Residential environment supports Incremental Development

Define, edit, debug in the live environment; write out to files to save for future session. System tracks what changed. Features work together.

DWIM - Do What I Mean, Spelling Correction

```
(DEFINEQ (FOO1 (X) (PLUSS X X]      <-- PLUS misspell.
Superparen
PP FOO1
(FOO1 6]
y
PP FOO1
```

History, UNDO <

```
??
UNDO <dwim event>
PP* FOO1
```

file manager - tracks what's been defined, changed, WHEREIS

```
FILES?)
y FILEB
(CLEANUP)
SEE FILEB
```

- Helpsys and DInfo interactive docs

```
MAN INFILE
MAN CL:WITH-OPEN-FILE
MAN CLHS.OPENER
<right click DInfo>
```

Masterscope (interactive cross reference)

```
LOAD(HELPSYS PROP)
. ANALYZE ON HELPSYS
. SHOW WHERE ANY CALLS CLHS.OPENER
USE EDIT FOR SHOW
. SHOW PATHS FROM HELPSYS
<load in fuller.database>
```

Common Lisp and Interlisp Integration

an Inter-medley of Uncommon Lisps

- freely intermix CL and IL function, macro, variable definitions

- All datatypes are common: lists, NIL, symbols, arrays, strings, structures, numbers

- many functions and special forms are the same: CAR, COND, ED

- Some are slightly different: CL:EVAL vs. IL:EVAL, CL:LAMBDA vs. IL:LAMBDA, ...

- Extend CL to include IL and IL to include CL

- Common Lisp definitions and declarations managed by Interlisp environment

Demo:

```
. SHOW WHERE ANY ON HELPSYS CALLS CL:WHEN
USE EDIT FOR SHOW
```

Structure editors

EDITMODE(TELETYPE) original structure editor: powerful, programmable, short commands (BO 3) P (SW 1 2). Useful for editing huge list structures or Lisp functions.

EDITMODE(DEDIT) (Load it to try). Like the TTY editor with menu and showing the results in a window.

EDITMODE(SEDIT:SEDIT) Default editor. SEdit maintains illusion that yo
u are editing structure. Parentheses are balanced at all times, but you can just type and backspace. Keyboard controls and attached menu.

Common to all: User never counts parentheses, modifies whitespace, or needs to fiddle with line breaks or indentation

The Virtual Machine and OS

MAIKO / MEDLEY

- D-machines had microcode to interpret a bytecoded insruction set

  Microcode ported to C ....

- The memory images are portable -only the emulator is machine/os specific

- **Medley is small** (relatively speaking). Bytecodes are compact

- Interlisp-D was the **whole operating system**:

  scheduler, window manager, network, drivers

- Now can rely on host OS for device drivers

Medley online uses 16MB/user (64MB max). Installed, 256mb max. Time to make new image 15 seconds; restart a saved image in a blink.

GIT: A Repository for Medley Definitions

Conventionally, GIT tracks and compares files

- Change detection and presentation based on line-editing semantics: Mismatching character sequence = significant difference

Medley's source files: external archives for structured definitions with metadata

- Saved and loaded, maybe printed, but never edited

- A given definition can be represented in different (but semantically equivalent) line and character sequences

Medley interface to GIT: Definition-based change tracking

P(ull)**R**(equest)**C**(ompare) command (demo)

- Retrieve changed files from GIT, find/parse PR vs master alternative definitions, compare as Lisp-structure differences

Eventually, load and manage definitions at commit-level granularity

- If function FOO is included in commits C and D, definitions FOO;C and FOO;D are co-resident: compare, edit, test

BACK TO POWERPOINT

Remnants (moved to PowerPoint)

```
OPEN-URL("https://www.youtube.com/watch?v=mI3Ga5LyIlI")
```

Revive LispUsers, Library, Internal

(Favorite LispUsers)

Some other development tools: SPY, File Browser

Revive Applications using Interlisp

Notecards - early HyperText

Rooms - Screen / Desktop management

LOOPS - Lisp Object Oriented Programming System (not CLOS)

Truckin’ - LOOPS game for teaching "Knowledge Representation"

LFG - Lexical Functional Grammar

.... and others

Modernizing Medley

Goals

- Adapt Medley UI to current hardware

- Make Medley work more like current applications for comfort

Mouse: Wheel Scroll, Window move on title, Window resize on corner. Warning: Exec windows. Consolation: you can do it all with menus. Reduction: some menus are invoked by "middle" mouse button which on some mice is hard to press.

Keyboard: Desired state: compatible keystrokes. Subtasks: decide what keystrokes do what; slash through umpteen layers of keyboard re-iterpretation.

Display: Color, high-resolution displays, modern fonts.

Reduce Barriers to Entry

- Running Online

```
OPEN-URL("https://www.youtube.com/watch?v=mI3Ga5LyIlI")

OPEN-URL("https://online.interlisp.org")
```

- Installation Instructions

```
OPEN-URL("https://interlisp.org/running/")
```

More on Running on Modern Systems

- about K&R C and C standards; type declarations, big-endian vs. little endian, 32 bit vs 64 eliminating errors, file-sytem changes, process handles, bugs left over from 24- to 28-bit address space

- docker, installers, CO/CI(?), two levels of virtualization

- online.interlisp.org: connect in seconds. Sessions

- Mysteries in the code we inherited, "software archeology"

- Robust "loadup process", all day vs seconds

- bytecode virtual machine aids portability.

```
fb {li}<maiko>bin/makefile-*.*-x
```

Linux, Mac, Windows (WSL1, WSL2, Cygwin)
Docker, FreeBSD, OpenBSD, SunOS5
x86_64, i386, arm7l, arm64, and older

Medley Interlisp

2023 Project Update

18 March 2023

Interlisp.org

What is Medley Interlisp?

- The software for the Xerox Lisp machines

developed from 1960's (BBN), thru 1970's and 80's (Xerox), and 90's (Envos, Venue)

- ACM Software System Award 1992

*The features of **structure editing**, **source code management**, **code analysis** and **cross-referencing** combined to support **rapid incremental development**. The 1992 ACM Software System Award was awarded to the Interlisp system for pioneering work in programming environments.*

- Then: expensive, slow, unwieldy... uvavailable

- Now: open source, 1000 times faster, on modern hardware of all sizes *enjoy*

What have we been doing?

+ Adapting the system to current environments and standards

hardware: keyboard, mouse, display, CPU (64bit, little endian)

software: standard C, Unicode, Posix, UI Guidelines

+ Lower barriers to entry

+ Demonstrate unique and original features

+ Help with revival of other applications built using system

+ Improve maintainability for future users

+ Gather and update documentation

Who might use Medley Interlisp?

- Retrocomputing enthusiasts

- Researchers, IDE tool creators, looking into ideas

- Software and AI historians (present and future)

- Software archivists (SPN, technical infrastructure)

- Developers of new applications

Who is involved?

+ Some original developers from Xerox PARC (from 70's and later)

+ Developers of s using Interlisp (recursive revivers)

- Open Source contributors

- Software History and Future Technology/ California State University Channel Islands

+ Friends, enthusiasts, and ... you?

```
(OPEN-URL "https://github.com/orgs/interlisp/people")
(MEDLEY-CONTRIB "medley")
(MEDLEY-CONTRIB "maiko")
(MEDLEY-CONRIB "online")
(MEDLEY-CONTRIB "Interlisp.github.io")
```

Medley Interlisp is an IDE

- Residential system for incremental development ... write, debug, edit in the live environment.

In Interlisp, you could refer to a function you hadn't written yet, run your code until it broke and from the break have it pop open an editor with the signature of your function. You could also inspect the values of the arguments passed. You could then write the function and continue the computation.

Similarly if your code broke with an error, you could edit the function on the stack to correct the error, and continue the computation. ...

When demonstrating this capability in Interlisp.the response was along the lines of "why would I ever want that?" ... If you've never used an environment focused on programmer productivity you have no idea how to even think along those lines. -- Simon Brooke

---

- Network emulation

Donate? See our GitHub sponsor page

Getting Involved

- Try things out, report new problems, or new reproducible cases

```
(OPEN-URL "https://github.com/interlisp/medley/issues")
```

```
(FILESLOAD PICK)
```

```
PICK ISSUE
```