. This will make the text look much better. Once you are happy with the position you can clear the “xxxxx” from the “Blank” state and click “OK”
- 13) You should now have four buttons that when moved over change an image and some text to something that relates to the button itself.





6.7 *Adding functionality to the buttons*

Visually we are pretty much done now, so we just need to add the functionality required to the buttons which is really easy.

- 1) Display the properties for “Button1” and look at the “General” tab. We want this button to run a specific application. You can “Change” the “Type of object” to “Shortcut” which gives you a range of options. Obviously in this demonstration there isn't an actual installer to run, so for now select one of the applications installed on your machine from the drop down list or “Browse” for a file. If you are going to be running a file from the install media you can use the %exedir% variable as described on Page 17 of this guide. Click “OK” when you have selected an application.
- 2) Display the properties for “Button2”. Here we want to run a file that is associated with an application, so again, change the type to “Shortcut”. Here you have two choices. If you want you can force the user to view the file in a specific application such as Notepad. If you want to do this, select Notepad for the “Shortcut”, but then add a link to the file you want in the “Arguments” section. For example “readme.txt”.
There is a better option however. It would be better to allow the user to launch the file in the application they prefer to use. For example, some users may use WordPad to read .RTF files, but others may use Microsoft Word. To solve this, enter a file extension in the shortcut rather than an application; e.g. “.RTF” or “.TXT”. linking to a file directly in the shortcut also allows me to use variables, so I can point to a location on the install media; for example “%exedir%/readme.txt”.
- 3) For “Button3” you can do as you did for the web link earlier. Simply set the type to “URL” and specify the page you wish to browse to; e.g. www.impressioncreator.com
- 4) “Button4” is a repeat of what we did for the close button, so set the type to “Widget” ...”Close”. All the buttons are now set to carry out the functions they are intended to do.

6.8 *Finishing touch*

There is one final thing to do to make this work like a regular application. Currently the application sits on the desktop but we need it to gain the focus like a regular application.

- 1) Right click your “Background” object and select “Properties”. Select the “Relation” tab. Change the “z-order” to normal. This means that your application will get focus just like regular applications. You may force the application to be “Always on top” but this isn't behavior a user would expect or appreciate from an installer.
- 2) That's it – we're ready to export the application now!



6.9 Exporting the application

All we need to do now is make this an application that users can run. This is a very easy process.

- 1) Drag your mouse in a rectangle around your objects to select them, and then click “Export” on the toolbar. On the dialog that appears state that you want to export “Selected and related”, as obviously you want to export all the objects, not just the background. Select “Export as an application” and click “Next”.
- 2) You then fill in a range of items as described in Chapter 7. You can use the “ic.ico” file for the icon, and ensure you set it to run on the Taskbar. Click “Next”
- 3) Set it to be a simple deployment. There’s no need for attach a read me file, as our Application does that. Obviously you can select any “.txt” file if you like just so you can see how this would work. Click “Next” again.
- 4) Acknowledge the “License agreement” and then choose a location to save your application. Once you have done this, click “Finish” and your application will be created.

Congratulations! You have just created your first great impression – now let your imagination run wild!

