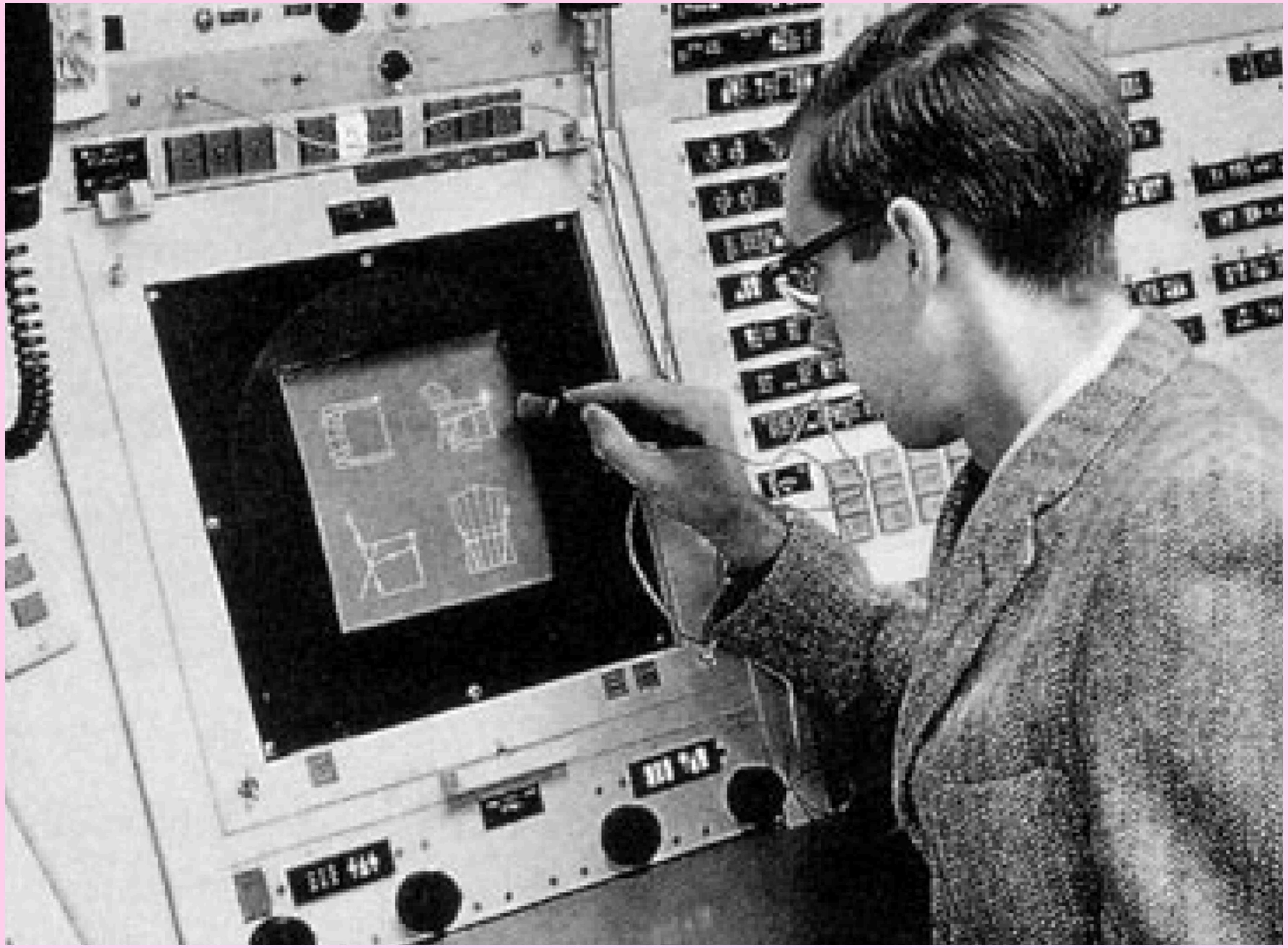
Interfaces from the 1960's

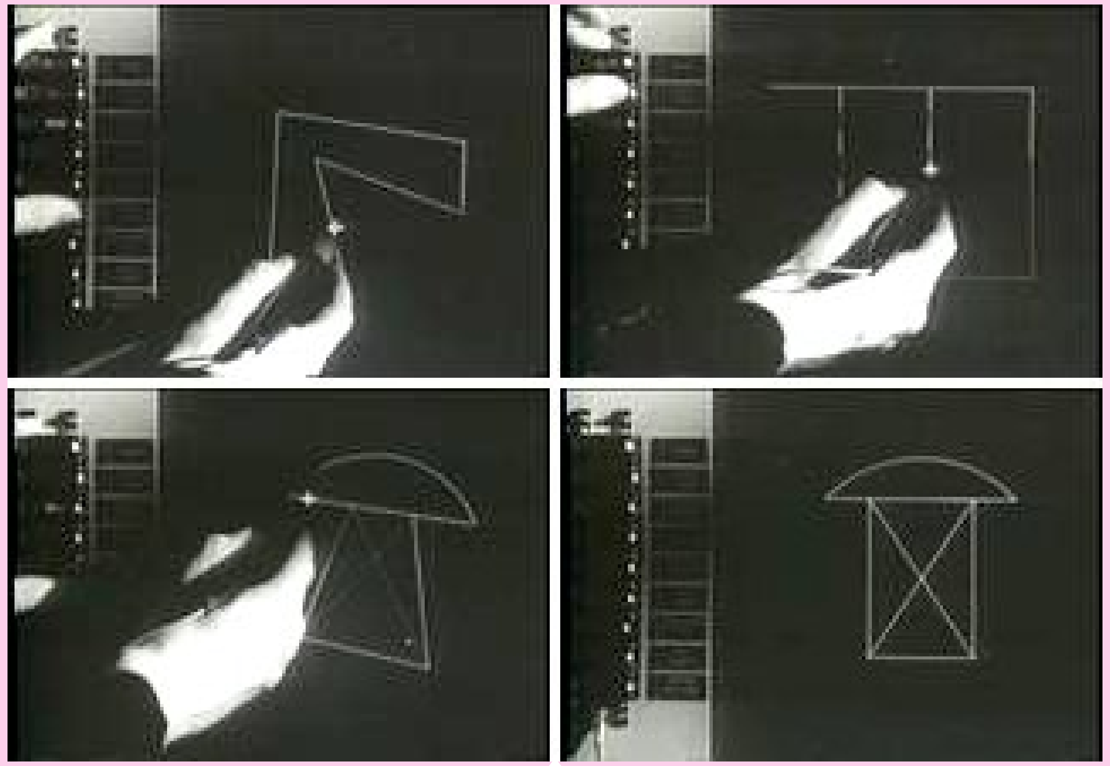
Sketchpad (1963)

Sketchpad, developed by **Ivan Sutherland**, was one of the first graphical user interfaces (GUIs) that **allowed users to interact** with the computer using a **light pen** to draw directly on a screen. This was revolutionary for the time, as it moved away from **text-based command interfaces.** Its key features were**: Constraint solivng Graphical Interaction, and Hierarchical Structure.**



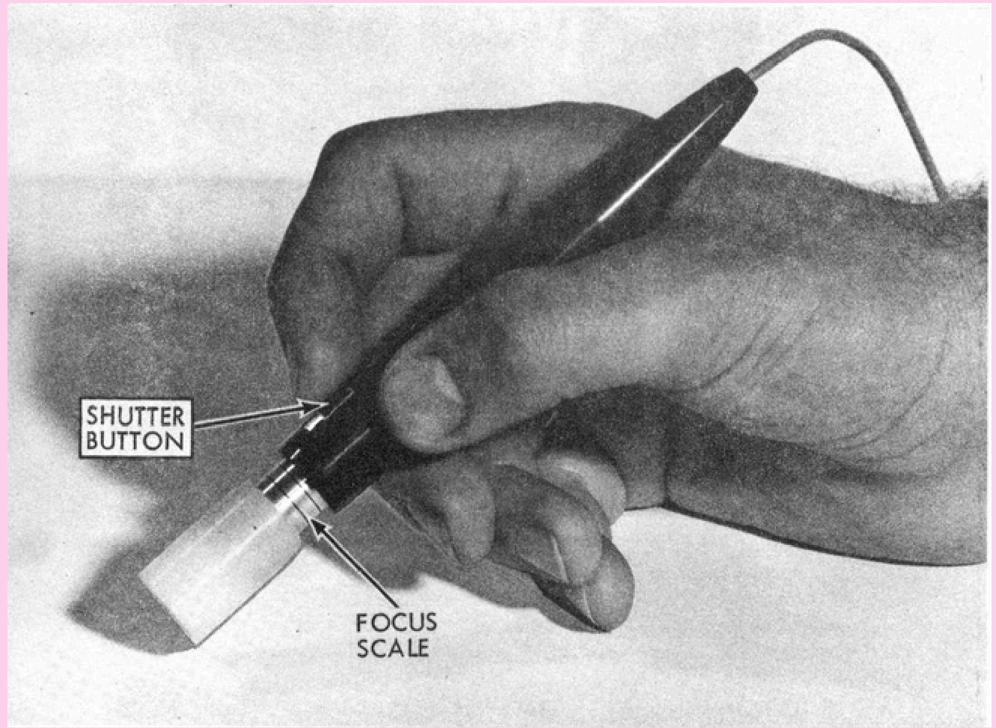
Hardware Components

Sketchpad ran on the **TX-2 computer** at MIT’s Lincoln Laboratory, a powerful machine for its time, equipped with a CRT (cathode-ray tube) **display** and a **light pen**. The light pen was used to interact with the graphical objects on the screen.



Display uses surface-barrier transistors in digital circuits

Naina<3



The light pen is used both to position parts of the drawing on the display and to point to them to change them.

Naina<3

SOFTWARE AND UI

Sketchpad's software architecture was highly innovative, **involving real-time processing** of graphical input, a **constraint solver** to maintain geometric relationships, and a **hierarchical data structure** for managing objects and instances.

Users could **draw basic shapes on the screen**, apply constraints, and **manipulate** these shapes in **real-time**. The interface was designed to be **intuitive**, allowing users to interact with the computer **graphically** rather than through text-based commands.

OBSERVATIONS BASED ON TEXT

Design Systems (Atomic Design)

Atomic Design talks about how, in chemistry, atomic elements combine to form molecules, which then form more complex organisms. This hierarchical structure offers a powerful analogy for interface design. In this case sketchpad allows for the **creation** and **reuse of graphical elements**, much like how Atomic Design emphasizes the **importance of modular components** to build complex interfaces. It breaks down the interface into simple elements like **points and lines** (atoms), which can be combined to **form shapes** (molecules) and more **complex structures** (organisms). These components can be arranged into **layouts** (templates) to create **complete drawings** (pages).

OBSERVATIONS BASED ON TEXT

Concepts of Interface (D5R1)

In "Interface," Reinfurt explores the idea that an interface is not just a surface for interaction but an essential **mediator** between the **human user and the machine**. Interface is not simply a tool for input and output, but it actively shapes the way humans **engage** with technology. Sketchpad allowed users to **draw directly on a screen** using a **light pen**, providing an intuitive, visual way to interact with a computer. It turned the computer into a mediator that translated user inputs into precise designs, introducing innovations like **object-oriented principles** and **constraints** that **enhanced design capabilities.**

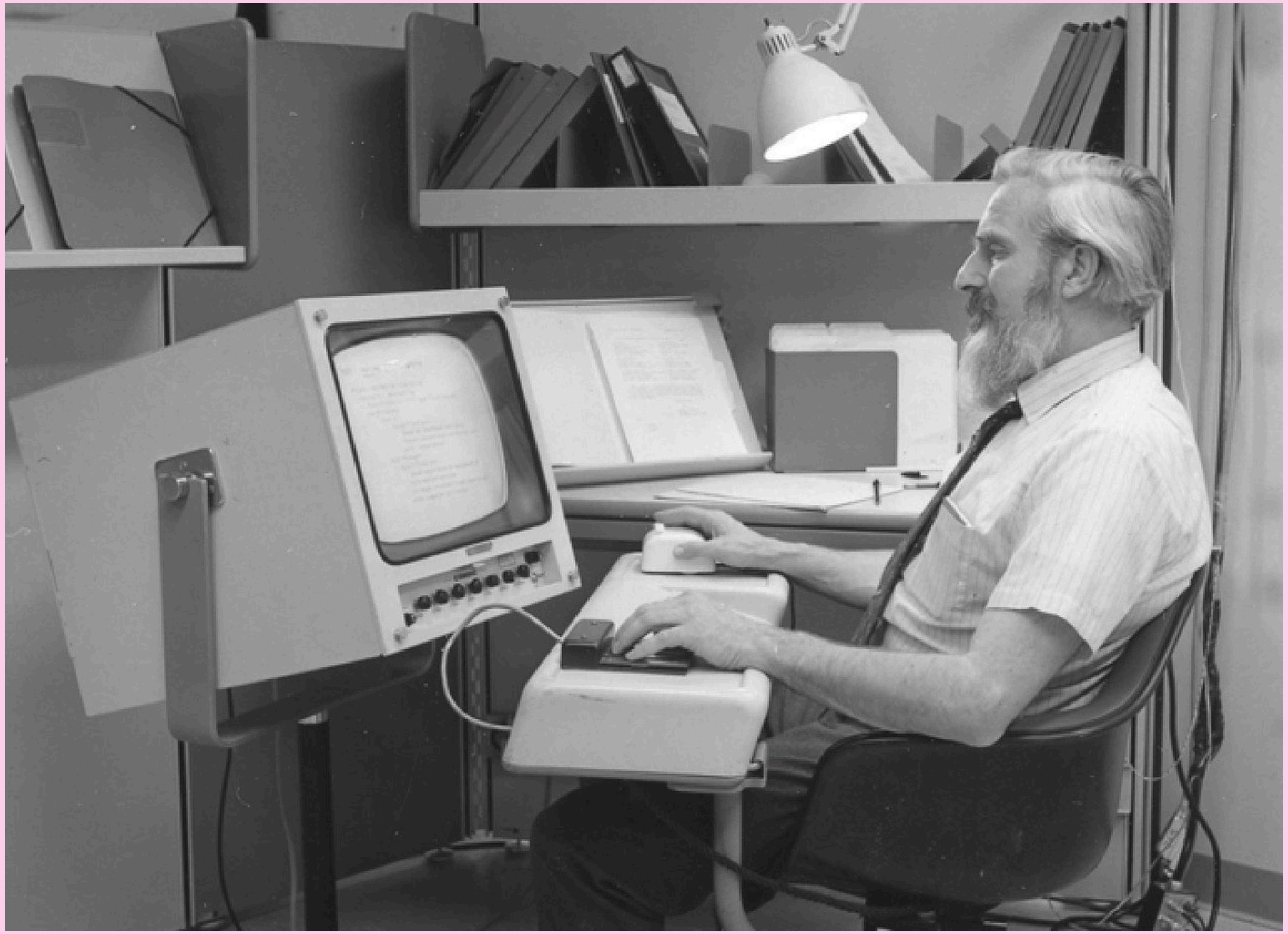
OBSERVATIONS BASED ON TEXT

Principles of Interaction

Principles of Interaction Design are essential guidelines that **ensure interfaces are user-friendly, intuitive, and effective**. These principles include *consistency, feedback, visibility, affordance, learnability, efficiency, flexibility, simplicity, user control, and accessibility*. These principles revolve around how users interact with an interface, focusing on making the experience intuitive, efficient, and effective. Sketchpad introduced the **concept of constraints**, allowing users to **define relationships between objects** (e.g., keeping lines parallel or ensuring shapes remain a certain size). It ensured **consistency** with **uniform tools and commands**, provided **immediate feedback** through **real-time updates**, and featured clear **affordances** for drawing and **selecting**. The interface was designed for **learnability**, mimicking manual drafting, and **achieved efficiency and flexibility** by *allowing users to create and modify designs with reusable components.*

Engelbart’s NLS (oN-Line System, 1968)

Developed by **Douglas Engelbart**, the NLS was a pioneering system that introduced many of the concepts we associate with modern computing, including the mouse, hypertext, and collaborative work. It was demonstrated in what is famously known as "**The Mother of All Demos.**"



Engelbart’s NLS (oN-Line System, 1968)

All the **features** of NLS were designed to help people **work together more effectively**, aiming to **enhance** the user's capabilities rather than just making the system **easier to use**. Engelbart's vision was to offer a rich, interactive experience for skilled users, in contrast to the later WYSIAYG (What You See Is All You Get) approach, which was more limited.

Hardware Components

- **CRT Monitor**: Used for displaying text and graphics, enabling interactive visual interfaces.
- **Mouse**: Introduced, allowing point-and-click interaction.
- **Chorded Keyset:** A five-key input device that allowed fast command entry.
- **Minicomputer (SDS 940)**: Powered the system, supporting time-sharing for multiple users.
- **Custom Terminals:** Integrated display, input devices, and networking for optimized user interaction.

User Interaction

- NLS was **not designed to be easy to learn**
- it employed the heavy use of **program modes,** relied on a strict hierarchical structure, did **not** have a **point-and-click interface**,
- forced the user to have to **learn cryptic mnemonic codes** to do anything useful with the system.
- The chord keyset, which complemented the modal nature of NLS, forced the user to learn a **5-bit binary code** if they did not want to use the keyboard**.**
- This was based on the piano chords

OBSERVATIONS BASED ON TEXT

Design Systems (Atomic Design)

Engelbart’s NLS embodies the principles of Atomic Design by creating a highly modular, flexible interface that can be broken down into fundamental elements and then built up into complex, functional systems. In Engelbart's NLS, **atoms** are the **fundamental interactive elements**, like individual commands (**text input, cursor movements**) and basic components (**characters, symbols**). **Molecules** are simple **combinations** of these atoms, enabling more complex tasks such as **text editing with cursor movements**. **Organisms** represent larger interactive systems within NLS, like the **text editor or file management system**, where various molecules interact to provide a cohesive user experience. **Templates** are **layouts** on the screen for specific tasks, combining organisms to support different workflows. **Pages** are actual instances of these templates, using **real data** to realize the interface's **functionality**.

OBSERVATIONS BASED ON TEXT

Concepts of Interface (D5R1)

Engelbart's NLS (1968) aligns well with David Reinfurt's concept of an interface. It acts as a medium that **simplifies interactions** by allowing users to **manipulate data** and see **immediate results** directly. This direct manipulation and graphical display make the system more **intuitive**. The **modular design** of NLS **breaks down complex tasks** into smaller, manageable components, which helps users perform actions more **easily** and **flexibly**. The system also integrates different input methods and provides **real-time feedback**, bridging the gap between **user actions** and **technological responses.** By introducing new ways to interact with computers and focusing on **user needs**, NLS demonstrates how interfaces can **effectively connect users** with technology as well as other users while offering innovative solutions.

OBSERVATIONS BASED ON TEXT

Principles of Interaction

Principles of Interaction Design are key guidelines for creating user-friendly and effective interfaces. These principles include direct manipulation, immediate feedback, consistency, affordance, learnability, efficiency, flexibility, error prevention, simplicity, user control, discoverability, and accessibility. **NLS provided real-time feedback through visual and auditory cues.** For example, when users interacted with the system using the mouse or keyboard, the system updated the display to reflect the changes instantly, such as updating text or graphics. This *immediate feedback helped users understand the effects of their actions and ensured that their commands were executed correctly*.The NLS interface also **demonstrated effective mapping** by **organizing controls and commands in a logical manner**. For example, the *layout of command buttons and input fields was designed to reflect their functional relationships, making it easy for users to predict the outcomes of their interactions.*

BIBLIOGRAPHY

- https://paruluniversity.ac.in/pu-mirror/7th-january-1963-the-day-that-changed-computer-graphics-forever-the-development-sketchpad#:~:text=Sketchpad%20allowed%20users%20to%20draw,computer's%20custom%20constructed%20ring%20structure.
- https://en.wikipedia.org/wiki/NLS_(computer_system)
- https://www.sitepoint.com/real-history-gui/
- chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/http://wexler.free.fr/library/files/sutherland%20(1963)%20sketchpad.%20a%20man-machine%20graphical%20communication%20system.pdf
- chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/http://wexler.free.fr/library/files/sutherland%20(1963)%20sketchpad.%20a%20man-machine%20graphical%20communication%20system.pdf