ImageView

Version 2.3.4 20 April 2008

Summary

ImageView is a simple image viewer. An image dragged onto the Widget's window is displayed.

ImageView can spawn copies of itself, so that several images can be viewed at the same time.

When ImageView is restarted, it can restart the spawned Widgets, so that they can display their respective images.

Each copy (spawn) of ImageView can also provide a slide-show using images from nominated folders.

Instructions

The Widget is designed to spawn instances of itself, so you set it up as follows:

Start the Widget and drag an image onto its window. If you wish, open the preferences and change the size, rotation and subtitle of the image. You now have the first instance of the Widget (the original ancestor).

If you wish to view a second image, click on the + button, and the Widget will make a clone of itself (its first child). Configure the child for your second image. You can now click on the + button of either the original Widget or the child to make another clone, and so on.

When you restart the original ancestor, it will restart (some of) the children for you.

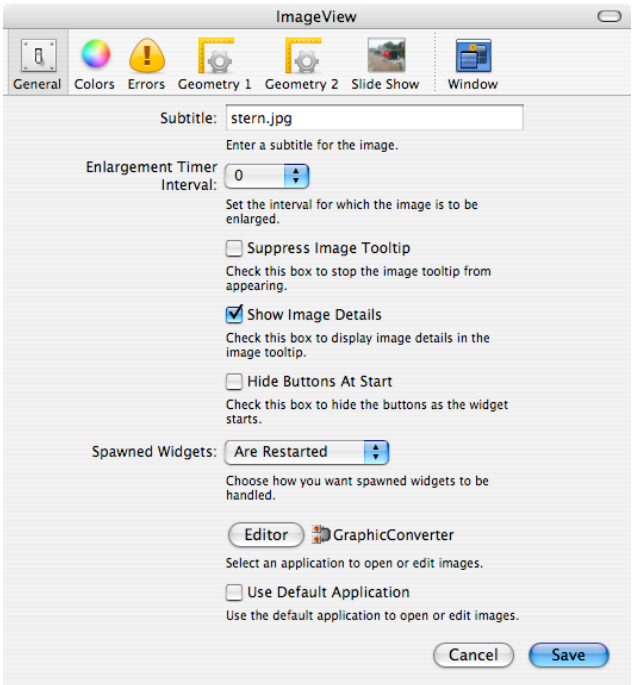
The appearance of a Widget instance can be changed by dragging an image file onto it.

The last image dragged onto a Widget is stored (in the WidgetDataFolder). It can be deleted by using the wrench (spanner) button.

ImageView can also provide a slide-show using images from a nominated folder. The folder is set in the Widget's preferences, together with the interval at which images are changed. Images can be chosen in sequence or at random from the nominated folder and its sub-folders.

An auto-enlarge feature can be enabled in the Widget's preferences. When enabled, an image can be enlarged by placing the mouse pointer in the image area for about two to three seconds. The image remains enlarged until a few seconds after the mouse pointer has been removed from the image area. This interval is also settable in the Widget's preferences.

This version has the ability to open a displayed image or slide in a user specified application (usually an editor). The application can be selected in the **General** preferences pane.



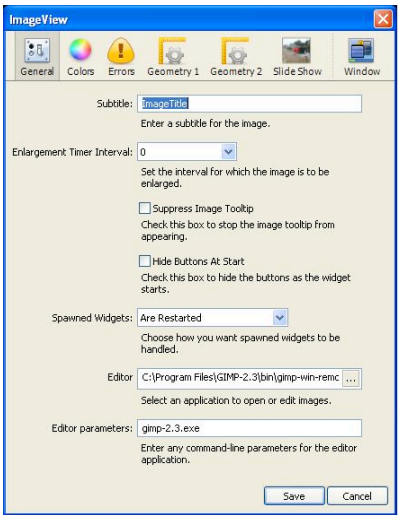
[*Windows Only*] Additional command line parameters can be entered.

For example: To use Gimp as the editor application:

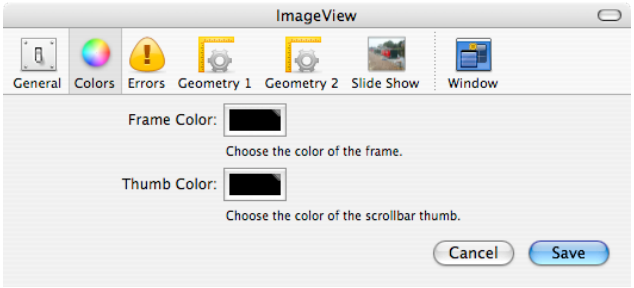
Select the application **C:\Program Files\GIMP-2.3\bin\gimp-win-remote.exe**, which is a little application that launches Gimp (if need be) and sends image pathnames to it.

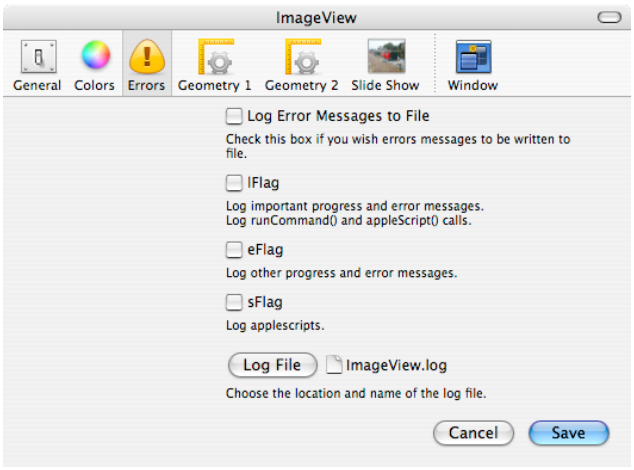
The parameters it takes (for Gimp version 2.3) are "gimp-2.3.exe file1 file2 ...",
so the Editor parameters are set to **gimp-2.3.exe** .

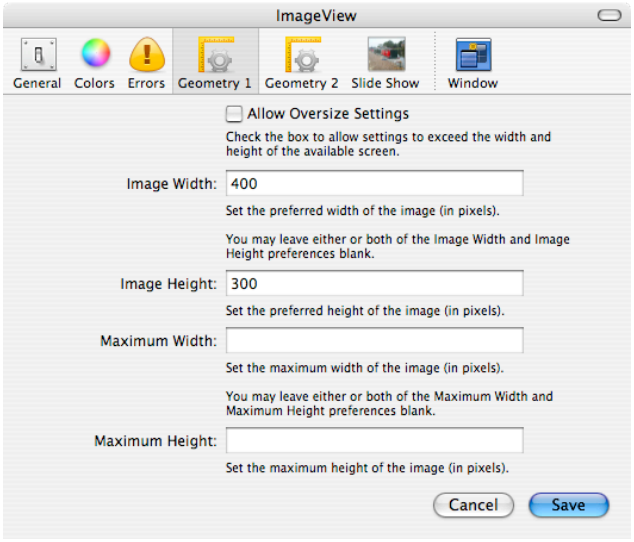
When the editor is called, ImageView appends the pathname of the displayed file (as file1).



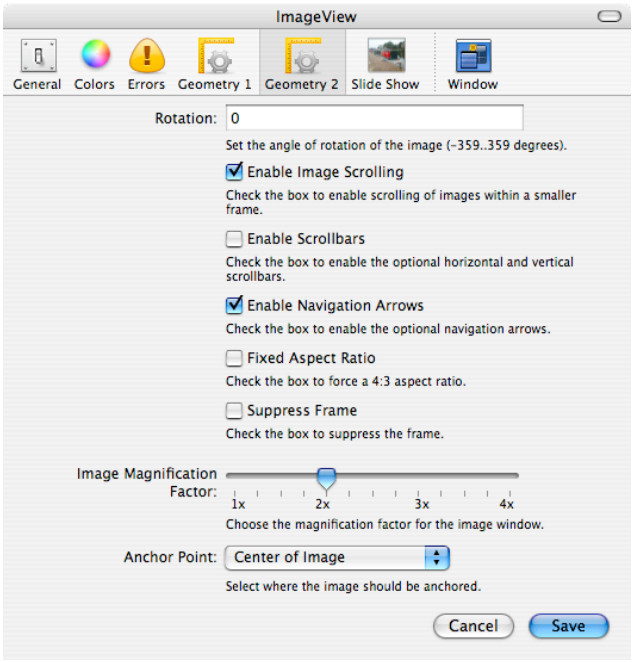
Other Preference Panes





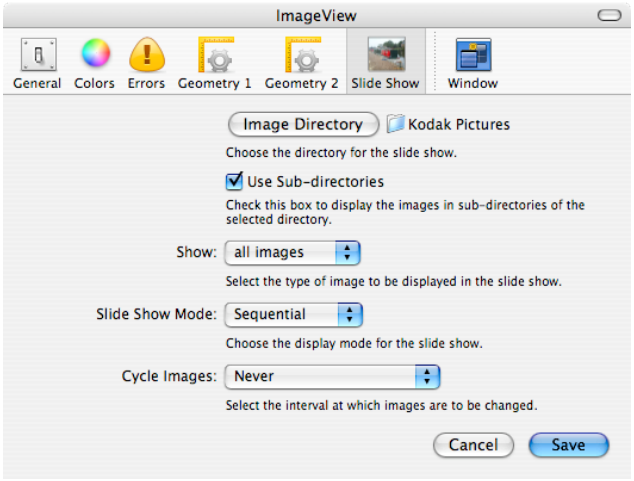


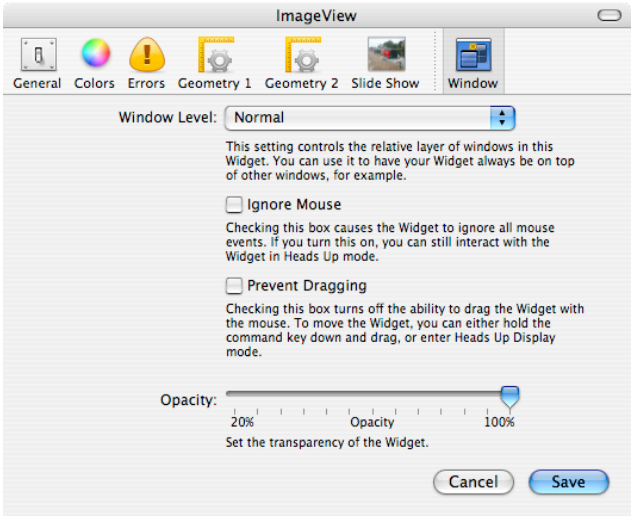
The size of the frame and the image is determined by the values set in the Geometry 1 pane.
You may fill in as many or as few values as you wish. The Widget takes note of any that are filled in.



This version has a new scrolling feature which enables the user to scroll a large image in a small window.
Images can be scrolled by dragging on an image. Optionally, users may enable scrollbars and navigation arrows.

Images can be rotated by filling in a rotation value in the preference pane, or by alt-dragging around the center of an image.





Widget Instances

A child instance has a unique name based on the name of its parent.

Say that the initial ancestor Widget is called **ImageView**

Its children are called **ImageView1** to **ImageView9**

ImageView1 's children are called **ImageView11** to **ImageView19**

ImageView13 's children are called **ImageView131** to **ImageView139** etc.

A child instance makes use of code and other resources held within the original ancestor Widget, but has its own preferences, which are initially the same as its parent's preferences, apart from its subtitle and displayed image.

In order to save a little disk space, an option is available to have Widget instances deleted when they close. Deleted instances can't be restarted from the "Recent Widgets" menu, and they are not reloaded automatically when Konfabulator restarts.

A user can repeat a spawning sequence to recreate Widget instances with names previously used, thereby picking up old (configured) preference files and images.

An option is available to have the original ancestor Widget restart any (undeleted) children, whenever it is started.

Option-click the + button of the original ancestor Widget to display the Spawn Management form, which allows for selective restarting and deletion of spawned Widgets.

[*Macintosh Only*] Option-click the x button of the original ancestor Widget to close any spawns that are running.

Graphics and Interface Design

Ricky Romero

Widget Concept, Coding and Documentation

Harry Whitfield <g6auc@arrl.net>

Credits

The Widget spawning code is based on code by Arlo Rose.

Thanks go to Rob Davenport <rob.davenport@gmail.com> for useful feedback and comments.