



Keyboard shortcuts

With an N-tier applications testers may have to go under the Web presentation layer. For example, that could be a configuration program running on Windows Desktop. Or you may have to work with UNIX programs, directly or through a terminal emulator program. Some UNIX programs are simply called from the command line. Some other may have a sophisticated user interface, with "windows", "menus", "pop-up dialogs", and keyboard shortcuts. Today's tip is about the latter.

Track the "life span" of keyboard shortcuts to see if they are enabled/disabled on timely manner. Because, if they're not, you'll be able to bring a program into a really weird state.

As for Windows Desktop applications, track if keyboard shortcuts are defined consistently across different modules, and do not override standard meanings. That is, if "Quick Save" command is "Ctrl-S" in one module, it should not be "Alt-S" in another.

Filed under: [DOS/UNIX Apps](#), [Heuristics](#)

[Leave a comment](#)

Comments (1)

Trackbacks (0)

(subscribe to comments on this post)

jessedyer

December 21st, 2010 - 09:46

For that matter, GUI programs, at a certain maturity level, tend to reach hotkey cruft stage. Look for hotkey's defined by underlined characters in function labels; Like underlined F in File means Alt-F pulls down the File menu by standard convention. See if an underlined F appears elsewhere in the same screen; which key wins, or does neither function? Either way, what bliss.

Also look for unintended hotkey goodies. Ctrl-C (break) causing a program to halt is an acceptable behavior in a bash or batch script, but not so acceptable in a compiled GUI program where the standard is instead for the sequence to mean "Copy". I've actually seen that happen.

Ah, then there's Tab order. Does Tab switch you through elements, or insert a tab character? What about shift tab? Does your movement through elements follow any kind of logical progression? Or does the highlight move willy-nilly through the program. Do you even _have_ a highlight?

Then there's Mac keys versus Windows/Linux conventions, upper ascii/unicode through alt sequences, typematic rate bombs, task switch lockout, the true meaning of escape, and so on....

Sorry, hot button issue.

Leave a comment

Name (required)

Mail (will not be published) (required)

Website

Submit

[« Area highlighter trick](#)

[A Fly in the Ointment »](#)

Categories

- [Accessibility](#) (4)
- [Automation](#) (23)
- [Availability](#) (1)
- [Bias](#) (8)
- [Bug Reports](#) (22)
- [Career Tips](#) (8)
- [Compliance](#) (1)
- [Database](#) (3)
- [Disaster Recovery](#) (2)
- [Documentation](#) (16)
- [DOS/UNIX Apps](#) (1)
- [Estimation](#) (1)
- [Exploratory Testing](#) (25)
- [Free Tools](#) (74)
- [Heuristics](#) (53)
- [Internationalization](#) (5)
- [Learning about the product](#) (7)
- [Mind Mapping](#) (11)
- [Performance Testing](#) (34)
- [Playing Well With Others](#) (40)
- [Practicing Testing](#) (8)
- [Regression Testing](#) (2)
- [Security Testing](#) (13)
- [SEQ](#) (1)
- [Skilled Bug Investigation](#) (13)
- [Stress Testing](#) (1)
- [Technical Tricks](#) (2)
- [Test Data](#) (16)
- [Test Management](#) (33)
- [Test Oracles](#) (2)
- [Test Planning](#) (32)
- [Test Theory](#) (5)
- [Testability](#) (1)
- [Testing mobile](#) (6)
- [Testing Techniques](#) (17)
- [Time Savers](#) (29)
- [Tools under \\$100](#) (9)
- [Uncategorized](#) (2)
- [Unit Testing](#) (2)
- [Usability](#) (13)
- [Web Testing](#) (31)

Authors

- [Albert Gareev's website](#)
- [Anne-Marie Charrett's blog](#)
- [Jonathan Kohl's blog](#)
- [Julian Harty's website](#)

[Karen Johnson's blog](#)

[Mike Kelly's blog](#)

Pages

[About](#)