NEWS

neuville

05 dmbre 2010

Contents

1 Changes for ECB version 2.40.1 **1**

1.1 Better compatibility with forthcoming Gnu Emacs 23.2 1

1.1.1 Works out of the box with the Emacs-integrated CEDET-suite 1

1.2 The special symboldef ECB-window is now stable and ready for usage 2

1.2.1 **TODO** description what is new 2

1.3 ECB now uses the partial reparse feature of semantic 2

1.4 ECB eliminates duplicates in the directories-buffer 2

1.5 New default-value for 'ecb-auto-update-methods-after-save': nil 2

1.6 The contents of the methods-window are updated even if currently hidden 2

1.7 New support for Bazaar as version-control system 3

1.8 New default value of the option 'ecb-history-make-buckets' . . 3

1.9 'defecb-window-dedicator-to-ecb-buffer' replaces 'defecb-window-dedicator' 3

1.10 Fixed Bugs 3

1.10.1 Command 'delete-windows-on' has only deleted one window 3

1.10.2 Fixed a bug which prevents the ecb-windows from being vertically resizeable 3

1.10.3 Fixed a bug concerning horizontal mouse-scrolling of ecb-windows via 4

1.10.4 Fixed a bug which prevented 'ecb-type-tag-display' from taking effect. 4

1.10.5 Killing a buffer failed when history-window is not visible 4

1.10.6 All compile-modes (e.g. compile, grep etc.) failed in XEmacs 4

2 Changes for ECB version 2.40 **4**
2.1 ECB now requires full CEDET-suite being installed (at least version 1.0pre6) . . . 4
2.2 Completely reworked synchronizing mechanism of the ECB-windows . . . 4
2.3 Better compatibility with CEDET >= 1.0pre6 and semantic >= 2.0pre6 . . . 5
2.3.1 The ECB-analyse-window now works very well with current semantic analyzer . . . 5
2.3.2 Fix compatibility about some changes of semantic concerning adopting . . . 5
2.3.3 ECB now works very well with semantic-idle-scheduler-mode (s.a.) . . . 6
2.4 More user-responsible buffer-parsing based on the idle-mechanism of semantic . . . 6
2.5 Complete reworked history-buffer . . . 7
2.5.1 The history is able to deal with indirect-buffer entries. 7
2.5.2 The history can now be bucketized, see new 'ecb-history-make-buckets'. . . . 7
2.5.3 There are now new faces for the history entries. . . . 7
2.6 Sticky parent-node for all special ECB-windows of type tree-buffer . . . 7
2.7 Much saver advice-backbone for all advices needed by ECB . 8
2.8 Changes in the version control support of ECB . . . 8
2.8.1 ECB now also offers version control support only if the current vc-package is . . . 8
2.8.2 New support for Git and Monotone as version-control systems . . . 8
2.8.3 'ecb-vc-supported-backends' offers now also a accurate recompute-state function . . . 8
2.8.4 CVS-support is now fully compatible with Emacs 22 and 23 . . . 8
2.9 New options rsp. commands . . . 8
2.9.1 New command 'ecb-goto-window-edit-by-smart-selection' 8
2.9.2 New command 'ecb-goto-window-ecb-by-smart-selection' 8
2.9.3 New option 'ecb-ignore-pop-up-frames' . . . 8
2.10 Much better compatibility with current Emacs-versions . . . 9
2.10.1 Full compatibility with Gnu Emacs 22 . . . 9
2.10.2 Much better compatibility with forthcoming Gnu Emacs 23 . . . 10

2.10.3 Better compatibility with XEmacs 21.4 and XEmacs 21.5 . . . 11
2.11 ECB is able to work with indirect buffers if the base-buffer is filebased . . . 11
2.12 Deprecated, disabled and deactivated features: . . . 12
2.12.1 The commands 'ecb-download-ecb' and 'ecb-download-semantic' are deactivated . . . 12
2.13 Fixed Bugs . . . 12
2.13.1 Fixed a bug which displayed empty tag-names in the methods-buffer when font-lock . . . 12
2.13.2 Fixed a bug in the ping-functionality . . . 12
2.13.3 Fixed a bug which prevented 'ecb-rebuild-methods-buffer' from working . . . 12
2.13.4 Fixed a bug in the internals of the methods-filter which prevented . . . 12
2.13.5 Fixed context-menu for VC-operations in the sources- and history-window . . . 12
2.13.6 Fixed a bug with the ECB-navigation feature (C-c . p and C-c . n) . . . 12
2.13.7 Fixed small bugs with synchronizing current tag with highlighted tag . . . 13
2.13.8 Fixed a bug in 'ecb-create-new-layout' which has displayed wrong modelines . . . 13
2.13.9 Fixed a bug in 'ecb-submit-problem-report' which prevented it from working . . . 13

3 Changes for ECB version 2.32 13
3.1 New backbone-add-on of ECB for getting the definition of the symbol at point . . . 13
3.2 New native ECB-window (tree-buffer) for the semantic-analyzer 14
3.3 New smart-arrow-navigation in a tree-buffer with up- and down-arrow . . . 14
3.4 Better handling of maximized ecb-windows . . . 14
3.4.1 Smarter handling of node-selections in maximized tree-windows . . . 14
3.4.2 Easier maximizing and "minimizing" . . . 15
3.5 Renamed options or options with new type . . . 15
3.6 Better assistance fixing errors related to semantic-load problems 15
3.7 More robust downloading of new ECB from within Emacs . . 15
3.8 Support for (X)Emacs < 21 has been officialy removed. 16

3.9 Full documentation of the library tree-buffer.el 16
3.10 Fixed Bugs 16
3.10.1 Version-controlled (VC) and not-VC history-entries are now sorted 16
3.10.2 Handles not anymore existing files in the stealthy tasks. 16
3.10.3 Fixed a bug when 'ecb-process-non-semantic-files' is set to nil. 16
3.10.4 Fixed a bug in displaying semantic-tags which have plain-string members 16
3.10.5 Fixed a bug when Emacs is started with -q so Emacs could not determine the 17

4 Changes for ECB version 2.31 **17**
4.1 Fixed bugs 17
4.1.1 Fixed a recently introduced bug with XEmacs and an active isearch. 17
4.1.2 Fixed a bug concerning accessing root-drives in native window-XEmacs 17
4.1.3 Better recognizing of remote CVS-root-repositories for the VC-support 17
4.2 Enhancement for the VC-support 17
4.2.1 New option 'ecb-vc-xemacs-exclude-remote-cvs-repository' 17
4.2.2 Added beta-support for Clearcase 18

5 Changes for ECB version 2.30.1 **18**
5.1 Enhancement to the automatic option-upgrading mechanism . 18
5.2 Enhancement to the VC-support 18
5.2.1 Better recomputing of the VC-state of a file when state changed outside 18
5.2.2 Added out-of-the-box support for VC-system Subversion 19
5.3 Fixed bugs 19
5.3.1 Fixed errors occured at load-time of ECB 2.30 19

6 Changes for ECB version 2.30 **19**
6.1 Enhancements to the file-browser 19
6.1.1 Much better performance of the file-browser display because all 19
6.1.2 ECB is now capable of handling remote paths. 20
6.1.3 ECB displays the Version-control-state of a file in the tree-buffers. 20

6.1.4 New hook which runs directly after the selected directory has changed. 20
6.2 The popup-menu of the methods-browser allows precisely expanding of the 20
6.3 Automatically upgraded ecb-option-settings are now not saved by default. 21
6.4 With XEmacs ECB temporary sets 'progress-feedback-use-echo-area' to t 21
6.5 Fixed bugs 21
6.5.1 Fixed resizing of the ecb-windows after opening a file . 21
6.5.2 Fixed a bug in the upgrading feature (command 'ecb-download-ecb') which has 21
6.5.3 Fixed a bug in restoring sizes of the ecb-windows. . . . 22

7 Changes for ECB version 2.27 **22**
7.1 The option 'ecb-auto-expand-tag-tree-collapse-other' now has three possible 22
7.2 Not accessible directories are displayed in a different face in the 22
7.3 Enhancements to the permanent compile-window 22
7.3.1 Enlarging the compile-window does not destroy some ecb-windows 22
7.3.2 Now 'ecb-store-window-sizes' and 'ecb-restore-window-sizes' also work when 23
7.3.3 Now ECB ensures that the permanent compile-window has always its specified 23
7.3.4 Now ECB ensures that all ecb-windows have always either their default or 23
7.3.5 ECB takes into account the new Emacs-option 'grep-window-height' when it 23
7.4 Fixed Bugs 23
7.4.1 Added a workaround for a bug in the custom-library of current release 21.3 23

8 Changes for ECB version 2.26 **24**
8.1 Improved the performance of the directories-buffer-display . . 24
8.2 Per default all mouse-actions in the special ECB-buffers are now triggered 24
8.3 ECB sets autom. temp-buffer-resize-mode rsp. temp-buffer-shrink-to-fit when 24
8.4 Some regexp-options has been changed to regexp-list-options: 24

8.5 New option 'ecb-history-exclude-file-regexps' for excluding certain files . . . 25
8.6 ECB now displays read-only sourcefiles in a different face. . . 25
8.7 Fixed bugs . . . 25
8.7.1 Fixed merging faces with XEmacs. . . . 25
8.7.2 Now the command 'ecb-expand-methods-nodes' and all the expand-* menu-entries . . . 25
8.7.3 Fixed a bug concerning 'ecb-auto-expand-tag-tree-collapse-other'. . . . 25
8.7.4 Handles not existing source-paths in 'ecb-source-path' correct. . . . 26

9 Changes for ECB version 2.25 **26**
9.1 More flexible sorting of the Sources- and the History-buffer . 26
9.1.1 'ecb-sort-history-items' has been renamed to 'ecb-history-sort-method' . . . 26
9.1.2 Two new options 'ecb-sources-sort-ignore-case' and . . 26
9.2 New icons for parent-display in the Methods-buffer. . . . 26
9.3 Fixed Bugs . . . 26
9.3.1 Fixed an annoying bug which results sometimes in an error "stack-overflow . . . 26
9.3.2 Fixed a bug in the mechanism which prescanes directories for emptyness. . . . 26
9.3.3 Fixed a bug in the autom. option-upgrading-mechanism. 27

10 Changes for ECB version 2.24 **27**
10.1 Enhancements for the Methods-buffer . . . 27
10.1.1 Enhancements to the icon-display in the Methods-buffer 27
10.1.2 New option for switching on/off distinction of function-prototypes from the . . . 27
10.1.3 New "current-type"-filter for the Methods-buffer . . . 27
10.2 Now directories are prescanned for emptyness so they are displayed as empty . . . 28
10.3 Fixed Bugs . . . 28
10.3.1 Fixed some newly introduced bugs which made ECB 2.23 incompatible with . . . 28
10.3.2 Fixed a bug which changed the highlighted directory even after just expanding . . . 28
10.3.3 Fixed two small bugs which have influenced 'ecb-create-new-layout' . . . 28

11 Changes for ECB version 2.23 **28**
11.1 New cedet1.0beta2 is supported. 28
11.2 Enhancements for the Methods-buffer 28
11.2.1 The lookfeel of the Methods-buffer is now much nicer because it has been 28
11.2.2 New feature for applying default tag-filters for certain files. 29
11.2.3 New keybindings and commands for faster applying certain tag-filters. 29
11.2.4 The popup-menu contains now mode-dependend tag-filter entries. 29
11.2.5 The option 'ecb-show-tags' is now defined on a major-mode-basis. 30
11.2.6 Distinction between functions and function-prototypes in the Methods-buffer 30
11.3 The command 'ecb-toggle-layout' now has a prefix-argument: 30
11.4 Better internal self-monitoring of ECB 30
11.5 New option group for the integrated speedbar and one new option: 30
11.6 Added a new section in the info-manual which describes how desktop.el can be 31
11.7 Fixed bugs 31
11.7.1 Double-clicking the mouse-1-button now works with integrated speedbar. 31
11.7.2 Fixed two small bugs in the advice of 'display-buffer': 31
11.7.3 Fixed a bug so now always the correct directory in the directories-buffer 31

12 Changes for ECB version 2.22 **31**
12.1 New nifty feature for filtering the tags displayed in the Methods-buffer 31
12.2 All tree-buffer filters are now guarded against filtering out all tags. 32
12.3 Much smarter mechanism to highlight the current tag in the methods-buffer. 32
12.4 The popup-menus can now be nested into 4 levels of sub-menus. In general there 33
12.5 The indication for ECB in the minor-mode is now hidden when the ECB-windows 33
12.6 Compatibility enhancements 33

12.6.1 Cause of the fixed bugs the compatibility for 'bs-show' has increased when . . . 33
12.7 Fixed Bugs . . . 33
12.7.1 Fixed a bug preventing the native Windows-port of XEmacs from working. . . . 33
12.7.2 Makes the behavior of adviced 'display-buffer' fully compatible to the . . . 33
12.7.3 Fixed a small bug in 'ecb-toggle-compile-window-height'. This bug has . . . 34
12.7.4 Fixed a bug related to the speedbar-integration. . . . 34
12.7.5 Fixed a bug for the special ECB-modeline-menu for the XEmacs-modelines. . . . 34
12.7.6 Fixed a bug which occurs with XEmacs in combination with func-menu. . . . 34

13 Changes for ECB version 2.21 **34**
13.1 Advice for 'balance-windows' so only the edit-windows are balanced. . . . 34
13.2 New option 'ecb-ignore-display-buffer-function' . . . 34
13.3 Compatibility enhancements . . . 35
13.3.1 Commands like 'bs-show' of the library bs.el now work correctly with ECB . . . 35
13.3.2 Xrefactory works even when all ECB-windows are visible (see Fixed Bugs). . . . 35
13.3.3 Applications like Gnus or BBDB run withing the ECB-frame without conflicts - . . . 35
13.3.4 Commands using 'Electric-pop-up-window' now work correctly with ECB. . . . 35
13.4 For XEmacs the package fsf-compat is no longer required by ECB. . . . 35
13.5 Fixed Bugs . . . 35
13.5.1 Fixed a fatal bug which prevents 'other-window' from working with . . . 35
13.5.2 If point stays in the current source-buffer on a function argument then still . . . 36
13.5.3 ECB now uses 'compilation-window-height' correctly when set buffer-local . . . 36
13.5.4 Fixed a bug in 'ecb-sources-filter' and 'ecb-history-filter'. 36

14 Changes for ECB version 2.20 **36**
14.1 New customization group "ecb-most-important" 36
14.2 The keybinding for the online-help has changed from [C-c . o] to [C-c . h] 36
14.3 The options 'ecb-major-modes-activate' and 'ecb-majors-mode-deactivate' have 36
14.4 The restriction of only two edit-windows has been gone! . . . 36
14.4.1 Beginning with this version 2.20 there are no restrictions about the 36
14.4.2 Option 'ecb-split-edit-window' has been renamed in . . 37
14.4.3 Compile-window can now be displayed even when the ECB-windows are hidden. 37
14.4.4 The option 'ecb-primary-mouse-jump-destination' has been renamed to 37
14.4.5 New keybinding [C-c . g l] for selecting the last selected edit-window. 37
14.4.6 Option 'ecb-other-window-jump-behavior' has been renamed to 37
14.4.7 New option 'ecb-scroll-other-window-scrolls-compile-window' and new 38
14.4.8 Higher compatibility of ECB with other packages . . . 38
14.5 Enhancements to the tree-buffers of ECB 38
14.5.1 Enhancements to the popup-menus of the tree-buffers 38
14.5.2 ECB supports the default modeline-mechanisms for deleting other windows. 39
14.5.3 Now all commands for selecting a tree-buffer work even when this buffer is 39
14.5.4 New option 'ecb-maximize-ecb-window-after-selection'. 39
14.5.5 Changes to the image-icons of the tree-buffers 40
14.6 New default-value for 'ecb-post-process-semantic-taglist'. . . . 40
14.7 Changes for the compile-window 40
14.7.1 The command 'ecb-toggle-compile-window' now displays a compile-window even if 40
14.7.2 New option 'ecb-change-layout-preserves-compwin-state'. 41
14.7.3 Adviced 'delete-window' and 'delete-other-windows' handle compile-window. 41
14.8 New option 'ecb-activation-selects-ecb-frame-if-already-active'. 41
14.9 The command 'ecb-redraw-layout' can be called with prefix-argument(s). 41
14.10New command 'ecb-display-news-for-upgrade'. 41

14.11 Fixed Bugs . . . 41
14.11.1 Adviced 'display-buffer' handles 'same-window-buffer-names' and . . . 41
14.11.2 Fixed a bug preventing tree-buffers with expand-symbol "before" to work . . . 42
14.11.3 Fixed a bug in the adviced version of 'other-window'. . 42
14.11.4 Fixed a bug when scrolling a window from within the minibuffer. . . . 42
14.11.5 Toggling the compile-window has not worked when 'ecb-compile-window-height' . . . 42
14.11.6 'ecb-cycle-through-compilation-buffers' works with hidden compile-window. . . . 42
14.11.7 Fixed a bug when an ECB-window was maximized and the user has stored sizes . . . 42
14.11.8 Fixed a bug in 'ecb-major-modes-activate' and 'ecb-major-modes-deactivate' . . . 42
14.11.9 Fixed a bug in the handling of the option 'ecb-kill-buffer-clears-history' . . . 42
14.11.10 Fixed a bug which prevents some older version of XEmacs 21.4 working well. . . . 42

15 Changes for ECB version 2.11 **43**
15.1 Using semanticdb to jump to type-tags defined in other files. . 43
15.2 New option 'ecb-ignore-special-display' . . . 43
15.3 Better reporting if there are errors during ECB-startup 44
15.4 Fixed bugs . . . 44
15.4.1 Clicking onto nodes with positionless semantic-tags as data doesn't fail . . . 44
15.4.2 Non-semantic sources (e.g. Perl, TeX) work when cedet 1.0 is loaded . . . 44
15.4.3 Fixed the versioning of ECB so the autom. upgrade works . . . 44

16 Changes for ECB version 2.01 **44**
16.1 Prepared for the new semantic 2.0 beta (contained in the cedet 1 beta) . . . 44
16.1.1 New naming convention . . . 44
16.1.2 ECB works with new semantic 2.0 which is shipped within cedet 1.0 . . . 45
16.2 Overhaul of the display-layer and -handling of the tree-buffers 45

16.2.1 Now there are 3 different styles available 45
16.2.2 No-leaf-tree-nodes with currently no subnodes are displayed with [x] 45
16.2.3 Fixed some inconsistencies in expanding, collapsing and selecting nodes 45
16.2.4 Moving all options related to the tree-buffer-style or -handling to the new 45
16.3 Completely rewritten popup-menu mechanism. 45
16.3.1 Now sub-menus are allowed for all popups 45
16.3.2 Added new defaults to 'ecb-directories-menu-user-extension' 46
16.3.3 Added new defaults to 'ecb-sources-menu-user-extension' 46
16.4 Possibility to define which types should be expanded at file-open-time 46
16.5 Adding a filter-feature to the Sources- and the History-buffer 46
16.6 Changes related to the modelines of the tree-buffers 47
16.6.1 Added faces to the mode-lines of the ECB-tree-buffers. 47
16.6.2 Moved all options related to the modelines in the new customize-group 47
16.7 No window-restrictions if the ECB-windows are hidden 47
16.8 Ediff runs per default in the ECB-frame. 47
16.9 More flexible directory-caching with new 'ecb-cache-directory-contents-not' 47
16.10New option 'ecb-advice-window-functions-signal-error' 47
16.11Fixed Bugs 48
16.11.1 When the special ECB-windows were hidden and a durable compile-window was set 48
16.11.2 XEmacs has not added the ECB-menu to the menubar of that buffers which are 48
16.11.3 JDEE-dialogs now work correctly. 48
16.11.4 A bug in displaying the tree-buffers in a different font or font-height 48

17 Changes for ECB version 1.96 **48**
17.1 ECB can work together with the window-managers escreen and winring 48
17.2 ECB can display the window-number in the modeline of the special windows. 49
17.3 Much better support of the ECB-compile-window 49
17.3.1 General overhaul for displaying buffers in the ECB-compile-window. 49

17.3.2 New layout-option 'ecb-compile-window-width'. 49
17.3.3 New layout-option 'ecb-compile-window-prevent-shrink-below-height' 50
17.3.4 Better and more reliable eshell-integration. Now the compile-window fits 50
17.3.5 All options related to the ECB-compile-window have been moved to the new 50
17.4 Safer handling of all adviced functions of ECB 50
17.5 ECB can visit tokens in a smarter way 50
17.5.1 Narrowing via popup-menu of the Methods-buffer includes now the documentation 50
17.5.2 More flexible definition what to do after visiting a token via the 50
17.6 ECB selects autom. the current default-directory after activation even if 50
17.7 ECB preserves the split-state of the frame after activation rsp. deactivation 51
17.8 New command 'ecb-toggle-window-sync' for fast toggling between autom. 51
17.9 The popup-menu of the Directories-buffer allows opening a dir with Dired. 51
17.10 Now manually resizing of the ECB-windows is possible even if 51
17.11 ECB does not call any functions of the cl-library at runtime. . 51
17.12 Fixed Bugs 51
17.12.1 Fixed all the file- and directory-commands in the popup-menus. 51
17.12.2 Jumping to some tokens for Perl sources failed. This is fixed now. 51
17.12.3 Now the buffer-names displayed in the History-window are always uptodate, ... 51
17.12.4 Fixed a conflict between ECB and 'tmm-menubar' (tmm.el) 52
17.12.5 Fixed a lot of misspellings in the ECB-online-help and also in all comments 52
17.12.6 Fixed a bug in 'ecb-toggle-enlarged-compilation-window'. 52
17.12.7 Fixed a bug in the layout-engine related to dedicated windows. 52
17.12.8 Fixed some references in the info-manual of ECB. . . . 52

18 Changes for ECB version 1.95.1 52

18.1 Now the ecb-windows can be "maximized", means all other ecb-windows are . . . 52
18.2 Some default key-bindings have changed; see RELEASE$_NOTES for details.$ 53
18.3 All popup-menus of the ECB-tree-buffers can be opened via [Meta-m] . . . 53
18.4 Fixed bugs . . . 53
18.4.1 Clicking onto the image-icons of an integrated speedbar has failed for . . . 53

19 Changes for ECB version 1.95 **53**
19.1 ECB now displays the expand- and collapse symbols in the tree-buffers with . . . 53
19.2 Adding hideshow to the popup-menu of the Methods-buffer. . 53
19.3 Horizontal scrolling of the tree-buffers by clicking the edges of the . . . 53
19.4 Changed default key-bindings: . . . 53
19.5 Starting ECB is now possible via the "Tools"-menu . . . 54
19.6 ECB now requires speedbar version 0.14beta1 or higher. An automatic . . . 54
19.7 Fixed bugs . . . 54
19.7.1 Now the "goto-window-..." menu-entries in the "ECB"-menu are working correct . . . 54
19.7.2 Preventing etags-supported sources from being parsed (and saved) too often. . . . 54

20 Changes for ECB version 1.94 **54**
20.1 Supporting of non-semantic-sources. . . . 54
20.1.1 Native parsing and displaying source-contents of imenu supported files . . . 54
20.1.2 Native parsing and displaying source-contents of etags supported files . . . 54
20.1.3 There are some new options like 'ecb-process-non-semantic-files', . . . 54
20.2 Better speedbar integration into the ECB-frame . . . 54
20.2.1 Now speedbar can be used not only as replacement for the . . . 54
20.2.2 Now speedbar can be integrated into an arbitrary layout in the same way as . . . 55
20.2.3 New option 'ecb-directories-update-speedbar' which helps you . . . 55

20.3 New option 'ecb-compilation-predicates' for better specifying which buffers . . . 55
20.4 Better customizing of the mode-line of an ECB-tree-buffer. . . 55
20.5 Fixed bugs: . . . 55
20.5.1 Using "Add source path" from the popup-menu now doesn't open the Windows . . . 55
20.5.2 Smart arrowkey-navigation in the tree-buffers is now even smarter. . . . 55
20.5.3 If ECB was activated with a layout which does not contain a . . . 55

21 Changes for ECB version 1.93 56
21.1 Fixed bugs: . . . 56
21.1.1 The adviced version of 'other-window' now works correct too if a . . . 56
21.1.2 The commands of 'winner-mode' are now only forbidden in the ECB-frame . . . 56
21.1.3 'scroll-all-mode' now works correct in other frames than the ECB-frame. . . . 56
21.1.4 Removed the annoying message "Partial reparsing..." . 56
21.2 Enhancements to the layout: . . . 56
21.2.1 Slightly better redrawing current layout especially if the layout contains . . . 56
21.2.2 ECB autom. displays hidden compile-window before starting a compilation. . . . 56
21.2.3 'switch-to-buffer-other-window' now works more naturally within the . . . 56
21.2.4 'ecb-toggle-compile-window' now has a default key-binding: [C-c . . . 57
21.2.5 Now also layouts with user-defined special ecb-windows can be created . . . 57
21.3 Enhancements to the tree-buffers: . . . 57
21.3.1 Now user-extensions can be added to the popup-menus of the tree-buffers. . . . 57
21.3.2 The methods-buffer now has a popup-menu with senseful actions . . . 57
21.4 New option 'ecb-sources-exclude-cvsignore' which allows to exclude files from . . . 57
21.5 ECB now also works with buffers "online" extracted from archives. . . . 57

21.6 ECB now checks at load-time if the required packages semantic and eieio are . . . 57

22 Changes for ECB version 1.92.1 58
22.1 Fixed bugs: . . . 58
22.1.1 Fixing eating up a lot of CPU-power just for updating the menu- . . . 58
22.1.2 Fixed a bug which always truncates lines in the ECB-windows regardless . . . 58
22.1.3 The upgrade library of ECB now explicitly requires and loads the . . . 58
22.1.4 'ecb-toggle-compile-window' now works also correct if called with . . . 58

23 Changes for ECB version 1.92 58
23.1 Fixed bugs: . . . 58
23.1.1 Fixed small bugs in the popup-menu of the history-buffer. . . . 58
23.1.2 Fixed some not working links in the FAQ. . . . 58
23.2 ECB now includes a file "ecb-autoloads.el" which contains all available . . . 58
23.3 Enhancements to the layout . . . 59
23.3.1 The sizes of the ecb-windows can be fixed even during frame-resizing . . . 59
23.3.2 The command 'ecb-store-window-sizes' can now also store fixed sizes. . . . 59
23.3.3 The command 'split-window' is now also adviced to work properly. . . . 59
23.4 Enhancements for the tree-buffers . . . 59
23.4.1 "Create Source" in the popup-menus now work also for non-java-files. . . . 59
23.4.2 ecb-truncate-lines can now be set different for each ECB-tree-buffer. . . . 59
23.4.3 Easy horizontal mouse-scrolling of the ECB tree-buffers. See new option . . . 59
23.5 Added a command 'ecb-jde-display-class-at-point' which displays the contents . . . 59
23.6 Renamed previous HISTORY file to NEWS and reformated it for outline-mode. . . . 60

24 Changes for ECB version 1.91.1 **60**
24.1 Fixed bugs: 60
24.1.1 Fixes a bug in the speedbar integration which prevented clicking onto a 60
24.1.2 Fixes a bug in upgrading stored window-sizes from ECB 1.80 to ECB >= 1.90. 60
24.1.3 Fixed a bug in the navigation-history of ECB which sometimes has prevented 60
24.2 Enhanced the downloading feature of ECB 60
24.2.1 Fixed a bug in the downloading feature of ECB. (Klaus) 60
24.2.2 Now ECB autom. checks at the download sites which versions are available. 60
24.2.3 New command 'ecb-download-semantic' for easy getting latest semantic 61

25 Changes for ECB version 1.90 **61**
25.1 The website of ECB has moved from http://home.swipnet.se/mayhem/ecb.html to 61
25.2 Fixed bugs: 61
25.2.1 Fixed an annoying bug which results in an error "Wrong type argument: 61
25.2.2 Fixed a small bug in 'ecb-add-all-buffers-to-history': If 61
25.2.3 Fixed some bugs concerning the eshell-integration (Klaus): 61
25.3 General enhancements: 62
25.3.1 Now ECB displays at every start a "Tip of the day" in a dialog-box. This can 62
25.3.2 Advicing 'scroll-all-mode' so it works correct with ECB, means 62
25.3.3 Autom. preventing 'winner-mode' from being activated as long as ECB is 62
25.3.4 Now autom. (de)activating ECB via the options ecb-major-modes-activate and 62
25.3.5 Now ECB can be "autoloaded", i.e. 62
25.3.6 Silencing the byte-compiler, i.e. byte-compilation of ECB should now be 63
25.4 Enhancements to the methods/variable/token tree-buffer: . . 63
25.4.1 New feature: Now the methods buffer is auto. expanded if the node related to 63
25.4.2 New command 'ecb-expand-methods-nodes' which allows precisely expanding 63

25.5 Enhancements to the directories tree-buffer: 63
25.5.1 New command ‘ecb-expand-directory-nodes’ which does the same for the 63
25.5.2 Rewritten cache-mechanism for directories and sources: 63
25.6 Enhancements to the sources tree-buffer: 64
25.6.1 The option ecb-source-file-regexps now works on a directory-base, which means 64
25.7 Enhancements to the layout-customization: 64
25.7.1 Naming and managing of layouts has been changed! Now a layout is not longer 64
25.7.2 Adding a new layout type "left-right" which allows the ECB-tree-windows to 65
25.7.3 Now ECB offers a command ‘ecb-create-new-layout’ for interactively creating 65
25.7.4 Rewritten the mechanism for storing customized window-sizes. See option 65
25.8 General Enhancements of the tree-windows: 66
25.8.1 Introduced a POWER-click (was SHIFT-click before) in the ECB-tree-windows - 66
25.8.2 Now the ECB-tree-buffers offer in their popup-menus a "Grep" and a "Grep 66
25.8.3 Modified syntax-table for all tree-buffers with all paren, braces and 66
25.9 Enhancements to the layout-engine of ECB: 66
25.9.1 Added a complete new section "The layout-engine of ECB" to the online-help 66
25.9.2 Completely rewritten how to program new layouts. Now there is a new macro 66
25.10 Enhancements to the compile-window 67
25.10.1 ECB now has a list of valid compilation buffers within the ECB menu. (Kevin) 67
25.10.2 ECB now supports ‘ecb-enlarged-compilation-window-max-height’ which can be 67
25.10.3 Added new command ’ecb-toggle-compile-window’: Toggle the visibility of the 67
25.11 Speedbar is now integrated in ECB and can be used instead of the standard 67
25.11.1 See new option ’ecb-use-speedbar-for-directories’ and the node "Integrating 67

25.11.2 Version-checking for correct speedbar version (>= 0.14beta1). For lower . . . 67
25.11.3 ecb-speedbar and speedbar first loaded direct before drawing the layout . . . 67
25.11.4 The speedbar-commands speedbar, speedbar-get-focus are adviced during . . . 67
25.12 Enhancements to the ehell-support of ECB: . . . 68
25.12.1 ecb-eshell now recenters after command execution. (burton) . . . 68
25.12.2 ecb-eshell now recenters if there are any window resizes. This is done so . . . 68
25.12.3 ecb-eshell integration now protects against buffer-list modifications during . . . 68
25.12.4 Added online documentation for the eshell integration. See section . . . 68
25.13 New hooks: . . . 68
25.13.1 New or renamed hooks: . . . 68
25.13.2 Added two general hooks: . . . 69
25.13.3 Added 5 new hooks for the ECB tree-buffers: . . . 69

26 Changes for ECB version 1.80 69
26.1 ECB now requires: . . . 69
26.2 Fixed bugs: . . . 69
26.2.1 Fixed: Now no empty tool-tips are displayed if mouse moves over nodes in the . . . 69
26.2.2 Fixed a bug in ecb-eshell. If we are running with current buffer sync, and . . . 70
26.2.3 Fixed a bug in ecb-speedbar. The speedbar was doing its own updating outside . . . 70
26.2.4 Fixed a bug: Sometimes there occurs a a "wrong-type-argument"-error (or . . . 70
26.2.5 Fixed formatting of "Parent" nodes in token buffer. (Jesper) . . . 70
26.3 Byte compiling ECB either interactive with ‘ecb-byte-compile’ or with the . . . 70
26.4 ECB now displays the section "First steps" of the online-manual after . . . 70
26.5 Now the adviced versions of ‘delete-window’ and ‘delete-other-windows’ can . . . 70

26.6 New adviced window command: 'delete-windows-on' now works correct with ECB. . . . 71
26.7 In the ECB-directories buffer now F4 adds a new source-path instead of F1. . . . 71
26.8 Enhancing the option 'ecb-window-sync': Now the synchronization can take . . . 71
26.9 Now the ECB online help is available in the standard Info-format and also in . . . 71
26.10 New feature and command: ecb-download-ecb. With this function you can . . . 71
26.11 New debug mode for ECB which prints some debug informations in the message . . . 71
26.12 Some default key-bindings have changed; see 'ecb-key-map'. (Klaus) . . . 72
26.13 New feature: Possibility to define a sequence of layouts with new option . . . 72
26.14 Two new hooks: ecb-hide-ecb-windows-hook and ecb-show-ecb-windows-hook. Read . . . 72
26.15 Two new options: ecb-major-modes-activate and ecb-major-modes-deactivate. . . . 72
26.16 Redefining of the option 'ecb-compile-window-temporally-enlarge': Now you . . . 72
26.17 Much better and more powerful auto. upgrading-mechanism of incompatible or . . . 72
26.18 New option ecb-tree-RET-selects-edit-window: Possibility not to select the . . . 72
26.19 Better grouping of external methods, i.e. methods which are implemented . . . 73
26.20 For application programmers using the tree-buffer library: . . . 73

27 Changes for ECB version 1.70 **73**
27.1 New option ecb-bucket-token-display. See doc-string for details. (Klaus) . . . 73
27.2 Reorganization of the faces used by ECB. The group 'ecb-face-options' . . . 73
27.3 Smarter incr. search in a tree-buffer with the [end]-key: If there are at . . . 73
27.4 New feature ecb-auto-compatibility-check: ECB checks on startup if the . . . 74

27.5 New feature ecb-tree-navigation-by-arrow: Smarter navigation in the . . . 74
27.6 New feature 'ecb-type-token-display'. See the doc-string of this new option . . . 74
27.7 Now the option 'ecb-post-process-semantic-tokenlist' has per default an . . . 74
27.8 Now in the ECB-tree-buffers RETURN works like the primary button (already in . . . 74
27.9 New feature: Now the token-display function can be defined separated for . . . 74
27.10 Fixed a bug with mouse-avoidance-mode (only relevant for GNU Emacs 20.X) . . . 75
27.11 New feature, ecb-select-edit-window-on-redraw. Mostly used if you are running . . . 75
27.12 New feature, ecb-auto-activate. If non-nil we can start the ECB after Emacs . . . 75
27.13 New package, ecb-cycle.el. Supports cycling through all known compilation . . . 75
27.14 Much better performance of the directory and file-browser in ECB, especially . . . 75
27.15 ecb-eshell now supports customization. (Kevin) . . . 75
27.16 New feature: ecb-layout-always-operate-in-edit-window. Look at the . . . 75
27.17 Fixed a bug in 'ecb-toggle-ecb-windows'. For starters we weren't setting . . . 76
27.18 Fixed bug in 'ecb-redraw-layout-quickly'. If for come reason any of the ECB . . . 76
27.19 New function ecb-toggle-enlarged-compilation-window allows users to expand and . . . 76
27.20 new option ecb-add-path-for-not-matching-files. Look at the documentation . . . 76

28 Changes for ECB version 1.60 **76**

28.1 ecb-highlight-token-header-after-jump is now a boolean; which face is used . . . 76
28.2 New option 'ecb-post-process-semantic-tokenlist': Special post-processing of . . . 76
28.3 Now also with C++ Sources the method- and variable-protection is displayed . . . 77

28.4 Fixed a bug which prevented ECB using the root path "/" on Unix-like systems . . . 77
28.5 If a file via the standard-mechanisms of Emacs (e.g. find-file) is opened, . . . 77
28.6 Added ecb-redraw-layout-hooks so that code can be executed after we redraw the . . . 77
28.7 New hook ecb-current-buffer-sync-hook allows developers to add code to be . . . 77
s own stuff are77subsection.28.8
28.9 ECB is now fully prepared for Emacs 21. The new feature . . 77
28.10The options 'ecb-show-node-name-in-minibuffer', . . . 78
28.11ecb-auto-expand-directory-tree now offers two options: . . . 78
28.12Fixed a bug in auto. expanding directories if a source-path has an alias. . . . 78
28.13If mouse is moved over an alias in the directories buffer then the real path . . . 78
28.14If mouse is moved over an history-entry then the full path is shown in the . . . 78
28.15New option for the history: ecb-kill-buffer-clears-history defines if . . . 78

29 Changes for ECB version 1.52 **79**
29.1 Added EIEIO requirement to ECB. . . . 79

30 Changes for ECB version 1.51 **79**
30.1 Now the names of all ECB buffers begin with a SPC per default. (Klaus) . . . 79
30.2 ECB now can handle not only full but also partial buffer reparsing if done . . . 79
30.3 Incremental node-search in the ECB-buffers now uses the value of . . . 79
30.4 Complete new customization of the ECB-key-bindings. The option . . . 79
30.5 ecb-add-source-path and ecb-delete-source-path now ask if saving should be . . . 79
30.6 Added new navigator functionality. Makes it easy to go to the back and forward . . . 79
30.7 Added an option to narrow the buffer to the token that is jumped to. See . . . 80

30.8 Fixed a small bug that placed the window start at the beginning of the token . . . 80

31 Changes for ECB version 1.50 **80**
31.1 Fixed a minor bug that occured when de-activating ECB under XEmacs. (Jesper) . . . 80
31.2 ecb-layout-window-sizes variable is now an association list, which makes it . . . 80
31.3 Added a new layout (nr. 12). Thanks to Nick Cross . . . 80
31.4 Each source path can now have an alias that is displayed instead of the path . . . 80
31.5 The history items can now use the buffer name instead of the file . . . 80
31.6 When jumping to a token the window can be scrolled so that the token starts . . . 80
31.7 Fixed a bug when retrieving parent names for a token. (Jesper) 81
31.8 Now the option 'ecb-prefix-key' is customizable so you can define another . . . 81

32 Changes for ECB version 1.41 **81**
32.1 Fixed bug when clicking on token in the methods buffer in XEmacs. . . . 81

33 Changes for ECB version 1.40 **81**
33.1 Tree-incremental-search in the ecb-windows now ignores all non interesting . . . 81
33.2 ECB now uses Semantic DB to find parents of types. (Jesper) 81
33.3 Long source paths in the directories buffer are now truncated at the . . . 81
33.4 Tokens can now be sorted by access. (Jesper) . . . 82
33.5 Added a function to semantic-clean-token-hooks that just updates the token . . . 82
33.6 Implemented a token tree cache that stores recently opened buffers' token . . . 82
33.7 Added a speedbar-like layout nr. 11. This is very useful (like also layout . . . 82
33.8 Better extraction of tokens from current buffer. ECB now displays any token . . . 82
33.9 ECB now uses Semantic to display tokens in the methods buffer. (Jesper) . . . 82

33.10Much saver ecb-redraw-layout: Now the layout can be restored ALWAYS . . . 82
33.11New advice for 'other-window-for-scrolling', so all scroll-functions for the . . . 82
33.12ecb-toggle-ecb-windows now preserves the split, the split-amount, the buffer . . . 82
33.13Some important changes in the ECB-layout concerning displaying compilation- . . . 83
33.13.1 'ecb-select-compile-window' has been gone. . . . 83
33.13.2 'ecb-use-dedicated-windows' is not longer a user-customizable option but . . . 83
33.13.3 'ecb-compile-window-temporally-enlarge' is now a boolean option. . . . 83
33.13.4 Now ECB does all necessary that your edit-window of ECB seems to be a . . . 83
33.13.5 The variables 'compilation-window-height' and 'temp-buffer-shrink-to-fit' . . . 83
s possible to83subsection.33.14
s own menu "ECB" and its own key-map83subsection.33.15
33.16Fixed a bug in highlighting current token in the method-buffer when . . . 84
33.17New option for highlighting the method-header in the edit-window after . . . 84
33.18Function for automatically submitting a problem report to the ECB mailing . . . 84
33.19Now not only for method-highlighting an idle delay can be set but also for . . . 84
33.20Smarter highlighting of current method (Klaus). . . . 84
33.21All tree-buffers now have as default-directory the current selected . . . 84

34 Changes for ECB version 1.32 **84**
34.1 Nil parent bug fixed. (Jesper) . . . 84
34.2 New advices for find-file and switch-to-buffer (Klaus). . . . 84
34.3 Now possible to set an idle delay before the current method is highlighted; . . . 84

35 Changes for ECB version 1.31 **85**
35.1 Parents (extends and implements) of the classes in the current buffer is now . . . 85

35.2 Now displaying the complete node name in minibuffer with the track-mouse . . . 85
35.3 Fixed a bug in 'ecb-current-buffer-sync' and 'ecb-redraw-layout' (Klaus) . . . 85
35.4 Fixed a bug in loading semantic 1.3.3 or semantic 1.4 (Klaus) 85

36 Changes for ECB version 1.30 85
36.1 If not all ECB-tree-windows of current layout are visible at redraw-time . . . 85
36.2 Added a section "Tips and Tricks" to the ECB online-help. (Klaus) . . . 85
36.3 Added a new layout Nr. 10 for very small screen resolutions where all . . . 85
36.4 The messages displayed after moving the mouse over a node in a tree-buffer . . . 85
36.5 ECB now also works with semantic >= 1.4 (Klaus). . . . 86
36.6 Now the ecb-compile-window-height is also preserved after displaying . . . 86
36.7 Added menu items for modification of the source paths in the directories . . . 86
36.8 Added buttons to directories buffer. (Jesper) . . . 86
36.9 Mouse over files, methods etc. now work even if follow-mouse isn't . . . 86
36.10 Popup menus now work in XEmacs. (Jesper) . . . 86
36.11 You can now specify paths with env-vars like $HOME in the option . . . 86
36.12 The user must now confirm if he tries to delete the ECB frame. If he wants . . . 86
36.13 ECB now optionally create its own frame when activated. See the new option . . . 86
36.14 Intelligent recentering of tree-buffers now completely implemented without . . . 87
36.15 Mouse over directories now shows directory path. (Jesper) . . 87
36.16 Methods and variables with point are now highlighted. (Jesper) 87
36.17 ECB windows are now updated when saving new files and deleting files. (Jesper) . . . 87
36.18 incremental-search in every tree-buffer for easier and faster selecting . . . 87
36.19 ecb-redraw-layout now restores both edit window buffers and the edit window . . . 87

36.20Spelling corrections. Thanks to Colin Marquardt 87
36.21Mode line prefixes can now be set individually for each ECB-buffer. There is 87
36.22Fixed small bug in directory tree that caused directories to be sorted by 87

37 Changes for ECB version 1.20 **88**
37.1 Now the ECB-buffers are intelligent in displaying nodes (Klaus): 88
37.2 The source file is only parsed if the ECB methods window is visible 88
37.3 The methods buffer now show fields of EIEIO classes. Only works with 88
37.4 Now all adviced functions behave only special in the ECB-frame. In each other 88
37.5 Fix "invalid method-buffer"-bug (Klaus, with a lot of help by Eric Ludlam 88
37.6 Source files can now be sorted by name, extension or not sorted at all 88
37.7 Now you can precisely define by an exclude and include regexp which files 88
37.8 Jumping to a method/variable from the ECB-method buffer now sets the mark so 89
37.9 Now you can define a primary and a secondary mouse-button for ECB. See the 89
37.10Now ECB is really ediff-friendly: During ediff is active the advices of ECB 89
37.11Now all tree-buffers (ECB-buffers) are read-only (Klaus)! . . . 89
37.12Better implementation of advising the functions. Now all advice-able 89
37.13Removes the redundant modeline info and shows important . 89
37.14Added new option 'ecb-advice-window-functions'. Enabling the intelligent 89
37.15Better online help display (Klaus). 89
37.16Added new option 'ecb-primary-mouse-jump-destination' (thanks to David Hay 89
37.17Adviced 'other-window' acts now exactly like the original but always related 90
37.18Fixed bug with 'ecb-compile-window-height'. Now ECB has always the correct 90
37.19New option 'ecb-compile-window-temporally-enlarge'. Allowing 90

38 Changes for ECB version 1.10 **90**
38.1 Automatic expansion of the directory tree. 90
38.2 Method parsing is now recursive, so that for example inner classes 90
38.3 Variables are now shown in the methods buffer. 90
38.4 Selected files and directories now use the secondary-selection . 90
38.5 Fixed some issues with XEmacs. Thanks to Colin Marquardt 90
38.6 Fixed bug with truncating lines in tree-buffer.el. 90
38.7 The methods buffer can now be automatically updated when the source 90
38.8 Help text is available with "ecb-show-help". 91
38.9 Moving the mouse over a node shows the name in the 91

39 Changes for ECB version 1.0 **91**
39.1 Changed name from JCB to ECB and prefix from "jde-jcb-" to "ecb-". 91
39.2 Works with semantic 1.3.1 and higher. 91
39.3 Fixed a couple of bugs when creating and deleting files. . . . 91
39.4 Grouped customization variables. 91
39.5 Contributed by Klaus Berndl <klaus.berndl@sdm.de>: 91
39.5.1 A lot of intelligent ECB window-functions (e.g. ecb-delete-window) which 91
39.5.2 New layout functionality. 92
39.5.3 The history buffer can optionally be sorted and also cleared. 92
39.5.4 The package expanded symbol can optionally be placed before the package name. 92
39.5.5 Syntax highlighting of the methods buffer. 92
39.5.6 Show argument names and return type in the methods buffer. 92
39.5.7 History clear function: ecb-clear-history. 92
39.5.8 Added pop-up menu to the history buffer. 92
39.5.9 Fixed edit buffer sync bug. 92

40 Changes for ECB version 0.04 **92**
40.1 Keyboard navigation in JCB buffers. Use Return to select and Tab to expand. 92
40.2 Tree view of packages. 92
40.3 Class files can be viewed in the packages buffer. 92
40.4 Better handling of mouse events. 92

40.5 Automatic update of the packages buffer. 92
40.6 Added pop-up menus to the packages and classes buffers. They are activated 92
40.7 Added method sorting. Thanks to Brian Anderson <brian.anderson@luna.com> 92

41 Changes for ECB version 0.03 **93**
41.1 Fixed naming of variables. 93
41.2 Fixed license and copyright comments. 93
41.3 Fixed class file regular expression bug. 93
41.4 Added three new layouts. 93
41.5 The JCB buffers can now be (almost) synchronized with the edit window. 93
41.6 When clicking on a class or method using mouse button 2, it will be loaded 93
41.7 Minor changes. Thanks to Jim for his suggestions. 93
41.8 Removed the "jde-jcb-window-skips-from-methods-buffer" and 93

42 Changes for ECB version 0.02 **93**
42.1 Changed prefix from "jcb-" to "jde-jcb-". This will cause JCB customization 93
42.2 Lists and parses non-Java file types. Currently the C and Lisp parsers from 93
42.3 History buffer implemented. 94
42.4 Layout function implement. Thanks to Jim Loverde <loverde@str.com> for 94
42.5 Packages matching a regular expression can be excluded from the packages 94

43 Changes for ECB version 0.01 **94**
43.1 Initial version. 94

1 Changes for ECB version 2.40.1

1.1 Better compatibility with forthcoming Gnu Emacs 23.2

1.1.1 Works out of the box with the Emacs-integrated CEDET-suite

If Emacs >= 23.2 is used then CEDET is already integrated into Emacs and ECB can be used out of the box without further requirements.

PLEASE NOTE: If ECB detects an author version of CEDET (as available at 'http://cedet.sourceforge.net') then ECB will ALWAYS try to use that one even if you use Emacs >= 23.2! This is for users who want to use latest Emacs >= 23.2 but still prefer using the latest author version of CEDET instead of the Emacs-integrated one.

So if you want to use the Emacs-integrated CEDET-suite you have to ensure that no author-version of CEDET is in the 'load-path'! This means that the library cedet.el of the author-version MUST NOT be loaded into Emacs (as described in the file INSTALL of CEDET)! This is a valid check: (locate-library "semantic-ctxt") must return nil!

Details about the Requirements of ECB are available in the README.

1.2 The special symboldef ECB-window is now stable and ready for usage

1.2.1 TODO description what is new

1.3 ECB now uses the partial reparse feature of semantic

The main effect a user will notice is that edit actions which trigger just a partial reparse will just update the related node in the methods-window and not the whole methods-window. One of the main advantages is that this preserves the expand/collapse state of the methods-window.

Editing examples which trigger only a partial reparse:

- editing a function- or methodbody
- changing the name of a variable or function/method
- modifying the return-type of a function/method
- and some other...

In these cases only the related node in the methods-window will change nothing else, ie. the expand/collapse-states of all other nodes in the methods-window will not be changed.

1.4 ECB eliminates duplicates in the directories-buffer

There can be duplicates either .... TODO description

1.5 New default-value for 'ecb-auto-update-methods-after-save': nil

Switching off this by default makes sense because now ECB uses the idle-scheduler of semantic (if it is switched on but this is at least very recommended) and therefore the methods-buffer will be up-to-date most of the time. Therefore updating it after saving makes not much sense. In addition this auto update is always a full-fetch for the tags which does not preserve the expand/collapse-state of the methods-window whereas a partial reparse does (s.a.).

1.6 The contents of the methods-window are updated even if currently hidden

If at least the current ecb-window-layout contains a methods-window then the contents will be autom-updated even if the methods-window is currently invisible (e.g. because the ecb-windows are all hidden or another ecb-window is currently maximized. But the updates takes only place if the semantic idle scheduler is active (see instructions in the manuals of ECB and/or cedet).

The advantage of this new behavior is that now editing changes done with invisible methods-window will be reflected immediately after showing the methods-window (e.g. by unhiding the ecb-windows via the command 'ecb-toggle-ecb-windows' - bound to [C-c . 1 w]).

1.7 New support for Bazaar as version-control system

If Bazaar is supported by VC (means vc-bzr.el is distributed with Emacs) then ECB supports it out of the box.

1.8 New default value of the option 'ecb-history-make-buckets'

This option offers now an additional choice 'directory-with-source-path: This substituts the best (= deepest) matching source-path of the directory-window in the directory-bucket with the alias of this matching source-path (if an alias is set in the option 'ecb-source-path'). A new face is used for this: 'ecb-history-bucket-node-dir-soure-path-face'.

1.9 'defecb-window-dedicator-to-ecb-buffer' replaces 'defecb-window-dedicator'

This is only relevant for people who program own special ecb-windows.

1.10 Fixed Bugs

1.10.1 Command 'delete-windows-on' has only deleted one window

Now this command works also with ECB as expected.

1.10.2 Fixed a bug which prevents the ecb-windows from being vertically resizeable

This bug occured only when using Gnu Emacs.

1.10.3 Fixed a bug concerning horizontal mouse-scrolling of ecb-windows via

modeline-click when the ecb-window was not the selected window. This bug occured only when using Gnu Emacs.

1.10.4 Fixed a bug which prevented 'ecb-type-tag-display' from taking effect.

1.10.5 Killing a buffer failed when history-window is not visible

Killing a buffer failed in case the history buffer is not visible (e.g. when currently a layout without a history buffer is used or when another buffer is maximized and the history buffer thus hidden). This is fixed now.

1.10.6 All compile-modes (e.g. compile, grep etc.) failed in XEmacs

This came from a bug introduced in 2.40 but it is fixed now.

2 Changes for ECB version 2.40

2.1 ECB now requires full CEDET-suite being installed (at least version 1.0pre6)

This means the support for the single packages semantic 1.4.X, eieio 0.X and speedbar 0.X is completely switched off.

As a consequence ECB is no longer runnable as XEmacs-package via the package-manager of XEmacs. This is because CEDET is not available as XEmacs-package. Therefore ECB can also not be run as XEmacs-package. If CEDET will later become a XEmacs-package then probably ECB will come back as XEmacs-package. But in the meanwhile you have to install ECB "by hand".

In general the current CEDET is much more powerful than older semantic-versions and ECB uses this power, especially the power of the semantic-analyzer.

2.2 Completely reworked synchronizing mechanism of the ECB-windows

Now there are separate options for synchronizing:

- The basic ECB-windows (directories, sources, methods, history): 'ecb-basic-buffer-sync', 'ecb-basic-buffer-sync-delay' and 'ecb-basic-buffer-sync-hook'.
- The analyser-window (s.a.): 'ecb-analyse-buffer-sync', 'ecb-analyse-buffer-sync-delay' and 'ecb-analyse-buffer-sync-hook'
- The symboldef-window: 'ecb-symboldef-buffer-sync', 'ecb-symboldef-buffer-sync-delay' and 'ecb-symboldef-buffer-sync-hook'
- The integrated speedbar (if used): 'ecb-speedbar-buffer-sync', 'ecb-speedbar-buffer-sync-delay' and 'ecb-speedbar-buffer-sync-hook'
- The integrated eshell-support (if used): 'ecb-eshell-buffer-sync', 'ecb-eshell-buffer-sync-delay' and 'ecb-eshell-buffer-sync-hook'

But all options for synchronizing not the basic ECB-windows offer an additional choice 'basic: Then ECB uses the setting of the related option for the basic ECB-windows! Example: If you set 'ecb-eshell-buffer-sync' to 'basic then the value of 'ecb-basic-buffer-sync' is used. If you set 'ecb-eshell-buffer-sync-delay' to 'basic then the value of 'ecb-basic-buffer-sync-delay' is used.

This enhancement allows much more control for the synchronizing of certain windows.

2.3 Better compatibility with CEDET >= 1.0pre6 and semantic >= 2.0pre6

2.3.1 The ECB-analyse-window now works very well with current semantic analyzer

There is an own synchronize-option-set for the analyse-window: 'ecb-analyse-buffer-sync', 'ecb-analyse-buffer-sync-delay' and 'ecb-analyse-buffer-sync-hook'. See the docstrings. The advantage is, that now the synchronizing of the analyse-window can be customized separately from the other ECB-windows. THis is important because for the analyse-window a higher delay (or even disabled auto.synchronizing) increases usability a lot. See the info-manual.

2.3.2 Fix compatibility about some changes of semantic concerning adopting

external member-functions (as in C++ or eieio).

2.3.3 ECB now works very well with semantic-idle-scheduler-mode (s.a.)

2.4 More user-responsible buffer-parsing based on the idle-mechanism of semantic

Force a reparse of the semantic-source if the idle-scheduler is off. Generally ECB calls semantic to get the list of tags for current source-file of current edit-window. Per default ECB does never automatically force a reparse of the source-file - this is only done on demand by calling 'ecb-rebuild-methods-buffer'. So per default the idle-scheduler of semantic is responsible for reparsing the source-file and when this is necessary (see 'semantic-idle-scheduler-mode' for further details). This is the most user-resonsible and therefore the recommended approach.

So it's strongly recommended to enable 'semantic-idle-scheduler-mode' because then reparsing is always done during idle-time of Emacs and is also interruptable.

But if this idle-scheduler is switched off then ECB offers now two possibilities (via 'ecb-force-reparse-when-semantic-idle-scheduler-off'):

- Not forcing itself a reparse when tags are needed by ECB: then a user declines knowingly Emacs/semantic-driven parsing of code when he/she switches off the idle-mode of semantic. This is the default behavior of ECB and the default value of this option. But this has also the consequence that the methods-buffer is only filed on demand via 'ecb-rebuild-methods-buffer' (bound to

$$C - c.r])!$$

This means also in consequence that the methods-buffer is not automatically filled when a source-file is opened but first on demand after calling the command 'ecb-rebuild-methods-buffer'!

- Forcing a reparse when tags are needed: Then ECB forces semantic to parse the source-file when ECB needs tags to display. For this behavior this option has to be set to not nil.

  The term "forcing a reparse by semantic" is a simplification: ECB uses then the function 'semantic-fetch-tags' which can decide that the cached tags are up-to-date so no real reparsing is necessary - but it can also run a full reparse and this reparse is not being done when Emacs is idle but immediatelly and not interruptable (as with the idle-scheduler of semantic), which can be quite annoying with big source-files.

2.5 Complete reworked history-buffer

2.5.1 The history is able to deal with indirect-buffer entries.

See new option 'ecb-history-stick-indirect-buffers-to-basebuffer'.

2.5.2 The history can now be bucketized, see new 'ecb-history-make-buckets'.

This option allows to define several criterias for building buckets in the history-buffer all the history entries are sorted in (e.g. by major-mode, directory, file-extension or regular expressions).

You can change the bucketizing type on the fly via the popup-menu of the history-window.

The regular expressions bucketizing allows in combination with the indirect-buffer ability to work with something like "virtual folders" (well, "virtual folders" light).

For example, there is a large project with a huge number of files, and there are various tasks in this project. So it could be convenient to group buffers according to various tasks. It could be fulfiled through using indirect buffers, for example like this

$\text{task}_1 - aaa.pl task_1 - bbb.c task_1 - ccc.sh ... task_N - xxx.java task_N - ccc.sh$

This means create indirect buffers with a name-part which can be used for grouping together buffers with same name-part (here e.g. $\text{task}_1 - ... task_N -$). $In the example above you wou$ $ccc.sh, the other named task_N - ccc.sh$).

Then use the new option 'ecb-history-make-buckets' to define regexps for bucketizing all (indirect) buffers according their task-part in the buffer-name.

2.5.3 There are now new faces for the history entries.

See new options 'ecb-history-bucket-node-face', 'ecb-history-dead-buffer-face' and 'ecb-history-indirect-buffer-face' and equaly named new faces.

2.6 Sticky parent-node for all special ECB-windows of type tree-buffer

Only for GNU Emacs: in the header-line of a tree-buffer always the current-parent node of the first visible node is displayed (if there is a parent node). This sticky node is exactly in the same manner clickable as all other nodes. There is a new option 'ecb-tree-make-parent-node-sticky' which enabales/disables this new feature (default is enabled).

2.7 Much saver advice-backbone for all advices needed by ECB

This is not a user-visible change but enhances the stability of ECB by using now a new advice-backbone which guarantes that all ecb-advices are enabled rsp. disabled correctly depending on the surrounding context. Introducing four new macros 'defecb-advice-set', 'defecb-advice', 'ecb-with-original-adviced-function-set' and 'ecb-with-ecb-advice'.

2.8 Changes in the version control support of ECB

2.8.1 ECB now also offers version control support only if the current vc-package is

installed which offers support for modern vc-backends like subversion, git and monotone. Because vc-package lacks having a version number which can be

checked ECB uses the following check if the right vc-package is installed: (locate-library "vc-svn") must return not nil.

2.8.2 New support for Git and Monotone as version-control systems

If Git rsp. Monotone are supported by VC (means vc-git.el rsp. vc-mtn.el are distributed with Emacs) then ECB supports now both of them out of the box.

2.8.3 'ecb-vc-supported-backends' offers now also a accurate recompute-state function

This is by offering 'ecb-vc-recompute-state' as an alias to the function 'vc-recompute-state' as state-check function for this option.

2.8.4 CVS-support is now fully compatible with Emacs 22 and 23

2.9 New options rsp. commands

2.9.1 New command 'ecb-goto-window-edit-by-smart-selection'

2.9.2 New command 'ecb-goto-window-ecb-by-smart-selection'

2.9.3 New option 'ecb-ignore-pop-up-frames'

The new option is for customizing the behavior of ECB concerning 'pop-up-frames'. It allows three value: Always, only when a permanent compile-window is used and never. This makes ECB fully compatible with the option 'pop-up-frames'.

2.10 Much better compatibility with current Emacs-versions

2.10.1 Full compatibility with Gnu Emacs 22

- 'balance-windows' now works with Emacs 22 too

  Cause of the completely new implemantation based on 'window-tree' ECB uses a new machanism for enabling balance-windows to work properly with active ECB, so only the edit-windows are balanced but all ecb-windows remain on their sizes.

- view-mode works also when a permanent compile-window is active

  This is especially important for displaying help and completions.

- 'master-mode' now works with Emacs 22

- Grepping from ECB now uses per default 'lgrep' rsp. 'rgrep'.

  - The option 'ecb-grep-function' defaults to 'lgrep' if available

    If not it tries 'igrep' or 'grep'.
  - 'ecb-grep-find-function' has been renamed to 'ecb-grep-recursive-function'

    It defaults to 'rgrep' if available. If not it tries 'igrep-find' or 'grep-find'. The old value is automatically upgraded to the new option name.

- Fixed problems with 'ecb-fix-window-size' and active compile-windows

  With Emacs >= 22 the bugs of Emacs 21.3.X concerning 'window-size-fixed' are fixed by the Emacs-team so now ECB supports this feature also with active compile-window.

- Fixed small lack in the 'switch-to-buffer-other-window'-advice.

  In Emacs 22 this command ignores the settings in 'same-window-*'. Now ECB adopts this behavior also for its adviced version so the command works in a smart manner optimized for ECB.

2.10.2 Much better compatibility with forthcoming Gnu Emacs 23

- Removed an annoying behavior which has set the mark hundred of times even

  when the user just moves the cursor.

This comes from a change within Emacs 23 and the function 'goto-line' which sets the mark in Emacs 23. This is not necessary when not used as command but as internal utility. Now ECB uses its own goto-line-function which does not set the mark –> No unnecessary mark settings with Emacs 23.

- The adviced 'display-buffer' is now fully compatible with Emacs 23

  This means simply that you can work with ECB and Emacs 23 as you expect.

  Especially for displaying temporary buffers (like completion, help, compilation etc.) this means: Gnu Emacs 23 has a much smarter window-manager for displaying temporary buffers (like help-buffers) and "compile"-buffers. Therefore ECB uses completely the logic of Emacs 23 and does exactly what Emacs 23 would do without activated ECB: This means mainly that the split-behavior depends on the settings of the options 'split-width-threshold' and 'split-height-threshold'.

  So if you are wondering why displaying temporary-buffers (like help, compile, grep, completion etc.) shows different behavior compared to deactivated ECB, then you probably have to adjust the values of the options 'split-width-threshold' and 'split-height-threshold' so they matches the sizes of the edit-area of ECB (at least nearly).

- Fixing some incompatibilities with Emacs 23 concerning using temp. buffers

  The fit-window-to-buffer version of Emacs 23 can destroy windows and is therefore not useable within ECB - now ECB has its own - stolen and adapted from Emacs 22.

  In addition to this the internal help-window-manager of Emacs 23 (a.o. responsible for displaying *Help*-buffers) chooses sometimes dedicated windows (which is a problem because ECB uses dedicated windows for its own special ECB-windows) whereas Emacs 22 doesn't do this. Now ECB works smoothly with Emacs 23 help-system. Same for view-mode.

- ediff-support of ECB is now also fully compatible with Emacs 23

  Emacs 23 uses dedicated windows: Now ECB can deal with frame-layouts where some windows are neither an ecb-, edit- nor a compile-

window. Therefore now also single-frame mode (see 'ediff-windows-setup-function') of ediff wirks smoothly with ECB.

- All this together makes the new layout-engine of ECB much more robust so ECB

  works smoothly with forthcoming Emacs 23.

2.10.3 Better compatibility with XEmacs 21.4 and XEmacs 21.5

Compatibility has been increased but is still not perfect. There is a need for much more tests with XEmacs. This can not been done by the ECB-maintainer because it costs too much effort. Any help from XEmacs-gurus using ECB for their daily work is very appreciated.

2.11 ECB is able to work with indirect buffers if the base-buffer is filebased

Now you can work with indirect-buffers as well as with normal file-buffers, i.e. indirect buffers are shown in the history, their contents are displayed in the methods-buffer, the ECB-analyse-buffer works with them, autom. synchronizing the ECB-tree-buffers works for them etc...

2.12 Deprecated, disabled and deactivated features:

2.12.1 The commands 'ecb-download-ecb' and 'ecb-download-semantic' are deactivated

These features are not supported anymore. It costs two much effort to keep the needed links and mechanism to the sourceforge-download-area up to date. Please download the needed release-archives and extract and install as described in the shipped README rsp. INSTALL-files.

2.13 Fixed Bugs

2.13.1 Fixed a bug which displayed empty tag-names in the methods-buffer when font-lock

is not loaded.

2.13.2 Fixed a bug in the ping-functionality

Now 'ecb-ping-options' has a new default value which should operate well on most os.

2.13.3 Fixed a bug which prevented 'ecb-rebuild-methods-buffer' from working

correctly

2.13.4 Fixed a bug in the internals of the methods-filter which prevented

all commands 'ecb-methods-filter-*' from working correctly.

2.13.5 Fixed context-menu for VC-operations in the sources- and history-window

Now all vc-commands are called interactively by ECB.

2.13.6 Fixed a bug with the ECB-navigation feature (C-c . p and C-c . n)

Now the commands 'ecb-nav-goto-next' and 'ecb-nav-goto-previous' work saver and also for indirect buffers.

2.13.7 Fixed small bugs with synchronizing current tag with highlighted tag

in the methods buffer. This was when the buffer was reparsed with 'semantic-idle-scheduler-mode' or after saving the buffer.

2.13.8 Fixed a bug in 'ecb-create-new-layout' which has displayed wrong modelines

in the layout-creation frame.

2.13.9 Fixed a bug in 'ecb-submit-problem-report' which prevented it from working

3 Changes for ECB version 2.32

3.1 New backbone-add-on of ECB for getting the definition of the symbol at point

It allows to display in a new special ECB-window the semantic (do not confuse this term with the semantic-library!) context of the definition of the current symbol under point. Per default ECB tries to find this context via semanticdb (part of cedet) and etags (shipped with (X)Emacs) but in general this is completely customizable; see the options of the new options-group "ecb-symboldef".

Either use the layout "left-symboldef" (e.g. via [C-c . l c]) or create a new ecb-layout via the command 'ecb-create-new-layout' and add a buffer of type "other" and name "symboldef" into this new layout.

This add-on interactior can be seen as a "backbone" infrastructure, because it is/should be highly customizable how to get the definition of the symbol at point (e.g. via semanticdb or etags or whatever).

Thanks to Hauke Wintjen <hauke.jans@tietoenator.com> for having the idea and writing the first implementation

This add-on ECB-window of ECB is currently in beta or gamma stage but works already quite well. So please try it out and send feedback to the ECB-mailing-list. Every suggestion is highly appretiated!

3.2 New native ECB-window (tree-buffer) for the semantic-analyzer

There is a new tree-buffer ECB-analyse which displays the tags and informations of the semanic-analyzer in a smooth way. There are already new prebuild layouts "left-analyse" and "leftright-analyse" which contain this new tree-buffer ECB-window. But of course it can be added to own build layouts - either via the command 'ecb-create-new-layout' (add a buffer of type "other" and name "analyse" into the new layout) or via a elisp-programmed new layout.

3.3 New smart-arrow-navigation in a tree-buffer with up- and down-arrow

Point jumps to the first character of the previous (up) rsp. next node (down). "First" character means either the first character of the expand-symbol (in case 'ecb-tree-expand-symbol-before' is not nil) or of the displayed node-name. Or with other words: The first non-indentation and non-guide-line (see 'ecb-tree-buffer-style') character of a node.

Left- and right-arrow were already smart - see the existing option 'ecb-tree-navigation-by-arrow'.

3.4 Better handling of maximized ecb-windows

3.4.1 Smarter handling of node-selections in maximized tree-windows

Now the default behavior of ECB is:

Maximized directories-window: When selecting a directory then first automatically the maximized directories-window will be "minimized" (i.e. all ecb-windows of current layout are displayed) if the current layout contains a sources-buffer and no sources are shown in the directories-window. So the source-files can be displayed in the sources-window.

Maximized sources- or history-window: When selecting a source-file in one of these buffers then first automatically the maximized window will be "minimized" (i.e. all ecb-windows of current layout are displayed) if the current layout contains a methods-buffer. So the tag-contents of the selected source-file can be displayed in the methods-window.

For a even smarter behavior ECB introduces a new option 'ecb-maximize-next-after-maximized-select' which automatically maximizes the next logical tree-window after a node selection. The definition of "next logical is": Directories –> sources, sources/history –> methods. See the docstring of this new option for more detailled informations.

Thanks for suggestion to Michael Reiher.

3.4.2 Easier maximizing and "minimizing"

ECB supports the default modeline-mechanisms for deleting other windows. GNU Emacs binds 'mouse-2' in its modeline to 'delete-other-window'. XEmacs binds a popup-menu with some window commands to 'button-3' in its modeline.

ECB combines the best of both worlds by supporting both of these mechanisms for both Xemacs and Emacs: ECB binds a toggle-command

to ‘mouse-2’ in the modeline of each tree-buffer which maximizes the current tree-window if all ECB-windows are visible and displays all ECB-windows if current tree-window is maximized. In addition ECB binds a popup-menu to ‘mouse-3’ which offers exactly 2 commands: Maximizing current tree-window and displaying all ECB-windows.

3.5 Renamed options or options with new type

The following options have been renamed: ecb-truncate-lines –> ecb-tree-truncate-lines

The following options have now a new type: ecb-tree-RET-selects-edit-window ecb-tree-image-icons-directories

ECB automatically upgrades the old values to the new options!

3.6 Better assistance fixing errors related to semantic-load problems

Often users report errors somehow related to problems running ECB with the right semantic-version. There are different reasons for this: - People do not install the cedet-suite as described in the INSTALL-file of cedet. - People byte-compile ECB with another semantic (also eieio and speedbar) version than loaded into Emacs. All of them prevent ECB from working correctly.

Now ECB detects the error at startup-time and assists the user in finding the reason of the problem and how to fix it.

3.7 More robust downloading of new ECB from within Emacs

Now ‘ecb-download-ecb’ should work also with native Windows-XEmacs and also when ECB is installed in a directory which name contains spaces. If there are still problems in running this command please check the new options ‘ecb-wget-setup’, ‘ecb-gzip-setup’ and ‘ecb-tar-setup’.

3.8 Support for (X)Emacs $<$ 21 has been officialy removed.

ECB requires now (X)Emacs $>=$ 21. ECB stops activation with an appropriate error-message when (X)Emacs $<$ 21 is detected.

3.9 Full documentation of the library tree-buffer.el

The ECB-info-manual contains a new section "tree-buffer" in the chapter "Entry points for elisp-programers". This describes in detail how to program

with the library tree-buffer.el which is shipped with ECB.

3.10 Fixed Bugs

3.10.1 Version-controlled (VC) and not-VC history-entries are now sorted

correctly according the setting in 'ecb-history-sort-method'.

3.10.2 Handles not anymore existing files in the stealthy tasks.

If a stealthy task detects that a displayed file does not exist anymore it removes this source-file from the tree-displays.

3.10.3 Fixed a bug when 'ecb-process-non-semantic-files' is set to nil.

This bug only occurs with XEmacs. As a side effect of this fix now the methods-buffer is correctly cleared out when loading a non-semantic-sourcefile (i.e. not parsable by semantic) and processing of non-semantic-files is switched off ('ecb-process-non-semantic-files' is set to nil) or the sourcefile is not parsable by a parser like imenu or etags/ctags.

3.10.4 Fixed a bug in displaying semantic-tags which have plain-string members

E.g. parsing 'defstruct' has failed in previous ECB-version cause of this bug.

3.10.5 Fixed a bug when Emacs is started with -q so Emacs could not determine the

'custom-file'. This case is correctly handled now.

4 Changes for ECB version 2.31

4.1 Fixed bugs

4.1.1 Fixed a recently introduced bug with XEmacs and an active isearch.

This bug occured when a user has clicked onto a node in a tree-buffer during an active isearch and it had as result completely out-of-order tree-buffers. This buf is now fixed and should never occur.

4.1.2 Fixed a bug concerning accessing root-drives in native window-XEmacs

Fixes an annoying behavior of the native windows-version of XEmacs: When a path contains only a drive-letter and a : (e.g. C:) then 'expand-file-name' (and a lot of other file-operations) do not interpret this path as root of that drive. So ECB adds temporary a trailing 'directory-sep-char' and calls file-operations like 'expand-file-name' with this new path because then 'expand-file-name' treats this as root-dir of that drive. For all (X)Emacs-version besides the native-windows-XEmacs this fix is not needed and takes no effect.

4.1.3 Better recognizing of remote CVS-root-repositories for the VC-support

4.2 Enhancement for the VC-support

4.2.1 New option 'ecb-vc-xemacs-exclude-remote-cvs-repository'

This excludes directories with a remote cvs-repository from VC-check. This option takes only effect for XEmacs and is needed cause of the outdated VC-package of XEmacs which offers no heuristic state-checking and also no option 'vc-cvs-stay-local'. So this option takes only effect if 'vc-cvs-stay-local' is not avaiable. In this case ECB treats directories which are managed by CVS but have a remote repository as if the directory would not being managed by CVS (so the files are not checked for their VC-state). This is done to avoid blocking XEmacs cause of running full cvs-commands (e.g. "cvs status") for a bunch of files over the net.

4.2.2 Added beta-support for Clearcase

There is already an identify-backend-function 'ecb-vc-dir-managed-by-CC' and a check-state-function 'ecb-vc-check-CC-state' and an advice for 'clearcase-sync-from-disk' which enable automatic state-update after checkin/out. These code seems to work quite well but nevertheless there are some problems so this Clearcase-support is not added per default to 'ecb-vc-supported-backends'.

So if you want Clearcase-support you have to customize the option 'ecb-vc-supported-backends' and add these functions mentioned above (the advice of 'clearcase-sync-from-disk' takes only effect if the function 'ecb-vc-dir-managed-by-CC' can be found in 'ecb-vc-supported-backends' and ECB is active!). But use it at your own risk. Of course the best would be if a lot of

people would test this Clearcase-support and fix the encountered problems ;-). Thanks to David Ostrovsky for providing a first implementation!

5 Changes for ECB version 2.30.1

5.1 Enhancement to the automatic option-upgrading mechanism

ECB now automatically makes a backup-file of that file which will be modified by storing the upgraded rsp. renamed ECB-options. This backup file gets a unique name by adding a suffix ".before$_{e}cb_{<}version > "tothenameofthemodifiedfile.Ifsuchafil$ $file$!

5.2 Enhancement to the VC-support

5.2.1 Better recomputing of the VC-state of a file when state changed outside

With the new check-state-function 'ecb-vc-state' the heuristic state is always computed right which is especially useful if the state for a file has been changed outside Emacs (e.g. by checking in from command line or Windows Explorer). This function is now added to the default-value of 'ecb-vc-supported-backends' for GNU Emacs.

5.2.2 Added out-of-the-box support for VC-system Subversion

For this the latest version of the VC-package incl. the library vc-syn.el is needed. Latest CVS Emacs contains this VC-version. The new function 'ecb-vc-dir-managed-by-SVN' is now added to the default-value of 'ecb-vc-supported-backends'. Thanks for first implementation to Ekkehard Grlach <ekkehard.goerlach@pharma.novartis.com>.

5.3 Fixed bugs

5.3.1 Fixed errors occured at load-time of ECB 2.30

6 Changes for ECB version 2.30

6.1 Enhancements to the file-browser

6.1.1 Much better performance of the file-browser display because all

time-consuming tasks (like the check if the displayed directories are empty of not) are now performed "stealthy" - means when Emacs is idle. Each stealthy task is interruptable by the user just by hitting any key or clicking the mouse so Emacs/ECB will not be blocked by such tasks; next time Emacs is idle again the interrupted task automatically proceeds from the state it has been interrupted. There is a new macro 'defecb-stealthy' which can be used by a user to program own stealthy tasks.

Currently ECB performs three stealthy tasks:

Prescann directories for emptyness: Prescann directories and display them as empty or not-empty in the directories-buffer. See the documentation of the option 'ecb-prescan-directories-for-emptyness' for a description.

File is read only: Check if sourcefile-items of the directories- or sources-buffer are read-only or not. See documentation of the option 'ecb-sources-perform-read-only-check'.

Version-control-state: Checks the version-control-state of files in directories which are managed by a VC-backend. See the option 'ecb-vc-enable-support'.

There is also a new option 'ecb-stealthy-tasks-delay'.

There are three options which allow excluding certain directories from these stealthy tasks: 'ecb-prescan-directories-exclude-regexps', 'ecb-read-only-check-exclude-regexps' and last but not least 'ecb-vc-directory-exclude-regexps'.

6.1.2 ECB is now capable of handling remote paths.

"Remote" means file- or directory-paths in the sense of TRAMP, ANGE-FTP or EFS. Such paths can now being added to the option 'ecb-source-path' with no limitation compared to "local" paths. Just work with remote-paths in the same manner as with local paths. See also the additional choices of the options 'ecb-prescan-directories-for-emptyness', 'ecb-sources-perform-read-only-check' and 'ecb-vc-enable-support' (new). This new support is tested with the combinations GNU Emacs+TRAMP and XEmacs+EFS but

it should (hopefully) also work with all other combinations. Thanks a lot to Tomas Orti for beta-testing!

6.1.3 ECB displays the Version-control-state of a file in the tree-buffers.

There are four new options 'ecb-vc-enable-support', 'ecb-vc-supported-backends', 'ecb-vc-directory-exclude-regexps' and 'ecb-vc-state-mapping' which define if and how ECB should check the state of a sourcefile in a directory managed by a version-control system. By default ECB supports the same VC-backends as the builtin VC-support of Emacs: CVS, RCS and SCCS. But the option 'ecb-vc-supported-backends' allows to add support for arbitrary VC-backends (e.g. Clearcase). New image-icons are also included for a cute display of the VC-state in the directories, sources and history-buffer. Thanks to Markus Gritsch <gritsch@iue.tuwien.ac.at> for contributing the icons.

It's recommended to read the section "Version-control support" in the chapter "Tips and Trick" of the ECB-info-manual!

6.1.4 New hook which runs directly after the selected directory has changed.

See documentation of 'ecb-after-directory-change-hook'.

6.2 The popup-menu of the methods-browser allows precisely expanding of the

current node. This means you can precisely expand a certain node to an exact indentation level relative to the node. This means all subnodes <= this level will be expanded (full recursive expanding is therefore of course also possible) and all subnodes indented deeper than this level will be collapsed - this is very different from using the expand/collapse symbol of a node. For forther details and examples the the manual and the section "Expanding" and here the subsection "Explicit expanding of the current node to a certain level".

6.3 Automatically upgraded ecb-option-settings are now not saved by default.

This means that ECB has now the new policy "Never touching the customization-files of a user without asking". The result is a completely redesigned upgraded-options-buffer: Now at the bottom of this buffer (displayed by 'ecb-display-

upgraded-options') two clickable buttons [Save] and [Cancel] are displayed which give the user the choice between saving the upgraded options for future Emacs-sessions ot just to cancel this buffer. In the latter case ECB has also upgraded the not compatible or renamed options (as listed in the displayed upgraded-options-buffer) but they will be not saved, i.e. no customization-file is touched and the changed and upgraded values will be lost after quiting Emacs.

6.4 With XEmacs ECB temporary sets 'progress-feedback-use-echo-area' to t

This is necessary because otherwise the progress-display with native widgets modifies the window-sizes of ECB and does not exactly restore the window-sizes as before that progress-display. Deactivating ECB automatically restores the old value of this option.

6.5 Fixed bugs

6.5.1 Fixed resizing of the ecb-windows after opening a file

Sometimes (X)Emacs (the behavior has only been reported for XEmacs) resizes the ecb-windows after opening a file by clicking onto a sourcefile or calling 'find-file' (or similar functions). This is not a bug of ECB but nevertheless it is annoying for the ECB-users. Therefore ECB has now a workaround which prevents the ecb-windows from resizing. The work around is done via two simple advices of 'find-file' and 'find-file-other-window'.

6.5.2 Fixed a bug in the upgrading feature (command 'ecb-download-ecb') which has

occured when the user has set a different LOKALE (e.g. "de$_D$E@*euro*").

6.5.3 Fixed a bug in restoring sizes of the ecb-windows.

Now a check will be performed if there are ecb-windows visible. If not nothing will be done (versions $<$ 2.30 have failed in such a case). This bug has prevented ediff from working together with ECB when a compile-window was visible and the user has stored window-sizes for the current layout.

7 Changes for ECB version 2.27

7.1 The option 'ecb-auto-expand-tag-tree-collapse-other' now has three possible

choices. You can decide between auto-collapsing only when point stays onto a tag in the edit-window or even when point doesn't stay onto a tag (which then results in a full collapsed methods-buffer). ECB autom. upgrades the old setting of this option to the new type. Thanks to Javier Oviedo <joviedo@telogy.com> for the suggestion.

7.2 Not accessible directories are displayed in a different face in the

directory-browser. See the new option 'ecb-directory-not-accessible-face' and the new face with same name.

7.3 Enhancements to the permanent compile-window

7.3.1 Enlarging the compile-window does not destroy some ecb-windows

With previous versions of ECB enlarging the permanent compile-window has often destroyed some ecb-windows if the compile-window has occupied a large portion of the frame-height (e.g. half of the frame). Beginning with ECB 2.27 such a ecb-window-damage is either prevented or - if it still occurs (can not be prevented in all situations) - ECB now has a "self-healing"-mechanism which immediatelly repairs the layout via idle-checks if the current window-layout is correct for the current 'ecb-layout-name'. Conclusion: Enlarging or shrinking the compile-window should never damage the ecb-windows! Thanks to Javier Oviedo <joviedo@telogy.com> for a great problem-report!

Caution: If bytecompiled user-defined layouts are used (ie. the file, which contains the user-defined layouts is byte-compiled) then you MUST recompile this file with ECB 2.27 active!

7.3.2 Now 'ecb-store-window-sizes' and 'ecb-restore-window-sizes' also work when

called during a permanent compile-window is visible and also when these commands are called with another compile-window state than the other (e.g. when the window-sizes are stored without a compile-window and restored with a compile-window).

7.3.3 Now ECB ensures that the permanent compile-window has always its specified

size when shrinked either by Emacs/ECB or by the command 'ecb-toggle-compile-window-height'.

7.3.4 Now ECB ensures that all ecb-windows have always either their default or

stored size size after the compile-window has been shrinked either by Emacs/ECB or by the command 'ecb-toggle-compile-window-height'.

7.3.5 ECB takes into account the new Emacs-option 'grep-window-height' when it

compute the max. allowed size of its permanent compile-window. More generally spoken: It takes into account all options with name *-window-height defined for modes which are defined by the macro 'define-compilation-mode'. This mechanism is available in the forthcoming Emacs 21.4 (and of course with current CVS-Emacs 21).

7.4 Fixed Bugs

7.4.1 Added a workaround for a bug in the custom-library of current release 21.3

of GNU Emacs which causes under certain circumstances the autom. option-upgrading-feature of ECB to fail. Same for the command 'ecb-store-window-sizes'. Thanks for reporting this annoying bug and great help to Christoph Ludwig <cludwig@cdc.informatik.tu-darmstadt.de>

8 Changes for ECB version 2.26

8.1 Improved the performance of the directories-buffer-display

Reduced the need of completely rebuilding the whole directories-tree-buffer when just switching between buffers and the related directories are already expanded. This can dramatically increase the speed of the tree-buffer display when displaying the sources-files in the directories-buffer (see 'ecb-show-sources-in-directories-buffer'). Thanks to Rob Walker <rob.lists@tenfoot.org.uk> for tracking down the performance bottleneck and supplying a first patch.

8.2 Per default all mouse-actions in the special ECB-buffers are now triggered

first after releasing the mouse-button and not when pressed as with previous ECB-versions. This is the standard behavior of Emacs and of most applications. But with the new option 'ecb-tree-mouse-action-trigger' a user can switch back to the old behavior.

As a side effect now the ECB-tree-buffers can be used during an active isearch - at least with GNU Emacs. Thanks to Markus Gritsch <gritsch@iue.tuwien.ac.at> for pointing to this problem.

8.3 ECB sets autom. temp-buffer-resize-mode rsp. temp-buffer-shrink-to-fit when

needed. This modes are needed when a permanent compile-window is used because otherwise the correct sizing of the compile-window according to the settings in ecb-compile-window-temporally-enlarge and ecb-enlarged-compilation-window-max-height is not possible. After disabling the permanent compile-window or when deactivating ECB the original settings of these modes are restored!

8.4 Some regexp-options has been changed to regexp-list-options:

'ecb-excluded-directories-regexp' –> 'ecb-excluded-directories-regexps' 'ecb-source-file-regexps' –> 'ecb-source-file-regexps' 'ecb-exclude-parents-regexp' –> 'ecb-exclude-parents-regexps'

All these options now accept (and require) a list of regexps instead of exactly one big regexp. This allows easier regexp-definition and follows now the way Emacs goes with other regexp-options like 'same-window-regexps' or 'special-display-regexps' which are also lists of regexps.

ECB autom. upgrades your old settings to the new option-types rsp. also -names (if the option has been renamed).

8.5 New option 'ecb-history-exclude-file-regexps' for excluding certain files

from being displayed in the history-buffer of ECB. Currently the filenames TAGS and semantic.cache are excluded per default. Thanks to Javier Oviedo <joviedo@telogy.com> for the suggestion.

8.6 ECB now displays read-only sourcefiles in a different face.

For this a new option and a new face are offered both named 'ecb-source-read-only-face' (default: italic) which is used to display source-files in the sources-tree-buffer (or in the directories-tree-buffer if 'ecb-show-sources-in-directories-buffer' is not nil) if the file is read-only.

8.7 Fixed bugs

8.7.1 Fixed merging faces with XEmacs.

Now options like 'ecb-type-tag-display' (merges own ECB-faces to already by semantic faced tags) work also correct with XEmacs. ECB-versions prior to 2.26 has not merged the ECB-faces to the already applied faces of a text but instead replaced the applied faces with the ECB ones. Now all faces are merged.

8.7.2 Now the command 'ecb-expand-methods-nodes' and all the expand-* menu-entries

in the popup-menu of the methods-buffer work for non-semantic-buffers too.

8.7.3 Fixed a bug concerning 'ecb-auto-expand-tag-tree-collapse-other'.

Thanks to Javier Oviedo <joviedo@telogy.com> for pointing to the problem.

8.7.4 Handles not existing source-paths in 'ecb-source-path' correct.

These paths are now ignored with a warning.

9 Changes for ECB version 2.25

9.1 More flexible sorting of the Sources- and the History-buffer

9.1.1 'ecb-sort-history-items' has been renamed to 'ecb-history-sort-method'

and offers now the same choices as the already existing option 'ecb-sources-sort-method'.

9.1.2 Two new options 'ecb-sources-sort-ignore-case' and

'ecb-history-sort-ignore-case' which allow to ignore case when sorting the Sources- and/or the History-buffer. Thanks for suggestion to Markus Gritsch <gritsch@iue.tuwien.ac.at>. Per default case is now ignored.

9.2 New icons for parent-display in the Methods-buffer.

Thanks to Markus Gritsch <gritsch@iue.tuwien.ac.at> for contributing the icon-images.

9.3 Fixed Bugs

9.3.1 Fixed an annoying bug which results sometimes in an error "stack-overflow

error in equal" when using 'senator-try-expand-semantic' which is called when you use hippie-expand for example. Maybe the bug occured also in other situation but now this bug has been extirpated!

9.3.2 Fixed a bug in the mechanism which prescanes directories for emptyness.

Now a directory is checked if it is accessible before it is prescanned - otherwise the prescan could fail with an error.

9.3.3 Fixed a bug in the autom. option-upgrading-mechanism.

Thanks to Javier Oviedo <joviedo@telogy.com> for pointing to the underlying problem.

10 Changes for ECB version 2.24

10.1 Enhancements for the Methods-buffer

10.1.1 Enhancements to the icon-display in the Methods-buffer

- New icon-images for enumerations and abstract/virtual tags (interfaces are

  currently displayed with the icon for abstract classes - at least til someone provides special icons ;-)

- Some slightly fixes in the other icons - e.g. a better display with a

  not-white background

- Static and abstract tags are no longer underlined rsp. italic when icons are

  displayed because this is somehow "tautological" when a special static rsp. abstract icon is already displayed.

10.1.2 New option for switching on/off distinction of function-prototypes from the

real function-definitions: 'ecb-methods-separate-prototypes'. This is useful for C++ and C because these languages distinct between a method-prototype (rsp. function-prototype for C) and the method (rsp. function for C) itself. If this option is not nil (default) then ECB separates the prototypes from the real function/methods in the Methods-buffer.

10.1.3 New "current-type"-filter for the Methods-buffer

In languages which have types (like Java or C++) this filter displays only the current type and all its members $e.g.attributesandmethods).Thecurrent-typeiseitherthetypeofthecurrenttaginthesource-buforthetypeofthecurrentnodeinthemethods-buffer.IfECBcannotidentifythecurrenttypeinthesource-borinthemethods-windowthennothingwillbedone.$

10.2 Now directories are prescanned for emptyness so they are displayed as empty

in the directories-buffer even without user-interaction (i.e. in previous ECB-versions the emptyness of a directory has been first checked when the user has clicked onto a directory). ECB optimizes this check as best as possible but if a directory contains a lot of subdirectories which contain in turn a lot of entries, then expanding such a directory or selecting it takes of course more time as without this check - at least at the first time (all following selects of a directory uses the cached information if its subdirectories are empty or not). Therefore this feature can be switched of via the new option 'ecb-prescan-directories-for-emptyness'.

10.3 Fixed Bugs

10.3.1 Fixed some newly introduced bugs which made ECB 2.23 incompatible with

semantic 1.4.

10.3.2 Fixed a bug which changed the highlighted directory even after just expanding

or collapsing other directories.

10.3.3 Fixed two small bugs which have influenced 'ecb-create-new-layout'

11 Changes for ECB version 2.23

11.1 New cedet1.0beta2 is supported.

11.2 Enhancements for the Methods-buffer

11.2.1 The lookfeel of the Methods-buffer is now much nicer because it has been

polished with a lot of new icons. ECB contains now a lot of new icons for displaying the Method-buffer as pretty and professional as possible. The new default-value of the option 'ecb-tree-image-icons-directories' points autom. to the new icon-images. Thanks a lot to Markus Gritsch <gritsch@iue.tuwien.ac.at> for providing these new icons.

To use this you should reset the option 'ecb-tree-image-icons-directories' to the new default-value if you have customized this option!

You can disable this icon-display via the new option 'ecb-display-image-icons-for-semantic-tags' if you do not like the new icons.

11.2.2 New feature for applying default tag-filters for certain files.

The new option 'ecb-default-tag-filter' allow to define default tag-filters for certain files which are applied automatically after loading such a file into a buffer. The possible filters are the same as offered by the command 'ecb-methods-filter' and they are applied in the same manner - the only difference is they are applied automatically. The files can be specified on a combined major-mode- and filename-regexp-basis.

Usage-example: This can be used to display for outline-mode files (e.g. NEWS) in the Methods-buffer only the level-1-headings by defining a regexp-filter "$.*$"$(seethedefault-valueofthisnewoption)$.

11.2.3 New keybindings and commands for faster applying certain tag-filters.

Here are the new bindings: [C-c . fr] –> 'ecb-methods-filter-regexp' [C-c . ft] –> 'ecb-methods-filter-tagclass' [C-c . fp] –> 'ecb-methods-filter-protection' [C-c . fn] –> 'ecb-methods-filter-nofilter' [C-c . fl] –> 'ecb-methods-filter-delete-last' [C-c . ff] –> 'ecb-methods-filter-function'

11.2.4 The popup-menu contains now mode-dependend tag-filter entries.

This means for sources not supported by semantic no protection- or tag-class filters will be offered. And for semantic-supported sources exactly these tag-classes are offered the semantic-parser for the current major-mode offers. Same for the command 'ecb-methods-filter'. For example texi-sources can only be filtered by the tag-classes "Definitions" and "Sections" and java-sources can be filtered by "Methods", "Variables", "Classes" etc. In general the semantic-variable 'semantic-symbol->name-assoc-list' is used to get the right tag-classes.

11.2.5 The option 'ecb-show-tags' is now defined on a major-mode-basis.

This means you can have different settings for each major-mode. ECB autom. upgrades your old setting to the new option-type.

11.2.6 Distinction between functions and function-prototypes in the Methods-buffer

This is for example useful for C++ and C because these languages distinct between a method-prototype (rsp. function-prototype for C) and the method (rsp. function for C) itself. The new default value of 'ecb-show-tags' displayes per default the methods as flattened and the method-prototype as collapsed when opening a C- or C++-buffer. ECB autom. updates your old setting of this option to the new default value.

11.3 The command 'ecb-toggle-layout' now has a prefix-argument:

If this optional argument is not nil (e.g. if called with a prefix-arg) then always the last selected layout was choosen regardless of the setting in 'ecb-toggle-layout-sequence'. The last selected layout is always that layout which was current direct before the most recent layout-switch. So now a user can switch to another layout via 'ecb-change-layout' [C-c . l c] and always come back to his previous layout via [C-u C-c . l t].

11.4 Better internal self-monitoring of ECB

ECB now monitors itself when Emacs is idle, ie. it monitors if all necessary ecb-functions are still member of the hooks 'post-command-hook' and 'pre-command-hook'. This is because Emacs resets these hooks to nil if any of the contained hook-functions fails with an error-signal. The ECB-hook-functions are error-save but if any of the "other" hook-functions of these hooks fail then the ECB-hooks would also have been removed from these hooks. The new monitoring-mechanism checks this periodically and re-adds the ECB-hooks if necessary.

11.5 New option group for the integrated speedbar and one new option:

The new hook 'ecb-speedbar-before-activate-hook' runs directly before ECB activates the integrated speedbar. For example this hook can be used to change the expansion-mode of the integrated speedbar via 'speedbar-change-initial-expansion-list'. Example: (speedbar-change-initial-expansion-list "buffers").

11.6 Added a new section in the info-manual which describes how desktop.el can be

used best in combination with ECB. This has been done because some conflicts have been reported. See the "Conflicts"-Chapter in the info-manual.

11.7 Fixed bugs

11.7.1 Double-clicking the mouse-1-button now works with integrated speedbar.

11.7.2 Fixed two small bugs in the advice of 'display-buffer':

One bug occured when called for one of the special ECB-buffers. The other bug has not preserved the current-buffer when called for a buffer in the sense of 'ecb-compilation-buffer-p' and a hidden compile-window is shown by display-buffer to display this buffer. The latter one for example has prevented the ant-server of JDEE from working properly.

11.7.3 Fixed a bug so now always the correct directory in the directories-buffer

is higlighted.

12 Changes for ECB version 2.22

12.1 New nifty feature for filtering the tags displayed in the Methods-buffer

You can filter by regexp, protection, tag-class etc. See the new command 'ecb-methods-filter' (bound to [C-c . f m] and the new option 'ecb-methods-filter-replace-existing' for getting all details.

Filters can be layered filters, i.e. you can apply filters on top of already applied filters: An example of 3 combined filters would be: Display only all public methods having the string "test" in its name. The modeline of the Methods-buffer always displays the topmost filter. But you can see all currently applied filters by moving the mouse over this section of the modeline: The filters will be displayed as help-echo.

The new functionality is also completely available via the popup-menu of the Methods-buffer.

Currently tags of class 'type can not be filtered out, because especially in OO-languages like Java or C++ this makes only less sense because if the class is filtered out then no filter can be applied to its members like methods or attributes.

These tag-filters can also applied to sources which are not supported by the semantic-parser but "only" by imenu or etags. But because for these sources not all information are avaiable the protection- and tag-class filter can not used with such "non-semantic"-sources.

12.2 All tree-buffer filters are now guarded against filtering out all tags.

Each filter which a user can apply to the sources-, methods- or history-buffer is now guarded against filtering out all nodes by accident, ie. if a filter would empty the whole tree-buffer then this filter will be autom. unwinded.

12.3 Much smarter mechanism to highlight the current tag in the methods-buffer.

Previous versions of ECB have always fully expanded the whole tree in the Methods-buffer if the current tag in the source-buffer was not visible in the current tree - e.g. because the variables-bucket was collapsed or the containing type of a tag (e.g. the class of a method) was collapsed. So in most cases much more was expanded as needed to make the current tag visible.

The mechanism of ECB 2.22 only expands the needed parts of the tree-buffer to make the related node visible: First we try to highlight the current tag with current expansion-state of the Methods-buffer. If the node is not visible so the tag can not be highlighted then we go upstairs the ladder of type-tags the current tag belongs to (e.g. we expand successive the nodes of the whole class-hierachy of the current method-tag until the related node becomes visible). If there is no containing type for the current tag then the node of the tag is probably contained in a toplevel-bucket which is currently collapsed; in this case we expand only this bucket-node and try highlighting again. Only if this has still no success then we expand the full tree-buffer and try to highlight the current tag.

There is also a new option 'ecb-auto-expand-tag-tree-collapse-other': If this option is set then auto. expanding the tag-tree collapses all not related nodes. This means that all nodes which have no relevance for the currently highlighted node will be collapsed, because they are not necessary to make the highlighted node visible. This feature is switched off by default because if always collapses the complete Methods-tree before the the following highlighting of the node for the current tag expands the needed parts of the tree-buffer.

12.4 The popup-menus can now be nested into 4 levels of submenus. In general there

could be an infinite depth of nesting but it makes no sense - if possible at all - to define infinite nested defcustom-types. So there is a limit of 4 levels until

a user asks for more - then this wish can be satisfied during 1 minute: Just increase the value of the 'ecb-max-submenu-depth' before loading ECB. This enhancement affects the four options with name 'ecb-*-menu-user-extension'.

12.5 The indication for ECB in the minor-mode is now hidden when the ECB-windows

are visible because in this case the activity of ECB is quite obvious. When the ECB-windows are hidden the the value of 'ecb-minor-mode-text' is displayed in the modeline.

12.6 Compatibility enhancements

12.6.1 Cause of the fixed bugs the compatibility for 'bs-show' has increased when

the buffer-selection should be displayed in the permanent compile-window.

12.7 Fixed Bugs

12.7.1 Fixed a bug preventing the native Windows-port of XEmacs from working.

12.7.2 Makes the behavior of adviced 'display-buffer' fully compatible to the

original version for all buffers which are neither ecb-tree-buffer nor compilation-buffers in the sense of 'ecb-compilation-buffer-p'. These "source-buffers" are now also handled correctly in all situations.

12.7.3 Fixed a small bug in 'ecb-toggle-compile-window-height'. This bug has

influenced the behavior of 'display-buffer' when called from programs for "compilation-buffers".

12.7.4 Fixed a bug related to the speedbar-integration.

This bug complained "Can not switch buffer in dedicated window" when the intergrated speedbar-buffer was selected and the user tries to open a file or to jump to a tag.

12.7.5 Fixed a bug for the special ECB-modeline-menu for the XEmacs-modelines.

Now only in the special ECB-tree-buffer a special ECB-modeline-menu is bound to mouse-3, all other buffers display the standard modeline-menu of XEmacs.

12.7.6 Fixed a bug which occurs with XEmacs in combination with func-menu.

But there are still some incompatibilities between ECB and func-menu. So the recommendation is to disable func-menu-support when using ECB. Normally using func-menu makes no sense in combination with ECB because ECB provides the same and even more informations as func-menu - so func-menu is redundant ;-)

13 Changes for ECB version 2.21

13.1 Advice for 'balance-windows' so only the edit-windows are balanced.

Thanks to David Ishee <782zp6a02@sneakemail.com> for suggestion.

13.2 New option 'ecb-ignore-display-buffer-function'

Per default the adviced version of 'display-buffer' ignores the value of 'display-buffer-function' when called for the ECB-frame. If this variable should not be ignored then the function of 'display-buffer-function' is completely responsible which window is used for the buffer to display - no smart ECB-logic will help to deal best with the ECB-window-layout! You can define if and when 'display-buffer-function' should be ignored: Always (default), when a compile-window is used or never.

13.3 Compatibility enhancements

In general the current layout-engine of ECB is so flexible that there should be no - or at least very few - conflicts between ECB and any other elisp-library - even when another lib is running during the ECB-windows are visible.

13.3.1 Commands like 'bs-show' of the library bs.el now work correctly with ECB

13.3.2 Xrefactory works even when all ECB-windows are visible (see Fixed Bugs).

13.3.3 Applications like Gnus or BBDB run withing the ECB-frame without conflicts -

even when the ECB-windows are visible.

13.3.4 Commands using 'Electric-pop-up-window' now work correctly with ECB.

This ensures that the electric-* commands (e.g. 'electric-buffer-list' or 'electric-command-history') work well with ECB. If the related "display-buffer" of such an electric command is a "compilation-buffer" in the sense of 'ecb-compilation-buffer-p' then this buffer will be displayed in the compile-window of ECB - if there is any shown.

13.4 For XEmacs the package fsf-compat is no longer required by ECB.

But it is still necessary to check if fsf-compat is required by the packages semantic, eieio and speedbar which in turn are required by ECB.

13.5 Fixed Bugs

13.5.1 Fixed a fatal bug which prevents 'other-window' from working with

arguments < 0. This bug has also prevented Xrefactory from working correct when ECB is active - see compatibility enhancements above.

13.5.2 If point stays in the current source-buffer on a function argument then still

ECB highlights the function in the Methods-buffer.

13.5.3 ECB now uses 'compilation-window-height' correctly when set buffer-local

as possible with latest CVS-version of GNU Emacs 21.

13.5.4 Fixed a bug in 'ecb-sources-filter' and 'ecb-history-filter'.

14 Changes for ECB version 2.20

14.1 New customization group "ecb-most-important"

This group centralizes the most important options of ECB in one group which is also offered by the "Preferences" submenu ob the ECB-menu. These are options you should at least know that they exist.

14.2 The keybinding for the online-help has changed from [C-c . o] to [C-c . h]

This has been done because the key [C-c . o] is needed for the new command 'ecb-toggle-scroll-other-window-scrolls-compile'.

14.3 The options 'ecb-major-modes-activate' and 'ecb-majors-mode-deactivate' have

been replaced by one new option 'ecb-major-modes-show-or-hide'. The purpose of the old option is now quite out-of-date because: - ECB supports window-managers like winring.el or escreen.el very well, - as of version 2.20 ECB has no restrictions about the number of edit-windows and - the new option simplifies things a lot.

14.4 The restriction of only two edit-windows has been gone!

14.4.1 Beginning with this version 2.20 there are no restrictions about the

window-layout in the edit-area of ECB. This means you can split the edit-area of ECB in as many windows as you like. The split-state of the edit-area will be preserved when toggling the visibility of the ECB-windows and even between activation and deactivation of ECB. Deleting of certain edit-windows will never destroy the special ECB-windows or the compile-window. Just work with the edit-area of ECB as if it would be an extra frame!

14.4.2 Option 'ecb-split-edit-window' has been renamed in

'ecb-split-edit-window-after-start' because this new name reflects much better the purpose of this option. In addition there is offered a new value 'before-deactivation which is also the new default value. This new value

preserves the full state between activations of ECB, i.e. the visibility of the ECB-windows, the visibility of a compile-window and also the full split-state of the edit-area. ECB auto. upgrades your setting!

14.4.3 Compile-window can now be displayed even when the ECB-windows are hidden.

So now you can have the same compile-window functionality when the ECB-window are hidden as when the ECB-windows are visible. The state of the compile-window will be preserved when toggling the ecb-windows or when maximizing one ecb-windows! So you have the advantage of one special window for all help-, grep or compile-output also when the ecb-windows are hidden - a window which will not be deleted if you call 'delete-other-windows' (bound to [C-x 1]) for one of the edit-windows.

14.4.4 The option 'ecb-primary-mouse-jump-destination' has been renamed to

'ecb-mouse-click-destination' and has also changed its default value to 'last-point. ECB autom. upgrades the old settings.

14.4.5 New keybinding [C-c . g l] for selecting the last selected edit-window.

Also available via the ECB-menu. The command is 'ecb-goto-window-edit-last'.

14.4.6 Option 'ecb-other-window-jump-behavior' has been renamed to

'ecb-other-window-behavior' and has also two new allowed values: 'smart (the new default) and an arbitrary function-symbol. With the latter one the user can define his own other-window-behavior and the former one chooses in a smart and intuitive way the "other window" for going to it via 'other-window' or for scrolling it via one of the "scroll another window"-functions (e.g. 'scroll-other-window'). Thanks to John S. Yates, Jr. <john@yates-sheets.org> for suggesting the new smart behavior. ECB autom. upgrades the old value of 'ecb-other-window-jump-behavior' to the new option-name.

14.4.7 New option 'ecb-scroll-other-window-scrolls-compile-window' and new

command 'ecb-toggle-scroll-other-window-scrolls-compile' (bound to [C-c . o]). If this new option is nil then ECB chooses very smart and intuitive the window which will be scrolled by commands like 'scroll-other-window' (see documentation of the the option 'ecb-other-window-behavior'). But sometimes the user wants to scroll the compile-window from another window. With this new command the user can fest and easy toggle the behavior ECB chooses another window for scrolling.

14.4.8 Higher compatibility of ECB with other packages

Without the 2-edit-window-restriction ECB is now more compatible with other packages. For example the package calculator.el can now also being used without setting 'calculator-electric-mode' to not nil - regardless in how many edit-windows the edit-area of ECB is splitted. Also the package bs.el (command 'bs-show') benefits from the new layout-flexibility. As of ECB-version 2.20 there should be no conflicts between this package and ECB.

14.5 Enhancements to the tree-buffers of ECB

14.5.1 Enhancements to the popup-menus of the tree-buffers

- All popup-menus of the tree-buffers can be used with the tmm-library

  The already existing command 'tree-buffer-show-menu-keyboard' (bound to [M-m] in every tree-buffer of ECB) accepts now a prefix argument. If called with a prefix-argument (hit [C-u M-m]) then the popup-menu is displayed with the tmm-mechanismus (like the Emacs-[menu-bar] is displayed when 'tmm-menubar' is called). Thanks for suggestion to Yvonne Thomson <yvonne@thewatch.net>.

- Select an edit-window via popup where a source should be opened or a token

  should be displayed.

- The popup-menus can be dynamically extended.

See the new options 'ecb-directories-menu-user-extension-function', 'ecb-sources-menu-user-extension-function', 'ecb-methods-menu-user-extension-function' and 'ecb-history-menu-user-extension-function'.

- All popup-menu-commands respect the setting of the option

  'ecb-mouse-click-destination' (formerly known as 'ecb-primary-mouse-jump-destination' - see above).

14.5.2 ECB supports the default modeline-mechanisms for deleting other windows.

GNU Emacs binds [mouse-2] in its modeline to 'delete-other-window'. ECB now supports this mechanism by binding a toggle-command to [mouse-2] in the modeline of each tree-buffer: If all ECB-windows are visible then this command maximizes the current tree-window and if current tree-window is maximized all ECB-windows are displayed again. XEmacs binds a popup-menu with some window commands to [button-3] in its modeline. ECB supports this mechanism by replacing this menu by a menu which offers exactly 2 commands: Maximizing current tree-window and displaying all ECB-windows.

14.5.3 Now all commands for selecting a tree-buffer work even when this buffer is

not visible because another tree-buffer is maximized. If the tree-buffer is contained in the layout then the layout is redrawn with all its tree-buffers visible so that tree-buffer can be selected. Same for maximizing. An example for such a command is 'ecb-goto-window-methods'.

14.5.4 New option 'ecb-maximize-ecb-window-after-selection'.

Automatic maximizing of a tree-window after selecting it. Thanks for suggestion to John S. Yates, Jr. <john@yates-sheets.org>.

14.5.5 Changes to the image-icons of the tree-buffers

- All image-files have been renamed from <IMAGENAME>.xpm to

  ecb-<IMAGENAME>.xpm. This was necessary because the icons for "open" and "close" in Emacs toolbar got superseeded by ECB's

icons for "open" and "close". The problem was, that ECB was using the same names for the icon files "open.xpm" and "close.xpm" as emacs toolbar does. This problem occurs if the image-directory of ECB is contained in the 'load-path' which is when ECB is installed in the site-lisp directory of Emacs with the default subdirs.el of Emacs which adds automatically all subdirectories to the load-path (and therefore also subdirectories which does not contain elisp-files but only images - btw: not really smart;-). Thanks to Roland Schaeuble <roland.schaeuble@siemens.com> for pointing out that.

- Added all image-icons in a 10-point height.

  Thanks to Nick Cross <nick@goots.org> for sizing down the images.

14.6 New default-value for 'ecb-post-process-semantic-taglist'.

Now for buffers with major-mode 'c-mode' all function-prototypes are filtered out. For example this is useful for editing C files which have the function prototypes defined at the top of the file and the implementations at the bottom. This means that everything appears twice in the methods buffer, but probably nobody wants to jump to the prototypes, they are only wasting space in the methods buffer. Of course in C-header-files function-prototypes will be displayed!

Thanks for suggestion and a first implementation to Rob Walker <rob.lists@tenfoot.org.uk>.

14.7 Changes for the compile-window

14.7.1 The command 'ecb-toggle-compile-window' now displays a compile-window even if

the option 'ecb-compile-window-height' is nil. This is a shortcut for customizing this option via the customize-feature if you just want displaying a compile-window for the current Emacs-session. In this situation ECB asks for the height of the compile-window, sets this height as new value of 'ecb-compile-window-height' and displays the compile-window.

14.7.2 New option 'ecb-change-layout-preserves-compwin-state'.

Changing the layout preserves now the state of the compile-window if this option is not nil. This is for example useful if the user toggles between

several layouts and wants to preserve the hidden-state of the compile-window. Thanks to John S. Yates, Jr. <john@yates-sheets.org> for suggestion.

14.7.3 Adviced 'delete-window' and 'delete-other-windows' handle compile-window.

The former one hides the compile-window if called in or for that window whereas the latter one hides the compile-window if called in or for an unsplitted edit-window. Thanks to John S. Yates, Jr. <john@yates-sheets.org> for the suggestion.

14.8 New option 'ecb-activation-selects-ecb-frame-if-already-active'.

See the docstring for more details. Thanks for suggestion to Jeff Jensen <jeffjensen@nospam.visi.com>

14.9 The command 'ecb-redraw-layout' can be called with prefix-argument(s).

Called with ONE prefix-argument the command behaves as with ECB-versions < 2.20, i.e. redrawing the full layout regardless of the current visibility-state of the ecb-windows or the compile-window. Called without a prefix-arg this command now preserves the state of the ecb-windows as well as the state of the compile-window. Called with TWO prefix-args means an emergency-redraw (see documentation of 'ecb-redraw-layout').

14.10 New command 'ecb-display-news-for-upgrade'.

ECB autom. displays after an upgrade a short summary of the most important NEWS. With the new command you can do this also on demand.

14.11 Fixed Bugs

14.11.1 Adviced 'display-buffer' handles 'same-window-buffer-names' and

'same-window-regexps' correctly.

14.11.2 Fixed a bug preventing tree-buffers with expand-symbol "before" to work

14.11.3 Fixed a bug in the adviced version of 'other-window'.

Now it works correctly also when a durable compile-window is set but currently hidden.

14.11.4 Fixed a bug when scrolling a window from within the minibuffer.

For example this bug has prevented scrolling the *Completion* buffer from within the minibuffer. Now 'minibuffer-scroll-window' is set correctly.

14.11.5 Toggling the compile-window has not worked when 'ecb-compile-window-height'

has not been saved for future sessions. This is fixed now.

14.11.6 'ecb-cycle-through-compilation-buffers' works with hidden compile-window.

14.11.7 Fixed a bug when an ECB-window was maximized and the user has stored sizes

for this layout.

14.11.8 Fixed a bug in 'ecb-major-modes-activate' and 'ecb-major-modes-deactivate'

14.11.9 Fixed a bug in the handling of the option 'ecb-kill-buffer-clears-history'

14.11.10 Fixed a bug which prevents some older version of XEmacs 21.4 working well.

For XEmacs 21.4.6 a bug was reported which causes this XEmacs-version to fail when 'compile' or 'grep' is called. The reason for this was that these old versions of XEmacs 21.4 contain versions of 'display-buffer' and/or 'get-frame-for-buffer' which do not have the fourth argument SHRINK-TO-FIT. Now ECB has fixed this and handles correctly the option 'temp-buffer-shrink-to-fit' and all shades of displaying and compiling buffers even for these versions of XEmacs.

15 Changes for ECB version 2.11

15.1 Using semanticdb to jump to type-tags defined in other files.

In OO-languages like CLOS, eieio and C++ there can be type-tags in the method-buffer which are somehow virtual because there is no definition in the current source-file. But such a virtual type collects all its outside defined members like methods in C++ which are defined in the *.cc file whereas the class-definition is defined in the associated header-file. ECB uses semanticb to open the definition-file of such a tag and to jump to the definition of this tag. Same for parent-tags in the methods-buffer. This feature can only work correctly if semanticdb is well configured!

15.2 New option 'ecb-ignore-special-display'

The options 'special-display-buffer-names', 'special-display-regexps' and 'special-display-function' work as expected. Per default this in only true, when no durable compile-window is used (see option 'ecb-compile-window-height'), i.e. with a durable compile window the special-display-options are ignored per default. But this behavior can be changed with the new option 'ecb-ignore-special-display'.

Thanks to Rob Walker <rob.lists@tenfoot.org.uk> for a first suggestion.

15.3 Better reporting if there are errors during ECB-startup

15.4 Fixed bugs

15.4.1 Clicking onto nodes with positionless semantic-tags as data doesn't fail

15.4.2 Non-semantic sources (e.g. Perl, TeX) work when cedet 1.0 is loaded

15.4.3 Fixed the versioning of ECB so the autom. upgrade works

16 Changes for ECB version 2.01

16.1 Prepared for the new semantic 2.0 beta (contained in the cedet 1 beta)

16.1.1 New naming convention

Forthcoming semantic 2.0 (part of the new cedet-library) introduces a new naming convention. Here is the relevant part of the semantic 2.0 NEWS-file:

token - Refers to a lexical analyzer production. stream - A list of tokens tag - Refers to a datastructure created by a grammar to represent something in a language file table - A hierarchical list of tags. tag-class - A tag may represent a function, data type, or variable. parse - Run a source file through a parser.

Normally there is no need for ECB users to bother with all details of semantic naming conventions but for the sake of consisteny between semantic and ECB the ECB-package has renamed all its variables and functions from "token" to "tag". Therefore also all options which contain "token" in their name have been renamed by replacing "token" with "tag". See RELEASE$_{N}OTESforalistofthisoptions$.

ECB autom. upgrades all old-values of these options to the new options! There is nothing to do for you.

16.1.2 ECB works with new semantic 2.0 which is shipped within cedet 1.0

16.2 Overhaul of the display-layer and -handling of the tree-buffers

16.2.1 Now there are 3 different styles available

* Image: Looks very nice and modern - just give it a try or see the screenshots-section at [[http://ecb.sourceforge.net][http://ecb.sourceforge.net]] * Ascii with guide-lines: Drawing the trees with ascii-symbols * Ascii without guide-lines: This is the style of ECB <= 1.96

For details and a visualization of these different styles see the new option 'ecb-tree-buffer-style' (replaces the old option 'ecb-tree-use-image-icons').

Especially the new image-style was inspired by the library tree-widget.el written by David Ponce <david@dponce.com>. Also some images shipped with ECB are "stolen" from this library.

16.2.2 No-leaf-tree-nodes with currently no subnodes are displayed with [x]

An example are directories in the Directories tree-buffer which have currently no subdirectories. Leafs like source-files or methods are of course handled as leafs and not as currently empty nodes.

16.2.3 Fixed some inconsistencies in expanding, collapsing and selecting nodes

16.2.4 Moving all options related to the tree-buffer-style or -handling to the new

customize-group "ecb-tree-buffer".

16.3 Completely rewritten popup-menu mechanism.

16.3.1 Now sub-menus are allowed for all popups

To give a better lucidity of the popup-menus of the tree-buffers these menus can now being arranged in sub-menus. The default values now already use sub-menus. So if you have added menu-entries to one of the options 'ecb-directories-menu-user-extension', 'ecb-sources-menu-user-extension', 'ecb-methods-menu-user-extension' or 'ecb-history-menu-user-extension' ECB resets these options to the new defaults of ECB and adds at the top your "old" personal

entries. This comes because the type of these options has changed. Thanks for general suggestion to Ole Arndt <ole@sugarshark.com>.

16.3.2 Added new defaults to 'ecb-directories-menu-user-extension'

Three new entries "CVS status", "CVS examine" and "CVS update" in a new sub-menu "Version control" for running CVS-commands against the directory. These menu-entries are added as new default values to 'ecb-directories-menu-user-extension' so a user can delete or change them if he does not use CVS but another revision-tool (e.g. Clearcase). Thanks for suggestion and first implementation to Ole Arndt <ole@sugarshark.com>.

16.3.3 Added new defaults to 'ecb-sources-menu-user-extension'

One new sub-menu "Version control" with some senseful commands. See also new default for the option 'ecb-directories-menu-user-extension'. Same is added to the 'ecb-history-menu-user-extension'. Thanks for suggestion and first implementation to Ole Arndt <ole@sugarshark.com>.

16.4 Possibility to define which types should be expanded at file-open-time

Semantic groups types into different type-specifiers. Current available type-specifiers are for example "class", "interface", "struct", "typedef", "union" and "enum". With the new option 'ecb-type-tag-expansion' you can specify on a major-mode-basis which type-specifiers should be expanded at file-open-time and which not. So for example in C++ it could be senseful not to expand the types with a type-specifiers "typedef", "struct" or "enum" (see the default-value of this option).

16.5 Adding a filter-feature to the Sources- and the History-buffer

Now a filter can be applied to the Sources- and/or History-buffer which desides which entries are displayed in these buffers. See the new commands 'ecb-sources-filter' (bound to [C-c . fs]) and 'ecb-history-filter' (bound to [C-c . fh]) and the new entries for the popup-menus for these buffers. In the sources-buffer each directory has its own filter. The currently applied filter is displayed in the modeline of that tree-buffer with the new face 'ecb-mode-line-prefix-face' (s.b.)

16.6 Changes related to the modelines of the tree-buffers

16.6.1 Added faces to the mode-lines of the ECB-tree-buffers.

See the three new options 'ecb-mode-line-win-nr-face', 'ecb-mode-line-prefix-face' and 'ecb-mode-line-data-face' and also they related (and equally named) new faces.

16.6.2 Moved all options related to the modelines in the new customize-group

"ecb-mode-line".

16.7 No window-restrictions if the ECB-windows are hidden

If the special ECB-windows are hidden (e.g. by 'ecb-toggle-ecb-windows') then there are no restrictions about the window-layout of the ecb-frame: The frame can be splitted in any arbitrary windows. All adviced functions behave as their originals. So the frame can be used as if ECB would not be active but ECB IS still active in the "background" and all ECB-commands and all ECB-keybindings can be used. Of course some of them doesn't make much sense but nevertheless they can be called.

Therefore it should be enough to hide the ECB-windows to run other Emacs-applications which have their own window-layout-managing. There should be no conflicts. But nevertheless the most recommended method for running ECB and other applications (e.g. xrefactory, Gnus etc.) in the same frame is to use a window-manager like winring.el or escreen.el!

16.8 Ediff runs per default in the ECB-frame.

Now ediff can run per default in the ECB-frame because ECB ensures that all special ECB-windows are hidden before ediff sets up its own window-layout. ECB also restores exactly the "before-ediff" window-layout of the ecb-frame See new option 'ecb-run-ediff-in-ecb-frame'.

16.9 More flexible directory-caching with new 'ecb-cache-directory-contents-not'

16.10 New option 'ecb-advice-window-functions-signal-error'

Now the adviced window functions of 'ecb-advice-window-functions' do not signal per default an error if called in situations which are not allowed - they

simple do nothing. An example is calling 'delete-window' in a special ecb-window or in the compile-window; if this new option is nil then nothing is done otherwise an error is signaled. If you want the old behavior of signaling an error just set this new option to not nil.

16.11 Fixed Bugs

16.11.1 When the special ECB-windows were hidden and a durable compile-window was set

then ECB shows the special ECB-windows when a temp-buffer (e.g. *Help*-buffers) has been displayed. This annoying behavior is fixed.

16.11.2 XEmacs has not added the ECB-menu to the menubar of that buffers which are

already alive while activating ECB. On the other side XEmacs has not removed the ECB-menu from the menubar for all living buffers after deactivating ECB. Both of these two bugs are fixed now.

16.11.3 JDEE-dialogs now work correctly.

JDEE offers sometimes "dialogs" where the user can choose among several options. These "dialogs" now work correctly regardless if a durable compile-window is used or not.

16.11.4 A bug in displaying the tree-buffers in a different font or font-height

17 Changes for ECB version 1.96

17.1 ECB can work together with the window-managers escreen and winring

This allows to run applications like Gnus, VM or BBDB in the same frame as ECB! See the new lib ecb-winman-support.el and read the section "Window-managers and ECB" in the chapter "Tips and tricks" in the info-manual of ECB.

Thanks to Johann "Myrkraverk" Oskarsson <myrkraverk@users.sourceforge.net> who has reported some bugs related to escreen and has therefore motivated a lot to add support for the well known window-managers escreen and winring.

17.2 ECB can display the window-number in the modeline of the special windows.

The left-top-most window in a frame has the window-number 0 and all other windows are numbered with increasing numbers in the sequence, functions like 'other-window' or 'next-window' would walk through the frame. This allows to jump to windows by name as described at the Emacs-Wiki [[http://www.emacswiki.org/cgi-bin/wiki.pl?SwitchingWindows][http://www.emacswiki.org/cgi-bin/wiki.pl?SwitchingWindows]].

See the new option 'ecb-mode-line-display-window-number'. Currently this feature is only available for GNU Emacs 21.X, because neither GNU Emacs < 21 nor XEmacs can dynamically evaluate forms in the mode-line.

Thanks to Johann "Myrkraverk" Oskarsson <myrkraverk@users.sourceforge.net> for suggesting this.

In addition the already existing options 'ecb-mode-line-data' and 'ecb-mode-line-prefixes' are now more flexible and are now able to define special modelines not only for the 4 builtin ECB-tree-buffers but for every special ECB-buffer. ECB will autom. upgrade the old values of these options as best as possible to their new types!

17.3 Much better support of the ECB-compile-window

17.3.1 General overhaul for displaying buffers in the ECB-compile-window.

The whole mechanism has been rewritten so now using a durable compile-window is much safer and works as well as without one. Now all buffers displayed with 'display-buffer' are checked if they are "compilation-buffers" in the sense of the function 'ecb-compilation-buffer-p' and then displayed in the compile-window of ECB. Especially temp-buffers displayed with 'with-output-to-temp-buffer' (e.g. *Help*-buffers) are now much better supported.

Simply said: All buffers for which 'ecb-compilation-buffer-p' says t are now *always* displayed in the compile-window (if there is any) with complete consideration of Emacs-options like 'compilation-window-height', 'temp-buffer-max-height', 'temp-buffer-resize-mode' (GNU Emacs), 'temp-buffer-shrink-to-fit' (XEmacs) etc. All other buffer not.

17.3.2 New layout-option 'ecb-compile-window-width'.

Thanks for suggestion to John S. Yates, Jr. <jyates@netezza.com>

17.3.3 New layout-option 'ecb-compile-window-prevent-shrink-below-height'

17.3.4 Better and more reliable eshell-integration. Now the compile-window fits

always autom. to the output of the last eshell-command.

17.3.5 All options related to the ECB-compile-window have been moved to the new

customization-group "ecb-compilation" which is a subgroup of "ecb-layout".

17.4 Safer handling of all adviced functions of ECB

All ECB-advices are now active if and only if ECB is running. Just loading the ECB-library doesn't activate any ECB-advice! Also if there is an error during the activation-process of ECB then all advices are always disabled automatically.

17.5 ECB can visit tokens in a smarter way

17.5.1 Narrowing via popup-menu of the Methods-buffer includes now the documentation

of the token if located outside of the token (e.g. Javadoc for Java).

17.5.2 More flexible definition what to do after visiting a token via the

Methods-buffer. See the new option 'ecb-token-visit-post-actions' which now replaces the options ecb-highlight-token-header-after-jump', 'ecb-scroll-window-after-jump' and 'ecb-token-jump-narrow'.

17.6 ECB selects autom. the current default-directory after activation even if

no file-buffer is displayed in the edit-window. This is useful if ECB is autom. activated after startup of Emacs and Emacs is started without a file-argument. So the directory from which the startup has performed is auto. selected in the ECB-directories buffer and the ECB-sources buffer displays the contents of this directory. See new option 'ecb-display-default-dir-after-start'.

17.7 ECB preserves the split-state of the frame after activation rsp. deactivation

This is done if the already existing option 'ecb-split-edit-window' has the new value t.

17.8 New command 'ecb-toggle-window-sync' for fast toggling between autom.

synchronization of the ECB-buffers is switched on and off.

17.9 The popup-menu of the Directories-buffer allows opening a dir with Dired.

The Sources-Buffer offers the same for the current selected directory. In the History-buffer the directory of the selected file-buffer can be "dired".

17.10 Now manually resizing of the ECB-windows is possible even if

'ecb-fix-window-size' is not nil for current layout. Thanks for suggestion Massimiliano Mirra <mmirra@libero.it>. Only relevant for GNU Emacs 21.X.

17.11 ECB does not call any functions of the cl-library at runtime.

17.12 Fixed Bugs

17.12.1 Fixed all the file- and directory-commands in the popup-menus.

Now the Directories- and Sources-buffers are autom. updated after creating rsp. deleting source-files or directories.

17.12.2 Jumping to some tokens for Perl sources failed. This is fixed now.

17.12.3 Now the buffer-names displayed in the History-window are always uptodate,

even if file-buffers are killed/opened during hidden ECB-windows.

17.12.4 Fixed a conflict between ECB and 'tmm-menubar' (tmm.el)

17.12.5 Fixed a lot of misspellings in the ECB-online-help and also in all comments

and strings of the sources.

17.12.6 Fixed a bug in 'ecb-toggle-enlarged-compilation-window'.

This function has been also renamed to 'ecb-toggle-compile-window-height'.

17.12.7 Fixed a bug in the layout-engine related to dedicated windows.

Thanks to Johann "Myrkraverk" Oskarsson <myrkraverk@users.sourceforge.net> who helped to find this bug.

17.12.8 Fixed some references in the info-manual of ECB.

18 Changes for ECB version 1.95.1

18.1 Now the ecb-windows can be "maximized", means all other ecb-windows are

deleted so only the edit-window(s), and this maximized ecb-window are visible (and maybe a compile-window if active). This can be done either via the popup-menus of the ecb-windows or via calling 'delete-other-windows' (bound to [C-x 1]) in one of the ecb-windows or via the new commands 'ecb-maximize-window-*' where * stands for directories, sources, methods, history, speedbar. There is also a command 'ecb-cycle-maximized-ecb-buffers'.

Note: If there are byte-compiled(!) user-defined layouts - either created interactively by 'ecb-create-new-layout' or programmed with the macro 'ecb-layout-define' - then the file where these user-defined layouts are saved (see option 'ecb-create-layout-file') must be re-byte-compiled with latest ECB version >= 1.95.1! If the user-defined layouts are not byte-compiled then there is nothing to do.

18.2 Some default key-bindings have changed; see RELEASE$_{N}OTESfordetails.$

18.3 All popup-menus of the ECB-tree-buffers can be opened via [Meta-m]

18.4 Fixed bugs

18.4.1 Clicking onto the image-icons of an integrated speedbar has failed for

XEmacs. This comes from a bug in speedbar which is now fixed locally in ECB by advicing 'dframe-mouse-set-point' if speedbar is integrated in the ECB-frame.

19 Changes for ECB version 1.95

19.1 ECB now displays the expand- and collapse symbols in the tree-buffers with

image-icons - the same icons as speedbar uses. See the new option 'ecb-tree-use-image-icons'.

19.2 Adding hideshow to the popup-menu of the Methods-buffer.

This popup-menu now offers two entries for hiding rsp. showing the block of that token in the Methods-buffer for which the popup-menu was opened. This is done with the hideshow.el library. Thanks to Christoff Pale $<$christoff$_{p}ale@yahoo.com>$ $forsuggestion.$

19.3 Horizontal scrolling of the tree-buffers by clicking the edges of the

modeline with mouse-1 or mouse-2. I.e. if you click with mouse-1 onto the left (rsp right) edge of the modeline you will scroll left (rsp. right) with the scroll-step defined in 'ecb-tree-easy-hor-scroll'. This is only for GNU Emacs because XEmacs has hor. scrollbars.

19.4 Changed default key-bindings:

- C-c . r: 'ecb-rebuild-methods-buffer' - C-c . lc: 'ecb-change-layout' - C-c . lr: 'ecb-redraw-layout' - C-c . lt: 'ecb-toggle-layout' - C-c . lw: 'ecb-toggle-ecb-windows'

19.5 Starting ECB is now possible via the "Tools"-menu

19.6 ECB now requires speedbar version 0.14beta1 or higher. An automatic

requirements-check is done by ECB. This is because now ECB has three dependencies to speedbar-code: Integrating whole speedbar in the ECB-frame, using speedbar-logic to parse files with imenu or etags and using the icons of speedbar for the ECB-tree-buffers.

19.7 Fixed bugs

19.7.1 Now the "goto-window-..." menu-entries in the "ECB"-menu are working correct

if a speedbar is integrated into the layout.

19.7.2 Preventing etags-supported sources from being parsed (and saved) too often.

20 Changes for ECB version 1.94

20.1 Supporting of non-semantic-sources.

20.1.1 Native parsing and displaying source-contents of imenu supported files

20.1.2 Native parsing and displaying source-contents of etags supported files

20.1.3 There are some new options like 'ecb-process-non-semantic-files',

'ecb-non-semantic-parsing-function' and 'ecb-method-non-semantic-face' and one new face 'ecb-method-non-semantic-face'. See new customize group "ecb-non-semantic"!

20.2 Better speedbar integration into the ECB-frame

20.2.1 Now speedbar can be used not only as replacement for the

ECB-directory-browser in the directory-window but also instead of the sources- or the methods-buffer/window. See new option 'ecb-use-speedbar-instead-native-tree-buffer' which replaces the old option 'ecb-use-speedbar-for-directories'.

20.2.2 Now speedbar can be integrated into an arbitrary layout in the same way as

the other four special buffers (directories, sources, methods and history). See new example layout with name "left-dir-plus-speedbar".

20.2.3 New option 'ecb-directories-update-speedbar' which helps you

to update a speedbar-window to the current selected directory of the ECB-directories-window.

20.3 New option 'ecb-compilation-predicates' for better specifying which buffers

should be treated as compilation-buffers and therefore be displayed in the compile-window of ECB - if there is any.

20.4 Better customizing of the mode-line of an ECB-tree-buffer.

See the new option 'ecb-mode-line-data'.

20.5 Fixed bugs:

20.5.1 Using "Add source path" from the popup-menu now doesn't open the Windows

file-dialog-box onto Windows-systems because this would prevent this command from working.

20.5.2 Smart arrowkey-navigation in the tree-buffers is now even smarter.

Thanks to John Russel <jorussel@cisco.com> for suggestion.

20.5.3 If ECB was activated with a layout which does not contain a

directory-buffer then you got always an empty directory-buffer even after switching to a layout with a directory-buffer. This is fixed now.

21 Changes for ECB version 1.93

21.1 Fixed bugs:

21.1.1 The adviced version of 'other-window' now works correct too if a

minibuffer-window is active.

21.1.2 The commands of 'winner-mode' are now only forbidden in the ECB-frame

but work in any other frame.

21.1.3 'scroll-all-mode' now works correct in other frames than the ECB-frame.

21.1.4 Removed the annoying message "Partial reparsing..."

21.2 Enhancements to the layout:

21.2.1 Slightly better redrawing current layout especially if the layout contains

a compile-window.

21.2.2 ECB autom. displays hidden compile-window before starting a compilation.

This is done if 'ecb-compile-window-height' is not nil but the compile-window is currently hidden by 'ecb-toggle-compile-window'. Of course this works also for grep and other compile-modes. Same for 'switch-to-buffer' and 'switch-to-buffer-other-window' if called for a compilation-buffer in the sense of 'ecb-compilation-buffer-p'.

21.2.3 'switch-to-buffer-other-window' now works more naturally within the

ECB-frame, especially if a compile-window is used.

21.2.4 'ecb-toggle-compile-window' now has a default key-binding: [C-c .

21.2.5 Now also layouts with user-defined special ecb-windows can be created

interactively. See the online-manual for further details.

21.3 Enhancements to the tree-buffers:

21.3.1 Now user-extensions can be added to the popup-menus of the tree-buffers.

See new options 'ecb-directories-menu-user-extension', 'ecb-sources-menu-user-extension', 'ecb-methods-menu-user-extension', 'ecb-history-menu-user-extension'.

21.3.2 The methods-buffer now has a popup-menu with senseful actions

21.4 New option 'ecb-sources-exclude-cvsignore' which allows to exclude files from

being displayed in the sources-buffer if they are contained in a .cvsignore file.

21.5 ECB now also works with buffers "online" extracted from archives.

Buffers extracted from an archive in 'tar-mode' or 'archive-mode' are now correct handled as if they were normal file-buffers. This feature doesn't work with XEmacs cause of its tar-mode and arc-mode implementation which does not handle extracted files as normal files.

21.6 ECB now checks at load-time if the required packages semantic and eieio are

at least available - regardless of the needed version. If at least one of these packages is missing then ECB offers to download and install it. More details are available in the installation instructions.

22 Changes for ECB version 1.92.1

22.1 Fixed bugs:

22.1.1 Fixing eating up a lot of CPU-power just for updating the menu-

entries of current compilation-buffers in the ECB-menu. Now the mechanism is based on a idle-timer and a cache. Fixed with great testing-help by Dan Debertin <airboss@nodewarrior.org>

22.1.2 Fixed a bug which always truncates lines in the ECB-windows regardless

of the value of 'ecb-truncate-lines'

22.1.3 The upgrade library of ECB now explicitly requires and loads the

library executable.el because Emacs 20.X does not autoload the function 'executable-find'.

22.1.4 'ecb-toggle-compile-window' now works also correct if called with

a prefix arg.

23 Changes for ECB version 1.92

23.1 Fixed bugs:

23.1.1 Fixed small bugs in the popup-menu of the history-buffer.

23.1.2 Fixed some not working links in the FAQ.

23.2 ECB now includes a file "ecb-autoloads.el" which contains all available

autoloads of ECB. Therefore loading this file is enough.

23.3 Enhancements to the layout

23.3.1 The sizes of the ecb-windows can be fixed even during frame-resizing

This new feature is only available with GNU Emacs 21.X. See new option 'ecb-fix-window-size'. Suggested by Gunther Ohrner <G.Ohrner@post.rwth-aachen.de>

23.3.2 The command 'ecb-store-window-sizes' can now also store fixed sizes.

If called with a prefix arg then fixed sizes are stored instead of fractions of frame-width rsp. -height.

23.3.3 The command 'split-window' is now also adviced to work properly.

23.4 Enhancements for the tree-buffers

23.4.1 "Create Source" in the popup-menus now work also for non-java-files.

23.4.2 ecb-truncate-lines can now be set different for each ECB-tree-buffer.

23.4.3 Easy horizontal mouse-scrolling of the ECB tree-buffers. See new option

'ecb-tree-easy-hor-scroll'.

23.5 Added a command 'ecb-jde-display-class-at-point' which displays the contents

of the java-class under point in the methods-buffer of ECB.

23.6 Renamed previous HISTORY file to NEWS and reformated it for outline-mode.

24 Changes for ECB version 1.91.1

24.1 Fixed bugs:

24.1.1 Fixes a bug in the speedbar integration which prevented clicking onto a

directory in the speedbar: Always after clicking onto a directory speedbar has re-synced itself to the directory of the current source-buffer in the edit-window. This bug is fixed, now directory navigation in the speedbar works. (Klaus)

24.1.2 Fixes a bug in upgrading stored window-sizes from ECB 1.80 to ECB >= 1.90.

Fixes also two bugs in 'ecb-store-window-sizes' if default-values are used in 'ecb-layout-window-sizes' or if ecb-windows have frame-height as height (rsp. frame-width as width) . (Klaus)

24.1.3 Fixed a bug in the navigation-history of ECB which sometimes has prevented

that a user can open files from the sources/history buffer or clicking onto tokens in the method-buffer. Now the back-forward-navigation works stable. (Klaus)

24.2 Enhanced the downloading feature of ECB

24.2.1 Fixed a bug in the downloading feature of ECB. (Klaus)

24.2.2 Now ECB autom. checks at the download sites which versions are available.

With the new option 'ecb-download-package-version-type' you can define which types of version you allow to download (only stable, stable and betas or stable, betas and alphas); see related section in the online-manual! (Klaus)

24.2.3 New command 'ecb-download-semantic' for easy getting latest semantic

versions. (Klaus)

25 Changes for ECB version 1.90

25.1 The website of ECB has moved from http://home.swipnet.se/mayhem/ecb.ht to

http://ecb.sourceforge.net. Maintainance has switched from Jesper Nordenberg to Klaus Berndl.

25.2 Fixed bugs:

25.2.1 Fixed an annoying bug which results in an error "Wrong type argument:

integer-or-marker-p nil)" after a full buffer reparse. ECB 1.80 has repaired its internal state but nevertheless the user had to re-click on the same token after this error to really jump to a token. With ECB 1.90 this bug has been gone (Klaus).

25.2.2 Fixed a small bug in 'ecb-add-all-buffers-to-history': If

'ecb-sort-history-items' is nil then now the most recently used buffers are added to the top of the history, the seldom used buffers at the bottom. Thanks to Stefan Reichr <xsteve@riic.at>! (Klaus).

25.2.3 Fixed some bugs concerning the eshell-integration (Klaus):

- Synchronizing does not set the mark anymore

- No longer synchronizing after every command if current source resides in

  the HOME directory.

- Redrawing the layout preserves the eshell-buffer in the compile-window.

- Standard emacs-hooks are not clobbered with ecb-eshell-functions if ECB is

  not active.

25.3 General enhancements:

25.3.1 Now ECB displays at every start a "Tip of the day" in a dialog-box. This can

be switched off with option 'ecb-tip-of-the-day'. There is also a new command 'ecb-show-tip-of-the-day' (Klaus).

25.3.2 Advicing 'scroll-all-mode' so it works correct with ECB, means

'scroll-all-mode' scrolls only the windows of the edit-area, i.e. max. the two edit-windows (Klaus).

25.3.3 Autom. preventing 'winner-mode' from being activated as long as ECB is

running (Klaus).

25.3.4 Now autom. (de)activating ECB via the options ecb-major-modes-activate and

ecb-major-modes-deactivate also works when opening files via dired (Klaus).

25.3.5 Now ECB can be "autoloaded", i.e.

- ecb-activate,
- ecb-minor-mode,
- ecb-byte-compile and
- ecb-show-help

are autoload-able via '(autoload ...)' (Klaus)

25.3.6 Silencing the byte-compiler, i.e. byte-compilation of ECB should now be

warning free. If byte-compilation still throws warnings during byte-compilation with 'ecb-byte-compile' (rsp. using the Makefile or make.bat) please send this to our mailing list. (Klaus)

25.4 Enhancements to the methods/variable/token tree-buffer:

25.4.1 New feature: Now the methods buffer is auto. expanded if the node related to

the current token in the edit-window is not visible (probably because its parent is collapsed). See new options 'ecb-auto-expand-token-tree' and 'ecb-expand-methods-switch-off-auto-expand' and the new command 'ecb-toggle-auto-expand-token-tree' (Klaus).

25.4.2 New command 'ecb-expand-methods-nodes' which allows precisely expanding

tokens with a certain indentation-level. There are also two new related options 'ecb-methods-nodes-expand-spec' and 'ecb-methods-nodes-collapse-spec' (Klaus).

25.5 Enhancements to the directories tree-buffer:

25.5.1 New command 'ecb-expand-directory-nodes' which does the same for the

directories tree-buffer as the new command 'ecb-expand-methods-nodes' for the methods tree-buffer (Klaus).

25.5.2 Rewritten cache-mechanism for directories and sources:

Depending on the value of the option ecb-cache-directory-contents the following is cached for a directory:

- The disk-contents of the directory means subdirs and files
- If ecb-show-sources-in-directories-buffer is nil (i.e. sources are displayed in the extra ECB-sources-window) then also the complete sources-buffer is cached. This results in a speed-boost for big-size directories.

The cache of a directory is refreshed and actualized with a POWER-click onto the related directory in the directories-buffer (s.b.) (Klaus)

25.6 Enhancements to the sources tree-buffer:

25.6.1 The option ecb-source-file-regexps now works on a directory-base, which means

for each directory-regexp the files to display can be specified. Default is an all-matching directory-regexp. Old saved values are autom. upgraded (Klaus).

25.7 Enhancements to the layout-customization:

25.7.1 Naming and managing of layouts has been changed! Now a layout is not longer

identified by an integer but by an arbitrary string! Example: The layout with index 0 in ECB <= 1.80 is now named "left1" in ECB 1.90.

- Therefore the name of the option 'ecb-layout-nr' has changed to

  'ecb-layout-name'! See the docstring of 'ecb-layout-name' for the names of all built-in layouts.

- In this context also the type of ecb-show-sources-in-directories-buffer has

  changed so it is now a list of layout-names where the sources should be displayed in the directories window (or 'always/'never). Now every change of the layout also checks the value of ecb-show-sources-in-directories-buffer.

- This change also introduces three new commands:

  - ecb-change-layout: Change layouts by selecting a name
  - ecb-delete-new-layout: Delete an user-created (by 'ecb-create-new-layout') layout.
  - ecb-show-layout-help: Displays the docstring of a layout which at least for built-in layouts contains a picture of the outline of this layout.

All three commands offer TAB-completion for easy selecting a layout name.

- ECB autom. upgrades the following options to theirs new names/types:

  - ecb-layout-nr: The stored value is transformed to the new name of the related layout and stored in 'ecb-layout-name'
  - ecb-toggle-layout-sequence: The stored value is transformed to the new type with layout-names instead of layout-numbers.
  - ecb-major-modes-activate: The stored value is transformed to the new type with layout-names instead of layout-numbers.
  - ecb-layout-window-sizes: The stored value is transformed to the new type with layout-names instead of layout-numbers.
  - ecb-show-sources-in-directories-buffer: Reset to the new default value.

  (Klaus)

25.7.2 Adding a new layout type "left-right" which allows the ECB-tree-windows to

be located at the left and the right side of the ECB-frame. See the new layouts "leftright1", "leftright2" and "leftright3" for examples. Other layouts of this type can be created or programmed very easy with 'ecb-create-newlayout' rsp. the macro 'ecb-layout-define'. (Klaus)

25.7.3 Now ECB offers a command 'ecb-create-new-layout' for interactively creating

new layouts "by example". Read the online-help. (Klaus)

25.7.4 Rewritten the mechanism for storing customized window-sizes. See option

ecb-layout-window-sizes and the command ecb-store-window-sizes and ecb-restore-window-sizes. Now the sizes are always saved as fractions of the width (rsp. height) of the ECB-frame, so the stored sizes are always correct regardless of the current frame-size (Klaus). Suggested and first implementation by Geert Ribbers [geert.ribbers@realworld.nl]

25.8 General Enhancements of the tree-windows:

25.8.1 Introduced a POWER-click (was SHIFT-click before) in the ECB-tree-windows -

see documentation of ecb-primary-secondary-mouse-buttons and see also the online-help section “Usage of ECB –> Using the mouse” (Klaus).

25.8.2 Now the ECB-tree-buffers offer in their popup-menus a "Grep" and a "Grep

recursive” (= grep-find) command where you can perform an easy grep in every directory without buffer-switching before. See also the new options ‘ecb-grep-function’ and ‘ecb-grep-find-function’. (Klaus)

25.8.3 Modified syntax-table for all tree-buffers with all paren, braces and

brackets handled as whitespace so for example no paren-matching takes place (Klaus).

25.9 Enhancements to the layout-engine of ECB:

25.9.1 Added a complete new section "The layout-engine of ECB" to the online-help

which describes in detail how to program new layouts and new special windows. This is for foreign packages which want to display own informations in own special windows and synchronizing these windows with the edit-window of ECB (a graphical debugger could be an example). Also a full working example is added in the new file ecb-examples.el. (Klaus)

25.9.2 Completely rewritten how to program new layouts. Now there is a new macro

‘ecb-layout-define’ which offers very easy programming of new layouts. (Klaus).

25.10 Enhancements to the compile-window

25.10.1 ECB now has a list of valid compilation buffers within the ECB menu. (Kevin)

25.10.2 ECB now supports 'ecb-enlarged-compilation-window-max-height' which can be

used for defining the max. height of the ecb-compile-window if enlarged by 'ecb-toggle-enlarged-compilation-window'. (burton)

25.10.3 Added new command 'ecb-toggle-compile-window': Toggle the visibility of the

compile-window of ECB (Klaus).

25.11 Speedbar is now integrated in ECB and can be used instead of the standard

ECB-directories buffer (or instead of all tree-buffers if you like :-). Most of the important basic work has been done by Kevin A. Burton.

25.11.1 See new option 'ecb-use-speedbar-for-directories' and the node "Integrating

speedbar" in the online-help. (Klaus)

25.11.2 Version-checking for correct speedbar version (>= 0.14beta1). For lower

versions ecb-use-speedbar-for-directories has no effect. (Klaus)

25.11.3 ecb-speedbar and speedbar first loaded direct before drawing the layout

with integrated speedbar. (Klaus)

25.11.4 The speedbar-commands speedbar, speedbar-get-focus are adviced during

running speedbar within ECB, so they behave in a senseful way. (Klaus)

25.12 Enhancements to the ehell-support of ECB:

25.12.1 ecb-eshell now recenters after command execution. (burton)

25.12.2 ecb-eshell now recenters if there are any window resizes. This is done so

that the prompt is always at the bottom of the window. (burton)

25.12.3 ecb-eshell integration now protects against buffer-list modifications during

transparent eshell updates in the compile window. In the past, if the eshell was running, and you want from buffer "foo", to "bar", and then tries to go back to "foo" the next entry in your buffer list is "*eshell*" so this would break a lot of completion tools like 'switch-to-buffer. (burton)

25.12.4 Added online documentation for the eshell integration. See section

"Tips and tricks". (Klaus)

25.13 New hooks:

25.13.1 New or renamed hooks:

- ecb-redraw-layout-after-hook (new)
- ecb-redraw-layout-before-hook (new)
- ecb-hide-ecb-windows-after-hook (new)
- ecb-hide-ecb-windows-before-hook (was ecb-hide-ecb-windows-hook)
- ecb-show-ecb-windows-after-hook (new)
- ecb-show-ecb-windows-before-hook (was ecb-show-ecb-windows-hook)

Read the documentation! (Klaus)

25.13.2 Added two general hooks:

- ecb-before-activate-hook: Activate ECB only if all added hooks return not nil
- ecb-before-deactivate-hook: Deactivate ECB only if all added hooks return not nil

(Klaus)

25.13.3 Added 5 new hooks for the ECB tree-buffers:

- ecb-common-tree-buffer-after-create-hook
- ecb-directories-buffer-after-create-hook
- ecb-sources-buffer-after-create-hook
- ecb-methods-buffer-after-create-hook
- ecb-history-buffer-after-create-hook

These hooks are called directly after tree-buffer creation so they can be used for example to add personal local key-bindings either to all tree-buffers (ecb-common-tree-buffer-after-create-hook) or just to a certain tree-buffer. Please read the documentation. (Klaus)

26 Changes for ECB version 1.80

26.1 ECB now requires:

- Latest stable release of semantic: Version 1.4
- Latest stable release of eieio: Version 0.17

ECB checks at start-time if the correct versions are available and if not ECB offers you to download and install them from within Emacs. After this you have only to restart Emacs and you are fine. (Klaus)

26.2 Fixed bugs:

26.2.1 Fixed: Now no empty tool-tips are displayed if mouse moves over nodes in the

tree-buffers of ECB which can be displayed completely in the tree-window. (Klaus)

26.2.2 Fixed a bug in ecb-eshell. If we are running with current buffer sync, and

there is garbage on the current line, when we change buffers this garbage will be executed which could result in a 'command not found' or even worse, a damaging command executed. We now cleanse the command line prior to directory changed. (Kevin A. Burton)

26.2.3 Fixed a bug in ecb-speedbar. The speedbar was doing its own updating outside

of the ECB. This means that if you were to change buffers in a frame outside of the ECB frame, the ECB's Speedbar would be updated to reflect a buffer not within the ECB. We now disable Speedbar's automatic update and have ECB handle this. (Kevin A. Burton)

26.2.4 Fixed a bug: Sometimes there occurs a a "wrong-type-argument"-error (or

"destroyed extent"-error with XEmacs) under some circumstances. Now ECB auto-reparses the buffer and after that ECB is in correct state again. (Klaus)

26.2.5 Fixed formatting of "Parent" nodes in token buffer. (Jesper)

26.3 Byte compiling ECB either interactive with 'ecb-byte-compile' or with the

supplied Makefile now checks also the versions of semantic and eieio. (Klaus)

26.4 ECB now displays the section "First steps" of the online-manual after

activating first time. (Klaus)

26.5 Now the adviced versions of 'delete-window' and 'delete-other-windows' can

also handle the optional WINDOW argument of the original versions correctly so the adviced versions now also work correct if called from program (Klaus).

26.6 New adviced window command: 'delete-windows-on' now works correct with ECB.

See 'ecb-advice-window-functions'. (Klaus)

26.7 In the ECB-directories buffer now F4 adds a new source-path instead of F1.

That is because F1 is such an important key in many OS and tools (e.g. opens help in all windows-programs), so ECB should not hardly "rebind" it. (Klaus)

26.8 Enhancing the option 'ecb-window-sync': Now the synchronization can take

place for all buffers which have a relation to files AND directories, i.e. now also for dired-buffers. But per default this is switched off, see the doc-string. (Klaus)

26.9 Now the ECB online help is available in the standard Info-format and also in

HTML-format. With the new option 'ecb-show-help-format' you can choose which format is used by the function 'ecb-show-help'. (Klaus)

26.10 New feature and command: ecb-download-ecb. With this function you can

download and install any arbitrary ECB-version from the ECB-website from within Emacs. This is especially useful for upgrading to the latest version. (Klaus)

26.11 New debug mode for ECB which prints some debug informations in the message

buffer for certain critical operations with semantic-overlays/extends. New option 'ecb-debug-mode' (also available in the Help-menu of ECB). (Klaus)

26.12 Some default key-bindings have changed; see 'ecb-key-map'. (Klaus)

26.13 New feature: Possibility to define a sequence of layouts with new option

'ecb-toggle-layout-sequence' and toggle very fast between them via 'ecb-toggle-layout'. Read the online-help section "Simulation a speedbar without an extra frame"! (Klaus).

26.14 Two new hooks: ecb-hide-ecb-windows-hook and ecb-show-ecb-windows-hook. Read

the documentation. (Klaus) Suggested by Daniel Hegyi <daniel_hegyi@hotmail.com>

26.15 Two new options: ecb-major-modes-activate and ecb-major-modes-deactivate.

Allow (de)activation of ECB on major-mode base. Instead of deactivation also just hiding the ECB-windows is possible. Read the docstrings. (Klaus) Suggested by Daniel Hegyi <daniel_hegyi@hotmail.com>

26.16 Redefining of the option 'ecb-compile-window-temporally-enlarge': Now you

can specify that the compile-window is auto. enlarged after selecting it and auto. shrinked after leaving it. Read the docstring! (Klaus) Suggested by Daniel Hegyi <daniel_hegyi@hotmail.com>

26.17 Much better and more powerful auto. upgrading-mechanism of incompatible or

renamed ECB-options to latest ECB-version. See the option ecb-auto-compatibility-check! (Klaus)

26.18 New option ecb-tree-RET-selects-edit-window: Possibility not to select the

edit-window of hitting RET in a tree-buffer; see the doc-string. There is also a new function ecb-toggle-RET-selects-edit-window which is bound to [C-t] in each tree-buffer. (Kevin A. Burton, Klaus)

26.19 Better grouping of external methods, i.e. methods which are implemented

outside the class-definition, like in C++, CLOS and EIEIO. Now this is done completely by semantic. (Klaus)

26.20 For application programmers using the tree-buffer library:

- The function tree-buffer-create has a new argument NODE-DATA-EQUAL-FN. See the doc-string for a description (Klaus).
- For each tree-buffer a special user-data-storage (for any arbitrary data) is created which can be accessed via 'tree-buffer-set-data-store' and 'tree-buffer-get-data-store'. See the docstring of 'tree-buffer-create'. (Klaus)

27 Changes for ECB version 1.70

27.1 New option ecb-bucket-token-display. See doc-string for details. (Klaus)

27.2 Reorganization of the faces used by ECB. The group 'ecb-face-options'

contains now all options which type is 'face (i.e. the defcustom's) and the group 'ecb-faces' contains all face-definition (the defface's). In addition it is now possible to change easily and fast the general face for ALL ECB-tree-buffers (same for the highlighting face). See the docstring of the new faces 'ecb-default-general-face' and 'ecb-default-highlight-face' for further details. (Klaus).

27.3 Smarter incr. search in a tree-buffer with the [end]-key: If there are at

least two nodes with the same greatest common-prefix than every hit of [end] jumps to the next node with this common prefix. (Klaus)

27.4 New feature ecb-auto-compatibility-check: ECB checks on startup if the

current value of an ECB-option OPT is not compatible with the type of OPT in current ECB-release. This is done during activation. If there are incompatible options ECB resets them to the default value of current ECB release and displays after activation in a temporary window which options have been reset, incl. their old-value (before the reset) and new-value (after the reset). (Klaus)

27.5 New feature ecb-tree-navigation-by-arrow: Smarter navigation in the

tree-windows with left- and right-arrow key (Klaus). Thanks for the suggestion and a first implementation to Geert Ribbers <geert.ribbers@realworld.nl>!

27.6 New feature 'ecb-type-token-display'. See the doc-string of this new option

for a description. (Klaus)

27.7 Now the option 'ecb-post-process-semantic-tokenlist' has per default an

entry for emacs-lisp-mode, so for eieio-code the methods (defmethod) of a class are grouped with respect to the class they belong. The new default value of this option does this for c++- and eieio-code. (Klaus)

27.8 Now in the ECB-tree-buffers RETURN works like the primary button (already in

previous versions) and C-RETURN works like the secondary button (new in version 1.70). For an explanation of primary and secondary see the option ecb-primary-secondary-mouse-buttons. (Klaus)

27.9 New feature: Now the token-display function can be defined separated for

each major-mode. Furthermore the token-display for C++ has enhanced, especially for template-classes (but template displaying is first available with semantic > version 14beta13!). For this the option ecb-token-display-function

has been completely rewritten and is now incompatible with the old values before ECB 1.61. (Klaus)

27.10 Fixed a bug with mouse-avoidance-mode (only relevant for GNU Emacs 20.X)

Klaus.

27.11 New feature, ecb-select-edit-window-on-redraw. Mostly used if you are running

with ecb-redraw-layout-quickly and want to make sure you are always in the edit window. (Kevin)

27.12 New feature, ecb-auto-activate. If non-nil we can start the ECB after Emacs

is started. See 'ecb-auto-activate' for more information. (Kevin)

27.13 New package, ecb-cycle.el. Supports cycling through all known compilation

buffers. (Kevin)

- ecb-cycle-switch-to-compilation-buffer - switch to available compilation buffers.
- ecb-cycle-through-compilation-buffers - cycle through all compilation buffers.

27.14 Much better performance of the directory and file-browser in ECB, especially

for directories with a lot of entries (means ~ >1000, dependent on your machine and the net-connection in case of network-drives). See also the new feature: ecb-cache-directory-contents. (Klaus)

27.15 ecb-eshell now supports customization. (Kevin)

27.16 New feature: ecb-layout-always-operate-in-edit-window. Look at the

documentation. (Kevin & Klaus)

27.17 Fixed a bug in 'ecb-toggle-ecb-windows'. For starters we weren't setting

'ecb-window-hidden' when we redraw the layout. This means that the ecb windows could be hidden but never shown. Other things:

- ecb-windows-hidden now has documentation
- we display status messages when we hide windows. (Kevin)

27.18 Fixed bug in 'ecb-redraw-layout-quickly'. If for come reason any of the ECB

windows are not alive, we do not return with a hard error. Instead, we use the scratch buffer. (Kevin)

27.19 New function ecb-toggle-enlarged-compilation-window allows users to expand and

then contract the compilation window with a new key binding C-c . / (Kevin)

27.20 new option ecb-add-path-for-not-matching-files. Look at the documentation

(Klaus).

28 Changes for ECB version 1.60

28.1 ecb-highlight-token-header-after-jump is now a boolean; which face is used

for highlighting is defined in ecb-token-header-face. (Klaus)

28.2 New option 'ecb-post-process-semantic-tokenlist': Special post-processing of

the semantic tokenlist for certain major-modes. This is useful for c++-mode. Now with the default value of this option all methods in a c++-implementation-file (no class/method-declaration but only method implementations) are grouped with respect to the class they belong. (Klaus)

28.3 Now also with C++ Sources the method- and variable-protection is displayed

correct in the ECB-method buffer (Klaus).

28.4 Fixed a bug which prevented ECB using the root path "/" on Unix-like systems

as a source-path. (Klaus)

28.5 If a file via the standard-mechanisms of Emacs (e.g. find-file) is opened,

then the auto. window sync (see 'ecb-window-sync') has worked only correct if the new file is located in any of the paths in 'ecb-source-path'. Now ECB adds the path of the new file at least temporally via 'ecb-add-source-path' to the list of paths in 'ecb-source-path', so the window-sync works always correct. ECB asks the user if the new path should saved also for future sessions. (Klaus)

28.6 Added ecb-redraw-layout-hooks so that code can be executed after we redraw the

ecb layout.

28.7 New hook ecb-current-buffer-sync-hook allows developers to add code to be

evaluated after the ECB is synchronized with the current buffer.

28.8 Now all faces used by ECB for highlighting and displaying its own stuff are

customizable. See new group 'ecb-faces' (Klaus).

28.9 ECB is now fully prepared for Emacs 21. The new feature

'ecb-show-node-info-in-minibuffer' is implemented with the new 'help-echo property of Emacs 21, therefore the ugly 'track-mouse' mechanism is not needed anymore with Emacs 21. (Klaus).

28.10 The options 'ecb-show-node-name-in-minibuffer',

'ecb-show-complete-file-name-in-minibuffer' and 'ecb-show-file-info-in-minibuffer' are gone and have been replaced by one single new option 'ecb-show-node-info-in-minibuffer' where you can define separately for every tree-buffer when and which node info should be displayed in the minibuffer (Klaus).

28.11 ecb-auto-expand-directory-tree now offers two options:

- best: Expand always the best matching source-path for a file
- first: Expand always the first matching source-path for a file (Klaus)

28.12 Fixed a bug in auto. expanding directories if a source-path has an alias.

(Klaus)

28.13 If mouse is moved over an alias in the directories buffer then the real path

is shown in the echo-area; see also 'ecb-show-node-info-in-minibuffer' (Klaus).

28.14 If mouse is moved over an history-entry then the full path is shown in the

echo-area so you can distinct better between two entries with the same name but with different paths; see also 'ecb-show-node-info-in-minibuffer' (Klaus).

28.15 New option for the history: ecb-kill-buffer-clears-history defines if

'kill-buffer' should also remove the corresponding history entry. There are several options (Klaus).

29 Changes for ECB version 1.52

29.1 Added EIEIO requirement to ECB.

30 Changes for ECB version 1.51

30.1 Now the names of all ECB buffers begin with a SPC per default. (Klaus)

30.2 ECB now can handle not only full but also partial buffer reparsing if done

by semantic and other tools (e.g. JDE >= 2.2.9). The method buffer will always be uptodate if for a major-mode auto. reparsing after buffer changes is enabled (the auto. buffer reparsing itself is not an ECB feature but must be supplied by the major-mode, e.g. JDE). (Klaus)

30.3 Incremental node-search in the ECB-buffers now uses the value of

'case-fold-search'. (Klaus)

30.4 Complete new customization of the ECB-key-bindings. The option

'ecb-prefix-key' has been removed. There is one new option 'ecb-key-map' where you can customize all key-settings of ECB (including a common prefixkey). (Klaus)

30.5 ecb-add-source-path and ecb-delete-source-path now ask if saving should be

done for future sessions (Klaus).

30.6 Added new navigator functionality. Makes it easy to go to the back and forward

in navigated tokens and buffers. See ecb-nav-goto-next and ecb-nav-goto-previous. (Jesper)

30.7 Added an option to narrow the buffer to the token that is jumped to. See

ecb-token-jump-narrow. (Jesper)

30.8 Fixed a small bug that placed the window start at the beginning of the token

instead of the beginning of the token line. (Jesper)

31 Changes for ECB version 1.50

31.1 Fixed a minor bug that occured when de-activating ECB under XEmacs. (Jesper)

31.2 ecb-layout-window-sizes variable is now an association list, which makes it

easier to add and remove layouts. (Jesper)

31.3 Added a new layout (nr. 12). Thanks to Nick Cross

<nick.cross@prismtechnologies.com> for this layout.

31.4 Each source path can now have an alias that is displayed instead of the path

name in the directories buffer. (Jesper)

31.5 The history items can now use the buffer name instead of the file

name. Customizable with ecb-history-item-name. (Jesper)

31.6 When jumping to a token the window can be scrolled so that the token starts

at the top or center of the window. This behavior is customizable with the variable ecb-scroll-window-after-jump. (Jesper)

31.7 Fixed a bug when retrieving parent names for a token. (Jesper)

31.8 Now the option 'ecb-prefix-key' is customizable so you can define another

prefix if there are conflicts with other minor-modes or packages (Klaus).

32 Changes for ECB version 1.41

32.1 Fixed bug when clicking on token in the methods buffer in XEmacs.

33 Changes for ECB version 1.40

33.1 Tree-incremental-search in the ecb-windows now ignores all non interesting

stuff:

- any leading spaces
- expand/collapse-buttons: [+] rsp. ⊟
- protection-signs (+, -, #) in the method-window if uml-notation is used
- variables types or return-types in front of variable- or method-names.
- const specifier for variables

This makes incremental-searching in a tree-buffer much faster and easier. (Klaus)

33.2 ECB now uses Semantic DB to find parents of types. (Jesper)

33.3 Long source paths in the directories buffer are now truncated at the

beginning. Customizable with variable ecb-truncate-long-names. (Jesper)

33.4 Tokens can now be sorted by access. (Jesper)

33.5 Added a function to semantic-clean-token-hooks that just updates the token

changed instead of re-building the entire token tree. (Jesper)

33.6 Implemented a token tree cache that stores recently opened buffers' token

trees. This makes the buffer switching much faster and also saves expanded nodes and window positions. The cache is unlimited at the moment. (Jesper)

33.7 Added a speedbar-like layout nr. 11. This is very useful (like also layout

nr. 10) either for users with small screens or users which normally do not need/want the ECB-windows but sometimes browsing/selecting methods/variables. (Klaus)

33.8 Better extraction of tokens from current buffer. ECB now displays any token

type in any order. (Jesper)

33.9 ECB now uses Semantic to display tokens in the methods buffer. (Jesper)

33.10 Much saver ecb-redraw-layout: Now the layout can be restored ALWAYS

regardless what messy thing has been done before. (Klaus)

33.11 New advice for 'other-window-for-scrolling', so all scroll-functions for the

"other" window can also scroll the first edit-window if point stays in the second one. (Klaus)

33.12 ecb-toggle-ecb-windows now preserves the split, the split-amount, the buffer

contents, the window starts and current point. (Klaus)

33.13 Some important changes in the ECB-layout concerning displaying compilation-

and temp-buffers. (Klaus):

33.13.1 'ecb-select-compile-window' has been gone.

33.13.2 'ecb-use-dedicated-windows' is not longer a user-customizable option but

it is always set to t, because this is essential for correct working of the ecb-layout engine!

33.13.3 'ecb-compile-window-temporally-enlarge' is now a boolean option.

33.13.4 Now ECB does all necessary that your edit-window of ECB seems to be a

normal Emacs frame, especially concerning displaying temp-buffers (like help- and completion-buffers) and compilation-buffers.

33.13.5 The variables 'compilation-window-height' and 'temp-buffer-shrink-to-fit'

(XEmacs) and 'temp-buffer-resize-mode' (GNU Emacs) are now fully supported by ECB if no durable compilation-window is shown.

33.14 New function to toggle visibility of the ECB windows. Now its possible to

hide all ECB windows without deactivating ECB (see 'ecb-toggle-ecb-windows') (Klaus).

33.15 ECB is now a global minor mode with its own menu "ECB" and its own key-map

with prefix "C-c .". New function to (de)activate/toggle ECB with 'ecb-minor-mode'. (Klaus)

33.16 Fixed a bug in highlighting current token in the method-buffer when

font-lock-mode is nil for the source-buffer. (Klaus)

33.17 New option for highlighting the method-header in the edit-window after

clicking onto the method in the method-window (like Speedbar does). (Klaus)

33.18 Function for automatically submitting a problem report to the ECB mailing

list: ecb-submit-problem-report. (Klaus)

33.19 Now not only for method-highlighting an idle delay can be set but also for

synchronizing the ECB windows with current edit window (see option ecb-window-sync and ecb-window-sync-delay; default is 0.25 sec delay) (Klaus).

33.20 Smarter highlighting of current method (Klaus).

33.21 All tree-buffers now have as default-directory the current selected

directory in the directory buffer. So you can also open files with find-file etc. from within the tree-buffers. (Klaus).

34 Changes for ECB version 1.32

34.1 Nil parent bug fixed. (Jesper)

34.2 New advices for find-file and switch-to-buffer (Klaus).

34.3 Now possible to set an idle delay before the current method is highlighted;

useful for slow machines but prevents also "jumping backward/forward" during scrolling within java-classes if point goes out of method-definition into class-definition. Default is an idle time of 0.25 seconds. (Klaus).

35 Changes for ECB version 1.31

35.1 Parents (extends and implements) of the classes in the current buffer is now

shown in the methods buffer. (Jesper). Possibility to define a regexp which parents should not be shown. (Klaus).

35.2 Now displaying the complete node name in minibuffer with the track-mouse

mechanism works also with mouse-avoidance-mode on; the mouse-avoidance will be deactivated as long ECB is activated and the node-name display-mechanism is on. This refers only to GNU Emacs! (Klaus)

35.3 Fixed a bug in 'ecb-current-buffer-sync' and 'ecb-redraw-layout' (Klaus)

35.4 Fixed a bug in loading semantic 1.3.3 or semantic 1.4 (Klaus)

36 Changes for ECB version 1.30

36.1 If not all ECB-tree-windows of current layout are visible at redraw-time

('ecb-redraw-layout') then the redraw synchronizes the contents of the tree-windows with the source displayed in current edit-window (Klaus).

36.2 Added a section "Tips and Tricks" to the ECB online-help. (Klaus)

36.3 Added a new layout Nr. 10 for very small screen resolutions where all

square-centimeters are needed for the editing itself. This layout only displays a method-window and a edit-window. (Klaus).

36.4 The messages displayed after moving the mouse over a node in a tree-buffer

do not longer wasting the log, means they will not be logged (Klaus).

36.5 ECB now also works with semantic >= 1.4 (Klaus).

36.6 Now the ecb-compile-window-height is also preserved after displaying

temp-buffers (e.g. help-buffers after C-h f) - if you want this. See the documentation of the option 'ecb-compile-window-temporally-enlarge'. (Klaus)

36.7 Added menu items for modification of the source paths in the directories

buffer. (Jesper)

36.8 Added buttons to directories buffer. (Jesper)

36.9 Mouse over files, methods etc. now work even if follow-mouse isn't

activated (Jesper). ECB adds a more intelligent mouse tracking mechanism, so not only the mouse-over-node stuff of ECB works now very well and savely but also follow-mouse itself works better and saver as without activated ECB. (Klaus)

36.10 Popup menus now work in XEmacs. (Jesper)

36.11 You can now specify paths with env-vars like $HOME in the option

'ecb-source-path'. (Klaus)

36.12 The user must now confirm if he tries to delete the ECB frame. If he wants

to proceed then ECB will be first deactivated before deleting the frame. This works for all ways to delete a frame (shortcut, window-manager-button, ...). (Klaus)

36.13 ECB now optionally create its own frame when activated. See the new option

'ecb-new-ecb-frame'. (Klaus)

36.14 Intelligent recentering of tree-buffers now completely implemented without

the function 'recenter'. This means no flickering and flashing the whole frame anymore after each 'recenter'. Now the ECB display is stable like a rock :-) (Klaus)

36.15 Mouse over directories now shows directory path. (Jesper)

36.16 Methods and variables with point are now highlighted. (Jesper)

36.17 ECB windows are now updated when saving new files and deleting files. (Jesper)

36.18 incremental-search in every tree-buffer for easier and faster selecting

nodes with the keyboard (see new option 'ecb-tree-incremental-search' and new function 'tree-buffer-incremental-node-search') for more details. (Klaus)

36.19 ecb-redraw-layout now restores both edit window buffers and the edit window

start. (Jesper)

36.20 Spelling corrections. Thanks to Colin Marquardt

<colin.marquardt@usa.alcatel.com> for spotting them.

36.21 Mode line prefixes can now be set individually for each ECB-buffer. There is

a default-prefix for each buffer but the user can also define a custom prefix or no prefix at all. (Jesper/Klaus)

36.22 Fixed small bug in directory tree that caused directories to be sorted by

extension. (Jesper)

37 Changes for ECB version 1.20

37.1 Now the ECB-buffers are intelligent in displaying nodes (Klaus):

- Expandable nodes: (e.g. a directory with its subdirectories). An ECB-tree-window now tries always to make visible as much as possible of the sub-nodes of a node if a node is expanded (e.g. by clicking onto its expand-symbol).
- Other nodes: A ECB-tree-window is always best filled, means no empty lines if there are enough lines to fill the whole window.

37.2 The source file is only parsed if the ECB methods window is visible

(Jesper).

37.3 The methods buffer now show fields of EIEIO classes. Only works with

semantic 1.4. Thanks to Eric M. Ludlam <eric@siege-engine.com> for this patch.

37.4 Now all adviced functions behave only special in the ECB-frame. In each other

frame they behave exactly like the not adviced functions. (Klaus)

37.5 Fix "invalid method-buffer"-bug (Klaus, with a lot of help by Eric Ludlam

and David Ponce).

37.6 Source files can now be sorted by name, extension or not sorted at all

(Jesper).

37.7 Now you can precisely define by an exclude and include regexp which files

should be displayed by ECB. (Jesper).

37.8 Jumping to a method/variable from the ECB-method buffer now sets the mark so

the user can easily jump back (see 'ecb-method-jump-sets-mark'). (Klaus)

37.9 Now you can define a primary and a secondary mouse-button for ECB. See the

new variable 'ecb-primary-secondary-mouse-buttons'. Note: The name of the option 'ecb-left-mouse-jump-destination' has been changed to 'ecb-primary-mouse-jump-destination'!

37.10 Now ECB is really ediff-friendly: During ediff is active the advices of ECB

are temporally deactivated! (Klaus)

37.11 Now all tree-buffers (ECB-buffers) are read-only (Klaus)!

37.12 Better implementation of advising the functions. Now all advice-able

functions are completely independent from each other (Klaus)!

37.13 Removes the redundant modeline info and shows important

information.

37.14 Added new option 'ecb-advice-window-functions'. Enabling the intelligent

window functions via customize. No hook hacking necessary anymore! Now the intelligent window functions are implemented by advicing (Klaus).

37.15 Better online help display (Klaus).

37.16 Added new option 'ecb-primary-mouse-jump-destination' (thanks to David Hay

for suggestion <David.Hay@requisite.com>.

37.17 Adviced 'other-window' acts now exactly like the original but always related

to 'ecb-other-window-jump-behavior' (Klaus)!

37.18 Fixed bug with 'ecb-compile-window-height'. Now ECB has always the correct

compile-window height after compilation-output (compile, grep etc.)

37.19 New option 'ecb-compile-window-temporally-enlarge'. Allowing

Emacs-compilation to enlarge the ECB-compile-window temporally.

38 Changes for ECB version 1.10

38.1 Automatic expansion of the directory tree.

38.2 Method parsing is now recursive, so that for example inner classes

in Java are shown in the methods buffer.

38.3 Variables are now shown in the methods buffer.

38.4 Selected files and directories now use the secondary-selection

face. Thanks to Kevin A. Burton <burton@relativity.yi.org> for this patch.

38.5 Fixed some issues with XEmacs. Thanks to Colin Marquardt

<colin.marquardt@usa.alcatel.com> for this.

38.6 Fixed bug with truncating lines in tree-buffer.el.

38.7 The methods buffer can now be automatically updated when the source

file is saved.

38.8 Help text is available with "ecb-show-help".

38.9 Moving the mouse over a node shows the name in the

mini-buffer. Currently this only works when follow-mouse is activated.

39 Changes for ECB version 1.0

39.1 Changed name from JCB to ECB and prefix from "jde-jcb-" to "ecb-".

39.2 Works with semantic 1.3.1 and higher.

39.3 Fixed a couple of bugs when creating and deleting files.

39.4 Grouped customization variables.

39.5 Contributed by Klaus Berndl <klaus.berndl@sdm.de>:

39.5.1 A lot of intelligent ECB window-functions (e.g. ecb-delete-window) which

gives you the feeling as if the edit-window is the only window of the ECB frame.

39.5.2 New layout functionality.

39.5.3 The history buffer can optionally be sorted and also cleared.

39.5.4 The package expanded symbol can optionally be placed before the package name.

39.5.5 Syntax highlighting of the methods buffer.

39.5.6 Show argument names and return type in the methods buffer.

39.5.7 History clear function: ecb-clear-history.

39.5.8 Added pop-up menu to the history buffer.

39.5.9 Fixed edit buffer sync bug.

40 Changes for ECB version 0.04

40.1 Keyboard navigation in JCB buffers. Use Return to select and Tab to expand.

40.2 Tree view of packages.

40.3 Class files can be viewed in the packages buffer.

40.4 Better handling of mouse events.

40.5 Automatic update of the packages buffer.

40.6 Added pop-up menus to the packages and classes buffers. They are activated

with mouse button 3 (right button).

40.7 Added method sorting. Thanks to Brian Anderson <brian.anderson@luna.com>

for this improvement.

41 Changes for ECB version 0.03

41.1 Fixed naming of variables.

41.2 Fixed license and copyright comments.

41.3 Fixed class file regular expression bug.

Thanks to Jim Loverde <loverde@str.com> for finding it.

41.4 Added three new layouts.

41.5 The JCB buffers can now be (almost) synchronized with the edit window.

41.6 When clicking on a class or method using mouse button 2, it will be loaded

into another window. A new customization variable was added for this.

41.7 Minor changes. Thanks to Jim for his suggestions.

41.8 Removed the "jde-jcb-window-skips-from-methods-buffer" and

“jde-jcb-window-skips-from-history-buffer” customization variables.

42 Changes for ECB version 0.02

42.1 Changed prefix from "jcb-" to "jde-jcb-". This will cause JCB customization

variables to be saved in JDE project files.

42.2 Lists and parses non-Java file types. Currently the C and Lisp parsers from

semantic-ex.el are used.

42.3 History buffer implemented.

42.4 Layout function implement. Thanks to Jim Loverde <loverde@str.com> for

helping out with this.

42.5 Packages matching a regular expression can be excluded from the packages

buffer. Thanks to Mark Gibson <mark@markg.co.uk> for this feature.

43 Changes for ECB version 0.01

43.1 Initial version.

Local variables: mode: outline paragraph-separate: "[
]*$" end: