

*Venue*

# MEDLEY 1.2-S RELEASE NOTES

Release 1.2  
October, 1990

---

Address comments to:  
Venue  
User Documentation  
Suite 102  
1555 Lakeside Drive  
Oakland, CA 94612  
408-737-4443

---

## MEDLEY 1.2-S RELEASE NOTES

Release 1.2

October, 1990

Copyright © 1990 by Venue.

All rights reserved.

Envos is a trademark of Envos Corporation.

Medley is a trademark of Venue.

UNIX® is a registered trademark of UNIX System Laboratories.

The following are trademarks of Sun Microsystems Inc.:

Sun® is a registered trademark

SunOS

SPARC

Sun Workstation® is a registered trademark.

Copyright protection includes material generated from the software programs displayed on the screen, such as icons, screen display looks, and the like.

---

The information in this document is subject to change without notice and should not be construed as a commitment by Venue. While every effort has been made to ensure the accuracy of this document, Venue assumes no responsibility for any errors that may appear.

Text was written and produced with Venue text formatting tools; Xerox printers were used to produce text masters.

These release notes provide warnings and information important to the successful running of Release 1.2-S of Medley for the Sun Workstation. These sections are followed by listings of known and fixed bugs in Release 1.2 of Medley. A section containing changes for specifying the size of UNIX process space follows the first (warning) section.

---

## Warnings

---

Following is a listing of warnings based on the current performance of Medley 1.2-S. They are listed in numerical order by Action Request (AR) number.

- 
- 7260** The XCL compiler cannot compile a LET form that includes more than 15 special variables to be bound to non-NIL values in the local variable list. If you want to bind more than 15 special variables in a single LET form, you should break up the group into several nested LET forms.
- 9352** Using **DEFSTRUCT** to define a structure whose name is the same as any built-in data type will cause unpredictable problems. **DO NOT USE IT.** There is no workaround. This behavior is required by the CLtL specification.
- 11037** Sketch hardcopy causes a printer error. Due to the difference of scanning direction for different printers, you may cause a printer error when you try to print shaded patterns.

This is not a problem in the US.

**NOTE:** Medley prints shaded regions (filled curves and polygons) on Interpress printers by scan-converting the texture it is filling with. If the scan direction (X or Y) Medley uses does not match the laser-scan direction on your printer, you may crash the printer.

There are two global variables you can use to avoid this problem.

**IL:PRINTER.DEFAULT.SCAN.DIRECTION** [Variable]

Use this if the printer name specified by a user cannot be found in **IL:PRINTER.SCAN.DIRECTIONS.LIST**. This variable is initially set to **Y**.

**IL:PRINTER.SCAN.DIRECTIONS.LIST** [Variable]

This variable contains a listing of the scan direction associated with a printer name specified as the destination for hardcopy.

For example:

<u>Printer</u>	<u>Direction</u>
8040 series	Y
M-35 series	Y
XP-9 series	X
XP-11	X

Example:

```
(SETQ PRINTER.SCAN.DIRECTIONS.LIST
      ('("Lautrec:" . Y) ("Dali:" . X)))
```

- 11173** If a Sketch is larger than one page, it cannot be printed using TEdit.
- If you try to print a TEdit document that includes a Sketch larger than one page, TEdit keeps reformatting the document.
- To avoid this, make the Sketch small enough to fit on one page, or separate the Sketch into smaller sketches.
- 11208** The XCL compiler recognizes a **DECLARE** form in Common Lisp's executable position as an Interlisp **DECLARE** form. You can therefore only use an Interlisp specifier in that position. If you try to use a Common Lisp declaration specifier, an error message will print.

---

## Changes for Specifying Size of UNIX Process Space

---

In prior versions of Medley, there was no way to specify the amount of virtual memory Medley should use. It always allocated 32 Mbytes for the sysout. With Medley 1.2-S, you can specify an arbitrary virtual memory size for Medley. However, there are some restrictions, as explained below.

To specify the memory size at system start-up, use the following:

```
ldeether [<SYSOUT-name>][-m<memory-size>][other options]
```

-m Specifies the memory size

*memory-size* 8 through 32 (Mbytes)

**NOTE:** When you use -m, the value of `IL:STORAGEFULLSTATE` in the sysout you start should not be 3 or 4. Those values mean it already used more than the 8 Mbyte space in the sysout. Because of the Medley storage management architecture, the virtual memory size cannot be changed after `IL:STORAGEFULLSTATE` has been set to 3 or 4.

This value can be examined just before ( `IL: LOGOUT` ) if you want to specify the virtual memory size during the next start-up.

Example: `ldeether /usr/LISP.SYSOUT -m 16`

The example above means 16 Mbytes of virtual space will be assigned for Lisp.

---

## Information

---

Following is a listing of information you will need to successfully run Release 1.2-S for the Sun Workstation. These information notes are listed in numerical order by Action Request (AR) number.

**NOTE:** The Medley 1.2-S garbage collector (GC) has been modified to trigger the reclamation process according to the actual number of objects registered in the reference count table, rather than the number of allocated objects.

**10238** { `DSK` } and { `UNIX` } devices support the notation in that the three meta characters ( `.`, `..` and `~` ) can be used together, as shown in the following example:

`{ DSK } ~/../Tom/foo.lcom`

In the example, { `DSK` } device interprets `Tom` as one of the subdirectories of the parent directory belonging to the user's home directory.

**11049** File names are represented in more canonical form. The function which returns the full file name returns it in the canonical form: `{ DSK } <usr>etc>` rather than `{ DSK } /usr/etc/`.

This change will make some tools (such as `COPYFILES` and `SAMEDIR`) which depend on the conventional file name representation described in the *Interlisp-D Reference Manual* work correctly on the Medley 1.2-S file system.

**11071** Uppercase "C" can now be used in URaid. It checks the contents by scanning all stack space in the sysout. For example:

```
0x11880 BF,[ivar:0x1800]
0x11802: FX for CL:T[ ]
0x11816 BF,[ivar:0x1816]
0x11818: FX for IL:\TURN.ON.PROCESSES[ ]
```

For more information about URaid, refer to Chapter 8 (Error Recovery) in the *Medley for the Sun Workstation User's Guide*.

- 11123** If you try to use `IL:RENAMEFILE` to move a file from one device to another (e.g., NS file server to {`DSK`}) when you don't have delete access to the source, the following is returned:  
`XCL: :FS-RENAMEFILE-SOURCE-COULDNT-DELETE`. Two proceed cases are provided:
- Invoke `DELETE-DESTINATION` to restart: the destination file is also deleted and `RENAMEFILE` returns `NIL`.
  - Invoke `DONT-DELETE-DESTINATION` to restart: `RENAMEFILE` returns the destination filename without deleting the destination file.

---

## Known Bugs

---

The following is a list of known bugs in Medley 1.2-S. Bugs are listed in numerical order by Action Request (AR) number. Each AR in this list has an explanation of the problem and, where appropriate, a workaround.

- 11102** (`FQUOTIENT 0.0 0.0`) should obey `\OVERFLOW`, and not always return `0.0`

Floating-point division of `0.0` by `0.0` returns `0.0` instead of obeying Interlisp's `\OVERFLOW` flag.

---

## Fixed Bugs

---

The following is a list of bugs fixed in Release 1.2-S. Bugs are listed in numerical order by Action Request (AR) number. Where applicable, the item includes an explanation of the bug as it occurred in prior Medley releases, followed by how it has been fixed for the current release.

- 2999** `MAKEFILE` always puts the `LCOM` file on the connected directory, rather than on the specified directory.

Prior: `MAKEFILE` always put the `LCOM` file on the connected directory, even though a filename was specified using the full pathname.

Current: `MAKEFILE` now puts the `LCOM` file on the specified directory, as long as the filename supplied uses the full pathname. If the full pathname is not supplied, `MAKEFILE` puts the `LCOM` file on the connected directory.

- 6953** The `:EXPORT` option is not available for `DEFSTRUCT`.

Prior: There was no `:EXPORT` option in `DEFSTRUCT`.

Current: `DEFSTRUCT` now accepts the `:EXPORT` option. The `:EXPORT` option's specifications are as follows:

**:EXPORT** can be **T** or the following `<export-args>` expressions:

- `<export-args> ::= <export-arg> | (<export-arg-list>)`

- `<export-arg-list> ::= = <export-arg> |`  
`:constructor :accessor |`  
`:constructor :predicate |`  
`:constructor :copier |`  
`:accessor :predicate |`  
`:accessor :copier |`  
`:constructor :accessor :predicate |`  
`:constructor :accessor :copier |`  
`:constructor :predicate :copier |`  
`:accessor :predicate :copier |`  
`:constructor :accessor :predicate`  
`:copier`

- `<export-arg> ::= = :accessor | :constructor |`  
`:copier | :predicate | T`

**T** implies '(**accessor** :**constructor** :**copier**  
:predicate).

When the **:EXPORT** option is used, **DEFSTRUCT** exports the specified auxiliary functions generated automatically. For example, if (**:export** :**copier** :**accessor**) is used, only the **copier** and **accessor** generated automatically by **DEFSTRUCT** are exported.

**NOTE:** If auxiliary functions are exported, they will remain the exported functions until the user explicitly unexports them. (Redefinition without the **:EXPORT** option has no meaning.)

**7337** **PARSE-DEFMACRO** generates a spurious **IGNORE** declaration.

Prior: The **PARSE-DEFMACRO** utility, for parsing **DEFMACRO**-style destructuring argument lists, generated an incorrect **IGNORE** declaration when the argument list was **NIL**.

Current: **PARSE-DEFMACRO** now does not generate an incorrect **IGNORE** declaration with a meaningless new symbol.

**7800** Although **CLtL** says the **CL:FUNCTIONP** returns **T** for a **LAMBDA** list only, it actually returns **T** for any list.

Prior: The function returned **T** for any list.

Current: It now returns **T** only for a **LAMBDA** list.

**8017** **BRECOMPILE** no longer finds the old **LCOM** if it is not on the same directory.

Prior: **BRECOMPILE** gave up searching for the **LCOM** file if the **LCOM** file did not exist in the same directory.

Current: **BRECOMPILE** now searches **DIRECTORIES** to find the **LCOM** file if the **LCOM** file does not exist in the same directory.

8578 Neither **.FONT FONT1** nor **.FONT 1** works with **PRINTOUT** (IRM III:25.27).

Prior: **PRINTOUT** did not work correctly with **.FONT FONT<n>** or **/FONT<n>**.

Current: The new module called **FONTNUMBER** has the following variable definitions. If this module is loaded, **PRINTOUT** works correctly with these variables (**FONT1** through **FONT7**).

```
FONT1  (GACHA 10) (GACHA 8) (CLASSIC 8)
FONT2  (GACHA 10 MRR) (CLASSIC 8 BRR)
        (CLASSIC 10 MRR)
FONT3  (GACHA 8) (CLASSIC 6 MRR) (CLASSIC 8 MRR)
FONT4  (GACHA 10) (CLASSIC 10 BRR)
        (CLASSIC 10 MRR)
FONT5  (HELVETICA 10) (HELVETICA 8)
        (CLASSIC 8 MRR)
FONT6  (HELVETICA 10 BRR) (HELVETICA 8)
        (CLASSIC 8 MRR)
FONT7  (GACHA 12) (GACHA 12) (CLASSIC 12)
```

8912 Although **CLtL** says the **CL:REQUIRE** loads files only when not previously loaded, it actually always loads files.

Prior: The function always loaded files regardless of whether it had been loaded before.

Current: The function now checks whether the file has been loaded previously, and loads the file only when it has not.

9334 **MANAGER** is confused by **MAKEFILE** without **COMPILE**. The file is not marked "changed," although the file manager actually reflects the change.

Prior: When you used **MAKEFILE**, the main **MANAGER** window did not reflect the new changes correctly.

Current: The main **MANAGER** window now correctly reflects the new changes.

9361 Although **CLtL** says the **CL:MERGE-PATHNAMES** 'default-version' option is assumed to be **:newest** unless otherwise specified, the default is actually **NIL**.

Prior: If you tried (merge-pathname "foo.bar" "{dsk}"), you got "{dsk}foo.bar", which is what (merge-pathname "foo.bar" {dsk}) nil) returned.

Current: If you now try (merge-pathname "foo.bar" "{dsk}"), you get "{dsk}foo.bar;", which is what (merge-pathname "foo.bar" {dsk} :newest) returns.

- 9376 (DECLARE: ... (FILES X)) marks files as dirty.**
- Prior: If file A included (FILESLOAD B) form, and file B included (DECLARE: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY (FILES A)) form, FILES A was marked as changed after file A was loaded.
- Current: The FILEPKG does not mark FILES A as changed after the loading of file A.
- 9454 CL:GETF expands its argument form illegally. This causes argument forms to be evaluated twice or more.**
- Prior: If you ran (setf (getf (getf x (baz)) t) z), the argument form (baz) was evaluated twice.
- Current: If you now run (setf (getf (getf x (baz)) t) z), the argument form (baz) is evaluated only once.
- 9936 The XCL compiler does not do tail-recursion elimination for Interlisp LAMBDA form.**
- Prior: Tail-recursion elimination was performed only for Common Lisp LAMBDA forms.
- Current: Tail-recursion elimination is now performed for both Common Lisp and Interlisp LAMBDA forms.
- 10020 IL:CALLS returns a list of three elements instead of four when run on compiled code.**
- Prior: When CALLS worked on the compiled code, it returned a list of three elements (free variables and global variables were merged). And, when CALLS' USEDATABASE argument was non-NIL, CALLS also returned a list of three elements.
- Current: CALLS always returns a list of four elements.
- 10196 OUTPUT FILENAME appended to the MasterScope command fails.**
- Prior: The following three comands failed:
- WHO CALLS WHO OUTPUT CROSSREF did not close the CROSSREF output file.
  - WHO IS USED FREELY OUTPUT FREE could not be parsed.
  - WHO IS CALLED BY SomeFuncs OUTPUT TEMP sent the output to the terminal rather than the TEMP output file.
- Current: The three commands now succeed, as follows:
- WHO CALLS WHO OUTPUT CROSSREF creates the output file, sends the output, and closes it.
  - WHO IS USED FREELY OUTPUT FREE is parsed correctly, and the output file is created and closed.

- **WHO IS CALLED BY SomeFuncs OUTPUT TEMP** sends the output to the **TEMP** output file and closes it.

**10463 THROWS compiled by the XCL compiler waste CONS cells.**

Prior: The XCL-compiled **THROW** form wasted **CONS** cells because it used multiple-value lists for passing the return value.

Current: The XCL-compiled **THROW** form no longer wastes **CONS** cells.

**10937 FileBrowser does not get information for files ending in a tilde.**

Prior: **FileBrowser** could not get any information about the files that end in a tilde, but did get numerous errors in the prompt window saying: error status = 2 "No such File."

Current: On both **{DSK}** and **{UNIX}**, **FileBrowser** can now get information for files ending in a tilde.

**10953 Although CLtL says the CL:GET-SETF-METHOD guarantees that exactly one store variable will be returned, it actually does not check the number of store variables.**

Prior: **CL:GET-SETF-METHOD** did not check the number of store variables returned by the **SETF** method. Also, **CL:GET-SETF-METHOD-MULTIPLE-VALUE** was the same as **CL:GET-SETF-METHOD**.

Current: **CL:GET-SETF-METHOD** now checks the number of store variables to be returned by the **SETF** method. **CL:GET-SETF-METHOD-MULTIPLE-VALUE** does not check it.

**11013 DIR \* COUNTSIZE results in an error when it hits a subdirectory.**

Prior: The **COUNTSIZE** option of **IL:DIRECTORY** resulted in the error message **NON-NUMERIC-ARG** when it encountered a subdirectory.

Current: The **COUNTSIZE** option of **IL:DIRECTORY** works on a subdirectory. It now counts the proper size of the directory file.

**11016 Reading directory from maiko {UNIX} loops printing error message.**

Prior: **(IL:OPENSTREAM "{UNIX}dirfile" 'IL:INPUT)** succeeded, but the **BIN** looped repeatedly, printing the error message "System call error: read errno = 21 is a directory".

Current: An input stream can be opened with the **{UNIX}** device, and you can then read from it. No error message is printed in a prompt window.

- 11017** Calling the **DIRECTORYNAMEP**( {**UNIX**}**FOO**) function when **FOO** is not a directory results in an inappropriate error message.
- Prior: On {**DSK**}, the **DIRECTORYNAMEP** function returned **NIL** or **T** when appropriate. On {**UNIX**}, the same function also resulted in the following error message being printed in the prompt window: "System call error: stat errno = 2 No such file or directory."
- Current: The function **IL:DIRECTORYNAMEP** on a {**UNIX**} device returns **NIL** or **T** as appropriate, without printing a redundant message in the prompt window.
- 11058** Filepackage command (**DECLARE:FIRST . . .**) is not implemented in the **CL:COMPILE-FILE**.
- Prior: If filepackage command (**DECLARE:FIRST . . .**) was specified, the following warning message was printed: Warning: (**DECLARE:--FIRST-- --**) not implemented in **COMPILE-FILE**.
- Current: (**DECLARE:FIRST . . .**) is now supported, and the form specified by it is put to the head of the file and compiled.
- 11078** {**DSK**} should write new versions of a file in the same case as the old.
- Prior: On {**DSK**}, new versions of an existing file were created isomg the exact case you specified, rather than the same case as the previous version.
- Current: {**DSK**} now writes all new versions of a file in the same case as the old version.
- 11083** Recognizing a file on {**DSK**} with the **NEW** mode creates a directory.
- Prior: Some file recognition functions (e.g., **IL:FULLNAME**, **IL:INFILEP**, **IL:OUTFILEP**) tried to create a directory if the file to be recognized with the **NEW** mode specified a non-existent directory.
- Current: Recognizing a file does not create a directory.
- 11087** **CL:GET-SETF-METHOD** does not expand access-form and update-form. This makes it difficult to expand it in each access form or update form invocation.
- Prior: **CL:GET-SETF-METHOD** did not expand the access-form and update-form passed to the **SETF** invocation method.
- Current: **CL:GET-SETF-METHOD** expands the access-form and update-form gotten by the **SETF** invocation method.

**11103 DATABASEFNS reads databases with FILERDTBL.**

Prior: There was no consistency in using readtable between the creator (**DUMPDB**) and the loader (**LOADDB**). The creator used the invoked environment's readtable; the loader always used **FILERDTBL**. The creator and loader both used the invoked environment's package.

Current: The creator and loader for Masterscope databases always use the new Interlisp readtable as their readtable and the Interlisp package as their package.

**11112 Non-standard nicknames of the LISP package hurts!**

Prior: The **LISP** package had a nickname of **CL**.

Current: The new function **xcl:package-prefix** and its **setf** is now defined. You can change a package's prefix as follows:

```
(xcl:package-prefix package)           [Function]
```

```
(setf (xcl:package-prefix package)
new-prefix)                             [Function]
```

**package-prefix** returns the prefix name of the specified package. The package must be a symbol or string. For example:

```
(xcl:package-prefix 'lisp) returns #:cl
```

```
(xcl:package-prefix "il") returns il:il
```

You can use **setf** with this function. **new-prefix** must be a symbol or string. For example:

```
(setf (xcl:package-prefix 'lisp) ' #:lisp) changes
the Lisp package prefix name to ' #:lisp and returns #:lisp
```

The system's predefined packages have the following prefix names:

<u>Package</u>	<u>Nickname</u>
<b>LISP</b>	<b>CL</b>
<b>KEYWORD</b>	<b>KEYWORD</b>
<b>SYSTEM</b>	<b>SI</b>
<b>USER</b>	<b>USER</b>
<b>INTERLISP</b>	<b>IL</b>
<b>XCL</b>	<b>XCL</b>
<b>XCL-USER</b>	<b>XCL-USER</b>

**11127 HASDEF and EDITDEF do not edit the correct structure.**

Prior: The functions **IL:EDITDEF** and **IL:HASDEF** did not check correctly whether the argument is the proper name of a structure. For example:

If

```
(defstruct foo a b)
(defstruct foo-a a b)
```

then

(il:editdef 'foo-a 'il:structures) edits the structure definition for foo rather than foo-a

(il:hasdef 'foo-a 'il:structures) returns foo rather than foo-a.

Current: The functions IL:EDITDEF and IL:HASDEF correctly check whether the argument is the proper name of a structure.

**11145 The XCL compiler substitutes a variable reference past side-effecting form.**

Prior: The XCL compiler replaced a variable reference which is only used once in the LET form by its initialization form, regardless of whether that form has a side effect.

Current: The XCL compiler does not substitute a variable reference by its initialization form.

**11166 Although CLtL says CL:LOAD merges the specified pathname and CL:\*DEFAULT-PATHNAME-DEFAULTS\*, and loads it, CL:\*DEFAULT-PATHNAME-DEFAULTS\* actually has no effect on CL:LOAD.**

Prior: CL:LOAD did not use the CL:\*DEFAULT-PATHNAME-DEFAULTS\* to determine the name of the file to be loaded.

Current: CL:LOAD now merges the specified pathname and CL:\*DEFAULT-PATHNAME-DEFAULTS\*. The resulting pathname is used as the name of the file to be loaded.

**11172 You cannot type the circular expressions to the Exec.**

Prior: The Exec could not accept circular representations. For example, if you typed the circular expression as '#1=(#1# . #1#) in the exec, a Stack Overflow message would result.

Current: The Exec now accepts circular representations.

**11180 A variable declared by CL:DEFCONSTANT can be MAKUNBOUNDED, but it cannot be SETQed.**

Prior: The variable declared by CL:DEFCONSTANT could be MAKUNBOUNDED, but it would still exist as a constant, as a result of which it could not be SETQed.

Current: The variable declared by CL:DEFCONSTANT can be MAKUNBOUNDED, and then it becomes a simple global variable. The value of the DEFCONSTANTed variable can be changed by redefining the new value using DEFCONSTANT.

- 11184 CL:READ-FROM-STRING does not support the :preserving-whitespace option.**
- Prior: CL:READ-FROM-STRING did not support the :preserving-whitespace option.
- Current: CL:READ-FROM-STRING now supports the :preserving-whitespace option.
- 11186 Cannot load a plain-text file.**
- Prior: The loader could not reset the character set if it encountered an NS-encoding sequence in skipping separators. An error occurred because the consistency between the reading character set mode and the actual character set was lost.
- Current: The loader resets the character set in the above case. Consistency is maintained and the loading is accomplished correctly.
- 11189 The ByteCompiler does not support non-top level CL:LAMBDA forms.**
- Prior: The ByteCompiler did not correctly convert (CL:LAMBDA...) forms used within the body of a function definition. For example, the top level CL:LAMBDA might indicate the following CL:LAMBDA forms, and the ByteCompiler would convert and compile this form incorrectly:
- ```
(DEFINEQ (Funcs (CL:LAMBDA (x) ... )))
```
- Current: The ByteCompiler correctly converts and compiles (CL:LAMBDA...) forms used within the body of function definitions.
- 11192 TIME does not recognize keyword :datatypes correctly.**
- Prior: (time ?= printed out as (time form &key :repeat :output :datatypes). However, it should have printed out (time form &key :repeat :output :data-types).
- Current: (time ?= now prints out as (time form &key :repeat :output :data-types).
- 11252 MULTIPLE-VALUE-PROG1 was not treated as a Common Lisp special form.**
- Prior: (special-form-p 'multiple-value-prog1) returned NIL.
- Current: (special-form-p 'multiple-value-prog1) returns cl:|interpret-MULTIPLE-VALUE-PROG1|.