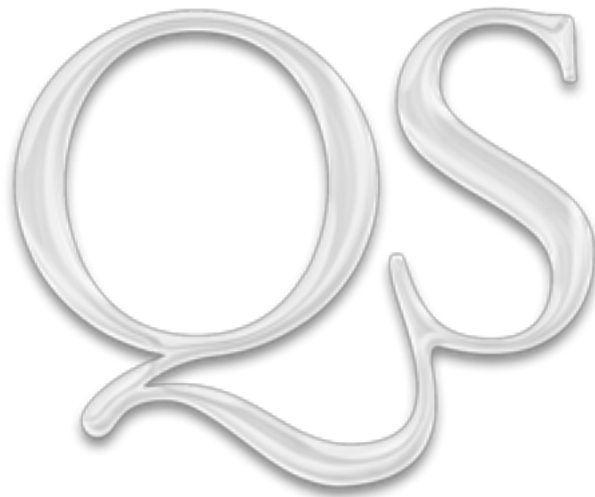


Quicksilver User's Guide

by Howard Melman

Version 0.11 (for Quicksilver version β51)



Please provide feedback on this manual at the Quicksilver forum in this thread:
<http://blacktree.cocoaforge.com/forums/viewtopic.php?t=3747>

© Copyright 2006 Howard Melman

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Table of Contents

Introduction	7
Concepts and Terminology	8
Installation	11
Invoking Quicksilver	12
<i>Selecting Items</i>	<i>13</i>
<i>The Comma Trick</i>	<i>14</i>
<i>Helping the Matching Algorithm</i>	<i>14</i>
<i>TextMate Ranker</i>	<i>16</i>
Configuration	16
<i>Preferences</i>	<i>16</i>
<i>Plug-ins</i>	<i>21</i>
<i>Catalog</i>	<i>23</i>
<i>Interfaces</i>	<i>26</i>
<i>Handlers</i>	<i>29</i>
<i>Notifications</i>	<i>29</i>
Triggers	30
<i>Trigger Basics</i>	<i>30</i>
<i>Hot Key Triggers and Proxy Objects</i>	<i>32</i>
<i>Using the Mouse</i>	<i>36</i>
<i>Abracadabra Mouse Gestures</i>	<i>37</i>
<i>Constellation Radial Menus</i>	<i>38</i>
<i>Command Objects and Droplets</i>	<i>41</i>

Files and Folders	42
<i>Basic File Actions</i>	42
<i>File Tagging and Spotlight</i>	44
<i>File Compression and Email</i>	45
<i>Pathfinder</i>	46
<i>File Transfer Programs</i>	46
<i>Disk Image</i>	46
Applications and Processes	47
iTunes	49
Contacts	51
<i>Address Book</i>	51
<i>Entourage</i>	53
<i>Now Contact</i>	54
Mail	54
<i>Mail.app</i>	54
<i>Gmail</i>	56
<i>Entourage</i>	56
<i>MailSmith</i>	56
<i>Other Mail Applications</i>	56
<i>Teleflip</i>	56
Chat	57
<i>iChat</i>	57
<i>Adium</i>	58
Calendar	58

<i>iCal</i>	58
<i>Google Calendar Module</i>	60
Web	60
<i>Browsers</i>	60
<i>Online Services</i>	61
<i>Web Pages</i>	63
<i>Web Searches</i>	63
<i>Adding Web Searches Manually</i>	65
<i>NetNewsWire</i>	66
Text	66
<i>Math Calculations</i>	69
Clipboard and Shelf	70
<i>Clipboard History</i>	70
<i>Clipboard Storage</i>	72
<i>Shelf</i>	72
<i>Comparing Clipboard History, Storage and Shelf</i>	73
<i>Cl1p.net</i>	73
Note Taking Applications	73
<i>VoodooPad</i>	73
<i>Yojimbo</i>	74
<i>WikityWidget</i>	75
Images	75
<i>iPhoto</i>	75
<i>Slideshow</i>	76
<i>Desktop Picture Action</i>	76

<i>Screen Capture</i>	76
<i>Image Manipulation</i>	77
<i>Flickr Upload</i>	77
System	77
<i>Devices</i>	77
<i>Networking</i>	79
<i>Configuration</i>	79
<i>Exposé</i>	80
<i>Extra Scripts</i>	80
Development	81
<i>Automator</i>	82
<i>Services</i>	82
<i>Terminal Programs</i>	82
<i>CLIX</i>	84
<i>Command Line Tool</i>	84
<i>Developer Documentation</i>	85
<i>Subversion</i>	85
<i>Remote Desktop</i>	86
Troubleshooting	86
Extending Quicksilver	87
Random Notes	87
<i>Results List</i>	87

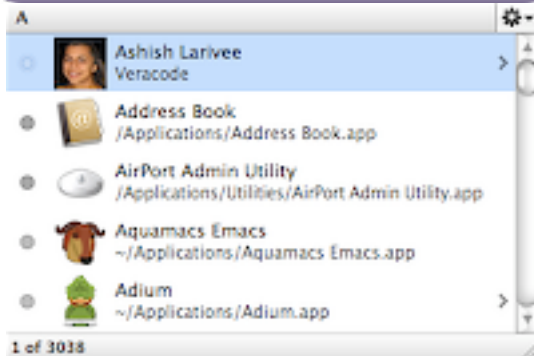
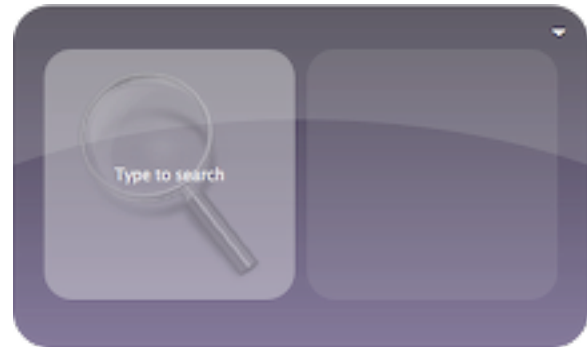
Introduction

When I want to do something on my Mac, my first reaction is to use Quicksilver. Whether I want to send an email, search for something on the web, open a bookmark, open a file, start an application, send an IM, revisit a web page I looked at yesterday, pause iTunes or find a song, or virtually anything else, I type ⌘-space and activate Quicksilver to do it. That makes Quicksilver very powerful, but also a little difficult to explain. I'll use an example of sending email to my friend Ashish.

First, I could open Mail, type ⌘N to open a new message, type enough of Ashish to have her address appear in the To: field and then tab to the Subject and continuing writing the message. I could instead open Address Book, search for Ashish's card, and ⌘-click (or right-click) on the email address and choose Send Email.

This is how I do it using Quicksilver. I type ⌘-space, A, Return. That's it. Let's walk through it. ⌘-space at any time, in any application, activates Quicksilver bringing up a two pane window shown here.

When I type A, Ashish appears in the first pane because I often send her email. Her picture appears because I have her picture in her Address Book entry (mostly because Quicksilver makes such good use of it). Also another window ap-

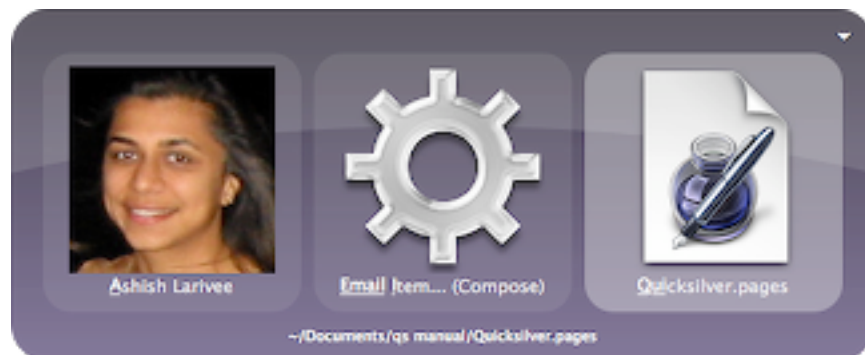


pears with other choices. If I kept typing, those would change to be some other choice, but since Ashish is what I want, I'm done. The second pane shows the **Compose Email** action which is what I want to do. Again this appears because it's the most common thing I do when I call up Ashish. Typing return performs the selected action so I see a new Compose Message window appear from Mail.app with Ashish's address filled in.

You might think using Quicksilver doesn't seem that much easier than the other methods I described, but the Quicksilver method is done entirely via keystrokes. There's no mousing to the Dock to open a particular icon or having to select a specific field, just type 3 keystrokes (one shifted). You might also think the comparison is unfair because I said some of the choices appeared so easily since I do this often, but that's one of the advantages of Quicksilver. It learns what you do and makes your most frequent tasks easier. The other methods don't change and don't get any eas-

ier.

Now let's say I wanted to send a document to Ashish. Instead of choosing the default **Compose Email** action, I can `tab` to the second pane, type `email i` to choose **Email item...(Compose)**, this opens a third pane which we `tab` to and type `~/Q` to choose the document in my home directory.

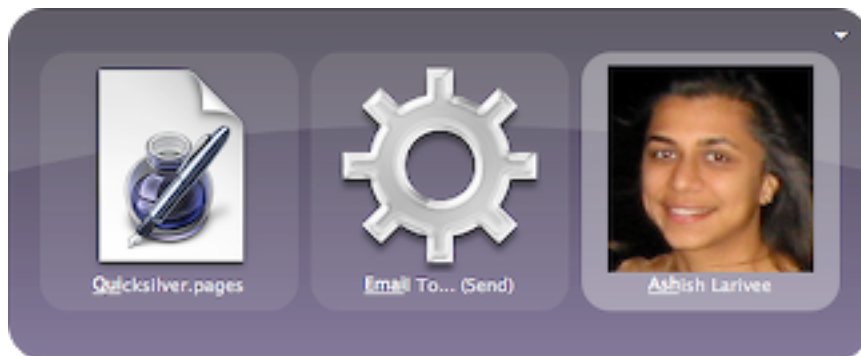


Now the message window is opened with the attachment all set up. Say Ashish was expecting this document and I didn't need to include any text in the message, just the attachment. I could choose the action **Email item...(Send)** and then the message window does not open but the message is sent in the background

without ever disturbing what I was doing before. The subject is set to the name of the attachment and a short sentence says the file is attached to the message.

The above works if I'm thinking, "I want to send Ashish this document". Say instead I thought "I need to send this document to Ashish". I can do this as well. Select the document in the first

pane, choose the **Email to...(Send)** action and then choose Ashish. Say she is online in iChat. There are actions to open a chat window with her or to send the file to her via iChat. I find these options the real strength of Quicksilver. It lets me easily do what I want, how I want. I don't need to change my thinking to how Quicksilver



wants me to do things, and it learns from me and gets easier to use over time. The amazing thing about Quicksilver is how flexible it is. Via a wide variety of Plug-ins, Quicksilver can select just about anything on your Mac as an object and do potentially hundreds of different things to it. Of course, it can also do a lot more, get used to reading that.

Concepts and Terminology

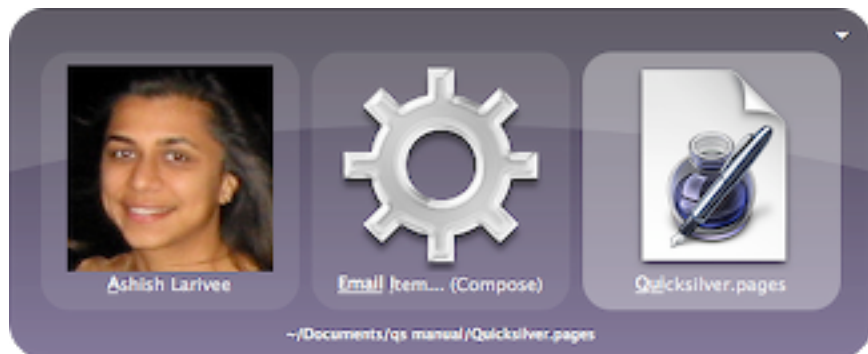
Quicksilver is a very modular application. This manual is organized around data types that Quicksilver can manipulate such as files, text, music, etc. In order to explain various ways to manipulate those data types I have to introduce various Quicksilver concepts and facilities that you'll use in each section.

The key conventions I'll use are to the right. Combinations are achieved by holding down one or more of these and typing another key such as a letter, number or punctuation character. E.g., `⌘;`, `⌘S`, etc. The arrow keys are shown as: `→`, `←`, `↑`, and `↓`.

⌘	Command
⌥	Option
⌘	Control
⇧	Shift

Quicksilver runs as an application in the background at all times. Typically there is no menu, dock icon, menu bar icon or other indication of Quicksilver on screen until you activate it with the hot key (by default ^-space), though there are preferences to change this. Preferences are available by typing ⌘, when Quicksilver is activated.

Quicksilver commands are entered via two or three panes. Unfortunately there is no standard terminology for these. Sometimes they are just called the first, second and third panes. In grammatical terms you can think of them as subject, verb, and object; although direct object, verb, and indirect object is a bit more accurate. I'll try to stick to object, action and argument. Plug-ins are optional modules that are installed which can add objects, actions or other capabilities to Quicksilver. You manage Plug-ins entirely from within Quicksilver Preferences including finding, installing, updating, enabling, configuring and removing them.



Command window showing an object (a contact), an action (Email Item...(Compose)) and an argument (a file).

The Catalog is the list of objects that you can directly bring up in Quicksilver. Objects are not entered individually, instead you configure sources (in the Catalog Preferences) and Quicksilver scans and indexes them at regular intervals (by default every 10 minutes). For example, there is a source for Safari and from it are all your bookmarks and history is indexed. Each of your bookmarks are objects to Quicksilver. So are your files and folders from your home directory, your contacts from Address Book, all the apps from your Applications directory, your albums from iPhoto, the songs from iTunes, and many other things.

There is also a Preference pane showing all the actions that can be performed. All actions work on an object and some actions also require an argument to be entered in the third pane. Those actions that need an argument end in "...". Actions expect certain types of objects and arguments. E.g., the **Open** action works on an object that's a file, a folder or an application. The Mail plug-in adds two more **Open** actions, one that works with a mailbox object and one that works with a mail message object. Bring up any of these objects in the first pane and the **Open** action is available in the second.

Sometimes actions come in pairs. E.g., the **Email To...(Compose)** action uses a file object and an email address argument, so the command looks like: "*file, Email To...(Compose), address*". There is a complementary action called **Email Item...(Compose)** that reverse this. Their command looks like "*address, Email Item... (Compose), file*". When you select an object in the first command pane, the list of available actions narrows to only those that work with that type of object. With some of these pairs of actions the effect is that you can enter the command which ever way is more natural for you. Since **Email To... (Compose)** and **Email Item... (Compose)** are so similar you might not even notice that Quicksilver picks different actions for you. Other pairs are named differently and you might not know that you can enter the command the other way. E.g., a pair of objects is used for searching web sites like google. You can enter a command as "*site, Search For..., query-words*" or "*query-words, Find With..., site*". However there are also some actions that don't have complements and you have to know what type of object

they expect in order to use them. Most well-named actions try to hint at what the argument they take. **Email To....** wants an address to follow. **Email Item...** wants some kind of item to send.

Interfaces are configurable skins that affect the appearance of Quicksilver. They are installed as Plug-ins and can be configured from the Appearance Preference pane. The default interface is called Primer. A very popular interface is called Bezel and it's what is shown throughout this manual. Install it by downloading the Bezel Interface plug-in and then choosing it for the "Command Interface" in the Appearance Preference pane (more details are explained later).

When you use Quicksilver you type to select the objects, actions and arguments of commands. As you type your choices are filtered down from all possible choices to only those that match what you've typed. Typically the interface will show a scrolling results list of the possible choices ordered by the most likely match based on what you typed. The matching method is one of the strengths of Quicksilver. Matches can be by the beginning of the phrase (e.g., type *Deskt* for "Desktop") or by the first letter of each word of the choice (e.g., *BA* for "Browse Artists" which is an iTunes object). The more you use specific items (objects and actions) the more quickly they will be matched. After a few times, once you type *Des* Quicksilver will start guessing "Desktop" and if you do it enough, you might be able to choose it just by typing *D*.

Some items have sub-items, e.g., folders have files, contacts have phone numbers and addresses, musical artists have albums, etc. Usually when an item has sub-items you'll see a > on the right side of the item in the results list. The way you select the sub-items is by typing → or /. The object then changes to the top sub-item and the list shows the sub-item's peers. You can type to match one of them or use the arrows to scroll up and down. This is an easy way to navigate folder hierarchies, or any other hierarchies.

Quicksilver's strength is that it can do so many different things and the matching algorithm and navigation techniques make getting at all those things easy. However, for those commands that you do all the time, there's an even faster way. You can configure triggers to execute specific commands when you type a HotKey or make a mouse gesture (a click, drag, moving to a corner of the screen, and various other things). Have a bookmark or a search you do all the time? Triggers are perfect for them. There are built-in triggers to control iTunes. Triggers can be invoked at anytime, without having to first activate Quicksilver (with ^-space).

There is a type of object known as a proxy object which stands for other objects. E.g., the Current Web Page proxy object stands for the URL of the web page currently shown in Safari (though it should be the default browser). The Current Selection proxy object stands for the current selection in the currently running application. Proxy Objects allow triggers to become dynamic, e.g., pasting the URL of the web page currently shown in the browser.



Installation

Download Quicksilver from the web site <http://quicksilver.blacktree.com/> or directly from <http://getqs.com/>. If you're adventurous you can download pre-release development builds from <http://getqs.com/dev/>. Open the .dmg file and it will mount the Quicksilver volume and automatically extract the Quicksilver.app. Drag it where you want, I put mine in ~/Applications/.

Double click Quicksilver.app to start it. The first time you run it a setup process will run. You can rerun it later with the Run Setup button from the Application Preferences. After a welcome screen and a license screen, Quicksilver will scan your computer for applications and then recommend plug-ins for you to install based on what

applications it finds. Check the plug-ins for all the applications you use regularly and click Continue. Next you can choose the HotKey to activate the Quicksilver command window and how often Quicksilver will scan your computer to keep its catalog updated (the default is every 10 minutes and is a fine choice). This is followed by some basic usage instructions and then the setup closes and a Quicksilver command window (using the primer interface) appears.

In spite of the instructions, first time users might be confused. While Quicksilver is running, there is no menu for it, no Dock icon and no indication it's there aside from the command window. If you close it and didn't read the instructions you might be a little lost.

The Quicksilver.app is usually in /Applications/ or ~/Applications/. When first used, the following per user files and folders are created:

- ~/Library/Application Support/Quicksilver/
 - Actions.plist - list of installed actions
 - Catalog.plist - the configured catalog sources
 - Mnemonics.plist - learned inputs, defaults and abbreviations
 - PlugIns.plist - the list of available plug-ins and how they are configured
 - Triggers.plist - the configured triggers
 - Caches/ - another cache folder



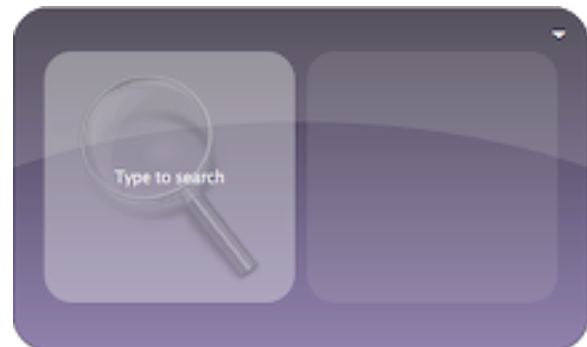
- Indexes/ - folder of indexed items
- PlugIns/ - installed plug-ins
- Shelves/ - where items on the Shelf and clipboards are stored
- Actions - not installed but can be used to add scripts that implement actions
- Templates - not installed but used with **Make New...**
- ~/Library/Preferences/com.blacktree.Quicksilver.plist - stores various preferences and internal state
- ~/Library/Caches/Quicksilver/ - stores various state in binary files

If you want to do a complete uninstall use the Uninstall Quicksilver button in the Application preferences, or remove the above files and folders. It can also be useful to move or rename these while troubleshooting a problem. If you are having trouble installing plug-ins check that the owner of ~/Library/Application Support/Quicksilver/ is the current user and not System. If it's not, change it and restart Quicksilver.

Note that when Quicksilver starts it contacts hrwebservices.net to do a version check. If you notice the outgoing connection it's perfectly normal and benign.

Invoking Quicksilver

Now that Quicksilver is installed and running how do you use it? The basic way is to activate the command window with the HotKey \wedge -space. When you do this the command window appears, the first pane is active and you can type to select some item in your catalog.



The HotKey key is configurable in the Command Preferences Pane under HotKey activation. Quicksilver defaulted to \wp -space for a long time and then in Tiger, Apple chose to use \wp -space for Spotlight. So Quicksilver changed its default to \wedge -space. If you look at older posts on the Web you'll often see \wp -space used to activate Quicksilver. Since I use Quicksilver a lot more often than Spotlight (and anything without \wp conflicts with Emacs), I use \wp -space for Quicksilver and $\wedge\wp$ -space for Spotlight (set in the Spotlight System Preference pane of OS X).

There are other ways to activate Quicksilver. Triggers let you define key sequences, mouse clicks, and mouse gestures that will activate Quicksilver. We'll discuss these in more detail below but it's worth describing two triggers now. If you look in Triggers under Quicksilver you'll find two that have been predefined **Command Window with Selection** and **Command Window in Text Mode**. The first opens a command window with the current application's selection in the first pane. This is typically text, though for the Finder this is one or more selected files or folders. Some other applications like iPhoto and iTunes also allow their non-text selection to be brought into Quicksilver. The second trigger will bring the selection into the first pane but will put that pane into text-mode so you can edit it.

Quicksilver installs two services available in other applications (look in the application's menu under Services and you'll see Quicksilver). The two services are **Get Current Selection (Internal)** and **Send to Quicksilver**. The fact that there are two of them is a quirk of the implementation, you can use either, but **Send to Quicksilver** is the one meant to be used from the menu. It

activates Quicksilver and puts the application's current selection (typically text) into the first pane. The other service is the internal mechanism of how ⌘-esc and the Current Selection proxy object (used in triggers) works. These basically accomplish the same thing as **Send to Quicksilver**. Front Row uses ⌘-esc and this conflicts with the Quicksilver service. You can change the Quicksilver key binding using the Alternate Services Menu Key setting in Quicksilver's Preferences, under Command. Or if you prefer, the Front Row shortcut can be changed in the Keyboard Shortcuts tab of the Keyboard & Mouse system preferences. If you unchecked it completely Front Row still works with the Apple Remote.

Yet another way to put the selection into the object pane is to activate Quicksilver and then type ⌘G . This is slower than the other methods since you have to activate Quicksilver first, but since I learned it first, it's in my fingers, so I use it a lot. By default ⌘G only puts the Finder's selection into the first pane. If you have advanced features enabled in the Application Preferences, then in the Extras Preferences checking "Pull selection from front application instead of Finder" will make ⌘G behave just like ⌘-esc , using the same mechanism.

If Quicksilver is already running, you can also click on the Quicksilver.app icon to bring up the command window. If you drag and drop something onto the icon it will be selected in the first pane. This is convenient if Quicksilver is in your Dock, but people also put it in the Finder's sidebar or toolbar to have it always handy.

Selecting Items

Once you activate Quicksilver with ⌘-space you type some characters of what you're looking for and it will appear in the first pane. If I'm looking for the Address Book entry for my friend Ashish Larivee typing any of the following might select it: `ashish`, `ash`, `al`, `alari`, or many other combinations of the letters. In fact Quicksilver keeps track of what you type and what you select and uses that past history to guess what you're looking for now. This means if I type `a` and pick Ashish's entry Quicksilver will start to guess that more often for `a` rather than "Address Book" or "Adium" or other things that begin with "a". Quicksilver uses what's known as a matching algorithm to order its choices, below it's discussed in some detail with how you can help it.

if you mistype something, you can type the `delete` key or ⌘X to clear the pane and start over. If you navigate to something via → or / then these keys will still keep you in the current position but clear the search. To reset entirely so that you're using the top-level global catalog use the `esc` key. There's no way to delete just the last letter typed.

In the Command preferences you can configure the "Reset search after" time. This is how long between keystrokes Quicksilver waits before resetting the search. So if it's set to .85 seconds and it's been 1 second since you've typed a key, if you type another one it will start the search over with that key.

By default Quicksilver will show its other guesses for what you typed after a few seconds, you can then scroll through them by typing ↓ . This behavior can be changed in the Command Preferences under "Show other matches" where you can choose: Never, Immediately, or Delayed (the default). You can also configure the spacebar to behave like ↓ .

You might not have realized this but Quicksilver shows what matches what you typed in two different ways. When first activated, Quicksilver will filter or narrow the whole catalog down to just the things that match what you typed. To be clear, the things that don't match what you typed, are removed from the results list. However if at some point you type → or / to "go into" an object, if you type you'll find that Quicksilver moves to the first item that matches alphabetically

what you typed. The non-matching items are not removed. This is most apparent when browsing the file system or your iTunes library. It's also evident when browsing a contact's information, if you type to find their work phone number, the other information is still displayed in the results list. This behavior is known as selection as opposed to narrowing.

There are other, less obvious ways to select something in a Quicksilver command window pane. As discussed above you can type ⌘G to choose the Finder's selection (or possibly the selection of the frontmost application). You can also paste something into the pane with the standard key binding ⌘V and you can drag and drop something into a pane as well (you need to start the drag before invoking Quicksilver).

You can select the last object used by typing ⌘[after activating Quicksilver. In fact it recalls the entire last command, that is the object, action and argument used. Unfortunately as of B50 you can only select the last command and not earlier ones by typing ⌘[again.

The Comma Trick

A less obvious feature is known as the comma trick. Using the comma key, you can select multiple items in the object pane and then have a single action operate on all of them at the same time. Using this you can move several files to the trash at once or address a new mail message to several different people. Activate Quicksilver, select an item, type , (comma) and then select another item. You'll see a collection of smaller icons accumulate at the bottom or left of the object pane depending on the interface used (the menu interface does not support the comma trick). You can select as many items as you want. When you have them all, tab to the action pane and continue as normal. If you have several collected you can type the delete key to remove the last item collected (i.e., the last small icon).

Another way to select multiple items is with ⌘A. It will select all the items in the results list as if you had used the comma trick for all of them. If your results list is long this can take a long time. It's most useful if you navigate to a folder with a small number of items that you want to operate on (e.g., deleting, moving, or tagging them all).

Helping the Matching Algorithm

Quicksilver learns what you do (and makes that easier to do) because of its matching algorithm. It remembers what commands you execute and what you did to call them up, do them a few times and they get easier to call up. However if you do things that are similarly named it confuses Quicksilver and it won't guess as quickly as it could. Fortunately you can teach it. Triggers, described above, seem to be the fastest way to do things in Quicksilver, but there are only so many keystrokes you can remember. Here are some techniques you can use.

If you activate Quicksilver and start typing into the object pane you'll see a results list appear of items that match what you've typed. The order of the items is known as their rank. The first item is ranked 1 the second is ranked 2, etc. For objects, the rank is determined based on the score of the items. Score is computed based on how well an item matches what you typed.

Theoretically, items move to second place rank the first time they are used to match some input. The second time they are used they get ranked first. You can manually adjust the score of an item by ^-clicking on it to bring up a menu of choices. For some items (TODO which ones?) you can choose Decrease Score to lower it in the list. For any item you can choose Set as Default for *string-you-typed*. Then when you type that same string again, that item will be the top choice. The circle to the left of the item in the results list indicates any manual settings you've done. TODO: How specifically?

Setting a default works for what you've typed. But say you want to set as a default something that is mid-way in a very long list that you don't want to scroll through. As an example say you want z to bring up your Amazon bookmark and that if you type z, Amazon is further down the list than you want to look. In that case, use the **Assign Abbreviation** action (you need to have advanced features enabled). Bring up the object you want in the first pane (www.amazon.com), choose **Assign Abbreviation** as the action, and in the third pane enter the abbreviation you want via text mode (z). Now if you type that abbreviation in either the first or third pane, your choice will be ranked first. This happens because exactly matching an abbreviation causes the item to have a very high score. Note abbreviations will only work if they match the input string in some way.

You can't set `irc` to be an abbreviation for Colloquy.app because if you typed `irc`, Colloquy would not appear in the results list. Say you don't use Activity Monitor often and when you want it you can never remember its name but you think of the word "processes". Quicksilver doesn't help here since it doesn't relate "processes" to Activity Monitor, and as a result Abbreviations don't work for this. But you can use Finder aliases. Find the Activity Monitor.app and create an alias for it called Processes.app. Put the alias someplace that Quicksilver will index it. Now if you type `processes` quicksilver will suggest your new alias.

The above is all for objects in the first and third panes. Actions in the second pane work a little differently. Actions have no score, their rank is statically determined by their order in the Actions Preferences. You can manually change the order by dragging and dropping them (for this to work you must sort by the Rank column).



The user ^-clicked on the 4th item in the results list after typing `Des`. The pop-up menu shows the item's score, 95 and the option to set it as the default for `Des`.

TextMate Ranker

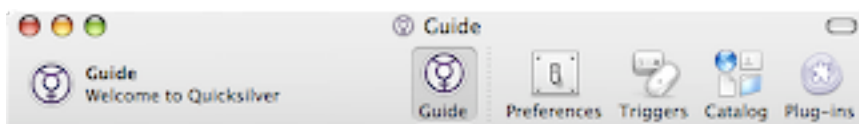
You can also choose a slightly different matching algorithm by installing the TextMate Ranker (+) plug-in (which requires advanced features to be enabled) and selecting it as the String Ranker handler in Preferences.

It makes two changes to the algorithm. The first is that it favors more beginning of word letters that are touched. E.g., if you type `AM` it will match Activity Monitor over Adium since two beginning of word letters are matched instead of just one. The second difference is that it favors input that matches a higher percentage of the words. So the input `AM` will favor Activity Monitor over Audio MIDI Setup since 2/2 (100%) of words are matched instead of 2/3 (67%).

As a result, if you're the type to favor typing acronyms over continuous letters in the name of something, you'll probably prefer the TextMate Ranker. However there are reported problems with matching things in the third pane such as when you use the **Open With...** action or the **Email to...** action. You may find that no results match in the third pane.

Configuration

Quicksilver has a configuration window that gives you access to the Guide, Preferences, Triggers, Catalog and Plug-In. You can access each of these



directly from a menu or via these shortcuts after activating Quicksilver (by default with `^`-space):

- `⌘?` Open Guide
- `⌘,` Open Preferences
- `⌘;` Open Catalog
- `⌘'` Open Triggers
- `⌘"` Open Plug-ins

We'll describe these now in the following sub-sections: Plug-Ins, Catalog, Interfaces, Handlers, and Notifications. Triggers are a large topic and are covered in their own entire section immediately after this one.

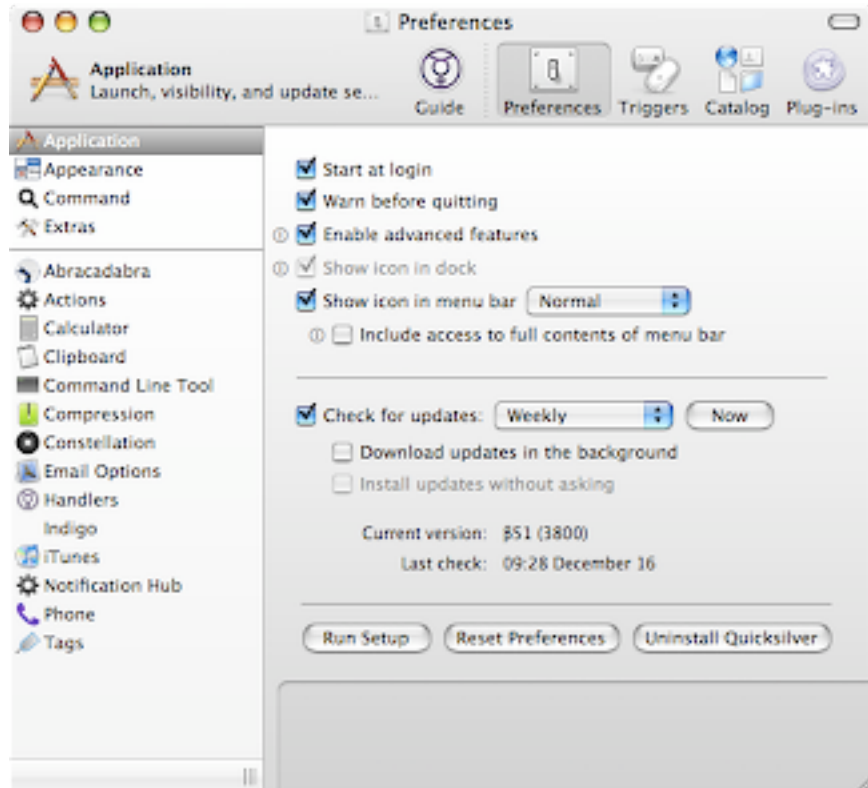
Preferences

Quicksilver's Preferences are divided into sets at the left of the window. Those listed at the top are built-in to Quicksilver and those in the bottom section are enabled by installed plug-ins.

The Application Preferences control the most fundamental aspects of Quicksilver as an application. “Start at login” and “Warn before quitting” are recommended. The next two preferences require a restart before they take effect, this is what the small circle with a line icon indicates. The easy way to restart is to activate Quicksilver (^-space) and then type ^⌘Q

There are many features described in this manual that say you must enable advanced features. To do so check the third preference shown. After restart you'll see the “Extras” Preferences set shown in the image and described below. I recommend you enable Advanced features

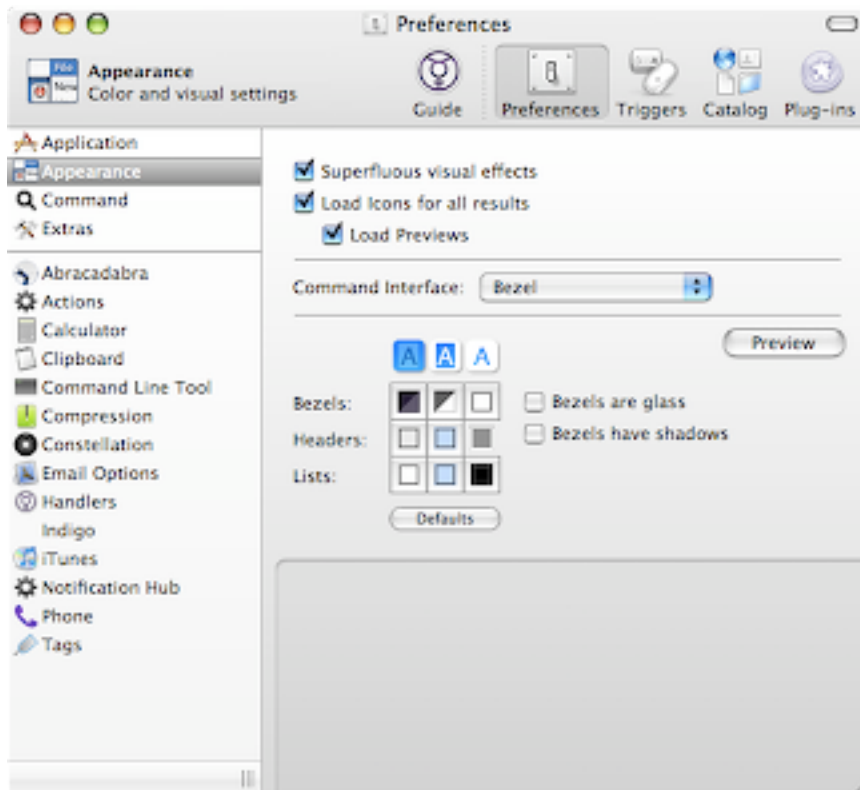
as a lot of useful Plug-ins require it (those are indicated with a + symbol in their name). These aren't necessarily unstable (as those Plug-ins marked β usually are).



The “Show icon in dock” preference will let you not waste Dock space with Quicksilver. It is only available (not greyed out) if your user account has admin privileges. If you do so the main Quicksilver menu will not appear while the preference window is open. I tend to keep it in the Dock so that it's easy to restart if Quicksilver crashes. The “Show icon in menu bar” preference puts an indicator in the top right part of the screen, some people like this, particularly if they don't configure a Dock icon or have their Dock hidden. It provides a simplified menu to access the various configuration panes of Quicksilver. If you want it to provide the full menu options (via more sub-menus) set the option “Include access to full contents of menu bar”. The popup allows you to configure the menubar icon on the left, right side of the menubar or “Normal” lets you configure it as any other menubar item,

The bottom half allows you to configure if Quicksilver checks its website automatically for updates. You can specify if it should automatically download updates and whether or not they should be installed automatically. Some betas have had some problems if installed this way. Usually a manual download of a new version from the Quicksilver website will solve any problems.

The three buttons at the bottom will rerun the installation setup described in the [Installation](#) section, Reset Quicksilver's preferences, and completely Uninstall Quicksilver.



The Appearance Preferences control how Quicksilver looks. The first three options all have the trade off of being a little prettier at the cost of some performance. If your machine can handle it enable them, if you're finding Quicksilver a bit sluggish try disabling them.

The Superfluous visual effects vary per interface but include the command window unzipping open and fading closed.

Normally Quicksilver doesn't load all the icons of items in the results sets, the second option changes this behavior. The "Load Previews" option is similar to the "Show icon preview" in the Finder's View Options. It will display a

thumbnail image of a file or other item in a pane in the command window.

The Command Interface preference is how you choose the interface Quicksilver uses for the main Command window (aka skin or theme). The popup is filled with choices based on the Interface plug-ins installed. See the Interface section below for details. The default interface is Primer and it shows a little more context information than other interfaces so it's recommended for new users. Shown here, the popular Bezel interface is configured which is used throughout this manual. The various controls below the interface selection allow you to further customize the appearance of the interface and vary per interface. The column headers have tooltips showing they represent: Background, Selection & Accents, and Text colors. Clicking on the nine colors in the grid will bring up a color picker.

The Command Preferences allows configuration of the command window's behavior.

The top of the pane affects how you activate Quicksilver. The HotKey Activation is the basic way to activate Quicksilver. The default is to type \wedge -space, though as shown a common alternative is to use \mathring{S} -space and to remap Spotlight to something else via the Spotlight System Preferences.

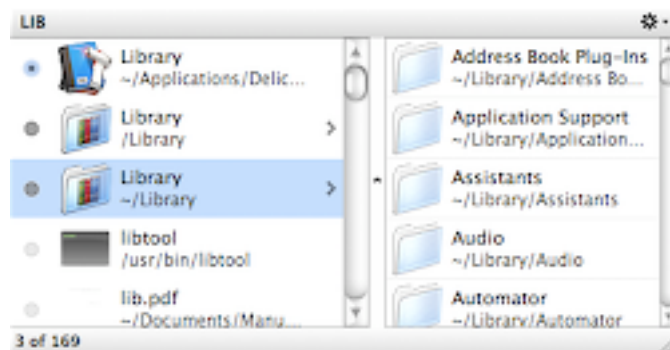
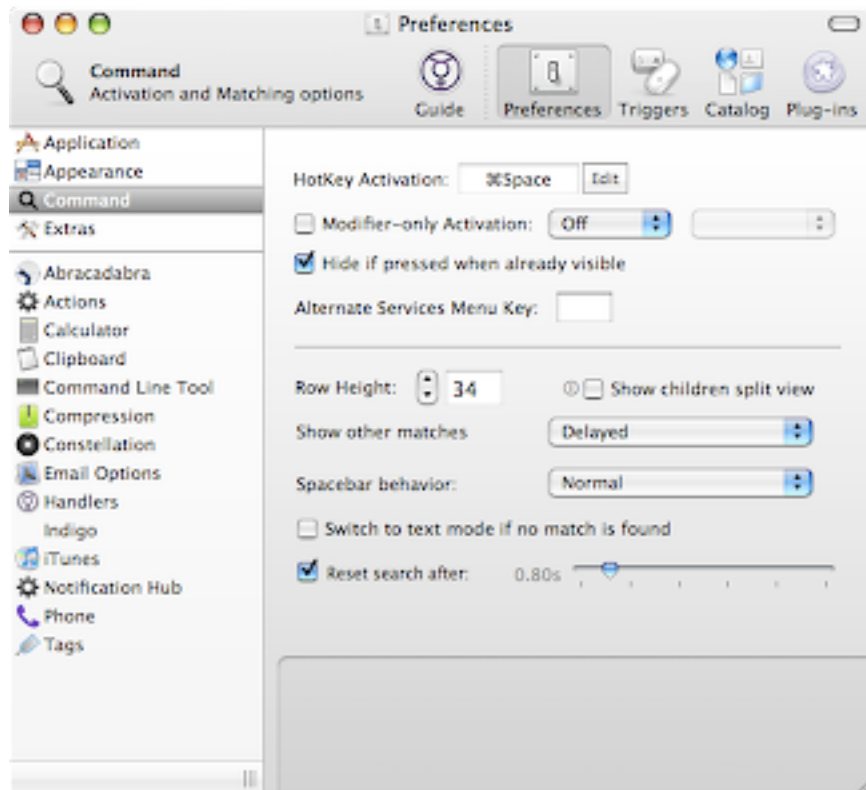
“Modifier-only Activation” allows you to activate Quicksilver by single or double presses of modifier keys such as \mathring{S} , \wedge , \uparrow or even $\pounds n$ on a laptop.

“Hide if pressed when already visible” affects what happens if you activate Quicksilver when it's already active. Regardless of the setting of this option, typing `esc` will close the command window.

The “Alternative Services Menu Key” is how to change the shortcut assigned to the Quicksilver service Get Current Selection. The default value is \mathring{S} -`esc`, so you might want to change it if you use Front Row as described above in [Invoking Quicksilver](#).

The bottom of the pane affects how the results list appear. You can specify the Row Height in pixels. Selecting “Show children split view” will split the results list vertically into two columns, the right one showing the contents of the selected item in the left one (much like the column view of the Finder). You must restart Quicksilver for changes to this setting to take effect. When first enabled you'll only see one column, but notice the small dot in the right edge, you can click on and drag that to the left to reveal the second column.

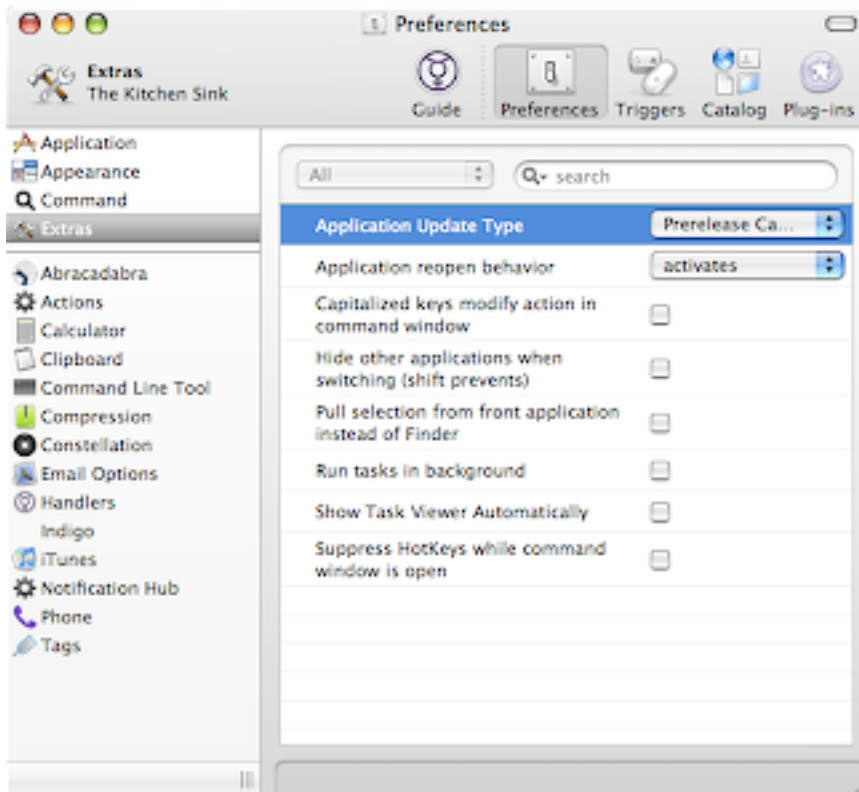
“Show other matches” controls if the results list appears after you type into the command window immediately, after a short delay or not until you type \downarrow . By virtue of Quicksilver's matching algorithm, the spacebar isn't particularly useful in the command window. The “Spacebar behav-



ior” preference lets you reclaim that key to make Quicksilver a little faster to use. You can configure it to:

- Select Next Result - select the next result in the results list, like typing ↓
- Jump to Argument List - change to the 3rd pane
- Switch to text mode - change the current pane to text mode entry, like typing . or ‘
- Show Item's Contents - goes into the item, like typing → or /

“Switch to text mode if no match is found” will allow you to save typing . or ‘ to enter text mode in a pane. “Reset search after” lets you configure how much of a pause in typing into the command will start a new search as opposed to appending to the existing search.



The Extras Preferences are only available if Advanced features are enabled in the Application Preferences.

“Application Update Type” allows you to choose between Prerelease Candidates and Final Releases when Quicksilver checks for updates as configured in the Application Preferences. Sometimes after installation this is set to a blank value and that has been observed to cause problems, make sure it's set to some value in the popup.

“Application reopen behavior” configures what happens if you use the **Open** action on an application that is already running. You can have it activate

the application, show the front window or show all windows.

Since the matching algorithm is case-insensitive the shift key is perhaps something we can put to better use in Quicksilver. If you check “Capitalize Key modifies action in command window” then shifted letters are used to select the action, eliminating the need to tab to the second pane. Once moved to the action pane, unshifted letters don't change the first pane, all typing counts for the second pane. If you also hold down the ⌘ key the action will be performed immediately, no need to type `return`. E.g., shift-i selects the action for i (perhaps **IM**) while `cmd-shift-i` performs the action for i. Use caution as it's not always clear which action is invoked.

If you prefer concentrating on one thing at a time, check “Hide other applications when switching” so that Quicksilver will do a “Hide Others” after switching applications. You can prevent the hiding behavior by holding down the shift key when completing the command. There have been problems reported with this feature that applications no longer appear when doing `⌘-tab`.

Normally ⌘G will take the Finder's selection and put it into Quicksilver. By checking the "Pull selection from front application instead of Finder" option, it will use the same mechanism as ⌘-esc and will pull the selection from whatever the active application is. See more details at [Invoking Quicksilver](#).

Checking "Run tasks in background" will do just that. E.g, if you use the **Compress** action it does the compressing in the background allowing you to do other things with Quicksilver. However some have reported some instability from using this option that manifests in odd ways. E.g., some have reported that running the same Compress action via a trigger with this option checked causes Quicksilver to hang during the compression.

Quicksilver has a small window that shows current status known as the Task Viewer. You can bring it up by activating Quicksilver (^-space) and typing ⌘K. Most people don't use it. You can also check the "Show Task Viewer Automatically" option to have it appear when Quicksilver is performing tasks such as refreshing the catalog. There's an image of it in the [Catalog](#) section below.

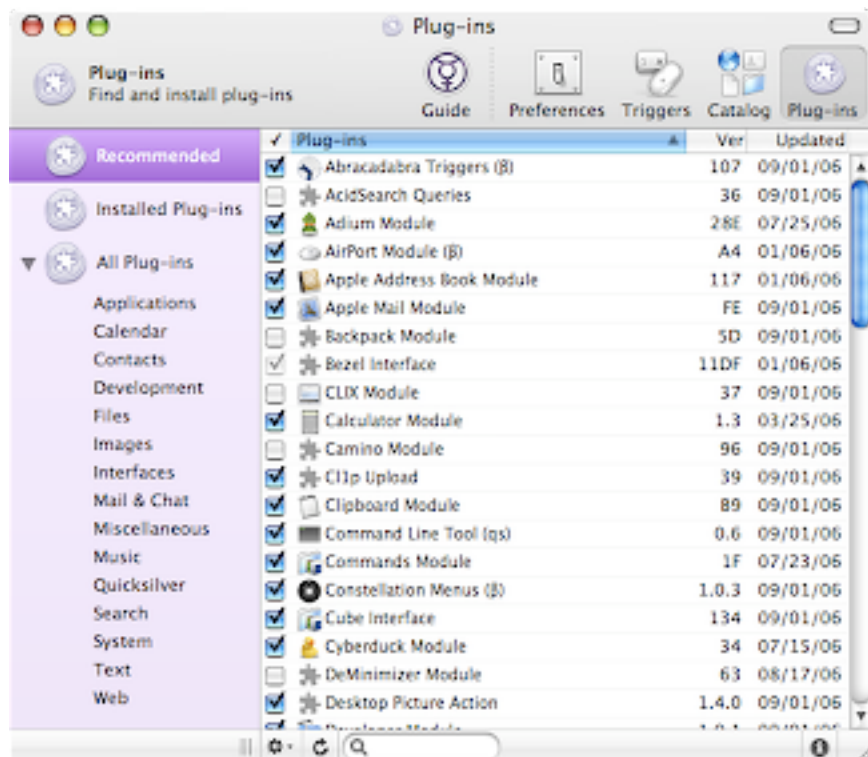
If you enable "Suppress HotKeys while command window is open" then HotKey triggers will not work while Quicksilver is activated (i.e., the command window is open). I have no idea why you would want to enable this.

Plug-ins

Quicksilver is designed with a central core that implements some basic functionality but most of the features are implemented in plug-ins. You can pick and choose which plug-in functionality you want, but must install the plug-ins before that functionality is available. Managing the plug-ins including finding, installing, enabling, and removing is done entirely from within Quicksilver in the Plug-ins panel (though additional configuration in Preferences or the Catalog may be useful or necessary). You can bring it up from the menu or by activating Quicksilver and typing ⌘".



The plug-ins are shown in the right pane, with a checkbox showing if they are enabled, a column showing the version number (which follow no consistent pattern) and the date the plug-in was last updated. The left panel shows sets of plug-ins:

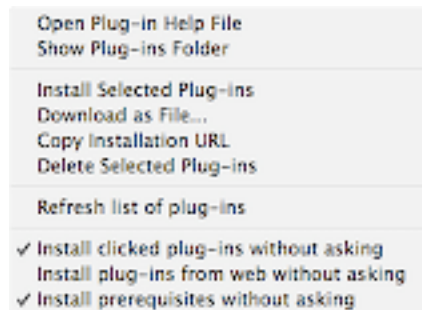
- **Recommended** - recommended for you based on a scan of what applications are installed in the machine.
- **Installed Plug-ins** - those plug-ins that are installed in ~/Library/



Application Support/Quicksilver/PlugIns/. Only those that are checked are enabled, those not checked are installed, but disabled.

- **All Plug-ins** - all available plug-ins from the quicksilver.blacktree.com server. Checked plug-ins are installed and enabled, unchecked plug-ins are not installed. \backslash -click on this to show some hidden built-in plug-ins like Core Support and E-mail Support.
- **Categories** - shows plug-ins collected into relevant categories by feature. Plug-ins can be in more than one category. Otherwise behaves like All Plug-ins, checked plug-ins are installed and enabled, unchecked plug-ins are not installed.

Checking a plug-in will enable it, downloading and installing it if necessary. Selecting a plug-in and clicking the  button will open a drawer with (usually limited) information about what the plug-in does. Selecting one or more plug-ins and clicking the \pm button opens a pop-up menu of things you can do to plug-ins including install, download, copy, and delete. Delete is the only one you'll typically use from this menu. Note that if a plug-in is loaded and you delete it, a restart will be required to unload it from memory. The last three items in the menu are options you can enable or not. The  button will refresh the list of plug-ins from quicksilver.blacktree.com.



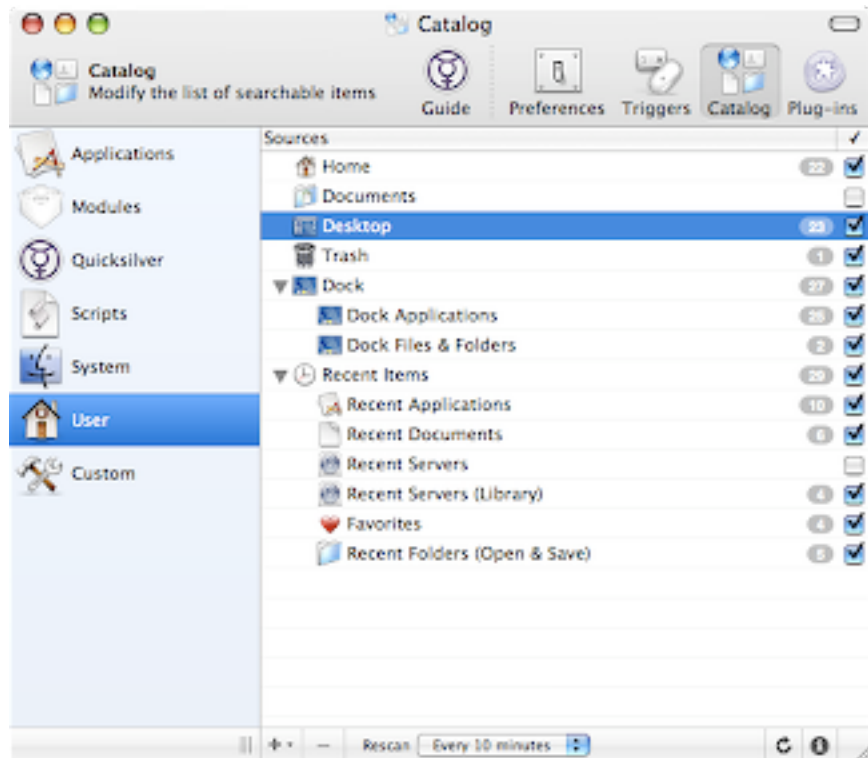
If you expect to see a plug-in in the list and don't, try refreshing the list of plug-ins. Note that some plug-ins are only visible if Advanced features are enabled. Usually these plug-ins have names that end in (+). Plug-ins that end in (β) should be considered to be of beta quality. If you're having problems installing plug-ins check the ownership and permissions on ~/Library/Application Support/Quicksilver/PlugIns/ and its parent directory. It should be owned by you and you should have permission to read and write it. Usually quitting Quicksilver and removing (or moving) the PlugIns folder or its parent Quicksilver folder and restarting Quicksilver (allowing it to recreate the folder) will solve any problems. Of course removing the Quicksilver folder will remove any customizations you've made.

Some people have reported that using the Little Snitch network monitor interferes with Quicksilver's ability to download plug-ins. If your plug-in list is empty and you run Little Snitch, try disabling it.

Catalog

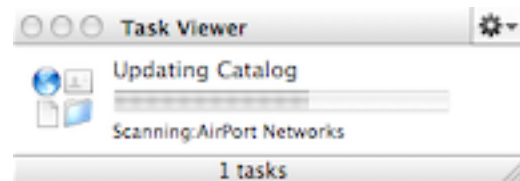
The Catalog is the collection of items indexed by Quicksilver. You populate it by configuring catalog sources which Quicksilver periodically indexes. This is done in the Catalog panel. You can bring it up from the menu or by activating Quicksilver and typing \mathfrak{C} :

Catalog sources are configured into sets shown in the left panel. Most are pre-configured by various plug-ins and many plug-ins add sources to the Modules set. The checkbox enables or disables the source. Enabled sources that have been indexed show the number of items found. Note that some sources (such as Proxy Objects in the Quicksilver set) require advanced features to be enabled in the Application Preferences and not all sources are enabled by default after plug-ins are installed.

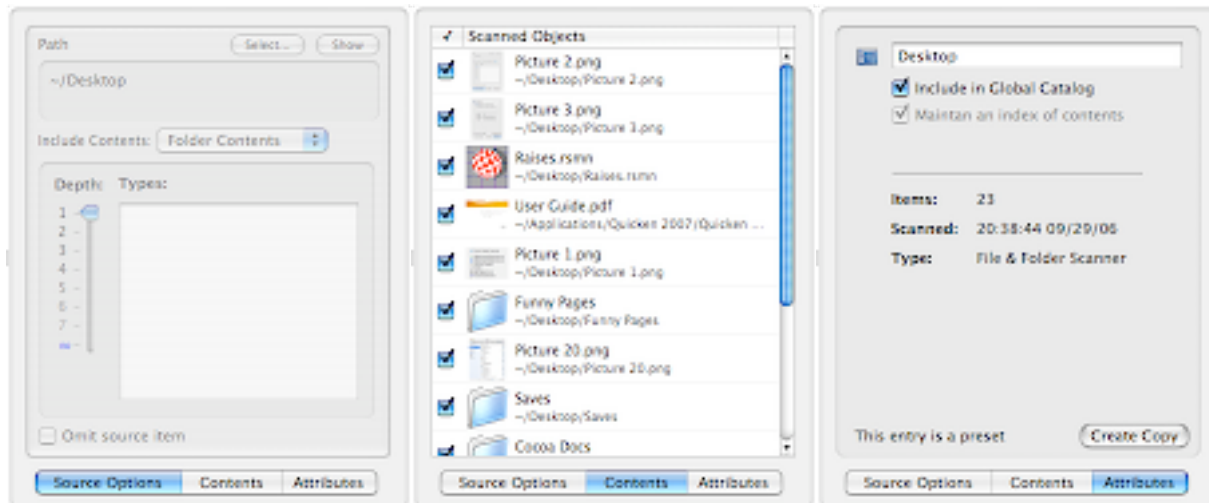


At the bottom of the window you can configure how frequently Quicksilver rescans the catalog sources to find (and remove) items. The default is 10 minutes. The \mathfrak{C} button will manually rescan a selected catalog source. You can manually rescan the whole catalog with the Rescan Catalog command in the Quicksilver menu (only if you have the Dock icon visible) or more conveniently by typing \mathfrak{R} after activating Quicksilver. Some interfaces (e.g., Bezel) will show a progress icon while rescanning the catalog.

You can also configure the Task Viewer to show automatically during a catalog rescan. Show the Task Viewer by selecting it from the Window menu (if the Quicksilver Dock icon is enabled) or by typing \mathfrak{K} after activating Quicksilver. The gear menu in the top right (which is there even if it's not visible in some interfaces) will let you configure two options: Show Automatically and Resize Automatically. The Task Viewer isn't all that useful as things usually just work. It could be used to notice when Quicksilver is doing rescans or perhaps to help troubleshoot a slow catalog source (if you've configured a file scan to be too deep).



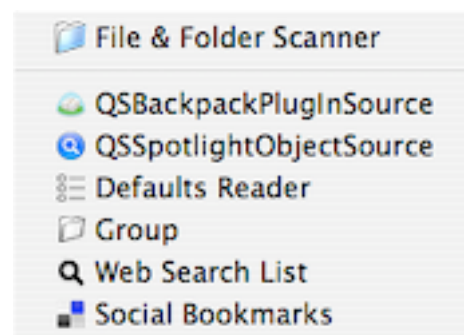
With a source selected click on the \mathfrak{C} button to show a drawer with three tabs. Shown below are the tabs for the Desktop source selected in the above image.:



- Source Options - varies based on the kind of source. A file source (shown) allows you to configure depth of scan and file types to be included, see below for details. This is shown greyed out since the Desktop source is not configurable.
- Contents - the list of all items found by indexing this source. You can remove a specific item by unchecking it
- Attributes - provides some info about the source and allows you to enable it and change the name. Include in Global Catalog is whether the source is enabled or not and is the same as the checkbox in the main Catalog panel. Predefined sources have a Create Copy button that will duplicate the source in the Custom catalog set, allowing you to change the source options.

If you want to know what catalog source an object comes from use the **Show Source in Catalog** action. Select any object in the first pane use this action and the Catalog window will open with the source of the object selected. If you find items in your catalog you don't want, this is a way to track down what sources you want to modify or remove.

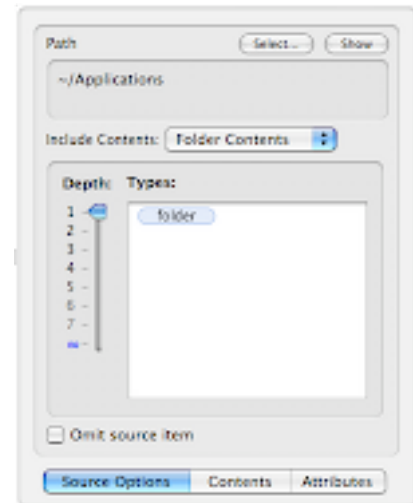
Using the + and - buttons in the Custom set you can add and remove additional catalog sources. You can also create custom sources via the Create Copy button in the Attributes tab of pre-defined sources. The + button will show a drop down menu of various source types (aka scanners) which varies based on the plug-ins installed. Shown at the right are all available scanners. Note that some plug-ins install catalog sources in the Modules set and others install new scanner types. When trying to determine what a new plug-in does, remember to check both places (or read this manual).



The File & Folder Scanner lets you add folders to scan into the catalog. You can also just drag folders into the main catalog pane to add a source. A common thing to want to do is to scan the ~/Documents/ folder deeper than the depth of 2 the default source (in the User set) uses. To do this, select the Users set and the Documents source, open the drawer and select the Attributes tab. Click on Create Copy to create a new source in Custom named Documents. The difference is that the Source Options tab of it is editable. Change the depth slider to what you want, I have mine at 3. Many people just select infinity but that's probably not what you want. Your indexes will take longer and your catalog will be huge which will slow Quicksilver down. Also if you have many extra items in your catalog your

searches are more likely to have extraneous results. Remember that you can always navigate to any file in Quicksilver (see the [Files and Folders](#) section) so all you need in your catalog are your more commonly used files. Also you probably don't want to just index your entire home directory. ~/Library has many preference and cache files that you don't want indexed and the Music and Pictures folders are better served by the iTunes and iPhoto plug-ins and their catalog sources. The File & Folder Scanner can also scan the contents of files to add to the catalog. The scanner can recognize either HTML links or text lines depending on the setting of the Include Contents popup.

The Applications set has 4 sources defined. The Applications (User) source indexes ~/Applications/ to a depth of 3 but also only finds the applications, not the intervening folders. I want those in the catalog so that I can easily move an application into them (e.g., ~/Applications/Games/ and ~/Application/Browsers/). The Types field in the Source Options tab is what's useful for this. If you type file extensions (e.g., .txt, .c, .doc, etc.) into it you can filter the kinds of files indexed. You can also enter Mac Type Codes surrounded by single quotes and type `tab` to have it interpreted. So, to get applications entered as it is in the pre-configured source, enter 'APPL' (including the single quotes). In this example I want folders, so I enter 'fold' and type `tab` and it's replaced with "folder". I choose a depth of 1 to include my Games and Browsers folders but not folders that some applications come with (or the Contents folders inside the .app packages).



Note that the Find All Applications source under the Applications set will search the whole system for application packages. It also indexes any external drives connected. Some people complain that when connecting drives to their machine the drives spin up. To prevent Quicksilver from causing this, uncheck the Find All Applications catalog source.

The QSSpotlightObjectSource allows you to specify a Spotlight query to find items to add to the catalog. See Apple's documentation for Spotlights [Query Expression Syntax](#) and [Common Metadata Attribute Keys](#). Unfortunately in B51 this is buggy. It's difficult to get one of these sources to scan and it sometimes goes into a loop continuously rescanning and using lots of CPU.

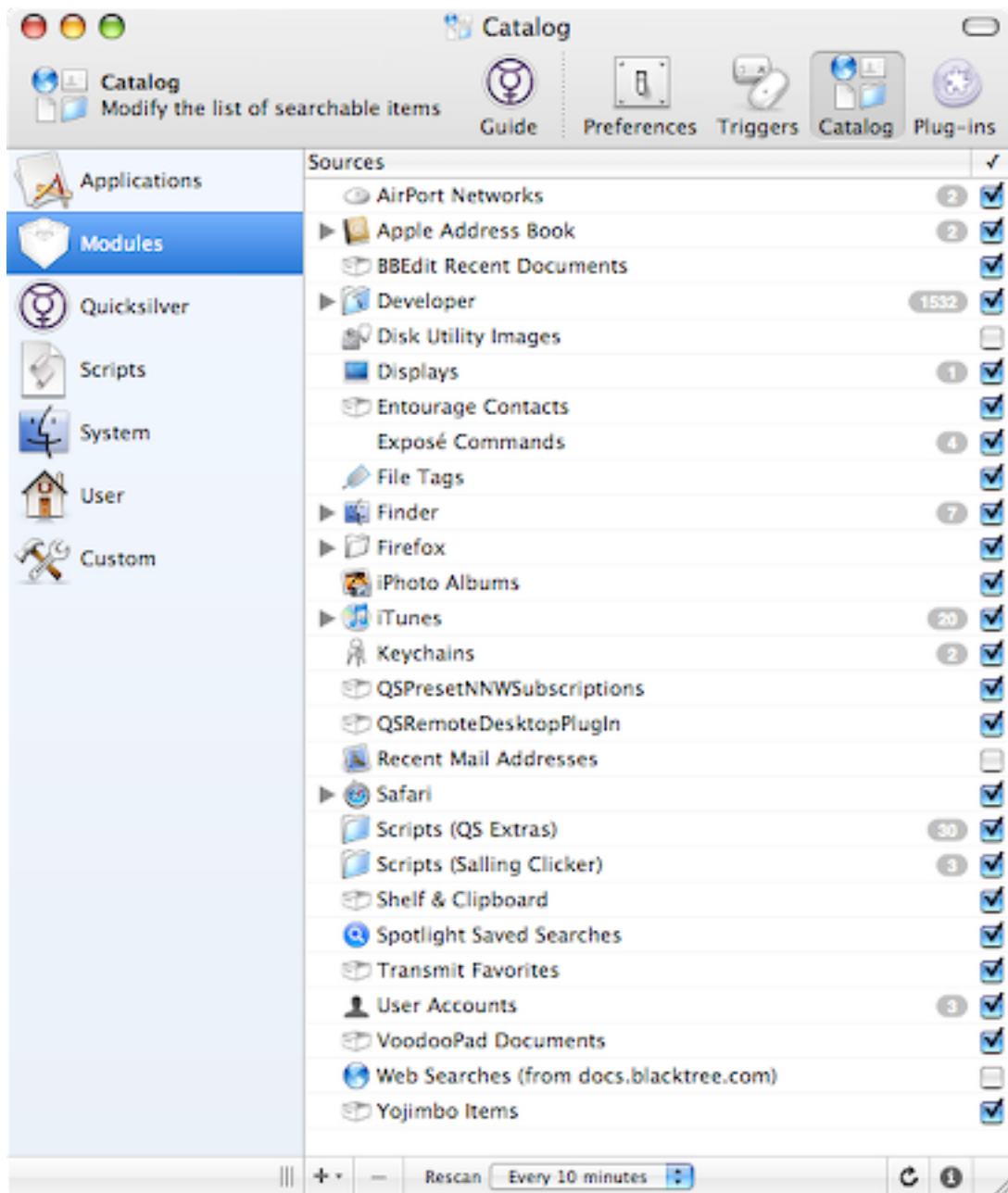
The Defaults Reader allows you to index some keys from preferences files (.plist), though it's a little flakey (doesn't deal well with paths), is difficult to configure (you must manually specify each key) and doesn't let you change the plist file, so it doesn't seem useful.

The Group type is just a a folder for custom catalog sources to be able to group many custom sources in the catalog preferences to make them easier to read.

QSBackpackPluginSource, Web Search List, and Social Bookmarks are described in other places in this manual.

If Advanced features are enabled, in the Quicksilver set there is a source called Quicksilver Catalog Entries. If enabled, an item is added to the catalog for each source configured. These items have names that end with "(Catalog)". You can select one of these items in the first pane and then type `→` to navigate through just that source. If you want to do this often, create a trigger. E.g., the trigger Applications (Catalog) (**Show Contents**) lets you search through just the applications in your catalog (well those that are found via this catalog source). While you can

often do this just by typing → into an object, if you have custom sources configured (say for files of a particular project) this can be very handy.

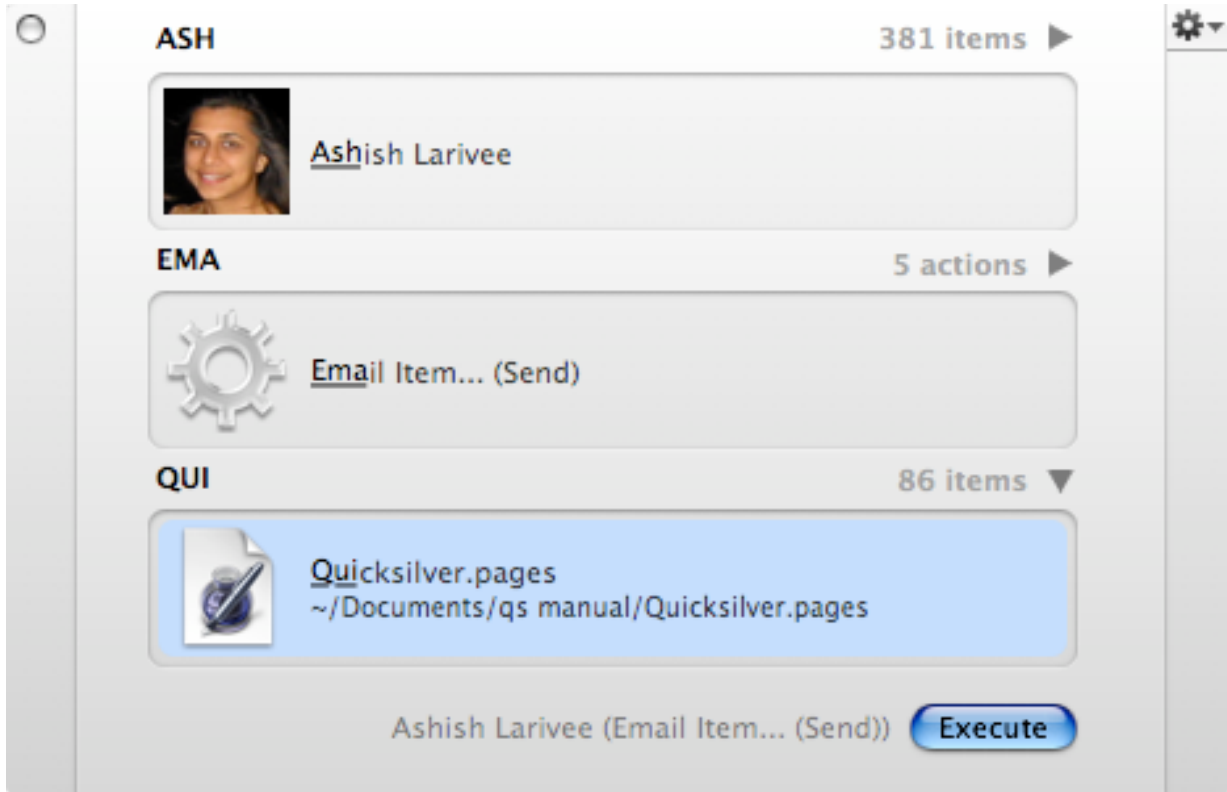


Above is a screenshot showing all the catalog sources installed under Modules with all plug-ins installed in B51.

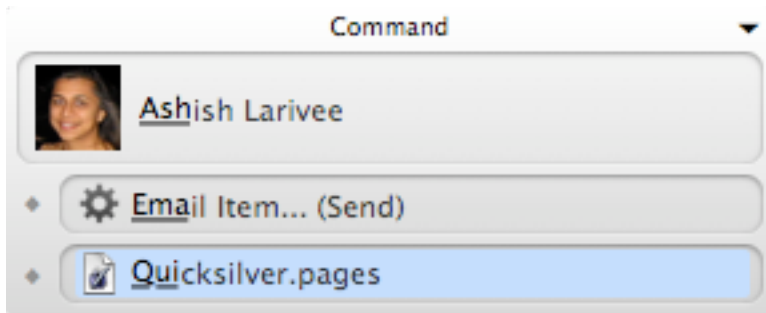
Interfaces

Interface plug-ins change the look of Quicksilver's command window. The following screenshots show each of the six interfaces at full size so you can compare. You choose the interface in the Appearance Preference pane from a pop-up menu showing all the interface plug-ins you've in-

stalled. If you VNC to different machines, you might want to configure them with different interfaces to distinguish them.



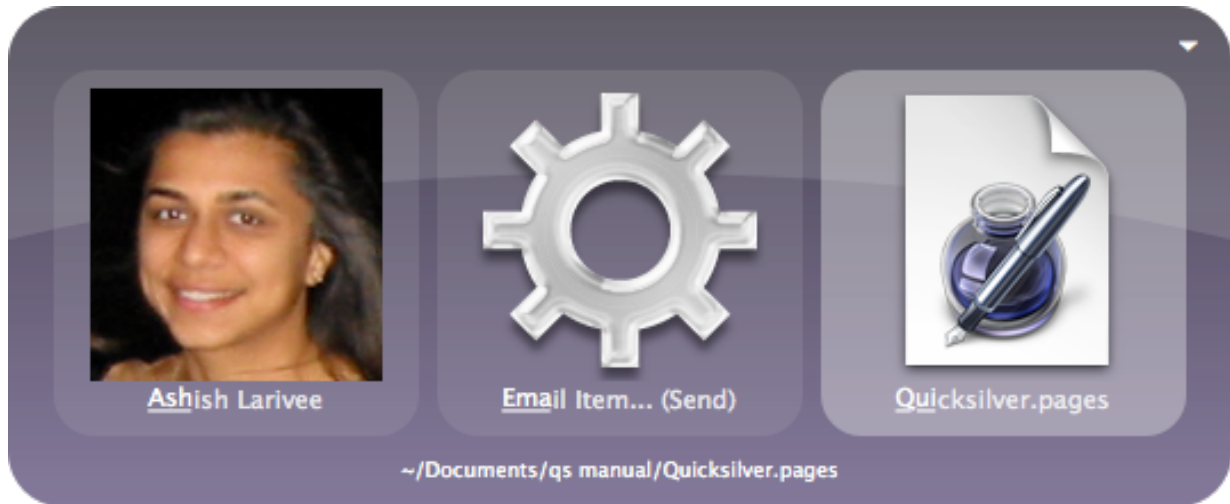
- **Primer** - The default interface is large and provides more info to help you learn



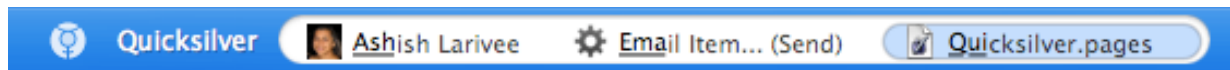
- **Window** - Like Primer but smaller and with less info



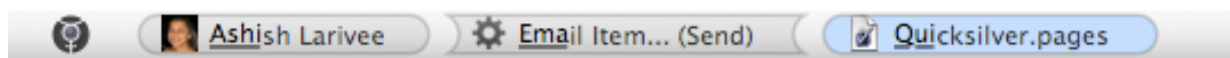
- **Mini** - Like Primer and Window but even smaller



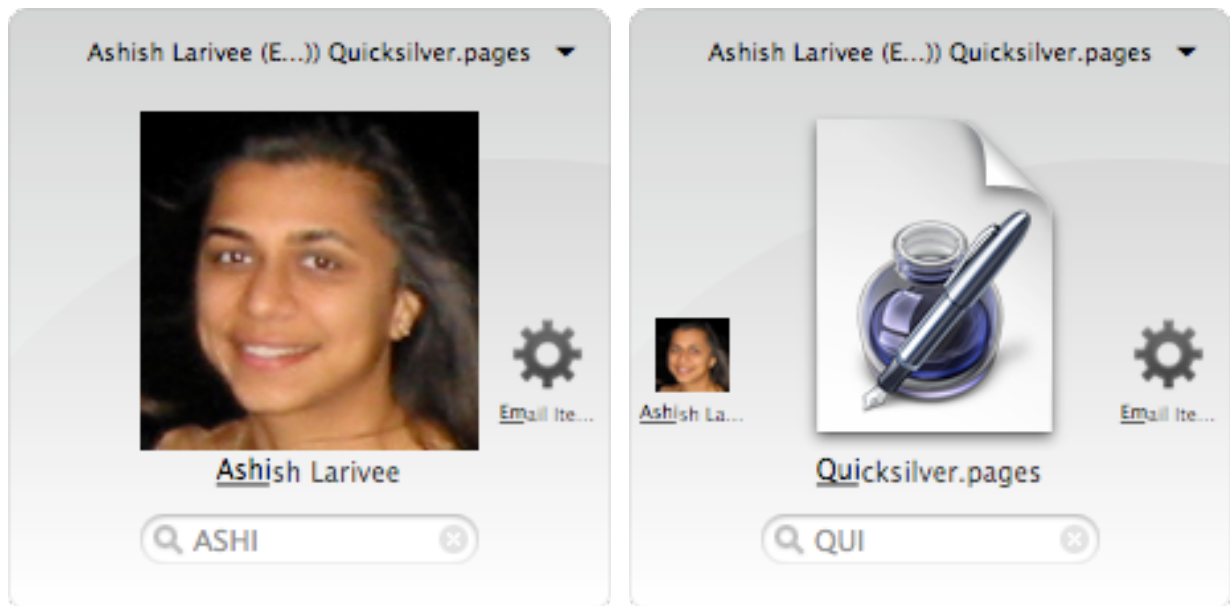
- **Bezel** - Grey, horizontal panes. Very popular (used in this manual at reduced size)



- **Flashlight** - Looks like Spotlight and appears in top left corner of the screen



- **Menu** - Fills the OS X menu bar with input panes, doesn't support [the comma trick](#).



- **Cube** - A single square pane that rotates as you tab between the panes. Shown above are the first and third sides of the cube on this one command. Note that the order of the command parts (object, action, argument) changes on each side. It's very slick seeing it rotate.

Handlers

Quicksilver not only makes it easier to work with a variety of applications, it sometimes uses other applications to perform commands you tell it to. These are of course configurable and you find them in the Handlers Preferences pane. You can configure a handler for Mail actions, IM actions (iChat or Adium), Terminal actions (Terminal or iTerm), String Ranker (Default or Text-Mate), File System Browser (Finder or Path Finder), Missing Object Selector (Quicksilver or Constellation Menus), and Notifications (Quicksilver, Console or Growl).

Notifications

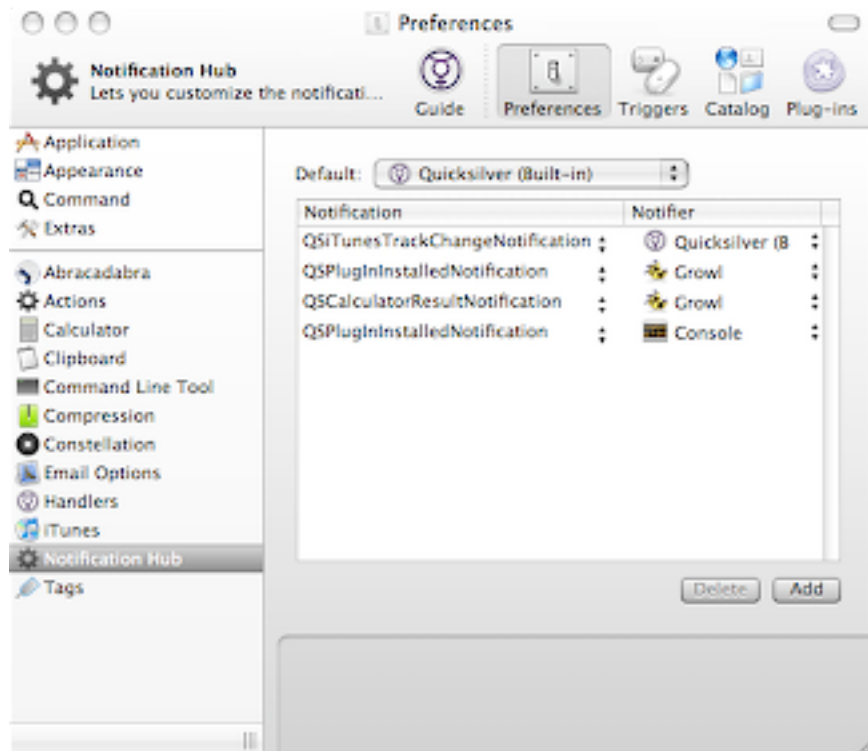
Quicksilver supports 3 different notification events:

- iTunes Track Changes
- Calculator Results
- Plug-in Installation

These are typically displayed using Quicksilver's built-in notification system which pops up a small window (shown at left for an iTunes track). In Handlers preferences under Notifications you can change to have notifications sent to the Console. If you install the Growl Notifier plug-in you can instead choose to have [Growl](#) handle notifications. One limitation is that iTunes notifications generated by Quicksilver and displayed via Growl will not show the track's rating. Ratings are shown when using Quicksilver's built-in notifications.



The Notifications Hub plug-in allows finer granularity in configuring notifications. It adds a new preference pane called Notification Hub. There you can configure a default notifier as well as configure specific notifiers for specific events. E.g., iTunes notifications go to the Quicksilver built-in notification notifier while Plug-in Installation notifications go to Growl. You can even send the same notification to multiple handlers by adding multiple lines for the same Notification and with different Notifiers.



Triggers

Trigger Basics

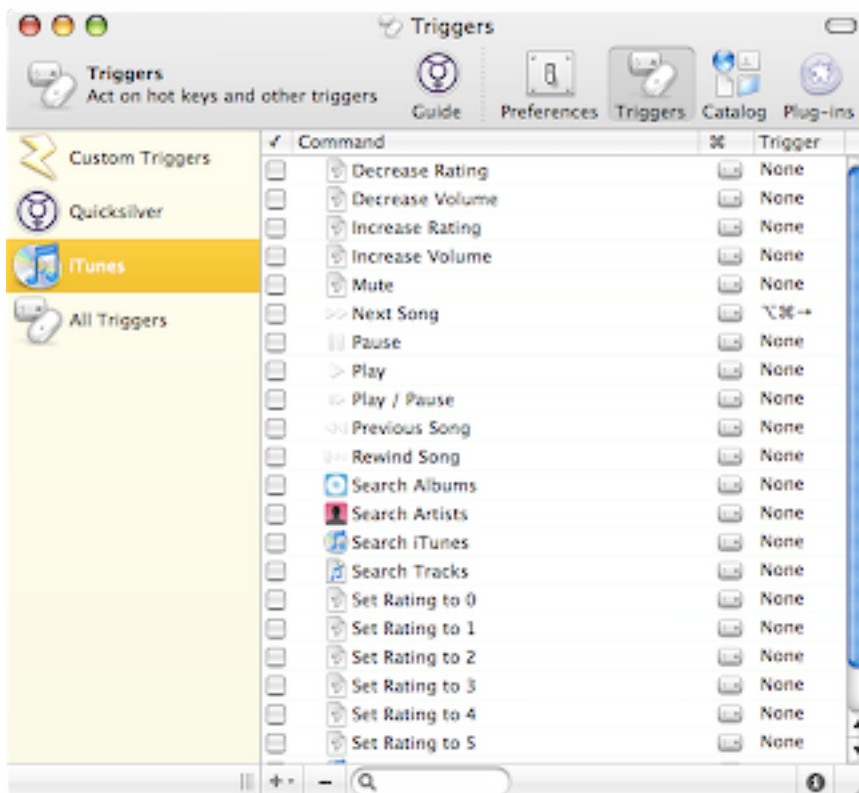
Triggers allow you to execute Quicksilver commands without having to use the command window. Quicksilver supports executing triggers by typing a HotKey, clicking or dragging the mouse, or (if you have the Abracadabra plug-in installed) making a mouse gesture. Triggers are available whenever Quicksilver is running and you do not have to activate Quicksilver (with control-space) to use them. Triggers make it easier to control iTunes, launch applications, perform web searches, or do anything else Quicksilver can do. You can create as many as you want but you probably want them only for the operations you do frequently.


Your Trigger configuration is stored in `~/Library/Application Support/Quicksilver/Triggers.plist`. This is useful to know if you want to have the same configuration on several machines, just copy the file manually between the machines while Quicksilver is not running.

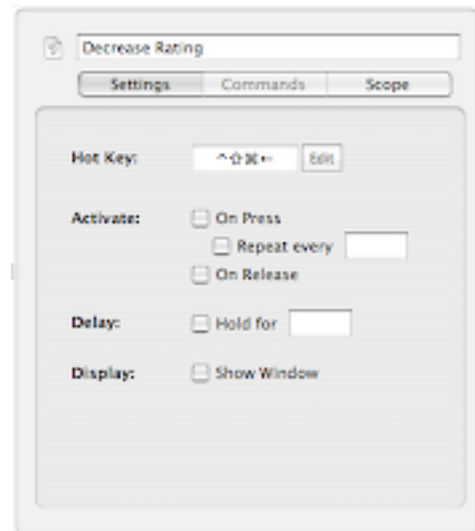
You define triggers in the Triggers Preference pane. You can go to this pane directly by activating Quicksilver and typing `⌘`. You'll see several sets in the left side of the preference pane. Some triggers are predefined by Quicksilver itself or various plug-ins, e.g., you'll find several in iTunes, and two in Quicksilver. Triggers you create will be in the Custom Triggers set.

Prior to B51 the Constellation plug-in installed its own trigger set with three triggers defined, but it no longer does so..

Each trigger is shown by its name which is usually a combination of the object and action to be invoked. The checkbox shows if the trigger is active. The `⌘` column shows the type of trigger, in this case they are all activated by a

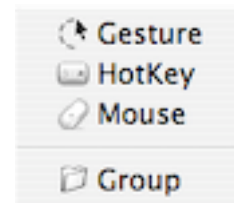


HotKey which is shown in the Trigger column (some are not assigned to keys). If you select a trigger and click on the  button at the bottom a drawer is revealed with various settings for the trigger. The settings tab looks different for each type of trigger (Gesture, HotKey, or Mouse). Shown here is the one for a HotKey. You set the HotKey by clicking the Edit button and then typing whatever key you want. You can change the name of the trigger in the top text field. There are various other options to tailor the keystroke to behave exactly the way you want, we'll discuss them below.

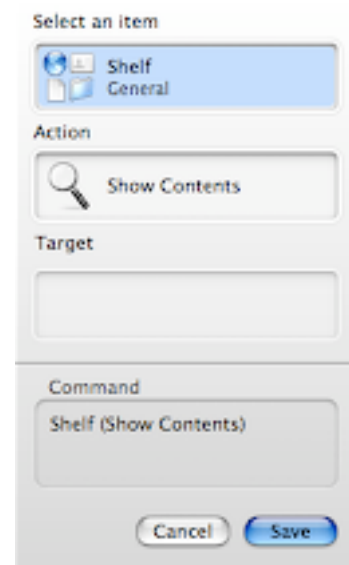


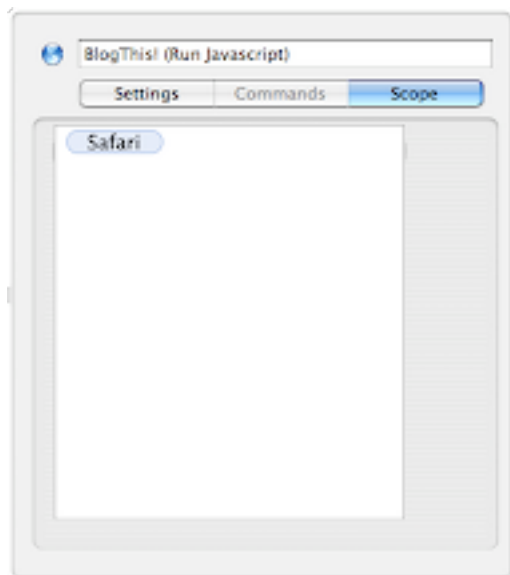
You can change some options by clicking in the main trigger window. If you click on the command name you can get an edit field to change the name of the trigger. If you click in the Trigger column you can set or change the HotKey. If you click on the icon in the ⌘ column you can add another means (Gesture, Mouse) to activate the trigger. Though once you create an additional way, there doesn't seem to be a way to delete it without deleting the entire trigger.

You create a new trigger by clicking the + button at the bottom and selecting a trigger type from the pop-up menu (Gesture, HotKey, or Mouse, depending on installed plug-ins). The Group option is just a way to collect triggers in a group or folder. They don't perform any function other than helping you organize a lot of triggers. You can't activate all the triggers in a group at once. Create a Group from the + menu and drag triggers into it.



Regardless of which type you choose a special command window appears (populated with the last command you did) to let you define the command for the trigger. Enter the command and click Save. Then open the drawer to the Settings tab to assign a HotKey, mouse click, or gesture to use to activate the trigger. If you're creating a HotKey trigger, just click in the Trigger column for this new trigger, the drawer will open and you can just type the HotKey as if you had clicked the Edit button.





By default, triggers are available all the time, regardless of what application is active (as long as Quicksilver is running). They can also be limited to function only when a certain application is active or when a certain application is not active. You do this by opening the drawer for a trigger and choosing the Scope tab. One place I use this is for a BlogThis bookmarklet in Safari. Like any bookmarklet it's javascript which in this case opens a small browser window with a blog entry form and the current URL filled in. Since Safari treats it as a bookmark I can't easily assign a shortcut to it. So I have a Quicksilver HotKey trigger which opens that bookmark (it's in the Catalog with my other Safari bookmarks) and I have the scope set to work only when Safari is active because it's not useful in any other application.

Note if you have triggers that use certain features, be cautious about deleting the plug-ins that supply those features. In particular if you have triggers using mouse

gestures from Abracadabra (see below) and if you remove the Abracadabra plug-in, the trigger panel may display oddly with some blank lines and missing icons. To correct this, reinstall the plug-in (Abracadabra in this case), remove the triggers, and uninstall the plug-in.

In the following sections we'll discuss each type of trigger with many examples. The commands used will come from many different plug-ins not covered here. If you want more information on them, look in the relevant sections in this manual. While you have a lot of freedom in defining triggers anyway you want, some commands are better suited to certain trigger types. We'll start with HotKey triggers as they are most common and we'll introduce proxy objects there as they are very useful when combined with triggers. After that we'll cover Mouse triggers and Abracadabra gestures. Then we'll cover Constellation menus (including context menus) which while not triggers they are commonly used with them. Lastly we'll cover Droplets which are the only other way to use the mouse with Quicksilver.

Hot Key Triggers and Proxy Objects

If you enter a complete command into the trigger, like choosing an application and the **Open** action, it will be run when you execute the trigger. If you choose just a partial command, like a web-search and the **Search For...** action, when you execute the trigger a command window will appear with as much filled in as your trigger defines. You can't have gaps (e.g., you can't leave the first pane blank and use the **Get Info** action) but we'll discuss this more below. Triggers are given a default name based on the command they represent. Here are some examples:

1. Safari (**Open**)
2. Shelf (**Show Contents**)
3. Clipboard History (**Show Contents**)
4. Current Web Page (**Paste**)
5. Wikipedia Quicksearch (**Search For...**)
6. Current Selection (**Find With...**) Wikipedia Quicksearch

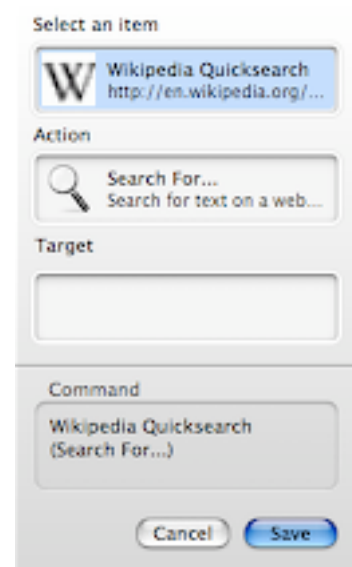
The first is a very basic trigger, it opens Safari. You can make similar triggers to open your favorite applications. Some people create other triggers like Safari (**Quit**) or Safari (**Hide Others**) and

put them on the same keys but with different modifiers (e.g., ⌘ or ⇧). The second and third examples give you quick keyboard access to the items on your Shelf or Clipboard History.

The fourth lets you paste the URL of the page Safari is displaying into any other application without switching applications and without having to do a copy step. I have it bound to ⌘F and use it all the time to paste an URL into a mail message or a chat. Current Web Page is a proxy object, which is an object that represents (i.e., proxies) for something else, in this case it represents the URL in Safari at the time the trigger is invoked (as opposed to just having a specific URL or bookmark in the trigger which would always paste itself). To use proxy objects make sure advanced features are enabled in the Application Preferences and make sure Proxy Objects are enabled in the Catalog under Quicksilver. Note the Current Web Page proxy object is kind of misnamed. It should be called Current Safari Page because it doesn't use the default browser but only Safari (it's defined in the Safari Module plug-in so install that if you don't see it). If you use OmniWeb you can use the OmniWeb Active Page proxy object. Unfortunately as of B51 there are no equivalent proxy objects for other browsers.

The fifth example is the first one that uses a third pane in the command. The **Search For...** action takes an argument of some text to search for. If you filled out the third pane in the trigger it would always search for the same text which is not what you want. If you leave the third pane blank, then when the trigger is run, it will open a command window with the first two panes filled out, in this case with Wikipedia Quicksearch as the object and **Search For...** as the action and leave the third pane selected, ready for you to type the query. In this case, because of the **Search For...** action, the third pane will already be in text mode and will have default text from the OSX Shared Find Clipboard (which you can set in many Cocoa applications with ⇧⌘E). Note if the third pane isn't empty and you want it to be, you can type ⌘X to cut out whatever is there.

I have the fifth example configured on the HotKey ⌘W , I type that, my query and return and my browser opens with the search results. Now what if I want to search for some text in a document? I could select the text, copy it, type ⌘W and then type ⌘V to paste in the text. That's not bad, but with the Current Selection proxy object I can make it even faster, just select the text and type ⇧⌘W (the same key with an added shift) which is what I have the sixth example configured to.



Current Selection gets the selection from the frontmost application, it uses a Service internally so it only works with Cocoa applications (e.g, Safari, Mail, Preview, Vienna, Colloquy, iWork, etc.) but that's most modern Mac apps, the notable exceptions being MS Office and Firefox (try Camino instead). This trigger does the search with the **Find With...** action which is the reverse of the **Search For...** action. It takes the text in the object pane, we use the Current Selection proxy object for that, the **Find With...** action and the search site (Wikipedia) in the third pane. I would have thought that Wikipedia (Search For...) Current Selection would work, but at least in B51 I can't get Current Selection into the 3rd pane (even with pasting). I have similar pairs of triggers for other quick searches such as Google (⌘G), IMDB (⌘M), and Amazon (⌘Z).

If you're now convinced that proxy objects are amazing, you probably want to know what other proxy objects are available. Open the Catalog, select the Quicksilver set and then select Proxy Objects source, click the ⊞ button at the bottom to open the drawer and look at the contents

tab. Or you can look at the table below which organizes them into similar categories. Note that some third party plug-ins might install additional proxy objects.

iTunes	Finder	iPhoto
<ul style="list-style-type: none"> • Album Now Playing • Artist Now Playing • Current Playlist • Current iTunes Selection • Selected Playlist • Track Now Playing 	<ul style="list-style-type: none"> • Finder Selection • Mounted Disks • Network Disks • Removeable Disks 	<ul style="list-style-type: none"> • Current iPhoto Selection • Selected iPhoto Album
Quicksilver	Mouse	Applications
<ul style="list-style-type: none"> • Clipboard Contents • Current Selection • Current Web Page • Last Command • Last Object • Quicksilver Selection 	<ul style="list-style-type: none"> • Mouse Trigger • Dragged Object • Droplet Item 	<ul style="list-style-type: none"> • Current Application • Previous Application • Hidden Applications • Visible Applications

It's obvious from the name what most of them do, with a few exceptions. We've already mentioned that Current Web Page really should be Current Safari Page. The Mouse ones we cover below. TODO: The Quicksilver Selection proxy object represents what is in the object pane (aka 1st pane). This allows you to assign HotKeys to particular actions (or actions and arguments). E.g., create the trigger Quicksilver Selection (**Get Info**) and assign it to ⌘I. Now you can activate Quicksilver, select a file object in the catalog or navigate to one in the filesystem and type ⌘I to bring up the Finder's Info pane of that file. You don't even have to tab to the action pane. You might think you could scope this trigger to only work in Quicksilver but that's probably not what you want. If you do so then the trigger will only work if the Quicksilver preferences window is active and frontmost. Otherwise if you're in some other application and activate Quicksilver, that other application is still considered active so the trigger isn't available.

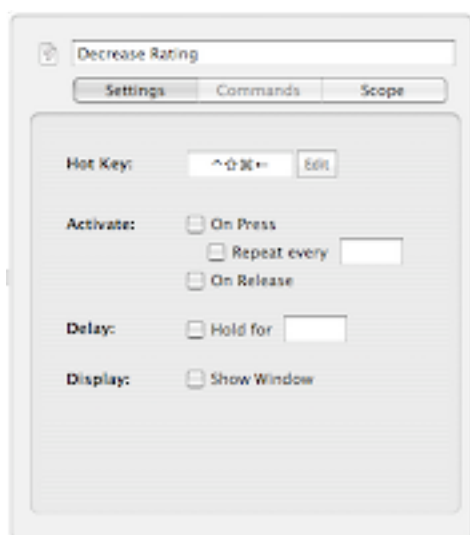
Sometimes when creating triggers with proxy objects, the action you want doesn't appear in the list. As you've probably noticed, Quicksilver normally narrows down the action list to only those that are relevant. Sometimes proxy objects confuse it because they represent something else. If this happens you can work around it with copy and paste. It's not obvious but if you bring up a regular command window (with ^-space), when you tab to the action pane you can type ⌘C to copy the action. You can then paste it into the second pane of the trigger definition with ⌘V. You can also cut an item out with ⌘X. This is useful when creating commands that do a web search with the **Search For...** action and you want a blank argument pane (3rd pane). There are many other proxy objects giving you access to interesting things to use in triggers such as Track Now Playing (in iTunes), Current Application, Previous Application, Finder Selection, etc. Experiment to see what you find useful.

The possibilities for triggers seem endless. Here are some triggers people have posted on the Quicksilver forums as their favorites:

- Documents (**Open**) - the Finder doesn't have a key binding for this
- Finder Selection (**Go To Directory in Terminal**)

- Current Application (**Menu Bar...**)
- Current Application (**Menu Bar Items...**)
- Current Application (**Hide Others**) - though `⌘H` already works in most every application
- Artist Now Playing (**Show Contents**)
- Browse Albums (**Search Contents**)
- killall Dock (**Run Command in Shell**) - or any other frequent shell command
- Current Web Page (**Open With...**) Camino - or any other browser
- Network Disks (**Eject**)

There are some limits to triggers that you can create. You can't save a trigger with a dynamic action. Dynamic actions are those that ask the application at runtime for their content. So the Current Application (**Menu Bar...**) trigger above works, but you can't do Current Application (**Menu Bar...**) Help. Also actions that appear only for particular applications can't be saved as triggers, so you can't do Mail (**Get New Mail**). For some of these you can physically create the trigger but they won't save properly and will be broken on Quicksilver relaunch.



The Settings tab in the drawer lets you set various options for a HotKey trigger. Hot Key: lets you set the key. Activate lets you determine if the trigger activates when the key is pressed or released, and lets you set it to repeat if the key is held down for a period of time (useful for a fast-forward script). With a Delay set, the trigger won't activate until the key has been held down for the number of specified seconds. The Display: Show Window option shows a command window when the trigger is activated which is useful to see feedback that a trigger has executed.

A delay can be useful for some dangerous commands that you don't want to execute if you accidentally type the key. E.g., if you had a HotKey to Run the Quit All Visible Applications script from the Extra Scripts plug-in, you might configure it to only execute if you hold the key down for 2 seconds.

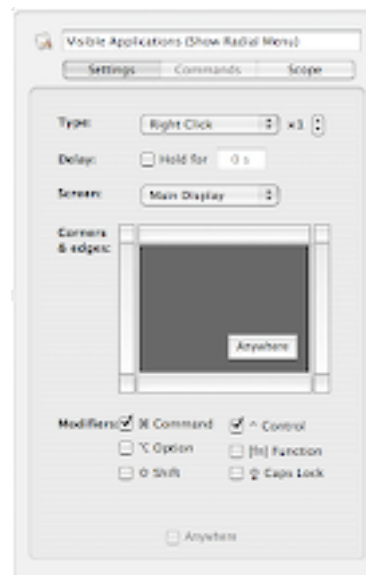
TODO: You can define combinations of triggers to do one operation on immediate press and a second on a press and hold. E.g., you can define `^⌘I` to open your Inbox and a second `^⌘I` trigger to Get New Mail if you hold it down for 2 seconds. (one key for mail, holding it checks for new mail and then opens it) You can also configure triggers that use two scripts, one on start, one on finish.

Using the Mouse

With the Mouse Triggers plug-in installed triggers can also be assigned to the Mouse. The Settings tab of the drawer for the trigger is different for a mouse trigger. You can assign triggers to a number of mouse clicks, with or without modifier keys. Note that the Anywhere button in the desktop is actually a button. If you want a trigger to work if you⌘-right-click anywhere in the window you have to click the Anywhere button for it to work. Also as of B50 it seems the Anywhere checkbox at the bottom is always greyed out.

You can also activate a trigger when the mouse enters or exits an edge or corner of the screen or if you drag something to an edge or corner of the screen. If you have multiple monitors connected to your Mac you can choose if the trigger will work on all displays or on a particular one. Dragging triggers work well with the Mouse Trigger Draggged Object proxy object. E.g, configure these when dragging to a corner or edge for your commonly used application or folder:

- Mouse Trigger Draggged Object (**Open With...**) *application*
- Mouse Trigger Draggged Object (**Move to...**) *folder*



Here are some more advanced ones that might require additional plug-ins

- Mouse Trigger Draggged Object (**Email to...(Compose)**)
- Mouse Trigger Draggged Object (**Compress (Create Archive)**)
- Mouse Trigger Draggged Object (**Set Desktop Picture**)
- Mouse Trigger Draggged Object (**Add Tags...**) *tags*
- Mouse Trigger Draggged Object (**Upload to Site...**) *Transmit favorite*

Note that if you configure a trigger to activate when mousing or dragging to an edge it might interfere with the [shelf or clipboard](#) windows if you have them on an edge. In such a case the trigger will win and you'll need to use some other means like a key stroke to open the shelf or clipboard windows.

One thing might not be obvious, you can configure the same trigger to activate by multiple means. E.g., if you have an Open Safari trigger assigned to the HotKey $\hat{\text{⌘}}\text{S}$ you can also assign it to having the mouse enter the left edge of the screen, and both will work. You do this by clicking the icon of the type of trigger in the ⌘ column of the Triggers preference pane. A pop-up menu appears of the other choices and you can select another one and configure it. Now both work. This can be useful to have two ways to invoke a trigger, one when your hands are on the keyboard and another when one hand is on the mouse. I don't see a way to remove only one method from a trigger. The - button at the bottom of the screen deletes the whole trigger. For a HotKey you can always change it to not be bound to a specific key by clicking the Edit button and then the delete key. For a mouse trigger you can achieve the same effect by choosing Mouse Entered and selecting no edge or corner.

Most useful on mouse triggers are the **Show Menu**, **Show Contents Menu**, and **Show Action Menu** actions. These bring up a context menu for the object. **Show Contents Menu** creates a menu for the items that would appear if you typed \rightarrow to go into the object, it's like **Show Con-**

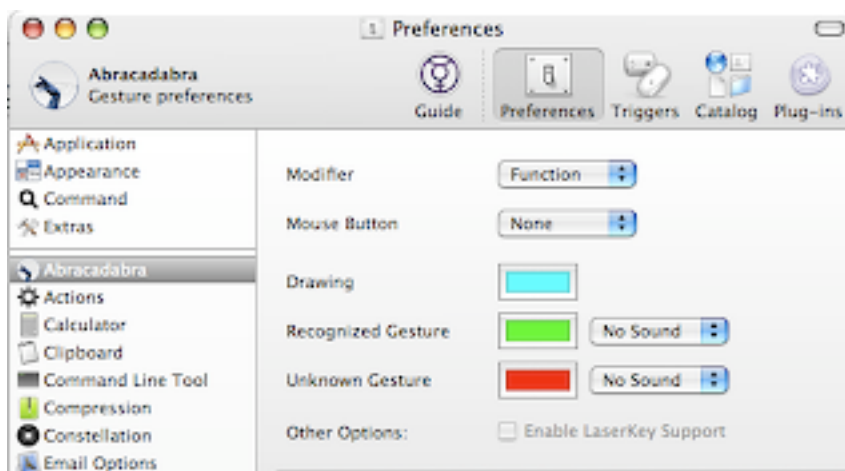
tents but in menu form. Items have sub-menus showing the actions you can perform on them and any sub-items they have. **Show Action Menu** shows a menu of the actions you can perform on an object. **Show Menu** shows a menu that combines the contents and actions you can perform. Mouse triggers for Shelf (**Show Contents Menu**) or Clipboard History (**Show Contents Menu**) are very useful.

Abacadabra Mouse Gestures

For this section you'll need to have Advanced features enabled. Then install the Abacadabra Triggers plug-in. This enables a new trigger type known as a Gesture to go along with HotKey and Mouse triggers. It also installs an Abacadabra preference pane.

Abacadabra lets you invoke a trigger by drawing a shape with the mouse. E.g., you can configure Quicksilver to activate the Open Safari trigger by drawing an S on the screen. You configure your shape by drawing with the mouse button down in the pane in the Settings tab of the trigger's drawer. The "Show zooming trigger window" will show a bezel window that enlarges when a gesture trigger is recognized and executed (like the Display: Show Window option of a HotKey trigger).

In the Abacadabra Preference pane you configure how you invoke all gestures. That is what mouse button you hold down and/or what modifier key you hold down while drawing any gesture. If you have a mouse with extra buttons it's probably most convenient to use one of them for this as you'll only need one hand to invoke the gesture. You can also choose the colors used to draw the shape on the screen as you draw and after you finish for a recognized gesture and an unrecognized gesture, as well as sounds to play.



There's also an option to Enable LaserKey Support which is a virtual keyboard device made by Cellulon. With it you can make gestures with your finger. Wherever I say mouse gestures here, LaserKey gestures is implied if you're one of the lucky few to have one of these devices.

Simple gestures are best as you'll have an easier time remembering them and Quicksilver will have an easier time recognizing them. You can create a gesture for any Quicksilver command (that is trigger). E.g., skipping to the next track in iTunes by drawing a line from left to right. Try a gesture for the **Quit** or **Hide** actions using the Current Application proxy object.

Gestures also go well with limiting the scope of a trigger and using the **Menu Bar Items...** action. This way you can make a trigger to invoke Safari's Back command or Mail's Reply command. This usually works best for things that don't need the keyboard, e.g., navigating in Safari. Using a gesture to bring up a Find dialog probably doesn't make much sense since you'll need to type the text to find. Expose actions work well with gestures.

An interesting idea is using the Current Application proxy object and the **Menu Bar Items...** action to invoke an action common in all applications, e.g., Undo or Close. The problem is that many applications dynamically update their menus (e.g., "Undo Typing") or slightly rename these commands (e.g., Close Window and Close Tab) so the trigger won't work in all applications. Also there's no way to select a menu command that is generic and not specific to an application, so this isn't possible.

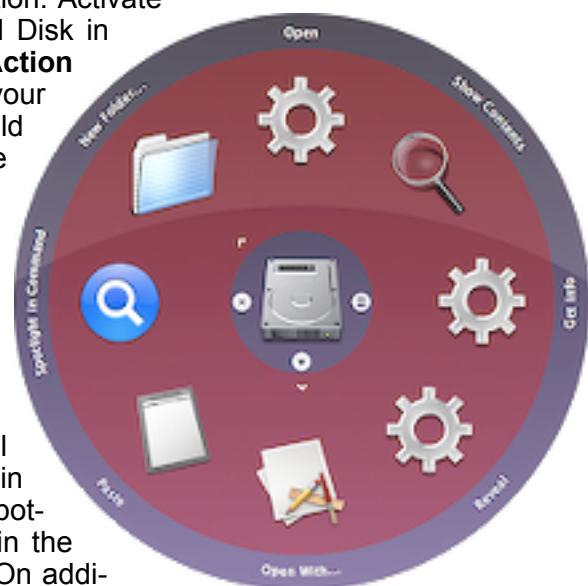
Constellation Radial Menus

Most menus on computers look pretty similar. You click on something and a rectangle appears with commands listed vertically. Move the mouse over the command you want and click and you're done. The one inefficient part is that if you want to choose the last item you have to move the mouse much farther than if you want the first item. Radial menus take a different approach. They pop up a circle around the mouse and divide the circle into pie pieces, one for each command. In this configuration choosing any item is just a different direction to move the mouse, but not any extra distance. Of course Quicksilver can use radial menus, and they look very very pretty.

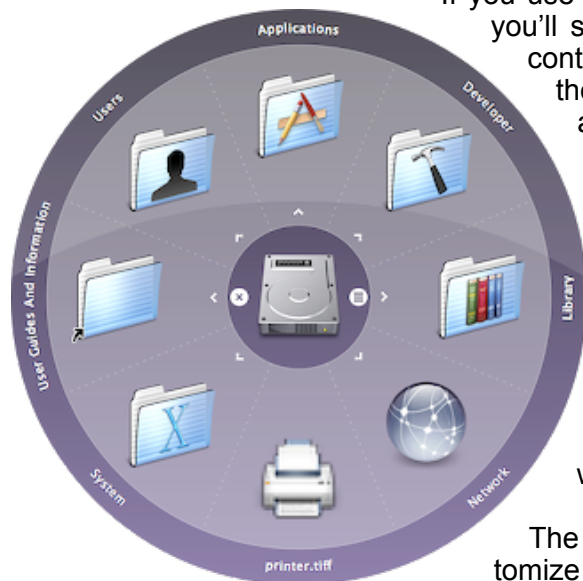
For this section you'll need to have Advanced Features enabled. You'll also need to check "Enable access for assistive devices" in the Universal Access System Preferences. In addition to Constellation you'll need to install the Mouse Triggers and User Interface Access plug-ins. Note that the plug-in installs a Missing Object Selector Handler (in Preferences, Handlers) that you can configure, but don't do that, leave it set to Quicksilver (see note below when discussing the Constellation triggers). The Constellation plug-in installs two actions, **Show Radial Menu** and **Show Radial Action Menu**. It also installs a preference pane called Constellation listed in General. Up until B51 it installed a new trigger set on the left of the Triggers preference pane with 3 triggers predefined.

Let's start with the **Show Radial Action Menu** action. Activate Quicksilver and bring up a folder (I use my Hard Disk in these examples) and choose the **Show Radial Action Menu** action. You'll see a menu appear around your mouse pointer with items for the actions you could choose, as if you had brought up the folder in the first pane and looked at the available actions in the second pane. Moving the mouse into one of the pie slices highlights it, click to perform the action.

In the center to the left is an x in a circle, clicking that will close the menu. To the right is a button with 3 horizontal lines, clicking it will bring up a traditional menu of the items shown in the radial menu. If there are more items to display than fit in the pie slices a down-arrow button appears at the bottom of the center circle. Clicking it or anywhere in the bottom half of the center icon will bring them up. On addi-

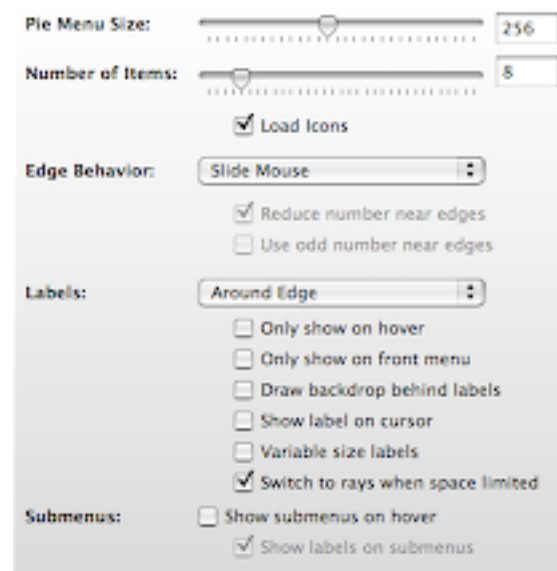


tional menus an up arrow is displayed and clicking it or anywhere in the top half of the icon will go back.



If you use the **Show Radial Menu** action for the same folder you'll see not the actions you can do to it but rather the contents of the folder appear in the menu. By clicking the items you can navigate into the folders bringing up additional radial menus of the contents of the sub-folders. The little arrows at the center of the pie slices correspond to >'s in a results list, it shows that there are sub-items. If you click on one without an arrow, a menu opens as if you had done **Show Radial Action Menu** on that object. If you control-click (or right-click) on an item with an arrow in it you will pop up an action menu for that item. If you double-click on an item you will perform the default action for that item immediately. So double-clicking on a folder will open a Finder window of that folder.

The Constellation preference pane allows you to customize the look of the menu. You can set the size of the menu and the number of slices in the pie. The Edge Behavior setting controls how the menu is displayed when the mouse is too close to an edge to show the whole menu. You can crop the menu (Slice Edge), move the mouse (Slide Mouse), or move the whole screen temporarily (Slide Screen). There are also various options for showing the labels (the above image is using "Around Edge") and whether submenus are shown when hovering.



TODO: The trigger set is not installed in B51, will it come back? Should I remove the following 3 paragraphs?

The Constellation plug-in installs a trigger set called, Constellation. There are three pre-configured triggers: Show Radial Menu for iTunes (on F6), Show Menu Bar Contents (Change Helper) (on F7) and Email Dragged Items (on mouse drag and drop into the lower-right corner). The first shows choices as if you had typed → into iTunes (Recent Tracks, Browse Artists, Genres, Albums, etc.). It's just the iTunes object with the **Show Radial Menu** action. You can click on these to open other radial menus to browse your music library. With album art showing it's very pretty. Control-clicking brings up an action menu with Play and Party Shuffle actions.

The second pre-configured trigger pops up a menu that shows the current application's menu items (like File, Edit, etc.). If you like the idea of radial menus you'll like this. It's somewhat special in that there seems to be no way to create this yourself. It's like the Show Menu Items action but in a radial menu. Note: this trigger only works if the Missing Object Selection Handler is set to Constellation. The problem is that setting it to Constellation can (as of B51) cause Quicksilver

to crash on some occasions (see Bug [1017](#)). In particular I have problems when using Web Search triggers with a blank third pane. My understand is that this handler was originally a hack and I suspect it will be removed in a future version of Quicksilver.

With the third pre-configured trigger you drag text or a file to the lower right corner and it pops up a radial menu of Contacts to send to. It's populated the same as the results list in the 3rd pane of a command with the **Email to...(Compose)** action so it's a list of commonly used contacts. The trigger is the proxy object Mouse Trigger Dragged Object with the action **Email to...(Compose)** but it pops up this radial menu (which with pictures in your Address Book entries is very pretty).

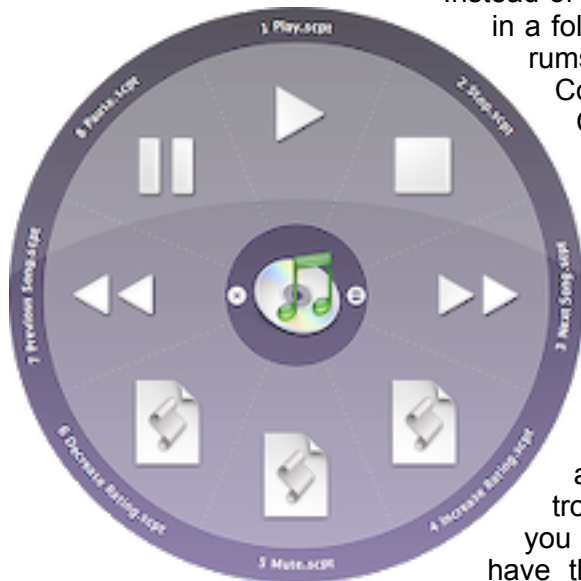
Those are the basics. You can of course use radial menus for lots of things and create your own triggers for commonly used ones (these will be in the Custom trigger set, not the Constellation trigger set). If you create triggers you probably want them bound to the mouse or at least to HotKeys easily typed with your non-mouse hand. What I think works well for radial menus are things that don't change position (which isn't so easy to arrange for) so you can learn them and things with clearly differentiated icons so you can recognize them quickly.

Here are some examples of objects to use the **Show Radial Menu** action on. Many of these are proxy objects which you must enable in the catalog under Quicksilver, Proxy Objects.

- Visible Applications - this is a cool mouse-based alternative to the ⌘-tab method of switching applications. (shown at right).
- Track Now Playing - Shows a four part menu of the album, artist, genre, and composer of the current track, ready for easy browsing.
- Artist Now Playing - this shows all the albums by this artist you have in iTunes (with album art)
- Clipboard History
- Shelf
- Finder Sidebar Items
- Desktop
- Comma trick - create a trigger collecting several things you use frequently via [the comma trick](#) and use **Show Radial Menu** to display them. Works well for contacts, bookmarks, playlists, presentation documents, etc.



I haven't thought of any good uses of **Show Radial Action Menu** as a trigger.



Instead of using the comma trick you can collect useful things in a folder. Here's a fun idea I got from the Quicksilver forums. Create a folder and call it something like iTunes Controls. Do a Get Info on it and on the iTunes.app. Copy the iTunes icon to the new folder. Now activate Quicksilver and bring up the Play.scpt that controls iTunes and use the **Reveal** action on it. This will open a finder window inside the iTunes Module.qsplugin showing the scripts. Now copy them to the iTunes Controls you created. Remove the ones you don't want and rename the remaining ones putting number at the beginning of the name to order them as you want. Now bring up the iTunes Control folder and use the **Show Radial Menu** action to bring up the controls as shown. Save this as a trigger to make these controls easily accessible. If you want to be more creative you can add your own icons for the scripts that don't have them. Jacob Rus has put some icons and extra scripts together you can download [here](#).

Command Objects and Droplets

Commands in Quicksilver can actually be represented as objects in Quicksilver. You create a command object as follows. Activate Quicksilver and enter a command as you normally do, but instead of typing return to execute it, instead type `^-return`. You see a new command window appear where the object is a new command object of the command you entered and a default action which is probably **Run**.

Other actions that work on commands include **Run**, **Run after Delay...**, and **Run at Time...** **Run** is obvious and not something you'll use often. The other two are useful as alarms when used with actions like **Large Text** or **Display Dialog**. You can enter a time as 9:35 or a delay as 30m or 1h. TODO: describe all valid syntax. There's also an **Add Trigger** action but in B51 all it seems to do is open the Triggers preference pane. It actually does add a HotKey trigger representing the command but it doesn't show up in the Triggers list until you restart Quicksilver. You can then assign a HotKey to it.

The action we're interested in is **Save Command to File...** which takes a folder as an argument to create the file in. It creates a file with the extension `.qscommand` which if you double-click it will run the command. Now creating a file to open an application would be pretty redundant, however you could create one that composes an email message to a group of people attaching the current draft of a document.

After not too long you'll want to drag a file or some text onto one of these saved commands to have it used for something and that finally brings us to Droplets. If your command uses the proxy object Droplet Item then you can do just that and the command file will have a special icon and a `.app` extension. Unfortunately in B50 these don't seem to always work and sometimes create running Droplet processes that have to be quit before you can delete the files.



Files and Folders

I'll begin with the most commonly used items on your computer, files and folders. You can select any file or folder on your computer as an object. In the catalog you'll find under the User object entries for your Home, Documents and Desktop folders. You'll also see under Dock an entry for Dock Files & Folders and under Recent Items you'll find Recent Documents, Favorites and Recent Folders (Open & Save). Because you probably have a lot of files on your machine, Quicksilver doesn't scan them all into the catalog. Instead the folders only scan to a configured depth. The default Home and Documents sources are only indexed to one level down. These sources are not configurable themselves, but if you want more you can open the info pane for the item, go to the attributes and click on the Create Copy button. This will make an editable copy of the item in the Custom section of the catalog. If you bring up its info pane you can change the Source Options and make the depth something else, e.g., 3. You can also add other folders to be indexed to the Catalog with the + drop down button and choosing File and Folder Scanner.

As I write this manual, the file is called Quicksilver.pages and it's on my Desktop. Since my Desktop is in the catalog I can select it in the object pane by just typing its name. As I type `Qui` it appears with the default action of **Open**. If I type `return`, Pages will open the file. You can also select the Desktop itself by typing `Desktop` (or as much of it as I need to match it).

Quicksilver isn't limited to only what's in the catalog. If a folder is in the object pane, you can navigate into it by typing `→`. Then the object pane changes to a results list of all the child files and folders. Since it's more natural when thinking of files to use `/`, that works too. If you hold down the `⌘` key when navigating, the results list will include hidden files (those that begin with a period) and you can browse into packages (a la the Finder's Show Package Contents). Quicksilver also understands the unix shortcut for the home directory, the tilde (`~`). So since the Desktop is just a folder in your home directory I could navigate to this manual's file (the path is `~/Desktop/Quicksilver.pages`) by typing `~/De/Qui`. Quicksilver also interprets a backquote (```) as a tilde (I guess because on US keyboards the tilde is a shifted backquote). Also, while navigating down a folder hierarchy you can back up by typing left-arrow or `?` (which is `⇧/` on a US keyboard). You can also backup all the way to the root directory of your mac. Or you can start at the root directory by typing `/` and holding it down for a second. Disks are also included in the catalog so you can begin at them by typing their name.



Basic File Actions

For a file the default action is **Open** which will open the file in its default application just as if you had double-clicked on the file's icon. If you select a folder, the default action, **Open**, will open a Finder window for that folder, just as if you had double-clicked on the folder's icon. But there are other actions you can choose. There's **Open With...** to open the file with a different application that you specify in the third pane. The third pane will have a results list of applications to choose from. If your desired application is not there, you can navigate to it starting with `/` or `~` or by selecting the application in the Finder and using `⌘G`. Sometimes the third pane is in text-mode, exit text mode by typing `esc` and navigate to the application you want.

Another way to use **Open With...** is to select the file you want to open in the Finder, then activate Quicksilver and select the application you want to use in the object pane. Then instead of tabbing to the action pane, just type `\⌘G`. The command window will change so the Finder's selection is the object, the action is **Open With...** and the application is in the third pane. It's a little convoluted but it's quick.

We saw how the common action for a file object is **Open**. For scripts however this could be unclear, should they be opened in an editor or executed? Quicksilver solves this by having the **Open** action open them in an editor and having several **Run** actions. More about them in the section on the [Terminal](#).

The **Reveal** action will open the containing folder and select the item. If you're finding the item directly via the catalog instead of navigating down to it, this can be a very easy way to find what folder it's in. I do this often to find applications (which are in my catalog) and might be in `/Applications` or `~/Applications` or a couple of layers down. In fact it's so useful Quicksilver has a built-in keybinding for it. If you select some file in the first pane and the action is **Open**, if you type `⌘-return` Quicksilver will preform the **Reveal** action.

Quicksilver includes three methods that take a file and return a form of the files name. All return an absolute path on the machine to the file. **Get File Path** returns a unix style path with directory separated by slashes. **Get File Location** returns a mac style path with colons. **Get File URL** returns a `file://` URL to the file. They all show the path in a new command window. I can't think of what you'd want to use this for except to copy and paste it somewhere.

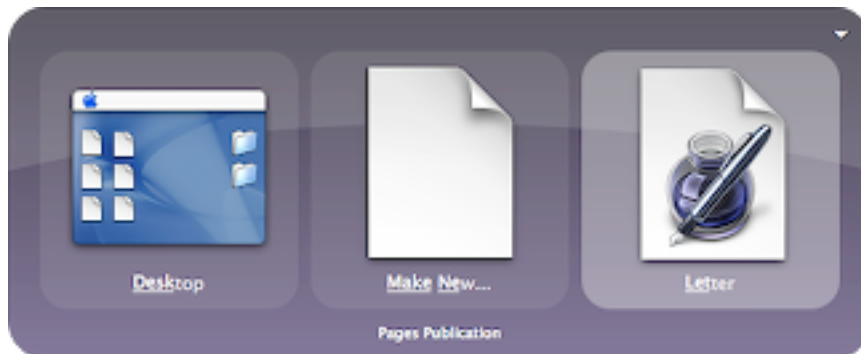
Much more commonly used are the actions **Rename...**, **Copy to...**, **Move to...** which do what you'd expect. The third pane opens for you to enter the new name or location. I prefer renaming files with Quicksilver vs the Finder because I can use Cocoa editing keys (like `^B` and `^D`). Since moving something to the trash is so common, there's an action called **Move to Trash**. There's also a **Delete (Erase)** action which skips the trash and actually deletes the file, as if you had put it in the trash and then emptied the trash (but just for that object). Since it's undoable, I've disabled this action in the Action Preference pane.



The `\⌘G` trick also works with the **Move to...** action. Select the file or files you want to move in the Finder. Activate Quicksilver and bring up the destination folder in the object pane. Then, without tabbing to the action pane, type `\⌘G`, the command window changes so the files are in the object pane, the action is **Move to...** and the destination folder is the argument.

There are actions **Make Link in...** and **Make Hard Link in...** which create the unix-style links in the folder specified in the third pane. There's also **Make Alias in...** for OS X-style links known as aliases. The **Make New...** action requires a little configuration. Create the folder `~/Library/Application Support/Quicksilver/Templates/`. Now put some template files in there. E.g., an empty `untitled.txt` file or a `Letter.doc` with your return address filled in or a `Letter.pages` with the return address created by Pages cleaned up (e.g., removing your homepage and phone numbers). Now activate Quicksilver, select a folder as an object, the **Make New...** action,

and tab to the third pane to choose your template. A copy of the template is created in the object folder and a new Quicksilver command window appears with the new file selected as the object with a default action (probably **Open**) which you can use or change to **Rename...**, etc. It's a bit like the Finder's Stationery Pad functionality integrated into Quicksilver.



The **Get Info** action will show the Finder's info panel for the object, just like choosing Get Info from the Finder's menu or typing \mathbb{I} . There are several actions that will modify information in the info panel. These following actions are part of the File Attributes Actions plug-in. **Set Icon...** will change the files icon to one you specify in the third pane. If you are a spotlight user and like adding metadata to files you'll like the **Set Comment...** action. This opens a third pane to let you enter text which will become the Spotlight comment (visible in the Get Info panel of the Finder). It doesn't add to the comment but replaces whatever the current comment is with the text you specify. If you use labels you'll like the **Set Label...** action. The third pane will open to let you choose one of your labels. The **Lock File** and **Unlock File** actions do what they sound like. The **Always Open Type With...** action will tell OS X to always open files of that type with the specified application, just as if you had used the Change All... button in the info panel. The **Make Invisible (hide)** and **Make Visible (Show)...** actions change the visibility attribute of files. To find invisible files use the Finder's Find command (\mathbb{F}) and from the search attributes pop-up menu choose Other and then Visibility, then pick the desired value (Visible, Invisible, or either).

File Tagging and Spotlight

For a richer metadata experience consider installing the File Tagging Module plug-in. Note that people have reported problems using this module if File Vault is enabled. This plug-in adds five actions and a configuration preference pane. Tags are words stored in Spotlight comments, so they are searchable. To differentiate them from other comments they are prefixed with something. In version 71 of the plug-in the default Tag Prefix of @ confuses spotlight. Consider changing it in the Tags preference pane to & which works well. Three of the tag actions are obvious: **Set Tags...**, **Add Tags...**, and **Remove Tags...**. All three actions take files or folders as objects and use a third pane to enter the tags you want to use via text mode. Tags are separated by spaces so they are single words, when entering them do *not* include the Tag Prefix, it will be added by Quicksilver. **Set Tags...** replaces all the tags and **Add Tags...** will append more tags leaving the current ones in place. A detail that might seem like a bug but isn't is that **Set Tags...** will *not* overwrite any existing Spotlight comments, but **Set Comment...** will overwrite any tags that are there.

You can also bring up tags in the first pane and perform some actions them. There are two ways to do this. First you can bring up the the File Tags (Catalog) object (In the Catalog, in the Quicksilver set, in the Quicksilver Catalog Entries source) in the first pane and type \rightarrow to see all the tags you've defined. To look at the tags assigned to just one file, bring up that file in the first pane and use the **Show Tags** action to bring up a new command window with a results list of all tags of the requested item. With a tag in the first pane there are several available actions such as **Copy to Clipboard** and **Paste** but the default is **Show Files Matching Tag**. Since the tags

are searchable with Spotlight, this action opens a Spotlight Finder search window (like when editing a Finder smart folder) with all the files containing the tag. As an alternative to get these results in an object pane in Quicksilver, just type → while on a tag in the first pane, after a little bit of time for a search you'll get a results list of all files with that tag.

Speaking of Spotlight, there are 3 actions that work on folder objects. **Spotlight Contents...** and **Spotlight Filenames of Contents...** return their results into a new results list in Quicksilver. **Spotlight Contents in Finder...** runs the search in a Finder window. All three prompt for the the string to search in the third pane. The second only searches filenames not the contents of files.

File Compression and Email

With the File Compression Module plug-in installed you get three additional file actions: **Decompress**, **Compress (Create Archive)**, and **Compress Using...**. The first is obvious, if you give it an archive (zip, cpio, or cpgz) for an object it will decompress it, except in B51 it won't appear as a valid action, instead use the **Open** action which will decompress the archive.. The second will zip the object, creating a new zip archive in the same folder (the original item remains), and a new command window appears with the new zip archive as the object so you can choose another action such as **Move to...**, **Email To...** or whatever. The **Compress Using...** action opens a third pane giving you a choice of compression tools to use such as tgz, cpio, cpgz, zip, or tbz. As of B51 it only works if you call it with a single file in the first pane. If you use [the comma trick](#) or drag multiple files into the first pane, then executing the Compress Using... action in the zip format will merely result in a ding sound.

From the Mail Support plug-in (installed by default) we get the three actions **Email To... (Compose)**, **Email To... (Send)**, and **Email To... (Send Directly)** which all send a file object as an attachment in an Email message to an address or contact you select in the third pane. The **Compose** variant uses Mail.app and opens a message compose window for you to fill out. The **Send** variant also uses Mail.app but doesn't open a window it sends the mail automatically. Both the **Compose** and **Send** variants will start Mail.app if it's not running. The **Send Directly** variant doesn't use Mail.app to the send the message but does so directly from Quicksilver, as a result Mail.app doesn't need to be running and the outgoing message is not stored in Mail.app's Sent folder. The **Send Directly** variant does get the SMTP configuration from the first mail account listed in Mail.app, so make sure that's setup correctly. In the Email Options Preference pane you can select the default message text and reply address to use for the generated message. If you include %@ in your template it will be replaced with the name of the file (unless you send more than one file using the comma trick, in which case it will say some variation of "# Files in Folder" such as "2 Files on Desktop" or "2 [PDF Document] in Documents".

There are many more file and folder actions, but let's talk about a Quicksilver feature that makes these all more powerful. By using the comma trick (select an item in the object pane, type a comma, select another item, repeat) you can select multiple objects. Then choosing a single action works on all of them. This lets you move several files to the trash at the same time, or add or remove a tag to several files in one shot or send multiple attachments in a single email message to someone. As you learn more functions in Quicksilver the possibilities seem endless.



3 objects will be moved to the trash

Pathfinder

If you use Cocomatech's Path Finder instead of the OS X Finder have no fear. Install the Path Finder Module plug-in and go to the Handler's preference pane. Under File System Browser choose Path Finder Module. Now the **Open** and **Reveal** actions will open in Path Finder. So with the **Get Info** action. There are two catalog objects Path Finder Recent Files and Path Finder Recent Folders (listed in the catalog under the Path Finder Module), that you can bring up in the first pane and type → to see the corresponding list. If you have a file selected in Path Finder, you can activate Quicksilver and type ⌘G to bring the selection into the first or third pane.

TODO: I see there's a new Path Finder proxy object and I suspect you'd want to use that instead of Finder Selection. Can you → into the Path Finder.app?

File Transfer Programs

The Cyberduck Module plug-in let's you type → into the Cyberduck application to see its bookmarks. You can then use the **Open** action to bring up Cyberduck opened to that site. It's not a lot, but if you have several bookmarks, it's convenient, particularly as Cyberduck does not have to be running for → to work.

Install the Transmit 3 Module plug-in and you get similar access to Transmit's favorites. You can type → into the Transmit application to see your Transmit favorites. The favorites are also listed in the catalog under Modules, Transmit Favorites, so you can type them directly in the object pane. The plug-in also adds the **Upload to Site...** action which works on a file object and takes a Transmit favorite as an argument in the third pane. Transmit remote sidebar items are accessible by typing → into a favorite (even though there isn't a > in the results list indicating you can do so).

You can launch Transmit connected to a specific location by bringing up a favorite or sidebar item as an object and using the **Open URL** action. The one gotcha is that if you connect this way, Transmit won't show the relevant remote sidebar items (it will if you connect to the remote site from within Transmit).

Disk Image

For this section you'll need to install the Disk Image Support plug-in. It installs 3 actions: **Create Disk Image**, **Compress Disk Image**, and **Internet-Enable Image**. The Create Disk Image action works on a folder object and creates the image with the same name as the folder (of course

with a .dmg extension) in the same location as the folder. The other two actions work on Disk Image objects (that is .dmg files). The **Compress Disk Image** action takes a disk image as an object and creates a new disk image in the same location with a .c.dmg extension. It's supposed to use zlib compression but in my experiments the .dmg did not get significantly smaller, though it behaved the same as using Disk Utility to compress it. The **Internet-Enable Image** action will make the image be self-opening when downloaded.

Applications and Processes

If you needed to say what category of application Quicksilver was you'd have to say it's a launcher. Of course it's a lot more but it's also very good at launching applications.

If you look in the Catalog sources the first item is Applications. Open that up and you'll see four items. The first, Applications, finds all system-wide apps in /Applications/. The second, Applications (User) finds all user-specific applications in ~/Applications/. Both look three levels deep in the folder hierarchy. The third, Running Applications & Processes, shows all running apps. It has an option (in the Source Options tab in the drawer) you can check to include background processes as well. The fourth, Find All Applications, will scan the whole system to find all things that end in .app or .service and scripts saved as applications. I include the first three in my catalog, the fourth is a little too much, it finds 412 items on my system. In the Catalog under User and Dock you'll find a Dock Applications source, There is also a source for Recent Applications under User, Recent Items. Even if an application is found by multiple sources (e.g., where it's installed, the Dock and recently used), it's still only in the catalog once. The end result for Quicksilver is, you can activate it, type the name of an application and type `return` to run it since, the default action is **Open**.

The **Open** action works on Dashboard widgets as well. You can add a custom catalog source for /Library/Widgets/ and another for ~/Library/Widgets/ to bring your widgets into the catalog. Using the **Open** action on one (or more with the comma trick) will activate the Dashboard and open the widget. This is useful for widgets you don't keep open in the Dashboard but use occasionally. You could even assign a trigger to them.

Since applications are really just files on your disk (well actually a type of folder called a package) all the file and folder actions described above work on applications. If you want to know which folder an application is in or just want to make it appear in a Finder window (for example to use Show Package Contents) the **Reveal** action works very well.

Some plug-ins add features for specific applications. Often for an application you can select it as an object and type `→` (or `/`) to navigate into the application's data. E.g., for iTunes typing `→` will allow you to browse the music library, for Mail you can browse mailboxes, for iPhoto you can browse photo libraries and albums, etc. Even if there isn't special support to show you info often you can type `→` into an application to see a list of recently opened documents you can select. If you use `⌘→` to navigate you'll go into the application's package as if you had done Show Package Contents in the Finder.

The User Interface Access plug-in allows you to access the menus of running applications. To use this you must have "Enable access for assistive devices" checked in the Universal Access System Preference and you must have advanced features enabled. If you select a running application in the object pane the **Menu Bar Items...** action will bring up a third pane where all the menu items are available in a results list. This basically lets Quicksilver do anything any application can do. In some applications, menu items aren't necessarily unique. For example, in IM application Adium, each IM account has "Available" and "Away" commands and if they are in one

long list you can't tell them apart. In such cases the **Menu Bar...** action is more convenient. Instead of one long list of all items from all menus in the third pane, you get a results list of the top level menu items which you can navigate with → to find sub-items. This is also good for Browser applications because with many bookmarks, Quicksilver can take a long time to build a menu of all commands. **Show Menu Items** is similar but it opens a new command window with a results list of all items from all menus in the first pane. Selecting one of those items lets you choose **Press** or **Perform Action** among others. TODO, how are they different? TODO the **Select** action. Read: <http://developer.apple.com/releasenotes/Accessibility/AssistiveAPI.html>

Quicksilver has several built-in actions for running applications. If you bring up the application in the object pane you can **Quit** the application or **Quit Others**. If you want to Force Quit an application use the **Kill Process** action from the Process Manipulation Actions plug-in. **Relaunch** will quit and then start the application. This is great if you want to load new plug-ins or settings that require a restart.



You can **Hide** an application or **Hide Others**, just as you can from the application's menu. **Toggle Application** will show a hidden application and hide a visible application; it also will start an application that isn't running. Some people like creating triggers for favorite applications with the **Toggle Application** action to let them get to and hide applications quickly. Using **Toggle Application** allows both functions to be on the same trigger which lets you mentally associate an application with a key rather than separate keys for show and hide. E.g., you could assign ^⌘S to Safari.app (**Toggle Application**) and then when you start or stop using Safari you type ^⌘S. Some people use mnemonic triggers, others use function keys for different applications, whatever works for you. **Toggle Application** works on the whole application, including all of its windows, there's no equivalent in Quicksilver to show and hide specific (e.g., document) windows.

Activate makes the application active, just as if you used ⌘-tab to choose it. Note the application must be running to use **Activate**. If an application's window is open or hidden, **Activate** will make it appear. If the window is closed (e.g., via the red button in the top left), **Activate** will make the menu for the application appear but not the window. **Activate** seems to bring all the windows of an application to the front (they may overlap). This can be useful for the Finder if you want to drag and drop between windows. There isn't a mac standard command to bring all the windows to the front like there is for Hide or Minimize. If you do this often, create a trigger for Current Application (**Activate**) and bind it to a HotKey.

What happens if you use **Open** on an already running application? There's a setting in the Extras Preference pane to define this. You can choose to make it do the same as **Activate**, or to always show the front-most window, or to always show all windows. I'm not sure why you'd ever want to use **Activate** instead of **Open** but that's the difference.

For more specific control of running applications you can install the Process Manipulation Actions plug-in. These actions are probably only of interest to programmers. You can **Raise Priority** and **Lower Priority** of the process or set a specific priority with **Set Priority...** which opens a third pane to enter a number. You can also send an arbitrary signal to a process with **Send Signal...** which will open a third pane to choose a signal to send by its symbolic name. For the common signals there are the shorthand actions **Kill Process** (which is SIGKILL), **Resume Application (SIGCONT)**, and **Pause Application (SIGTSTP)**. Note that **Kill Process** is the same

as Force Quit. **Launch a Copy** will start the application a second time. List Open Files will open a new results list with all the files the application has open. **Sample Process** will TODO.

iTunes

For this section you'll need to make sure you have the iTunes module installed in the Plug-ins Preferences pane. There are no configuration options in the Plug-in but a new iTunes Preference pane is installed. In the catalog under modules you'll find iTunes listed with two sub-entries, "iTunes Playlists" and "Control Scripts". You'll also find several iTunes related proxy objects installed.

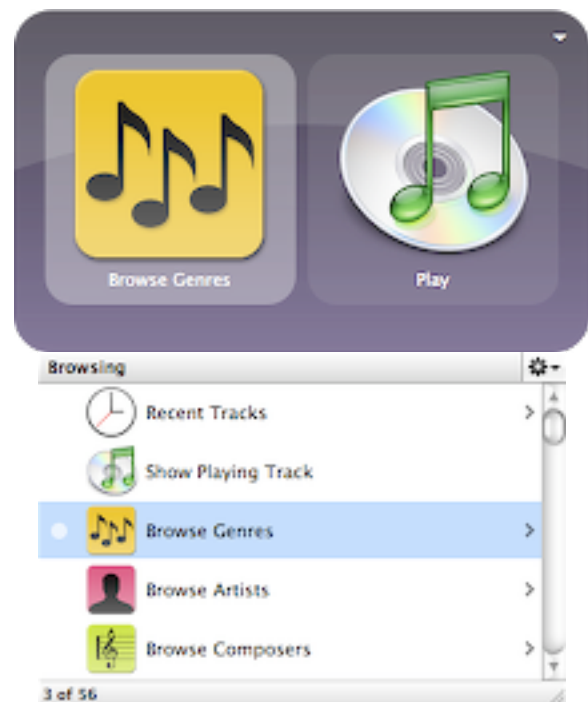
Activate Quicksilver and select the iTunes application. You'll see a > next to iTunes in the results list. If you type → you'll move to a new results list with iTunes specific choices. You can now browse your music library by genres, artists, composers, albums, tracks and recent tracks. The list also includes all your playlists and smart playlists. All of these items are also in the main catalog and you can get to them by just activating Quicksilver and typing enough to select them.

For the Browse options you can type → to navigate through the list to find a track or grouping you'd like to play. Note that all your songs and artists are not directly in the main catalog, you can't just type the name of a song to select it. This is to keep the catalog uncluttered and snappy. If you want this ability, you can create a trigger for Music Library, **Show Contents** and assign it to the HotKey of your choice. You could use Browse Artists or Browse Albums as the object for those kinds of searches.

If you select a song or playlist (or any grouping) you can use the **Play** action to play it (simple enough). There are also iTunes specific actions for **Play in Party Shuffle**, **Play in Next Party Shuffle** and **Add to End of Party Shuffle**. See the iTunes documentation for how to use Party Shuffles.

The iTunes actions are probably the ones you want to use, but since a track is also a file, all the file actions are accessible as well. Choosing **Open** will play the song just as **Play** does. **Reveal** will open a Finder window showing the track file. If you want to use a different player you can choose **Open With...** Whatever you can do to a file you can do to the track, email it to a friend, add a tag, compress it, etc.

The selection tricks don't stop there. If you have a track selected you'll see there isn't a > symbol in the results list, but you can still use →. It will show you a list of items to browse for the same genre, artist, composer and album of the song. Now here's a place where proxy objects shine. There's a proxy object called Track Now Playing, if you select it as the object, you can type → (provided iTunes is currently playing a song) to see the track object now playing in the results list (so the generic proxy is replaced with the actual track object). Then you can type → to browse by genre, artist, album, etc. There are also proxy objects for Album Now Playing, Art-

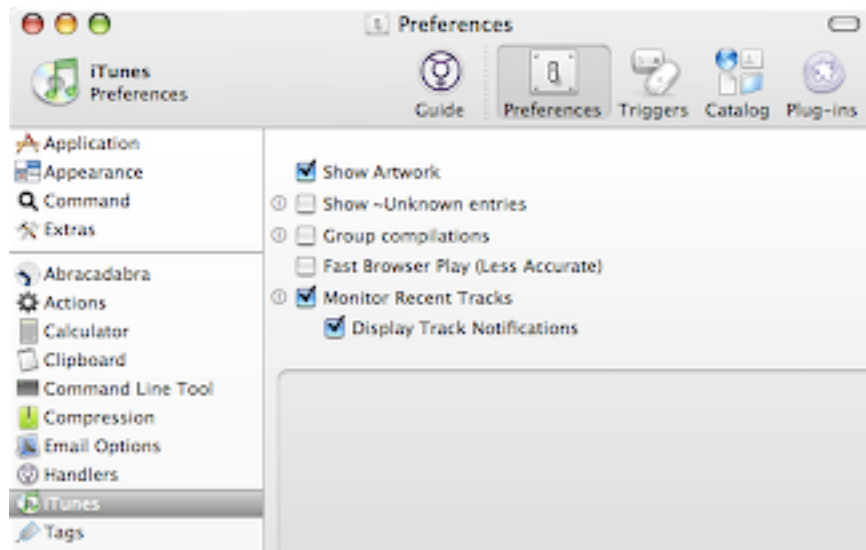


ist Now Playing and (the differently named) Current Playlist. If you're in iTunes and selecting songs with the mouse you might find the Current iTunes Selection and Selected Playlist proxy objects useful.

The iTunes Plug-in also adds scripts to control iTunes. These scripts are selectable as objects and you can choose **Run** as the action. You can find them in the catalog under iTunes as "Control Scripts". The scripts will let you play and pause iTunes as well as move to next or previous tracks, increase or decrease volume, mute, etc. If you like rating your songs and are frustrated that neither the minimized iTunes controller nor the Apple iTunes widget let you set a rating, you'll love the 6 scripts that let you set the rating to 0, 1, 2, 3, 4, or 5 stars.

This is a place where Triggers shine. In fact, the scripts are installed in the Triggers pane and there is a drop down to filter to just the iTunes triggers. I've configured the rating scripts to the HotKeys $\text{^}\text{⌘}0$ through $\text{^}\text{⌘}5$. I also have $\text{^}\text{⌘}\rightarrow$ and $\text{^}\text{⌘}\leftarrow$ keys set to move to next and previous track and $\text{^}\text{⌘}\uparrow$ and $\text{^}\text{⌘}\downarrow$ as mute and play/pause. Adding a shift to the combination I've set to change the volume and increase/decrease rating. With these Triggers I can control iTunes without changing from whatever I'm doing. Have a favorite song? Create a trigger that plays it.

The plug-in also adds a preferences pane in Quicksilver with some iTunes specific options. Monitor Recent Tracks and Display Track Notifications will pop up a small window when the playing song changes that shows the name, artist and album playing. I keep iTunes hidden and use the notifications to see the titles of songs as they change. If you want to use Growl for notifications you can install the Growl plug-in and select it in the Handlers Preferences pane. I only want iTunes notifications



and not others such as for email, so I don't use Growl and Quicksilver's iTunes notifications is a perfect Growl substitute for me. For a more manual approach the iTunes plug-in also adds a catalog entry for Show Playing Track. I have this as a trigger with the **Run** action and can see the song info mid-song (sometimes I've forgotten or missed the automatic notification at the beginning of the song). You need Display Track Notifications enabled for Show Playing Track to work.

If the Show Artwork option is checked then Quicksilver will look in the song file for album art and display it in the command window. If notifications are enabled they will show the album art as well. If the art isn't displaying for mp3 files that you know have art, open iTunes, select the track, and try converting the ID3 tags to the latest version with the command in the Advanced menu. (One person reported that he had to convert before adding artwork to the file and if the art was already there he had to remove it, convert and then re-add the art). The new feature in iTunes 7 to download artwork stores it in a new database (not inside the actual MP3 or AAC files) and as of B51 Quicksilver's iTunes Module can't display it.

If you don't have art in the song file you can have Quicksilver use one of the online art services to find the album cover of the playing song for notifications. The plug-ins that do this are: Clutter Artwork, Synergy Artwork, Sofa Artwork and Music ARTchive Artwork. Obviously if the services are unavailable these plug-ins won't work.

The Fast Browser Play option chooses albums, genres, etc. using a faster but less accurate algorithm. It may add tracks with similar names. The Group Compilations option will create a Compilations object to collect those songs marked as being in a compilation. The Show ~Unknown Entries option does TODO.

If you have a large iTunes library you might find that Quicksilver startup is slow. If this is the case consider disabling the iTunes Playlists catalog source. The plug-in is still useful as you can navigate your library (even the playlists) by typing → and you can still use the Control Scripts and notifications.

Contacts

Address Book

If you use the Apple Address Book you'll love this. For this section you'll need to make sure you have the Apple Address Book module installed in the Plug-ins Preferences pane. There are no configuration options in the Plug-in. In the catalog under modules you'll find Apple Address Book listed with one sub-entry called "Address Book Contacts".

Activate Quicksilver and start typing the name of someone in your Address Book, their contact will show up in the object pane. You can enter their first name, last name, initials, or nickname (if you have entered it in Address Book). If you have a picture of them in Address Book it will appear in the object pane. I find the pictures help me in differentiating various people with similar names (like various Johns) and making sure I'm choosing the one I want when sending mail.

The Address Book Specific actions are **Show Contact**, **Edit Contact**, and **Add Note...** The first two open the person's entry in Address Book in view and edit mode respectively. The **Add Note...** action takes a text object argument in the third pane which is appended to the note of the contact.



If you have a contact in the object pane, typing → will show a results list with fields of the contact including phone numbers, email addresses, IM accounts, URLs, addresses, and the note. Some items are treated like text. Try bringing up a phone number and using the **Large Type** action. You'll see the phone number in a giant font, easily seen across the room. The **Speak Text** action will have the OS X Speech system say the number so you can hear it across the room. The **Type Text** action will add it to the current app, much like the **Paste** action will. **Paste**

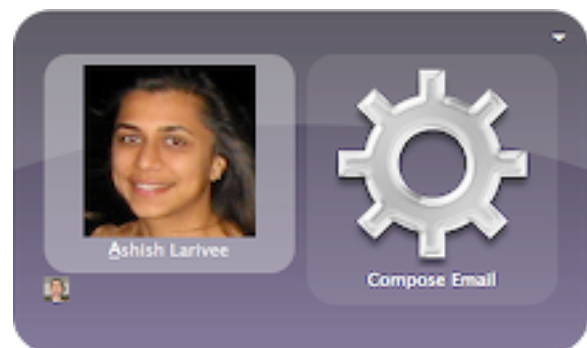
puts a copy on the clipboard, **Type Text** does not. This is very useful if someone IMs you asking for someone else's phone number. Just find them in Quicksilver, select the phone number and use the **Paste** action and type `return`. Or use one of the **Email to...** actions to send the phone number to someone via email. You may find the **Search with Google** action useful, particularly for a company's phone number. It's available via the Services Menu Module plug-in as a service from Safari.



The Phone Actions plug-in adds a **Dial With...** action. It opens a third pane with two choices: Modem and Speaker. The first has the computer's modem dial the number, the second has the computer's speaker play the touch tones of the phone number which might be enough to dial a phone held up to the speaker. (Unfortunately there's a bug in B50 so it will only dial by speaker once, to dial another number you'll need to restart Quicksilver). There's a Phone Preferences pane with various options for dialing area codes prefixes, etc. With the Bluetooth Module Plug-in the **Dial With...** action has a third choice, Bluetooth. This should allow a bluetooth connected phone to be dialed from Quicksilver.

Email addresses in contacts work similarly to phone numbers but of course the most common actions will be sending email. If you have a contact in the object pane, the **Compose Email** action will start a mail message in the default mail application to the primary email address of the contact. To send to a non-primary address, select it via `→` and choose the appropriate action. If you have someone you send email to often, consider, configuring a trigger for the email address and **Compose Email** action. Mail.app. has an annoying deficiency in that you can't type an Address Book nickname to select someone's email address. What's the point of entering a nickname if you can't use it? Remember that Quicksilver indexes the contact's nickname. So you can activate Quicksilver, enter a nickname until the contact object appears, and then use **Compose Email** to start a properly addressed email message. No need to wait for Apple to add a "feature" to Mail.app.

If you want to send a message to several people, use [the comma trick](#). Select the first person in the object pane, type `,` then type to select the second person. Repeat for each addressee. You'll see a small icon below the object pane for each of the previous addresses. Choose the **Compose Email** action (or one of the other email actions) and a message will be opened, addressed to all the listed people.



In addition to just starting a blank message in the default mail program you can use one of the **Email Item...(Compose)**, **Email Item...(Send)**, or **Email Item...(Send Directly)** actions. These are the reverse of the **Email to...** actions described in [Files and Folders](#) above. These all work on an object which is an email address or a contact and take an argument that's either text or a file. If it's text, it's used as both the subject and the body of the message unless it's of the form `subject>>message body`. In the Email Options Preference pane you can select the default message text and reply address to use for the generated message. If you include `%@` in your template it will be replaced with the name of the file (unless you send more than one file using the comma trick, in which case it will say

some variation of “# Files in Folder” such as “2 Files on Desktop” or “2 [PDF Document] in Documents”

Of course you might also want to paste an email address into an IM or email message you're composing and you do that with the **Paste** action, just like you did for a phone number. The **Spotlight in Window** action will do a Spotlight search for the email address which will show all the email messages sent to or from that address as well as documents and other things related to the address.

Instant Message addresses function like email addresses. Yes all the text actions work but there are also two IM actions you'll use. **IM** opens an IM chat window to the address. **IM Item...** is similar but opens a third pane to let you select a file to send. Remember that your IM application can be configured in the Handlers Preference pane based on Plug-ins installed. There are plug-ins for iChat, and Adium.

Postal addresses work similarly to the other fields. They don't have any commands of their own, they act like text. However here's one of my favorite uses of Quicksilver. First I have a custom web search configured for Google Maps with an abbreviation called “map”. See [Web Searches](#) for how to configure this. Select a contact in the object pane, → to their contact info, select a postal address. tab to the action field and choose




Find With... (usually fw works). tab to the third field and select map and type return. The browser appears showing a Google Map of the address. You can reverse this too, selecting the map search in the object pane, choose the **Search For...** action and in the third pane select the contact and then type → to select the postal address.

URL values work as text, but the default action is (usefully) **Open URL**. The Note value also acts as text. Probably you'll want to use the **Append to...** or maybe the **Prepend to...** actions with it. Both open a third pane to select what to add. Though rather than **Append to...** it's probably easier to just select the contact and use **Add Note...**, that avoids the → and selection of the note field. Note that the **Change to...** action is available for text lines and not plain text, so it's not available for a note value.

There doesn't seem to be any support for Address Book groups in Quicksilver. You can't select a group as an object. Use the comma trick defined above to select several contacts. If you want it to be repeatable then create a trigger using the comma trick. Of course it would be much easier if Quicksilver knew about Address Book Groups.

Entourage

The Entourage 2004 Module Plug-in provides similar but more limited support for contacts stored in Entourage. Install the plug-in and start Entourage and bring up the preferences. Under General Preferences, Security you'll see an item "Warn before allowing an external application to access the address book", make sure it is unchecked. Now open the Quicksilver Catalog and under Modules select Entourage Contacts and click the  (Rescan Source) button at the bot-

tom, also make sure this catalog entry is checked. The Entourage contacts should now be in the catalog, though it does seem to be a bit buggy. If your contacts don't appear, rescan the catalog.

Activate Quicksilver and start typing the name of a contact, it should appear in the first pane. There is a **Show Contact** action that will bring it up in Entourage. Other actions that do Email and IM should work as well. At least in B50 you can't type → to see the details of the contact.

Now Contact

The Now Contact Module plug-in allows Quicksilver to scan for contacts from the Now Contact address book. Contacts found can be used as described above, in fact the Now Contact Module adds no actions of its own. Now Contact must be running for Quicksilver to index it. As of B51 there have been reports that this plug-in doesn't work very well or forgets about contacts it has scanned. If you run into problems check the Quicksilver forums for help.

Mail

Quicksilver supports several mail clients including Mail.app, Gmail, Entourage, and Mailsmith. In the preferences under General, Handlers, you can select the default email client you want to use for various Email actions. These email actions are in the Email Support plug-in and are described above in Files and Folders and in Contacts but we'll go over them here as well, in some more detail.

To send email you use the **Compose Email**, **Email Item...** and **Email To...** actions. If you bring up a contact or email address in the object pane you can use **Compose Email** to open a new composition window in the default email client. The **Email Item...** and **Email To...** actions come in three variants each. They are: **Email Item... (Compose)**, **Email Item... (Send)**, and **Email Item... (Send Directly)**, and there are similar ones for **Email To...**. The **Email Item...** actions use a contact or email address in the first pane and a file or text to send in third pane. If text is entered it's used as both the Subject and body of the message unless it's of the form *subject>>message body*. The **Email To...** actions are the reverse, a file or text in the first pane, and a contact or email address in the third. The **Compose** variant uses the default email client and opens a message compose window for you to fill out. The **Send** variant also uses the default email client but doesn't open a window, it sends the mail automatically. Both the **Compose** and **Send** variants will start the default email client if it's not running. The **Send Directly** variant doesn't use the default email client to send the message but does so directly from Quicksilver, as a result the default email client doesn't need to be running. Also if your email client keeps a Sent folder (like Mail.app does) the outgoing message is not stored there. The **Send Directly** variant does get the SMTP configuration from the first mail account listed in Mail.app, so make sure that's setup correctly. In the Email Options Preference pane you can select the default message text and reply address to use for the generated message. If you include %@ in your template it will be replaced with the name of the file (unless you send more than one file using [the comma trick](#), in which case it will say some variation of "# Files in Folder" such as "2 Files on Desktop" or "2 [PDF Document] in Documents").

Mail.app

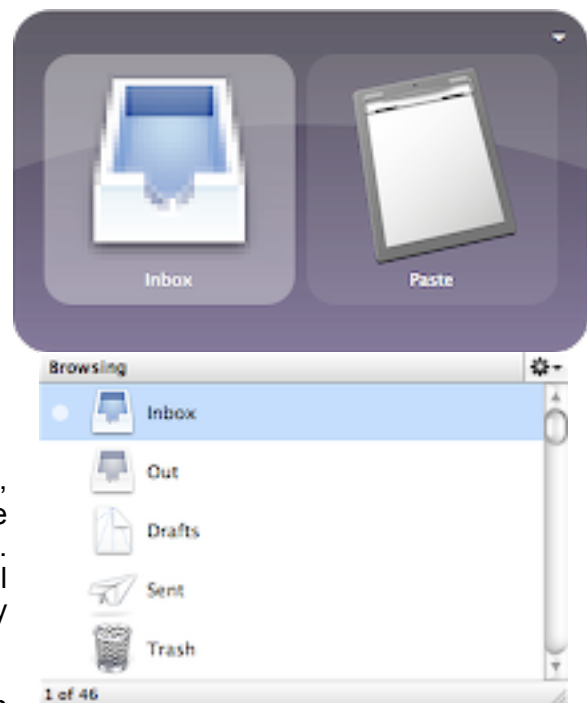
For this section you'll need the Apple Mail Module plug-in installed. This installs actions to manipulate Apple Mail mailboxes and messages. It also lets you browse through your mailboxes in a results list. It installs a handler to select Apple Mail as the program to handle some mail actions. Note that the actions to send files and text as email are part of another plug-in called Email Support which is installed by default. For the various **Email To...** and **Email Item...** actions

to work, you need to make sure that the first mail account has it's Outgoing Mail Server (SMTP) configured correctly to send mail.

Activate Quicksilver and select Mail.app. Since it's an application the default action is to open it. You'll notice there is a > in the results list next to Mail.app. If you have Advanced Features enabled, and you type → the object pane changes to inbox and the results list shows all your local mailboxes, e.g., Inbox, Sent, and Trash as well as all custom ones you have created to store your messages. IMAP mailboxes won't appear. To use them in Quicksilver add them to the Catalog by dragging them from Mail.app into the Catalog (e.g., under Custom). They won't appear when you → into Mail.app but they are selectable in the object pane.

Even though there is no > next to the mailboxes, you can type → to move into the mailbox and see a results list of all the messages in that mailbox. The messages are listed in reverse chronological order, ie., newest messages on top, for easy browsing.

The most common actions you'll want to use on message objects are installed by the Apple Mail Module plug-in: **Open**, **Delete**, and **Move to Mailbox....** They all do what you'd expect. For a mailbox object the only Mail action available is **Open**. Do you have a mailbox you access often? Create a trigger, select Mail and then → in the first pane to find your mailbox, then select **Open** as the action and bind this to the HotKey of your choice. If you just open the Mail app, it opens to whatever the last mailbox you were looking at, so I have ^⌘i set to open my inbox.



There are two actions that are only available if Mail.app is in the object pane. One is **Get New Mail** which will get new mail in all accounts. If Mail is not running it will be started. If it is hidden it will still get new mail, but Mail will remain hidden. If you have Mail hidden and want to check mail you can do so and just listen to the beep if there is no mail. The other action available on Mail.app is **Open New Mail**. If there is no new mail this will just beep. If there is a new message it will open in a new window, same as if you double clicked on it in the subject list in Mail.app.

The Apple Mail Module plug-in also adds a catalog source under Modules called Recent Mail Addresses. This is a list of email address you've sent mail to that are not in the Address Book that is kept by Mail.app. You can see the list in a dialog box in Mail.app by choosing Previous Recipients from the Window menu.

Gmail

If Mail.app isn't for you, don't worry, you have choices. There is a Gmail Module plug-in which adds a new mail handler in the Handler Preferences pane. There's no other configuration. You might need to install the Apple Mail Module plug-in to create the mail handler option, the Gmail plug-in can't create it, it can just add a new choice to the list.

With the handler set to Gmail then the **Email To...** actions use Gmail rather than Mail.app to compose messages. Type some text in the object pane, choose **Email To...(Compose)** and then choose one or more contacts or email addresses in the third pane. The browser opens on a Gmail compose mail page, with the addresses filled in the To: field and the text you entered as the Subject. **Email To...(Send)** will work too, if you have gmail configured as an account in Mail.app.

Unfortunately as of version 3C of the plugin that is the only integration. There's no way to read your mail via Gmail or browse your mailboxes as there is with Mail.app. There's also no way to use one of the **Email Item...** actions to send an attachment with Gmail.

Entourage

Install the Entourage 2004 Module plug-in according to directions in the [Entourage section under Contacts](#). In the General preferences under Handlers choose Microsoft Entourage for Email. Now the Email actions will use Entourage for sending mail. There isn't any other mail support provided by the plug-in.

MailSmith

The MailSmith Module plug-in adds a Mailsmith option for the Email handler (so the Email actions will use it) but doesn't add any other support.

Other Mail Applications

TODO: Well, Thunderbird doesn't support AAB yet, but Quicksilver does, and Quicksilver has (*SOME*) Thunderbird support. Set Thunderbird as your default e-mail app, type ^-Space, type in a name from your address book, tab over, type "Compose," and Thunderbird will pop up in a compose window with that e-mail address.

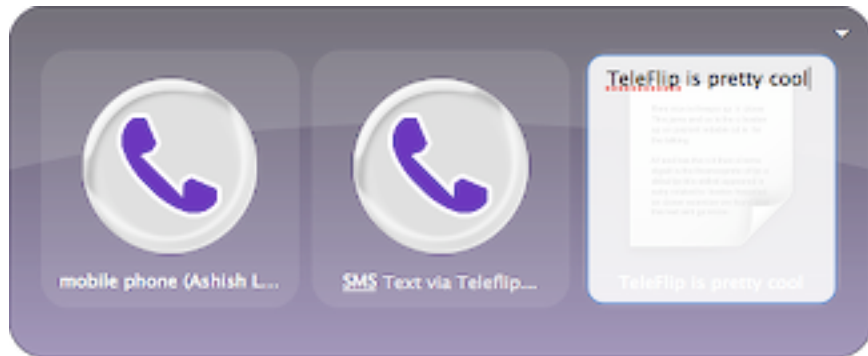
The comma trick works too. Cntrl+Space, name, comma, name, comma, name, comma, ..., tab, compose, and Thunderbird will pop up an e-mail addressed to each of those address book entries.

Well, technically the Quicksilver thing works just because you can assign Thunderbird as the default handler ("protocol helper") for "mailto:". Quicksilver sends a "mailto:blah@blah.com" to the operating system, the operating system looks up "mailto:" and calls up Thunderbird. So it's really "Default e-mail client support."

Teleflip

[Teleflip](#) is an online service that allows you to send SMS messages to a United States cell phone by sending an email message. Currently the service is free up to 100 messages a month. Send email to an address of the form `###-###-####@teleflip.com` filling in the cell phone number. The subject and body are sent, there is a 160 character limit. The Teleflip Module plug-in

makes it easier to send such messages from Quicksilver by constructing the address from a contact's phone number. It installs two actions. With a phone number selected as an object use the **SMS Text via Teleflip...** action to send the a text message argument via Teleflip. The reverse action is **SMS via Teleflip...**, use it to send text in the first pane to a phone number you select in the third pane. Note that a phone number object can also be treated as text, so the **SMS via Teleflip...** action appears for phone numbers too (it would send the number as a message. If this is what you want to do it can be very convenient, bring up a phone number, type `.` to enter text mode and add more text if you want. Since this is probably less common, make sure **SMS Text via Teleflip...** is above (i.e., ranked higher than) **SMS via Teleflip...** in the action preferences.



Chat

The basic actions, **IM**, **IM Item...**, and **IM to Account...** are all part of the Chat Support default plug-in. There is a Handler called Instant Messaging which defines which application to use for the basic actions. Currently there is support for iChat and Adium, each via their own plug-ins.

The **IM** and **IM Item...** actions work on an IM address object (or contact). **IM** opens a chat window to the contact that you can type in. **IM Item...** opens a 3rd pane where you can select a file to send via IM or type `.` to enter text mode and type text to send as a message. **IM to Account...** is the reverse of this, you select a file or text in the first pane and the IM account or contact in the 3rd.

Personally I keep a small buddy list of just the people I chat with often and care if they are on-line. For other people I keep their IM addresses in Address Book and if I want to IM them I can bring up their contact or one of their IM accounts and use the **IM** action. Conversely if you had a large buddy list, then Quicksilver's matching algorithm would be a great way to search through it quickly to find the buddy you were looking for.

iChat

With the iChat Module plug-in installed if you select iChat.app in the object pane you'll see a `>` in the results list and you can type `→` to see all the buddies currently online. If you select one you can then choose **IM** or **IM Item...** as an action. There are some other actions available but remember your object isn't a contact, it's an IM account. iChat also has one action specific to it, **Set iChat Status** which uses the text of the object pane as the away message (although in version 0.5.0 of the plug-in this seems buggy).

TODO **Audio Chat, Video Chat**

Adium

With the Adium Module plug-in installed if you select Adium.app in the object pane you'll see a > in the results list and you can type → to see a results list of all groups with buddies currently online. Another → into one of them will show a list of buddies in that group, which you can IM. It's similar to iChat but with an extra level.

The Adium plugin (as of B51) only works with Adium 0.89.1, it does not work with Adium 1.x. It can also be a little tricky to install if you're upgrading from an older version. It's actually two plug-ins (one for Quicksilver and one for Adium) that communicate with each other. Both parts come with the Quicksilver plug-in and when you run it the first time it installs the Adium part as ~/Library/Application Support/Adium 2.0/PlugIns/Quicksilver.AdiumPlugin. If the plug-in isn't working it's probably an issue with the Adium part. To reinstall it do the following:

1. Quit QS and Adium
2. Open Terminal and run the command:

```
defaults delete com.blacktree.Quicksilver AdiumPluginInstalled
```
3. Remove ~/Library/Application Support/Adium 2.0/PlugIns/Quicksilver.AdiumPlugin. Note that the PlugIns directory might have a lowercase i, if so remove any Quicksilver.AdiumPlugin from there too.
4. Double-click on ~/Library/Application Support/Quicksilver/PlugIns/Adium Module.qsplugin

Calendar

Quicksilver has some limited support for creating calendar events and to-do's in iCal and Google Calendar. As of B50 the Entourage Module plug-in only supports contacts and email but not calendars.

If you're more comfortable with using text files than iCal or Google Calendar, see the Text section for the description of the [Text Manipulation Actions plug-in](#) and using it to manage a to-do.txt file.

iCal

For this section you'll need to enable Advanced features and install the iCal Module plug-in. At this point the features enabled by the plug-in are limited. You cannot type → into the iCal.app and there's no way to see or interact with existing events and to-do's. You can create new ones using the two installed actions **Create iCal Event** and **Create iCal To-Do**. Using either of these actions will open iCal if it's not already running though won't open the window if it's hidden or closed.

To create a new to-do, activate Quicksilver, enter text-mode by typing `.` or `'` and enter the text of your to-do. Type `tab` and choose the **Create iCal To-Do** action, type `tab` and choose the calendar you want the to-do to be put in. The format of the to-do text allows you to also enter a priority and a due date. To enter a priority



precede the text with one to three exclamation marks representing the number of priority marks in iCal. So no `!`'s sets a priority of None, `1 !` is Not important, `2 !`'s is Important, and `3 !`'s is Very important. If you precede the text with a date and two hyphens you'll set a due date. The date is somewhat flexible, allowing `8/1`, `8/1/2006`, `tue`, and `next wed`. In the example image I use the text: `8/1--!Upload User Manual`.

To create a new event do the same thing except use the **Create iCal Event** action. The format of the text string adds a time and removes the priority, so: `date-time--text`. If you leave out the date and time the event is created now. If you give just a date, the event is created at noon on the specified day. Times are specified as `1pm`, `1:00` (which is am), or `1:05pm`, or `22:30`. You cannot use `p` for `pm`. All created events are 1 hour long. The created events use the whole string as the event name, so if you enter `wed 6pm -- dinner` that's also the name of the event. It seems that if you don't use `--` you can put the date and time anywhere in the string and it will work. E.g., `dinner - wed 6pm` or `wed dinner 6pm` and the event is created at the right time and with the string at the front which is a little friendlier if it gets cropped in the iCal display.

With this plug-in there's no way to specify attendees, repeats, alarms or other iCal fields. There are a few scripts on the Quicksilver site that you can install that will prompt for more info but they are under-documented and a little complicated to get setup. Rather than enter event info into AppleScript dialog I think you're better off just entering the info into iCal. Try these commands (you'll need the User Interface Access plug-in installed):

- iCal (**Menu Bar Items...**) New Event
- iCal (**Menu Bar Items...**) New To Do

Making triggers for the above would be very convenient but as of B51 you can't make triggers with the **Menu Bar Items...** action and the third pane filled in. Hopefully this will be fixed in a future version.

If you want a little more flexibility with creating events, some people like the freeware OSX service [Calendar Creator](#). This installs two Services, Add Calendar Event and Add To Do. See Calendar Creator's documentation for what it can parse. With the Services Menu Module Quicksilver plug-in installed you can send any text to these services via Quicksilver.

Google Calendar Module

If you eschew iCal and prefer Google's Calendar, try the Google Calendar Module plug-in. It adds one action **Google Calendar Event** that takes text as the object in the first pane and sends it to Google Calendar's Quick Add function that can parse text like "Dinner with Michael 7pm tomorrow".

If needed, Quicksilver will prompt you for info to login to your Google Calendar. If you have problems logging in, check your keychain entry for `calendar.google.com` to make sure it's correct.



TODO: The Google Calendar Module works great for me with just one hitch -- the events created via the module don't inherit my default reminder. Instead, the reminder is set to "No reminder", but, curiously, the reminder icon (the little alarm clock) does appear next to the event in the Day and Week views.

Web

Browsers

Quicksilver integrates with several web browsers. You'll need a different plug-in for each family of browser. There are four: Safari, Firefox, Camino, and OmniWeb. None of them add actions to Quicksilver, they just add items to the catalog. All add bookmarks. The Safari, Firefox, and OmniWeb ones add history items as well. I find this remarkably useful. If you configure the browser to save a few hundred items in history, you have easy access via Quicksilver to everything you've surfed for the past few days. If you remember part of the title of a page, Quicksilver's matching algorithm will make finding it a snap.

If you bring up Safari in the object pane you can type `→` to see all the Collections listed in the left pane when you Show All Bookmarks in Safari. You can then `→` into them. This works if Safari is running or not. You cannot `→` into Firefox. You can `→` into Camino but it shows all the bookmarks in a flat results list.

If you have a URL in the object pane you can use the **Open URL** action to open it in the default browser or the **Open URL With...** action to pick a specific browser to use. There are several ways to get a URL into the object pane. You can type `.` to enter text mode and type it in. You can use `⌘V` to paste a URL from the Clipboard. If you select text in a Cocoa application you can type `⌘-esc` to activate Quicksilver and automatically paste the URL into the object pane. The most common way is to just type the name of a bookmark to select it in object pane.



I have a trigger defined that I use all the time. It's Current Web Page (**Paste**) and I have it on `^⌘F`. It pastes the URL of the active tab in Safari into the current application. I use this to paste URLs into emails, IMs, and html documents. I also use it to put URLs into iCal events and Address Book contacts. The great thing is you don't need to leave the current app at all. Without this trigger I'd need to go to Safari, select the URL, `⌘C` and then go back to my app and use `⌘V`. This is one of my favorite triggers. Note that the Current Web Page proxy object is poorly named as it only works for Safari's current web page.

If you use Camino occasionally but don't have it as your default browser then you might find the **Open URL in Camino** action a little faster than **Open URL With....** This is enabled when you install the Service Menu Module plug-in and turns the system service installed by Camino into a Quicksilver action.



If you want bookmarks and URLs opened from Quicksilver to appear in their own tab you need to configure this in the browser. In Safari, open Preferences and in the General tab for "Open links from applications" choose "in a new tab in the current window". In Firefox goto Preferences, Tab Browsing and for "Load external links in" choose "New Tab". In Camino open Preferences, Tabs, for "Link from other application" choose "Opens in a new tab in the frontmost window". You can't add bookmarks to a browser via Quicksilver (well maybe with an AppleScript).

	Safari	Firefox	Camino	OmniWeb
Bookmarks	Yes	Yes	Yes	Yes
History	Yes	Yes	No	Yes
→	Yes, collections	No	Yes, flat	?
Proxy Object	Current Web Page	None	None	OmniWeb Active Page
Other			Open URL in Camino action via Services Menu Module	

Online Services

The Social Bookmarks plugin (which replaced the del.icio.us Bookmarks plug-in) makes your del.icio.us and/or ma.gnolia bookmarks available in the Catalog. Instead of an entry appearing under Modules, you must create a custom source for your del.icio.us or ma.gnolia account. In the Catalog, click on the + button and choose Social Bookmarks. Select the newly created entry and click on the  button to open the drawer. Under Source Options choose the site and enter the Name and Password of your account. Note the plug-in only works with passwords that are 8 characters or less. Click the  (Rescan Source) button and you should see your bookmarks appear. If you select the option to Include Tags then your tags are in the catalog as well. You can bring them up as an object and type → to get a results list of all your bookmarks with that tag. To add a bookmark to your del.icio.us or ma.gnolia account use your normal means. If you use a post to del.icio.us bookmarklet in your browser and it's in the catalog, you can bring it up in Quicksilver as an object and use the Run Javascript action.

If you have bookmarks in several sources (e.g., Safari, Firefox and del.icio.us) you may want to create a custom catalog folder and put them all together (for safari bookmarks you need to create a copy of the preset - go to the attributes tab of the info drawer for the button). Now you just browse into the catalog entry for the Bookmarks folder (it will have a "(catalog)" after it's name) and with the **Search Contents** action you can search all your bookmarks at once.

Backpack is an online service that lets you create and maintain your own web pages. The Backpack Module plug-in lets Quicksilver index those pages in the catalog. It works similarly to the del.icio.us Bookmarks plug-in in that it creates a new type of catalog source option. In the Catalog, click on the + button and choose Backpack Pages. Select the newly created entry and click on the  button to open the drawer. Under Source Options enter your Backpack Name and API Key. Click the  (Rescan Source) button and you should see your pages appear. If you select the Backpack Pages (Catalog) object you can type → to see a results list showing all your Backpack pages. There's an oddly named action **QSBackpackOpenURLAction** which should open the page in a browser. The Backpack Module plug-in does not support https so if you use it you'll have to disable it in the Backpack Settings page.

Backpack pages have an email address enabling you to send additions to the page via email. You can upload images or files by just sending them to the address. With specially formatted subjects and message bodies you can add notes and to-dos as well. You can use the regular Quicksilver **Email To...** and **Email Item...** actions to perform these additions, Quicksilver will automatically convert the Backpack page to its email address for you. E.g., you can bring up an image in the object pane, choose the **Email To... (Send Directly)** action and select the Backpack page as the argument in the third pane. To add a note to a page, select the page, choose the **Email item...(Send Directly)** action and enter text-mode for the argument. Enter the note as follows: `note:note title>>note body`, which uses the `subject>>message body` syntax that the Email actions understand. Of course Quicksilver will let you work the way you want, with the backpage page in the first or third pane depending on which action you choose to use.

Have you ever wanted to post long text or a file, such as command output or error messages to an irc channel, IM chat or forum where it was too long to do so? cl1p.net is a free service that lets you temporarily post any text to their own web pages, you can then just post the URL to the forum or wherever. The Cl1p Upload plug-in lets you upload text or files from Quicksilver to cl1p.net. Select the file or enter text into the object pane and choose the **Post to Cl1p...** action. If you type `return` it will be uploaded to an anonymous cl1p page, if you tab to the third pane you can enter a page name in text-mode. After the command executes your browser will open to the newly created page. To save the page you'll need to click the Save button.

Web Pages

Here's another hidden feature in Quicksilver. If the object is a web page, (i.e., a bookmark, URL, or HTML page in the filesystem) you can type `→` to fill an object pane results list with all the links on that page. No there isn't a `>` in the results list but it still works. It can be a little slow if it has to get the page from a remote machine, particularly if the page is large. While this is happening, depending on the interface you'll see a "working" indication. You can supposedly stop a lookup that's in progress by typing `⌘`. (thats a period) but in B51 I haven't seen this work too often.

For each link in the page there are two lines shown. The first is the text of the link and the second is the URL. If the link is on an image, there is no title but instead a very scaled down version of the image is shown.

I'm not sure how useful it actually is. In the example at right I'm browsing the Quicksilver [online documentation](#) I can get to links marked Tutorials and Overview, but there is a lot of other stuff in there and it's a little awkward to browse. Still if you know you're looking for Tutorials you can just type it in and Quicksilver will match it. If you wanted to send someone the URLs of an FAQ you could open the site and type `→` to navigate into it, type `⌘q` and then `⌘a` to choose the **Paste** action. This seems like a very cool feature in search of a good use case.



Web Searches

For this section you'll need to make sure you have the Web Search module installed in the Plugins Preferences pane. The plug-in has no configuration options. In the catalog under modules you'll find "Web Searches (from docs.blacktree.com)" listed. Check the box to add all the listed web searches (currently 387 of them, so it may take a while) to your catalog. The list of web searches is also at this [web page](#). Also in the Add Custom Source drop down you'll find an option to add your own custom web searches to the catalog.

To perform a Web Search you select the search by name in the object pane (e.g., "Quicksilver" or "google"). The action should default to **Search For...**, so type `return` (or `⌘` twice) to get to the third pane to type your search terms. The third pane will be in text mode and will default to text in the OS X Shared Find Clipboard. This is a system-wide alternative clipboard that many applications populate with text entered in search fields. You can explicitly put something in the Shared Find Clipboard by selecting text and typing `⌘⌘E` in those applications that support it. Type `return` and your browser pops up with the results. If **Search For...** isn't the default action, make it be so by finding it in the results list (while in the action pane) and `⌘`-clicking (or right-clicking) and choosing "Make Default".

The Web Search Module actually adds three actions to Quicksilver: **Search For...**, **Find With...** and **Show Results for Search...**. The first is described above, put the search site in the first

pane and the query string in the third. The **Find With...** action is its reverse, you put your query string in the first pane, choose the **Find With...** action and then choose your search in the third pane. The third action, **Show Results for Search...** is similar to **Search For...** but instead of opening the resulting page in the browser, it opens it in Quicksilver, showing all the links on the page in an object pane results list of a new command window (as described above in [Web Pages](#)). Unfortunately in B51, Show Results for Search... seems to be broken, it hangs when run and it is disabled by default in the Actions preferences.

I found I used some searches ("Google", "Wikipedia", "Amazon", and "IMDB") very often and I wanted to get to them quickly, with just a single letter. You could type the letter you want to use, find the item in the results list and ^-click (or right-click) and choose "Set as Default for *letter*". I found for Amazon I wanted to use the letter z but it was further down in the results list than I was willing to scroll, so I found another way to do it (there's always another way in Quicksilver). First I brought up the Amazon search in the object pane. Then instead of using the default action I chose the **Assign Abbreviation** action and in the third pane entered z. Now I have one letter access to all my frequent searches.

Now that you can get to your searches quickly, what if you want to search several search engines for the same thing? Use [the comma trick](#). Select your search (e.g., Google), type comma, select another search (e.g., Wikipedia) and continue for all the searches you want. Then tab to the second pane and select the **Search For...** action and then tab to the third to type your query. Type return to have all the searches open at one in different tabs in your browser. This also works if you use the **Find With...** action and use the comma trick in the third pane.

If you want to search really fast, create a trigger for your search. I have ^⌘M configured to be "imdb (**Search For...**)". I type it, enter the search terms, type return. I also use ^⌘G for searching Google and ^⌘W for searching Wikipedia and ^⌘Z for searching Amazon. Once you do this you'll wonder how you lived without it.

But you can make it even faster. Create a trigger of the proxy object Current Selection, **Find With...**, wikipedia web search. Bind it to a key like ⌘^⌘W, select some text, and then the key stroke. Any search you want with one key. Now this trigger can be tricky to create in Quicksilver. First activate Quicksilver, type . to go into text mode, type some text, tab to the action pane and select **Find With...**. Now while the second pane is active type ⌘C to copy the **Find With...** action (yes you can do that!). Then you want to select some text in some application before defining your trigger, otherwise selecting the proxy might go into a beeping loop (if it does, just wait for 8 beeps and hopefully Quicksilver doesn't crash). Go to the trigger pane, create a new trigger, select the proxy object Current Selection as the object, tab to the action pane and type ⌘V to paste the **Find With...** action (otherwise it doesn't come up because the selection might not be text). tab to the argument pane and choose your wikipedia web search. Click Save and assign the key of your choice.

Many browsers have similar functionality. Some have a drop down in the search box to let you choose from various search engines. Some use keywords that you type in the address box before your search terms which chooses your search engine. Safari does not have this functionality built-in but several add-ons exist. If you use Acid Search, SafariStand, or Sogudi you'll want to use the Quicksilver plug-ins to have access to their searches. The OmniWeb plug-in also enables web searches based on OmniWeb shortcuts. But once you use these from Quicksilver you'll find opening your browser, clicking in the search box, and using a drop down to choose a search type will seem painfully slow. An advantage of using Quicksilver web searches for this is that the web search list is independent of the browser. If you change browsers you don't have to worry about migrating your keyword searches, which sometimes doesn't go smoothly.

Quicksilver has a few shortcuts for getting information into it. If you use the trigger Command Window With Selection (commonly bound to ⌘-esc) it activates Quicksilver putting the selection from the current application in the object pane ready for the **Find With...** action. I use this a lot. For example, I select an address in a web page or mail message, type ⌘-esc, fw (to choose **Find With...**), return, map, return, and my browser opens with a Google Map of the address. I also have a Netflix web search configured, when someone sends me a movie recommendation in email I select it, type ⌘-esc, fw return, ne return, and the movie shows up in Netflix and I click on the add button to put it in my queue.

AcidSearch, SafariStand, and Sogudi are extensions to Safari that (among other things) allow more web searches to be configured in Safari. The AcidSearch Queries, SafariStand Queries, and Sogudi Queries Quicksilver plug-ins allow these web searches to be used with the **Search For...** and **Find With...** actions. Some of these plug-ins haven't kept up with changes in the Safari extensions, if you have issues see the Quicksilver forums for help.

Adding Web Searches Manually

If you're not happy with the built-in web searches it's easy to add your own. The list of built-in web searches is at this [web page](#) and a link at the top of that page lets you add to that list. It's just a wiki page so anyone can add to it (so if you do, get the formatting correct).

I had a few searches that weren't in the Quicksilver list and I wanted to add them. Also I don't use the Quicksilver built-in list because I don't need 300 searches that I don't use cluttering up my catalog. At some point Quicksilver performance will improve but now I have about 5000 items in the catalog and try to keep the number down. Before I describe how to add your own, I want to describe how web searches work.

When you perform a search on a web page information is sent from your browser to the server. This is usually in the form of a URL with your search terms included in it, though the styles and formats vary considerably. Here are two examples of searching for "Salma Hayek", the first is for the Internet Movie Database and the second is for searching Google News.

<http://www.imdb.com/find?tt=on:nm=on:mx=20;q=Salma+Hayek>

<http://news.google.com/news?hl=en&edition=us&q=Salma+Hayek&btnG=Search+News>

They're different but you can see the search term "Salma+Hayek" in both of them. You might notice that the space in the search terms was converted to a +, this is how the web works. Web Searches work by changing the URL to include a special indicator for where Quicksilver should substitute the search terms (and do conversions like space into +). This indicator is three asterisks (%s works too). Here are the above URLs suitable for use as a web search:

http://www.imdb.com/find?tt=on:nm=on:mx=20;q=***

http://news.google.com/news?hl=en&edition=us&q=***&btnG=Search+News

An easy way to generate these is to go the web page of interest and search for *** and then look in the address bar of the results page and copy the URL. Some search pages hide this info, but if the URL looks similar to the above it's usable. To tell Quicksilver about it, go to the Catalog, add a new item (depending on the Quicksilver Preferences UI this is probably a + sign at the bottom of the catalog) and choose a new Web Search List (it might be ugly like "QSWeb-SearchSource"). Select the newly created folder and open the info drawer. In the Source Options click + to add a new item and then enter the URL and a name for it. For the above two ex-

amples I use the names “imdb” and “news”. Rescan the catalog source and you're all set. Activate Quicksilver, choose your item as the object, choose **Search For...** as the action, type your query in the third pane, and type `return`. Your browser pops up with the search results.

In addition, Quicksilver registers itself to handle URLs that begin with `qss-http:` and `qssp-http:`. If the URL begins with one of these they will open in Quicksilver instead of the browser. If it has three asterisks in it, it will be treated as a web search. The second variety (with the p) does the search using the HTTP POST method instead of the default GET; some web sites require this. It's best to convert all manually added web search to use one of the `qss-` style URLs but not required.

There's also another way to keep track of URLs and web searches in Quicksilver. You can add a custom Catalog source that uses the File & Folder scanner and select an `.html` file as the source. For the Include Contents popup choose HTML Links. Quicksilver will scan the `.html` file and find all the links and include them in the catalog. If you're adept at managing a file of HTML this method works well. Some find it less fragile because all your links are in a simple file that's easy to backup, rather than in Quicksilver's internal storage.

NetNewsWire

The NetNewsWire Module (β) plug-in works with the NetNewsWire newsreader (not the Lite edition). It adds a new catalog source under Modules called `QSPresetNNWSsubscriptions` which has one item called New Headlines. You can bring it up in the first pane and then type `→` to see a results list of the new headlines and use the **Open URL** action with. You can also just type `→` on the `NetNewsWire.app` itself and accomplish the same thing. Unfortunately as of B51 I find both of these unstable and it always crashes Quicksilver when I type `→`.

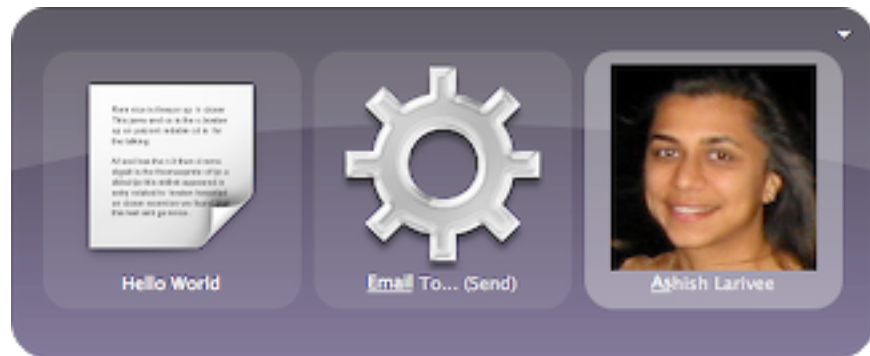
Text

Most of the time in Quicksilver you type into one of the panes to select some object on the computer like a file, a contact, a song, etc. Occasionally we've seen that you can type in arbitrary text into the third pane, such as when searching for something or renaming a file. You can also type arbitrary text into the object pane. Type a period or single quote to enter text mode. This is a regular Cocoa text field so all the text editing key bindings work. If you paste text into a text pane it will include the formatting of the source and this will stay in effect in future text boxes until you change it, either explicitly or by pasting in text with different formatting. Sometimes the text comes in center justified. If this bothers you like it bothers me you can use the standard OS X key binding `⌘{` to change it to be left justified. Finally a common way to bring text into Quicksilver is to use the Send to Quicksilver Service or the predefined triggers Command Window in Text Mode or Command Window with Selection, commonly bound to `⌘-esc` until Front Row started using that key binding. These work with Cocoa applications, unfortunately MS Office and Firefox are *not* Cocoa applications.

So what actions can you do on text? We've already seen a number that are the reverse order of other commands. From the Web Search Module plug-in we get the **Find With...** action which

lets you do web searches which you select in the third pane. From the Mail Support plug-in (installed by default) we get the three actions **Email To... (Compose)**, **Email To... (Send)**, and **Email To... (Send Directly)** which all send the text as Email to a contact you select in the third pane. The first opens a compose window in Mail.app, the second doesn't

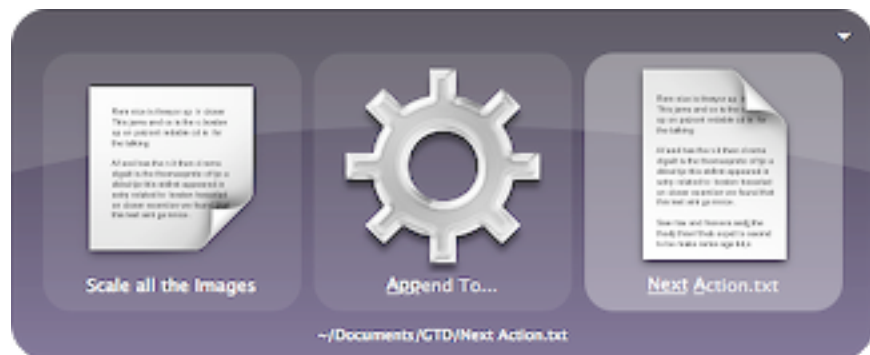
but uses Mail.app to send the message (starting it if needed), the third also doesn't open a window but sends it directly from Quicksilver and doesn't open Mail.app at all. All three need the first mail account in Mail.app to have its Outgoing Mail Server (SMTP) configured correctly to send mail.



IM to Account... sends an instant message to the user you specify in the third pane. With the iChat plug-in you can set your away message by using the **Set iChat Status** action. With the iCal plug-in you can use the text as a new event or to-do item. **Create iCal Event** asks in the third pane for a calendar to use and then creates a new event using the current date and time. **Create iCal To-Do** does the same thing for a new to-do, it's created with no priority or due date.

The **Create File...** action makes a new text file with your object text as the contents. It brings up a standard apple save dialog to get the file name and location, saves the file, and then shows a new command window with the new file as the object.

The Text Manipulation Actions plug-in lets you add text to text files. **Append To...** will append the object text to the end of the file selected in the third pane. **Prepend To...** does the same thing except at the beginning of the file. The reverse commands are **Append Text...** and **Prepend Text...** which want the file as the object and the text in the third pane.



The text entered in the first pane will be appended to the text file (Next Action.txt) specified in the third pane.

These are very useful if you want to keep a `to-do.txt` file or are a fan of the book *Getting Things Done*. I have a trigger defined for `NextAction.txt` (Append Text...). Whenever I think of stuff to do, I type `^⌘A` and type the idea into the third pane and continue with what I was doing. Then I review the file regularly to figure out what I want to do.

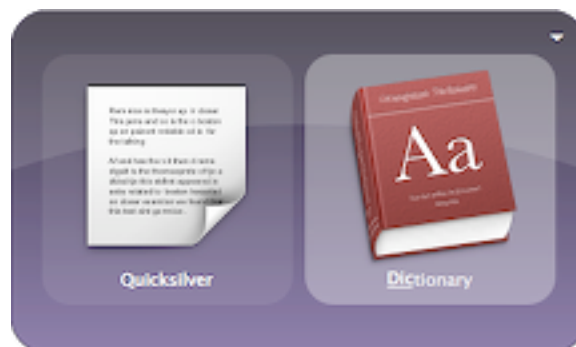
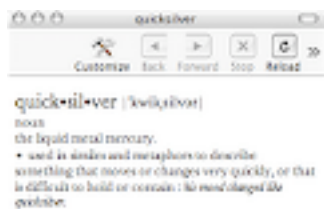
There are the common actions **Copy** and **Paste**. If you commonly paste the same text into something, like a signature file into a Mail message or a long phrase, you can create a trigger with the text and **Paste** action. There's also the **Type Text** action which unlike the **Paste** action, will not put a copy on the clipboard. The **Large Type** action will show the text in a giant dialog visible from across the room, if you click anywhere it will disappear. The **Display Dialog** action

uses a regular dialog with an OK button so you can leave it up until you want to dismiss it. The **Speak Text (Say)** action will use OS X's speech synthesizer to speak the text out loud. I couldn't think of too much of a use for this either. Then I thought it might be useful for pronouncing foreign words but my attempts at Japanese and French didn't work very well. It might be useful with the **Run after Delay...** action, see below.

If you're the programming type, **Run as AppleScript**, **Run Command in Shell**, and **Run a Text Command in Terminal** might appeal to you. All let you run a script from Quicksilver, just type or paste it into the object pane. The last two require the Terminal Module plug-in. The second just exec's the command. The third opens a new Terminal window to run the command in and leaves the window open. It would be a good way to run the unix utility top.

The Services Menu Module plug-in makes all your Services available in Quicksilver. Services are in the application's menu and many are only enabled while you are selecting text. Depending on the applications you use, some services can be really useful as actions. E.g., if you use Sticky Notes then you'll love the **Make New Sticky Note** action.

The Dictionary Module plug-in gives you access to the OS X built-in dictionary. Enter a word in the object pane and use either the **Dictionary** or **Thesaurus** actions to access OS X's dictionary. The Services Menu Module plug-in makes **Look Up in Dictionary** available and is another way to do the above. **Look Up in Dictionary** opens the Dictionary application with the full window. The **Dictionary** action shows a smaller panel and this seems unaffected by how you have your Dictionary preferences setup. I have a trigger for Current Selection (**Look Up in Dictionary**) bound to $\uparrow^{\wedge}\mathbb{D}$



for those applications (e.g., Preview) that don't support the Cocoa key binding of $\wedge\mathbb{D}$. **Define with dict.org** is similar but aggregates definitions from several web sites and displays them in a simple HTML viewer built-in to Quicksilver.

When Tiger (OS X 10.4) shipped with Spotlight there was a lot of confusion as to what it would mean for Quicksilver. As you've probably noticed by now the two are quite different. Quicksilver can't search inside files but can do much more than just open things. One of the biggest criticisms of Spotlight is its interface which does incremental queries as you type. It tends to slow things down a lot. Quicksilver solves that problem. Install the Spotlight plug-in and you'll find three actions for text objects: **Spotlight in Window**, **Spotlight in Finder**, and **Spotlight in Command**. **Spotlight in Window** runs the search in a Spotlight window. **Spotlight in Finder** runs it in a Finder window, like when editing a Finder smart folder. **Spotlight in Command** returns the results in an object pane results list in a new command window. None of these searches start until you've typed all the text in the object pane (and selected the action) so many people prefer this to the normal Spotlight interface. If you have a spotlight saved search, you can bring it into the first pane and use the **Run Search with Spotlight** action on it to run the search.

Here's a cute trick from the Text Manipulation Actions plug-in we already talked about. If you bring up a text file in the object pane, even though there is no > next to it in the results list, you

can still type `→` or `/` to move into the file, like you can with html files. What you get is a new results list with one text item for each line in the file. These all show an `>` next to them but you can't `→` into them further. However you can choose from any of the normal text actions as well as a couple more that work on these text line reference objects. The ominously named **Delete Line (UNSTABLE)** will remove the line from the file. Yes, you're cautioned about using it. Less risky is **Change To...** which takes new text in the third pane and will replace the text in the file with it. In addition, the **Prepend Text...** and **Append Text...** actions can work on text line references. Their argument is added as a new line before or after the line respectively. I had a problem with line-endings when using **Prepend Text...** on the first line in a file.

TODO: Split Results List on text files [RE: Preview for Text Files](#)

If you use BBEdit you'll want to install the BBEdit Module plug-in. It adds one action **Select in BBEdit**, which works on text lines, opening them in BBEdit. It also adds a catalog source under Modules called BBEdit Recent Documents but it doesn't seem to work.

Math Calculations

The Calculator Module plug-in adds two actions called **CalculatorCalculateAction** and a preference pane called Calculator. It allows you to enter math expressions as text into the object pane and have Quicksilver compute the result. By default it uses the unix utility `bc` to compute the expressions but if you prefer an RPN calculator you can change it to use `dc` in the preference pane. `bc` understands the basic math expression including addition with `+`, subtraction with `-`, multiplication with `*`, and division with `/`. You can use parentheses to specify the order the operations. There's much more it can do, to get the details look in the documentation, which for unix commands is a man page. In the Calculator Preference pane there is a button to bring up the man page in a Terminal window.

To use this, activate Quicksilver, type `.` to enter text mode and type an expression, e.g., `3+5`, type `tab` and choose the action **CalculatorCalculateAction**. Type `return` and the command window is replaced with a new one, where the object is result of the expression, in this case `8`. You can then choose another action like **Copy** or **Paste** or whatever. In the Calculator Preference pane you can change the behavior when the result is computed. Instead of opening a new command window you can have the result displayed with the **Large Type** action or displayed in a notification window like the iTunes module uses (or with Growl if you have that configured).

Since not all text you enter in Quicksilver are math expressions, the default action is probably something like **Copy** and not **CalculatorCalculateAction**. To make it easier to perform a quick calculation you can use `=` instead of `.` to enter text mode and it will make the default action be **CalculatorCalculateAction**.

You can also use the **Run as AppleScript** action to perform basic math calculations with `+`, `-`, `*`, and `/`. In fact, since text mode is a Cocoa field, you can use the Services menu in them. Script Editor has a service called Get Result of Apple Script and it's bound to `⌘*` by default. So you can activate Quicksilver, type `.` to enter text mode, type some simple expression, type `⌘A` to select it all and then type `⌘*` to have the expression replaced with the result.

The Unit Conversion Module plug-in allows you to do conversions using the unix units program. The plug-in adds one action **Convert to Units...** which works on a text object specifying a number and a unit and converts to the units specified in the argument. E.g., `10 m` **Convert to Units...** `ft`. The result is shown in a new command window. The the plug-in's documentation in the info pane or the `units(1)` man page for more details.

You can also use a google web search to do calculations and conversions. The **Search With Google** action available via the Services Menu Module plug-in and the service of Safari is a good default for text and is very convenient. If not, a trigger like $\text{^}\text{⌘}\text{G}$ for Google Search (Search For...) makes it easy. Google lists some of the expressions it understands at <http://www.google.com/help/calculator.html> though it also knows how to do conversions like 30C in F and understand scientific notation and other things. Of course you need a network connection to get to google and the above Quicksilver plug-ins work without one. Also the result is shown in the browser and not in a new Quicksilver command window.

Clipboard and Shelf

Clipboard History

For this section you'll need the Clipboard Module plug-in installed. It doesn't install any actions but does add several catalog objects in the Quicksilver Set. Inside Shelf & Clipboard you'll find two items, Shelf, and Clipboard History. Make sure the latter is enabled for this section. Under Internal Commands you'll find several related objects, some of which behave like actions but are scripts (such as Clip Store Copy 2) that you use with the **Run** action. Also under Proxy Objects there's a Clipboard Contents proxy object.

One limitation of copy and paste on the Mac is that the clipboard only remembers the last thing you copied. With Quicksilver's Clipboard Module you can extend this to remember the last several items you've copied. In the Clipboard Preference pane you can configure the number of items to be saved in the clipboard. A large number is useful but takes up more memory, particularly if you put large images on the clipboard.

You can bring up a window showing the Clipboard History by activating Quicksilver and typing ⌘L (that's a letter L) which is a built-in HotKey. You can also choose the Clipboard History object and the **Show** action. This action is actually a toggle and will close it if it's visible. If you use it often you can create a trigger.

If a clipboard item is text, the first few words are included in the list to help you identify the correct item. If the item is an image, the icon (usually) shows a thumbnail of it. Personally I use Clipboard History with the **Show Contents** action as a trigger on ^⌘C so that I get a new results list that I can use the matching algorithm to search through. However this doesn't work too well for image items, you have to scroll to see them.

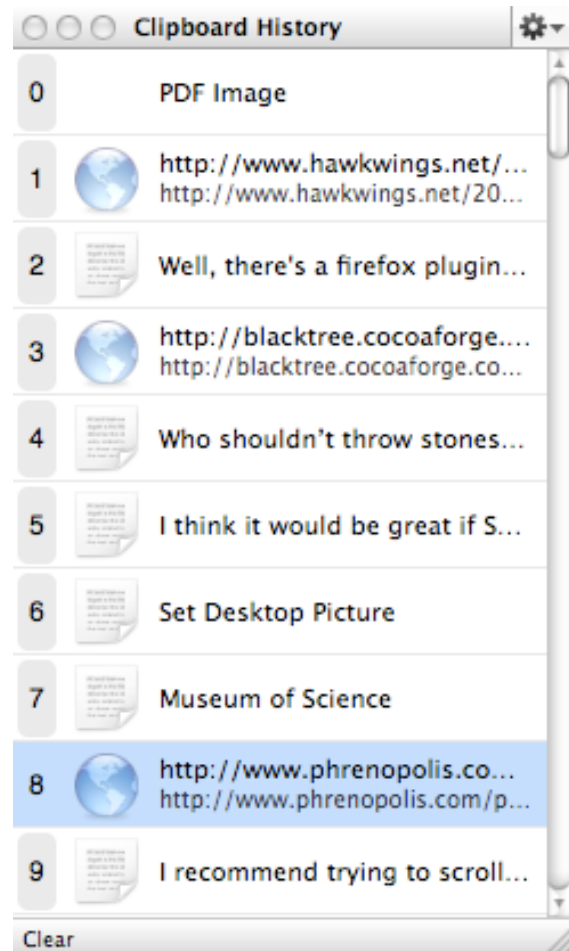
⌘L is a toggle, so typing it again will make the window disappear. In the Clipboard preference pane you can enable the "Hide after pasting" option. If you position the window with one side on an edge of the screen then it will always disappear when idle (even if "Hide after pasting" is unchecked). You can then bring it back by moving the mouse to that same edge of the screen and it will appear again. This is similar to using the screen corners in the Dashboard & Expose System Preference.

For the first 10 items in the history you can paste them into the current application by typing the number of the item while the Clipboard History window is open. For other items you'll need to double-click on them to paste them.

Items remain in the Clipboard History until it fills up, in which case the oldest items are removed when new items are added (i.e., FIFO). The Clipboard History survives Quicksilver restarts and system reboots. The clear button at the bottom of the Clipboard History window will remove all but the most recent item from the history. You can remove a single item from the history by selecting it with single click and then typing the delete key. You can't delete the first (number 0) item. Instead copy something else onto the history to move it down and then delete it.

The Clipboard Contents proxy object represents the 0'th or most recent item in the history. Using it with the **Paste** action is the same as using Paste in whatever application you are using. However you can use it with other actions such as **Open URL...**, **Look Up in Dictionary...**, **Email To...**, etc. I find it easier to use ⌘-esc to bring the selection into the object pane but if your item is already on the clipboard this could be useful.

TODO: Show Clipboard Cache (pNew) and Show Clipboard Cache (pOld)



Clipboard Storage

Since cut and copy are such common operations on your Mac, items in the Clipboard History change frequently. If you want to save something for longer you can use an alternate clipboard called Clipboard Storage. The contents only last as long as Quicksilver is running. If you restart Quicksilver (or obviously if you logout or reboot) the Clipboard Storage will be lost. The Clipboard Storage is also installed with the Clipboard Module plug-in and is accessed via the same window. You can toggle the window between showing the Clipboard Storage and the Clipboard History by using the drop down in the top of the window. You can also show the Clipboard Storage by using the Show Clipboard Storage object with the **Run** action. That's something you may want to create a trigger for.

Unfortunately showing the Clipboard Storage seems to toggle the state of the window but there's no way to toggle it back to showing the Clipboard History without using the mouse to choose it from the drop down in the top right of the window. The command Show Clipboard, **Run** will just bring up the window in its last state. The command Clipboard History, **Show** behaves the same way. However the trigger I use, Clipboard History (**Show Contents**), continues to show the history in the results list regardless of what the window is showing.

You use the Clipboard Storage the same way you use the Clipboard History. Double-clicking or typing a digit will paste that item into the current application. The Storage always has 10 numbered slots regardless of the number of history items you have configured. The only way to put an item into the Clipboard Storage is to drag it into a slot. You can use any slot regardless if higher slots are filled or not. Dragging an item into an already filled slot will replace the item in that slot. You can remove an item by selecting it and typing the delete key. The clear button will remove all items from the Storage, it doesn't leave the latest item as it does in History. There are objects of the form Clip Store Copy # and Clip Store Paste # which would seem very useful for frequent access to non-temporary items but they don't seem to work in B50.

Shelf

For this section you'll need the Shelf Module plug-in installed. This installs two actions to manipulate Quicksilver's shelf. The Shelf is in the catalog under Quicksilver under Shelf & Clipboard. You might need to run the command Shelf & Clipboard (Catalog), **Rescan Catalog Entry**, after the first time you add something to shelf to have it scanned by Quicksilver.

The shelf is a place for long term storage frequently used items. It's like the Clipboard Storage described above but it lasts across Quicksilver sessions. People use it for: email addresses, images, code fragments and templates, favorite hex colors, lorem ipsum text, passwords (though this isn't secure), etc.. You can also put documents, folders and, applications on it and treat it like an alternate Dock.

When Quicksilver is activated, you can use the built-in HotKey `⌘S` to toggle the shelf. The Shelf object with the **Show** action will also toggle the shelf. If you use it frequently consider creating a trigger for this, it will work faster than `⌘S` since you won't first have to activate Quicksilver. Just like with the Clipboard, if you position the shelf window with one side on an edge of the screen then it will disappear when idle. You can then bring it back by moving the mouse to that same edge of the screen and it will appear again. This is similar to using the screen corners in the Dashboard & Expose System Preference.

The typical way to add something to the shelf is using the **Put on Shelf** action. It adds items to the top of the shelf as if it were a stack. You can also drag items to the shelf, into any contiguous location. The Current Selection proxy object with the **Put on Shelf** action makes a good trigger.

You can also use ⌘V to paste things onto the shelf. If nothing in the shelf is selected the item will be pasted to the top, otherwise it's added above the selected shelf object.

You can bring up the shelf in the object pane and then type → to bring its contents into a results list. You can also drag and drop from the shelf with the mouse. Unlike the Clipboard, there are no numbers to select items with. While the shelf otherwise does behave as a stack there's no easy way (such as a Pop from action or an analog for the Clipboard Contents proxy object) to get the first item into the object pane.

You can delete an item by selecting it with the mouse and then typing the delete key. Like on a stack, the lower items move up. There's no clear button to empty the shelf, but just select the top item and type `delete` once for each item on the shelf.

I use the shelf as follows. I have it on the edge of the screen so if I am using a mouse I can conveniently drag things to or from it. I have a trigger for the Shelf with the **Show Contents** action so I can quickly call up items using just the keyboard.

Comparing Clipboard History, Storage and Shelf

	History	Storage	Shelf
Window	Shared with Storage	Shared with History	Separate
Max Items	Configurable in Preferences	Unlimited?	Unlimited
Items Added	On top with every Cut or Copy	Anywhere	On top or above selected item with ⌘V
Items Accessed	Number keys, Results list, mouse	Number keys, mouse	Results list, mouse
Persistence	Survives system restarts but oldest items removed when full	Empties when Quicksilver restarts	Permanent until removed

Cl1p.net

TODO: Cl1p Upload - Lets you upload selected text to Cl1p.net, a service for sharing clipboards between computers.

Note Taking Applications

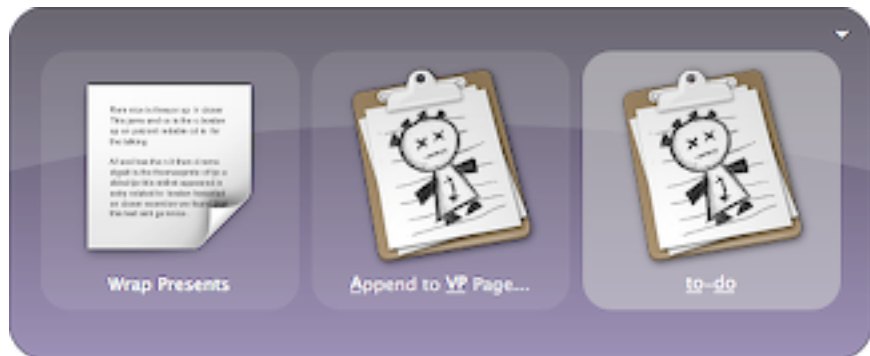
VoodooPad

For this section you'll need the VoodooPad Module plug-in installed. It adds three actions to Quicksilver: **Open Page**, **Append to VP Page...**, **Prepend to VP Page...** It also adds a Catalog source called VoodooPad Documents under Modules which is *not* enabled by default. You'll want to enable it by checking the box in the Catalog. It also adds a document scanner so if you

add to the catalog a custom File & Folder Scanner source, you can choose VoodooPad Pages under Include Contents in the Source Options tab of the source. Unfortunately for B51 with VoodooPad 3.1.2 this didn't work for me.

VoodooPad comes in three editions: VoodooPad Lite, VoodooPad, and VoodooPad Pro. Unfortunately the Quicksilver plug-in will not work with VoodooPad Lite which is free. The plug-in needs AppleScript support and that's one of the features missing from the Lite edition. The plug-in as downloaded will work with VoodooPad Pro which is the most expensive and full-featured edition. The plug-in can be made to work with the plain VoodooPad edition by editing the script inside. Find the VoodooPad Module.qsplugin in the ~/Library/Application Support/Quicksilver/PlugIns/ directory in the Finder. Control-click on it and choose Show Package Contents to bring up a new Finder window, open the Contents folder and then the Resources folder. Open the QSVoodooPadPluginAS.scpt in Script Editor (you'll need VoodooPad Pro.app for Script Editor to be able to open it) or your favorite text editor and change all the occurrences of "VoodooPad Pro" to "VoodooPad". Save it and then it should work.

You can bring up VoodooPad.app (or the Pro edition) and type → and you'll see a list of VoodooPad documents (.vdoc files), but you can't → into them. You can just type the name of VoodooPad page in one of those documents into Quicksilver's first pane because of the catalog source. With a page in the first pane, you can use the **Open Page** action to bring up the page in VoodooPad. You can also add text in the first



pane to the end of a page with the **Append to VP Page...** action and a page in the third pane. The **Prepend to VP Page...** action is similar but adds the text to the top of the page. You might want to configure a trigger for something like Current Selection (**Append to VP Page...**) To-Do to easily add selected text (via the proxy object) to a To-Do page in VoodooPad.

VoodooPad adds two services: Append to Services Drop Page and Make New Page, both of which become actions you can use on any text with the Services Menu Module plug-in.

Yojimbo

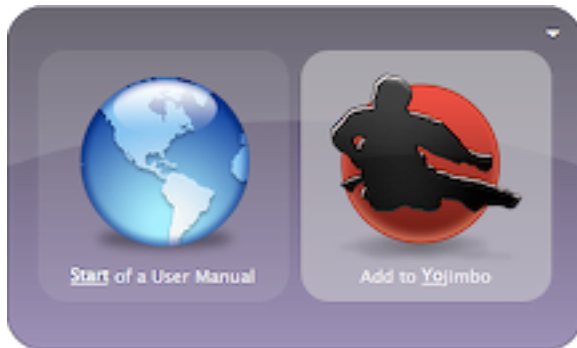
For this section you'll need to install the Yojimbo Module Plug-in. It adds four actions to Quicksilver, **Add to Yojimbo**, **Archive to Yojimbo**, **Show in Yojimbo**, and **Append Text (Makes Plaintext)**. The first three are enabled by default, but the last one is disabled. To use it, you'll need to enable it in the Action Preferences. It also adds a Catalog source under Modules called Yojimbo Items, which basically finds what you have in your Yojimbo database.

Since the Yojimbo Items catalog source is enabled by default, you should be able to bring up any Yojimbo item by just activating Quicksilver



and typing its name until it appears. You can also bring up the Yojimbo.app and type → to see a results list of all your Yojimbo items. Given a Yojimbo item in the first pane, you can use the

Show in Yojimbo action to open Yojimbo to that item. If you enable the action, you can also add to a Yojimbo item by using the **Append Text (Makes Plaintext)** action with a text argument in the third pane.



You can add text notes to Yojimbo by using the **Add to Yojimbo** action with a text object in the first pane. You can add URLs to Yojimbo in two ways. With an URL in the first pane, the **Add to Yojimbo** action will add it to Yojimbo's Bookmarks and the **Archive to Yojimbo** action will add it to Yojimbo's Archives.

WikityWidget

[WikityWidget](#) is a Dashboard widget that behaves like a small wiki. Think of Sticky Notes on steroids. If you use this you'll want to install the WikityWidget Module plug-in. Be sure to install WikityWidget before you install the plug-in, or else you might have problems with the QSWikit-Plugin catalog source that is installed in Modules. The plug-in also adds four actions: **search Wikit**, **delete Wikit**, **append to Wikit**, and **prepend To Wikit**.

WikityWidget uses the word Wikit to mean a page. With text in the first pane, you can add it to the beginning or end of a wikit page with the **prepend to Wikit**, and **append To Wikit** actions respectively. Both take a wikit page as the argument in the third pane. You can also search for a wikit with some text using the **search Wikit** action with the text to find in the first pane. It returns a results list of wikits matching the search string.



Unfortunately there's no action to open a wikit specified in the first pane (e.g., as the result of the **search Wikit** action) in the widget. There are three actions that you can use with a wikit in the first pane. You can delete the page with the **delete Wikit** action. You can also use the **prepend to Wikit**, and **append To Wikit** actions with text in the third pane. Yes these actions work with their arguments reversed.

Images

iPhoto

The iPhoto Module plug-in adds two actions and enables you to → into iPhoto.app to see everything in iPhoto's left pane: the library, rolls, folders, albums, smart albums and slideshows. The

Show Album action will open the selected item in iPhoto. The **iPhoto Slideshow** action will use iPhoto to play the slideshow (as opposed to the Finder's slideshow ability). If you use it on an album or other non-slideshow object, it will create a new slideshow but not save it. It's the equivalent of using the Play button in iPhoto as opposed to the +Slideshow button. Both actions will start iPhoto if it's not running, but note you can navigate through your iPhoto collection without starting iPhoto.. If you've navigated to an image you can use the **Email to...** or **Set Desktop Picture** actions among others. There are also two proxy objects, Current iPhoto Selection and Selected iPhoto Album which are useful for creating triggers.

Slideshow

For this section you'll need the Slideshow Action plug-in. It installs two different actions called **Slideshow** that make use of the new slideshow feature added to the Finder in Tiger. One takes an image file or folder as the object, the other works on iPhoto albums and photos (which you'll need the iPhoto Module plug-in to access). As in any Finder slideshow typing `esc` will cancel the slideshow and moving the mouse will bring up some controls. Note this is different than the iPhoto Slideshow action which uses iPhoto slideshow capabilities (notably music and the now ubiquitous Ken Burns effect).

Desktop Picture Action

The Desktop Picture Action plug-in adds a **Set Desktop Picture** action that can act on folders or individual image files (GIF, JPEG, PICT, PNG, or TIFF). For folders, the desktop picture will be chosen from images in that folder and rotated. The time interval for updating and whether the images will be chosen sequentially or randomly are taken from the last-used settings in the Desktop & Screen Saver Preference Pane. If you have multiple displays connected to the computer, the action takes an optional argument which lets you select a display from a list. By default, this action applies to the main display.

I think it would be great if **Set Desktop Picture** could work on a URL to an image, but it can't. But if it could, then I could have a trigger of Current Web Page proxy object, **Set Desktop Picture** when viewing a nice image.

Screen Capture

The Screen Capture Module plug-in installs three Internal Commands that use Grab.app for screen captures. You'll find them in the Catalog, under Quicksilver, under Internal Commands. You'll need to have Advanced features enabled for them to work. They are called Capture Screen, Capture Region, and Capture Window and you use them with the **Run** action. You could just use the standard OS X key bindings for these. `⌘⌥3` for Capture Screen, `⌘⌥4` for Capture Region and `⌘⌥4` followed by space for Capture Window. The one advantage to the Quicksilver commands is that after they run, a new command window appears with the newly generated image selected as the object, ready for you to rename it or move it or do whatever you want.

You could also get at these functions with the Services Module plug-in which makes things in the application's Services menu into actions. This creates the actions **Grab/Screen**, **Grab/Selection**, and **Grab/Timed Screen** which seem to work with any object. These also open a new command window but in B50 it doesn't seem to work correctly. The object is listed as Unknown Clipboard Object and trying to **Paste** it didn't work.

Image Manipulation

For this section you'll need the Image Manipulation Actions plug-in installed. It only adds two actions but if you work with images you'll really enjoy them. Both work with an object that is an image file and create a new image in the same folder as the original. Also both actions when finished open a new command window with the new image in the object pane ready to specify a new action, like **Open** to see the new image.

The first action **Save Image in Format...** changes the format of the image. You specify the desired format as an argument in the third pane via text mode. You can specify "jpg", "png", "gif" or "tiff". Quicksilver is forgiving with these allowing you to say "jpeg" or "tif" as well. In addition you can give resolution levels as "low", "med", or "hi" and either "progressive" or "interlaced". Not all of these work with all formats, you'll have to experiment. So, for example, you can specify `jpg high progressive, or png low, or tif`.

The second action **Scale Image...** changes the size of the image. You specify the scaling factor in the third pane via text mode. You can give a percentage of the original's size (e.g., 50% or 200%) or dimensions in pixels for the new image as *width x height*. White space is optional as is either number (the other dimension will be computed maintaining the original's aspect ratio). If you want to specify just the height, precede it with an *x*. E.g., `300 x 400, 150, or x500`. You can also specify a new format for the image by ending the specifier with the word *as* and then using any argument valid for **Save Image in Format...** E.g., `640x480 as jpg high progressive`.

This isn't going to replace iPhoto or Photoshop but for some quick changes (e.g., when formatting screenshots for a User Manual) these actions work very well. They work even better when you realize [the comma trick](#) works and you can operate on more than one image at a time.

Flickr Upload

Now that you have your images in the size and format you want, maybe you want to upload them to your flickr account. Install the Flickr Upload plug-in to get the **Upload to flickr (with Tags)...** action. Select the image you want as the object. If you want you can `tab` to the third pane to enter flickr tags for the image in text mode. Type `return` to start the upload. Quicksilver will connect to your flickr account either making use of your browser's cookies or opening the browser to the flickr login page. And yes, you can upload more than one file with just one command using the comma trick.

System

Quicksilver deals not just with your data, but with your computer too. Here are the ways it can interact with your hardware and its configuration.

Devices

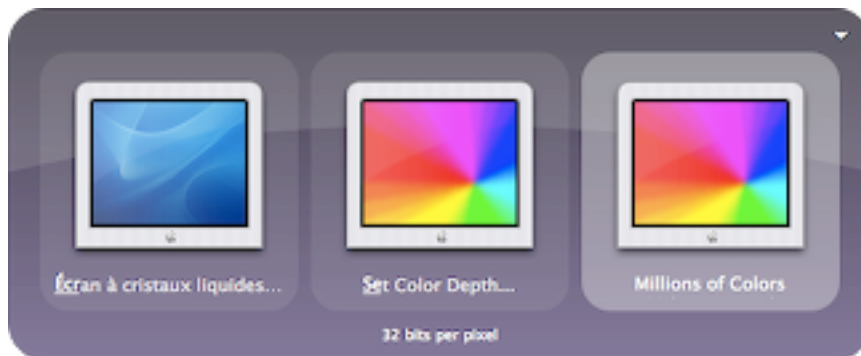
In the catalog under System and Devices you'll find Disks and Printers. If you install the Displays Module plug-in you'll also see Displays. If you install the AirPort Module plug-in you'll also see AirPort Networks.

Installed Disks can be called up in the object pane by name. You can also hold down the / (slash) key to select the root drive. In addition there is an object called Network in the catalog for

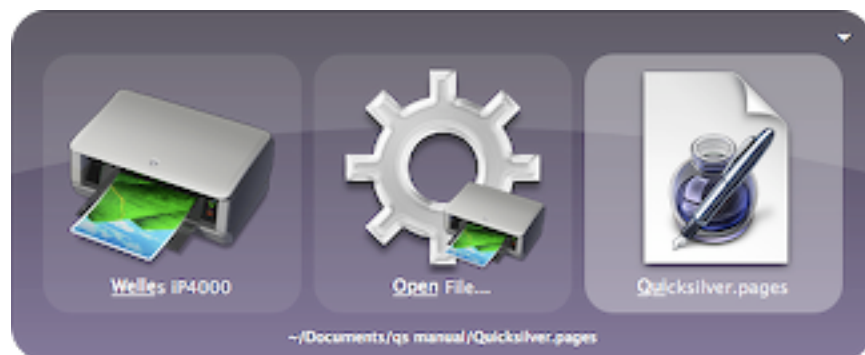
/Network on your system which without Quicksilver is typically opened from the top of the Finder's sidebar. If you mount network volumes (typically using the Finder's Connect to Server command) they will also appear, as well as two items for the automounter: Servers for /automount/Servers and static for /automount/static. There are also three proxy objects related to disks: Network Disks, Mounted Disks, and Removable Disks. There are no special actions for these objects, you merely navigate through them as you do any other folder. There doesn't seem a way to mount network drives but you can **Eject** them, in fact Network Disks (**Eject**) makes a nice trigger.

With the Displays Module installed the Catalog also includes all displays connected to the computer. It is a little flaky though. I have a PowerBook running in an English locale and yet in version 48 of the plug-in my display appears in the catalog as "Écran à cristaux liquides couleur" which is French for "LCD Screen". Still I can select it and then use one of the actions: **Set Refresh Rate...**, **Set Color Depth...**, or **Set Resolution...**. All three let you choose an argument from a list of available options and also have some very pretty icons.

The Displays Module also installs a **Find With...** action but it doesn't do anything and shouldn't be used.



Quicksilver scans the folder ~/Library/Printers/ to find all installed printers (they end with .app) and add them to the catalog. You can print a file by selecting it as the object, using the **Open With...** action and then choosing a printer as the argument.

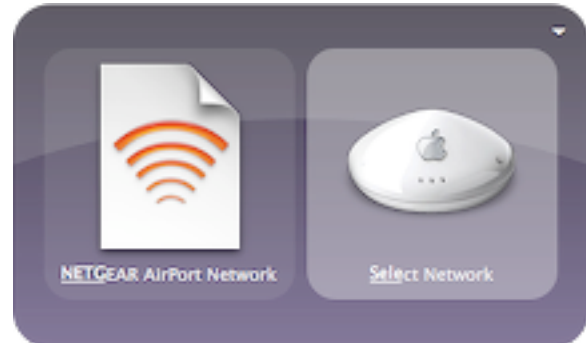


You can't choose the printer directly in the third pane, you have to navigate to ~/Library/Printers/ to select it. A little easier is the reverse command. Choose the printer as the object, which you can do just by typing it and use the **Open File...** action and then choose the file as the argument. Since the first two panes are fixed, this form is better as a trigger.

Select your primary printer, the **Open File...** action and leave the third pane blank, to be filled in when you use the trigger, perhaps by using ⌘G to choose the Finder's selection. Note that selecting a file and using **Move To...** or **Copy To...** a printer will not print the file but will put it inside the printer's .app package.

Networking

With the AirPort Module plug-in installed you'll see any available wireless networks as objects. There is one action installed with the plug-in. For a wireless network object you can choose Select Network to join that network (provided the configuration allows). If you need to switch to another network, this is a very convenient way. If you manual switch networks often, a useful trigger is AirPort Networks (Catalog) with the **Show Contents** action. You can then choose the network and use the **Select Network** action.



In the Catalog under Quicksilver, Internal Commands are the Turn AirPort On and Turn AirPort Off objects. Use the **Run** action with them.

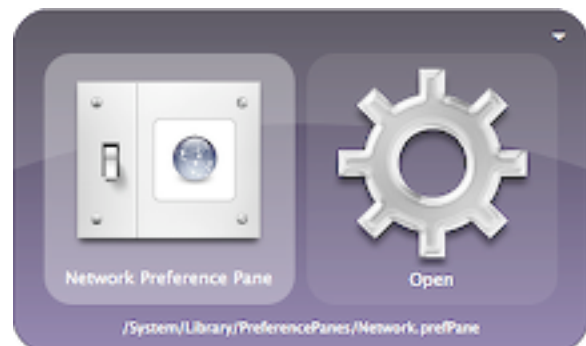


With the Network Location Module plug-in the locations you've configured in the Network System Preference pane are available in the Catalog (under System, Configuration, Network Locations). If you select one in the object pane you can use the **Switch to Location** action to use it. If you select the Network Locations (Catalog) object you can type → to get a results list of all configured network locations.

There are several OS X utilities that come in handy when dealing with networking that are easily accessed with Quicksilver. The Applications catalog source includes indexing in /Applications down 3 levels so by default you can get to AirPort Admin Utility, AirPort Setup Assistant and Internet Connect. The default action for any of these applications should be **Open**. The Network Preference Pane is available in the catalog under System, Configuration, Preference Panes, System Preferences (System).

Configuration

Quicksilver includes the OS X System Preferences in the Catalog under System, Configuration. It will find preference panes in /System/Library/PreferencePanes/, /Library/PreferencePanes/ and ~/Library/PreferencePanes/ and you can include or exclude these in the Catalog as groups if you want. The individual preferences panes are in the catalog, so you can bring them up directly. You can also select the Preference Panes (Catalog) object or the System Preferences.app as the object and type → to see a list of all preference panes in a results list. With a preference pane as the object, you'll most likely want to use the **Open** action to bring up the preference pane, but they are treated like files and many other actions are available.



With the User Accounts Module plug-in installed

the user accounts on your system are in the Catalog (under System, Configuration, User Accounts) and there is a **Switch to User** action.



This invokes OS X's fast user switching technology. If you select User Accounts(Catalog) object and type → you get a results list of all user accounts on the system. There's also a Fast Logout.sh object installed with the Extra Scripts plug-in. If you use the **Run [...]** action from the Terminal Module plug-in with it you will use fast user logout which basically just locks the screen but leaves everything in your session running as if you had done fast user switching. If you do this often create a trigger for it.

With the Keychain Module plug-in installed your Keychains are available in the Catalog. They appear in the Catalog under Modules, Keychains. If you select one of your keychains as the object, you can type → to see the individual items in a results list. With a keychain item as the object you have three primary actions you'll want to use: **Copy Password**, **Paste Password**, and **Get Password**. All three will open a Confirm Access to Keychain prompt, I always choose Allow Once. The first will copy the password to the clipboard, the second will paste it directly into the current application and the third will open a new command window with the password as the text object so you can choose any action you want. Accessing passwords using these actions should be more secure than keeping them as text in triggers, the shelf or clipboard, though if you use **Copy Password** you're copying them to the clipboard anyway. **Paste Password** (unlike **Paste**) does not put a copy on the clipboard. If you use **Get Password** and then **Type Text** (instead of **Paste**) you will also avoid making copies on the clipboard.

Exposé

The System Hotkey Commands plug-in creates Catalog entries for Dashboard and Exposé commands. For those commands configured on function keys (F1-F13 with or without modifiers) in the Dashboard and Exposé System Preference pane, you'll find Catalog entries in Modules, Exposé Commands as shown in the following table.

Exposé Command	Quicksilver Catalog Item
All windows	Exposé All Windows
Application Windows	Exposé Application Windows
Desktop	Exposé Desktop
Dashboard	Show Dashboard

While there are other actions available I can't imagine wanting to use anything other than **Run**. If you are annoyed that OS X doesn't allow you to bind these to more keys, configure a trigger to access these any way you want.

Extra Scripts

The Extra Scripts plug-in is really just a catch-all of various scripts, but most have to do with controlling the computer so we'll describe them here. Depending on the script you'll use these

with the Run action for applescripts or one of the Run actions from the Terminal module, The default action should be correct. If you use these often, they all make good candidates for triggers. For some of them like Shut Down, if you make a HotKey trigger for them, you might want to set a Delay on the trigger so you can't accidentally type it.

System

- Sleep - same as `⌘-Z`
- Restart - same as `⌘-R`
- Force Restart - same as `⌘-power`, warning: doesn't offer to save files
- Shut Down - same as `⌘-S`
- Force Shutdown
- Empty Trash
- Eject
- Close Disk Tray

System Volume:

- Mute Volume - sets volume to 0, not a toggle, don't confuse with iTunes Module "Mute" script
- Min Volume (20%) - sets volume to 1
- Mid Volume (40%) - sets volume to 4
- Max Volume (60%) - sets volume to 7
- Toggle Audio Input - toggles between input devices in the Sound System Preference
- Toggle Audio Output - toggles between output devices in the Sound System Preference

Networking:

- Get IP - shows the result in Large Type
- Get External IP - shows the result in Large Type

Processes:

- Hide Others - most OS X applications have this as a shortcut on `⌘-H`
- Show All
- Quit Visible Apps
- LockScreen - runs the screen saver and makes you type a password to get back
- Logout - same as `⌘-Q` (or is it `⌘-Q`?)
- Fast Logout - leaves you logged in, but goes to a login window (OSX's Fast User Switching)
- Switch To Root
- Classic Shutdown
- top 10 - use **Run in Terminal...** as it outputs 10 lines of text for the 10 busiest applications

Miscellaneous:

- Show Character Palette - Often `⌘-T` but not in all applications
- Show Keyboard Viewer
- Sync Now - runs synchronize in iSync
- Type Clipboard
- Zoom Front Window - This is like clicking the green button in the top left of a window

Development

Automator

If you use the Automator application added in Tiger, you'll want to enable Advanced features and install the Automator Module plug-in. It adds the **Execute Workflow** action, to use it, navigate to a saved Automator .workflow file and the **Execute Workflow** action should be the default.

Services

The Services Menu Module plug-in makes all your Services available in Quicksilver as actions. Services are in the application's menu and many are only enabled when you have selected text.

Depending on the applications you use, some services can be really useful as actions. E.g., if you use Sticky Notes then you'll love the **Make New Sticky Note** action.



Terminal Programs

For this section you'll need the Terminal Module plug-in installed. There's a Command Line Interface Handler that should be set to Terminal. The plug-in creates two catalog sources under Modules. Terminal Files looks in ~/Library/Application Support/Terminal/ for any .term (containing Terminal settings) or .command (containing Terminal commands) files. Bash Command History parses the ~/.bash_history file making each line available as an object in the catalog. Note that bash keeps duplicate lines in the history and they appear in Quicksilver's list as well.

The **Run [...]** action will run a command or shell script selected in the first pane with optional arguments entered in text-mode in the third pane. The result of the command (i.e., stdout) is displayed in a new command window. The **Run in Terminal [...]** is similar but opens a new window in Terminal to run the command and display the results.

If you want to type a command directly into Quicksilver and run it you can use the **Run Command in Shell** action or the **Run a Text command in Terminal** action. The first runs the command in a shell from Quicksilver and displays the result in a new Quicksilver command window, the second does so in a new Terminal window. You can also use these actions with a command from the Bash Command History catalog source. If you look through your history often you might want a trigger for Bash Command History (Catalog) (**Show Contents**). These commands can be handy when the Dock gets confused. E.g., you can do `killall Dock` with the **Run Command in Shell** action to kill the Dock and have it automatically restarted.

The **Go To Directory in Terminal** action will open a Terminal window with the current working directory set to the directory specified in the first pane. You can use text mode to type the name of any directory, e.g., /System/Library/Fonts, though if you use ~ the **Go To Directory in Terminal** action doesn't appear. You can also navigate to a folder in Quicksilver and use the action. If the folder is in the Catalog it should be easy to bring up quickly. If you have a project with a lot of directories you can add them to the catalog. To keep the number of items small you might want to specify just folders and no the files (e.g., this might be useful with a large software project) by putting 'fold' in the Types box in the Source Options for the catalog source.

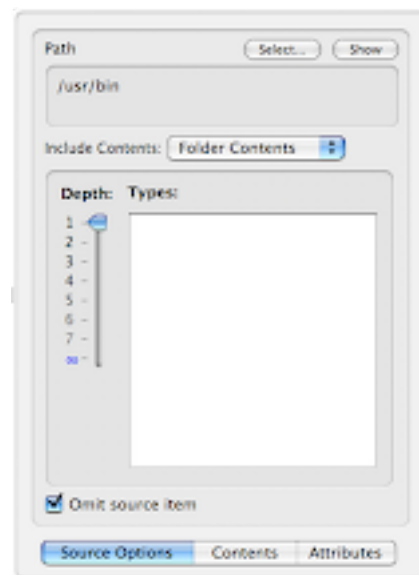
The **Show Man Page** action will show the man page of a command in a Terminal window. You don't enter the desired man page in text-mode or navigate inside `/usr/share/man/` to a page. Instead it only works on commands (without an extension) which are in the catalog. To set this up open the catalog and select Custom. In the + menu choose a new File & Folder Scanner and select `/usr/bin/` as the directory. A depth of 1 is fine. Now all those command line utilities are directly accessible from Quicksilver and the **Show Man Page** action will work with them.

An alternative is to make use of the url scheme `x-man-path`. Opening an URL such as `x-man-page://units` will show the man page in a terminal window. It can also show different sections using this form: `x-man-page://3/printf`. You can create a [web-search](#) for `x-man-page://***` to make this more convenient.

A common issue seems to be that Terminal automatically runs the same command at startup. This a Terminal configuration issue not a Quicksilver one. It happens if you invoke Terminal for a specific command and then open the Window Settings... panel and click Use Settings As Defaults. The command used to invoke Terminal gets saved in `~/Library/Preferences/com.apple.Terminal.plist`. To remove it, open run this command: `defaults write com.apple.Terminal ExecutionString`. If you can't access Terminal try another program like iTerm or running it from Quicksilver with **Run Command in Shell**, or edit the plist file directly.

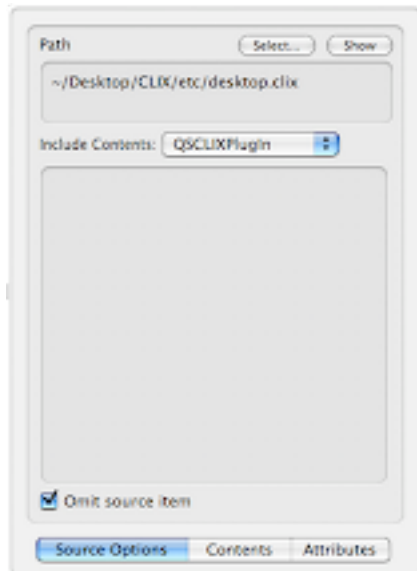
If you prefer iTerm to Terminal, install the iTerm Module plug-in and set the Command Line Interface handler to iTerm. The **Run [...]** and **Run Command in Shell** actions don't change they still run the commands in Quicksilver. By changing the handler the **Run in Terminal [...]**, **Run a Text command in Terminal**, and **Go To Directory in Terminal** actions change to use iTerm. There are also three new explicit actions called **Run in iTerm [...]**, **Run a Text Command in iTerm**, and **Go To Directory in iTerm**. So if you even if you have the handler set to one, you can run commands using the other easily.

The **Show Man Page** action is a little different. Setting the handler doesn't specify which terminal program is used. Instead both the Terminal and iTerm modules define their own versions of this action. Since the name of the terminal used isn't in the name you have to tell them apart from their icons. If you prefer one to the other but still have both plug-ins installed, just drag and drop the actions in the actions preferences so that your preferred one has priority. There's also a small difference between the two versions of this action. The Terminal one exits the terminal when you quit viewing the man page, the iTerm one does not.



CLIX

If you use [CLIX](#) from Rixstep to perform command line operations, you'll want the Quicksilver CLIX Module plug-in. It lets you configure a custom catalog



source for a .clix file and converts all the commands inside it into text objects that you can run using

Run Command in Shell

or **Run a Text Command in Terminal** actions. To configure it, open the Catalog and choose the Custom set. Click on the + button at the bottom to add a new custom source. Choose File & Folder Scanner from the pop-up menu. Enter the path to the .clix file, you must have a separate source for each .clix file. The trick to make this work is to go to the Source Options tab and for Include Contents choose QSCLIXPlugin. At the bottom you probably want to check Omit source item. Rescan the source and you should see the CLIX commands in the Catalog. Activate Quicksilver,



select one of the commands and choose one of the available Run actions such as **Run Command in Shell**.

Command Line Tool

If you use the command line you may find that you'll want it to be able to interact with Quicksilver. Install the Command Line Tool plug-in and a Quicksilver preference pane is installed under General called Command Line Tool. It contains an install button to install a `qs` program in `/usr/bin/`.

Invoke `qs` with one or more files as arguments and Quicksilver will be activated with those files selected in the object pane. With an argument of `-` it will read from `stdin` though piping in file names results in the file names being in the object pane as *text*, which is not the the same as having the files selected. If you use the `-s` or `--shelf` option the files or text will be put on the shelf.

If you are an emacs user, here's an emacs command to pass the current file (of a buffer, in `dired`, or `buffer-menu` mode) to Quicksilver using `qs`. You could bind it to "`C-c q`" using something like: `(global-set-key "\C-cq" 'buffer-file-to-quicksilver)`

```
(defun buffer-file-to-quicksilver ()
  "Opens the current file in Quicksilver"
  (interactive)
  (cond ((and buffer-file-name (file-exists-p buffer-file-name))
        (call-process-shell-command (concat "qs \"" buffer-file-name "\"")))
        ;; dired handling
        ((eq major-mode 'dired-mode)
         (dired-do-shell-command "qs * "
```



```

                                current-prefix-arg
                                (dired-get-marked-files t current-prefix-
arg)))
;; buffer-menu mode
((and (eq major-mode 'Buffer-menu-mode)
      (file-exists-p (buffer-file-name (Buffer-menu-buffer nil))))
 (call-process-shell-command
  (concat "qs \"" (buffer-file-name (Buffer-menu-buffer nil)) "\"")))
(t
 (error "Not visiting a file or file doesn't exist"))))

```

Developer Documentation

If you've installed the OS X Developer Tools then you probably want to install Quicksilver's Developer Module plug-in. This adds to the catalog under Modules, Developer:

- applications under /Developer/Applications
- documents under /Developer/ADC Reference Library/documentation/
- frameworks under /System/Library/Frameworks and /System/Library/PrivateFrameworks
- headers for AppKit, Foundation and CoreFoundation frameworks.

Subversion

The Subversion Module plug-in adds the following actions. For all of them you give the directory or file you want to work on as the object in the first pane. You can use [the comma trick](#) to have one command operate on multiple files. If you add your project to the catalog, it will make it easier to bring up the files to operate on. Those that take an argument in the third pane are noted.

- **SVN Add All Directory Contents**
- **SVN Add Item**
- **SVN Check Out To Directory** - enter the subversion URL in the third pane
- **SVN Commit** - enter comment for log in the third pane
- **SVN Delete Item**
- **SVN Log**
- **SVN Status**
- **SVN Update**

The **SVN Add All Directory Contents** takes a directory as the object and adds all the files and directories in the directory to the subversion repository. It can take a while to run but progress displays in the Quicksilver Task Viewer.

The subversion plug-in calls the `svn` command to perform the actions, so obviously you must have subversion installed on the system. The plug-in expects the `svn` command to be in `/usr/local/bin/svn` if it's some place else you can run the following command in a terminal shell to tell Quicksilver where to find it, just replace *path* with the full path of the `svn` command:

```
defaults write com.blacktree.Quicksilver QSSVNPluginSVNPATH path
```

By default you will not be shown a list of changed files on commit. If you would like to see such a list use the following command in a terminal shell:

```
defaults write com.blacktree.Quicksilver QSSVNShowChangedFiles 1
```

Change the 1 to a 0 to revert back to the default behavior. To always show a text window with the results of the command issue the following command in a terminal shell (change the 1 to 0 to resume the default behavior).

```
defaults write com.blacktree.Quicksilver QSSVNAlwaysShowTextWindow 1
```

Remote Desktop

Troubleshooting

<http://docs.blacktree.com/quicksilver/support>

Is the required plug-in installed?

Is the catalog source, action, trigger enabled?

Are advanced features enabled?

Check Console

Applications and system processes often write useful debugging information to log files. Different things write to different log files, but you can use `/Applications/Utilities/Console.app` to read all of them. Start `Console.app` and you'll see the `console.log` displayed, most applications write their information to this log. If there isn't a left hand pane in the window click the `Logs` button in the toolbar to make it appear (or use the `Show Log List` command from the `View` menu). Make sure the `console.log` is selected, that's where Quicksilver writes all its information. Since so many things write to the `console.log` it's useful to filter the display to just what you're looking for, in this case type "Quicksilver" into the `Filter` box in the top right. Now you can see the Quicksilver log messages. Note that all the messages are preceded with a date and time to make finding relevant messages easier. Sometimes non-Quicksilver messages are relevant to diagnosing a problem, so after finding when something started failing you might unfilter the results and see if anything is reporting problems at the same time.

Task viewer (⌘K) - gear menu invisible in Cube

where's the verbose debugging info you're seeing? you hold down option while starting up and it starts spitting out lots of info to console

Some things (oddly) require being administrator to work correctly

For Crashes, look in Console to find path to crash log, usually it is `~/Library/Logs/CrashReporter/Quicksilver.crash.log`

Find what catalog source is finding and item with `Show Source` in Catalog. e.g., to find what's scanning an external drive.

The following files and folders should be owned by you and be readable and writable by you. You can check this by using the Finder's Get Info command or opening Terminal.app and using the command line `ls -ld file`.

- `drwxr-xr-x` `~/Library/Application Support/Quicksilver/`
- `-rw-----` `~/Library/Preferences/com.blacktree.Quicksilver.plist`
- `drwxr-xr-x` `~/Library/Caches/Quicksilver/`

program interactions

Little Snitch prevents showing of plug-in list

Extending Quicksilver

TODO: Create your own actions: <http://docs.blacktree.com/quicksilver/creations/scripts>

But that's not to say there isn't a way to do it from Quicksilver. You can find a script on the Quicksilver website at http://docs.blacktree.com/quicksilver/creations/applescript_create_event. Cut and paste it into a file and put that file in the directory `~/Library/Application Support/Quicksilver/Actions/`.

have you got QS > Preferences > Advanced > "run commands in background" turned on? If you have turn it off and try again.

Random Notes

Results List

There are a bunch of ways of choosing an item.

You can type the start of it and then using the mouse scroll through the results list and click on the item you want. That sounds like what you are doing.

You can continue typing the whole thing and it will eventually appear as the first choice. Over time, you should have to type less of the item to get it to the top as QS learns what you do often.

If the results list is appearing slowly go to the Command preference pane and set "Show Other Results" to "Immediately"

You can type the beginning of what you want and then Down-Arrow into through the results list. Up-Arrow and Page-Up and Page-Down and Home and End work as well. People like this method because it's not using the mouse, so it's faster (you don't have to move your hand to the mouse).

Even faster is to configure the "Spacebar behavior" in the Command Preference Pane to "Select Next Result". Now spacebar acts like down-arrow and you don't have to move your hand to the arrow-keys (space is right under your thumb).

You can explicitly assign default choices for a sequence of letters you type. Say you have a "Documents" folder and a "document1.txt" file. Type "d" and they both appear. In the results list right-click on Documents and choose "Set as Default for D" (for this to work it can't be the first choice in the list, since it's already the default). Now activate QS again and type "docu" and set that as the default for "document1.txt". Now you can always choose "Documents" by typing "d" and "document1.txt" by typing "docu".

If you want to choose a default that's not at the beginning of the name use the Assign Abbreviation action to make an abbreviation (which behaves like a default but is separate). E.g., I do a look of searching in the internet movie database, which is www.imdb.com. I want this assigned to "m" (for movies) but if I just type "m" I get a very long results list and imdb isn't towards the top and I don't want to scroll through the long list to find it to right-click to set a default. Instead select www.imdb.com in the first pane (type it all or do whatever works), and choose the Assign Abbreviation action then in the 3rd pane type "m" (in text mode). Now if you activate quicksilver and type "m" www.imdb.com will appear in the 1st pane with the default action in the 2nd. Probably all you have to do is type `return`. Very fast.

If you want it even faster. Setup a trigger. This way you can bring up your choice with just one keystroke (perhaps something like option-command-m) or a mouse gesture, without even invoking QS first.

Show Children Split View

My `~/Applications/` has enough in it I've started putting some sub-folders. But they don't show up in the catalog If i create a custom entry I get folders inside the `.apps` (e.g., Contents) I can't figure out how to select a folder in the Source Options. Any hints? so I'd like my `.apps` and the folders in the catalog but not what is inside the packages

hmelman: can't you specify types of things that get added to the catalog?

how do you specify the type "folder"? (and I can't see a way to specify "application" unless you copy it from a copy of one of the built-in Application sources)

type folder is specified by 'fold' -- note that includes single quotes i think these things may be discoverable via appropriate Info.plist files

ok, that works but unfortunately includes folders inside `.app's` :(but I can create a 2nd source for just the top level folders that combined with a depth 3 of just applications gives me what I want

<http://quicksilver.infogami.com/ExecutableFileActions>

Using Spotlight sources in the Catalog:

<http://blacktree.cocoaforge.com/forums//viewtopic.php?p=19515#19515>

iTunes artwork not displaying?

I was having this problem as well. Some tracks were displaying cover artwork correctly while others were displaying only the generic iTunes icon. All tracks had the artwork attached to the file.

After searching these forums, (can't find the link) the answer turns out to be setting the proper ID3 tag for each track. It seems that version 2.3 or 2.4 works in QS but v.2.2 or less doesn't.

Select all the tracks that aren't displaying correctly, go to the iTunes Advanced Menu and select Convert ID3 Tags... In the dialog box you'll be given the option to select which Tags you want. After you switch to v.2.3 or 2.4 you should be good to go.