

MECAFF - tools

Multiline External Console And Fullscreen Facility - Tools

for VM/370 R6 SixPack 1.2

Release-Notes

Version 1.1.0

Dr. Hans-Walter Latz

WARNINGS:

This software is delivered as-is with no promise or commitment to be usable for any particular purpose.

Use it at your own risks!

The MECAFF software and documentation has been written by a hobbyist for hobbyists and should not be used for any important or even critical tasks.

MECAFF is work in progress and its current implementation may differ from this documentation.

1 1.0.0 ⇒ 1.1.0

- General
 - Starting with version 1.1.0, the MECAFF fullscreen tools EE, FSLIST, FSVIEW and FSHELP as well and IND\$FILE support the now available DIAG58 interface for VM/370 R6 SixPack 1.2
 - The fullscreen tools including IND\$FILE are now delivered in 2 separate suites, one suite together with MECAFF itself (as before) working with the MECAFF-console and one suite for the DIAG58 based usage with “plain” 3270 terminals.
 - Both suites in the same version are intended to support the same functionality and subcommand sets, with only minimal differences due to the basic technical differences of both fullscreen support approaches.
 - Documentation
 - To reflect the additional availability for the DIAG58 support, the MECAFF manual has been splitted to better support the 2 suites of the MECAFF tools.
 - The manual for the MECAFF-tools (this document) is the documentation common to both suites, describing the functionality and usage of the fullscreen tools (EE, FSLIST, FSVIEW, FSHELP) and IND\$FILE.
 - Each suite has a separate manual describing the specific functionalities and installation for the suite.
- EE
 - New functionality
 - Command NULLs
specify if white space after a line end is rendered as NUL or blank characters on the 3270 screen
 - Command WORKLrecl
specify the working (displayed and editable) line length for the file (see section *Configuration commands* in the Manual for relevant usage).
 - Modifications
 - Command FTDEFAULTS
new optional parameter to specify the default working line length for the file type.

2 0.9.8 ⇒ 1.0.0

- General
 - Bugfixes
 - Message “Error parsing file id: Fileid component too long” when invoking FSLIST or FSVIEW with correct file-ids
symptom: if the EXECs for FSLIST or FSVIEW are invoked with a fileid component with a length of exactly 8 (like *fslist * assemble*), the message mentioned is sometimes issued and the command ends with RC = 4
fix: more a work-around for a bug (apparently) in the GCCLIB runtime when building the argv-array for main(): in absence of an EPLIST, PLIST values with exactly 8 characters are sometimes terminated only after the 9. Character
 - Documentation
 - Section *IND\$FILE – File transfer* in the Manual now tries to explain how to create an IND\$MAP file for IND\$FILE.
- EE
 - New functionality
 - Prefix command: *[n]
duplicate this line; if the optional repetition-count *n* is given, this line is duplicated *n* times (i.e. repeated *n* times)
- FSLIST
 - Modifications
 - PF02/PF12
when returning from EE (PF02) resp. FSVIEW (PF12) the cursor will be at the position where it was when the PF key was pressed
- FSHELP
 - Modifications
 - Help menu
the menu page has a new group “others” where help topics of type HELPCMD not having the words CMS or CP on the first line are listed;
the help menu file should be regenerated after installing the new version with
FSHELP (REBUILD
 - PF02/PF12
these PF keys now behave like PF01 (open help topic under the cursor), giving some mental compatibility with FSLIST

3 0.9.7 ⇒ 0.9.8

- General
 - New command: IND\$FILE
 - Implementation of 3270 file transfer with MECAFF supporting CUT-mode transfer
 - Supports downloading and uploading files with LRECL up to 255 in ascii or binary mode
 - Allows customized remapping of characters in ascii transfer mode
 - Delivered with the C source code of the program
- MECAFF process / MECAFF API
 - Modifications
 - new API routine `__fslkto()`
the routine allows to specify the timeout value that will reserve the fullscreen mode to the CMS program under certain conditions: if a `__fsrdp()` call uses the exact timeout specified with `__fslkto()`, MECAFF will not take back the ownership over the screen if asynchronous output came in since the program last got the ownership, as long as the timeout (plus grace period) is not reached;
this allows IND\$FILE to do undisturbed file transfers, as messages from other users or like TAPE 18x ATTACHED will not force the console into the More... state, which would result in a protocol error for the file transfer
 - WCC-handling
the MECAFF process does no longer try to control the keyboard locking by resetting the *RestoreKbd*-bit of the WCC-byte of W-/EW-/EWA-commands on fullscreen writes and inserting synthetic Write-commands to set the *RestoreKbd*-bit on fullscreen reads; WCC-bytes are now transmitted unchanged through the MECAFF-API to the terminal, so programs using the API can longer rely on the keyboard always being unlocked after the user's input
- EE
 - New functionality
 - New subcommand MOVEHere: if the cursor is in the file area, the line under the cursor will be made the current line, with the cursor being moved with this line and staying at the position inside the line; MOVEHere is bound to PF09 per default
 - Modifications
 - Programmers input mode
when leaving PInput, EE will try to place the cursor in the file area, preferably leaving the cursor at the current position or if this not in the file area by placing it on the same or the last inserted line at the first non-blank column.
 - Prefix commands
some (up to now) single line prefix commands now accept an optional positive repetition-count modifier, which is added after the command character and has the value 1 if omitted:
Icount: insert *count* lines
Dcount: delete *count* lines, limited up to the next prefix command on the screen
"count: duplicate *count* lines (including the line with the prefix command)
 - command CHANGE
this command now accepts the optional CONFirm parameter; if given, each matching position is displayed on the screen and changes must be confirmed (PF12), can be skipped (PF04) or the change operation can terminated prematurely (PF03)

- commands PGUP and PGDOWN
these commands now accept the optional parameter MOVEHere; when specified and the cursor is in the file area when the command is issued (typically with PF07/PF08), the line under the cursor will be made the current line, with the cursor being moved with this line and staying at the position inside the line
- FSHelp
 - New functionality
 - New subcommands:
TOp → scroll to the top of the help topic
BOttom → scroll to the bottom of the help topic
- FSLIST
 - Modifications
 - PF07/PF08
if the cursor is in the file area, FSLIST will now attempt to (vertically) center the line under the cursor on the screen;
as before, scrolling up/down will occur by 2/3 of a screen height if the cursor is not in the file area
 - New functionality
 - New PF-key bindings:
PF01: if the cursor is in the file list, FSLIST will attempt to (vertically) center the line under the cursor on the screen;
if the cursor is on the command line, the cursor will be placed in the file area as near as possible to the middle of the screen
 - New subcommands:
TOp → scroll to the top of the list
BOttom → scroll to the bottom of the list
- FSVIEW
 - Modifications
 - PF07/PF08
if the cursor is in the file area, FSVIEW will now attempt to (vertically) center the line under the cursor on the screen;
as before, scrolling up/down will occur by 2/3 of a screen height if the cursor is not in the file area
 - PF11/PF12
if the cursor is in the file area, then horizontal shifting will now try to center the cursor's column; as before, shifting by 20 columns to the left/right will occur if the cursor is not in the file area
 - New functionality
 - New subcommands:
TOp → scroll to the top of the file
BOTtom → scroll to the bottom of the file
 - New PF-key bindings:
PF01: if the cursor is in the file area, FSVIEW will attempt to center the character under the cursor on the screen (scrolling both vertically and horizontally); if the cursor is on the command line, the cursor will be placed in the file area as near as possible to the middle of the screen
PF23/PF24: horizontal shift to the left/right by 10 columns

4 0.9.6 ⇒ 0.9.7

- General
 - Session initialization

the first installation tape now contains the file CKMECAFF EXEC intended to be called from the PROFILE EXEC to check if the 3270 terminal is connected via MECAFF. If this is the case, an automatic initialization is performed (see below) and a user specific script (MCFFPROF EXEC) is executed if present, allowing to customize the MECAFF session only if MECAFF is active (so FS-QRY will not block if the 3270 terminal emulator is directly connected).
 - Synonyms

a new part of the MECAFF installation is also a SYNONYM file allowing to register the aliases FSLIST and FSVIEW for the EE command. These synonyms are intended as replacements for the corresponding EXECs.

The synonym table is automatically registered by CKMECAFF if the MECAFF process is present.
- MECAFF process
 - Bugfixes
 - protocol error reported by fullscreen synchronous read after non-synchronous _fsrdp()
symptom: some combinations of a synchronous fullscreen read directly following an asynchronous read lead to a protocol error (returncode 2) after no user activity in the grace period of the asynchronous read
fix: the grace period counter is now reset in all cases when a synchronous read occurs, preventing the state mismatch due to the wrong grace period timing out.
 - The WCC-bytes (Write Control Character) for the W-, EW- and EWA-commands in outbound streams are now translated with the 3270 6-bit encoding scheme to provide the parity bits (e.g. a 0xC3 WCC code is now sent instead of 0x03). This should increase the compatibility with terminal emulators verifying the parity information.
 - New command line parameter -sendDelay:n
 - Defines the wait period in milliseconds used by the MECAFF console before it sends a screen modification to the terminal, with a default of 0 ms (as it was before 0.9.7)
 - Depending on the speed of the hosting hardware, setting it to at least 2 ms will allow to use the QWS emulator program which can be found on the MVS turnkey CD distribution (this QWS version has problems with the fast recurring screen redraws and crashed with MECAFF before 0.9.7 and still crashes if -senddelay is too small for the hosting hardware)
 - New command line parameter -minColorCount:n
 - Defines the minimal number of colors that an emulator must support to be accepted as a color terminal. The number of supported colors is reported by the emulator in response of the WSF-Query issued during the tn3270 negotiation phase (if the parameter -noDYNAMIC is *not* specified). The default value for the minimal color count is 4.
 - Support for 3270 emulators without extended data stream support

if the terminal type transmitted by the emulator during the tn3270 negotiation phase does not indicate that the terminal supports extended data streams (i.e. the terminal type name does not end with '-E'), MECAFF will not attempt to query the terminal capacities with an extended data stream query, sticking to the default capabilities as identified by the terminal type (e.g. 24x80, no colors, no extended highlighting if it is a IBM-3278-2)

- EE / FSVIEW / FSLIST
 - Bugfixes
 - Failures when opening files with LRECL > 255
symptom: trying to edit or browse such files from the command line or from the FSLIST mode resulted in different errors, ranging from simply closing the program without a message or with messages about overwritten memory blocks (either by DMSFRET or by the EE-internal memory manager) up to ABENDs
fix: the internal state handling for the file to be opened has been corrected
 - Modifications
 - file identifications can now be specified either with the traditional CMS convention *fn ft fm* or with the dot-notation (used by many other OSes) *fn.ft.fm* (i.e. the 3 file-id components can be separated with whitespaces or dots, even mixed)
 - when specifying a new file-id (or pattern for FSLIST) it is possible to reference components from the current context with the equal sign (=), for example:
 - while editing the file `FOO C A1`, specifying `= h` will open `FOO H A1`
 - having the pattern `X* MODULE S` in FSLIST and giving `l = . = . a` will list the files `X* MODULE A`
 - if file-ids with the dot-notation are to be used on the CMS command line, the EXECs for FSLIST and FSVIEW will no longer work, as the command line is tokenized by the EXEC interpreter to 8 character chunks, which will truncate most useful file-ids in dot-notation.
 Using the synonyms for EE instead of the EXECs resolves this problem (at least under VM/370 R6 SixPack 1.0 or newer). See the *Installation manual* for available options.
- EE
 - New functionality
 - New prefix command @
 marking a line with this prefix command tells EE to ignore any modifications made to this line (this is very useful if an EE command was entered by inadvertence in the file area instead of the command prompt line)
 - Modifications
 - commands PGUP and PGDOWN
 these commands now accept a numerical parameter to define the paging height either as percentage of a page height or as page height minus a number of lines;
 the keys PF19 and PF20 are now bound to scroll up resp. down by 2/3 of a screen
 - prefix commands
 after executing a prefix command, EE now tries to position the cursor in the file area, with the line selected for placement depending on the prefix operation and the cursor being on the first non-blank character of the line resp. preserving the indentation for the input prefix command

5 0.9.5 ⇒ 0.9.6

- General
 - New functionality :
 - Online help files for EE and FS* commands are provided on an additional tape file
 - Help files for the MECAFF CMS commands are standard CMS help files (filetype HELPCMD) supported by CMS HELP and FSHELP
 - Help files for the EE subcommands and EE (toplevel internal help) are non-standard help filetypes HELPEE and HELP_EE supported by FSHELP only
- EE
 - Bugfixes
 - ABEND after SplitJoin in programmers-input-mode
symptom: after appending the current inputline to the previous line with a SplitJoin (PF06/PF18), the inputline of the programmers-input-mode disappears and EE crashes (ABEND) when leaving the programmers-input-mode with PF03.
fix: a new inputline is generated if the current one is joined away with SplitJoin.
 - misplaced inputline after SplitJoin in programmers-input-mode
symptom: after splitting the current inputline with a SplitJoin (PF06/PF18), the inputline of the programmers-input-mode is placed between the first and second line of the split, which is unexpected.
fix: the new inputline is now inserted behind the second line of the split.
 - New functionality :
 - New Subcommand Help: Show the top-level help topic for EE
- FSLIST
 - Bugfixes
 - PF-Keys
symptom: the PF-keys PF06 to PF09 scrolled the double distance, as the handling routine for scrolling was called twice.
fix: redundant calling the mentioned routine introduced with code changes for repeating the last search was removed.
 - New functionality :
 - New Subcommand Help: Show the help topic for FSLIST
- FSVIEW
 - New functionality :
 - Horizontal scrolling:
if the LRECL of the file is larger than the line width available on the screen, only the first columns fitting in the space available were displayed.
The file content can now be scrolled horizontally inside the usable screen width with the following PF keys:
 - PF11: move the first displayed column of the file by 20 positions to the left
 - PF12: move the first displayed column of the file by 20 positions to the rightShifting the horizontal screen space over the file content is bounded to the file width (LRECL).
The file format and the currently visible column range is shown in the head line.
 - New Subcommand Help: Show the help topic for FSLIST

6 0.9.4 ⇒ 0.9.5

- MECAFF-Process / MECAFF-API
 - access to version numbers (*Major.Minor.Sub*) of the connected MECAFF process and of the MECAFF-API through the new API call `__fsqvr()`
 - Linking against PDPCLIB: a minimal version of the GCCLIB is now provided, allowing to use the MECAFF-API with the PDPCLIB
- FS-QRY
 - new command line parameter `VERsions` to output the version data of the connected MECAFF process and the MECAFF-API
- EE
 - New functionality
 - New Subcommands: `TABforward`, `TABBackward`, `TABSet`, `FTTABDEFaults`
 - Support for software tab stops:
 - `TABSet` (re)defines tab stops for the current file
 - `FTTABDEFaults` defines default tab stops for a file type
 - tab positions currently defined are marked on the scale line
 - Moving the cursor to a tab position is supported in the file area for normal edit mode, for input-mode and for programmers-input-mode:
 - `TABforward` (PF01) moves cursor to the next tab position
 - `TABBackward` (PF13) moves cursor to the previous tab position or the first column
 - If invoked from the command line, `TABforward` (PF01) moves the cursor to the last known cursor position (file line and column) in the file area or to the current line if the last known file line position is not visible or was deleted.
 - New subcommands: `SHIFT` and `SHIFTCONFig`
 - `SHIFTCONFig` `CHEckall` | `MINimal` | `LIMit` | `TRUNCate` [*shiftBy*]
define the default shift behavior when the shifted text reaches a line border and the default shift count
 - `SHIFT` [*by*] `Left`|`Right` [*count* | *:line* | *.mark*] [*shiftMode*]
shift lines around current line left or right, truncating or limiting to borders
 - New mode: `programmers-input-mode`
 - New command `PInput`, assigned per default to PF10
 - adds a single new line with the indentation of the preceding line on entering `PInput` resp. on each Enter-Key if the current line was modified or no other line was changed
 - all file lines on screen can be edited in addition to the current input line
 - `programmers-input-mode` has own fixed PF settings:
 - PF01/PF13 : TAB forward / backward
 - PF03/PF15 : leave `PInput`
 - PF06 : `SPLTJOIN`
 - PF10: move `PInput`-line after this line(if cursor is in the file area)
 - New subcommands: `SEArchnext` and `R[EV]SEArchnext`
 - `SEArchnext` (bound per default to PF04, with shortcut: `/`):
 - if the last `LOCATE` command had exactly one target specified as search pattern
 - repeat previous text search in the last search direction (down or up)

- REVSEArchnext / RSEArchnext (bound per default to PF16, with shortcut: -/):
 - if the last LOCATE command had exactly one target specified as search pattern
 - reverse the search direction and repeat previous search in the new direction
 - New prefix commands
 - single line shift: < , >
 - block of lines shift: <<..<< , >>..>>
 - shift prefix commands can have options: [1..9][?:|!|#]
 - 1..9 : shift-by count, default 1
 - ?|:|!|# : shift mode, with ? = CHEckall, : = MINimal, # = LIMit ! = TRUNCate
- Modified functionality
 - Subcommand CMS
 - when invoked without a CMS or CP command as parameter, EE leaves temporarily the fullscreen mode and the CMS Subset level is entered on the MECAFF console. This allows to enter CP/CMS subset compatible commands until the command RETURN is entered, which will leave the CMS subset and restore fullscreen EE editing.
 - Subcommand SPLTJOIN
 - Split: the new line (rest of the splitted line) now has the same indentation as the original line
 - Join: leading spaces are removed from the joined (second) line before appending to previous line at the cursor position
 - Subcommands Quit and QQuit
 - if prefix commands are present on the screen or are pending from a previous screen, these commands now act like RESET: prefix commands are removed (instead of executing them) and the current file (resp. EE) are not closed.
 - this gives the PF03 key (to which Quit is usually assigned) a generic "escape" functionality depending on the current mode:
 - leave Input resp. PInput mode
 - reset/cancel prefix operations
 - close the current file
 - close EE
- FSLIST / FSVIEW / FSHELP
 - New functionality : Repeat previous search
 - Subcommand / or PF04: repeat the last search in the same direction
 - Subcommand -/ or PF16: reverse search direction and repeat search in new direction
 - (both subcommands must be given without trailing blanks, as this would start a new search for the whitespace given)
- FSVIEW
 - New functionality : the line number of the first displayed file line on the screen and the total number of lines of the file are displayed in the top header line

7 0.9.3 ⇔ 0.9.4

- MECAFF-Process
 - Bugfixes
 - Dependency of the JRE setup type

symptom: the MECAFF-console does not work at all (hangs after showing the Hercules connection screen, ignoring any input from keyboard or output from VM/370) when the Java runtime (JRE) was installed with the *net installation* setup instead of the *full installation* setup. The net setup is a small file with about 900 KB, which downloads the remaining 11 MByte of the JRE setup, the full setup has about 16 MByte in a single file (all sizes for Win32); as the size difference suggests, the net setup seems to be a subset of the full install, at least missing the EBCDIC charset support required by MECAFF.

fix: MECAFF no longer relies on the Java built-in text conversion, but now uses its own internal translation tables for EBCDIC↔ASCII conversion (bracket EBCDIC charset).
- MECAFF-Console
 - New functionality
 - The PF01 key now issues the CMS command FSHELP as default, opening the (new) help menu page (see below).
- FSHELP
 - Support for navigation to a new help topic by moving the cursor to a word in the current help text and pressing the PF01 key. If available, the help topic corresponding to the word under the cursor is opened.

The help topics are stacked in a last-in-first-out fashion: the PF03 key now closes the current help topic and returns to the previous topic. Closing the first (initial) topic terminates the FSHELP program. The program can be left from any topic on the stack with the PF15 key (generally the shift-PF03 key combination).
 - Invoking FSHELP without parameters now displays a help menu giving an overview over all help topics available on disk U. This help topic page is generated if not available on an accessible disk (or if FSHELP is invoked with the option REBUILD), grouping the help topics on disk U by command type (CMS, CP etc.). This menu page is saved to the file MENU FSHELP A2 and is then displayed as help page. This help file will subsequently be used if the menu is to be displayed.
 - New subcommands:
 - *Help topic*
→ open the help for *topic*, if available.
 - */text/*
→ search forward for *text*
 - *-/text/*
→ search backward for *text*
- FSLIST
 - New subcommands:
 - */text/*
→ search forward for *text*
 - *-/text/*
→ search backward for *text*
- FSVIEW
 - New subcommands:
 - */text/*
→ search forward for *text*
 - *-/text/*
→ search backward for *text*

8 0.9.1 ⇒ 0.9.3

- MECAFF-Process / MECAFF-API
 - New functionality
 - GRAF-style:
 - MECAFF now emulates a minimal 3270 model 4 (80x43 geometry) instead of model 2 (with 80x24 geometry)
 - support for all 3270 model types (2 and 4) configurable in DMKRIO
 - all 3270 lines of a VM/370R6 under Hercules machine can now be used
 - support for LU-names (group names in Hercules)
 - connections from MECAFF to Hercules can optionally be bound to a group of 3270 lines
 - a default LU-name can be given on the command line for the MECAFF process
 - this default can be overruled by a LU-name of the connecting terminal
 - if no LU-name is specified, the MECAFF connection is bound to an arbitrary device assigned by Hercules (as in MECAFF versions before 0.9.3)
 - new command line parameter `-vmLUName` for the external MECAFF program
 - API:
 - additional variants for fullscreen reads, supporting
 - polling read / query availability of user input, allowing asynchronous processing
 - time out read, returning control to the program after the time interval if no user input occurred
 - new API calls:
 - `__fsrdp()` : polling/time-out fullscreen read
 - `__fsgp()` : set grace period for terminal ownership after `__fsrdp()`
 - `__fscncl()`: cancel fullscreen operation (release terminal ownership)
 - new sample program FSRDTEST (C source and executable)
 - Bugfixes
 - API-Call `__fsrd()`:
 - fix: rc 4 (protocol error) is now returned whenever the input buffer is too small for the 3270 input stream received
 - symptom: passing `outbufferlength <= 0` locks up communication with the MECAFF process
 - GRAF-mode connections to VM
 - fix: race and deadlock conditions removed in background MORE [→HOLDING] →RUNNING handshake, occurring mostly when outputting large amounts of lines on TN3270-emulators allowing fast AID keystrokes (which do not ensure a minimal time interval when sending consecutive user inputs, unlike wc3270/x3270)
 - symptom: pressing ENTER in a fast sequence could lock up the handshake mechanism for the session, preventing further output from VM/370 on that terminal
 - GRAF-mode connections to VM
 - fix: introduction of minimal time intervals between input sent to VM, both for internal handshaking input (ENTER-Aid for MORE-→HOLDING or CLEAR-Aid for MORE|HOLDING-→RUNNING) and for user input
 - symptom: fast input to VM (e.g. CLEAR then ENTER in less than 1 ms) can lead to a quantum-physical state of the VM after FORCE-ing it, as the user seems to be both logged off and on:
 - *QUERY user* and *FORCE user* claim that this user is not logged on

- QUERY NAMES lists the user a logged on
 - trying to log on claims the user is still logged on
- MECAFF-Console
 - New functionality
 - attribute settings (color, highlight) for visual elements of the console can now be queried (CMS command FS-QRY) and modified (new CMS command FS-CTL)
 - setting of PF-keys can now be queried (CMS command FS-QRY) and modified (new CMS command FS-CTL)
- FS-QRY
 - New functionality
 - Now allows to list the terminal characteristics (as previous versions, subcommand TERM), the display attributes of the console (subcommand ATTRS), the PF-key settings (subcommands PFKEYS) or all these informations (subcommand ALL)
 - Also allows to simply query if the terminal is connected via MECAFF and return the state in the exit code (return status 0 or 1)
- FS-CTL
 - New command
 - Set the display attributes for visual elements of the MECAFF console
 - Set the command issued through the MECAFF console when pressing a PF key is pressed, either as command sent to the VM or processed internally in the console (like paging through the output history etc.)
- EE
 - Bugfixes
 - Prefix command single-line duplicate ("): false error message ("invalid prefix command") after executing command removed
- General
 - The MECAFF tools for CMS are now delivered in 2 versions:
 - linked against the memory resident GCCLIB as it is standard in SixPack 1.2 (called "dynamically" linked)
 - linked against the runtime files of GCCLIB loaded into resident memory (called "statically" linked)

9 0.9.0 ⇒ 0.9.1

- EE
 - New functionality
 - Multi-file editing capability, including prefix block operations between files (source and target of copy/move operations in different files)
 - Integration of FSLIST with EE, allowing to use FLIST to open a file for editing
 - Handling of disconnect and reconnection to a new terminal (re-establish full-screen session, with fallback to a rescue session if not connected via MECAFF)
 - New commands
 - EEdit *fn* [*ft* [*fm*]]
 - EXit
 - RINGNext , RN
 - RINGPrev, RP
 - FSLIST [*pattern*]
 - PUT, PPUT, PUTD, PPUTD [*fn* [*ft* [*fm*]]]
 - GET, GETD [*fn* [*ft* [*fm*]]]
 - DElete [*count*]
 - Extended commands
 - QQuit [ALL]
(new optional parameter ALL to close all edited files without saving)
 - Input [*line-text*]
(optional text after the command will insert a new line with the text instead of entering input mode)
 - New prefix commands
 - " (duplicate line)
 - "" .. "" (duplicate block of lines)
 - Bugfixes
 - DD prefix command: defining the boundaries of a DD-block in different round-trips (different screens) no longer leads to repetitive error messages forcing EE out of full-screen mode
- FSLIST
 - New functionality
 - Integration of EE with FSLIST , allowing to edit a file from FSLIST (with PF2)
 - the color settings for FSLIST (and FSVIEW) are initialized from the corresponding ATTR commands in the profiles for EE
 - New commands
 - Sort
 - Bugfixes
 - when changing the list with command Listfile, the pattern showed in the title line is adjusted accordingly instead of displaying the initial file pattern of the FLIST session
 - the header line for the file list is now shown with the HEADLine attribute (blue as default) instead of the FILE attribute (green as default), as it not part of the list
- FSVIEW
 - File viewer functionality is now available directly from CMS
 - Switch to editing the file browsed with EE (with PF2)