COMMAND LINE

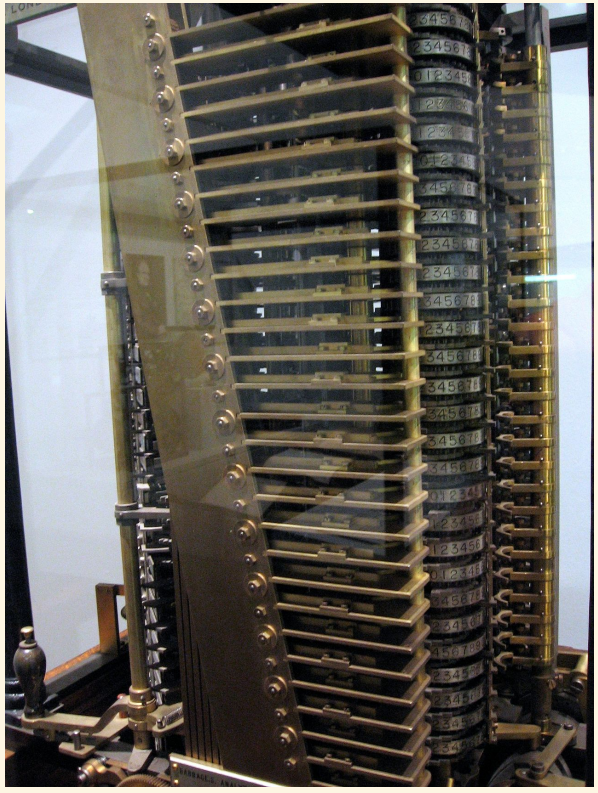
Architecture & Deployment

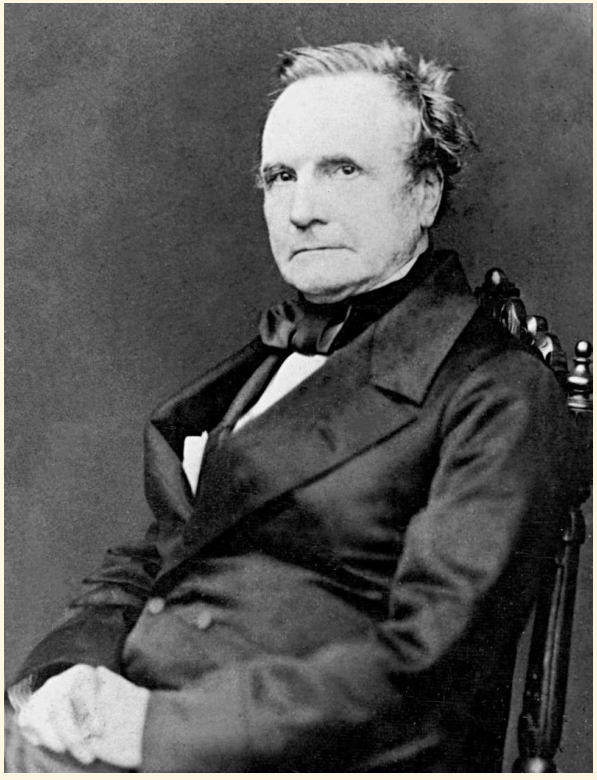
A SHORT HISTORY OF COMPUTERS & COMPUTER INTERFACES

For old time's sake.

THE FIRST GENERAL-PURPOSE COMPUTER (1837)

The Analytical Engine, proposed by Charles Babbage





Speaker notes

Charles Babbage, an English mathematician, proposed the mechanical Analytical Engine: the first digital programmable, general-purpose computer.

THE FIRST PROGRAMMER (1842)

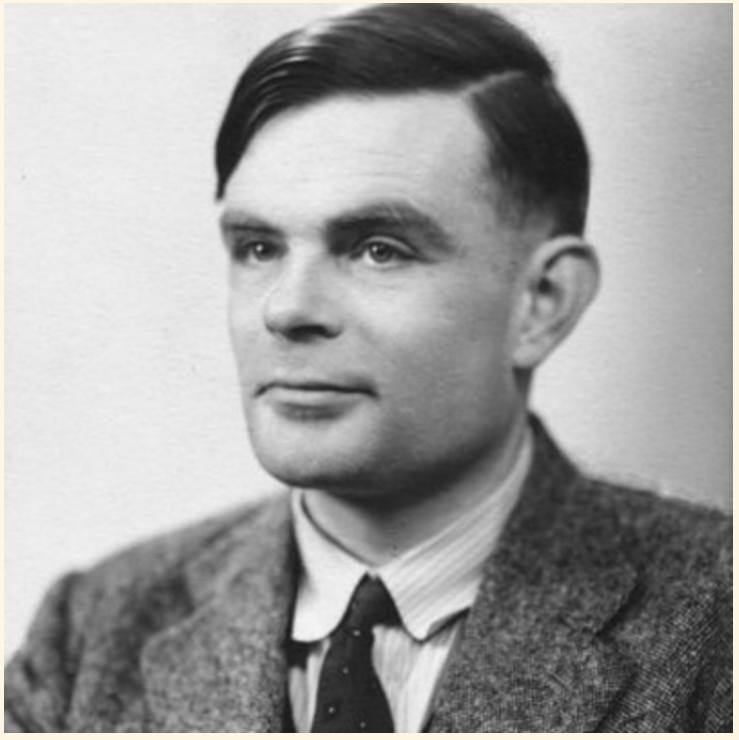
Ada Lovelace publishes the first algorithm

Speaker notes

In 1842, Ada Lovelace translated into English and extensively annotated a description of the engine, including a way to calculate Bernoulli numbers using the machine (widely considered to be the first complete computer program). She has been described as the first computer programmer.

A CENTURY LATER (1940S)

Alan Turing formalizes algorithms and computation

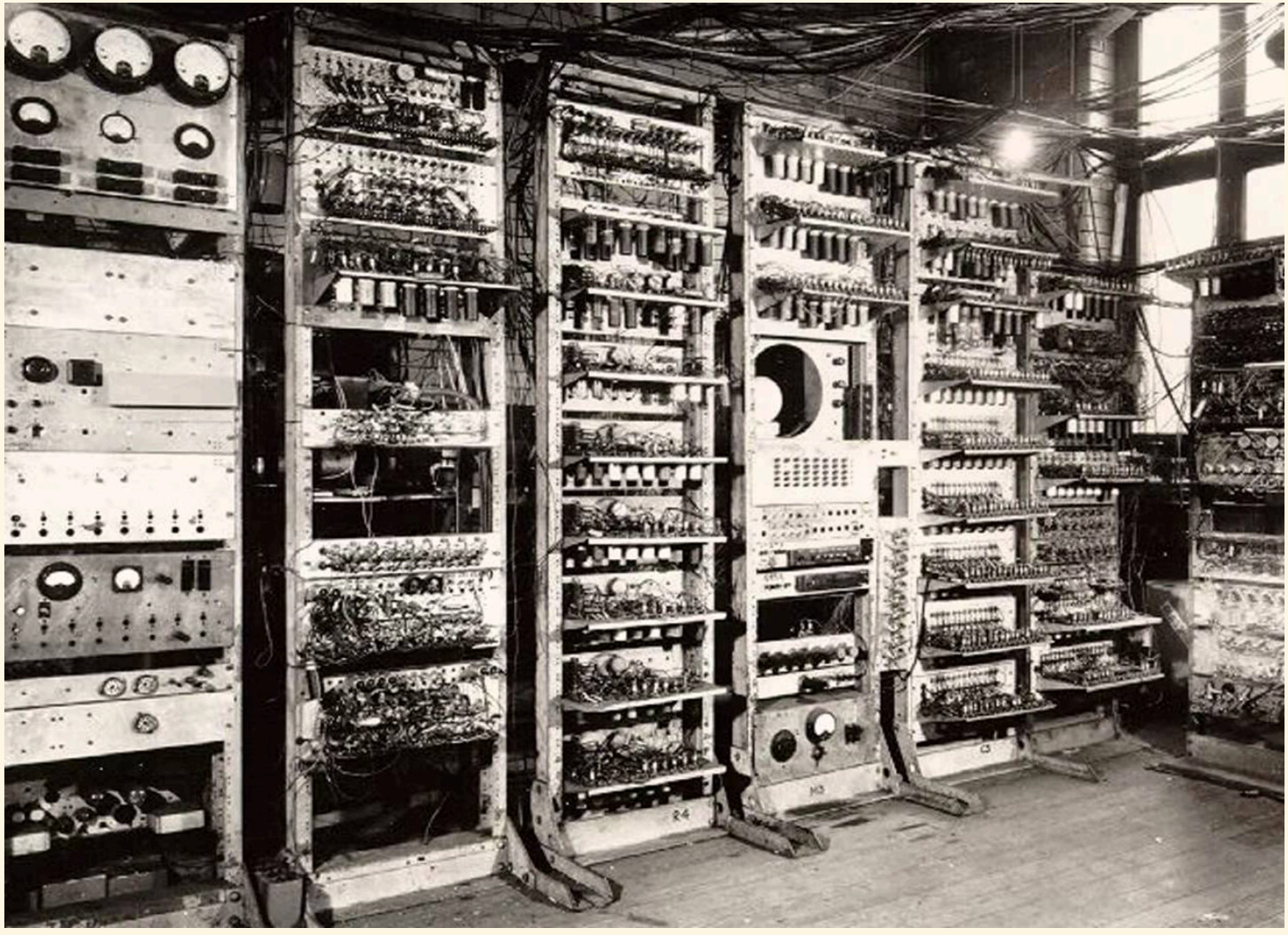


> *Did you see The Imitation Game?*

Speaker notes

Alan Turing formalized the concepts of algorithm and computation with the Turing machine. He is widely considered to be the father of theoretical computer science and artificial intelligence.

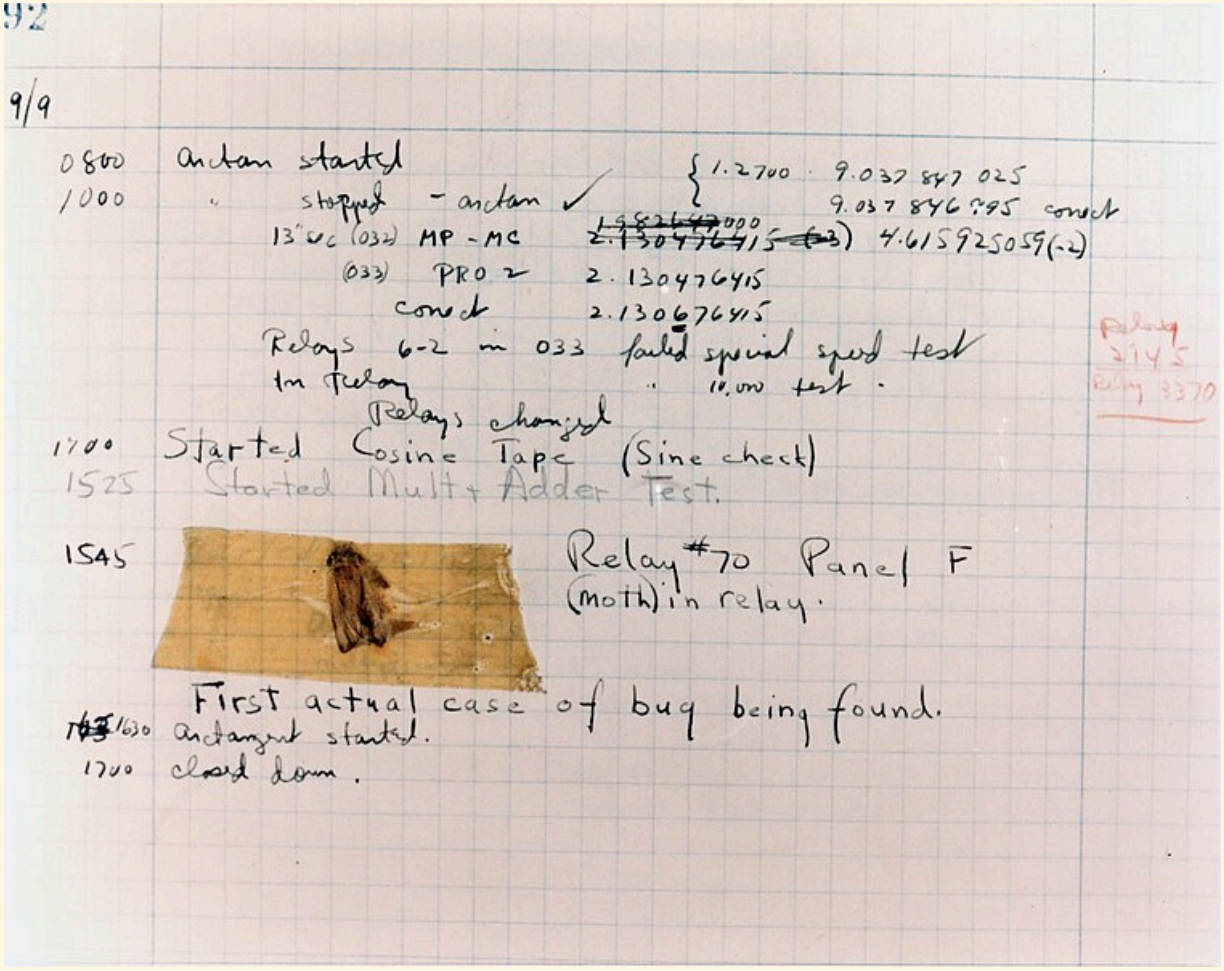
ENIAC (1946)



Speaker notes

At that time, there was no such thing as a stored computer program. Programs were **physically hard-coded**. On the ENIAC, this was done using function tables with **hundreds of ten-way switches**, which took weeks.

THE FIRST BUG THAT WAS CAUGHT (1947)



92

9/9

0800 Arctan started
1000 " stopped - arctan ✓
13"uc (032) MP - MC
(033) PRO 2
correct
Relays 6-2 in 033 failed special speed test
In relay " 10.00 test.
Relays changed
1700 Started Cosine Tape (Sine check)
1525 Started Multy Adder Test.
1545 Relay #70 Panel F
(moth) in relay.
First actual case of bug being found.
1700 closed down.

{ 1.2700 9.037 847 025
9.037 846 995 correct
2.13047 6415 (-3) 4.615925059(-2)
2.13047 6415
2.13067 6415

Relay 2145
Relay 3370

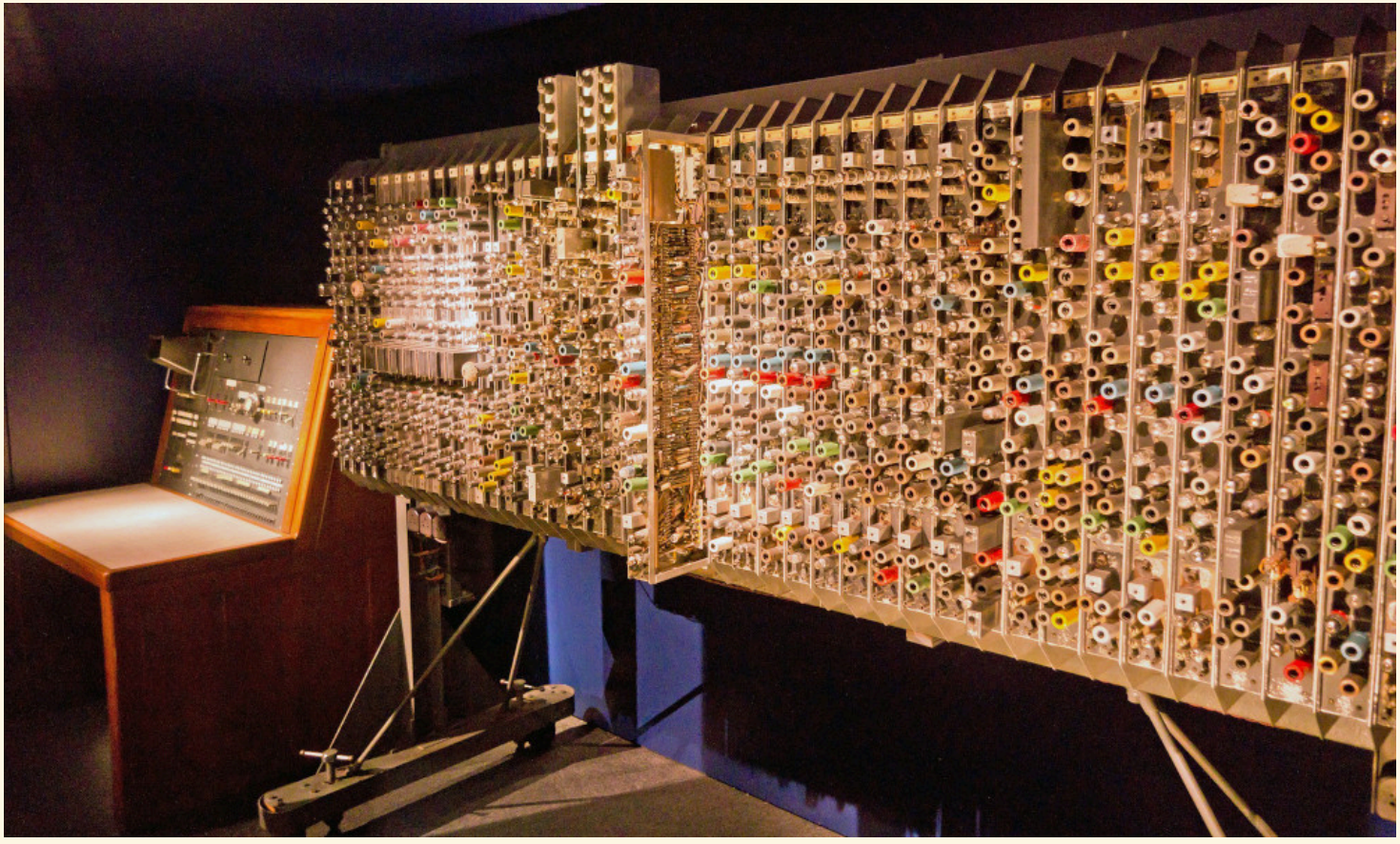
Speaker notes

Computers like these are electro-mechanical computers because they were based on switches and relays, as opposed to the transistors our current electronic computers are based on.

When you had a bug in one of these computers, *debugging* meant getting your hands dirty and finding the actual bug in the physical machine.

STORED COMPUTER PROGRAMS (1950S)

The Automated Computing Engine, designed by Alan Turing



Speaker notes

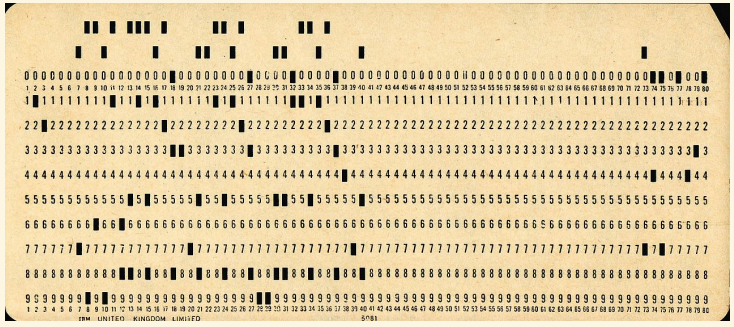
The Automatic Computing Engine (ACE) was a British early electronic serial stored-program computer designed by Alan Turing. It used mercury delay lines for its main memory.

MERCURY DELAY LINE MEMORY (1950S)

Better not spill it...

PUNCHED CARDS (1950S)

One of the first user interfaces



Invented in 1725

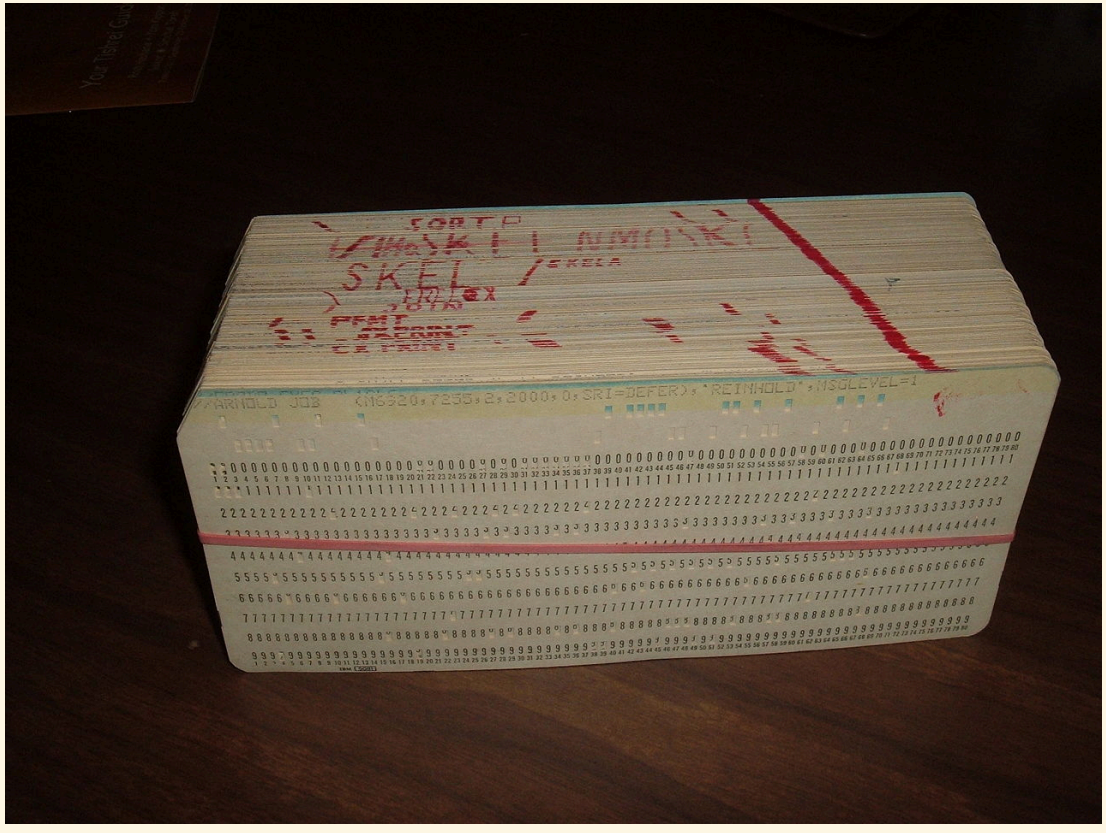
Speaker notes

Many early general-purpose digital computers used punched cards for data input, output and storage. Someone had to use a keypunch machine to write your cards, then feed them to the computer.

Punched cards are much older than computers. They were first invented around 1725 to control mechanical looms.

A TYPICAL PROGRAM (1950S)

*Whatever you do, **DON'T** drop it!*



TELETYPEWRITER (1960S)

The first **command line interfaces (CLI)**

Speaker notes

Teletypewriters (TTYs) became the most popular **computer terminals** in the 1960s. They were basically electromechanical typewriters adapted as a user interface for early mainframe computers.

This is when the first **command line interfaces (CLI)** were created. As you typed commands, a program running on the computer would interpret that input, and the output would be printed on physical paper.

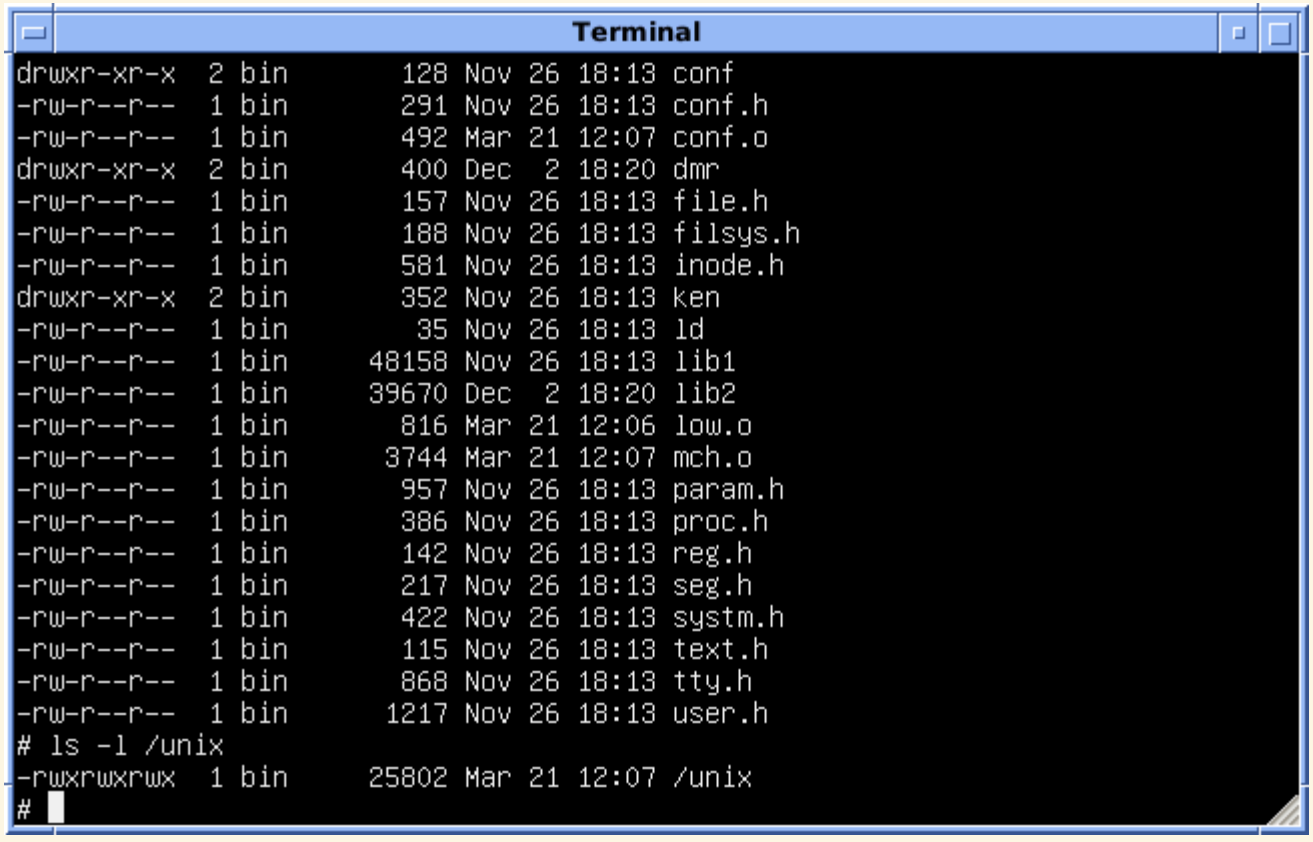
VIDEO TERMINALS (1970S)

Speaker notes

As available memory increased, **video terminals** such as the VT100 replaced TTYs in the 1970s. Initially they only displayed text. Hence they were fundamentally the same as TTYs: textual input/output devices.

UNIX (1970S)

The first portable operating system



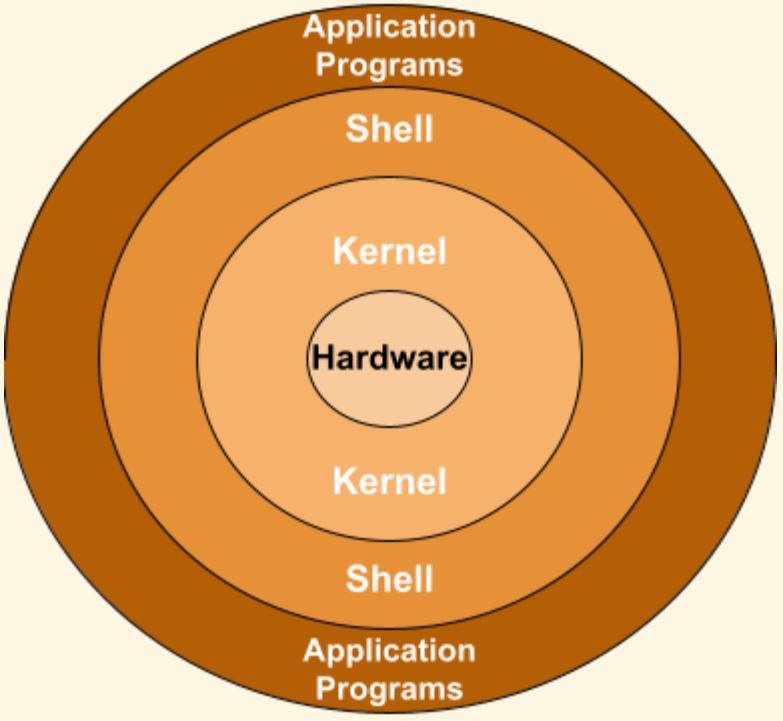
Speaker notes

It's also in this period that the Unix operating system was developed. Compared to earlier systems, Unix was the first **portable operating system** because it was written in the C programming language, allowing it to be installed on multiple platforms.

Unix is the ancestor of Linux. FreeBSD, a Unix-like system, is also used as the basis for macOS (since Mac OS X).

SHELLS (1970S)

Text-based at that time



Speaker notes

In Unix-like systems, the program serving as the **command line interpreter** (handling input/output from the terminal) is called a **shell**. It is called this way because it is the outermost layer around the operating system; it wraps and hides the lower-level kernel interface.

GRAPHICAL USER INTERFACES (1980S)

Also a type of shell

Speaker notes

Eventually, graphical user interfaces (GUIs) were introduced in reaction to the perceived steep learning curve of command line interfaces. They are one of the most common end user computer interface today.

Note that the GUI of a computer is also a shell. It's simply a different way to interact with the kernel (graphical instead of textual).

MOTION SENSING USER INTERFACES (2000S)

Invented 1940s, on TV 1950s, in wise use 2000s

Speaker notes

Motion sensing

TOUCH USER INTERFACES (2000S)

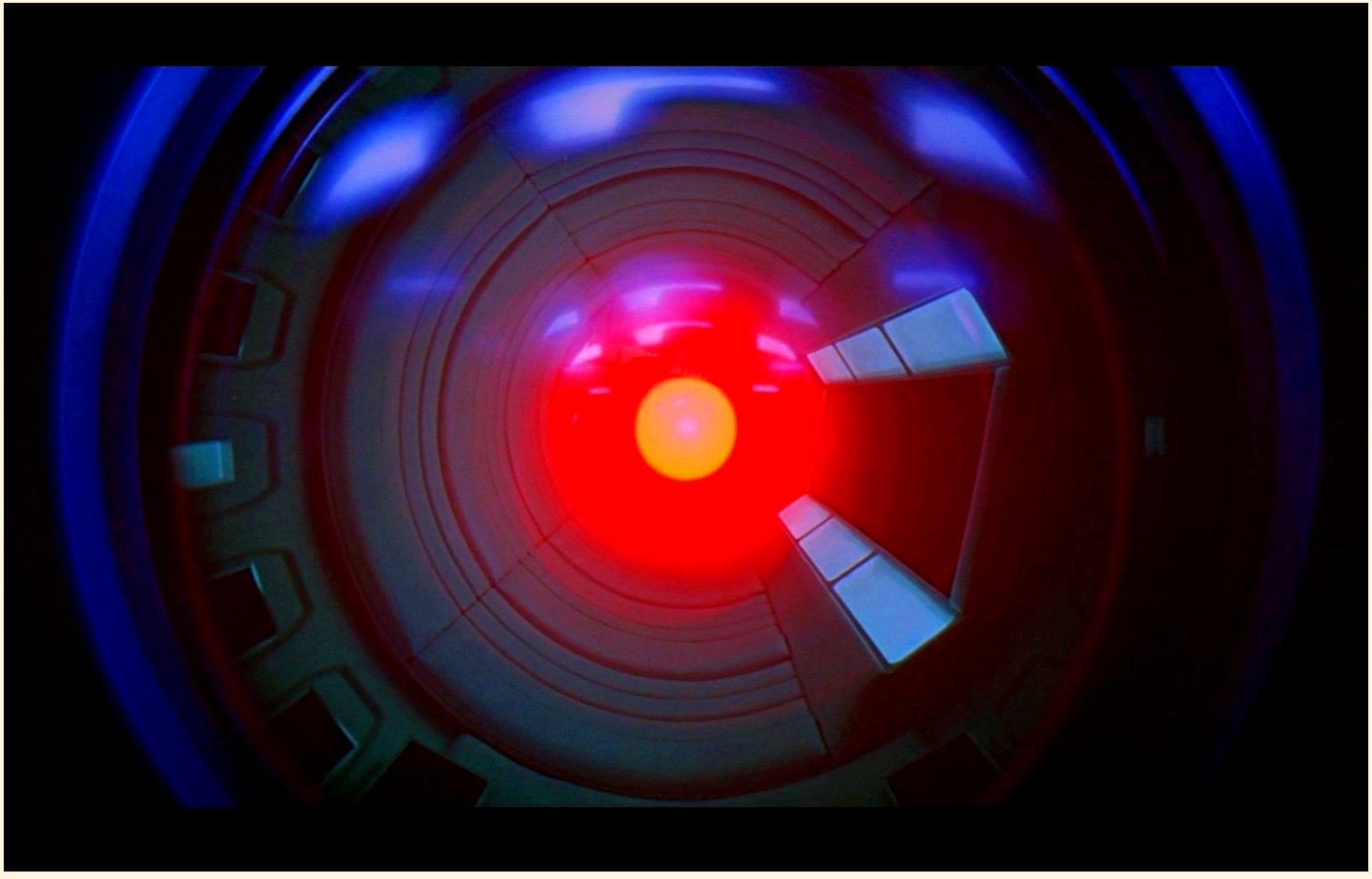
Invented 1960s, on TV 1980s, in wise use 2000s

Speaker notes

Touch user interface

VOICE USER INTERFACES (2010S)

Invented 1950s, on TV 1960s, in wise use 2010s



Speaker notes

Voice user interface

AUGMENTED REALITY (2010S)

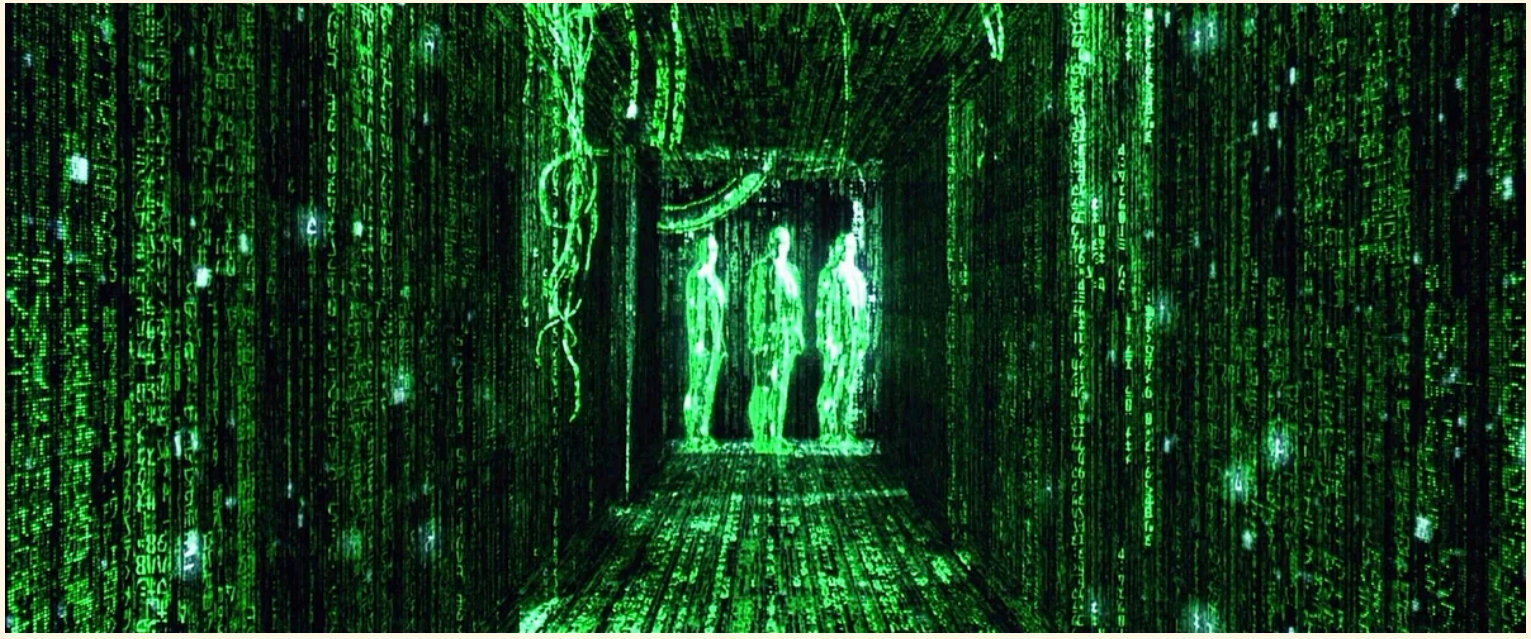
Invented 1960s, on TV 1970s, in wise use 2010s

Speaker notes

Augmented reality

VIRTUAL REALITY (2010S)

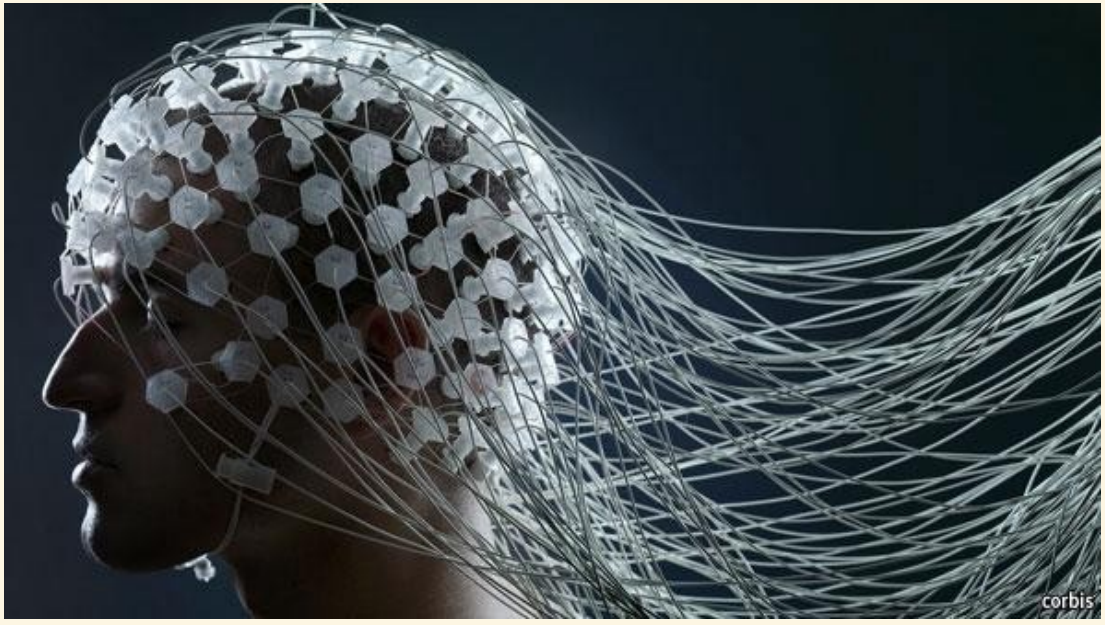
Invented 1960s, on TV 1980s, in wise use 2010s



Speaker notes

Virtual reality

TOMORROW?



Speaker notes

Brain-computer interface?