

Giving an application a window icon in a sensible way

Source: <http://coding.derkeiler.com/Archive/Java/comp.lang.java.programmer/2006-11/msg02416.html>

- *From:* "Twisted" <twisted0n3@xxxxxxxxx>
 - *Date:* 19 Nov 2006 17:24:57 -0800
-

Hrm. How to go about doing this?

I want to give a Java application a window icon in a manner that is independent of how it is installed, etc.

The trick is obtaining an Image object for myJFrame.setIconImage(). Loading it from a URL means it won't work without a working network connection, and I need to host the image somewhere. Every copy of the app running anywhere in the world will, on startup, hit that host with a request for the file(!). Loading it from a file path requires a separate installer that sets the image into a specific directory. Putting it in a JAR file with the app means learning a big new chunk of API, plus it won't work when running in the development environment rather than as a standalone executable JAR.

This suggests doing the ListMessageBundle sort of thing, and somehow packaging it as a class — an Image subclass, presumably. Is there a tool for turning a jpeg, gif, or png into Java source code for an Image subclass that will, when instantiated, behave as the appropriate jpeg?

I don't have the google-fu to find this — searches for queries like "image resource class java" didn't do much for me. Damn, we need *real* natural language search. :P

.