https://web.archive.org/web/20201129083332/https://qz.com/emails/quartz-obsession/1316525/ 22 September 2025 at 14:47 BST

INTERNET ARCHIVE Wayback Machine https://qz.com/emails/quartz-obsession/1316525/ Go OCT NOV JUL
7 captures 29 Nov 2020 – 27 Jun 2024 2019 29 2020 2022 About this capture

YOU'RE READING THE QUARTZ WEEKLY OBSESSION EMAIL FROM JUNE 28, 2018

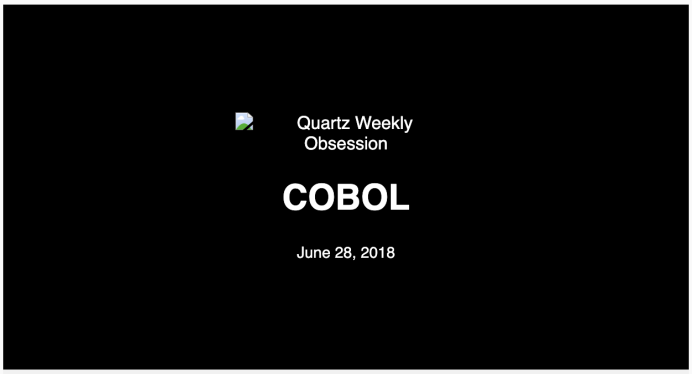
COBOL

📪 **Want to escape the news cycle? Try our Weekly Obsession.**

Enter your email

Sign me up

By providing your email, you agree to the Quartz Privacy Policy.



Vintage vibes

The last time you used an ATM, chances are the transaction was powered by a nearly 60-year-old computer programming language.

Common Business-Oriented Language—the ancient computer code better known as COBOL—was developed in 1959 as a business-focused standard programming language, and is still relied upon by banks around the world. It's responsible for $3 trillion in commerce in the US every day. As of 2014, 92 of the top 100 banks, as well as 71% of the companies in the Fortune 500, were still running COBOL programs on mainframe computers.

And if it's not broke, why fix it? Well, with COBOL, when something does break, there soon may not be anyone left who can. The baby boomers who know the language best are either retired or close to it, and those who would replace them just do not find COBOL very sexy.

So what happens next?

Tweet this

View this email on the web

Reuters/Kacper Pempel

BY THE DIGITS

220 billion: Lines of COBOL code in use today

95%: Proportion of ATM swipes that rely on COBOL

$749.9 million: Cost of replacing the Commonwealth Bank of Australia's core COBOL system in 2012

$15 billion: Estimated cost of upgrading the systems of the world's top 20 banks

294,900: Number of computer programming jobs in the US in 2016

138: Age of the first computer programming language, created by Ada Lovelace in 1883

ORIGIN STORY

A common language

In the 1950s, standardized programming languages were very rare, so most computers had to run custom languages. In 1959, the US Department of Defense organized a group of roughly 40 government and computer industry experts into CODASYL, the Conference/Committee on Data Systems Languages. They were inspired by FORTRAN, a standardized language used in the scientific community. "We thought, if the scientific programmers are going to get a single language, we could do the same for business," said Bob Bemer, who worked on FORTRAN at IBM.

COBOL (**Co**mmon **B**usiness **O**riented **L**anguage) was largely influenced by the MATH-MATIC and FLOW-MATIC English-syntax languages created by American computer scientist Grace Hopper and her team. Hopper would also be instrumental in the creation of COBOL, which was commercially ready in 1960 and would evolve throughout the decade. By 1970, it was the most widely used computer language in the world.

COBOL is now considered a legacy language, which means it's no longer supported by newer systems, but its usefulness in existing systems means it has stuck around. It's still widely used to support numerous business systems, including the operation of deposit accounts, check-clearing services, credit and debit card networks, ATMs, mortgage servicing, loan ledgers, and other services.

EXPLAIN IT TO ME LIKE I'M 5!

Programming for dummies

In the most simple terms, a programming language specifies a set of instructions that produces various kinds of output. In the computing world, this language is used to create programs, or applications, that implement specific algorithms, or a set of rules.

Writing a program is often compared to writing a food recipe, as detailed in this video from CBC News.

Giphy

TIMELINE

A brief history of programming

1883: Ada Lovelace creates the first computer programming language, an algorithm for an early mechanical computer called the Analytical Engine.

1957–1959: The first high-level languages are created for scientific, mathematical, and business computing are developed, including FORTRAN, LISP, and COBOL.

1970: The PASCAL language is created by Niklaus Wirth as a way of teaching programming, but its commercial use would spread in the 1980s.

1983: The general-purpose C language evolves to C++. It would become integral in the development of commercial desktop and server/client applications, as well as video games.

1995: Three of today's most popular programming languages — JAVA, JAVASCRIPT, and PHP — are released. They will go on to power things like Facebook, Gmail, and WordPress.

US Navy

NEVERTHELESS, SHE PERSISTED

The woman who paved the way

Grace Hopper was not your average computer programmer. In 1934, she became one of the first few women to earn a PhD in mathematics. She wanted to join the US Navy after the bombing of Pearl Harbor in 1941, but was rejected for being too short and, at 35, too old.

But Hopper wouldn't take no for an answer. Eventually, she got a waiver to join the US Naval Women's Reserve. Hopper was assigned to Harvard's Bureau of Ordnance Computation Project, where she programmed with IBM's Mark I, the first electromechanical computer in America. Her work there involved calculating rocket trajectories and calibrating minesweepers.

Hopper remained a Naval reservist while she continued her programming career. She's known for pioneering the idea of automatic programming, and for the groundbreaking concept of writing programs in words instead of symbols. At first she was told this wouldn't work, but a few years later her team released FLOW-MATIC, the first programming language to use word commands, which was a precursor to COBOL. She later served as a technical consultant to the group that created COBOL—acquiring the nickname "Grandma COBOL"—and developed software and tools for its use in the Navy.

Hopper is sometimes credited with coming up with the term "computer bug." As the story is commonly told, Hopper found a moth between the relays of a Harvard Mark II computer, taped it to a log book and captioned it, "First actual case of bug being found." In reality, she wasn't the one who made the joke, and "bug" had been used previously in the computing world. But while Hopper acknowledged that she was not the originator of the term, she did

enjoy recounting the story.

FUN FACT!

In 1959, in a fit of frustration over the speed of its development, CODASYL member Howard Bromberg bought a $15 tombstone with "COBOL" engraved on it and sent it to Charles Phillips, the director of the Data System Research Staff at the DOD.

WATCH THIS!

In the 1984 film *The Terminator*, samples of COBOL code of unknown origin are among those shown in the main character's vision display. The YouTube channel Behind the Screens has tracked other code snippets to an exacting degree to 1984 issues of *Nibble* magazine and a newsletter tracking Soviet satellites.

Giphy

MILLION-DOLLAR QUESTION

Why not upgrade?

The trouble of recruiting new COBOL programmers leads to an obvious question: Why not replace it with newer technology?

The answer centers around cost and risk. With COBOL so entrenched in the banking industry, any disruption would be costly, both in profits and reputation, as the UK's TSB bank learned recently. Its mobile banking system had a complete meltdown, and some customers logged into their account only to see other people's accounts instead of their own.

Studies show that improving core systems is a lower-risk option. But any modifications to existing systems, which contain decades of institutional knowledge, can be risky because of the dwindling number of coders still fluent in COBOL.

"As programmers either retire or leave for new jobs that living, human understanding of why the software was written the way it was gets gradually forgotten," said Neel Krishnaswami, a programming languages researcher at Cambridge University. "Changing a condition when you don't understand why it's there is a recipe for disaster."

COBOL is also a pretty solid programming language, according to its defenders. It's highly structured but readable compared to more complex machine code.

The Australian example

In 2012, the Commonwealth Bank of Australia began replacing its core COBOL system with software developed by the German company SAP. The bank's experience is not likely to inspire many other banks to try the same thing.

The process would take five years and cost more than 1 billion Australian dollars ($749.9 million), including many hiccups along the way.

During a June 2015 outage in CBA's payment and online systems, 600,000 payments and direct debits went missing. Another major disruption came three months later, which prevented customers from being able to use bank cards, apps, or pay bills. But it also gave some temporary hope that their debts had been forgiven, as some balances owed suddenly disappeared.

QUOTABLE

"He may have expired, so there is no option to get him or her to come back."

— Andrew Starrs of Accenture PLC, referring to retired COBOL programmers.

SITUATION REPORT

Bob Olson, a vice president at the global tech company Unisys, said one of his government clients "has someone who runs a chunk of their technology who's on oxygen, he's 70 years old, he knows the keys to the kingdom, he knows where everything is, it's all sitting in his head. They send out a police car to pick him up every morning and bring him in to work in a vault-like room."

THE PROGRAM GRAVEYARD

Is it dead?

As one computer programmer put it on Quora, "there are no dead languages but there are many zombies." These have not fared as well as COBOL has over the years, but you'll still find them lurking in places like dusty university math labs.

💻 **ADA:** Named after Ada Lovelace, widely considered the first computer programmer, ADA was created in the mid-1970s by the US DOD in an attempt to consolidate other languages. But ADA tried to do too much and eventually fell out of favor. It's still used by some medical professionals.

💻 **ALGOL:** COBOL's nerdier cousin (dealing with science rather than business) and a forefather to PASCAL, ALGOL brought the programming world code blocks ("begin/end") and nested code.

💻 **APL:** It was created to simplify complex mathematics, but many found its wide variety of symbols and lack of formatting confusing. APL is still used in university physics and math departments.

💻 **LISP:** Developed in the 1950s by early artificial-intelligence researchers, it grew due to the strength of the linked list, which turned out to be a an effective mechanism for working with multiple data types in a single structure. Some believe LISP foreshadowed concepts that came to be used in big data.

💻 **Pascal:** Niklaus Wirth developed Pascal as a way to teach the concepts of structured programming. It would play a major role in early programs on personal computers and still has its defenders, but it lost its mojo with the emergence of object-oriented programming.

💻**PL/I:** This was an attempt by IBM to create a single language that could do what COBOL, ALGOL, and FORTRAN did separately. It was ADA before ADA—and never caught on.

TAKE ME DOWN THIS 🐰 HOLE!

A small group of programmers wants to change how we code—before catastrophe strikes.

(What is code, anyway?)

Reuters/Kacper Pempel/files

POLL

Would you like your bank to upgrade its COBOL-based computer system?

Yes, I want a glitch to erase my student loan debt.

No, I haven't paid for anything with cash since 2012!

THE FINE PRINT

In yesterday's poll on Cheetos, **45%** of you said the snacks are "pure evil genius."

Today's email was written by Justin Sablich, edited by Jessanne Collins, and produced by Luiz Romero.

SOUND OFF

✏️ What did you think of today's email?

💡 What should we obsess over next?

🐰 What have you been obsessed with this week?

📭 Forward this email to a friend

facebook | twitter | external-link

Enjoying the Quartz Weekly Obsession? Send this link to a friend!

Want to advertise in the Quartz Weekly Obsession? Send us an email at ads@qz.com.

Not enjoying it? No worries. Click here to unsubscribe.

Quartz I 675 Avenue of the Americas, 4th Fl I New York, NY 10011 I United States

Inbox

An interactive email on the fascinating histories of everyday ideas

Book covers ›

Go ahead and judge

Electric airplanes ›

Flight of the batteries

QR codes ›

It's finally hip to be square



Subway maps

Underground design society



High heels

These shoes weren't made for walking



Micromobility

Tiny transport making tiny trips



Houseplants

Something to love



Flying cars

Where we're going, we don't need roads



H.P. Lovecraft

Oh, the cosmic horror



GPS

Guidance from above

Load more

QUARTZ

Make business better

Become a member

DISCOVER

- Field guides
- Emails
- Latest
- Popular
- Featured
- Obsessions

TOPICS

- Emerging industries
- Economics
- Lifestyle
- Politics
- Science
- Tech
- Work

MORE

- Search
- Careers
- About us
- Contact us
- Help center
- Quartz Creative
- Send us tips

Site map Terms & conditions Privacy policy Do not sell my personal information Notice at Collection Ethics and advertising agreements

© 2020 Quartz Media, Inc. All rights reserved.

↑ Beam me up, Scotty