The IBM mainframe: How it runs and why it survives



Enlarge / A Z16 Mainframe.

reader comments

162 with

Mainframe computers are often seen as ancient machines—practically dinosaurs. But mainframes, which are purpose-built to process enormous amounts of data, are still extremely relevant today. If they're dinosaurs, they're T-Rexes, and desktops and server computers are puny mammals to be trodden underfoot.

It's estimated that there are 10,000 mainframes in use today. They're used almost exclusively by the largest companies in the world, including two-thirds of Fortune 500 companies, 45 of the world's top 50 banks, eight of the top 10 insurers, seven of the top 10 global retailers, and eight of the top 10 telecommunications companies. And most of those mainframes

come from IBM.

In this explainer, we'll look at the IBM mainframe computer—what it is, how it works, and why it's still going strong after over 50 years.

Setting the stage

Mainframes descended directly from the technology of the first computers in the 1950s. Instead of being streamlined into low-cost desktop or server use, though, they evolved to handle massive data workloads, like bulk data processing and high-volume financial transactions.

Vacuum tubes, magnetic core memory, magnetic drum storage, tape drives, and punched cards were the foundation of the IBM 701 in 1952, the IBM 704 in 1954, and the IBM 1401 in 1959. Primitive by today's standards, these machines provided the functions of scientific calculations and data processing that would otherwise have to be done by hand or mechanical calculators. There was a ready market for these machines, and IBM sold them as fast as it could make them.

In the early years of computing, IBM had many competitors, including Univac, Rand, Sperry, Amdahl, GE, RCA, NEC, Fujitsu, Hitachi, Unisys, Honeywell, Burroughs, and CDC. At the time, all of these other companies combined accounted for about 20 percent of the mainframe market, and IBM claimed the rest. Today, IBM is the only mainframe manufacturer that matters and that does any kind of business at scale. Its de facto competitors are now the cloud and clusters, but as we'll see, it's not always cost-effective to switch to those platforms, and they're not able to provide the reliability of the mainframe.

Advertisement

Built-in redundancy

By any standard, mainframes are enormous. Today's mainframe can have up to 240 server-grade CPUs, 40TB of error-correcting RAM, and many petabytes of redundant flash-based secondary storage. They're designed to process large amounts of critical data while maintaining a 99.999 percent uptime—that's a bit over five minutes' worth of outage per year. A medium-sized bank may use a mainframe to run 50 or more separate financial applications and supporting processes and employ thousands of support personnel to keep things running smoothly.

Most mainframes process high-volume financial transactions, which include things like credit card purchases at a cash register, withdrawals from an ATM, or stock purchases on the Internet.

A bank's lifeblood isn't money—it's data. Every transaction a bank makes involves data that must be processed. A debit card transaction, for instance, involves the following data that must be processed:

- Retrieving a user's debit account info
- Validating the user ID and PIN
- Checking the availability of funds
- Debiting the user's account for the transaction amount
- Crediting the seller's account

All this must happen in seconds, and banks have to ensure they can maintain a rapid response even during high-volume events such as shopping holidays. Mainframes are designed from the ground up to provide both redundancy and high throughput for these purposes. High-speed processing is no good if processing stops during business hours, and reliable processing is no good if people have to wait minutes for a transaction to process.

When you process a financial transaction, it means you're

making money. If you’re processing a lot of transactions, you need to spend a lot of money on redundancy to keep things running smoothly. When parts inevitably fail, the show must go on. That’s where mainframes’ built-in redundant processing comes in.

https://arstechnica.com/?p=1930955