

## 3. GETTING STARTED

---

### Getting Ready

---

To prepare your system to run Medley, use the following checklist:

1. Exit `sunttools` or any other windowing system, unless you plan to run under X (in which case you can leave X running). Medley provides its own window system and must not run simultaneously with others.
2. Kill all your user processes (these have console as the control tty). Check to make sure you have killed any "selection\_svc" process. If you do not perform this step, certain error messages from UNIX (e.g., file system full) cause those processes to print to the console, resulting in scrolling of the display.
3. Check for the directory for the software, and add it to your path, if necessary:

```
prompt# set path = ($path /usr/share/lde/install.sunosx)
```

You can also add this to your `.login` file.

### Running Medley

---

#### Running Medley Directly

Invoke Medley by typing the name of the program, e.g.,

```
prompt% lde optional-sysout -k 'thishost-key' [-m memory-size]
```

If you are using either Xerox XNS or Xerox PUP Ethernet protocols, type instead

```
prompt% ldeether optional-sysout -k 'thishost-key' [-m memory-size]
```

This, in turn, runs `lde` and lets it use the Ethernet directly.

*optional-sysout* is the name of a Lisp virtual memory image file (see the section Where Medley Looks for Your Sysout below). *thishost-key* is the key you obtained from Venue for the machine on which you are running.

If the sysout was created on a machine with a different size display, the image will appear garbled for several seconds. After Lisp starts running, it readjusts the display to the current size.

The `-m` flag lets you control the maximum amount of memory Medley will use. *memory-size* is a number in the range 8 through 32, in megabytes. (See the detailed explanation on page 25.)

#### Using the Medley Shell Script

The script will try to find a key, an appropriate emulator, and a sysout. The script relies on information about where the Medley software was initially installed on your system. (The installation script `install-medley` automatically updates this information for you.) The `medley` script assumes that you have not changed the installation subdirectory structure from when it was originally installed.

The script will first try to find a key in the file *Installation directory/.medleyKey.hostname* or in *~/ .medleyKey.hostname*. *Installation directory* is where Medley was originally installed on your system. *hostname* is the name of the host for which the license key was issued. If neither file is found, you will be prompted for a key.

`medley [[emulator] sysout]` [Command]

*emulator*    Given a pathname or a simple name, the command will search for *emulator* as follows:

- If *emulator* is a relative or absolute pathname, e.g.,  
    `/share/medley/emulators/lde`, it will only try that pathname.
- If *emulator* is a plain file name, e.g., `lde`, the script uses the regular UNIX search path to find it. If it cannot find it, the script looks in the installation directory for Medley at your site, e.g.,  
    `/usr/share/lde/install.sunos4.1/lde`.
- If you omit *emulator*, the script uses `lde` as the default value, searching for it in the same fashion as above.

*sysout*        The command will search for *sysout* as follows:

- If *sysout* is a relative or absolute pathname, e.g.,  
    `../applications/my.sysout`, it will only try that pathname.
- If *emulator* is a plain file name, e.g., `my.sysout`, it will look for it in the following order:
  1. Current working directory `my.sysout`
  2. Installation directory for Medley at your site, e.g.,  
    `/usr/share/lde/lispsysouts/my.sysout`.
  3. Your home directory, `~/my.sysout`
  4. The `medley` subdirectory in your home directory,  
    `~/medley/my.sysout`.
- If you omit *sysout*, the script looks for it as explained in the Where Medley Looks for Your Sysout section below.

## Examples

- `prompt% medley`  
    To start Medley 2.0, a host access key is required.  
    Call Venue at (1-800-228-5325) for one,  
    and be prepared to give them your workstations host ID#  
    Your workstations host ID# is: 51006da3  
    Type in key or [^C] to abort: **8bf7723e 459aab34 73491feb**  
    Saving key '8bf7723e 459aab34 73491feb' into file  
    '`.medleyKey.hostname`' ...  
    Trying `/usr/share/lde/.medleyKey.hostname` ... Write  
    protected !

```
Trying home-directory/.medleyKey.hostname ... Done
Starting up Medley 2.0 ...
..... Medley 2.0 starts .....
```

If you had Medley installed in `/share/medley` on your system, it would try to run the emulator `/share/medley/install.sunos4.1/lde`, using the `sysout` `/share/medley/lispsysouts/LISP.SYSOUT`.

In this example you are prompted for a key, which is saved into the file: `home-directory/.medleyKey.hostname`

The script tried to save the key into the installation directory but did not have write access there. Instead it was put into your home directory (`~`) . `hostname` is the name of the host running medley.

The next time you use the script `medley`, you will not be prompted for the key.

- `prompt% medley application.sysout`

If you had `application.sysout` in your home directory, it would try running the emulator `/share/medley/install.sunos4.1/lde` using `~/application.sysout`.

## Where Medley Looks for Your Sysout

---

If you run Medley directly, the system searches the following places, in order, for the `sysout` to be used:

- *command line*

The name of the `sysout` file can be given on the command line when starting Medley; for example,

```
prompt% lde sysout -k 'thishost-key'
```

- `LDESRCEYSOUT`

If no `sysout` file name is given on the command line, the value of the environment variable `LDESRCEYSOUT` is used as the name of the `sysout` file. For example:

```
prompt% setenv LDESRCEYSOUT my.sysout
prompt% lde -k 'thishost-key'
```

would run the host key `my.sysout`.

- `~/lisp.virtualmem`

Finally, Medley looks for the file `lisp.virtualmem` on your home directory.

## Where Medley Looks for Your Site Initialization File

---

When Medley starts, it reads in a Lisp site initialization file. This site initialization file sets things like pathnames for fonts, site parameters, and the like.

Greeting and initialization are described in the *Interlisp-D Reference Manual*, Section 12.1.

Medley looks for a site initialization file in a number of locations:

- LDEINIT

If the environment variable `LDEINIT` is set to a complete Lisp file name, Lisp looks there first for the site initialization file:

```
prompt% setenv LDEINIT /usr/lisp/my-site-init.lisp
```

- /usr/share/lde/site-init.lisp

If `LDEINIT` is not set or there is no file with the name given, Lisp looks for a site initialization file called `/usr/share/lde/site-init.lisp`. The distribution tape contains a sample site initialization file in the Lisp library directory `/usr/share/lde/lisplibrary/site-init`. The system administrator should copy `site-init` into `/usr/share/lde/site-init.lisp` then customize it for the site. The comments in the sample `site-init.lisp` describe the parameters it sets and give guidelines for customizing it to your local conditions.

- {DSK}INIT.DFASL, {DSK}INIT.LCOM, {DSK}INIT.LISP

Finally, Lisp looks for a site initialization file on your Medley home directory (`{DSK}`). Chapter 5, *Medley File Systems*, describes the `{DSK}` device.

---

## Medley and X Windows

Medley 2.0-S supports the X Window System, Version 11 Release 4 (X11R4). Medley runs in a single X window; Medley's "screen" is displayed in that window, and you use Medley as usual.

### Starting X Windows

Start the X server on your console. Use the `xinit` command.

If necessary, start a window manager as a client of X (`xinit` often starts a window manager). The window manager provides many window management functions, such as moving, resizing and iconifying the window. Medley has no window management function of its own.

### Running Medley Remotely

You can run Medley on one machine, with the window on some other machine. To do so, perform these steps on the machine whose keyboard and display you will be using:

1. Add the host name to execute the Medley access control list:

```
xhost + hostname
```

2. Open a new xterm and rlogin to the Sun Workstation on which Medley is to run. Set the environment variable `DISPLAY` to the host name of the server machine:

```
setenv DISPLAY servername:0
```

3. Set the `LDEKBDTYPE` environment variable to tell Medley what kind of keyboard you will be using. Possible values are:

```
type3    Sun Type 3 keyboard
type4    Sun Type 4 keyboard
rs6000   IBM RS/6000 or PS/2
dec3100  DECstation 3100 or 5000
```

hp9000	HP9000 Series 700 or 800
X	Generic X terminal

If you don't set `LDEKBDTYPE`, it will default to `X`. The advantage of specifying a specific keyboard lies in how Medley treats the special function keys. The specific keyboard maps maximize the usefulness of keys marked, e.g., "Find". The generic keyboard code cannot do that reliably.

#### 4. Start up Medley.

A new window for Medley will appear on the X server's screen.

### The Medley Window

Normally, Medley uses the whole screen. Under X, Medley's "screen" appears in a single X window. Medley's screen is slightly smaller than the screen you are using to display it; if you make the X window full-screen-size, you see Medley's entire screen. If it is smaller, you will need to scroll to see parts of the screen.

The scroll bars (at the right and bottom of the X window) control what parts of Medley's screen appears in the window. Use the vertical scrollbar to scroll up and down, and the horizontal scrollbar to scroll left and right. The gravity buttons (at the lower right corner) set the bitgravity of the display window. Click the mouse button on one of these areas. The shade pattern is moved to the clicked area, and the bitgravity is set in the corresponding corner on the display window. The bitgravity determines how reshaping the X window affects what part of the Medley screen is visible.

---

## Environment Variables

Medley on the Sun uses several environment variables. They can be set from the shell with the `setenv` UNIX command. By convention, environment variable names use uppercase rather than lowercase letters, e.g., `LDEDESTSYSOUT`. The Medley environment variables are listed below, with a reference to sections in this *Guide* where further information can be found.

<code>LDEKBDTYPE</code>	See the Medley and X Windows or Sun Type 4 Keyboard sections in this chapter.
<code>LDEINIT</code>	See the Site Initialization File section in Chapter 4.
<code>LDESRCESYSOUT</code>	See the Where Medley Looks for Your Sysout section in this chapter.
<code>LDEDESTSYSOUT</code>	See the Saving Your State section in Chapter 4.
<code>LDESHELL</code>	See the UNIXCHAT section of the <i>Lisp Library Modules</i> .
<code>LDEFILETIMEOUT</code>	See the File System Errors subsection in Chapter 5.

---

## Keyboard Interpretation

This section describes how Medley interprets the Sun Type 3 and Type 4 keyboards. Except when running under X, Medley performs its own keyboard interpretation, taking raw up/down transitions directly from the keyboard. Medley uses its own key

numbering scheme; key numbers are used by Lisp functions such as `IL:KEYDOWNP` and `IL:KEYACTION`.

These key assignments were chosen to maximize compatibility with both the Xerox workstation keyboard and the normal Sun keyboards. You can attach a Sun Type 3 or Type 4 keyboard template, which also shows the Medley keyboard assignments, to your Sun Type 3 or Type 4 keyboard. Both templates are included with your Medley documentation set.

### Sun Type 3 Keyboard

Figure 3-1 shows the key number assignments for the Sun Type 3 keyboard. Figures 3 - 2 through 3 - 4 show Medley's key assignments for the Sun Type 3 keypads.



**Figure 3-1. Sun Type 3 Key Numbering**



**Figure 3-2. Sun Type 3  
Left Key Pad**



**Figure 3-3. Sun Type 3  
Right Key Pad**



**Figure 3-4. Sun Type 3 Center Key Pad Interpretation**

### Sun Type 4 Keyboard

Figure 3-5 illustrates the keyboard interpretation for the Sun Type 4 keyboard. Figures 3-6 through 3-8 show the keyboard and the left and right key pads for the Sun Type 4 keyboard.

NOTES: In SunOS 4.0, the NEXT (ALT/GRAPH) key on the Type 4 keyboard is inaccessible. Later versions of SunOS fix this.

Medley cannot detect whether it is running on a workstation with a Type 4 keyboard when running SunOS 4.0, 4.0.1, or 4.1. To make it work correctly on your workstation, enter the following before you start running Medley:

```
setenv LDEKBDTYPE type4
```



**Figure 3-5. Sun Type 4 Key Numbering**



**Figure 3-6. Sun Type 4  
Left Key Pad**



**Figure 3-7. Sun Type 4  
Right Key Pad**

Unknown IMAGEOBJ type  
GETFN: SKIO.GETFN.2

***Figure 3-8. Sun Type 4 Center Key Pad Interpretation***

[This page intentionally left blank]