

# On The Evolution of User Interfaces

Martin Dostál

Honeywell International

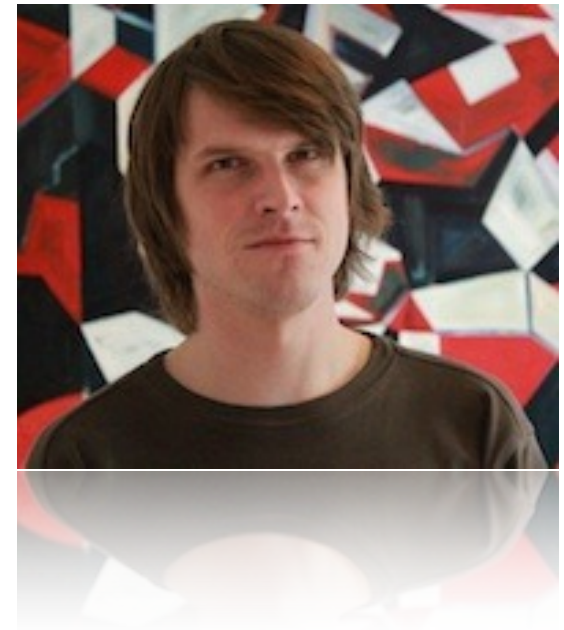
Aerospace Advanced Technology Europe

Human Centered Systems

November 2012

**Honeywell**

- Martin Dostál
- Ph.D. in computer science
- Human Factors Scientist at Honeywell
  - UI/HMI validations and studies, statistics and data analysis, eyetracking, UI/HMI design and meta-design, mentoring
- 10+ years in academia (UPOL); human-computer interaction, software development, machine learning and artificial intelligence



- history and evolution as a background
- demonstration of the most influential systems
- identify and explore principles behind modern graphical user interfaces
- what constitute graphical user interfaces these days
- quick look at some great ideas not present in GUIs these days
- we are faced with operational blindness



***WHAT IS THE ESSENCE OF GRAPHICAL  
USER INTERFACES?***

# Before the rise of personal computers

---

- direct control
- metaphors versus technical implementation
- no interactivity
- character I/O, punch cards...
- system-centered design

# Memex

- Memex (Memory+Index)
- Vannevar Bush
- „As We May Think”, 1945
- a concept „only”
- a desk with peripherals
- data oriented
- hypertext, searching, data association
- image processing



source: <http://maidig.wordpress.com>



# Sketchpad

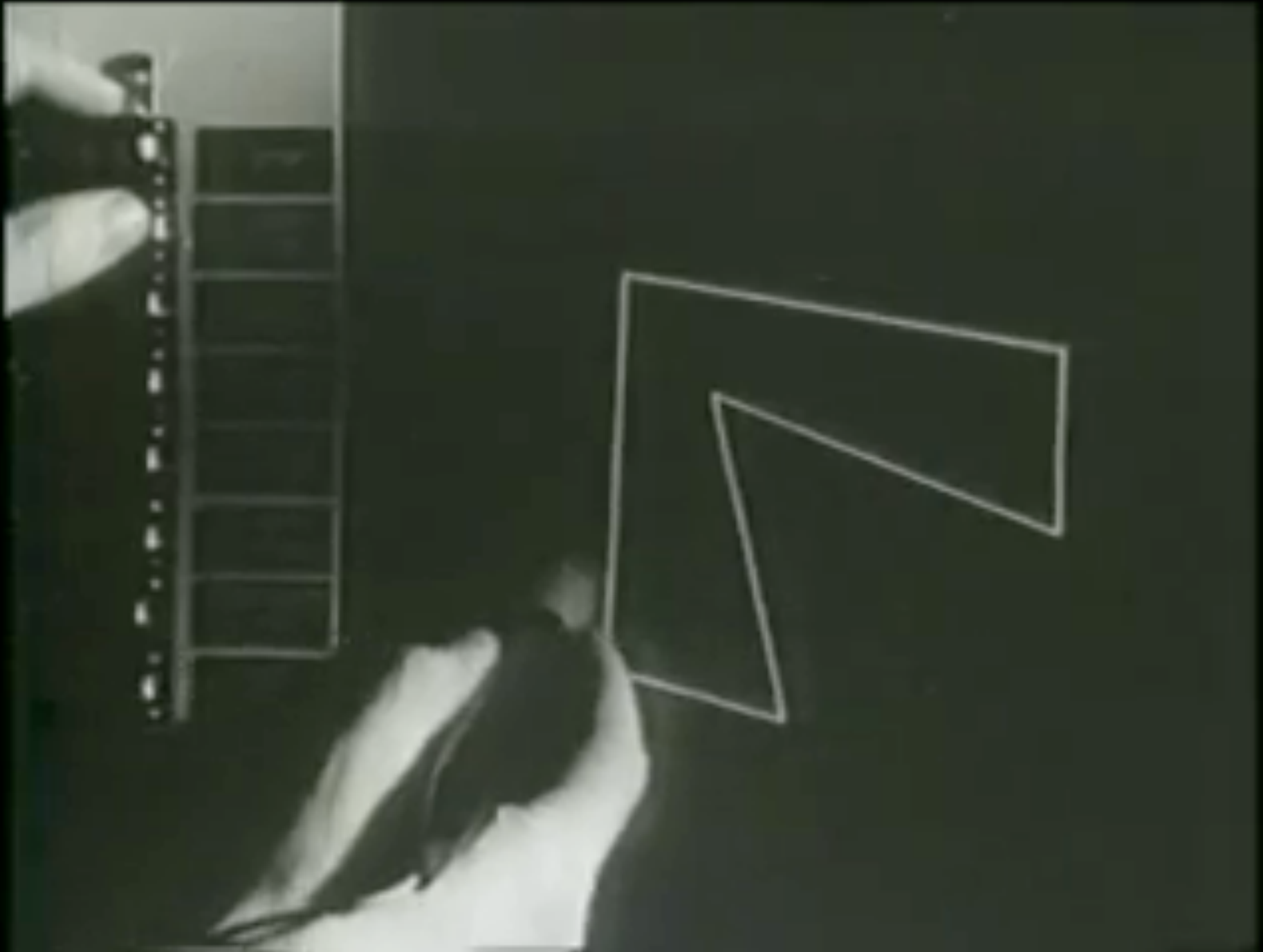
- 1963, Ivan Sutherland
- first interactive system
- first object oriented system
- first graphics oriented system
- CAD (Computer Aided Design) archetype



- canvas as a paper metaphor
- elements as prototype objects
- system controlled by
  - buttons (actions)
  - light pen (objects)
- intelligent interpretation of user input
- vector based
- object manipulation

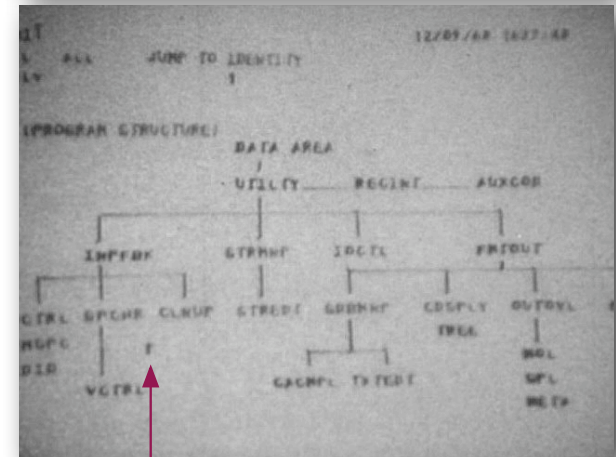
Video: [www.youtube.com](http://www.youtube.com)

Video: [www.youtube.com](http://www.youtube.com)



# **On-Line System (NLS)**

- Douglas Engelbart, 1968
- human augmentation
- text oriented
- canvas in a full screen window
- one window at time
- vector oriented
- hypertext
- groupware
- three button mouse
- cursor
- chord keyboard







**XEROX**

- Xerox Parc, 1978
- inspired by NLS, but many new ideas
- first bitmap oriented system (606x808)
- Ethernet



- office system
- task oriented
- could be considered as first system with GUI, but without consistent user interface
- multi window system
- both graphical and textual interfaces
- WYSIWYG output
- modal control

Start

Quit  
Clear  
Type

**Ready:**  
 Select file names with the mouse  
 Red-Copy, Yel-Copy/Rename, Blue-Delete  
 Click 'Start' to execute file name commands

Pages: 832 Log

Files listed: 60  
 Files selected: 0 Delete: 0  
 Copy/Rename: 0 Copy: 0

DP0: <SysDir.> \*.\*

```

      ~- BEGINNING ~-
      1012-AstroRoids.Boot.
      Anonymous.l.
      BattleShip.er.
      BattleShip.RUN.
      BlackJack.RUN.
      BuildKal.cm.
      CalcSources.dm.
      Calculator.RUN.
      Chess.log.
      Chess.run.
      Com.Cm.
      CompileKal.cm.
      CRTTEST.RUN.
      DMT.boot.
      EdsBuild.run.
      empres.run.
      Executive.Run.
      Fly.run.
      galaxian.boot.
      Garbage.$
      Go9.run.
      GoFont.AL.
      Invaders.Run.
      junk.
      junk.press.
      Kal.bcpl.
      Kal.cm.
      KalA.asm.
      KalMc.mu.
      Kinetic4.RUN.
      LoadKal.cm.
      MasterMind.RUN.
      maze.run.
      Mesa.Typescript.
      Missile.run.
      NEPTUNE.RUN.
      othello.run.
      Pinball-easy.run.
      POLYGONS.RUN.
    
```

Pages: 0 Log

Files listed: 0  
 Files selected: 0 Delete: 0  
 Copy/Rename: 0 Copy: 0

No Disk: <SysDir.> \*.\*

READY: Select operand or type command  
 Last command was LOOK  
 {A\_substa...!\_way1} {Computer... \XEROX1}\$

## Personal Distributed Computing The Alto and Ethernet Software

*Butler W. Lampson*  
 Digital Equipment Corp. Systems Research Center

### Abstract

The personal distributed computing system based on the Alto and the Ethernet was a major effort to make computers help people to think and communicate. A complex and diverse collection of software was built to pursue this goal, ranging from operating systems, programming environments, and communications software to printing and file servers, user interfaces, and applications such as editors, illustrators, and mail systems.

### 1. Introduction

A substantial computing system based on the Alto [Thacker et al.

Computer Science Laboratory  
 Xerox Palo Alto Research Center  
 3333 Coyote Hill Road  
 Palo Alto, California 94304

## XEROX

Glen J. Culler  
 608 Litchfield Lane  
 Santa Barbara, CA 93109

Dear Glen:

This is a follow-up to earlier correspondence you received from Alan Perlis regarding the ACM Conference on the History of Personal Workstations. As you know, the conference is scheduled for January

The screenshot shows the Xerox Alto's graphical user interface. At the top is the 'System Browser' window, which displays a hierarchical list of classes and objects. Below it is a window titled 'Fig. 1' showing a detailed 3D rendering of a bolt. To the right of the bolt is a code editor window with the following content:

```

collect: aBlock
  "Evaluate aBlock with each of my elements as the argument. C
  resulting values into a collection that is like me. Answer with
  collection. Override superclass in order to use add, not at put.

  | newCollection |
  newCollection + self species new.
  self do: [each | newCollection add: (aBlock value: each)].
  ^newCollection

User Interrupt
Paragraph>>characterBlockAtPoint:
Paragraph>>mouseSelectto:
CodeController(ParagraphEditor)>>processRedButton
CodeController(ParagraphEditor)>>processMouseButtons
CodeController(ParagraphEditor)>>controlActivity
CodeController(Controller)>>controlLoop

controlActivity
  self scrollBarContainsCursor
  ifTrue:
    [self scroll]
  ifFalse:
    [self processKeyba
    self processMouse]

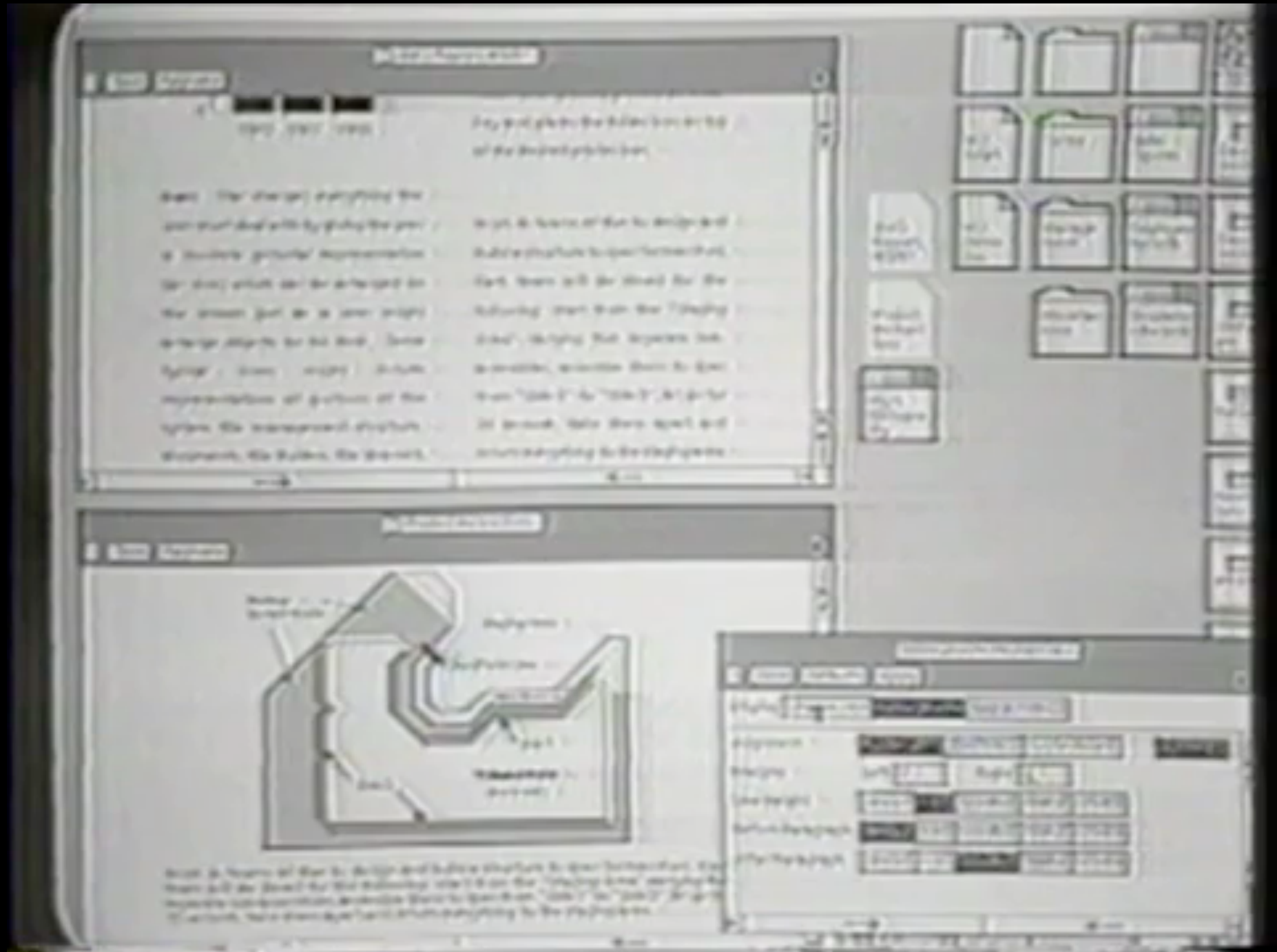
blueButt
31@507 corner:
60@770
Rectangle fromUser origin
ScreenForm setFullPageWidth.

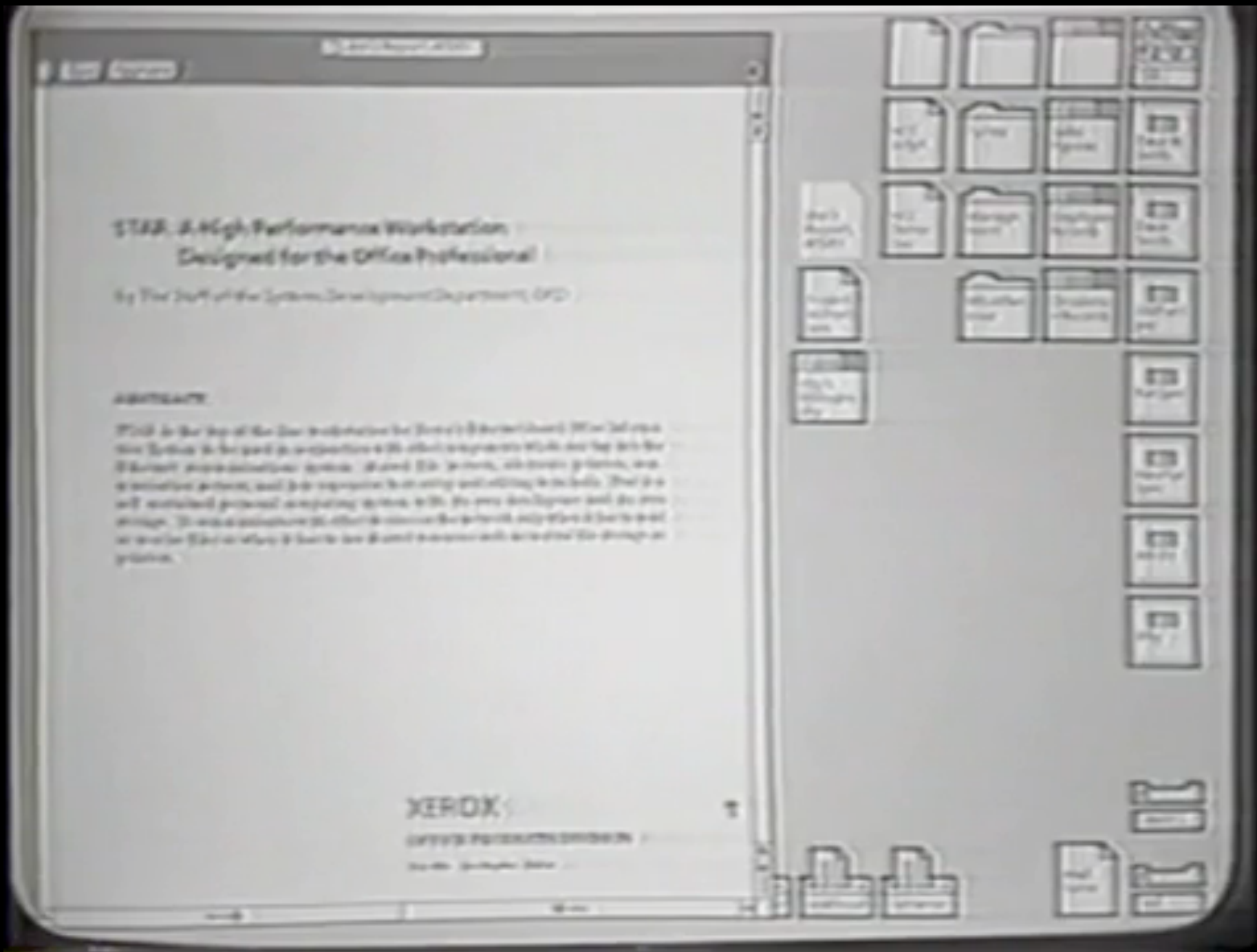
(Form readFrom: 'file:skate.form') edit
  
```

- 1981, 17.000 \$, dedicated for office usage
- highly innovative in terms of user interfaces
- document-centric system
- desktop metaphor
- object oriented at user level
- consistent graphical user interface
- controlled by keyboard and two button mouse

***APP-CENTRIC  
VS.  
DOCUMENT CENTRIC  
APPROACH***

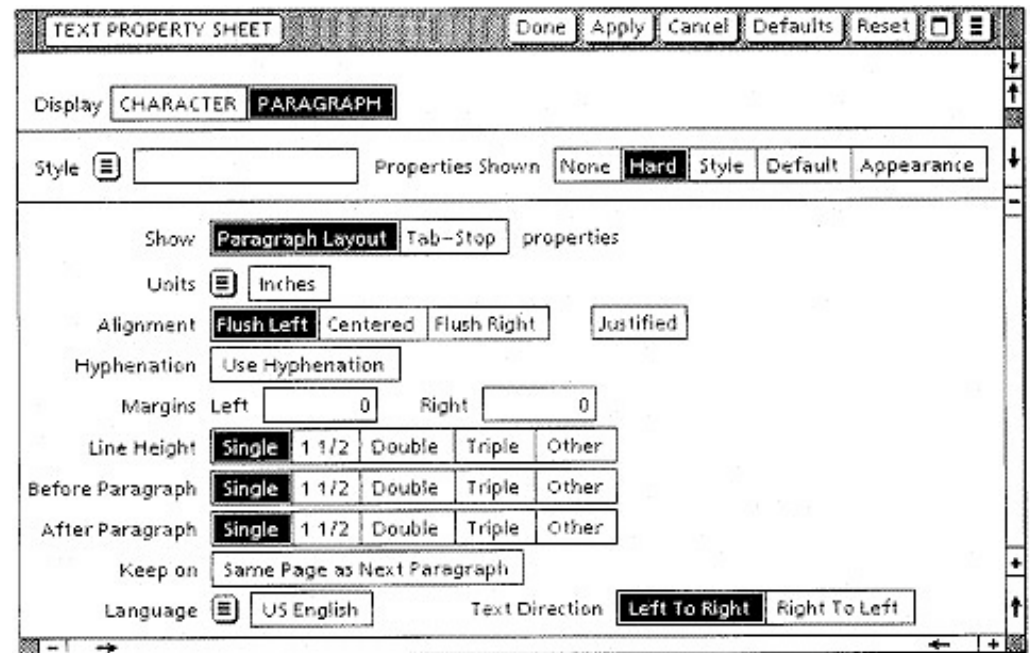
Video: [www.youtube.com](http://www.youtube.com)





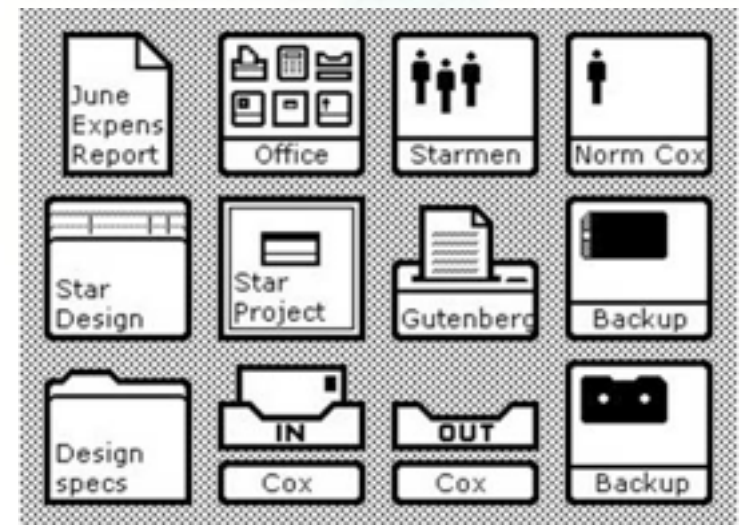
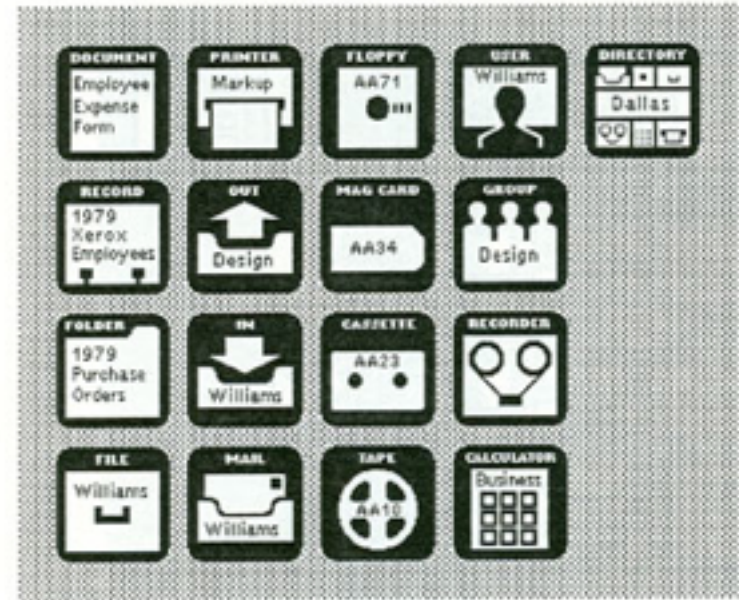


- physical keyboard - object manipulation
- virtual keyboard
- drag-and-drop archetype
- minimalist approach
- split windows
- progressive disclosure
- modal cursor
- object oriented, object-action paradigm
- property sheets (overlapping)
- pull-down menu archetype



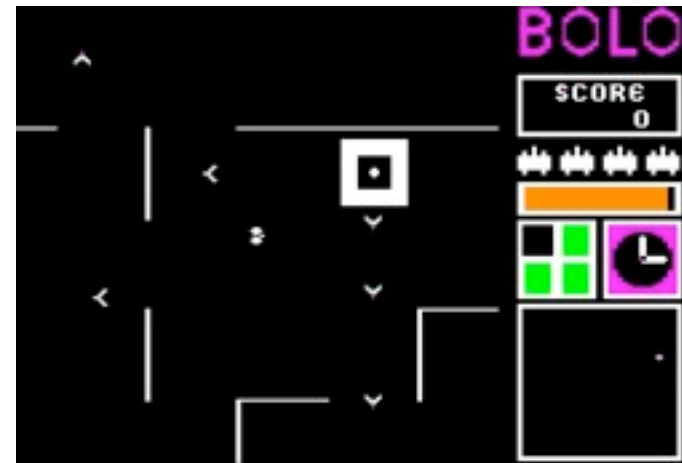
- first application of user-centered design
  - „design then code”
  - prototyping
  - task models and analysis
  - iterative approach
- user studies
- user interface design with strong visual considerations

Figure 2.  
Set 2 (Bowman)



**Apple**

- Apple II (1977)
- personal computer
- no GUI in modern sense
- games, simple office software

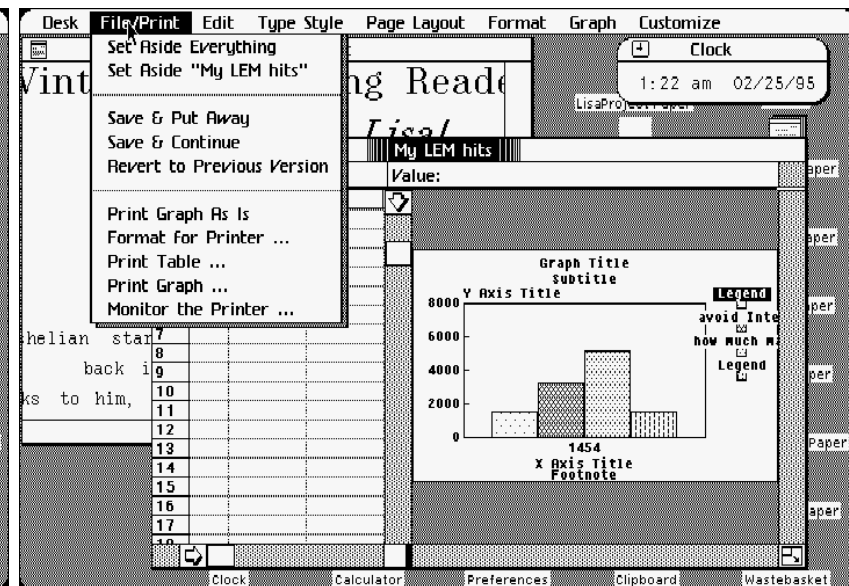
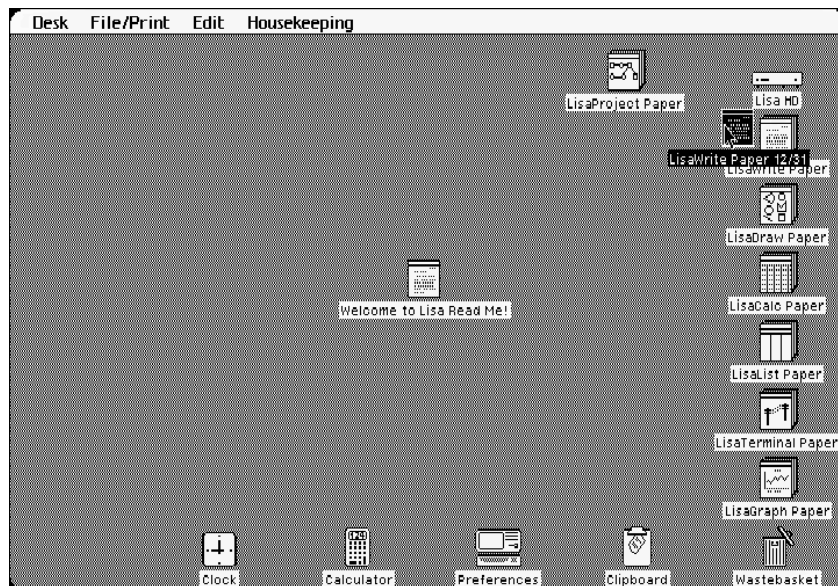


ITEM	NO	UNIT	COST
MUCK RAKE	43	130	556.00
BUZZ CUT	1	101	101.00
TONER	250	400	12480.00
SHUFF	2	9	18.00
SUBTOTAL			13155.68
9.75% TAX			1282.66
TOTAL			14438.16

- developed from 1979
- on market 1983, 10.000 \$
- Lisa Office System
- system opened for third party applications, but ...
- strongly influenced by Xerox
- commercial disaster

# Apple Lisa – innovations

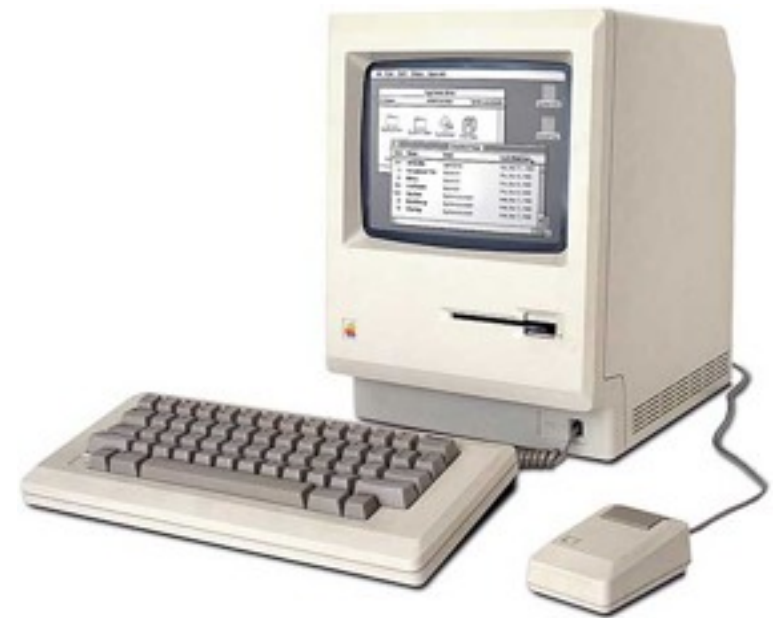
- freeform windows
- pull-down menu, disabling menu items
- drag-and-drop
- trash
- one button mouse
- double-click :(



- document centric
- applications at sideline (no „Quit“, but “Set Aside” instead, no “New” – Stationery Pad)
- direct manipulation
- modality minimization
- OS X dashboard



- Macintosh, 1984, 2.500 \$
- personal computer
- commercial success
- UI based on Lisa
- technical implementation completely different from Lisa
- no multitasking until 1987
- Macintosh Human Interface Guidelines - first HIG (1992)





[This document was written before the Macintosh project was operating under that name, and was still called "Annie". This note was written as an observer at that time not directly involved in the project. (Comments in brackets have been added on Oct. 11 79)]

This is an outline for a computer designed for the Person In The Street (or, to abbreviate: the PITS); one that will be truly pleasant to use, that will require the user to do nothing that will threaten his or her perverse delight in being able to say: "I don't know the first thing about computers," and one which will be profitable to sell, service and provide software for.

You might think that any number of computers have been designed with these criteria in mind, but not so. Any system which requires a user to ever see the interior, for any reason, does not meet these specifications. There must not be additional ROMS, RAMS, boards or accessories except those that can be understood by the PITS as a separate appliance. For example, an auxiliary printer can be sold, but a parallel interface cannot. As a rule of thumb, if an item does not stand on a table by itself, and if it does not have its own case, or if it does not look like a complete consumer item in [and] of itself, then it is taboo.

If the computer must be opened for any reason other than repair (for which our prospective user must be assumed incompetent) even at the dealer's, then it does not meet our requirements.

Seeing the guts is taboo. Things in sockets is taboo (unless to make servicing cheaper without imposing too large an initial cost). Billions of keys on the keyboard is taboo. Computerese is taboo. Large manuals, or many of them (large manuals are a sure sign of bad design) is taboo. Self- instructional programs are NOT taboo.

There must not be a plethora of configurations. It is better to offer a variety of case colors than to have variable amounts of memory. It is better to manufacture versions in Early American, Contemporary, and Louis XIV than to have any external wires beyond a power cord.

And you get ten points if you can eliminate the power cord.

Any differences between models that do not have to be documented in a user's manual are OK. Any other differences are not.

It is most important that a given piece of software will run on any and every computer built to this specification. There must be no differences between machines whether in terms of I/O, speed, memory size, configuration, or possible accessories.

source: Jef Raskin, "Design Considerations for an Anthropophilic Computer" (28-29 May 1979)-- in "The Macintosh Project: Selected Papers from Jef Raskin (First Macintosh Designer), Circa 1979," document 4, version 1.  
Location: M1007, Apple Computer Inc. Papers, Series 3, Box 10, Folder 1

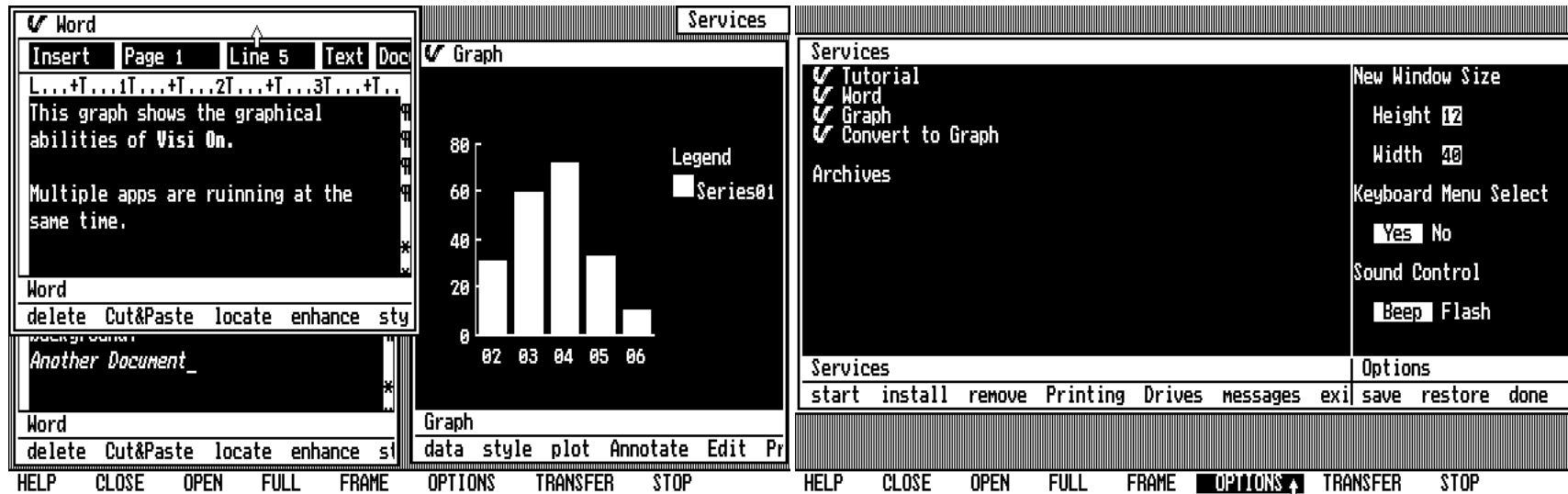
**IBM PC**

- cheap and open platform
- late rise
- bottlenecks
- textual interfaces, MS-DOS



- VisiCorp, 1983
- first GUI environemt for PC
- MS DOS app
- multitasking
- CGA graphics (640x200)
- resources demanding
- office apps (VisiCalc – first spreadsheet)

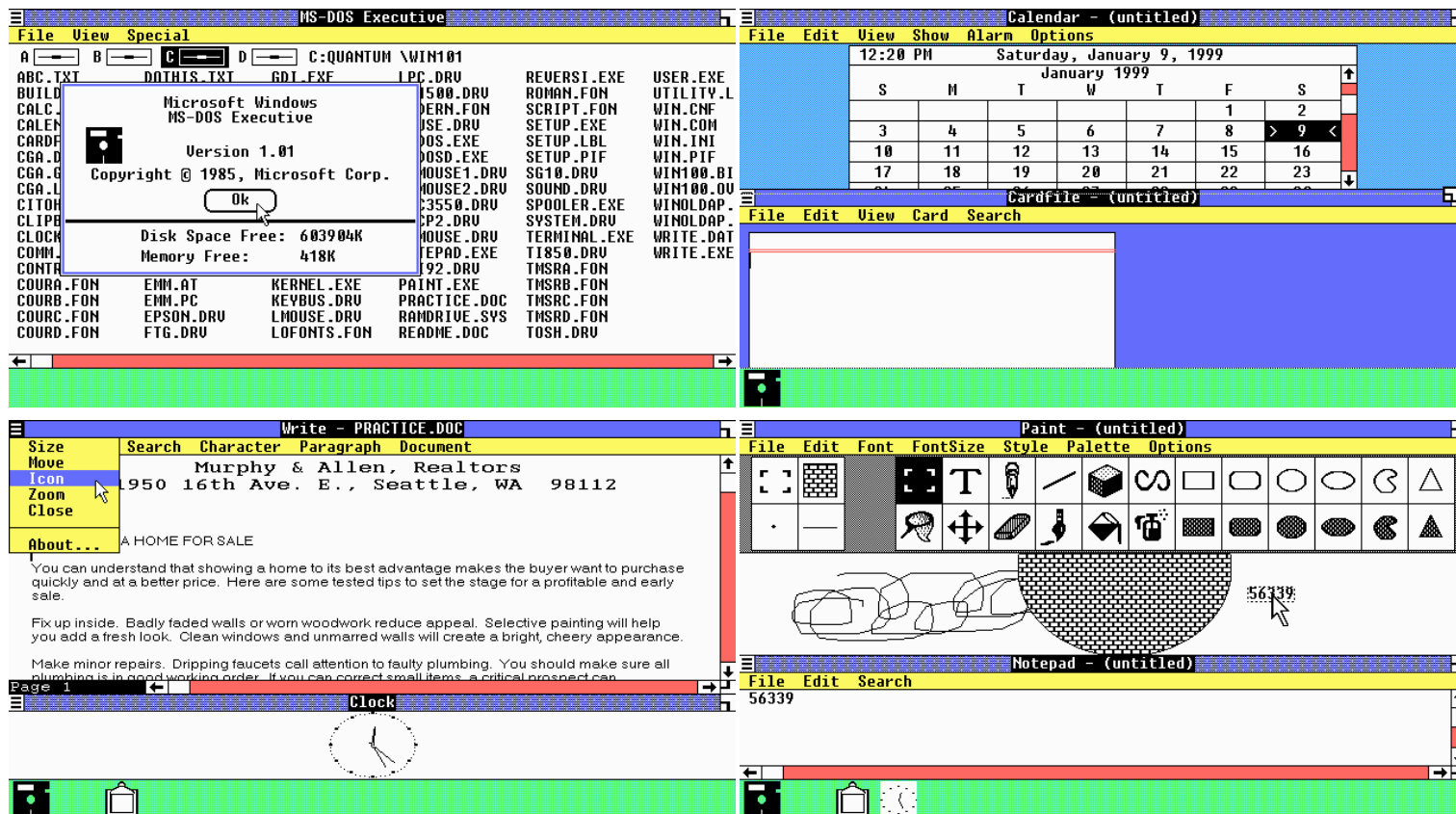
- consistent GUI
- widnows
- more text at the expense of graphics



- developed since 1981
- announced 1983
- introduced 1985, 99 \$
- consistent GUI
- multitasking
- file oriented vs object oriented
- application centric approach

# Windows 1.01

- strong text orientation
- split windows
- no desktop (until Win95)
- no standard dialogs...



- Windows 2.0
  - overlapping windows
- Windows 3.1 (3.0., 3.1. NT)
  - system dialogs
- Windows 95
  - files as icons, still not objects
  - long file names
  - desktop
- Windows Vista
  - Ribbon Interface (Office 2007)
- Windows 8
  - Metro, Tiles



- Mac OS (1984-2001)
  - System 7 (1991) colorized UI
- Mac OS X (since 2001)
  - completely new OS
  - NextStep
  - similar, but not the same interface



**Lessons learned?**

**martindostal@me.com**

**http://dostal.wserver.cz**