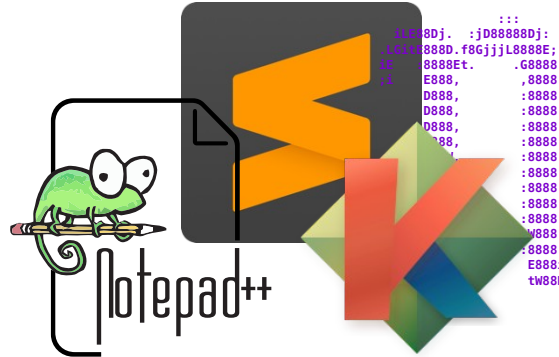
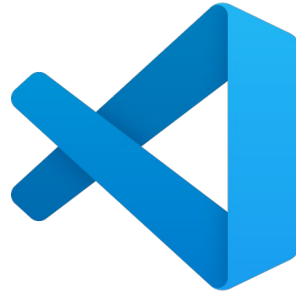




Emacs non è un editor di testo


E neanche un sistema operativo

Gli editor, quelli seri




Cosa la gente dice di



Quelli che non usano :

- Key bindings troppo lunghi
- Servono i pedali
- Feature creep
- Praticamente un sistema operativo
- GNU
- Brutto
- Lo usano le persone tipo Stallman

Quelli che usano :

“Posso fare tutto su Emacs!”

- Emulare Vim (evil)
- Leggere le email (Gnus)
- Andare su Internet (EWW)
- Chattare su Matrix (Ement.el)
- Torrentare roba (mentor)
- Guardare le mie lastre (dicom-mode)

“Emacs è il mio window manager (EXWM)”



???

Emacs 29.4 is out, download it [here](#)!



[About GNU](#) · [Philosophy](#) · [Licenses](#) · [Education](#) · [Software](#) · [Documentation](#) · [Help GNU](#) ·

[JOIN THE FSF](#)

[Home](#)

[Features](#)

[Releases](#)

[Download](#)

[Documentation & Support](#)

[Tour](#)

[Further information](#)



GNU Emacs

An extensible, customizable, **free/**
libre text editor — and more.

At its core is an interpreter for Emacs Lisp, a dialect of the Lisp programming language with extensions to support text editing.

Facciamo un salto indietro di 50 anni...



MIT AI Lab, 1974

TECO

Tape Editor and Corrector

successivamente

Text Editor and Corrector

*EBhello.c\$\$	Open file for read/write with backup
*P\$\$	Read in the first page
*SHello\$0TT\$\$	Search for "Hello" and print the line
printf("Hello world!\n");	The line
*-5DIGoodbye\$0TT\$\$	Delete five characters before pointer (ie "Hello"), insert "Goodbye", and print
printf("Goodbye world!\n");	The updated line
*EX\$\$	Copy the remainder of the file and exit

Da [Wikipedia](#)

Many people have said that TECO code looks a lot like line noise [...].

- *GNU Emacs FAQ*

Modifiche a TECO

- Control-R (*di Carl Mikkelsen*)
- 1974: **Richard M. Stallman (RMS)** migliora Control-R e aggiunge le macro riconfigurabili a runtime
- 1976: molti pacchetti di macro per TECO girano per il laboratorio
 - TECMAC
 - TMACS
 - ...
- **Guy L. Steele** decide che è tempo di unificare tutti i pacchetti in un'unica soluzione, “**?MACS**”
- RMS prende parte allo sforzo e diventa rapidamente il leader del progetto
- Fine 1976: il nuovo pacchetto di macro e patch a TECO diventa un'applicazione stand-alone, il cui nome viene cambiato in **EMACS**
 - Bacronym di “Editor MACroS”

EMACS is a real-time display editor which can be extended by the user while it is running.

Extensibility means that the user can add new editing commands or change old ones to fit his editing needs, while he is editing.

- “*EMACS: The Extensible, Customizable Display Editor*”, RMS 1981

LISP + EMACS

LISt Processor

Ideato da **John McCarthy** nel 1958, implementato da **Steve Russel**.

```
(defun heron (a b c)
  (let ((s (/ (+ a b c) 2)))
    (sqrt (* s
              (- s a) (- s b) (- s c))))))
```

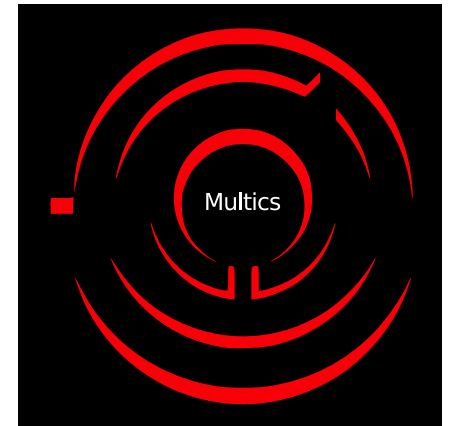
- Funzioni di ordine superiore
- Calcolo simbolico
- Codice e dati *omoiconici* tra loro
- Struttura dati di base (*cons cell*) universale
- Riprogrammabile dinamicamente
- **Molto popolare fra i ricercatori AI**

EMACS stood for Editing Macros, before we realized that
EMACS is composed of functions written in a
programming language rather than macros in the editor
TECO.

- *“EMACS: The Extensible, Customizable Display Editor”, RMS 1981*

La riscrittura di EMACS in Lisp

- Fine 1977: **EINE** (**Daniel L. Weinreb** e **Mike McMahon**), un EMACS implementato completamente in Lisp per le *Lisp machines*
 - ZWEI (1978)
 - Zmacs (1985)
- 1978: **Multics Emacs** (**Bernard S. Greenberg**), una riscrittura di EMACS in MACLISP
 - Supportato dall'avanzato compilatore MACLISP presente sulle macchine Multics di Honeywell



When large numbers of nontechnical workers are using a programmable editor, they will be tempted constantly to begin programming in the course of their day-to-day lives.

This should contribute greatly to computer literacy [...].

According to Bernard Greenberg, this is already happening with Multics EMACS.

- *“EMACS: The Extensible, Customizable Display Editor”, RMS 1981*

Verso GNU Emacs

Gosling Emacs (GOSMACS)

- Mentre era ricercatore alla Carnegie-Mellon University, **James Gosling** entra in contatto con Multics Emacs durante uno stage esterno
- Tornato alla CMU, si ritrova insoddisfatto a dover tornare a lavorare su **vi**
- 1981: **GOSMACS** vede la luce
 - Core scritto in C
 - Configurabile attraverso un falso Lisp, **Mocklisp**
 - Nato per girare su **UNIX**
 - Gratuito e con una “licenza” molto permissiva

GNU Emacs

- Dopo il collasso del MIT AI Lab e la comparsa di UNIX, volendo creare ciò che poi sarebbe diventato GNU, **RMS** decide di voler scrivere un nuovo Emacs:
 - sotto **licenza libera**
 - che andasse oltre le limitazioni di TECO
 - **basato su Lisp**
 - che girasse su UNIX
- RMS mette le mani su una copia di GOSMACS e comincia a modificarla pesantemente
 - Mocklisp viene sostituito da un nuovo dialetto di Lisp, conosciuto in seguito come **Emacs Lisp** (o **Elisp**)
 - Tutte le funzionalità non performance-critical vengono reimplementate in Elisp
- **20 marzo 1985: GNU Emacs 13** viene annunciato pubblicamente



Una storia parallela

Un futuro alternativo

At the time, you could make a computer that was about the same price range as other computers that weren't meant for Lisp, except that it would run Lisp much faster than they would [...].

We built the machine and had a Lisp operating system for it. It was written almost entirely in Lisp, the only exceptions being parts written in the microcode.

- *“My Lisp Experiences and the Development of GNU Emacs”*, RMS 2002

Lisp Machines (1973-1995)

Una classe di computer (progettati da Richard D. Greenblatt e Thomas Knight) CON processori specializzati nell'eseguire codice Lisp, essenziali per eseguire i complessi programmi AI al MIT.

Lisp Machine Manual

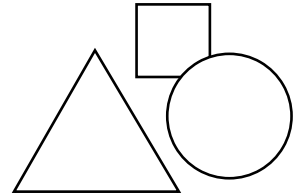
Second Preliminary Version

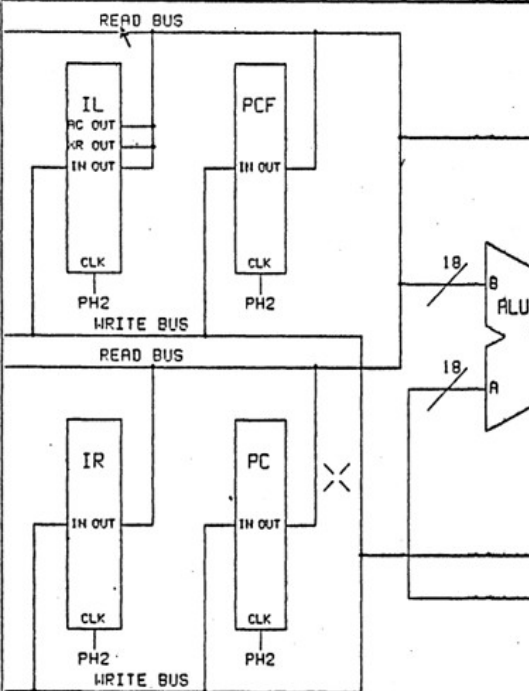
January 1979

Daniel Weinreb
David Moon

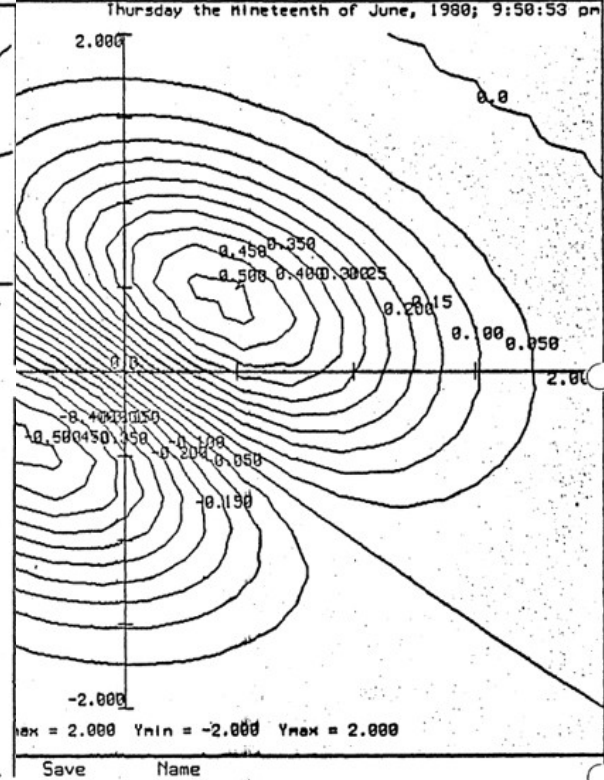
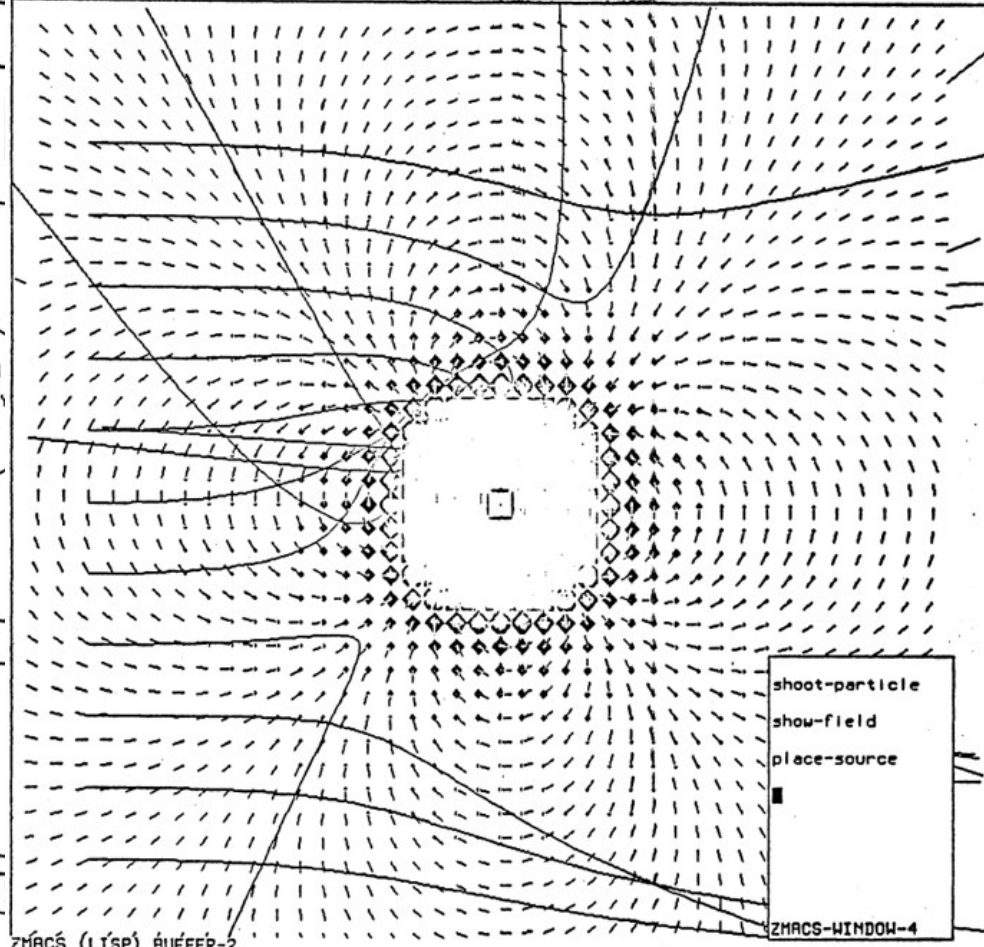


symbolics inc.





ALU Functions
 C: (0,1,ACC.Carry)
 A+C, B+C, B sh left, B sh right
 A*B+C, A-B-1+C, -B-1+C
 A xor B, A and B, A lor B ;?



shoot-particle
 show-field
 place-source
 Save Name
 DONE

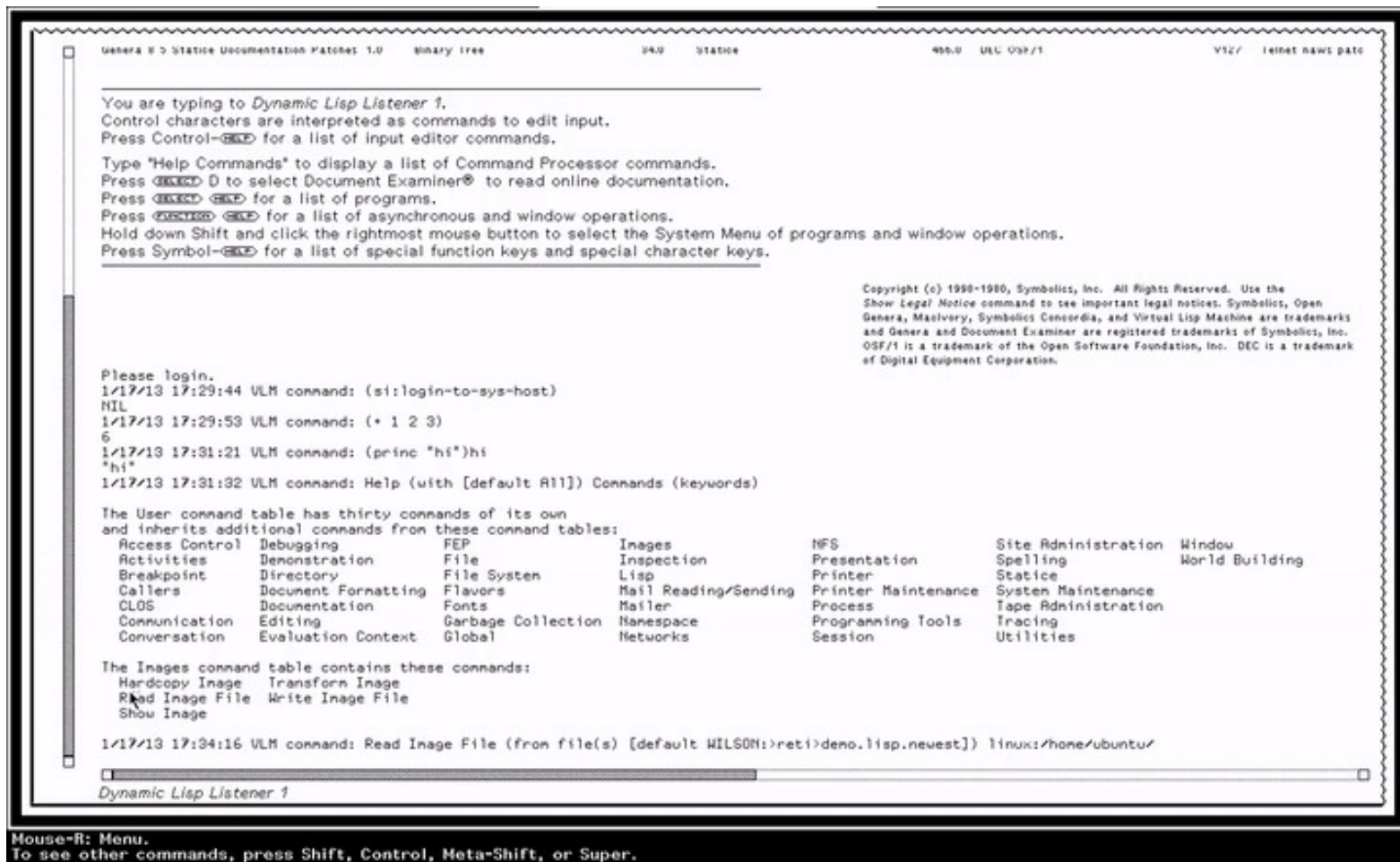
Body mode, Scale=1, (129,0). BLOCK-SCHEMATIC view of DATA PATHS? ac
 Edit Definition Edit Body List properties

Mouse: Move body Cursor to no
 Loading file AI: MOONDR; TEN DRAW into package DRAW
 Body mode.

```

done
(scan '(-15.0 10.0) '(-15.0 -10.0) '(0.2 0) tes2)
done
(scan '(-10.0 15.0) '(-10.0 15.0) '(0.1 0) tes2)
done
(scan '(-10.0 15.0) '(-10.0 -15.0) '(0.1 0) tes2)
done
    
```

Demo su YouTube



At first, I thought of making a Lisp-based system, but I realized that wouldn't be a good idea technically.

To have something like the Lisp machine system, you needed special purpose microcode. [...] Without that, you would be reduced to something like the Lisp compilers for other machines. The programs would be faster, but unstable. [...]

[S]o I rejected the idea of making a system like the Lisp machine.

- *“My Lisp Experiences and the Development of GNU Emacs”, RMS 2002*

GNU Emacs: la Lisp shell di GNU

I decided instead to make a Unix-like operating system that would have Lisp implementations to run as user programs. The kernel wouldn't be written in Lisp, but we'd have Lisp.

So the development of that operating system, the GNU operating system, is what led me to write the GNU Emacs.

- *"My Lisp Experiences and the Development of GNU Emacs", RMS 2002*

GNU Emacs non è un semplice editor di testo

GNU Emacs è esattamente complementare a quello che dovrebbe essere un sistema operativo

GNU Emacs è la shell grafica programmabile del sistema operativo GNU

- Interattiva
- Infinitamente malleabile grazie ad Elisp
- Fortemente integrata col resto del sistema GNU

Forse abbiamo tempo per una demo

O forse no

Per domande, discussioni, o per avere
una copia delle slide animate:
tomasantonio.lopez@mail.polimi.it