

V I S T A

# MSC File Manager 22.2v3

## Release Notes

March 2015



©Copyright 2015 by VISTA Expertise Network. Licensed under Creative Commons Attribution-ShareAlike 4.0. Details are available at <http://creativecommons.org/licenses/by-sa/4.0/>.

The developers of MSC Fileman 22.2 and the authors of this manual would like to extend their thanks to Medsphere Systems Corporation for their generous contribution of the MSC Fileman 1051 codebase to the OSEHRA community on 21 November 2014, and for their decade of support for Fileman guru George Timson.

## Revision History

Date	Description	Language	Authors
March 2015	MSC FM 22.2v3 update	English (US)	Frederick D. S. Marshall and Kathy Ice
March 2013	FM 22.2v2 release	English (US)	Sam Habel and Frederick D.S. Marshall
October 2012	FM 22.2v1 development	English (US)	Kathy Ice, Sam Habel, and Frederick D.S. Marshall, based on the work of Tami Winn, Michael Ogi, Thom Blom, and Susan Strack



# MSC File Manager 22.2v3 Release Notes

## 1. Principal Enhancements

### 1.1. Screenman Enhancements

#### 1.1.1. Use of Mouse in Screenman Forms

To use your mouse with Screenman forms, your terminal emulator needs to be able to receive ANSI standard control sequences to turn the mouse on and off. Set up your terminal emulator so that the keyboard emulation is in "VT100+ mode." Once you are in a Screenman form, click the mouse at the command line or any of the fields in the form.

If you do not want Screenman to recognize mouse clicks, use the Kernel parameter (created by running ^DINIT) that is named "DI SCREENMAN NO MOUSE".

#### 1.1.2. Record Selection as a Full Screenman Page

The Screenman Record Selection page can now be a full Screenman page using a computed multiple pointer, so that the user can select an entry by scrolling up or down. This new feature lets forms contain embedded lookups.

You can set this up automatically when you create a form. At the query "Do you want your Form to begin with a display of all entries, for selection," answer "Yes." The initial position can be set to be the user's last selection, rather than first, last, or new.

### 1.1.3. Expanded Multiples

Multiples within a single Screenman page can now be more than one line deep.

### 1.1.4. Custom Colors Option

The "Customize Colors" suboption within Screenman allows selection of ANSI colors for all Screenman presentations, on a parameterized basis (user, institution, etc.) using Kernel parameters.

### 1.1.5. Quick Exit from Word Processing Pages

Editing a Screenman word-processing page ends with two empty lines, so that "F1-E" is not required to leave the page.

### 1.1.6. Indication of Word-processing Data

A + now indicates in a Screenman form whether a word-processing field already contains data.

### 1.1.7. Screen Print

"F1-P" allows printing of the screen (including all multiples).

## 1.2. *Internationalization*

### 1.2.1. Fileman is Translation-ready

All hard-coded display strings in File Manager have been converted to the Fileman's Dialog framework, so translation can be table-driven.

### 1.2.2. New Entries in Dialog File (.84)

Many new entries have been added to the Dialog file to handle all end-user interactions.

### 1.2.3. Many New Languages in File .85

The Language File (.85) can now be extended by running ^DMLAINIT to include all ISO 639-standard languages.

### 1.2.4. New Dialog Framework for Data Dictionary Elements

File names, field labels, set values, and help messages can be entered into the ^DD schema for any of the languages listed in File .85.

### 1.2.5. New Entry Points to Help Translate DD Elements

New direct-mode tools have been created to help translate DD elements.

For example, to enter Spanish: DO SPANISH^DIALOGZ or DO LANG^DIALOGZ(3).

### 1.2.6. Consistent Date Formatting

Formatting of date output is now consistently done throughout all the end-user routines. Changing the global node ^DD("DD") will change the way all Fileman dates are output. Re-running ^DINIT will not change this node.

### 1.2.7. International Date Input

Fileman's internationalization framework has been extended to improve support for international dates. International date input ("I" parameter) allows alpha dates and forms such as "2010-1-31", "20100131", and "31 Jan 2010."

### 1.2.8. Upper/Lowercase Translations are Consistent

Fileman's internationalization framework has been made consistently independent of the ASCII character set, to improve support for international case conversion.

### 1.2.9. Two- and Three-letter Language Abbreviations

The Language file (.85) now can store two-and three-letter abbreviations for languages.

## 1.3. *Security Improvements*

### 1.3.1. Modified Auditing Menu

The auditing menu has been modified to be better organized and more intuitive.

### 1.3.2. Automatic DD Auditing

To improve version-control for data dictionaries, DD changes are always audited (in File .6). There is no need to turn on DD auditing file-by-file.

### 1.3.3. Showing Past Changes to Data Dictionary

The “Show Past Changes to DDs” auditing sub-option shows all DD changes since a certain date.

### 1.3.4. New API AUDIT^DICATTA

AUDIT^DICATTA(FILE,FIELD,OLD,NEW,ATTRIB) was added to support manually adding audit-log records for DD changes, such as when manually altering DDs during the installation of patches or new versions.

### 1.3.5. Showing Changes by a Specific User

“Monitor a User” is now the second auditing option. It shows every entry in an audited file touched by a given user in an audited file.

### 1.3.6. Entry Access Audit

The entry point ACCESSED^DIET will record access to a file entry in the

audit file. Previously Fileman could only record audits for added, modified, or deleted fields. Now it can record access to a specific entry in a file.

### 1.4. Data-analysis Improvements

#### 1.4.1. New Verify Pointers Option

The Verify Fields option has been converted into a new Verify Data menu, which contains Verify Fields and a new Verify Pointers option. Verify Pointers will find and report dangling and badly defined pointers.

#### 1.4.2. Check all Pointers into a Given File

A fourth Data Dictionary utility (“find pointers into a file”) checks all files with pointers into a given File. The utility gives 4 kinds of output (here using Patient file (2) as an example):

1	One particular PATIENT Entry
2	All PATIENT Entries
3	Non-existent PATIENT Entries
4	Entries from a PATIENT Search Template

#### 1.4.3. The Meta Data Dictionary

Run ^DDD during Fileman install to create the new Meta Data Dictionary file (.9). The Meta Data Dictionary lists all fields in all files in a searchable format.

#### 1.4.4. Improvements to the Verify Fields Utility

The Verify Field utility has had many improvements. It now looks for duplicates and dangling pointers in cross-references. It has been made interruptible. If a field has an Output Transform, it is now validated to make sure it produces valid input that would be accepted by the field’s Input Transform. It double-checks thirty-character-limit index values to make sure they are properly extracted and set. It displays the translated

field label, if any. A bug was fixed to suppress the accidental echo of dates when verifying a date-type field.

#### 1.4.5. DD Listings Now Validate Output Transforms

If an output transform has been set by hand by the programmer, it will lack the O flag in the DD that tells Fileman to execute it, so unbeknownst to the developer it will never run. DD Listings now flag such inactive output transforms with the message "NOT EXECUTABLE!! -- SPECIFIER NEEDS AN 'O!'"

#### 1.4.6. Comparing Data and Data Dictionaries across Environments

A new Transfer menu option, Namespace Compare, lets you identify differences in data and DDs between different MUMPS environments, to help with version control.

#### 1.4.7. Sorting: Computed FROM and TO Values

The Print File Entries option has been upgraded to allow FROM and TO values to be stored as computed expressions, so they are recalculated in the future whenever they are run. If computed expressions are entered at FROM and TO prompts preceded by an @, they will be stored as computed expressions instead of fixed values.

#### 1.4.8. Totals & Subtotals: Output Transform Now Applied

When totaling or subtotaling a numeric field, Fileman now applies that field's output transform to the total or subtotal, to preserve the field designer's intended formatting on these derived values.

### 1.5. *Documentation Improvements*

#### 1.5.1. New & Overhauled Manuals

An *OSEHRA Technical Journal Article*, a substantive *Security and Privacy*

*Manual*, and a *Value Proposition Manual* have been added to Fileman's documentation suite; see the *Tech Journal Article* for an expansion of these *Release Notes* and the *Value Proposition* for an explanation of MSC Fileman 22.2's strategic importance. The previous *Getting Started Manual* and *Advanced User Manual* have been recombined to form a new *User Reference Manual*, and *Getting Started* and *Advanced User* have been simplified and reorganized to serve as a two-part introduction to Fileman. The *Technical Manual* has been upgraded to include Fileman's Integration Agreements. All manuals in the suite have been upgraded to comply with OSEHRA's new Licensing and Attribution Policy.

## **2. Other Enhancements**

### *2.1. User Interface Changes*

#### **2.1.1. Select Prompt: Extended Selection by IEN**

Lookup enhancement: if the .01 field of the file being selected from is a pointer to another file, you can use a double accent grave (`) to pick a pointed-to entry by its IEN.

#### **2.1.2. Allow Terminal Emulators Deeper Than 24 Lines**

Fileman now supports longer screens.

#### **2.1.3. Printing Multiples in Sorted Order**

Until now, when printing sorted records, any subentries within those records were displayed unsorted, in order by internal entry number. A new B sort specifier will ensure that subentries are displayed in order.

#### **2.1.4. Enhanced Control of Input Templates**

Within input templates, subfiles can now be edited in more than one place within the template, so that different subfields can be edited each time.

### 2.1.5. Fileman Browser Enhancements

Fileman will be optimistic and assume the terminal emulator supports the browser, rather than consult the Kernel.

You can now quit the browser using CTRL-E.

You can now print the text being browsed using F1-F1-P.

### 2.1.6. Canonic Templates

For each Fileman file, the developer may designate a template to be the canonic template of that type for that file; printing, sorting, or editing (appropriate to the template type) will default to the use of that template, unless the user chooses otherwise. Fileman utility Template Edit includes new field Canonic for This File, where the attribute can be set. Once a template has been designated canonic, it can no longer be edited on the fly within the Edit, Print, or Inquire options; only with Template Edit. Canonic templates are identified as such in DD listings.

### 2.1.7. Utility Identifier Now Supports Write-only Identifiers

Write-only identifiers historically had to be created or edited with hard Set commands from programmer mode. The utility Identifier now includes user-interface support for creating, editing, and deleting them.

## 2.2. API Changes

### 2.2.1. Finder & Lister: Enhancements to FIND^DIC and LIST^DIC

- 1) Finder & Lister: Third argument (fields) can now be a computed expression, not just a field.
- 2) Finder & Lister: New E flag returns the complete list of matches even if errors are encountered during the generation of the results.

3) Lister: Eighth parameter (index) of LIST^DIC can now be either a sort template or a field or a computed expression, if the new X flag is included.

### 2.2.2. New API to Create Sort Templates Silently

BUILDNEW^DIBTED will silently create a sort template.

## 2.3. Data Dictionary Changes

### 2.3.1. Auditable Word Processing Fields

Fileman security has been improved by allowing word-processing fields to be audited.

### 2.3.2. Word Processing Fields Can be Made Uneditable

Reference files and clinically significant text can now be protected from subsequent change.

### 2.3.3. Set Explicit Maximum Length for Free-text Fields

Maximum Field Length is now an independent field attribute, not just a side-effect of the code contained in the field's Input Transform.

### 2.3.4. Override of Character Limit in Globals

^DD("STRING\_LIMIT"), if set, overrides the standard 255-character limit throughout File Manager.

### 2.3.5. Set Cross-references as Non-re-runnable

Cross-references can be defined so that they are never refired after their initial execution.

## *2.4. Installation and Distribution Changes*

### 2.4.1. DIFROM: Keys and New-style Indexes

DIFROM has been extended to be able to transport keys and new-style indexes.

### 2.4.2. DINIT: Virgin Install

Fileman has been changed to restore its ability to run correctly without any of the rest of VISTA being installed.

### 2.4.3. NOASK^DINIT: Install without Asking Questions

New entry point NOASK^DINIT has been added to run the Fileman install without asking the user any questions.

### 2.4.4. SAC & OSEHRA Compliant

The first and second lines of all routines—including INITs and NTEGs—have been standardized to comply with VA's VISTA Standards and Conventions. New standard third and fourth lines have been added to comply with OSEHRA's new Licensing and Attribution Policy.

## **3. Bug Fixes**

### 3.1. Computed Expressions: Multiple Contains with Word-processing Fields

The computed expression (WP1["GREEN"])(WP2["GREEN"]) is now correctly handled (where WP1 and WP2 are names of Word-processing fields).

### 3.2. Lookup: Input Value Longer Than 30 Characters

Lookups with input values longer than 30 characters now work correctly.

### 3.3. Cross-reference a Field or File: Duplicate Index Names

Duplicate index names are no longer allowed.

### 3.4. Uppercasing

Fileman uppercasing code in DILIBF is now User Language aware. Previously it worked only for English only.

### 3.5. Captioned Fields in Screenman When Language is Not English

Fileman didn't allow you to create a Caption Only field in Screenman Editor when the User Language isn't English. This has been fixed.

### 3.6. DIFROM Maximum Routine Size

DIFROM didn't use ^DD("ROU") to determine the maximum routine size, but rather had it hard-coded to 9999.

### 3.7. DIFROM Routine-size Calculation

DIFROM didn't correctly count the size of routines it created making routines larger than the maximum size.

### 3.8. Browser Display Routines Did Not Work on Cache

DR^DDBRU didn't work on Cache due to the way %RSEL works on Cache.

### 3.9. Maximum Routine Size Did Not Get Set with the First Installation of Fileman.

The node ^DD("ROU") does not get initialized when installing Fileman for the first time.

### 3.10. Browser Now Works without Kernel

The Fileman Browser now works without the device file and can be invoked from Fileman directly.

### 3.11. Replace/With Maximum Length

The replace prompt only supported input up to 245 characters long. Now it supports any length up to ^DD("STRING\_LIMIT").

### 3.12. Reverse Collation on Complex New-style Indexes when Doing Partial Matches

Previously, a compound new-style index configured to sort in reverse order still sorted in forward order upon partial matches even when it sorted in reverse when listing entries using "??". Now, it will correctly sort in reverse order when displaying partial matches.

### 3.13. U Flag on ^DIC and Q Flag on LIST^DIC and FIND^DIC and Partial Numeric Matches on Pointer Values

Previously, using the U flag on ^DIC and the Q flag on DBS DIC calls caused numeric matches on pointer fields to be partially matched as text matches. Now, partial numeric matches are not allowed in these circumstances.

### 3.14. Functions YEAR, MONTH, and DAY Now Operate on Internal Values

Previously, these three functions operated on the external field value, which although useful for end users was often unrecognizable to Fileman as a date and led to broken calculations (such as "YEAR(DOB)" resulting in a nonsensical answer such as "12/0000"). These functions now ignore any output transform on the field, to ensure the calculations produce accurate answers.

### 3.15. Date Concatenations No Longer Crash with Hard Errors

Computed expressions that included concatenations with date values (such as "TODAY\_(TODAY-1)") caused Fileman generated code that caused syntax errors when executed. This has been fixed.

### 3.16. Sort Qualifier ;TXT Now Only Allowed for Free Text Fields

The TXT sort qualifier was intended to give users control over how a free-text field was sorted (numerically or as text), but its use was not properly constrained. Now Fileman rejects its use on other data types.

### 3.17. Deleting Output Transform Sometimes Deleted Patient File's DD

A bug involving naked references sometimes caused the Patient file's data dictionary to be deleted! The code has been fixed to prevent this from happening.

### 3.18. Fileman RCP DDR FILER: Fix Flags Parameter in ADD Mode

DDR FILER accepts a mode parameter (the third), which can be set to "EDIT" to invoke the Filer or "ADD" to invoke the Updater. It also accepts a flag parameter (the fourth), but prior to this change, FILEC^DDR3 carefully prevented this flags parameter from being passed to the Updater; the flags only controlled the Filer. Electronic Health Solutions of Jordan noted that this "protection" was nonsensical, since the Updater's flags were equally useful. The specific example that inspired them was the ability to pass in the E flag to process external values, which is the most common way one wants to use the Updater.

## 4. Unit Tests

Fileman 22.2 comes with Unit Tests for classical date/time calls and new features of the DBS LISTER and FINDER. All unit tests are under the namespace DMU.