


This is [Beagle-Ears.COM](#) -> [Lars](#) -> [Professional](#) -> [Computer History](#) -> [IBM System/360](#)

IBM 360/370/3090/390

- [Introduction](#)
- [Generations of IBM360->370->390](#)
- [Operating Systems](#)
- [Pictures of System/360 Processors and Peripherals](#)
- [Individual S/360 Models](#)
- [Model Numbers of System/360 Peripherals](#)
- [Run OS/360 on your Linux box](#)

Outside links:

- [Timeline of IBM's Advanced Computing Systems](#) projects from 1953-1970.
- [Timeline of major events relating to System/360.](#)
- [The History of VM/370 and the VM Community](#) from [Melinda Varian's Home Page](#)
- [IBM's Reference Library for S/390](#)
IBM now sells most of the manuals on CD-ROM; a significant fraction is also available [online](#) but may only be accessed by licensed customers. (My experience has been that the link I was given worked for me only once!)
- ["ESA/390 Principles of Operation"](#) from [IBM BookManager](#)  [BookServer Library](#)
- [IBM Reference Manual Links](#) by Parker Gillespie (System Programmer at Stanford University)
- [Tom van Vleck's tribute to 360/67.](#) Tom is best known (to me) as the host of the [Multicians](#) homepage.

Introduction

Until the early 1960's, every computer model was generally designed independently, and sometimes individual machines were custom modified for a particular customer. IBM changed this forever, when they announced the IBM-360 family of computers in April 1964.

The IBM-360 family of computers ranged from the model 20 minicomputer (which typically had 24 KB of memory) to the model 91 supercomputer which was built for the North American missile defense system. Despite their differences, all these machines had the same user instruction set; on the smaller machines many of the more complex instructions were done in microcode rather than in hardware. For example, machines in the lower midrange did not have multiplier hardware, but the microcode implemented multiplications by repeated addition. It was rumored that the smallest machines did addition by repeated increments!

The machines had different operating systems. The smallest machines could not really support an operating system and were often used for specialized applications, where a program was loaded from binary punched cards at startup. The middle range used a system called DOS (not related to MS-DOS) and the higher end system was called OS/360. These were the machines that established 32 bits as the standard for computers.

The [IBM Journal of Research and Development](#) recently reprinted the 1964 paper [Architecture of the IBM System/360](#) by Gene Amdahl, Fred Brooks and G A Blaauw. (16 pages, PDF).

Generations of IBM360 -> 370 -> 390

The original 360 family was announced in 1964, and the lower midrange model 40 was the first to ship a year later. The most interesting version was model 67 (first shipped June 1966) which had hardware to support virtual memory. IBM had planned a special operating system for it (TSS/360), which they never managed to get to work well enough to be usable. Within IBM, model 67 was used with a system known as CP-67, which allowed a single 360/67 to simulate multiple machines of various models. This turned out to be very useful for developing operating systems.

In the summer of 1970, IBM announced a family of machines with an enhanced instruction set, called System/370. These machines were all designed with virtual hardware similar to 360/67, and eventually all the operating systems were enhanced to take advantage of it in some way.

When System/360 was successful, other companies started making their machines similar to IBM's, but not close enough to actually run the same software. In 1970, however, Gene Amdahl (who had been the chief architect for the 360 family) started a company to build a series of machines that were direct clones of the 360-370 architecture, and later Hitachi followed suit. (The first Amdahl machine was shipped in 1975.)

Big, fast disk drives were one of the strengths of IBM. In 1973, the big mainframe disk drive was model 3330-11: 400 MB for \$111,600 or \$279/MB. By 1980, you could get the 3380: 2.5GB for \$87,500 or \$35/MB. DRAM prices were dropping, too: In 1979 the price was cut from \$75,000/MB to \$50,000/MB.

Through the 1970's and 1980's, the machines got bigger and faster, and multi-processor systems became common, but the basic architecture did not change. Around 1982, addresses were extended from 24 bits to 31 bits (370-XA), and in 1988 extensions were put in to support multiple address spaces (370-ESA). In 1990, the ES/9000 models came out with fiber-optical I/O channels (ESCON), and IBM began using the name System/390.

In 1999, IBM released a new generation of S/390. [A special issue of IBM Systems Journal](#) describes it's technology.

Operating Systems

The 360 family was intended to have 3 operating systems:

- DOS/360 for the small machines.
- OS/360 for the midrange and high end.
- TSS/360 for Time-Sharing Multi-User systems.

In actuality, TSS never worked worth a damn. It was replaced by several very different systems:

- CP-67 (later VM/370)
- MTS (Michigan Time-Sharing System)
- CRJE, Wylbur, and other limited systems
- TSO (Time-Sharing Option for OS/360).
- Amdahl's UTS, a full Unix system.

DOS/360

Disk Operating System for S/360 was a small, simple system suitable for the S/360-30 and S/360-40 minicomputers. (There was also a version called TOS/360 for machines with tape drives and no disks.) A typical 360/30 might have 24 to 64 KILO-bytes.

DOS would let you run one program at a time, although there were provisions for writing special program that would run in a "foreground partition" to handle special I/O related functions, such as spooling cards to disk or printing output files from disk while the batch job stream was running. There were even some "foreground" programs available that could do (fairly simple) database updates via remote terminals while the batch job stream was running.

OS/360

Operating System for S/360. The name was intended to convey that this was ONE system that was suitable for the whole family of machines. In reality, there were 3 versions from the very beginning:

OS/360-PCP (Principal Control Program) was a very simple version which could only run one program at a time. Inside IBM, this was used to develop the tools that would eventually run on the larger OS/360 versions. In the real world, machines that were too small to run the other OS versions, always ran DOS.

OS/360-MFT (Multi-programming with a Fixed number of Tasks) could run several programs at once, but you had to divide the memory into a set of partitions, and each partition could run one job at a time. If a partition was idle, the memory was not available to programs running in the other partitions. The advantage was that the system was very stable, and in most data centers, the workload could be structured into small jobs and big jobs.

OS/360-MVT (Multi-Programming with a Variable number of Tasks) allowed partitions to be created and deleted on the fly as needed. If there was memory available, it would search the job queue for a job that would fit in the space available, and then create a partition of the size that the job requested. The advantage is obvious; the disadvantage was that after a while, small jobs with long running-times would be sitting in the middle of memory, leaving open spaces before and after that were too small to fit any of the jobs that were waiting. To make MVT work, you needed to install a package called the HASP job scheduler, which managed the queue, tagged each job with one of several fixed sizes, and released only one job at a time into the MVT scheduler's queue for each of a predefined number of "slots". The result was very much like MFT, but one could make room for a very large job to take the whole machine when needed.

After the virtual addressing hardware became available, these systems were renamed: MFT became OS/VS1 and MVT became OS/VS2. Both versions used the addressing hardware to combine leftover areas spread throughout memory into a logically contiguous area for another job to run in.

A further enhancement to OS/VS2 became MVS (Multiple Virtual Spaces).

TSS/360

TSS (Time-Sharing System) was an attempt to do everything that Multics did, but better. IBM knew this project was strategically important and put a lot of resources into it. Hundreds of programmers. The result was a system that was bloated and unstable. The first release took over ten minutes from power-up to the first login prompt, but usually crashed within 10 minutes after that. Eventually, IBM killed the project.

CP-67 -> VM/370

IBM had a research center at MIT, and their staff wanted to build a much simpler multi-user system. The idea was that the operating system would create a virtual machine for each user, and then a simple single-user system could run in each virtual machine. This had been tried before; it had worked poorly, but nobody knew why. The team was confident that they could find the problem and fix it. The resulting VM (Virtual Machine) operating system was a system programmer's ideal workshop, and turned out to be the most useful system for

developing other operating systems on.

The virtual machine looked like a real 360, but the machine it ran on had a translation table that converted the addresses into distinct addresses on the real machine. Entries in the table could be marked to indicate "this section not available", and when the user program touched a memory address in such a section, an interrupt would transfer control to the control program, which could bring in the section from a "swap file" on a disk or drum. Periodically, areas that had not been used, would be "swapped out" to such an area, and their table entries would be marked as "not available". This worked very well, except that if a program had less memory available than was used, it could cycle around and need to load a new page for each instruction executed, only to see the pages immediately swapped out. In fact, this turned out to be the reason that the idea had failed in the past. The solution had two parts: Write programs to keep the data that was used together near each other in memory (rather than spread over many sections) and keep track of how many different pages had been used recently, and give the program at least that many, and if you can't, then either kill it or postpone it entirely.

In time, the control program was extended so that it could simulate the address translation hardware for a virtual machine. This made it possible to run a copy of the control program inside the virtual machine. You could even simulate the address translation hardware of a new machine that had not been built yet. This was used to develop versions of OS/MVT for the S/370 machines while the hardware design was still being finished.

For many years, IBM was extremely reluctant to let customers run VM/370. At first, they were supposed to be running TSS; then they were supposed to be running TSO. Eventually, the user community persuaded IBM to release it, and 10-15% of all S/370 were running VM. By 1989 there were 20,000 licenses paying monthly rental fees.

MTS - Michigan Time-Sharing System

The Michigan Time-Sharing System was started in 1966 to exploit a modified 360/65 which was to be built specifically for this project. By the time it was delivered in 1967, others had heard about the machine, and over 100 units had been ordered. The history of the MTS project is told in these 3 articles published by University of Michigan in 1996, when the last MTS machine was turned off.

- [A history of MTS - 30 Years of Computing Service](#)
- [MTS Fostered creation of Computing Community](#)
- [MTS - Thanks for the Memories](#)

(These articles are also mirrored [here](#).)

University of Newcastle ran an MTS shop, and has some nice [pictures](#).

Wylbur, Waterloo, CRJE etc

Many academic computer centers had S/360's running OS/MVT. The "normal" way to run programs on these machines was to type the program on punched cards, and then submit them through a card reader. By the 1970's, everyone knew that this was cumbersome, and wanted to move to talking to the machine through typewriter terminals. But this was considered an expensive luxury. Some ingenious computer centers came up with the idea of building a program that could talk to terminals while running under the MVT batch system. The terminals would only be allowed to edit text files and send such a file to the batch job stream where it would be processed in the same way as if it had been read in from a card reader. These systems became very popular at universities. IBM offered a couple of systems like this as program products, but they were never as

nice as the university developed systems.

TSO

After TSS died, a new project was started to create a version of OS/360 which could run user-written programs conversationally from terminals like TSS had been meant to do. This became known as TSO (Time-Sharing Option). It worked, but not wonderfully. OS/MFT and OS/MFT were very efficient and stable by then, but TSO ate up half of the machine, and was likely to crash it once a day.

UTS: Amdahl Unix

In the 1970's, AT&T Labs had ported a version of Unix to run on S/370 (using some parts of TSS) but it had never been released outside the labs. In the late 1970's, a graduate student had ported Unix to run on VM/370. He had expressed an interest in continuing the work at IBM, but because of political arguments between different IBM divisions, he never got an offer, and went to Amdahl instead. In May 1981, Amdahl released UTS, a full Unix system that ran under VM.

My Personal Experiences

The first IBM-360 I used in Copenhagen was the spooling front-end for the 7094 at the Technical University. In 1970, the technical university installed a 360/65, later upgraded to a 360/75. When it came in, it had 1 MB of RAM (magnetic core memory in those days) and a roomful of disk drives, probably adding up to about 200 MB.

Today's S/390 mainframes are direct descendants of the IBM-360 family.

How to Make Yourself a (Simulated) S/360

[Roger Bowler](#) in England is working on a simulator called [Hercules-390](#) that will simulate an S/370 or S/390-EXA on a Linux machine, and [Jay Maynard](#) in Texas is getting [OS/360 MVT](#) to SYSGEN and IPL on it. The distribution seems to be available from www.snipix.freemove.co.uk/hercules.htm and there is a mailing list for implementers at www.onelist.com/community/hercules-390.

Further Reading

If you enjoyed this page, the following may also interest you:

- [Computer History](#) - from this website: a general introduction to the history of computing. Discusses what came before the IBM 360 family as well as other systems from the same era.
- [Computers and Operating Systems](#) - from this website: the general index to computer related stuff here.
- [Telecommunications](#) - from this website: general index to TeleCommunications, including the history of the field.

```
$Log: ibm360.htm,v $
Revision 1.7  2001/10/26 13:28:00  lars
Replaced CMC -> Beagle-Ears
```

```
Revision 1.6  2001/06/05 07:27:38  lars
*** empty log message ***
```

Revision 1.5 2001/04/08 16:39:53 lars
Update links.

Revision 1.4 2001/01/04 05:15:12 lars
Added file of S/360 Models

Revision 1.3 2000/11/26 19:32:29 lars
Added information about MTS.

Revision 1.2 2000/08/23 04:43:49 lars
*** empty log message ***

Revision 1.1 2000/08/15 01:25:06 lars
Rearranged files, moving these from computer/ to comphist/

Revision 1.6 2000/06/24 18:33:48 lars
Add cross-links to other sections of site.

Revision 1.5 2000/06/22 07:31:55 lars
Added links from my bookmarks file.

Revision 1.4 2000/02/15 06:56:04 lars
Added Hercules simulator information.

Revision 1.3 1999/11/30 04:19:47 lars
Add lots of details about operating systems.

Revision 1.2 1999/11/08 16:46:34 lars
More computer history links.

Revision 1.1 1999/09/07 04:20:41 lars
Added encryption page.
Added more detail in historical sections.
Improved index page organization.