Subject: IDLWAVE 4.15 Release -- idlwave.org
Posted by JD Smith on Wed, 11 Sep 2002 05:55:15 GMT
View Forum Message <> Reply to Message

IDLWAVE 4.15 -- http://idlwave.org

A hot new release of IDLWAVE is available for those of you addicted to new features, bug fixes, and the relentless march of ever-burgeoning version numbers. Following the time-honored tradition of small even and large odd releases, this version of IDLWAVE serves up a bevy of choice morsels:

====> New features: <====

- A special loadable module, `idlw-roprompt' allows Emacs 21 users to make the "IDL>" prompt read-only (like it is in an XTerm). Emacs 21 also brings many new features to the shell, including toolbars, the ability to middle-click on old input to have it automatically copied to the command line, colored prompt/input, etc. Highly recommended.

- When completing methods or method-keywords, or invoking help or routine info on objects in the shell, IDLWAVE now automatically queries IDL for the class of that object, e.g.:

```
    IDL> a=obj_new('idl_container')
    IDL> a->[Tab]
    "Select a procedure-method name (class is IDL_Container):"
```

  The class is stored as a property on the "->" for rapid recovery if custom variable `idlwave-store-inquired-class' is set, and is even recallable from the input history (just look for the bold arrow).

- Much improved class and structure tag completion and help. When completing class tags (e.g. self.[Tab]) or structure tags (when using the optional completion module idlw-complete-structtag), you can now right-click on an item to pop-up its source definition in the online Help window. Invoking context help on class tags (either with key command [M-S-?], or just with a shift-right-click on a self.tag expression) works too. This even works for inherited tags defined in superclasses (in which case the class of the tag will be appended in the completions buffer)! Never scroll to the bottom of your file, or hunt about for the superclass file(s) to find out what one of your class or structure tags means again.

- Conditional breakpoints are now easily set using a simple prefix-argument to the breakpoint setting command (e.g. [C-u C-c C-d C-b], or, e.g., [C-u H-d], for those who set up an alternate prefix modifier). Conditional breakpoints allow any IDL expression (like "i gt 25") to be used as a break condition. You can still of course set one-time or "after" breakpoints using the same easy numeric prefixes (e.g. [C-4 C-c C-d C-d]).

- Filename completion in the shell now works after batch `@' commands.

- A custom variable `idlwave-shell-show-commands' allows you to control which of the background commands sent to the shell are printed, including compiling, breakpoint, debug, and miscellaneous commands. A special choice ('everything') allows you to have a peek at all the traffic IDLWAVE is sending to the shell (for the insatiably curious). Pay no attention to the man behind the curtain.

- When using stepping commands "Continue to Here" and "Continue to End of Block", all other breakpoints are temporarily disabled to ensure you reach the desired break location.

- XEmacs binary "packages" of IDLWAVE are now available for download on each release.

====> Bug fixes and the like: <=====

- Filename completion in the shell and buffer now works better, fixing the occassional appearance of cruft at the end of your filename. Also, you can now complete blank filenames (e.g. open,'[Tab]).

- Now compiles correctly with "downgrade" versions of help.

- Multi-threading keywords are now correctly linked for both routine info and help.

- Certain syntax errors weren't being parsed correctly: fixed.

- Completion of tags in `self', and in other structures (like info), when the optional completion module `idlw-complete-struct-tag' is used, now catches a few tags which were being missed.

- Alignment on continued assignment statements now permits a more complex LHS (e.g. a.x[4] = $ ...).

- An incorrect non-matching-end error when using multi-word end abbreviations like "\ei" ("endif else if") was quelled. Begin/end block showing now works more smoothly.

- Fixed issue on completing system variable structure tags (like !D.TABLE_SIZE) when the optional `idlw-complete-struct-tag' completion module is also in use.

- Completing procedures on multi-statement lines (with `&') now works correctly.

- Indenting continuation lines which contain the text "pro" or "function" somewhere in their argument list now works correctly.

As always, all this and more is available for the taking at:

http://idlwave.org

Happy coding!

JD