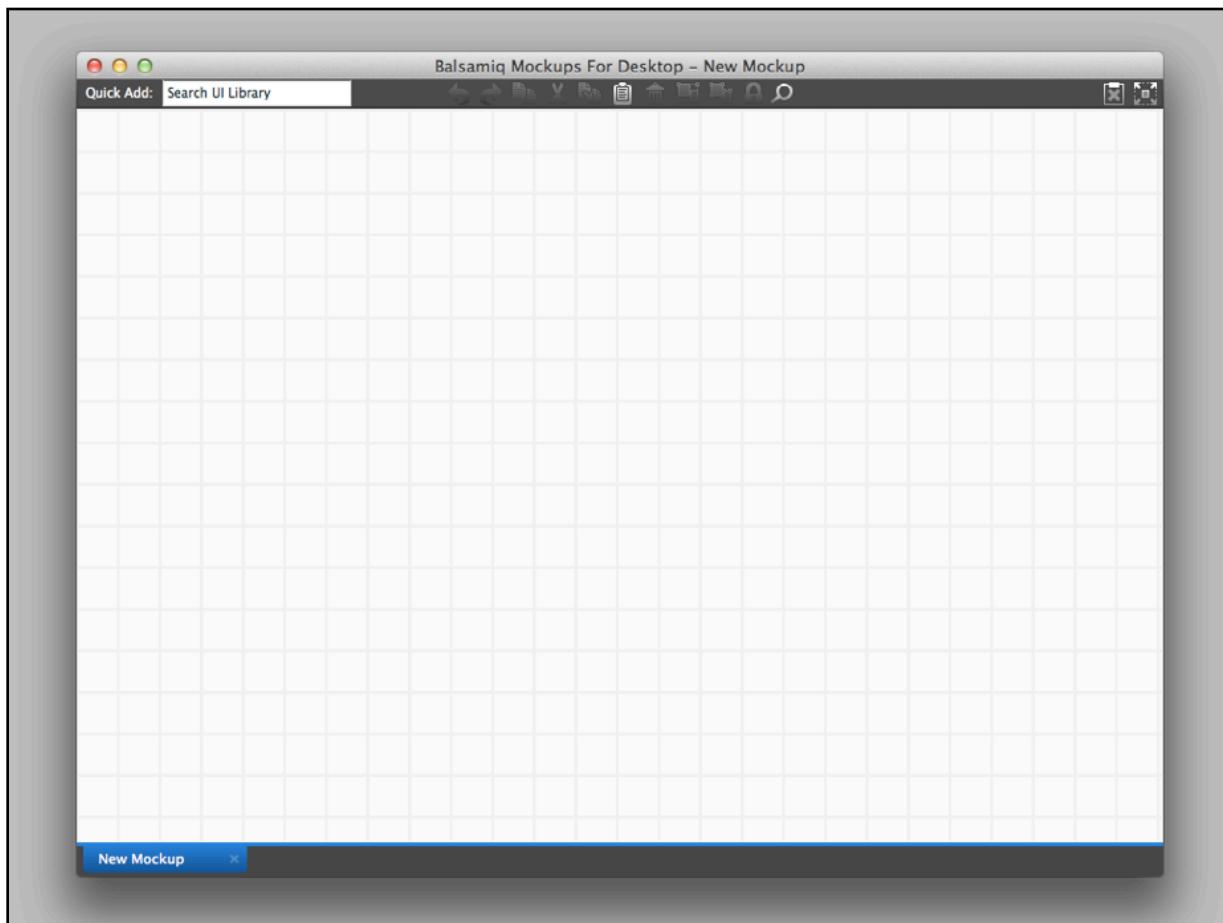


Balsamiq Mockups

Documentation

User manual for Version 2.1



By Balsamiq
Last updated 3 November 2011

balsamiq

Introduction	1
What This Document Covers	1
Viewing Online	1
Getting Support.....	1
Managing Projects	2
One Project = One Directory	2
Naming Convention to keep mockups sorted.....	3
The Project Assets folder.....	4
Sharing your Projects.....	6
More ways to collaborate	7
Looking Ahead.....	8
Application Overview	9
The Application Bar.....	9
The UI Library.....	11
The Mockup Canvas.....	12
File Browser.....	12
Working with UI Controls.....	15
Adding UI Controls.....	15
Selecting UI Controls	15
Moving UI Controls	15
Snapping and Smart Guides.....	15
Resizing UI Controls.....	16

Layering UI Controls.....	16
Deleting UI Controls.....	16
Grouping UI Controls	16
Available Controls.....	17
The Property Inspector.....	18
Common Inspectors	19
Other inspectors.....	20
Need More info about Property Inspectors?.....	25
Working with Text.....	26
Basic formatting.....	26
More Macros	27
Working with Data Grids / Tables.....	29
Column Width Options	29
Using Icons in Datagrid.....	29
Creating Line Returns in Datagrid.....	29
Pasting Data from Excel	30
Working with Icons	31
Icon Search	31
The Icon Library	31
Adding your own custom icons.....	32
Working with Images	35

Copying Images to Assets.....	35
Working with Markup.....	37
Toggling Markup On and Off	37
Markup Toggling with Keyboard Shortcuts.....	38
Making Items Markup and Non-Markup	38
Exporting Markup to PNG and PDF	39
Working with Symbols	40
Creating Symbols.....	40
Under the Hood	41
Overriding Symbol Properties	42
Editing a Symbol's Source	46
2 Methods for Creating Symbols Libraries.....	47
Symbol Libraries.....	48
One more thing... Using Whole Mockups As Symbols.....	51
Sharing Symbols Across Projects.....	52
Deleting Symbols.....	54
Next Steps and Feedback.....	55
Keyboard Shortcuts	56
Working with Controls.....	56
Quick Add.....	57
Working With Icons.....	57
Editing Text.....	58

Working With Mockup Files	58
Other	58
Importing and Exporting	60
Exporting and Importing via XML.....	60
Exporting to an image	61
Exporting to PDF.....	61
Printing.....	62
Linking Mockups Together.....	63
Presenting Your Work	67
Sharing Images, Symbols and Icons Between Projects using Account Assets.....	68
Setting Up the Account Assets Folder in Mockups for Desktop.....	68
Using Account Assets	69
The BalsamiqMockups.cfg Configuration File.....	70
Saving your Config File	70
Sample File.....	70
Options.....	71
Command Line Options	72
Commands.....	72
Log	72
Additional Resources	73

BMML File Format Specification	74
Quick Overview.....	74
The Mockup Tag.....	75
The Control Tags.....	75
The ControlProperties Tag.....	76
The Component Control	78
Control Type IDs and Icon IDs.....	79
Custom Properties.....	80
Using Custom Properties.....	80
Get Notified of Developer News	81

Introduction

What This Document Covers

This documentation eBook covers version 2.1 of Balsamiq Mockups for Desktop. Some of the instructions in this manual may also be relevant to users of Mockups plugins and myBalsamiq.

The final chapters include topics for advanced users and developers interested in learning about command line options, the BMML file format, custom properties, and how to extend Mockups files for use in other applications. You can view a list of extensions that have taken advantage of our open format here: <http://support.balsamiq.com/customer/portal/articles/135659>

Balsamiq releases new versions of Mockups frequently—we shoot for a new release every two weeks. It's a good idea to check the download page at <http://balsamiq.com/download> to see if there's an update, because there may be new features, improvements, and bug fixes available.

Viewing Online

Chapters may include links to online content such as videos or example documents that are not embedded in this document. Blue underline text indicates a link, as well as right-pointing triangles like the one shown below. Click those links to view that resource online.

► View this documentation online: <http://support.balsamiq.com/customer/portal/topics/50003>

Getting Support

A searchable, online version of this document, as well as for myBalsamiq and plugin integration with the systems Balsamiq supports is also available is available at: <http://support.balsamiq.com>

For Tech Support, our GetSatisfaction forum is the best place to start. You can search, submit, vote, and usually will get a very quick response there.

Sales & Licensing: sales@balsamiq.com

Technical Support: support@balsamiq.com or <http://community.balsamiq.com>

Telephone: +1 (415) 367-3531

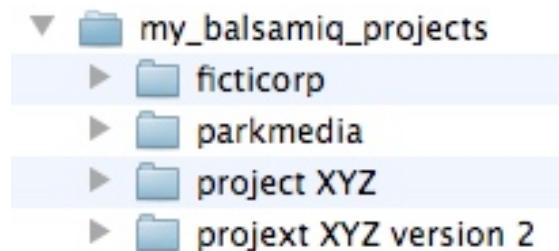
Managing Projects

Mockups for Desktop does not currently natively provide a way to bundle multiple wireframes together into projects.

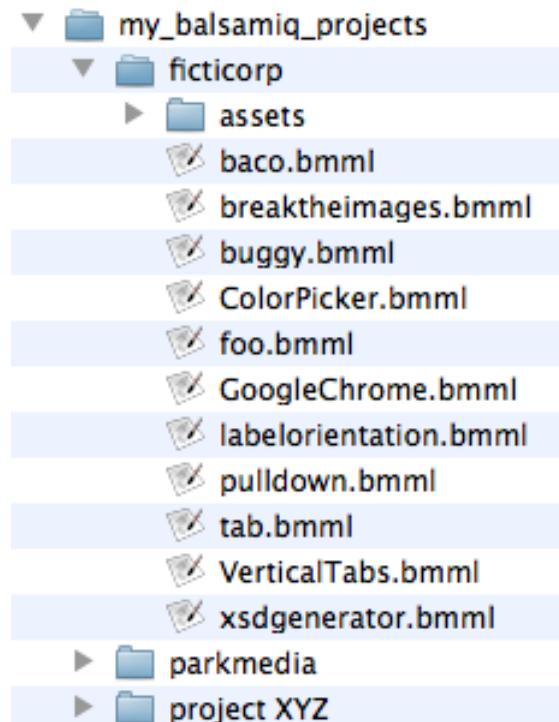
This document shows you how to overcome this limitation by using some best practices and other standard software tools already at your disposal.

One Project = One Directory

A best practice to organize mockups is to create a folder/directory for each project, as shown in the screenshot below:

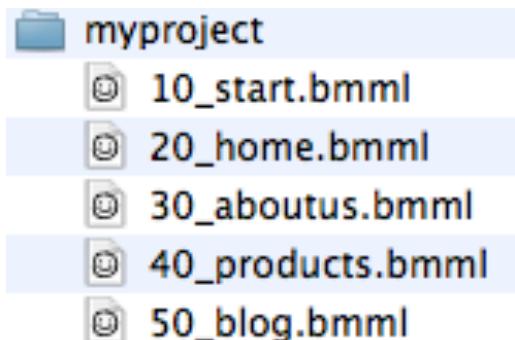


Within each project folder, we recommend storing all the BMML files together, like so:



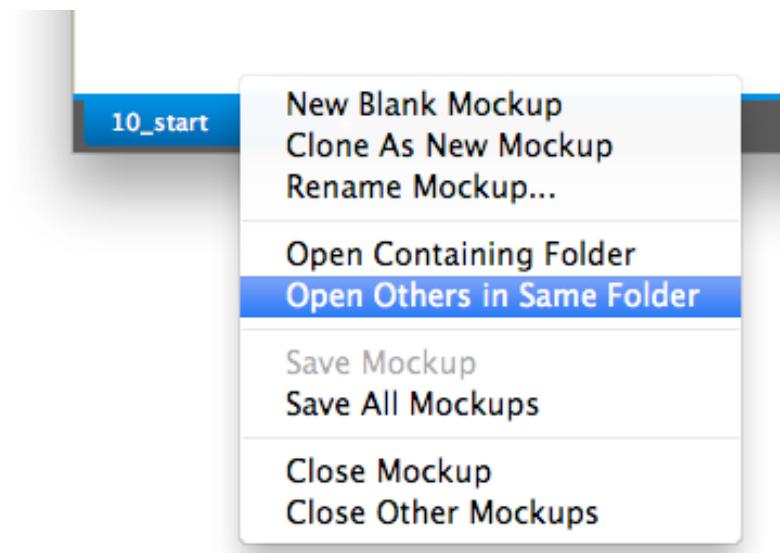
Naming Convention to keep mockups sorted

A best practice for keeping your mockups in order is to use a naming scheme like this one:

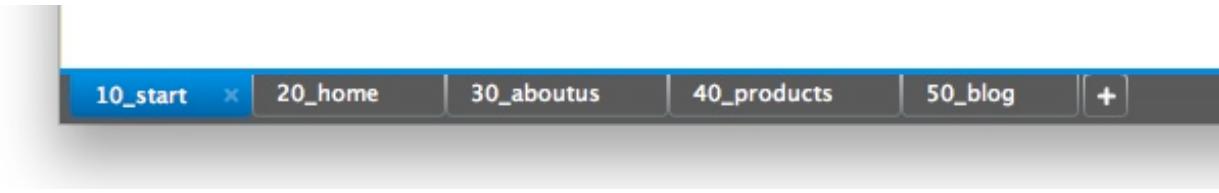


Skipping from 10 to 20 allows you to insert new pages in between later if you need to, without renaming everything (just call the new page 15_something).

Once you use the naming convention above, opening all mockups in order is simple: double-click on the first one to open it in Mockups for Desktop, then right-click on the tab and select "Open Others in Same Folder":



This will open all the mockups in your folder, in alphabetical order!

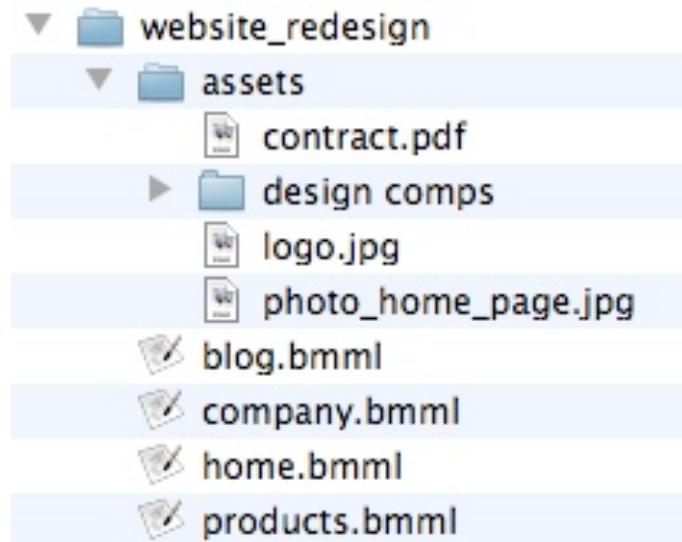


We realize this is a bit of a hack. Once we have full project support, this pain will go away. Hang in there! :)

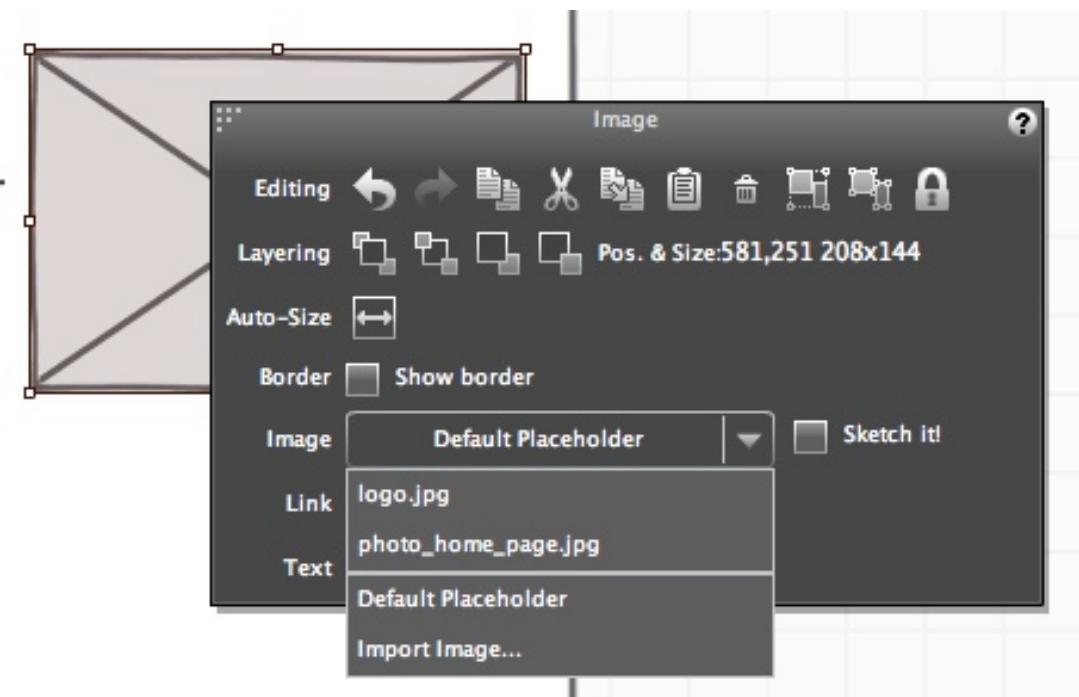
The Project Assets folder

We also recommend creating a subfolder called "assets", in which to store images that are used across your mockups.

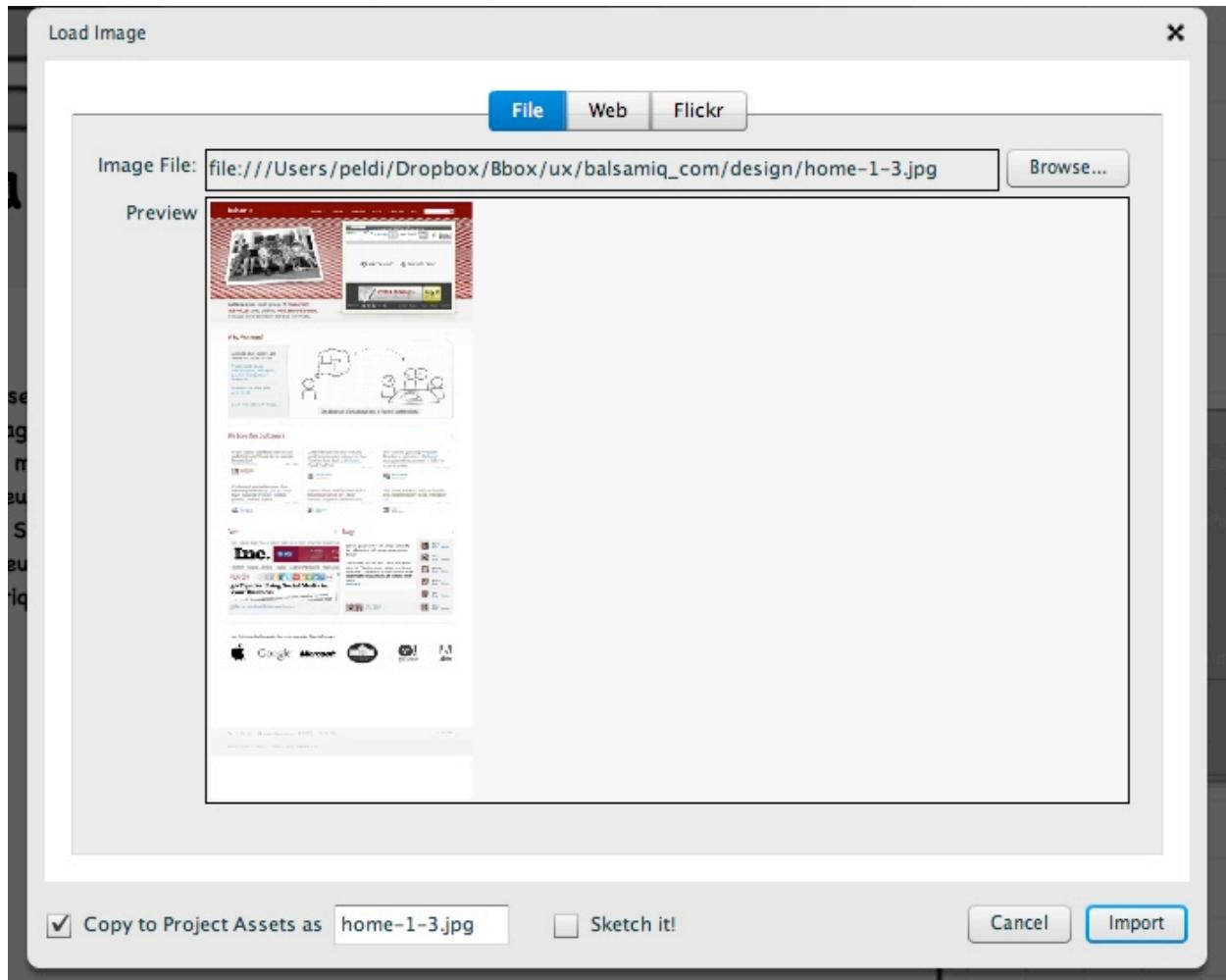
You may also use this subfolder to store PDFs, PSDs and any other file related to the project.



Mockups for Desktop is designed to work with the best practices outlined above. For instance, the software will list all the images it finds in the assets subfolder in the Image Property Inspector pulldown menu automatically:



Mockups for Desktop also gives you an option to automatically copy images you bring in from other sources into the assets subfolder for you:



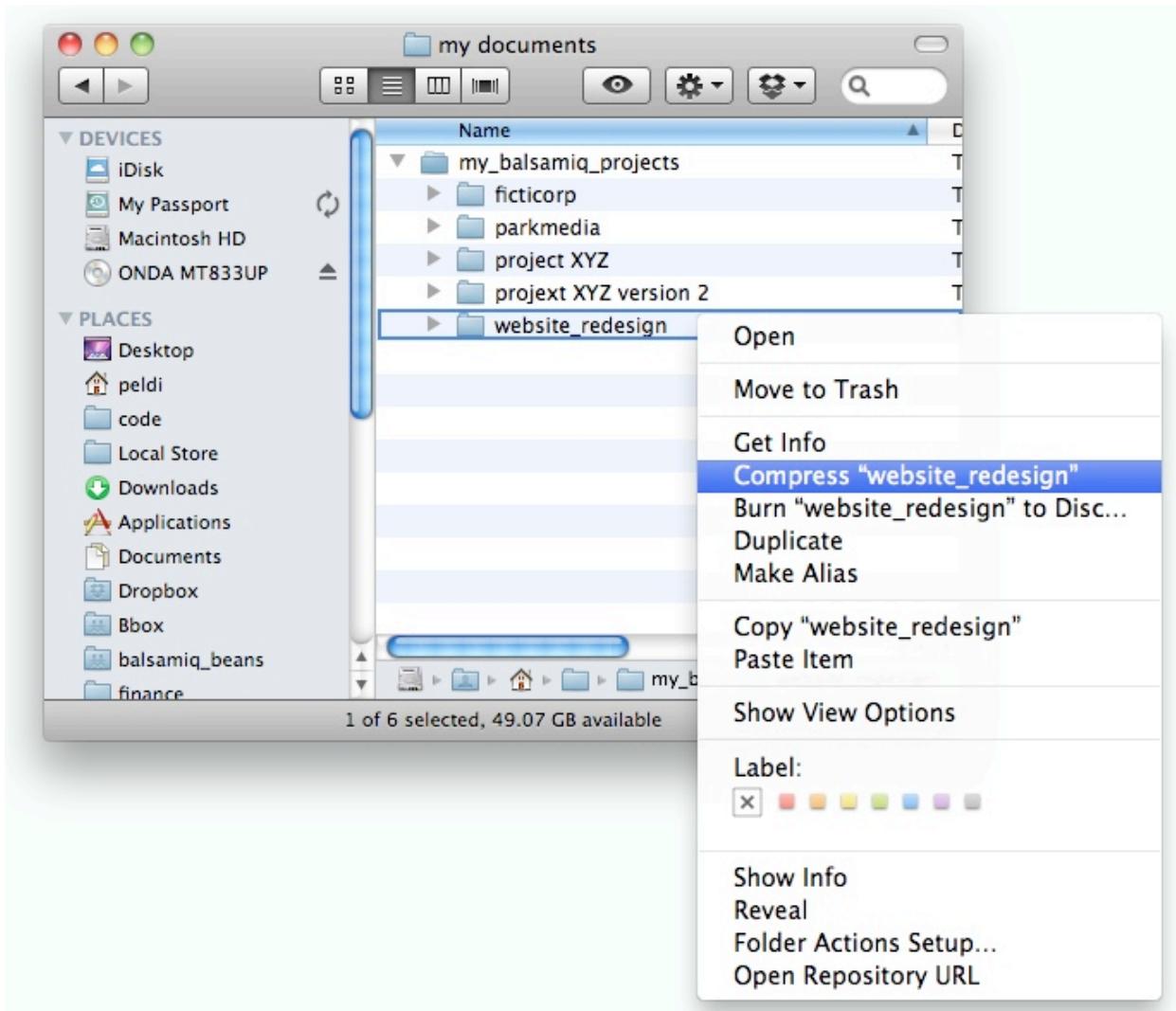
Balsamiq Mockups for Desktop also supports an Account Assets folder, useful for sharing assets between different projects. Read all about it here: [Sharing Images, Symbols and Icons Between Projects using Account Assets](#).

Sharing your Projects

When it comes to sharing your projects, the first question to ask yourself is: do they need to make changes to my wireframes, or just review them and comment? If the answer is the latter, perhaps exporting the project as a linked PDF is the best option. See this FAQ for other info about [how to create a "Mockup Viewer"](#).

If instead you'd like others to make changes to your wireframes, using the best practices outlined above will make it very easy.

To share a whole project, simply select the project folder and create a ZIP archive of it, using your archiving tool of choice (both Windows and OS X come with pre-built compression tools, in the right-click menu):



Once you have a zip ready, you can send it to other stakeholders by emailing it to them or save it on a shared disk, or using a file-sharing tool like [DropBox](#).

When your collaborators receive the project, they can just unzip the .zip file you sent them and get to work.

More ways to collaborate

We mentioned [DropBox](#) before. It's a free service (up to 2GB) which lets you share a folder on your computer with anyone you choose. It works extremely well and is what we at Balsamiq use to collaborate amongst ourselves and when working with external contractors.

Mockups for Desktop works extremely well when paired with DropBox. Here's a video that shows you how to quickly share your work with clients.

► Watch on YouTube: <http://youtu.be/adnMsMShTJw>

This is another that shows you how to achieve near-real-time collaboration on Mockups for Desktop by storing your files in DropBox:

- ▶ Watch on YouTube: <http://youtu.be/mnBUTkNo1vU>

Another option is to look at our [Plugin Versions of Balsamiq Mockups](#): they all provide easy ways to share and collaborate by storing your wireframes "in the cloud". All of our plugin licenses come with free Mockups for Desktop licenses as well.

We are also working on a web application called myBalsamiq. It supports projects natively and it aims to be the best way to do wireframing with remote teams. [Check it out!](#)

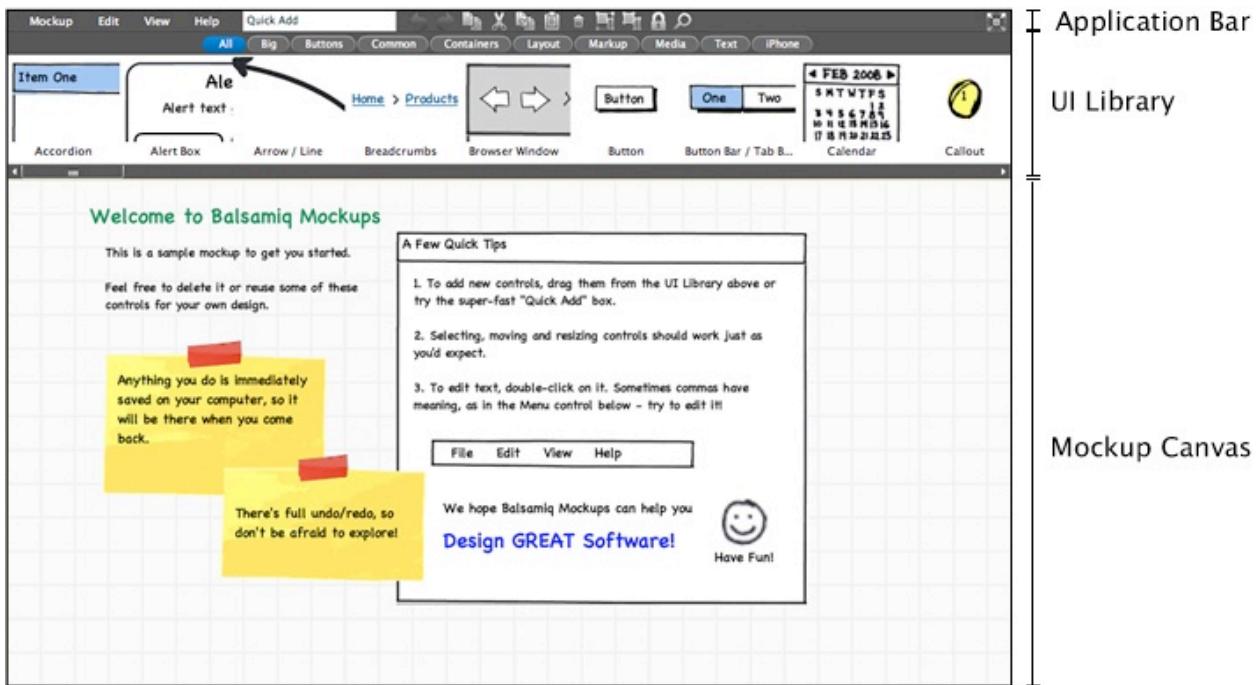
Looking Ahead

We hope this quick tutorial helps you better manage and share projects in Mockups for Desktop.

We plan on creating a new version of Mockups for Desktop which supports projects natively, but it will rely on the same folder structure outlined above, so if you start using it now, it will "just work" when Mockups for Desktop PRO(jects) comes out next year.

Application Overview

Balsamiq Mockups is made up of three major blocks. From top to bottom: the [application bar](#), the [UI library](#) and the [mockup canvas](#). Each is described below.



The Application Bar

The application bar includes [menus](#), the [Quick Add tool](#) and the [toolbar](#). Each is described below.



The Menus

The Mockups menus should be pretty self-explanatory. The only tricky bit is that the Mockup menu contains different items depending on what version of Balsamiq Mockups you are using. In general, that's where you'll find the About Box and commands related to your mockup as a whole, like saving, exporting to XML or PNG, etc.

File	Edit	View	Help
New Blank Mockup		CTRL+N	
New Clone of Current Mockup		Shift+CTRL+N	
Open...		CTRL+O	
Open Recent		>	
Save		CTRL+S	
Save As...		ALT+CTRL+S	
Save All		Shift+CTRL+S	
Close		CTRL+W	
Close All		Shift-CTRL-W	
Export to PNG		CTRL+R	
Export All to PNG...		Shift+CTRL+R	
Export All to PDF...			
Export Image to Clipboard		Shift+CTRL+C	
Export XML		CTRL+E	
Import XML		Shift+CTRL+E	
Print		CTRL+P	

File	Edit	View	Help
Undo		CTRL+Z	
Redo		Shift+CTRL+Z	
Cut		CTRL+X	
Copy		CTRL+C	
Paste		CTRL+V	
Paste In Place		Shift+CTRL+V	
Delete			
Duplicate		CTRL+D	
Select All		CTRL+A	
Select None		Shift+CTRL+A	
Group		CTRL+G	
Ungroup		Shift+CTRL+G	
Lock		CTRL+2	
Unlock All		CTRL+3	

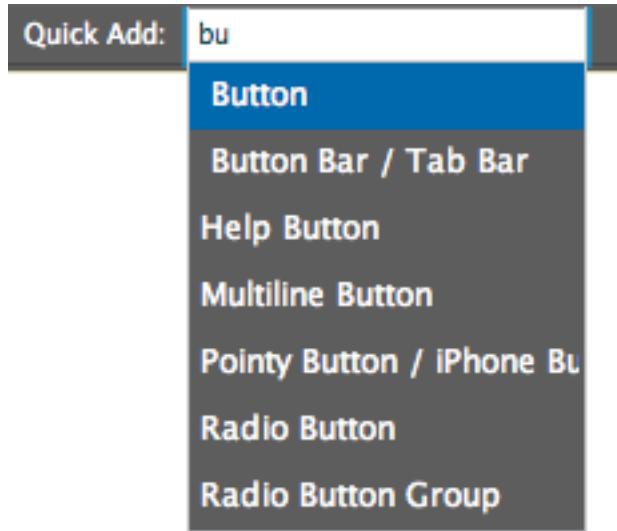
File	Edit	View	Help
✓ Show UI Library		CTRL+L	
UI Library Position		>	
Show Graph Paper			
✓ Show Markup		CTRL+K	
Zoom In		CTRL++	
Zoom Out		CTRL+-	
Actual Size		CTRL+I	
Zoom to Fit		CTRL+O	
Use System Fonts			
Full Screen Presentation		CTRL+F	

✓ Top
 Left
 Right
 Dock Inspector

File	Edit	View	Help
Online Help			
Balsamiq Mockups Support			
Download More Controls (MockupsToGo.net)			
Send Feedback			
Unregister this computer...			
I need inspiration! (inspireUX.com)			
What should I make for dinner?			

The Quick Add Tool

The Quick Add tool is the fastest way to add UI controls to your mockup. To use quick add, click inside the quick add input box (or use the / or + keyboard shortcut to enter it). Then you simply type a few letters from the name of the UI control you wish to add and Quick Add will show you a list of suggestions. Use your mouse or arrow keys to scroll down the list to the UI control you wish to add, hit the ENTER key and voila', a new control is placed on the mockup canvas for you. For instance, typing "bu" shows a list containing "Button", "Button Bar", "Help Button", "Radio Button" and "Round Button". Typing "hel", on the other hand, only returns "Help Button". Pressing the ESCape key makes the list disappear, as one would expect.



Try out Quick Add now! As you become familiar with it, you could even hide the UI Library (through the View menu) and simply use Quick Add to add UI elements to your mockup. This maximizes both your mockup canvas area and your speed!

The Toolbar

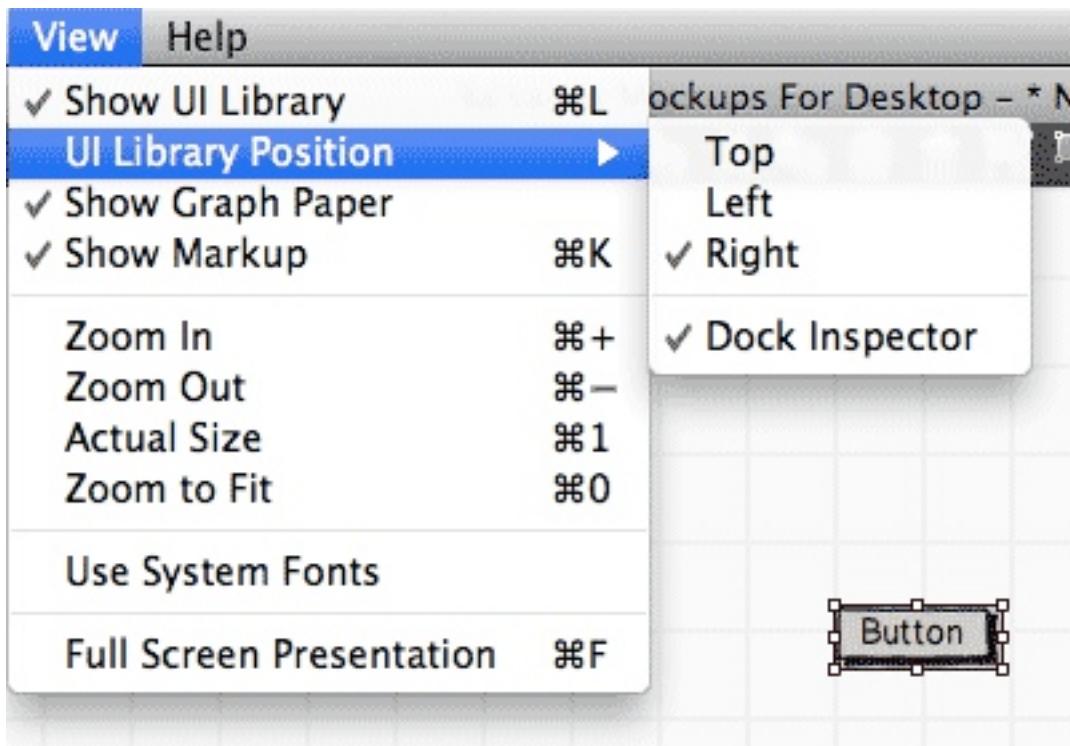
The toolbar includes most of the same commands as the Edit menu: undo, redo, duplicate, cut, copy, paste and delete. Not all versions of Balsamiq Mockups support the same keyboard shortcuts, so the Toolbar gives you a handy way to perform common actions. Notice that you'll find the same commands in the [Property Inspector](#) as well.

The UI Library

The UI Library, or UI Controls Library, is the long strip of UI Controls just below the Application Bar. It lists all of the different UI control types that Mockups supports, alphabetically. The main goal of the UI Library is to let you add UI Controls to the mockup canvas, but you can also use it to see what's possible and to get inspiration for your UI mockup.

To add a new UI control to the canvas, simply select the control type you wish to add, then you can either "drag and drop" it to the mockup canvas below or simply double-click and Balsamiq Mockups will place it on the mockup canvas for you.

The UI Library can be positioned to the top, left, or right of the application window using the menu, View > UI Library Position.



You can [view a list and screenshots of all of the components](#) that ships with Mockups.

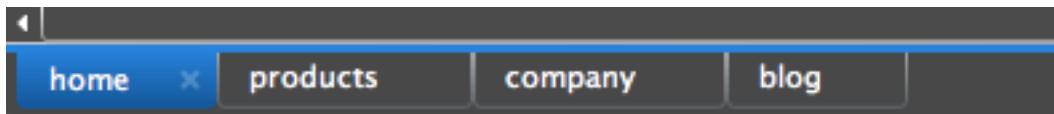
The Mockup Canvas

This is the main working area of Balsamiq Mockups, where your UI mockup comes to life. Once you add UI controls to it, you can move them, resize them and tweak them to your heart's content until your UI mockup is ready. See the sections below for instructions.

The mockup canvas grows and shrinks with your browser or application window, so you can make room for bigger mockups if you need to.

File Browser

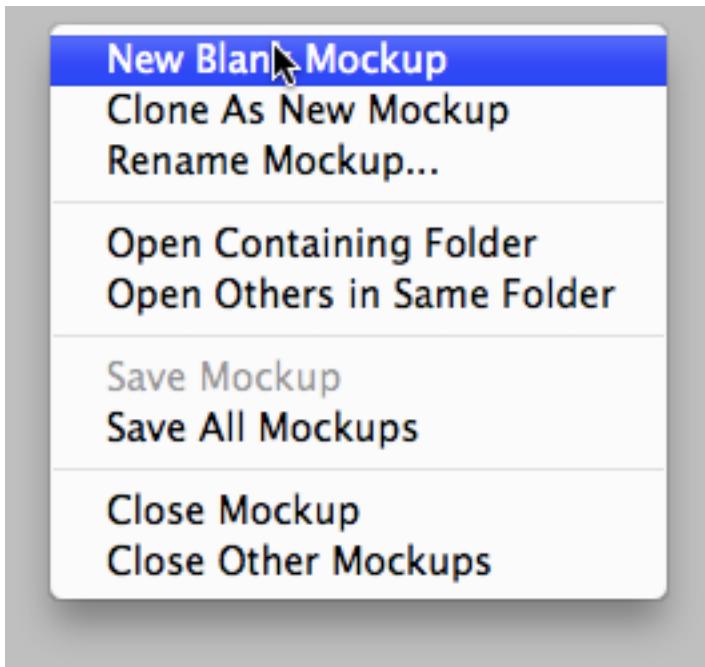
Below the canvas, at the bottom of the application window, you'll find the Mockups file browser. This bar displays the Mockups files you have open as a series of tabs, and the current Mockup in use is highlighted in blue.



When the tabs for the number of open Mockups cannot be displayed in the space available in the file browsing area, a pop-up menu is displayed to navigate the files.

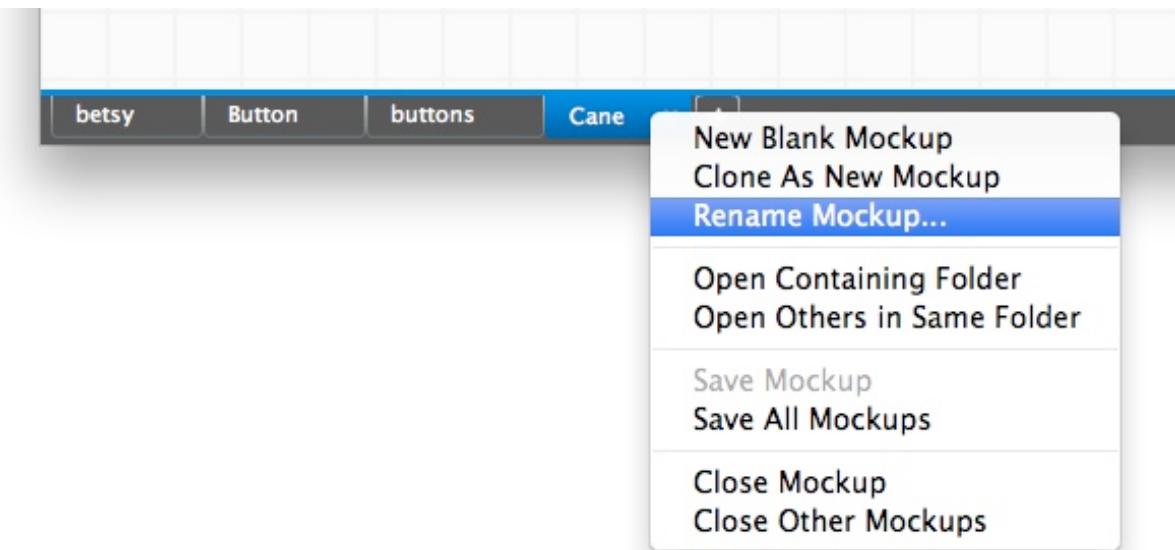
Context Menu

A context menu is also available for the files that are open in the file browser. Right-click the file to view available actions:

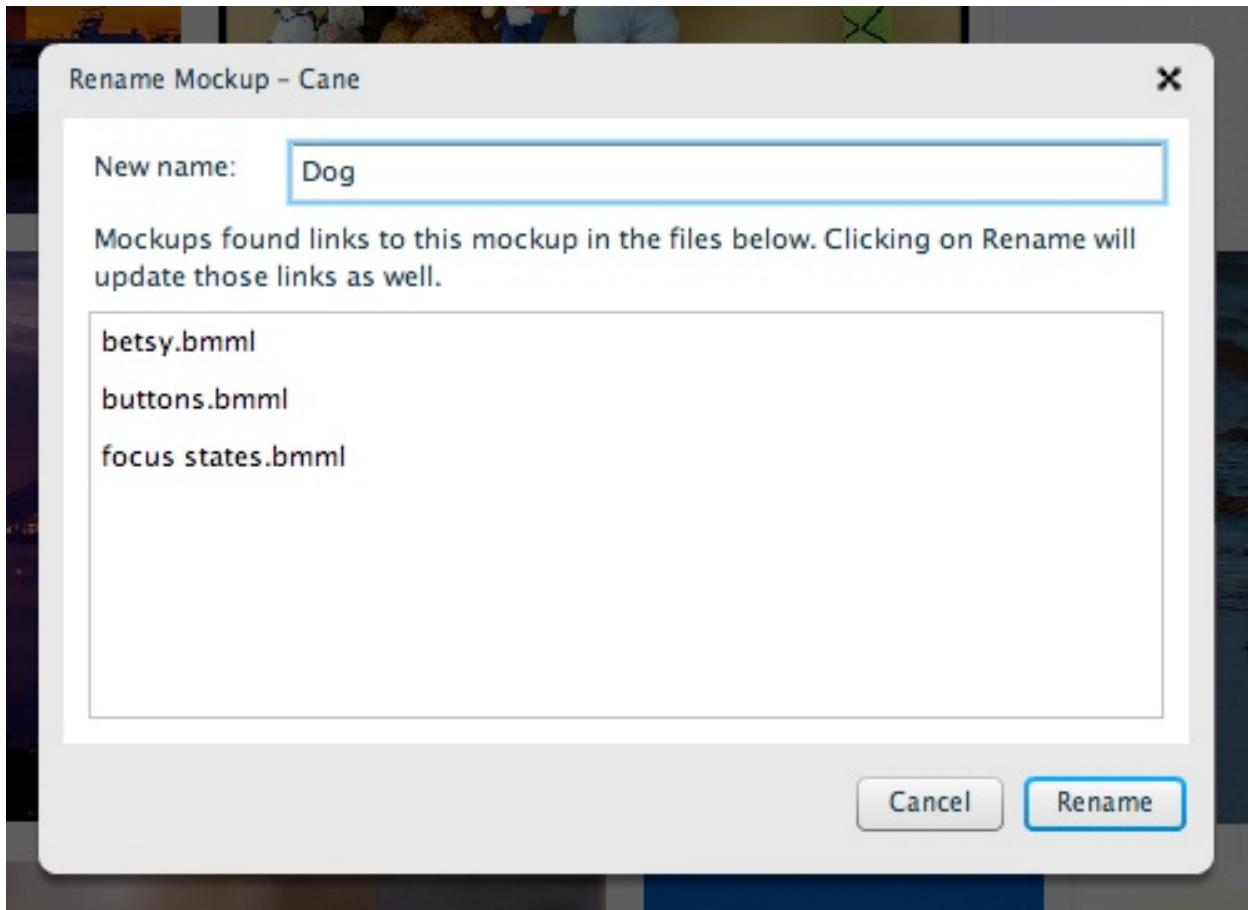


Rename Mockup

The rename feature will help those of you who use the [linking feature](#) extensively.



Just right-click on a file tab at the bottom of Mockups for Desktop and select "Rename Mockup...": the app will go through all the BMMLs contained in the same folder as the one you're renaming and look for links pointing to the current mockup. If it finds any, it will show you this:



Pick a new name, hit Rename and voilà, Mockups will rename your file and update all the links in all the other mockup files for you! This should save you a few hours here and there... ;)

Working with UI Controls

Adding UI Controls

To add new UI controls to your mockup, you can use [the UI Library](#) or [the Quick Add tool](#), described above.

Another way to add a new UI control is to duplicate the selected controls. You can either use copy and paste to do so, or the duplicate menu command or button, or simply hold down the ALT key (OPTION key on a Mac) and drag the selection. This will duplicate it in place and let you move it all at once.

Selecting UI Controls

There are a few different ways to select UI controls. If you are familiar with graphics software, these should be of no surprise.

Clicking on a UI control selects it.

Dragging a rectangle which encloses many UI controls selects them

SHIFT+CLICK on a control adds it to the selection

CTRL+CLICK on a selected control removes it from the selection

Hold ALT to ignore items behind the mouse and drag-select, as shown in [this short video](#):
<http://youtu.be/JOpaRBbwqNg>

Moving UI Controls

Once you have selected one or more UI controls, there are a few different ways to move it on the canvas.

you can drag it with your mouse (hold down the SHIFT key to maintain the original horizontal or vertical position)

you can "nudge it" a few pixels with your keyboard's UP/DOWN/LEFT/RIGHT keys

you can "nudge it" with bigger steps by holding down SHIFT and using your keyboard's UP/DOWN/LEFT/RIGHT keys

you can move your controls to be aligned in one direction by using the align tools found in the [Property Inspector](#).

Snapping and Smart Guides

When you move objects on the canvas, we try to suggest alignment relative to other objects on the canvas using smart-guides. This makes alignment a little easier and makes your Mockups generally look neater.

If you ever want to temporarily disable this feature, you can hold down the CTRL key (CMD key on Mac) while moving or resizing, and snapping will be turned off. If you want to turn this feature off entirely, you can do so by altering your [config file settings](#).

Resizing UI Controls

Resizing the selected controls is easy. You can just grab any edge or corner of the selection rectangle and drag it.

If you want to constrain proportions while dragging from a corner, hold the SHIFT key while dragging.

You can also use the keyboard to resize selected controls. CTRL+ALT+arrow keys to resize in 2px increments, CTRL+ALT+SHIFT+arrow keys for 20px increments.

Another way to resize some controls to their "natural size" is to use the Auto-Size function found in the [Property Inspector](#).

Layering UI Controls

You can layer controls as if they were pieces of paper on the mockup canvas. To do so, you select the controls you want to layer and select one of four layering commands available via the [Property Inspector](#): bring to front, send to back, bring forward and send backward.

Deleting UI Controls

To remove some UI controls from the mockup canvas, select them and hit the DELETE key. Alternatively you can click on the trashcan button in the toolbar or select "delete" from the Edit menu.

Grouping UI Controls

Sometimes you might wish to group some controls to better align them or to move them all at once. To group a set of controls, select them and hit CTRL+g on your keyboard. Once grouped, the controls will behave as one when moving or aligning. To ungroup controls, you hit CTRL+SHIFT+g on your keyboard. Groups can be nested.

Starting with version 1.8, you can double-click on a group control to "enter it" and edit its contents.

Here's a video that explains how to use the feature:

► Watch on YouTube: <http://youtu.be/OgX27oUrkLY>

You can optionally give a group a name via the group's property inspector. This can be helpful when you have a lot of groups and need to "know where you are" when you edit their contents.

Wait, no rotation?

Coming from generic drawing tools, the lack of a rotation feature in Mockups might be surprising.

This limitation is intentional, and not due to programming complexity (it's not hard to add technically). The reason we don't support rotating controls because we believe that in 90% of cases, it is not needed in wireframes. In fact, adding the ability to rotate any control will likely result in wireframes that are very hard if not impossible to implement by the development team.

We do support rotation in a few controls, where it makes sense. Some controls can be rotated in 90 degrees steps (icons, the triangle geometric shape...) and the label control has [a few rotation options](#) in its property inspector.

If you feel that you need to rotate any other control, let us know on [this forum thread](#) and we'll discuss whether to add it together.

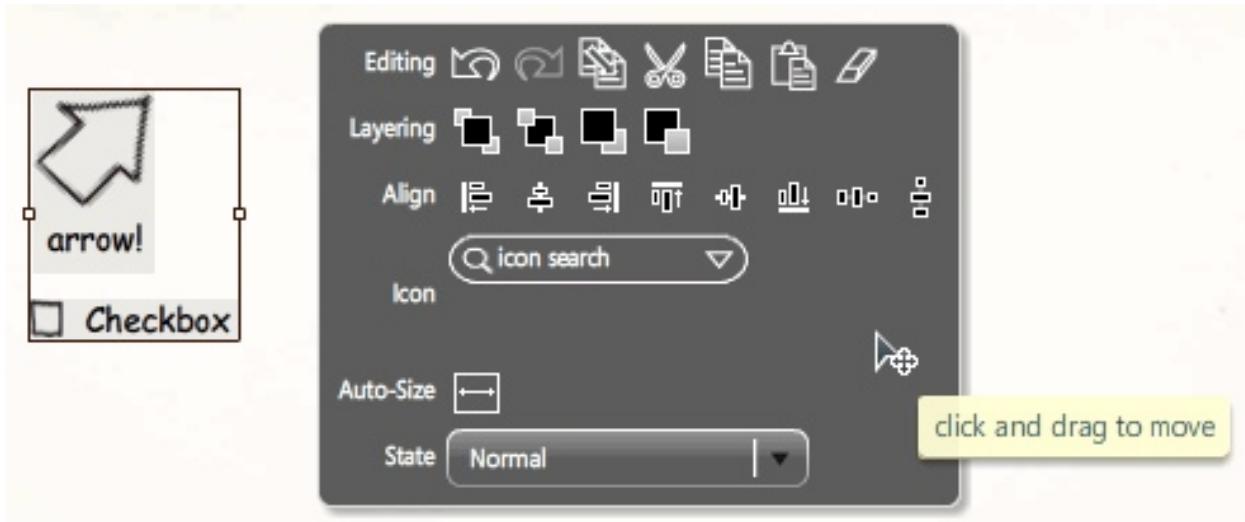
Also remember that you can always use the [image import feature](#) as a workaround.

Available Controls

You can [view a list and screenshots of all of the components](#) that ships with Mockups.

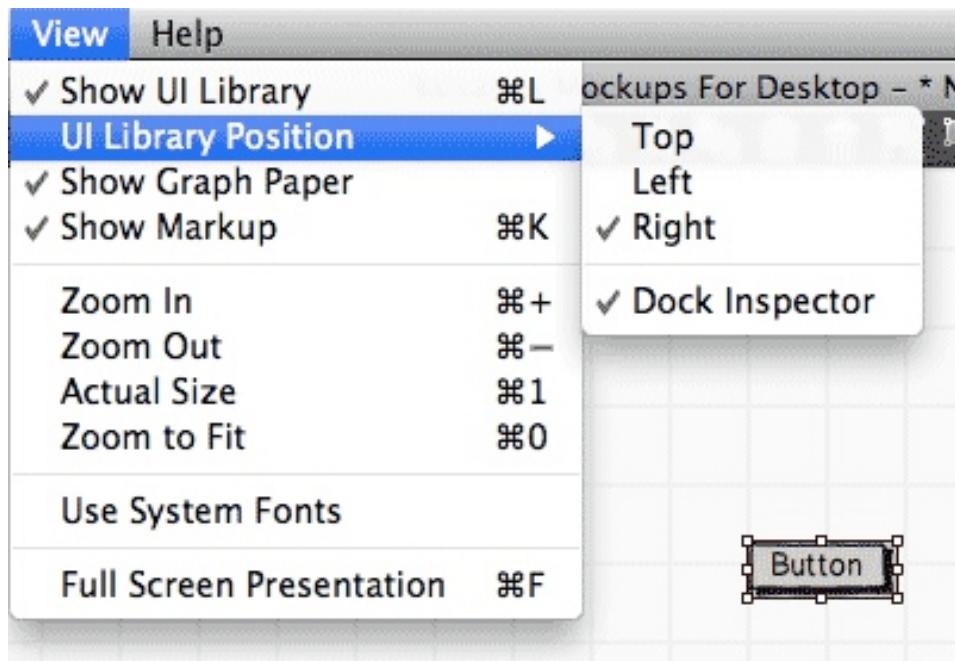
The Property Inspector

The property inspector is the little floating rectangle that appears once you select one or more UI controls. It allows you to perform some common functions such as copying/pasting, layering, and aligning controls, as well as some control-specific customizations.



The Property Inspector is displayed when you select an object on the canvas. You can drag and drop the Property Inspector to move it, it will re-appear where you left it next time you need it.

If your UI Library is visible, you can set the inspector to be docked over it. The menu View > UI Library Position shows this option.



Common Inspectors

Editing Inspector



Always present, always at the top. From left to right: undo, redo, duplicate, cut, copy, paste, delete, group, ungroup, lock

Layering and Position & Size



Always present, always in the second row (below the editing inspector).

Layering, from left to right: bring to front, send to back, bring forward, send backward.

Position and size shows you the size and position of the selection, in pixels. Clicking on the numbers lets you edit them. The format is "X,Y WxH" where X is the x position, Y is the y

position (0,0 is top-left, growing right and down), W is the width and H is the height of the selection.

Align



Shown whenever more than one control is selected. Lets you align the selected elements horizontally to the left, center and right and vertically to the top, middle and bottom. The last two icons on the right distribute the centers of the selected items horizontally or vertically.

Other inspectors

Arrow



Lets you decide the direction and curvature of the arrow, as well as whether to show or hide arrow pointers.

Auto-Size



Clicking on the autosize button will resize the component to its natural size, with everything showing and no extra space

Border



Shows or hides the selected controls; border

Color



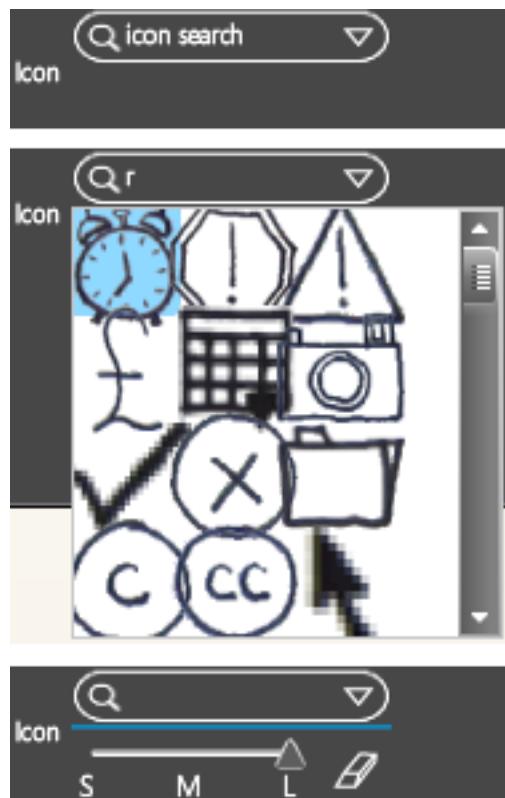
Lets you select the foreground or background color of the selected controls (depending on the control type), as well as the background color's opacity (only in some controls).

Horizontal Curly Brace Direction



Lets you specify whether the horizontal curly brace should point up or down.

Icon



The icon inspector lets you select, resize and remove an icon to the selected controls. You simply click on the search field, start typing the name of the icon you need and you'll be presented with suggestions that match your string. Use the arrow keys or your mouse to select the icon you want, then use the slider to resize it.

To remove an icon, click on the eraser icon next to the slider.

To see the full list of icons available, click the little triangle which will open the Icon Library.

Icon Label Position



Lets you specify whether the label should be positioned below or to the right of the icon

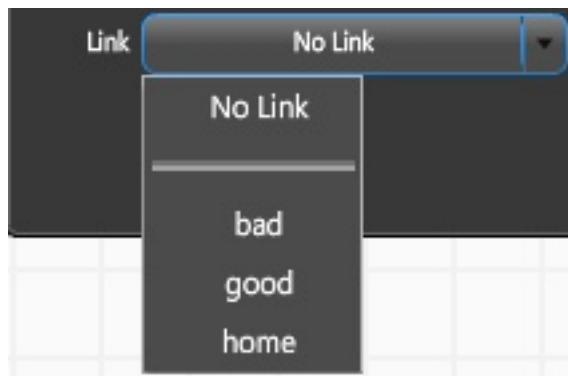
Image

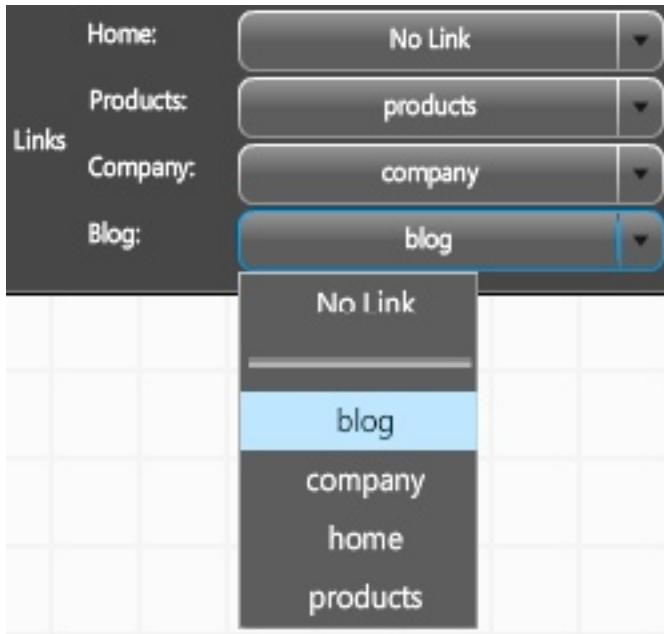


The Image inspector lets you select which image to use for the selected image controls. You can select the default placeholder, add a new image or select from a list of images already present in the mockup.

You can additionally "sketch" the image to make it look a bit more hand-drawn by selecting the "Sketch it!" checkbox.

Links





Lets you link a mockup to another. See [Linking](#) for more info.

List and DataGrid



Lets you specify whether the selected controls should have a header row, two colors for alternating row backgrounds, the height of the row and which row should be selected, if any.

In the DataGrid control, you can additionally toggle the visibility of vertical and horizontal dividing your table's columns and rows.

Progress Bar Type



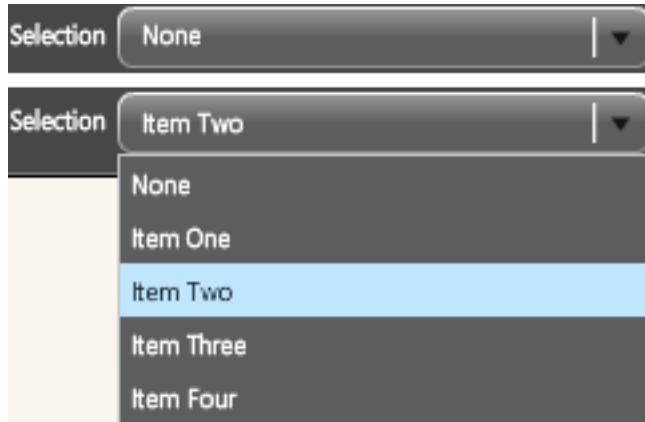
Lets you toggle between a progressive and an indeterminate (candy-cane) progress bar.

Scrollbar



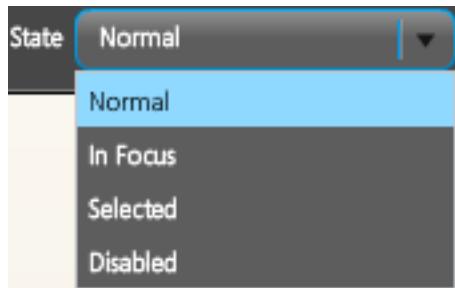
Lets you quickly add a scrollbar to the selected controls.

Selection



Lets you specify which elements of the selected control should be selected, if any.

State



Lets you specify the state of your control, whether it's in focus, selected, disabled etc. Different control types support different states.

Tabs Position



Lets you specify whether the tabs should be above or below the canvas, as well as their horizontal alignment relative to the canvas.

Text Styles



Lets you specify whether the font of the text in the selected controls should be bold, italic or underlined, as well as its horizontal alignment and point size.

Vertical Curly Brace Direction



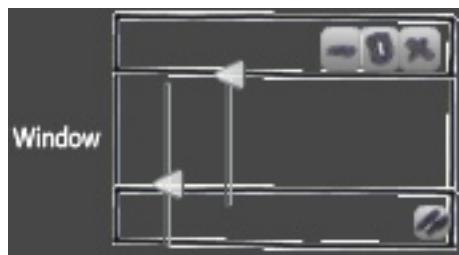
Lets you specify whether the vertical curly brace should point left or right

Vertical Tabs Position



Lets you specify whether the tabs should be to the left or right of the canvas.

Window / Dialog



Lets you toggle window chrome controls like minimize, maximize and close, as well as a dragging affordance. You can also specify the height of the top and bottom bars.

Need More info about Property Inspectors?

Want to know which property inspectors each control type supports or vice-versa? Refer to our handy [Control and Properly Inspector Matrix!](#)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
Control \ Inspector	Alternate Rows	Autosize	Background	Border	Border Style	Color	Direction	Header Row	Horizontal Lines	Icon	Image	Link	Opacity	Oriental Position	Right Left Arrow	Row Height	Scrollbar Selection	Shape	State	Text Format	Type Standard	Value Indefinite	Vertical Lines	Window			
2 Accordion	X											X			X												
3 Alert	X								X	X															X		
4 Arrow Line												X															
5 Arrowcomb																											
6 Browser																											
7 Button	X					X				X		X										X	X				
8 Button Bar	X											X															
9 Calendar																											
10 Card																											
11 Chart Bar																											
12 Chart Column																											
13 Chart Line																											
14 Chart Pie																											
15 Checkmark																											
16 Checkbox Group	X											X			X										X		
17 Color Picker																											
18 Combobox	X											X			X							X	X				
19 Comment																											
20 Cover Image																											
21 Data Grid	X	X	X						X	X	X							X	X	X							
22 Date Picker	X																										
23 Fieldset/Group												X															
24 Formatting Toolbar																											
25 Geometric Shape												X			X												
26 Group																											
27 Horizontal Curly Brace									X	X																	
28 Horizontal Rule																											
29 Horizontal Scrollbar																											
30 Horizontal Slider																									X		
31 Horizontal Splitter																									X		
32 Icon																											
33 Icon and Text Label	X		X	X								X		X													
34 Image																											
35 Label	X											X		X										X			
36 Link																											
37 List Bar																											
38 List	X		X	X																							
39 Menu																											
40 Menu Bar	X		X									X		X													
41 Model Sheet																											
42 Numeric Buttons	X																										
43 Numeric Stepper	X																										
44 On/Off Switch																											
45 Paragraph																											
46 Playback Controls																											
47 Progress Bar																											
48 Radio Button	X																								X		
49 Radio Button Group	X																								X		
50 Rectangle			X	X	X							X		X													
51 Red X																											
52 Screen Out																											
53 Search Box	X																										
54 Street Map																											
55 Subtitle																											
56 Tab Bar	X			X								X		X				X	X								
57 Tag Cloud																											
58 Text Area																											
59 Text Input	X			X								X		X								X	X				
60 Title Headline	X																										
61 Tooltip																											
62 Tree Pane		X			X							X		X													
63 Vertical Curly Brace					X	X																					
64 Vertical Rule																											
65 Vertical Scrollbar																											
66 Vertical Slider																											
67 Vertical Splitter																											
68 Vertical Table	X		X									X		X			X	X	X								
69 Volume Control																											
70 Volume Slider																											
71 Webcam																											
72 Window																									X		

Working with Text

Some UI controls, such as Button, Label or DataGridView, have text in them. To edit the text, double-click on the control and start typing. Or if you have the control selected, simply hit ENTER or F2 to start editing. To commit the text you typed, simply click anywhere other than the text field you typed in or hit Enter or CTRL+Enter. If you want to discard the text changes you just made, hit the ESCape key.

Basic formatting

Text style

You can use the following notation to format only certain parts of your control's text.

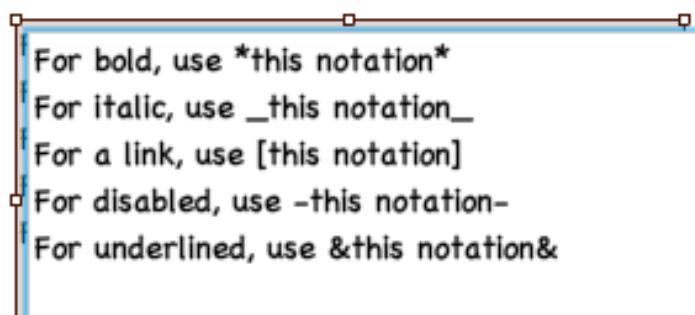
- for italic, use _this notation_

- for a link, use [this notation]
- for bold, use *this notation*
- for disabled, use -this notation-
- for underlined, use &this notation&
- for color, use {color:#FF0000}this notation{color}. The #FF0000 is the color in HEX form, just like HTML. The macro will work with or without the # sign.

All of the above syntax works around words separated by white spaces, and underline will work within words.

Here's a screenshot of what the above text looks like in a Paragraph control, for instance:

For bold, use this notation
 For italic, use *this notation*
 For a link, use [this notation](#)
 For disabled, use this notation
 For underlined, use this notation



The shortcuts will work almost everywhere, where it makes sense, and you can combine them, so to make an italic link use [_this_] or _[this]_.

Most of the text in the controls of Mockups is bold by default, so bolding some text within it won't make a difference. Some controls like Paragraph or Label allow you to "unbold" the text via the properties panel. If you do so, the words you *bolded* will stay bold. I know this is not consistent everywhere, but I think simplicity (i.e. less options) trumps consistency in some cases.

You can escape *, _, -, [and] characters with *, _, \-, \[and \], so if you want to write "this [is] some text" and don't want the "is" to become a link, just type "this \[is\] some text".

Bulleted Lists

In a paragraph control, you can create a bulleted list by preceding text with either a hyphen or an asterisk.

- * item 1
- * item 2
- item 3
- item 4

More Macros

lorem

Type `lorem` in a Paragraph or Text Area to use our Lorem-Ipsum generator. Try `10rem` for a variation showing number of characters.

{mockup-path}

Type `{mockup-path}` (all lower-case) in a Label or Paragraph control to show the full path of the current mockup.

{mockup-name}

Type `{mockup-name}` (all lower-case) in a Label or Paragraph control to show the current mockup name.

Working with Data Grids / Tables

Column Width Options

Starting with version 1.8 Mockups lets you specify individual width and alignment options for each data grid column.

Here's a quick video intro:

► Watch on YouTube: <http://youtu.be/yUUdt6vfc4s>

Here's the details for creating column widths:

- you have to add a special line of text as the last line of text in your table, and it has to be wrapped by {} curly brackets
- for each column, use 0 if you want the column to be as small as possible to fit the text in that column, or a number
- numbers specify the relative size of columns, so {2,1} means "make the first column twice as big as the second one in this two-column grid. Or {70,20,10} means "in this 3-column grid, make the 1st column 70%, the 2nd 20% and the 3rd 10% of the width of the whole table."
- you can combine numbers and zeros, as seen in the next example
- if you want to align column individually, add either L, C or R right after a number, like so: {0R, 2L, 1}, which means "in this 3 column grid, make the 1st column as small as possible and always align it to the right, the second twice as big as the third and always align it left, and use the datagrid's alignment (from the property inspector) to decide how to align the 3rd column".

Using Icons in Datagrid

Datagrids support a single checkbox or radio button in a table cell using the following text:

- Checkbox: [] or []
- Selected checkbox: [x] or [v] or [o] or [*] or [X] or [V] or [O]
- Radio Button: () or ()
- Selected Radio Button: (x) or (v) or (o) or (*) or (X) or (V) or (O)

Creating Line Returns in Datagrid

Sometimes you want to create line returns in a table cell. To do this, you can use "\r" to create a line return.

The code example below would print on two lines.

```
Name\r(job title)
```

Pasting Data from Excel

Some people like to prepare tables in a spreadsheet application like Excel. You can copy and paste a range of cells directly from Excel into the Data Grid/Table component. Just copy/paste and it will just work. It also works in the opposite direction from Mockups to Excel.

Note: If you have commas in your cells, you will have to escape them with a backslash like this:

```
1\,000\,000
```

The component recognizes both commas and tabs as delimiters. Any thing else should be populated into cells.

Working with Icons

Balsamiq Mockups comes packed with over 220 hand-drawn icons for you to use.

There are two ways to select an icon to use, the icon search and the icon library.

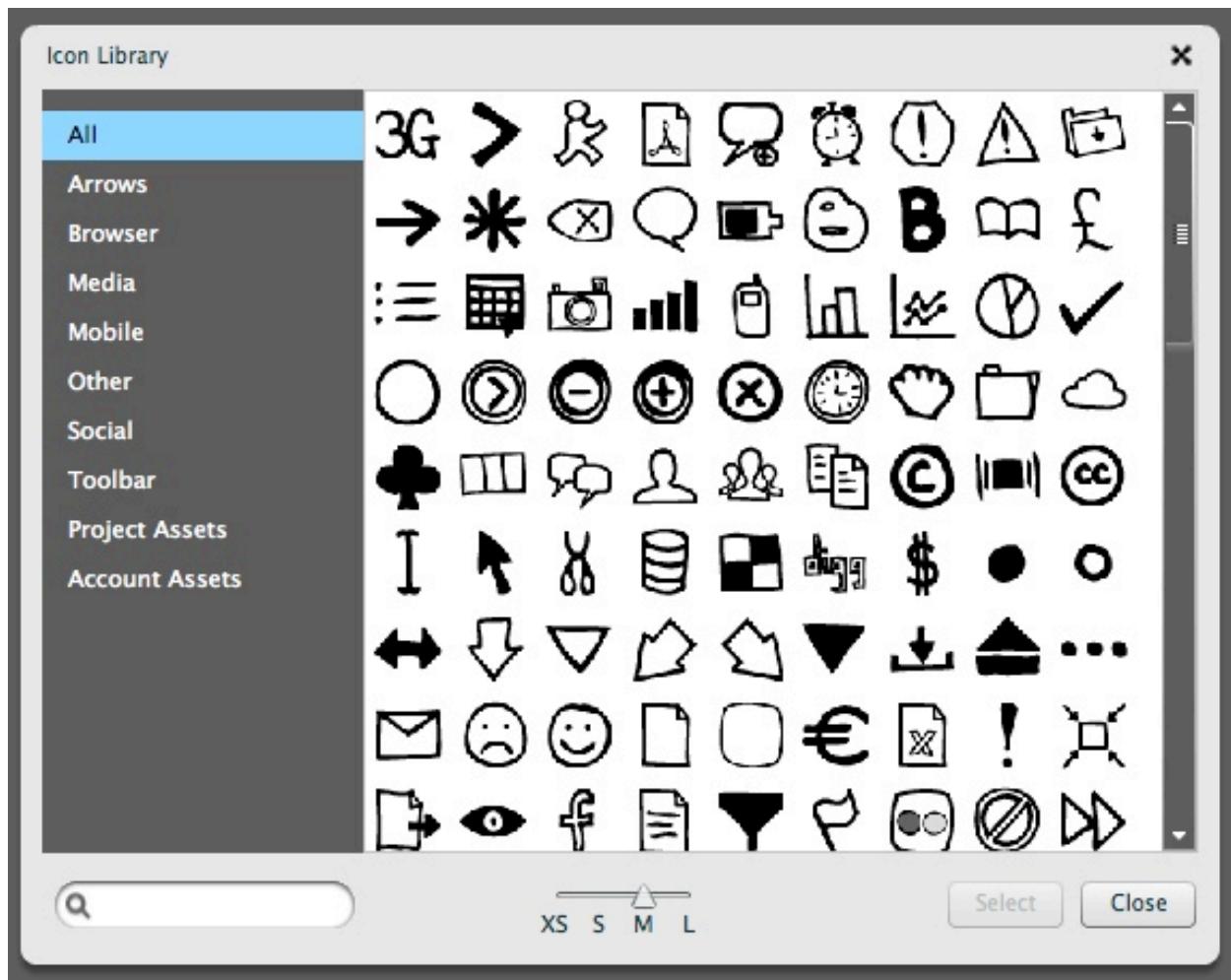
Icon Search

If the controls you selected support icons, you will see the icon search box in the property inspector. This works much like [the Quick Add tool](#): just type a few letters from the name of the icon you are looking for and a list of suggestions will pop up. In this case, the list is a grid of icons to choose from. Just click on the one you want to select it.



The Icon Library

If you'd like to explore all the icons that are available, click on the little down-pointing arrow to the right of the icon-search box. This will bring out the icon library. The icon library is pretty self-explanatory, let me know if you'd like more detailed info about it. One little trick is to double-click on the icon you'd like to select, instead of clicking and clicking on the "Select" button.



Once you have selected an icon, you can resize it, rotate it or remove it via the Property Inspector.

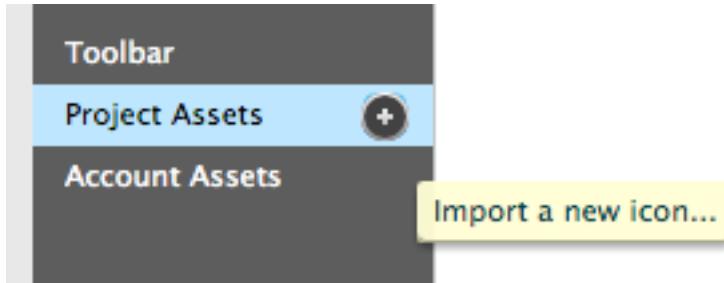
Adding your own custom icons

Starting with Mockups for Desktop version 2.1.7, you can now add your own icons to use in your wireframes.

If you want to share icons between your projects, [set up your Account Assets folder](#) first.

Adding via Icons dialog

There are two ways to add custom icons to Balsamiq Mockups. The easiest way is via the Icon Library. If you're working on a saved file, you'll see a Project Assets folder as below.



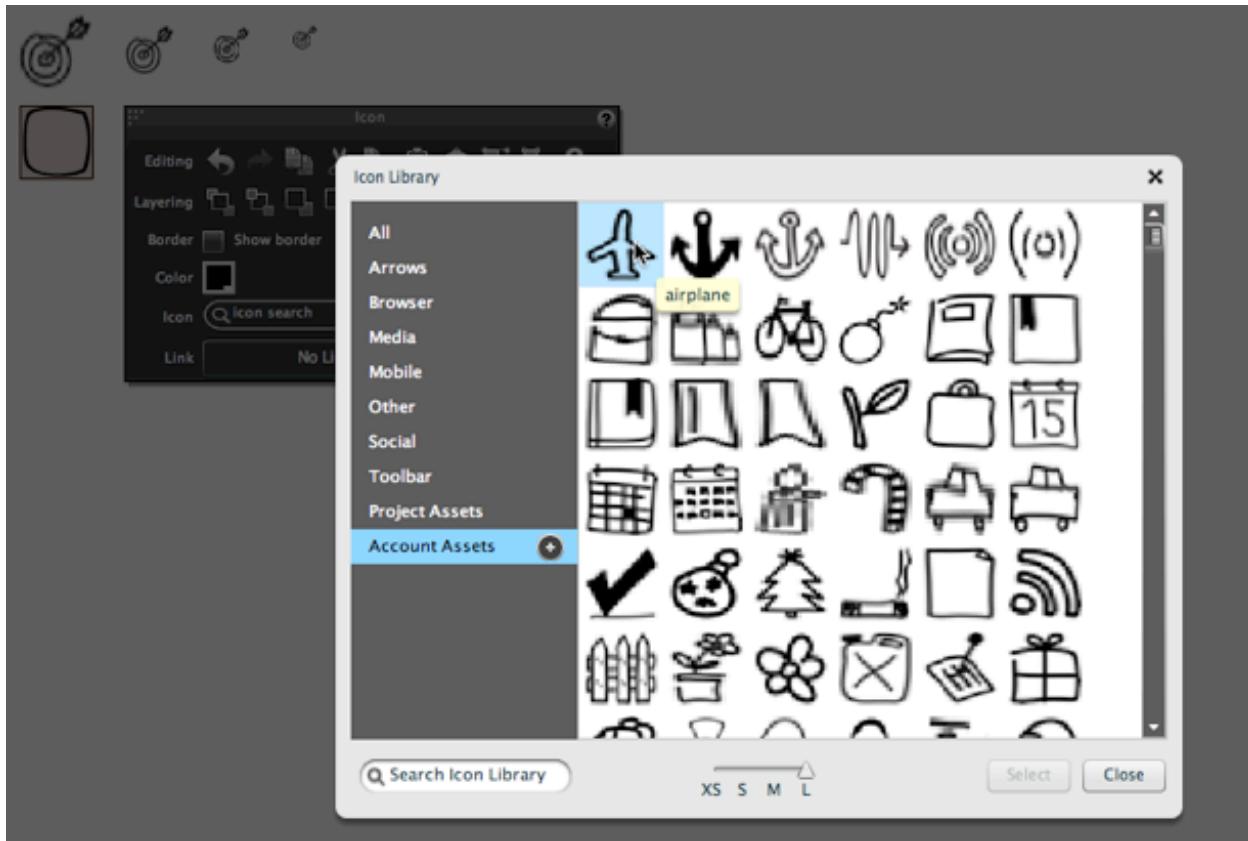
Just select Project Assets or Account Assets, then click the little plus button, select an image file and voila', your image will be resized down to fit in a 48x48 pixels square and copied to either the project assets or account assets folder for you.

Adding to assets folder on file system

The second way is to place icon files manually inside your [project assets](#) or [account assets](#) folders. Just add the "icon_" prefix to each file name and they'll show up in the icon library.

The words you use in your filename after icon_ are used as the searchable keywords in the icon dialog. So the example above shows airplane because the filename is icon_airplane.png. You can string together a few labels to get synonyms in there, e.g. if the file is named icon_bag briefcase portfolio.png, all of those words after icon_ will be searchable.

Here's an example showing how we added custom icons to a Mac by downloading the free [32x32px Doodlekit icons](#), ran them through the free [NameChanger app](#) to prepend files with icon_, and moved them to the ~/Documents/Balsamiq Mockups/assets folder. They now appear in the icon dialog like this:



The [Steedicons](#) also look like a perfect complement to the rough, thick style of Mockups. This [blog entry at tipsblogger](#) lists some more hand drawn icon sets that might be useful for this feature.

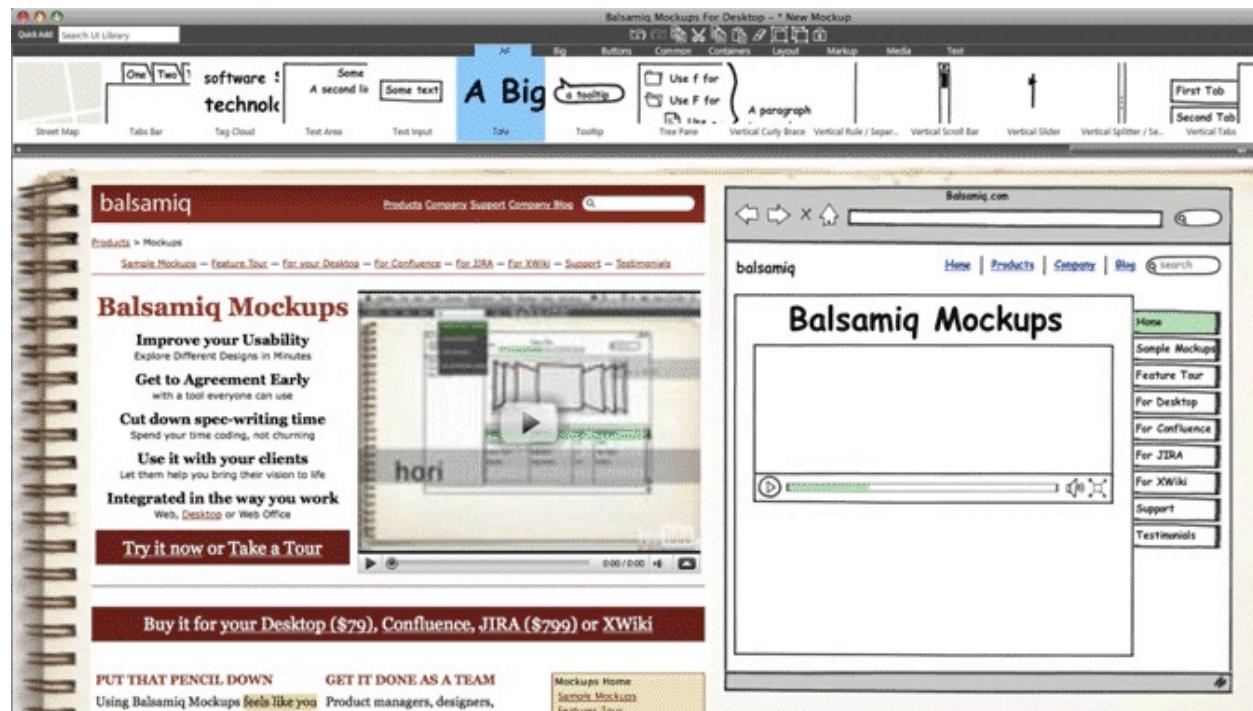
Working with Images

We have a video that gives a good introduction of the Image Import feature and what it enables.

► Watch on YouTube: <http://youtu.be/kniGOJXtKJ4>

The feature is simple to use and yet very powerful, enabling a number of different use-cases.

To take one example, let's say you're improving an existing interface. Take a screenshot of the current UI and import it into Mockups to tweak or use as reference, like this:



If you want to re-use images in multiple Mockups, you can import images of custom icons or custom control types using [Symbols](#).

Aside from the Image control, you can also embed images in the Cover Flow component (as seen in the video linked at the top of this chapter).

Copying Images to Assets

When you add images to Mockups, what we suggest is that you also add them to "project assets" so they will stay organized with the project and may be re-used in other mockups.

Here's how:

- If your Mockup isn't saved, Save your Mockup.
- Add an image to your Mockup.

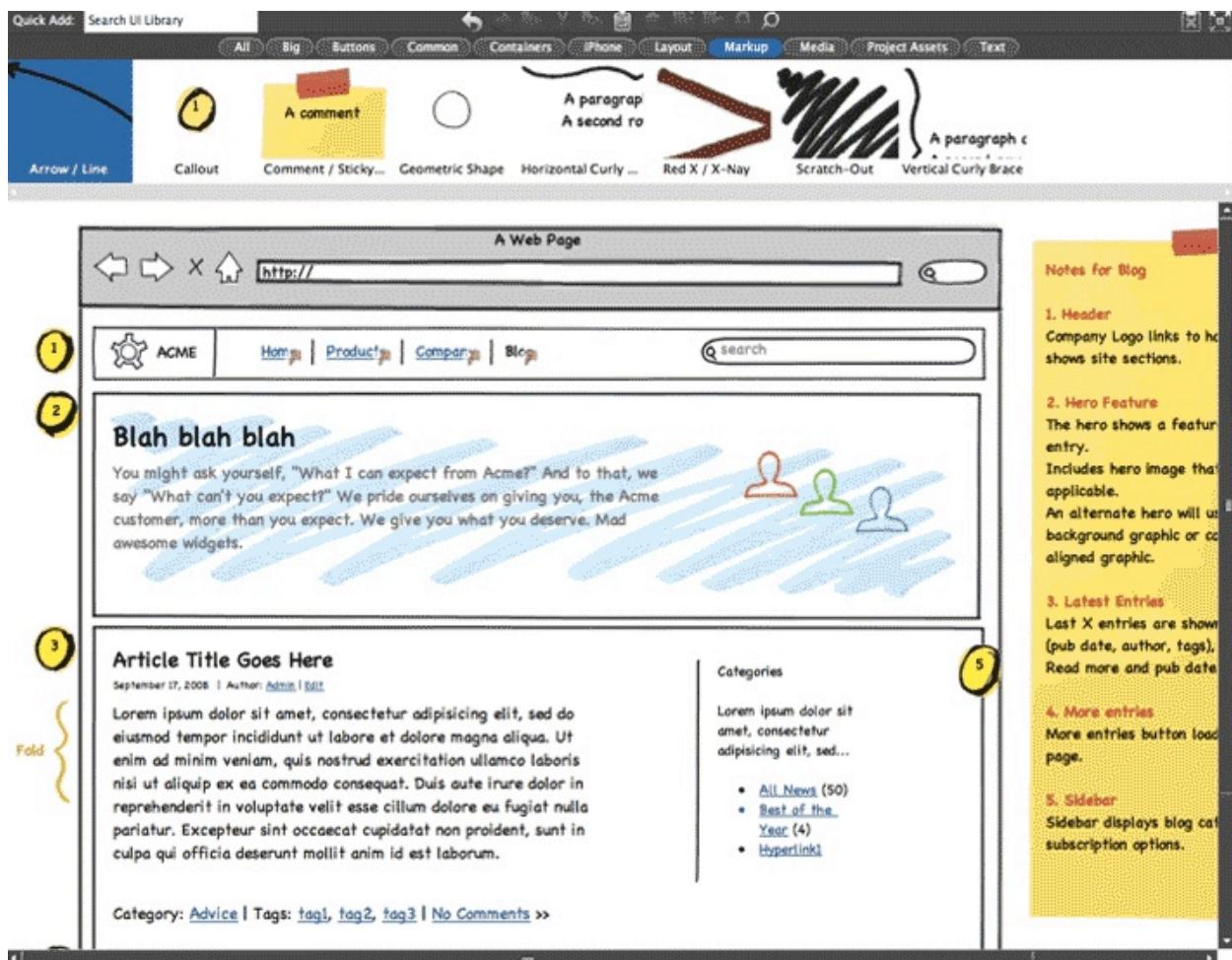
- In the lower left corner of the Property Inspector, check "Copy to Project Assets" so this image will stay organized with the project and can be re-used in other mockups.

After you add an image, it will be available in the property inspector.

Working with Markup

Markup is a special kind of component in the UI Library that is used for adding marks and explanatory notes.

Commonly used Markup items include: Arrow / Line, Callout, Geometric Shape, Sticky Note, Red X, and Curly Braces. The screenshot below shows the Markup button panel selected in the UI Library, and markup items on the page.

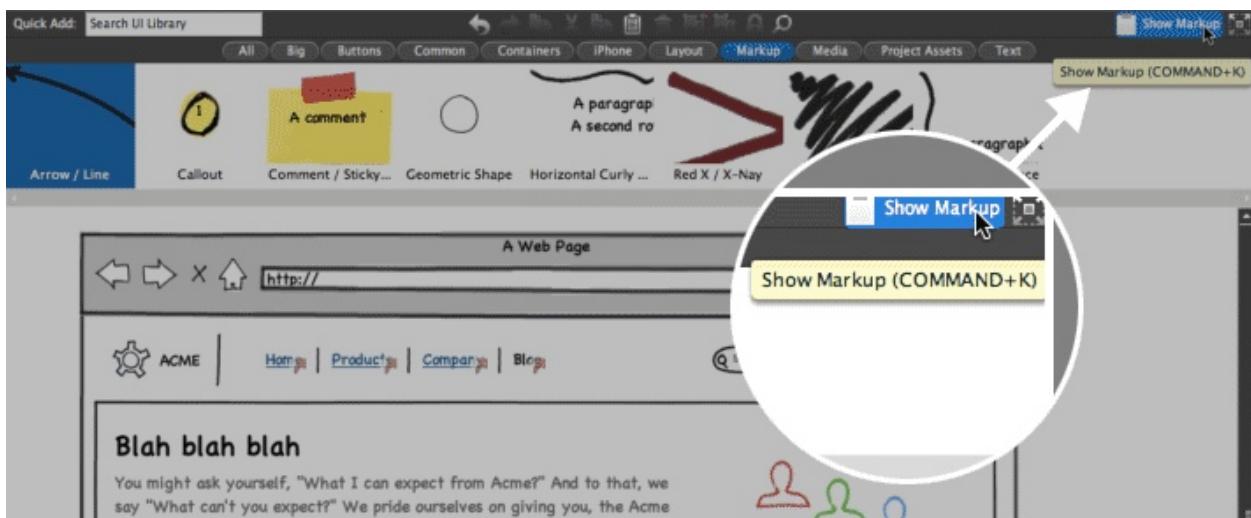


Toggling Markup On and Off

Markup does not typically appear when the Mockups are displayed in Full Screen Presentation mode or exported PDFs.

When you're editing your Mockups, you will sometimes want to see what the Mockup looks like with Markup off. You can do this by using the Show Markup toggle button in the upper right corner of the editor to the left of the Presentation Mode button.

The screenshot below shows our previous page with Markup off, and the toggle button is displayed in the corner.



Markup Toggling with Keyboard Shortcuts

We also have keyboard shortcuts for toggling visibility on and off.

Press Ctrl-K on Windows (CMD-K on Mac) to toggle markup on and off.

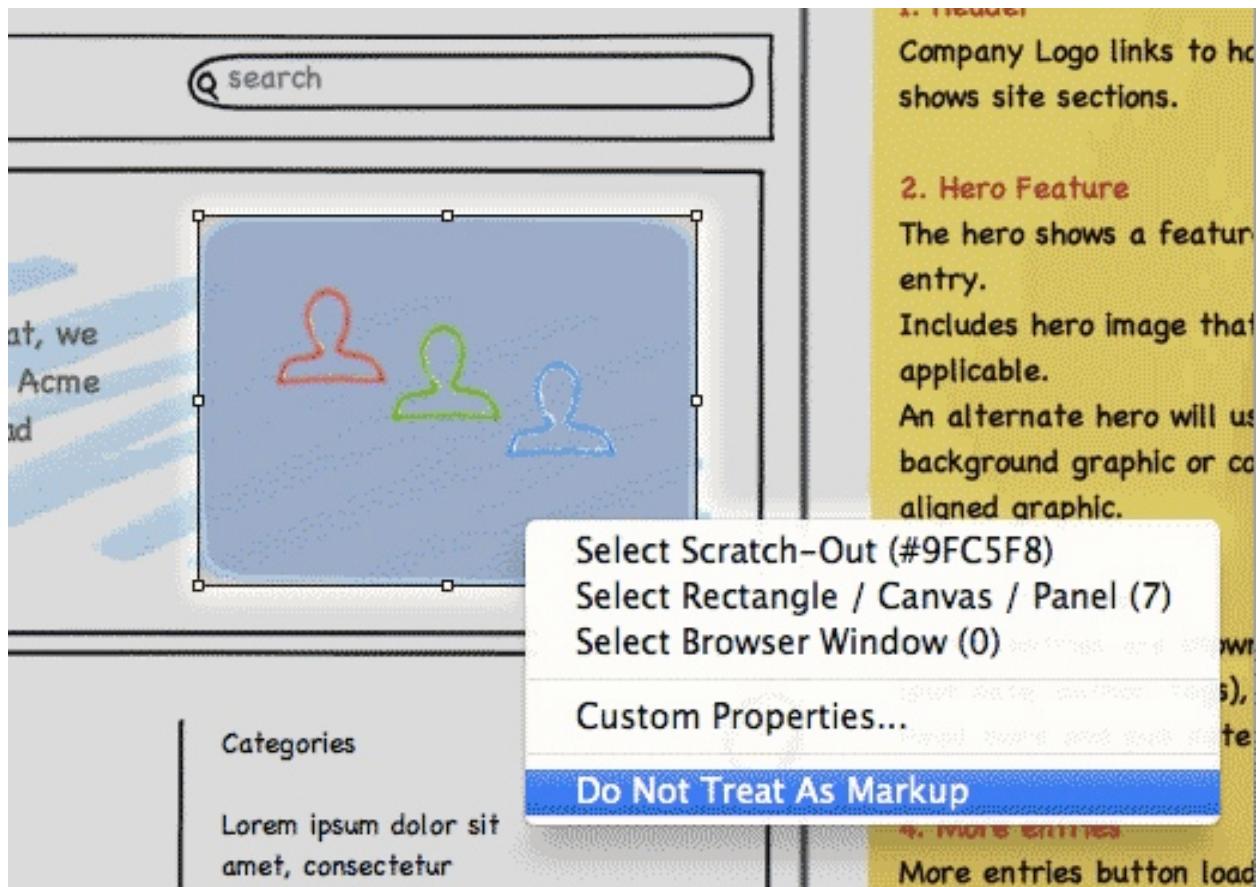
In Full Screen presentation mode you can just click the "K" key.

Making Items Markup and Non-Markup

Sometimes you will want Markup items to be treated as non-markup. For instance, it is popular to use the geometric component as an interface element because it offers more shape options than the rectangle component.

If you want to make a Markup item Non-Markup, right click on the component and you will see a "Treat as Non-Markup" option in the context menu. This will allow the component to be visible when the Markup visibility toggle is set to on or off.

You can use this "Treat As Markup / Treat As Non-Markup" feature on any component. If you right-click on a regular component, you can make it behave like Markup by selecting "Treat as Non-Markup." It will be hidden when you toggle Markup visibility off.



Exporting Markup to PNG and PDF

If you want your Markup components to appear or be hidden in your exported PNG images or your Exported PDF, simply toggle the Markup visibility as shown above, and Export.

Working with Symbols

Starting with version 2.0, Mockups for Desktop supports the concept of Symbols. Other software often refers to this same feature as templates, master pages, custom components, or widgets.

In essence, Symbols let you create reusable common elements across different mockups.

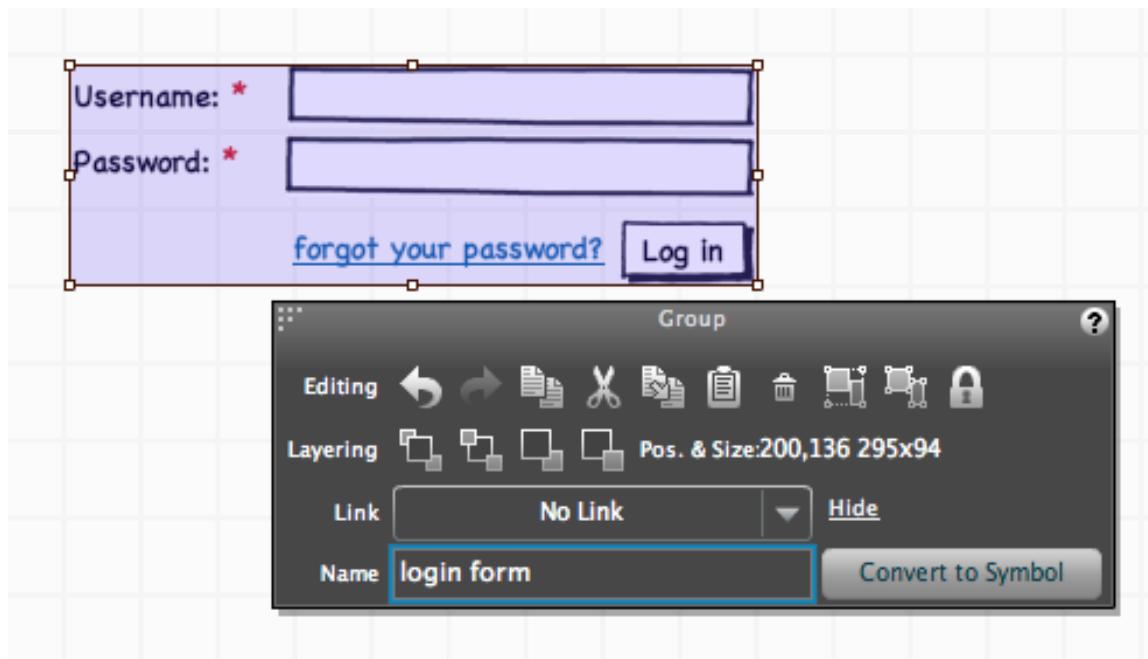
► Watch a video introduction to Symbols: http://youtu.be/ATh-V_RvIB8

Creating Symbols

Symbols in Mockups are simply named groups which have been "converted" to symbols (we'll talk more about what goes on under the hood later), but first, let's create a symbol!

You may create a new group to convert to a symbol or use a group you already have. The following steps assume you are creating a new group.

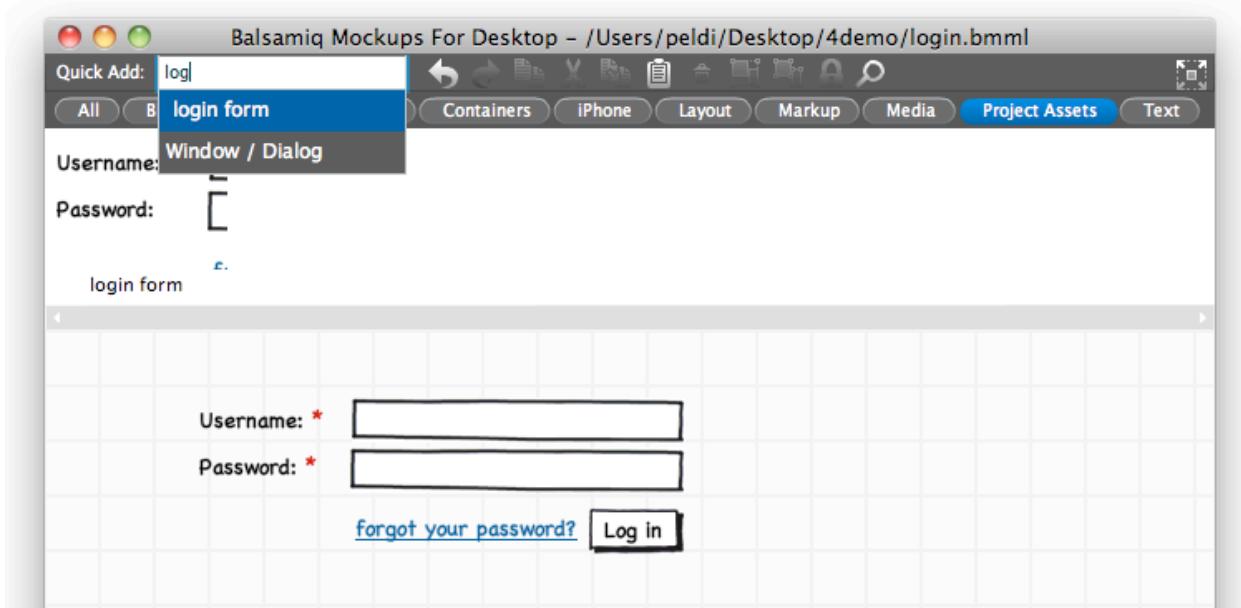
1. Select the controls you want to group
2. Group your selection (Select Edit->Group and group turns bluish color)
3. Name the Group (i.e. login form)



4. Save the File with the new group (File must be saved to convert a group to a symbol.)
5. Click on the 'Convert to Symbol' icon in the Property Inspector

You will notice that the group selection becomes a light green. This means that the selection is now an instance of a symbol.

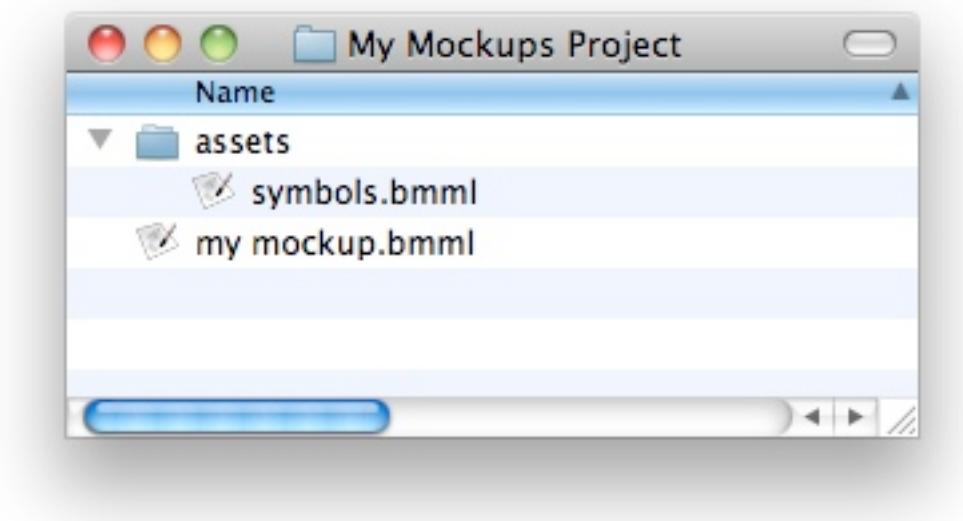
You now have a symbol named 'login form' which will be displayed in the Project Assets tab of the UI Library. (You can also add the symbol via Quick Add).



Under the Hood

When you hit "Convert to Symbol", Mockups for Desktop added the named group you selected to a file called symbols.bmm in a folder called "assets" right next to your bmm files. You might want to refer to our [best practices for creating projects with Mockups](#) if you haven't read it yet.

The key to understanding Symbols in Mockups is that Symbols are named groups contained inside BMML files that are stored in your projects' assets folder.



We know the definition above is pretty dense, but understanding it fully will help you make the most out of this awesomely powerful feature. Don't worry if it makes little sense to you right now, we'll come back to it. Just read it one more time and move on. :)

Overriding Symbol Properties

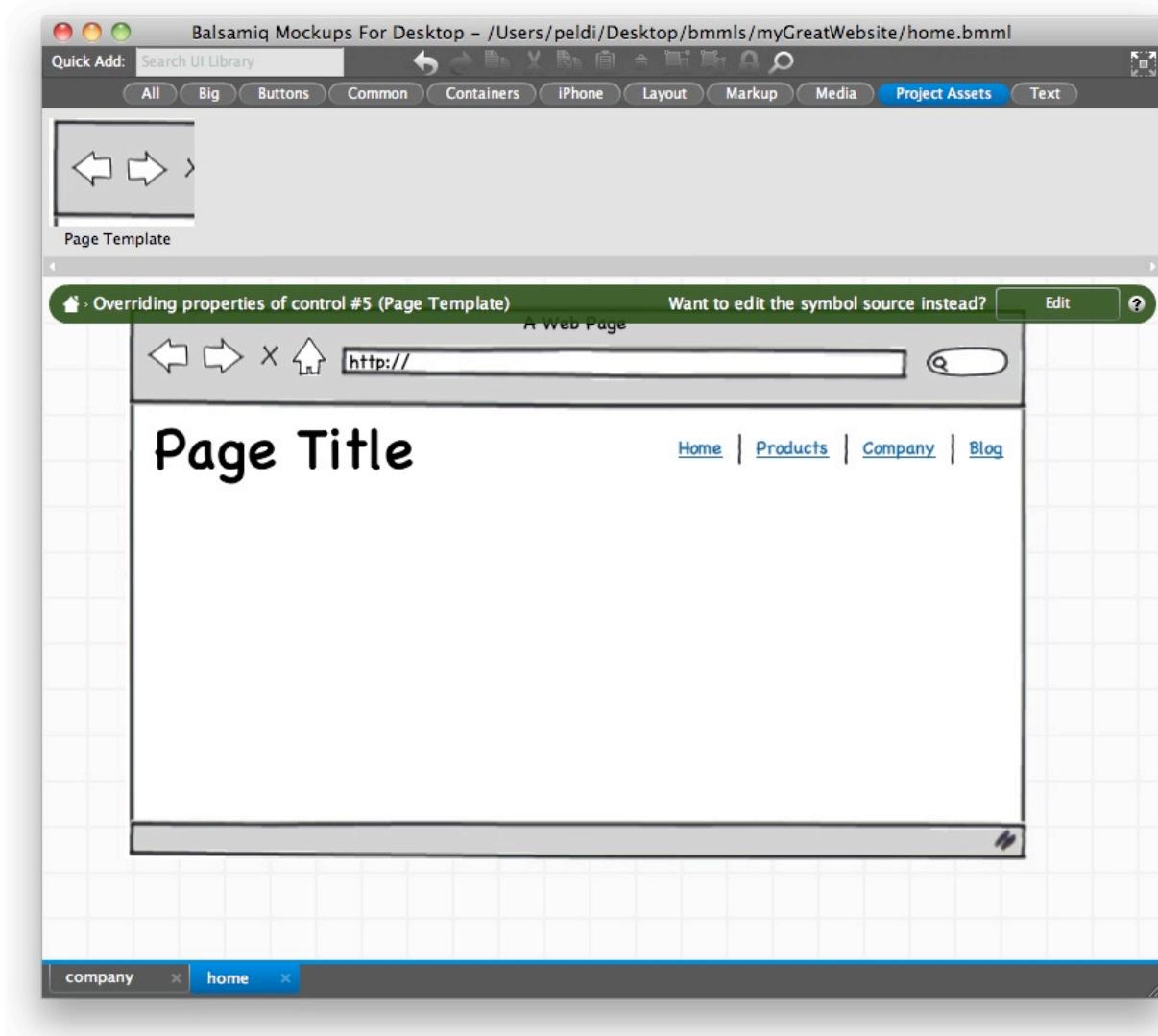
Once you have created a symbol, you will often want to change it a little each time you use it. Imagine for instance a Symbol you created as a master page / template, containing a web page's title and navigation.



All your website pages will have the same font size and position for the title, but the title's text should be different on each page. Same goes for which page should be shown as current in your navigation bar.

Symbols allow you to achieve this result by letting you override symbol properties each time you use a symbol. To do so, start by double-clicking (or hitting ENTER, or F2) on a symbol to "enter it".

Balsamiq Mockups Documentation



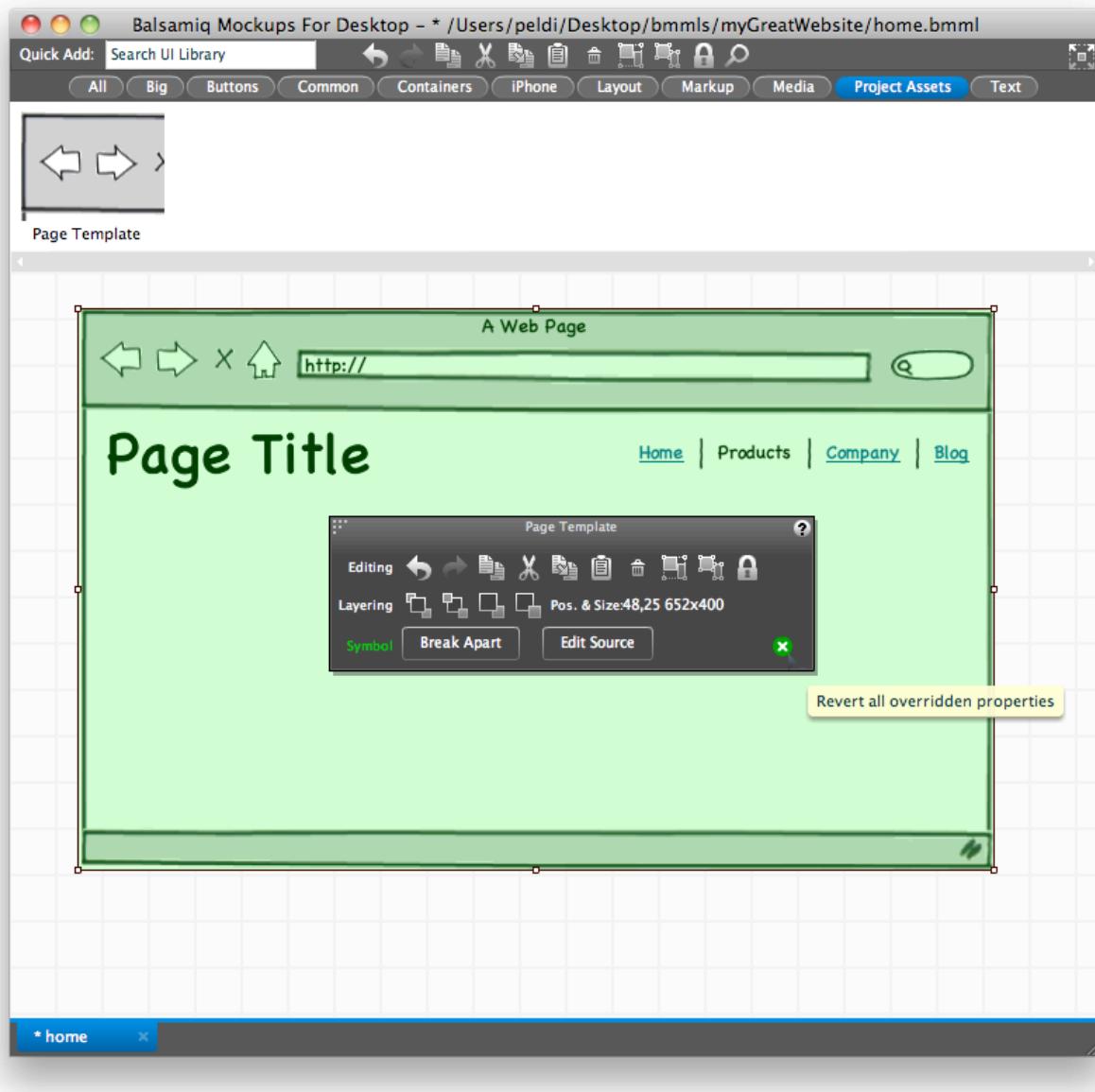
Although this experience is very similar to editing a group's contents, you will notice that Mockups for Desktop warns you that what you're doing is really overriding some properties of a Symbol.

At this point, you can manipulate each control inside the symbols at will, as if you were editing a group. Some operations are not permitted while overriding symbol properties, such as adding, deleting or grouping controls.

If you make a mistake, you can always undo to get back. If you want to remove a single property change and go back to a Symbol's default property, you can click on the little green "x" icon in the property inspector.



You can also revert all changes you made to a Symbol's instance at once, via the "x" icon in the property inspector you see when selecting the whole symbol.



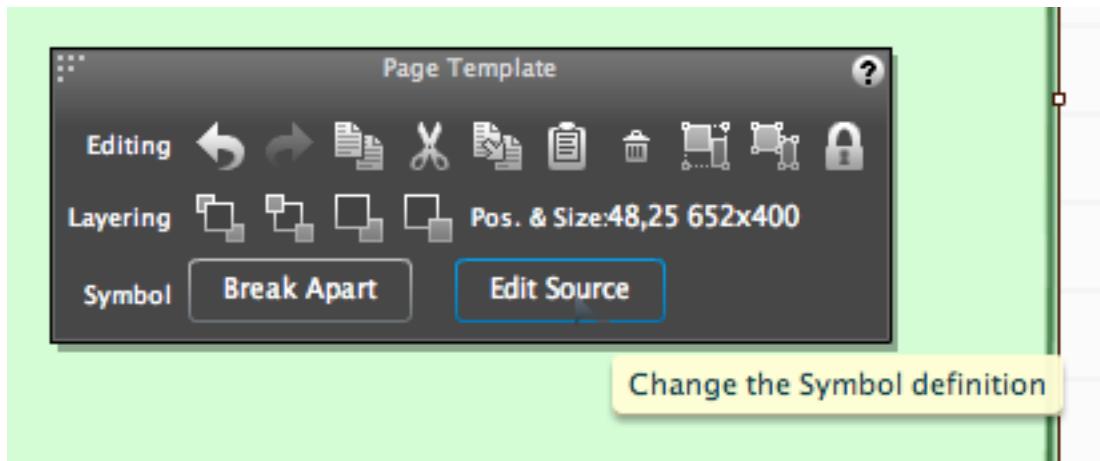
Editing a Symbol's Source

The main benefit of reusable Symbols is that if you need to make a change, you can just do it in one place and it will be propagated to all the instances (uses) of that Symbol.

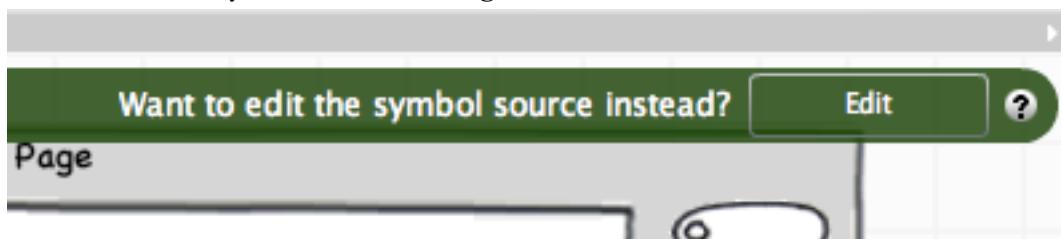
To edit a Symbol's source, you simply open the BMML file that contains the symbol and edit it normally, just as you would edit a group. Save your changes, go back to a mockups that uses that symbol and notice that your changes have been applied, just like that!

There are a couple of shortcuts in the app that help you reach a Symbol's source in order to edit it:

- You can select a Symbol instance and hit the "Edit Source" button in the property inspector.



- You can enter a Symbol for overriding, then hit the "Edit" button there.



2 Methods for Creating Symbols Libraries

There are 2 ways to create a symbol library if you're not using the Convert to Symbol feature. Below is a bulleted list showing the 2 methods. You can read on to the next section for illustrated step-by-step instructions.

Method I: Multiple symbols in a single symbols library file

1. Create a new mockup and save it to either your global assets/ (account assets) folder or to a project's assets/ folder. For instance, to create an icon library, create assets/icons.bmmml.
2. Add your component(s) onto the canvas, then select the ones you want to turn into a symbol and press Ctrl-G or use the menu Edit > Group.
3. With the group selected, the property inspector will show a name input. Enter the name for this symbol.
4. Repeat 2-3 to create more symbols.

Now your symbols will appear in Account Assets or Project Assets library tab whenever you're working on a Mockup. Mockups must be saved at least once for these tabs to appear in the library. Check the [illustrated step-by-step instructions](#) for more info.

Method II: One symbol per asset file

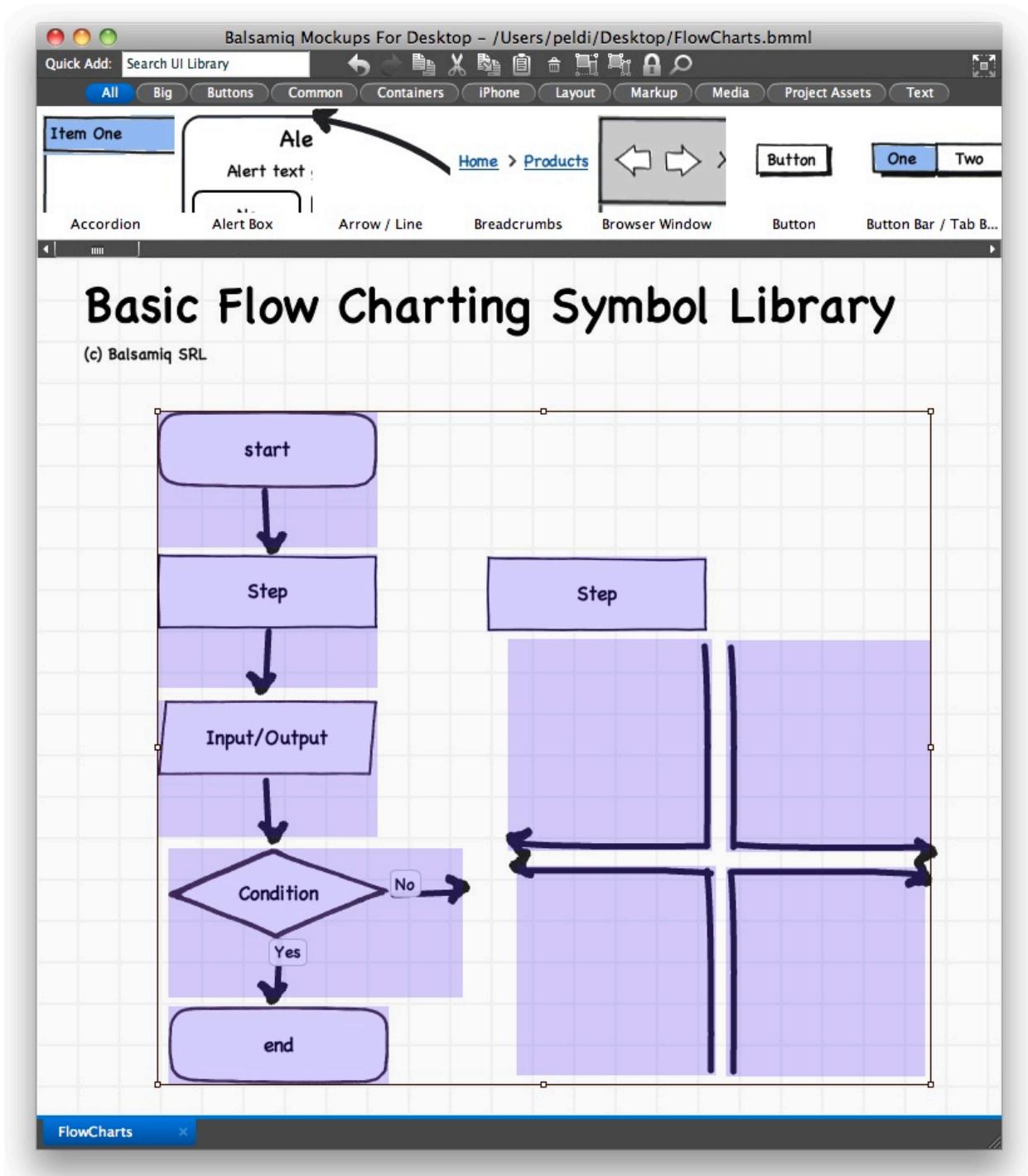
1. Create a new mockup and save it to either your global assets/ (account assets) folder or to a project's assets/ folder. For instance, to create an Arrow icon symbol, create assets/ arrow.bmml.
2. Add your component(s) onto the canvas and save.

Now your symbol will appear in Account Assets or Project Assets library tab whenever you're working on a Mockup. Mockups must be saved at least once for these tabs to appear in the library. Check the [illustrated step-by-step instructions](#) for more info.

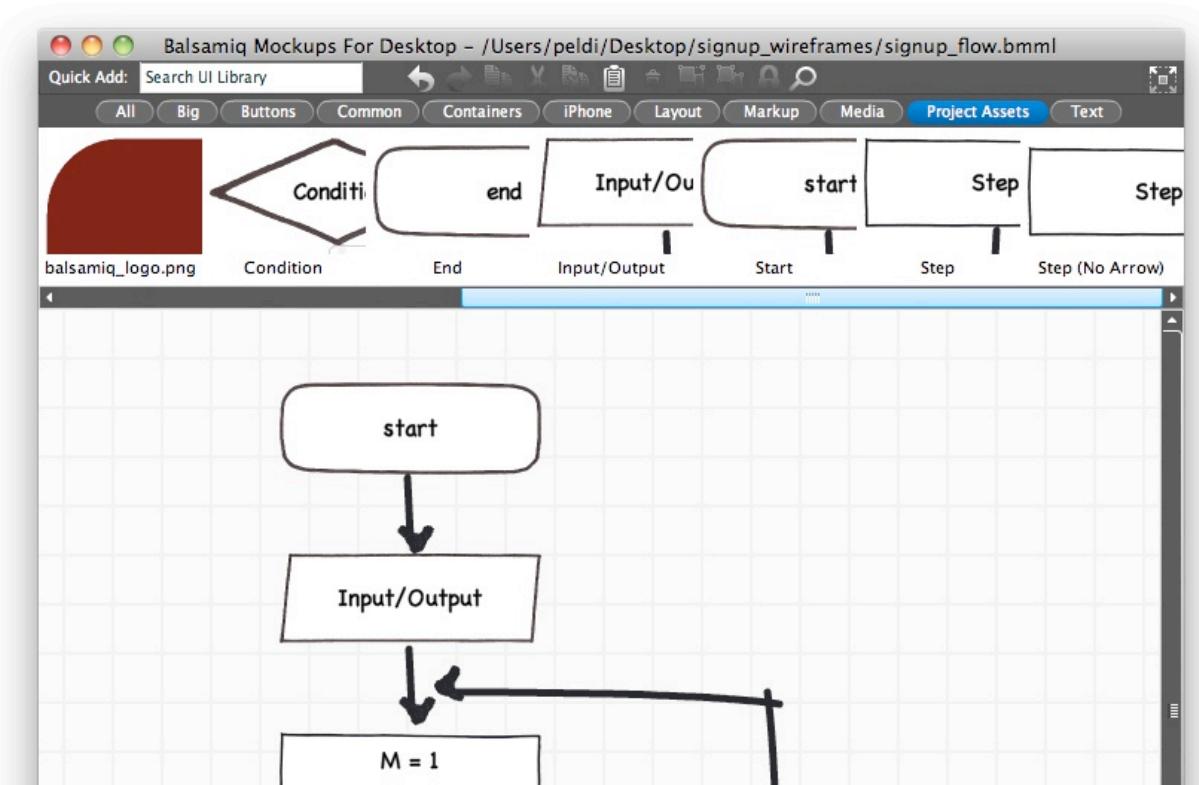
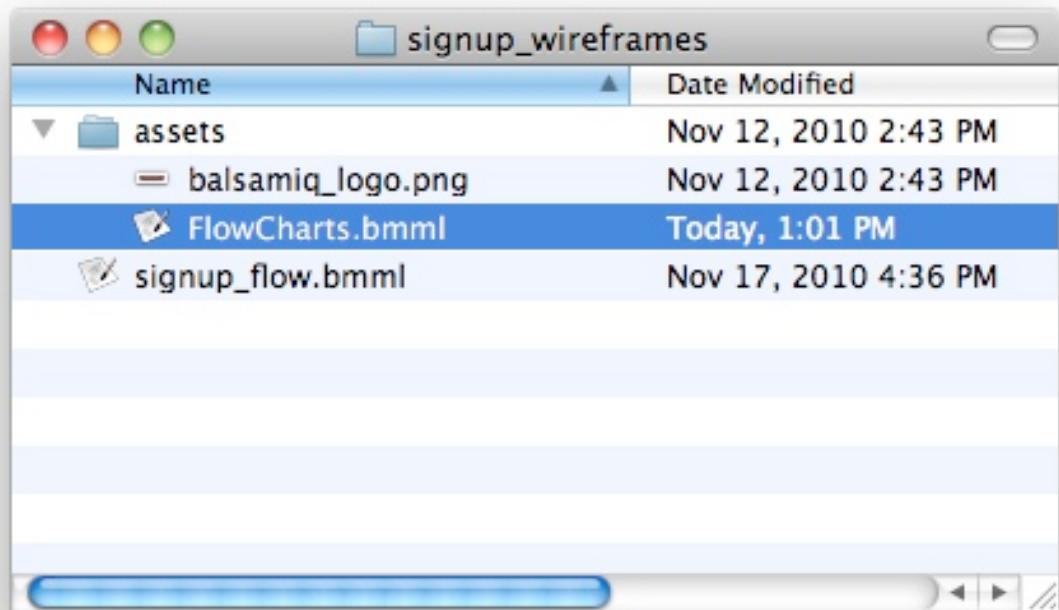
Symbol Libraries

A Symbol Library is a collection of symbol definitions. You might want to create a Symbol Library in order to share common controls with your team, or share them with the community via [Mockups To Go](#).

To create a Symbol Library, all you have to do is to create a bmml file, create some groups (one per symbol) and give each group a name.



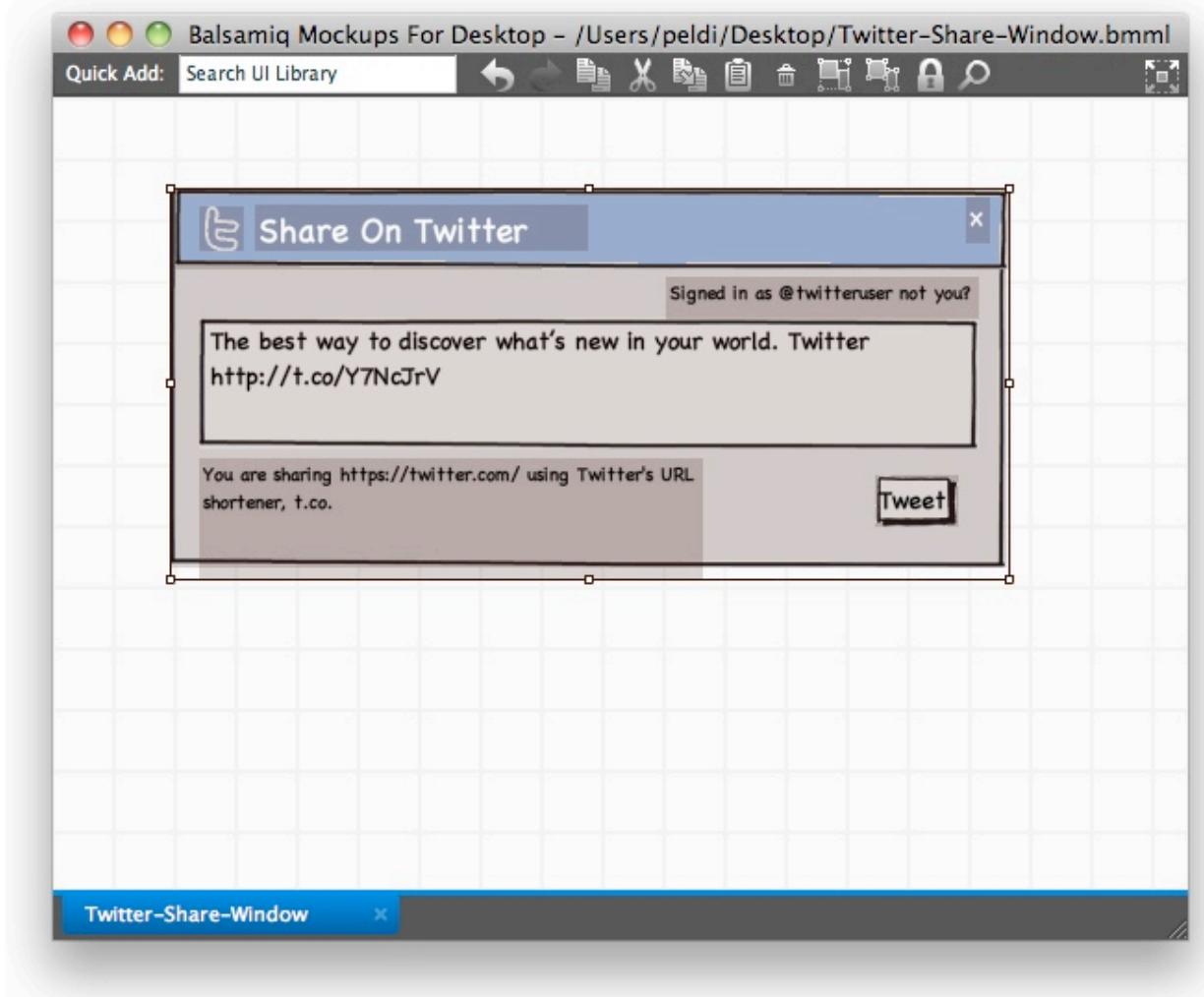
Whenever someone saves your BMML [in an asset folder](#), they'll be able to use the groups contained in it as Symbols in any mockup of that project.

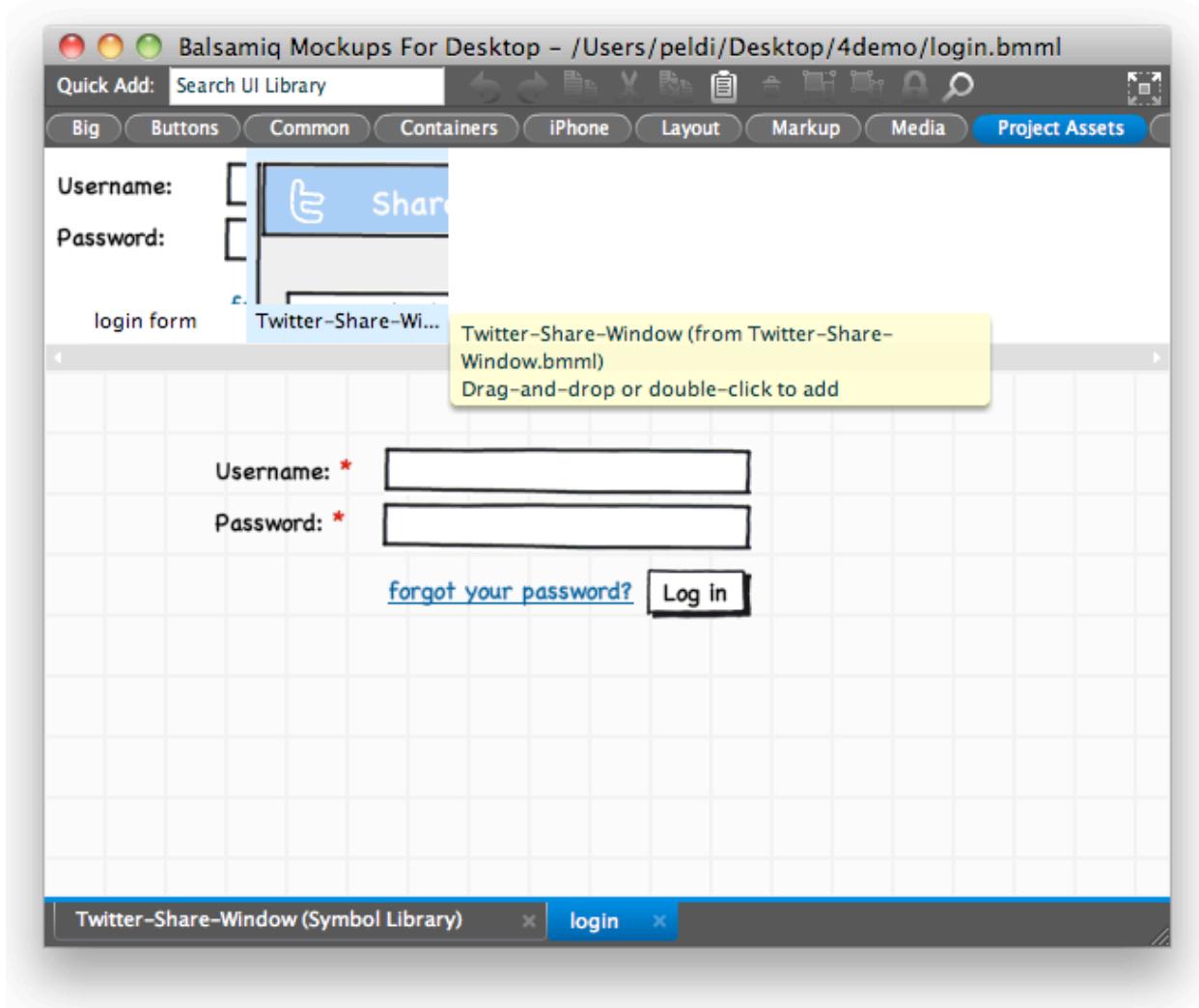


One more thing... Using Whole Mockups As Symbols

There's a little secret we haven't shared with you yet. Remember how *Symbols in Mockups are simply named groups contained inside BMML files that are saved in a project's asset folder?*

Well, it turns out that's not entirely true. There's one more way to create a Symbol in Mockups: just create a mockup with **no groups** in it and save it in an assets folder. Mockups will detect this, and treat **the whole mockup** as a symbol, named after the mockup's file name.





With this in mind, the full definition of symbols becomes as follows:

Symbols in Mockups are either named groups contained inside BMML files that are saved in a project's asset folder, or BMML files with no groups in them, saved in a project's asset folder.

Sharing Symbols Across Projects

What? You're still reading? You just can't get enough of Symbols can you? ;)

Allright, since you've asked for it, we'll spill all the beans for you. So far we've been talking about "project assets folders", the directory called "assets" sitting next to your BMML files.

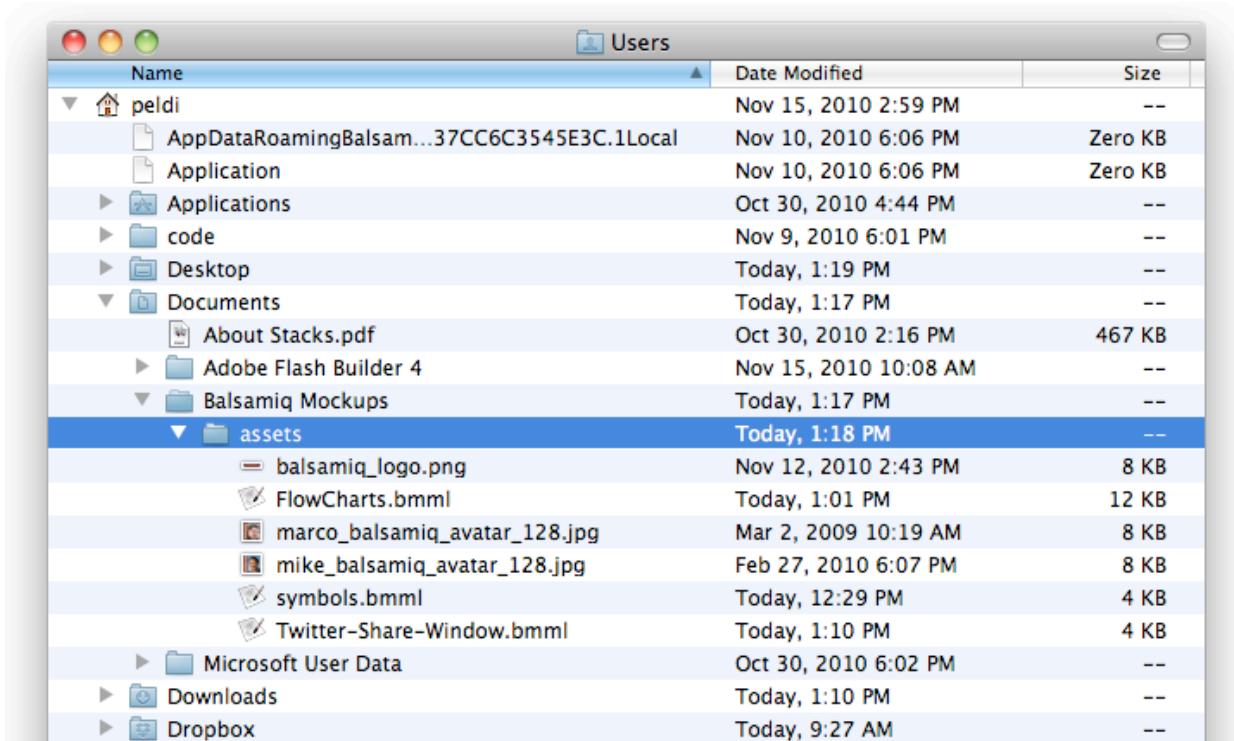
That's all nice and good, but what if you want to share Symbols across all of your projects?

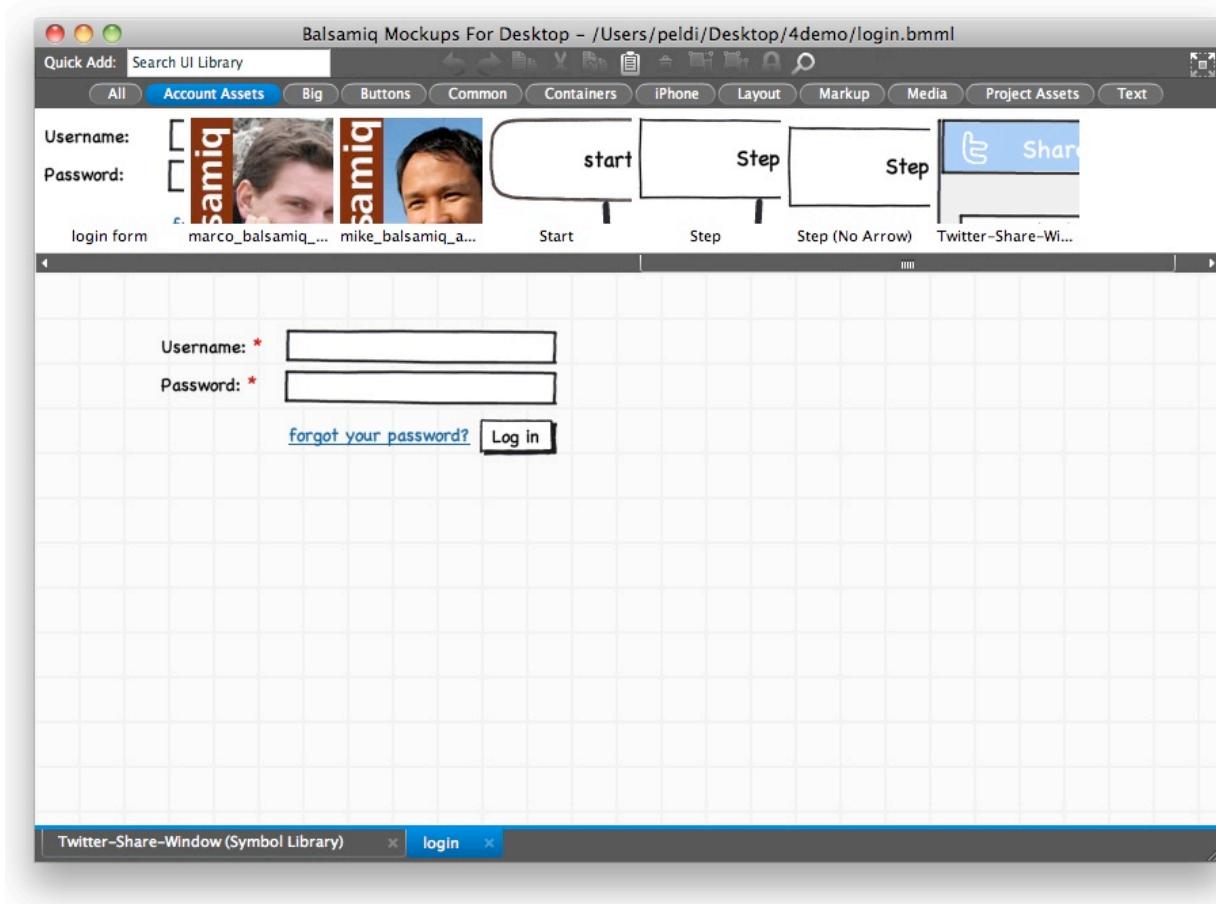
Balsamiq Mockups Documentation

To help uber-power-users like yourself, Mockups has a concept of an "account assets" folder. By default Mockups looks for it in your Documents folder, in a directory called "Balsamiq Mockups", but you can specify where Mockups should look for it via the [config file](#).

The easiest way to do this is shown below:

1. Create a folder under Documents called "Balsamiq Mockups" (Important: It must be named "Balsamiq Mockups")
2. Create a subfolder under Balsamiq Mockups called "assets"





If you put images or BMML files in a folder called "assets" inside of a folder called "Balsamiq Mockups" inside of your documents folder, those images and symbols will be accessible from any of your projects, under the "Account Assets" folder in your UI Library, which magically shows up only if the account assets folder exists and has something in it.

I'm writing this smaller because it's so exciting you might faint: consider creating a shared Account Assets folder in [DropBox](#), and tell everyone on your team to point their config files to that folder. Let that sink in for a minute. I know, wow. WOW.

Deleting Symbols

How you delete symbols depends on how they were created. Let's review how symbols are created.

- Symbols created by right clicking and selecting "Convert to symbol" are saved into the project symbol library.
- Symbols can be created by explicitly saving a .bmml file into a project or account assets folder.

The 1.0 release of symbols is powerful, but a lot of this management of symbols is a manual process, and is a little hard right now, but we hope to improve it with each subsequent release.

1. Deleting symbols saved to symbols.bmml

If you've created symbols using the "Convert to Symbol" method, your symbols are in the symbols.bmml file.

- The easiest way to delete these are to open a file in your project and click the project assets tab in the library. When you hover over symbols you'll know if they're in the symbols.bmml library if they read, for example, "My Menu (from symbols.bmml)".
- Now click the symbol in the library that you want to delete, right-click and select "Edit Symbol Source." This will open the symbol in the symbols.bmml file. The background will be green and your symbol will be visible.
- Select your symbol group, and delete it by pressing the delete key or the menu Edit > Delete.
- Save the file.

Now the symbol is gone, but you will also have "Symbol not Found" instances wherever you placed the symbol. You'll need to remove those as well.

2. Deleting individual symbol files

A less common method of creating symbols is to save .bmml files to an assets folder. You'll know when a symbol is a unique file when you hover over the symbol in the library. It will read, for example, "My Awesome Menu (from My Awesome Menu.bmml)".

The easiest way to get rid of these is to open the asset folder the symbol is saved to, and then delete the file.

Now the symbol is gone, but you will also have "Symbol not Found" instances wherever you placed the symbol. You'll need to remove those as well.

Next Steps and Feedback

At the moment, Symbols only work in Mockups for Desktop, and you cannot nest symbols inside of other symbols. If those features are important to you, let us know!

Keyboard Shortcuts

Below is a list of all the keyboard shortcuts and modifiers which you can use in Balsamiq Mockups. While all of the shortcuts will work in the Desktop version, some might not work when you edit mockups in a browser.

If you are running OS X, you can use the Command Key (the Apple Key) wherever you read CTRL in the list below, and the Option key whenever you see ALT.

You can also download the [Keyboard Shortcuts Cheat Sheet \(PDF\)](#) designed for us by Adrienne Travis here: <http://balsamiq.com/files/community/balsamiq-keyboard-shortcuts.pdf>

Working with Controls

Add to selection	SHIFT+CLICK
Toggle selection	CTRL+CLICK
Select All	CTRL+A
Edit the selected control's text	ENTER or F2
Resize maintaining aspect ratio	Hold SHIFT
Disable snapping during move or resize	Hold CTRL
Nudge selection 2px	arrow keys
Nudge selection 20px	SHIFT+arrow keys
Nudge-Resize selection 2px	CTRL+ALT+arrow keys
Nudge-Resize selection 20px	CTRL+ALT+SHIFT+arrow keys
Bring to front	CTRL+SHIFT+Arrow Up
Send to back	CTRL+SHIFT+Arrow Down
Bring forward	CTRL+Arrow Up
Send backward	Send backward
Clone selection	ALT+drag
Undo last command	CTRL+Z
Redo last command	CTRL+Y
Cut the selected controls	CTRL+X
Copy the selected controls	CTRL+C

Paste	CTRL+V
Paste In Place	CTRL+SHIFT+V
Delete the selected controls	DELETE or BACKSPACE
Duplicate the selected controls	CTRL+D
Select All	CTRL+A
Group the selected controls	CTRL+G
Ungroup the selected groups	CTRL+SHIFT+G
Lock the selected controls	CTRL+2
Unlock all locked controls	CTRL+3
Align selected controls left	CTRL+ALT+1
Align selected controls center	CTRL+ALT+SHIFT+2
Align selected controls right	CTRL+ALT+3
Align selected controls top	CTRL+ALT+4
Align selected controls middle	CTRL+ALT+5
Align selected controls bottom	CTRL+ALT+6

Quick Add

Sending focus to Quick Add	/ (forward slash) or + (plus sign)
Navigating the Suggestion List	arrow keys
Closing the Suggestion List	ESCAPE
Sending focus back to the Canvas	ESCAPE

Working With Icons

Navigating the Suggestion List	arrow keys
Using the selected icon	ENTER
Closing the Icons Library	ESCAPE
Closing the Suggestion List	ESCAPE

Editing Text

Start Editing text of the selected control	ENTER or F2
Committing the current text	ENTER (single-line controls), CTRL+ENTER (multi-line controls) or click anywhere
Ignoring the current edit	ESCAPE

Working With Mockup Files

New Blank Mockup	CTRL+N
New Clone of Current Mockup	CTRL+SHIFT+N
Open a mockup file	CTRL+O
Save the current mockup file	CTRL+S
Save as...	CTRL+SHIFT+S
Close current mockup	CTRL+W
Close all mockups	CTRL+SHIFT+W
Export this mockup's XML	CTRL+E
Import mockup	CTRL+SHIFT+E
Export Snapshot to Clipboard	CTRL+SHIFT+C
Export Snapshot to PNG	CTRL+R
Export All Snapshots to PNG	CTRL+SHIFT+R
Print mockup	CTRL+P
Quit Balsamiq Mockups	CTRL+Q
Looping through the open files in Mockups for Desktop	CTRL+TAB and CTRL+SHIFT+TAB

Other

Hide/Show the UI Library	CTRL+L
Enter Full Screen View	CTRL+F

Balsamiq Mockups Documentation

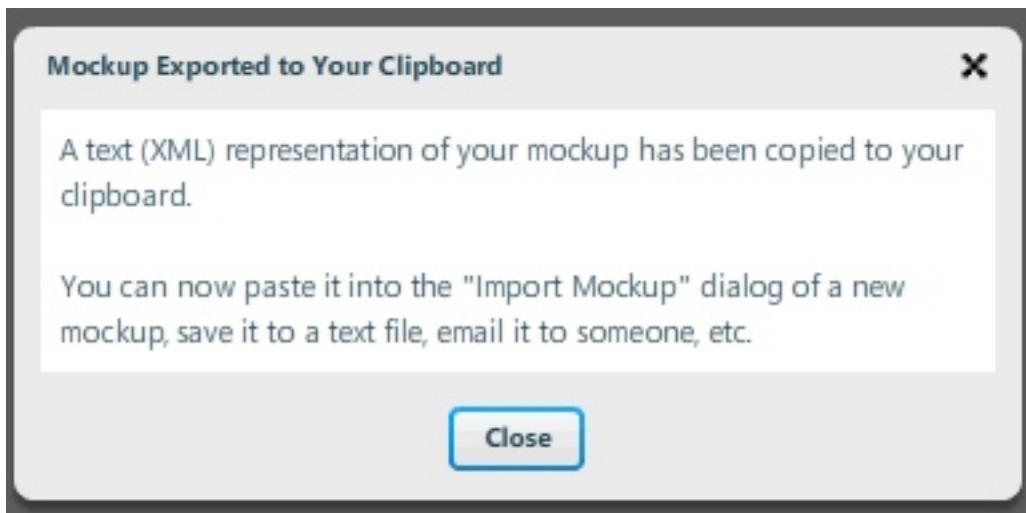
Exit Full Screen View	ESC
	type <i>lorem</i> in a Paragraph or Text Area. Try <i>l0rem</i> for a variation.
Show the full path of the current mockup	type <i>{mockup-path}</i> in a Label or Paragraph control
Show the current mockup name	type <i>{mockup-name}</i> in a Label or Paragraph control

Importing and Exporting

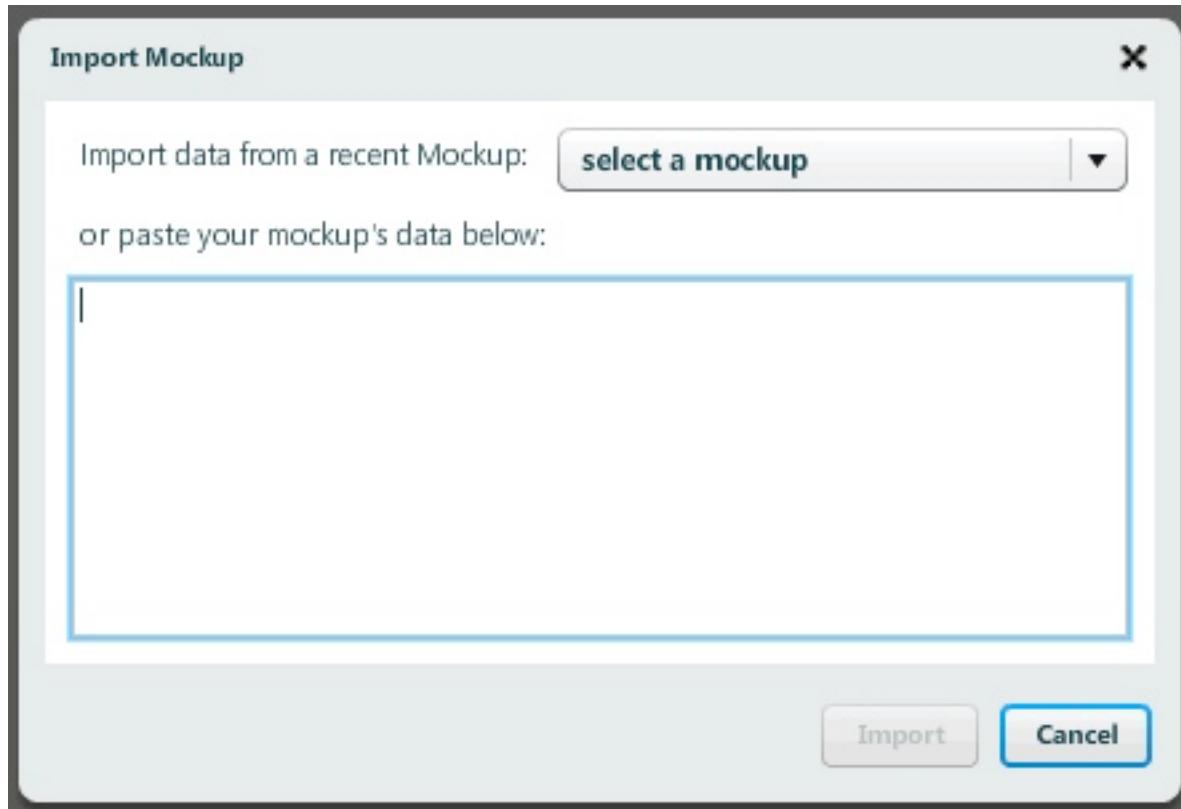
Exporting and Importing via XML

Balsamiq Mockups gives you the ability to export your mockup as an XML file, which is useful for creating copies of your mockup, or emailing the mockup to someone else, or possibly exporting it to another format (bring your own code for this).

Exporting a mockup is easy, just select "Export Mockup..." from the Mockup menu, and you'll see this dialog:



Once you have the XML, you can use it to create a new mockup by just importing it via the "Import Mockup" dialog:



Note that importing a mockup will erase the contents of your mockup canvas

Exporting to an image

If you just want to get an image of your mockup to send around, you can simply select the menu, File > Export to PNG image to export a single image.

If you have multiple Mockups open, you can export all to PNG images using the menu, File > Export All to PNG...

After exporting, a notification window will appear to show you where your PNG images were saved. Click the notification to open the containing folder.

Exporting to PDF

Mockups allows you to export open files to PDF, which can be useful for sharing single mockups or decks of Mockups for a project.

To export, open the files you want to export and select the menu File > Export all to PDF. Each individual mockup is exported on a page in the PDF deck, with 5cm of margin on each side.

To export an entire project, open all the files from the project, and then use the Export menu. Read the [Working With Projects](#) tutorial first for best practices on organizing your files in projects.

After exporting, a notification window will appear to show you where your PDF was saved. Click the notification to open the containing folder.

Note about Acrobat Fullscreen Mode and links:

If you're using fullscreen mode in Acrobat with a Mockup that has links, you may have to change some preferences to get links to function properly. To get things to work, follow these steps:

1. Select the menu: Edit -> Preferences
2. Select: Full Screen -> Full Screen Navigation
3. Un-tick the "Left click to go forward one page,..."

Printing

To print, select the menu File > Print or use the keyboard shortcut Ctrl+P (CMD+P on the Mac).

First, a Page Setup dialog will first appear, allowing you to select the page size, orientation, format, and any custom margins that your printer allows.

Finally, your Print dialog will appear. Select OK to print

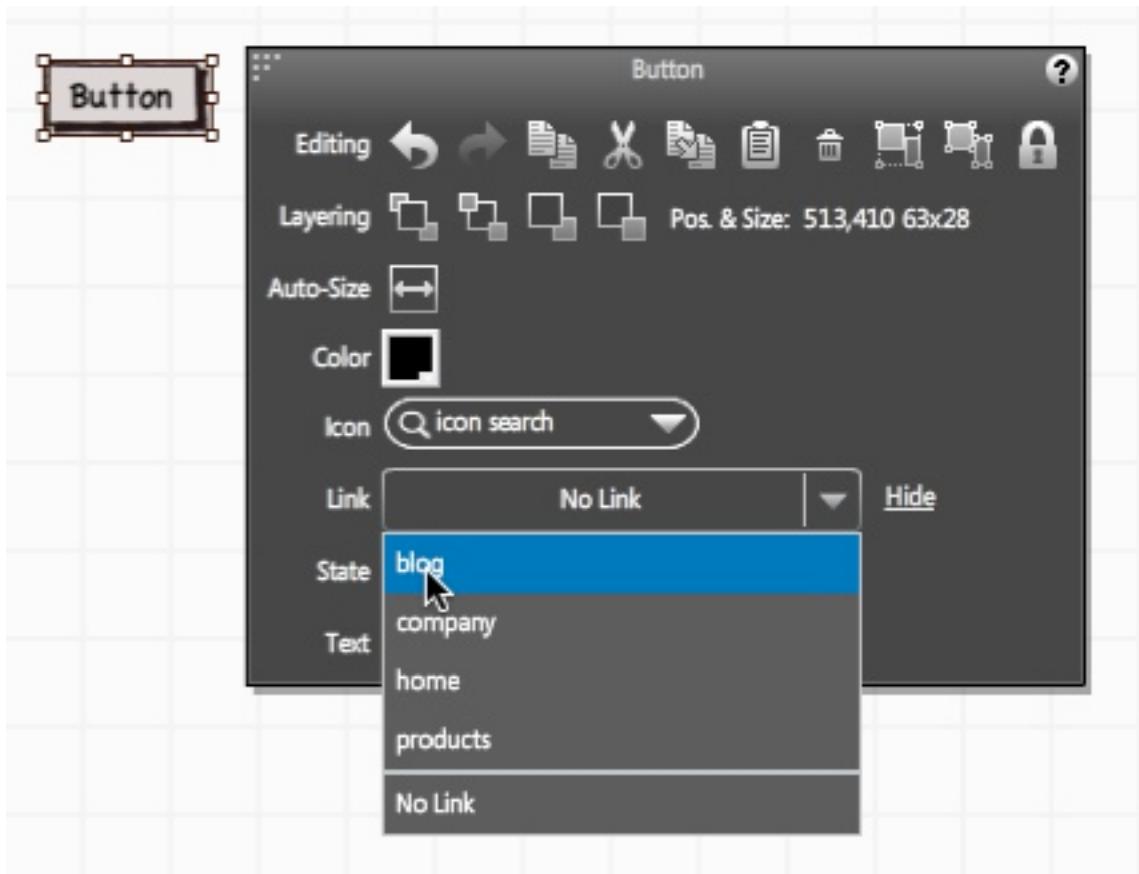
Note: Mockups automatically sets a top and left margin on your printed documents using an algorithm based on the dimensions of your mockup, but you may want to set a margin manually. When you're presented with the page setup dialog above, you may set a margin for the printer of your choice if the printer driver on your operating system allows you to.

Linking Mockups Together

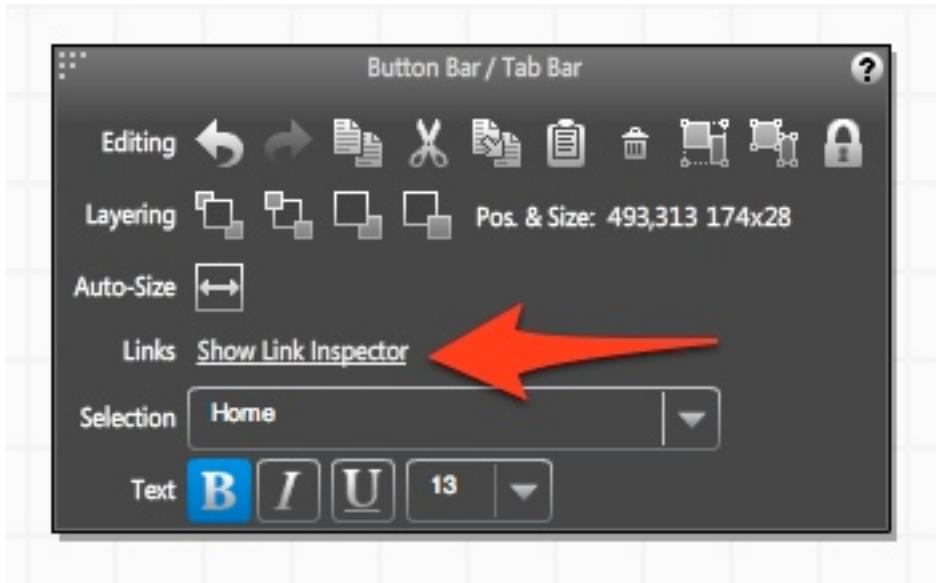
Starting with version 1.6 it is possible to link different mockups together in order to create a simple prototype for your site or software application. This can be useful for demonstrating click-through prototypes or for usability testing when you're displaying your project Mockups in [Full Screen Presentation](#) mode and as an [exported PDF](#).

At the moment the linking feature only works in the Desktop and Confluence versions, but we are actively working on adding linking capabilities to the plugin versions of Mockups as well.

Linking two mockups together as easy as choosing a file name from a pull-down menu. Simply click on a control that supports linking (most do, and [here's a workaround for those which don't](#)), and you'll see a new "Link" pull-down in the floating property inspector.



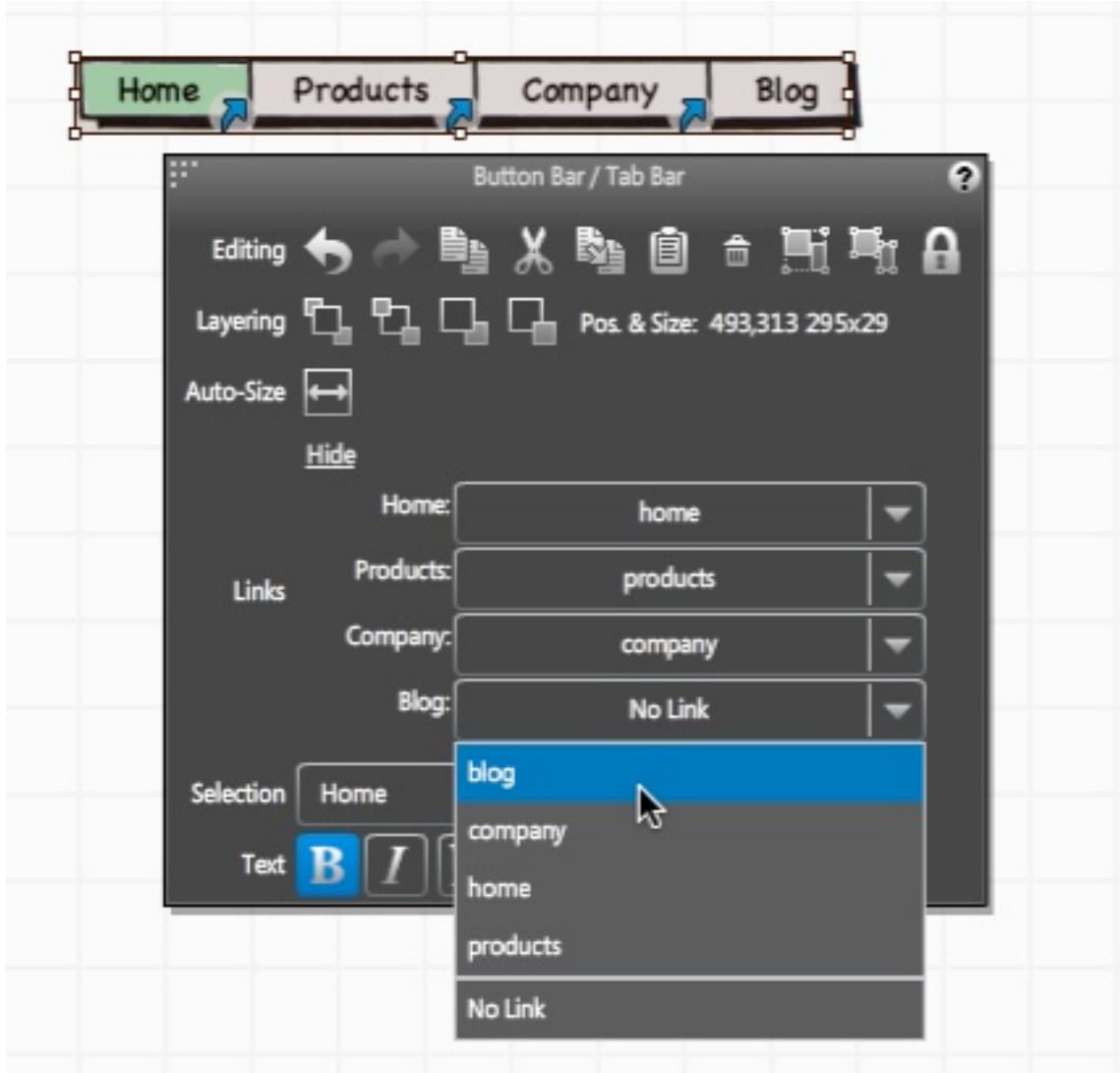
If you don't see the pull-down, you should click on the "Show Link Inspector" link:



If the pull-down is disabled, please save your mockup and select the control you want to link from again.

The list is pre-populated with the names of all the BMML files found in the same folder as the file you're editing. As long as you [keep all the related mockups in the same folder](#), this should be all you need.

For controls that can have multiple targets, you'll see something like this instead:

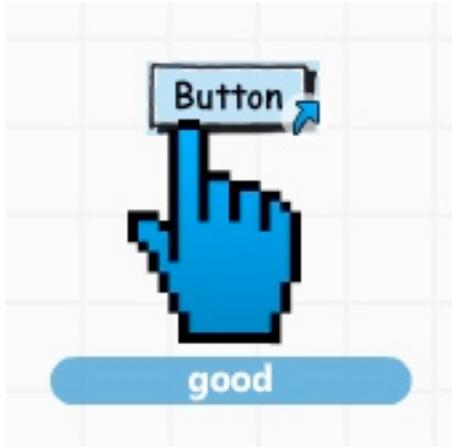


Once you have selected a mockup to link to from the pull-down menu, the control will show a little arrow icon at their bottom-right corner, as shown in this screenshot:



When you're happy with your link structure simply enter Full Screen Presentation mode (see[details here](#)) and you'll be able to click on the links you set up to test your prototype.

Controls with links will have a little blue overlay on them as well as the link icon and mousing over them will show a big hand pointer with the name of the target file for this link, like so:



Simply click on a link to load up the referenced mockup file.

Actually, there's one more thing: you can use the left and right arrow keys to go back and forth between the mockups you have visited in your run-through.

We have a quick screencast (38 seconds, no audio) of linking up three existing mockups and running through the resulting prototype.

► Watch on YouTube: <http://youtu.be/s2ZbMRikQRE>

Presenting Your Work

Mockups includes a "Full Screen Presentation" mode for those times when you need to present your wireframes to other stakeholders.

To enter Full Screen Presentation mode, just click on the full-screen icon in the top-right corner of Mockups, or select "Full Screen Presentation" from the View menu, or simply hit CTRL+F.

Mockups will expand to take up your whole screen (to eliminate distractions), and your mockup will be centered on the screen.

You will also see a big blue arrow pointer appear. The pointer will point towards the center of the screen in order to always stay out of the way of your mockup as much as possible.

You will also see three little buttons in the lower-right corner of your screen. They'll fade out in a few seconds, but you can always bring them back by mousing over the bottom-right corner of the screen.



The first toggle is for showing and hiding linking hints and the big pointers. You can now turn them off if you're using the presentation mode to test a new UI on some user and don't want to give them hints on where to click, or simply if you find the big pointer distracting.

The second toggle is for showing and hiding markup elements, such as Sticky Notes, Arrows and Callouts (any control under the "Markup" tab in the UI Library). This is useful if you want to just look at your UI without the annotations that surround it.

The third button lets you exit full-screen mode and go edit the mockup you're currently viewing.

The state of the two toggles is remembered across sessions, and in the Desktop version you can use the "m" (or "M") and "l" (or "L") keys to toggle the two settings as well.

To exit Full Screen Presentation mode, just hit the ESC key.

Note: Toggling Link Hints and Cursor off ("L" key) gives you access to your mouse scroll wheel or trackpad 2-finger scrolling on Macs.

Sharing Images, Symbols and Icons Between Projects using Account Assets

To get the most out of Balsamiq Mockups it's important to understand the concepts of Project Assets and Account Assets.

Project assets are described in details in the "Working With Projects" tutorial. You should read that one first if you haven't yet.

This document focuses on Account Assets, the Balsamiq Mockups way to share images, symbol libraries and even icons between all the different projects on your computer.

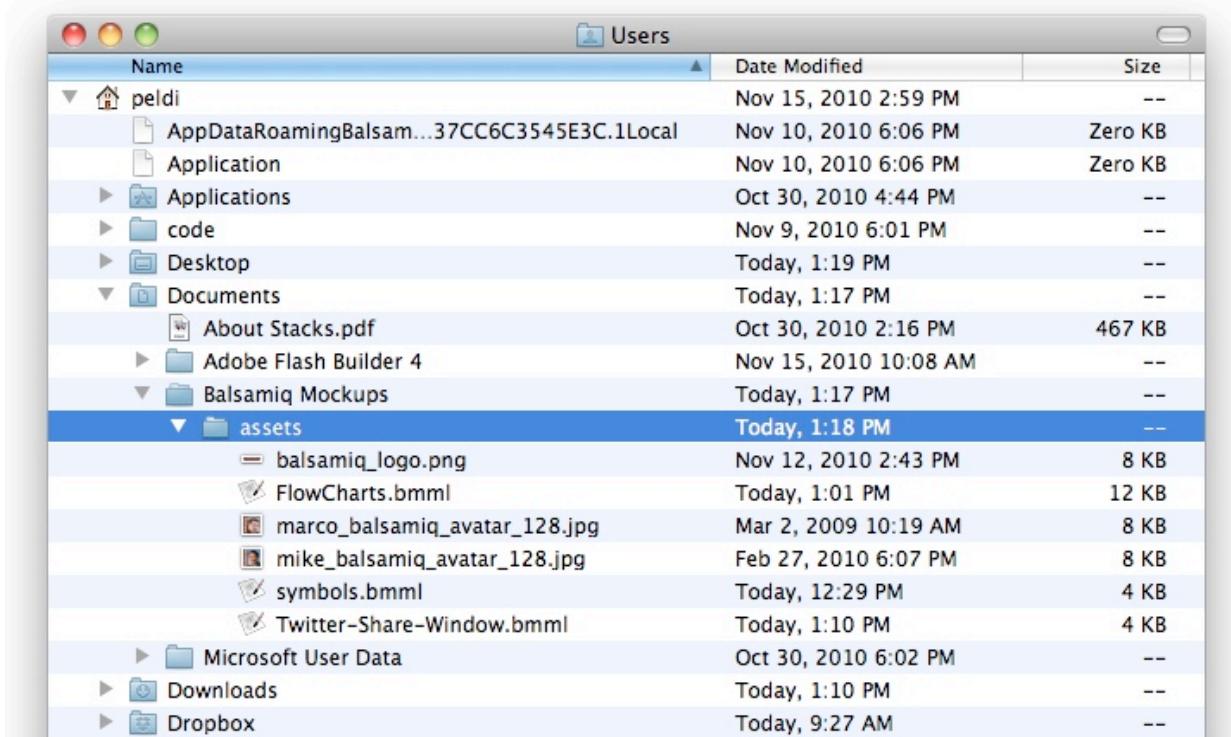
NOTE: Account Assets only work in Mockups for Desktop and in myBalsamiq right now. In myBalsamiq, Account assets are called Site assets. Please mentally replace Account with Site if you're using myBalsamiq.

Setting Up the Account Assets Folder in Mockups for Desktop

Account Assets are considered an advanced feature, so it's not turned on by default in Mockups for Desktop (Site Assets are always available in myBalsamiq).

To turn it on, you have to create two directories/folders on your computer, and restart Mockups for Desktop.

- On Windows and Linux, create a directory called Balsamiq Mockups inside your Documentsdirectory. Inside the new Balsamiq Mockups directory, create a new directory called assets.
- On Mac OS X, create a folder called Balsamiq Mockups inside your Documents folder. Inside the new Balsamiq Mockups folder, create a new folder called assets.



That's your Account Assets folder.

Any image, symbol library or custom icon file that you put in that folder will show up in Mockups as an Account Asset.

Using Account Assets

You can get to images saved in your Account Assets folder via the image selection dropdown, the UI Library and via Quick Add.

You can get to Symbols saved in Symbol Libraries saved in your Account Assets folder via the UI Library (click on the Account Assets tab), or via Quick Add.

You can get to custom icons via the Icon Library, or via the Icon Search box.

The BalsamiqMockups.cfg Configuration File

Balsamiq for Desktop allows for some customization via a simple XML file called BalsamiqMockups.cfg.

Saving your Config File

You create the file in a text editor (you can use [this sample BalsamiqMockups.cfg file](#) as a starting point) and save it in the following location:

- **On Windows XP:**

C:\Documents and Settings\<username>\Application Data\
BalsamiqMockupsForDesktop.EDE15CF69E11F7F7D45B5430C7D37CC6C3545E3C.
1\Local Store

- **On Mac OS X:**

<your user home>/Library/Preferences/
BalsamiqMockupsForDesktop.EDE15CF69E11F7F7D45B5430C7D37CC6C3545E3C.
1/Local Store

Note for Mac OS X Lion users: the ~/Library folder is hidden by default. To make it visible again, open a terminal window and enter: chflags nohidden ~/Library

- **On Windows Vista and Windows 7:**

C:\Users\<your username>\AppData\Roaming\
BalsamiqMockupsForDesktop.EDE15CF69E11F7F7D45B5430C7D37CC6C3545E3C.
1\Local Store

- **On Linux:**

~ / . a p p d a t a /
BalsamiqMockupsForDesktop.EDE15CF69E11F7F7D45B5430C7D37CC6C3545E3C.
1/Local Store

Note: config file must be saved to plain text format!

This will NOT work if you save your config file to RTF, or with the .cfg.txt or .cfg.rtf extensions. If you are usingTextEdit.app, to save in plain text format select the menu, Format > Make Plain Text.

Sample File

The config file might look like this:

```
<config>
<fontFace>Architect Small Block</fontFace>
```

```
<rememberWindowSize>false</rememberWindowSize>
<useCookies>false</useCookies>
<documentsPath>C:\Documents And Settings\Administrator
\Desktop</documentsPath>
<snappingEnabled>false</snappingEnabled>
<selectionColor>FF0000</selectionColor>
<transparentBackground>true</transparentBackground>
</config>
```

Options

- the fontFace tag allows you to specify the name of a font face on your computer to use instead of Comic Sans MS
- set the rememberWindowSize tag to false if you'd like Mockups to not try to remember the size and position of the application window (some video card drivers don't play well with Adobe AIR and Mockups won't start unless you set this to false)
- set useCookies to false if you'd like Mockups to not save any of your preferences on the current computer (preferences include the state of "show UI Library", "Show Notebook" and others)
- if you don't want to use the classical "My Documents" folder (often because it is on a remote server, not always connected) for the shared data of Mockups, then you can set documentsPath to the path of a folder of your choice. This is also useful to [share symbols across projects](#).
- set snappingEnabled to false if you'd like to turn off snapping altogether in the app.
- set selectionColor if you want to change the color we use for showing selection in the app.
- set transparentBackground to true if you want your PNGs to be exported with a transparent background instead of the default white one.

Command Line Options

Mockups for Desktop can be controlled via the command line with the following commands:

Commands

```
Balsamiq Mockups export inputBMMFilePath outputImagePath
```

exports the provided bmml file to a PNG file

```
Balsamiq Mockups export inputBMMFolderPath  
outputImagesFolderPath
```

exports all the bmml files in the the provided folder to PNG files

```
Balsamiq Mockups register organizationName serialKey
```

registers this Mockups copy with the provided serial key info

```
Balsamiq Mockups unregister
```

un-registers Mockups from this computer

```
Balsamiq Mockups version
```

prints the current version of Mockups for Desktop

```
Balsamiq Mockups file_1.bmml file_2.bmml...
```

opens Mockups for Desktop and loads the specified files (use absolute paths)

Notes:

- You only need to register your copy once per machine, then you can export PNGs from your Mockups' XML files whenever you need to.
- The export process is near-instantaneous, and you should only see a quick "flashing" of the app.

Log

The output of the application run will be in a BalsamiqMockups.log file which you can find in the following locations:

- **On Windows XP:**

C:\Documents and Settings\[your username]\Application Data
\BalsamiqMockupsForDesktop.#numbers#\Local Store

- **On Mac OS X:**

[your user home]/Library/Preferences/BalsamiqMockupsForDesktop.#numbers#/Local Store

- **On Windows Vista and Windows 7:**

C:\Users\[your username]\AppData\Roaming\BalsamiqMockupsForDesktop.#numbers#\Local Store

- **On Linux:**

~/.appdata/BalsamiqMockupsForDesktop.#numbers#/Local Store

Starting with the v1.6.65 (released on Feb 24th, 2010), you can also specify a full path to a log file as an additional parameter to any of the instructions above and Mockups will log to it instead of the default file.

Additional Resources

You can find some scripts to export multiple mockups using the command-line [here](#).

► http://community.balsamiq.com/balsamiq/topics/shell_scripts_to_export_multiple_mockups

Looking for information on how to perform a silent installation of Mockups for Desktop on many end-user machines? [Here you go](#).

► <http://support.balsamiq.com/customer/portal/articles/133390>

BMML File Format Specification

BMML is the Balsamiq Mockups Markup Language, the flavor of XML used by Balsamiq Mockups to save its data.

This document outlines the BMML file format, in the hope that this will be useful to you when integrating Mockups in your daily work. You could for instance import BMML into your tool, write a BMML parser which generates HTML or MXML or XAML or running code...the sky is the limit!

Warning: there are some things that aren't documented here, like what properties each control type supports. To figure those things out, just open Mockups, drag a control type to the stage, change some properties in the property inspector and look at the resulting BMML. In other words, we try our best to keep this document up-to-date, but you should trust shipped code more. :)

Remember that XML syntax is case-sensitive. `<tag>` and `<Tag>` are different.

Quick Overview

Here's a sample BMML containing a single Callout control:

```

<mockup version="1.0" skin="sketch" measuredW="941"
measuredH="169" mockupW="36" mockupH="40">

<controls>
  <control controlID="1"
    controlTypeID="com.balsamiq.mockups::CallOut"
    x="644" y="129" w="-1" h="-1"
    measuredW="36" measuredH="40"
    zOrder="1" locked="false" isInGroup="-1">
    <controlProperties>
      <text>Hello!</text>
      <backgroundAlpha>0.25</backgroundAlpha>
      <color>65280</color>
    </controlProperties>
  </control>
</controls>

```

```
</controls>
</mockup>
```

As you can see, the top-level tag is a 'mockup' tag. Within it, there's a 'controls' tag, which includes a list of 'control' elements, one for each control in the mockup. Each control elements can include an optional 'controlProperties' element, with different children depending on the type of element. We will see below what all the attributes mean in details.

The Mockup Tag

Here's the DTD snippet for the mockup tag:

```
<!ELEMENT mockup ( controls? ) >
<!ATTLIST mockup measuredW NMTOKEN #REQUIRED >
<!ATTLIST mockup measuredH NMTOKEN #REQUIRED >
<!ATTLIST mockup mockupW NMTOKEN #REQUIRED >
<!ATTLIST mockup mockupH NMTOKEN #REQUIRED >
<!ATTLIST mockup skin NMTOKEN #REQUIRED >
<!ATTLIST mockup version NMTOKEN #REQUIRED >
<!ELEMENT controls ( control? ) >
```

A 'mockup' tag includes a 'controls' tag, described below. measuredW and measuredH are the dimensions, in pixels, of the mockup, including the top-left white space around it. mockupW and mockupH (added on 9/14/2009) are the dimensions of the mockup without any space around it. In other words, this would be the size of the PNG if you exported it. The skin tag is not used right now, and is always "sketch". The version tag is for the version of the BMML specification, which is 1.0 so far. The controls tag is simple, it doesn't have any attributes and just contains a list of 'control' tags.

The Control Tags

Here's the DTD snippet for the control tag:

```
<!ELEMENT control ( controlProperties? ) >
<!ATTLIST control controlId NMTOKEN #REQUIRED >
<!ATTLIST control controlTypeID NMTOKEN #REQUIRED >
<!ATTLIST control w NMTOKEN #FIXED "-1" >
<!ATTLIST control h NMTOKEN #FIXED "-1" >
<!ATTLIST control measuredH NMTOKEN #REQUIRED >
```

```
<!ATTLIST control measuredW NMOKEN #REQUIRED >
<!ATTLIST control x NMOKEN #REQUIRED >
<!ATTLIST control y NMOKEN #REQUIRED >
<!ATTLIST control zOrder NMOKEN #REQUIRED >
<!ATTLIST control isInGroup NMOKEN #FIXED "-1" >
<!ATTLIST control locked NMOKEN #FIXED "false" >
```

The control tag can contain a controlProperties tag, described below.

- controlID is always unique for this list of controls and identifies this control's instance in the mockup controlTypeID is the control type (listed below).
- w and h represent the size in pixels of the control.
- A value of -1 means that you should look at measuredW or measuredH instead.
- measuredW and measuredH represent the size in pixels of the control, when shown in its 'natural state', i.e. it's the preferred dimension of the control as dictated by the control itself. For instance, for a Label control, measuredW would be large enough to show the whole text with no cropping.
- x and y represent the position in pixels of the control, relative to the top-left corner of the canvas.
- zOrder represents the layer ordering of this control.
- Values are unique and sequential within the current list of controls.
- isInGroup tells you if a control is part of a group (-1 means "no", otherwise it uses the controlID of the group)
- locked tells is a flag that tells you...you guessed it!

The ControlProperties Tag

The controlProperties tag contains a child for each property that can be modified for a control (using the [property inspector](#)). Here's a list of the different control properties and where they are used:

- alternateRowColor, rowHeight, hasHeader, vLines, hLines: used for Lists and Data Grids controls
- backgroundAlpha (0..1): the transparency of the background
- bgPattern (one of 'allWhite', 'allBlack', 'topOnly', 'topBlueBottom', 'topBlackBottom'), topBar (bool): used by the iPhone control
- bold, italic, underline, align, size: used by any control with text in it

- borderStyle ('none' or 'square'): used by all controls which can show a border or not
- close, minimize, maximize, dragger, topheight, bottomheight: used by the Dialog / Window control
- color (int): the color of the text or the background of a component, depending on the control type
- customData (String): added on 10/21/2009, see [here](#) for details
- customID (String): added on 10/21/2009, see [here](#) for details
- filter: used by the Image control: true if the image should be sketched
- direction ('left', 'center', 'right', 'bottom', 'top'): used by Arrows, Tabs, Curly Braces, Pointy Button etc
- href: used by all controls that can have a single link to other controls, like Button, Canvas, etc. Mockups saved before 9/14/2009 only contain the file name to link to, i.e. foo.bmml with no path. Mockups saved after 9/14/2009 contain a string of this form: <name>&bm;<viewURL>&bm;<loadURL>&bm;<editURL>. The last 3 values are all the same for Mockups for Desktop but are different in the plugin versions.
- hrefs: like href, but used by controls that link to multiple files, like the Accordion, Link Bar, etc. It's a comma-separated list of strings formatted the same way as a single href string (see previous bullet).
- icon: a string composed of both iconID of the icon to use for this control and its size, like this "iconID|48". There are over 220 icons, each with a unique ID (see below). The sizes are either 16, 24, 32 or 48. As of version 2.1.7, the left part can also be a src, either ./assets/ filename or \$ACCOUNT/assets/filename, based on where the custom icon is loaded from.
- indeterminate (bool) and scrollbarValue (0-100): used for Progress Bar controls
- labelPosition ('bottom' or 'right'): used in the Icon and Label control
- leftArrow (bool), rightArrow (bool), curvature (-1, 0, 1): used by the Arrow controls
- map (string): added on 9/14/2009: an escaped image map for the linked areas of the mockups, following the same format as the HTML map tag.
- onOffState ('on' or 'off'): used by the Switch control
- orientation ('portrait' or 'landscape'): used by the iPhone, iPhone control
- override: used by the Component control (see below)
- position ('left' or 'right'): used by Vertical Tab Bar
- selectedIndex (-1 for none): the selected item in a list
- src: used by the Image and Component controls. It's either a relative path (for images loaded from disk) or an http url. If it starts with \$ACCOUNT/assets, it means the image is in the account assets folder.

- state ('up', 'selected', 'focused', 'disabled', 'disabledSelected'): the state of the control
- tabHPosition and tabVPosition: used by the Tab Bar controls
- tooltipDirection (NW, N, NE, E, SE, S, SW, W): used by the tooltip control
- value: the value of a scrollbar, from 0 to 100
- verticalScrollbar (bool) and value (0-100): whether the control has a scrollbar or not, and its value (0 means "top", 100 means "all the way down")

The Component Control

The Component control, known as "Symbol" in Mockups, contains a link to another bmml (here are [the docs for the Symbols feature](#)). The link is an url with an optional anchor. If the anchor is missing, then the whole bmml is the target. Otherwise, the target is a group which has the same name of the anchor. The group must be at top level in the bmml.

```
<control controlID="1"
    controlTypeID="com.balsamiq.mockups::Component"
    x="588" y="92" w="407" h="314"
    measuredW="407" measuredH="314"
    zOrder="1" locked="false" isInGroup="-1">
    <controlProperties>
        <override controlID="1" x="0" y="0" w="116" h="70">
            <color>7775663</color>
            <text>This text is overridden</text>
        </override>
        <src>./assets/symbols.bmml%23Options%20dialog</src>
    </controlProperties>
</control>
```

The ControlProperties tag of a Component control can contain an override tag, for the properties of the Symbol that have been overridden in this symbol instance. Attributes of override tag:

- controlID: id of the overridden control as in the symbol definition. In case the control is nested in groups, then the controlID is the relative path to the control (for example, "2:3:0" is the control with id 0 belonging to group 3 nested in group 2 in the symbol definition).
- x, y, w, h: (optional) these attributes override the original size and position of the control. Measured dimensions are not saved in the bmml.

It is not possible to override the zOrder attribute. It is not possible to add or remove controls in the symbol instance.

Control Type IDs and Icon IDs

All the control type IDs start with the "com.balsamiq.mockups:" prefix. Here's the full list of the control type IDs supported as of today (case sensitive): AddCommentIcon, AddPageIcon, AIMIcon, AlarmClockIcon, Alert2Icon, AlertIcon, ArchiveIcon, AsteriskIcon, BackSpaceIcon, BalloonIcon, BatteryIcon, BloggerIcon, BoldIcon, BookIcon, BulletsIcon, CalendarIcon, CameraIcon, CartIcon, CellBarsIcon, CellPhoneIcon, ChartColumnIcon, ChartLineIcon, ChartPieIcon, CheckMarkIcon, CircledMinusIcon, CircledPlusIcon, CircledXIcon, CircleIcon, ClockIcon, CloudIcon, ClubIcon, ColumnsIcon, CommentsIcon, ContactIcon, ContactsIcon, CopyIcon, CopyrightIcon, CoverFlowIcon, CreativeCommonsIcon, CursorIcon, CutIcon, DashIcon, DataBaseIcon, DeliciousIcon, DiggIcon, DocEmptyIcon, DocExcelIcon, DocPDFIcon, DocWordIcon, DollarIcon, DotHollowIcon, DotIcon, DoubleArrowIcon, DownArrowIcon, DownFillTriangleIcon, DownLeftArrowIcon, DownloadIcon, DownRightArrowIcon, DownTriangleIcon, EjectIcon, EllipsisIcon, EmailIcon, EmoticonIcon, EmptyIcon, EuroIcon, ExclamationPointIcon, ExportIcon, EyeIcon, FacebookIcon, FileIcon, FilterIcon, FlagIcon, FlickrIcon, FolderClosed, FolderIcon, ForbiddenIcon, ForwardChapterIcon, ForwardIcon, FriendFeedIcon, FullScreenExitIcon, FullScreenIcon, GearIcon, GeoLocation2Icon, GeoLocationIcon, GMailIcon, GoogleIcon, GridIcon, GtCircledIcon, GtIcon, HandClosedIcon, HandOpenIcon, HandPointIcon, HeartIcon, Help2Icon, HelpIcon, HLinesIcon, HomeIcon, HomeMobileIcon, HourGlassIcon, IBeamIcon, ImageIcon, ImportIcon, IndentIcon, Info2Icon, InfoIcon, ItalicIcon, JewelIcon, JumpBackIcon, JumpOutIcon, LeftArrowIcon, LeftFillTriangleIcon, LeftTriangleIcon, LightBulbIcon, LightningIcon, LineIcon, LinkedInIcon, LinkGlobeIcon, LinkIcon, ListIcon, LockIcon, LockOpenIcon, MicIcon, MinusIcon, MusicIcon, Number0Icon, Number1Icon, Number2Icon, Number3Icon, Number4Icon, Number5Icon, Number6Icon, Number7Icon, Number8Icon, Number9Icon, NumberedListIcon, OpenIDIcon, OutdentIcon, PagesIcon, PaperClipIcon, PasteIcon, PauseIcon, PencilIcon, PhoneDialIcon, PhoneHangUpIcon, PlusBigIcon, PlusIcon, PopOutIcon, PortletEditIcon, PortletHelpIcon, PortletMaximizeIcon, PortletMinimizeIcon, PoundIcon, PowerIcon, PrintIcon, PushPinIcon, RecycleIcon, RedditIcon, RedoIcon, RegisteredIcon, ReloadIcon, RepeatIcon, RewindChapterIcon, RewindIcon, RightArrowIcon, RightFillTriangleIcon, RightTriangleIcon, RSSIcon, SadFaceIcon, SaveIcon, ScreenIcon, SearchIcon, ShapesIcon, ShareIcon, ShiftIcon, ShuffleIcon, SimpleArrowIcon, SkypeIcon, SlideShareIcon, SmallClockIcon, SpadesIcon, SpinningIcon, SquaredMinusIcon, SquaredPlusIcon, StarIcon, StopIcon, StrikeThroughIcon, StumbleUponIcon, SuitcaseIcon, TableIcon, TagIcon, TapeIcon, TelephoneIcon, TextAlignCenterIcon, TextAlignLeftIcon, TextAlignRightIcon, ThreeGIcon, ThumbnailsIcon, ThumbsDownIcon, ThumbsUpIcon, TrademarkIcon, Trash2Icon, TrashIcon, TruckIcon, TwitterIcon, UnderlineIcon, UndoIcon, UpArrowIcon, UpdateIcon, UpDownFillTriangleIcon, UpFillTriangleIcon, UpLeftArrowIcon, UpRightArrowIcon, UpTriangleIcon, UserIcon, UsersIcon, VideoCameraIcon, VLinesIcon, VolumeIcon, WirelessIcon, WorldIcon, WriteIcon, XIIcon, YIMIcon, YouTubeIcon.

Custom Properties

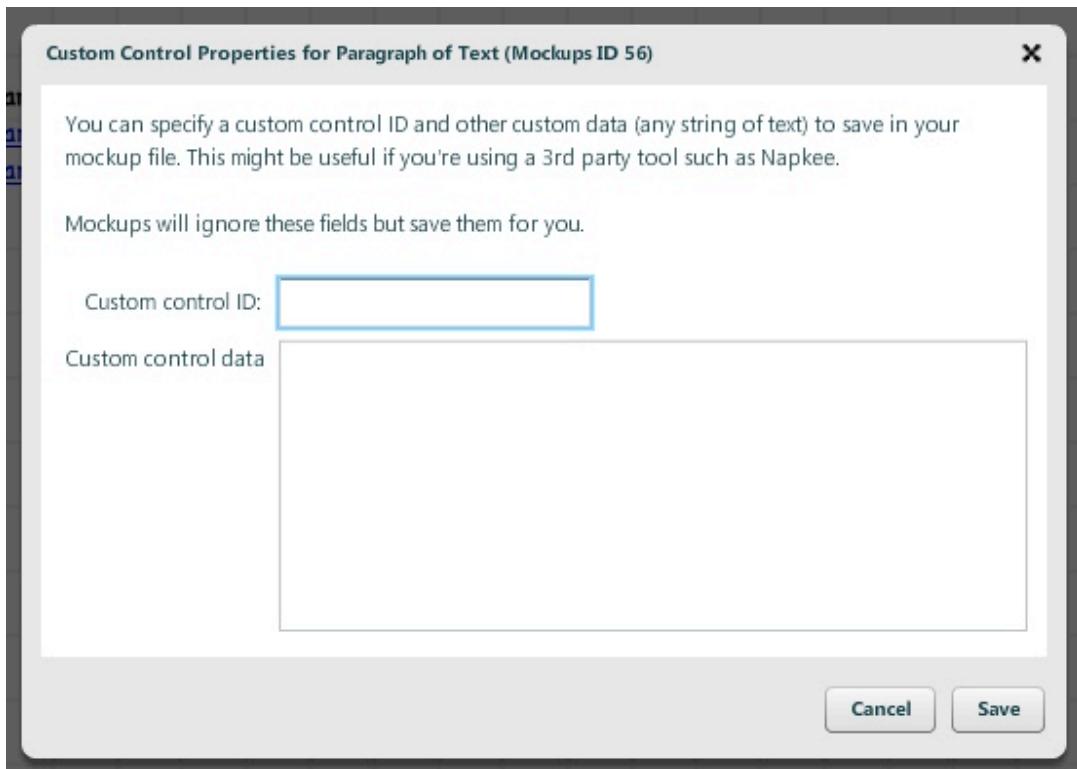
This feature is especially useful if you're using one of the [3rd party tools that extend Mockups](#), such as [Napkee](#) for instance.

Using Custom Properties

You can now right-click on any control of your mockup and select "Custom Properties..." from the little context menu:



This will open a dialogue like the one below:



Like the dialog says, you can enter a custom ID for the selected control and some custom data to go with it. Mockups will not use these properties directly, but it will save them in the BMML file along with the rest of your mockup.

Other tools can use this data to do some fancy stuff like allowing you to specify some javascript to run when the control is clicked or even just simply using the custom ID you specified in the HTML exported.

Enrico from [Napkee](#) put together a video showing how to work with Custom Data.

► Watch the Napkee video on YouTube: <http://youtu.be/RLe8F4wCOw0>

Get Notified of Developer News

If you are building an extension to Mockups and would like to be notified of these changes before anyone else, [email Peldi \(peldi@balsamiq.com\)](mailto:peldi@balsamiq.com) to be added to the bmmldev@balsamiq.com mailing list!