V I S T A

File Manager 22.2

Release Notes

March 2013

File Manager 22.2 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. Data Analysis Tools

1.3.1. 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
```

1.3.2. Automatic 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. 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.

1.3.5. 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.

1.3.6. 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.

1.3.7. 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.

1.3.8. 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.

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.

2.2. API Changes

2.2.1. Enhancements to FIND^DIC and LIST^DIC

1) Third argument (fields) can now be a computed expression, not just a field.

2) New E flag returns the complete list of matches even if errors are encountered during the generation of the results.

3) 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.

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.