

## Re: Image viewing – a different approach

---

*Source:* <http://coding.derkeiler.com/Archive/Tcl/comp.lang.tcl/2008-01/msg01048.html>

---

- *From:* Glenn Jackman <[glennj@xxxxxx](mailto:glennj@xxxxxx)>
  - *Date:* 22 Jan 2008 20:01:02 GMT
- 

At 2008-01-22 02:10PM, "ZB" wrote:

Dnia 22.01.2008 Glenn Jackman <[glennj@xxxxxx](mailto:glennj@xxxxxx)> napisa<sup>3/a</sup>:

[...]

launch::file myimg.jpg

If I properly understood, it's a method for platform-independent "launch" ("run", "exec"... whatever) command. But my intention was somewhat different: I would to create some kind of "link", which could be then pack-ed anywhere in chosen toplevel – and such link, when clicked by the user, would then be seen by OS as "shortcut to file"; exactly the same kind of shortcut, as every Windows user has his desktop filled with.

Such way we don't have to care about file type, launch command and other details anymore... everything has been set much earlier by the user, according to his needs.

This would be best solution for both programmer (less work, of course...), and the user (less work by setup/configuration, of course... all is prepared already along with his OS's file associations).

I'm not sure, is it currently possible at all. If not – perhaps could be \*something\* added in future versions of Tk?

The more I read your post, the more annoyed I get. It strikes me that you're dismissing my ideas even though you have nothing concrete to offer. I'll try to put that aside...

I think you missed my point entirely. To accomplish platform independence, surely \*somewhere\* must be listed how to do an action for various operating systems. "On windows do X, on darwin do Y, etc, etc"

That's what I was trying to address. On Windows, you use the "start" command to instruct the OS to take a file and give it to the user's preferred application. On Mac OSX, it's "open". I don't know how to do

## Re: Image viewing – a different approach

that for BSD/Linux/Unix...

If you want to make a "File Association Link" widget for a GUI, you have to define a command to do something with an instance of that widget. I was offering a starting point, something that could potentially be turned into a package, potentially included in tklib.

—

Glenn Jackman

"You can only be young once. But you can always be immature." — Dave Barry

.