

Why COBOL is Still Relevant Today & Where to Find COBOL Developers?

integrative systems
Integrating Strategy with Transformational IT Capabilities

TOP REASONS WHY COBOL IS STILL RELEVANT TODAY & WHERE TO FIND COBOL DEVELOPERS?

Inc 500

www.integrativesystems.com | contact@integrative-systems.com | + 1.866.468.7974 (INTSYSI)

The resurgence of COBOL is of no surprise. Expanded as Common Business Oriented Language, COBOL might not even feature in the university curriculum, and that's why there are fewer **COBOL developers**. But the matter is that this programming language is used by legacy systems on which the world still depends, for their day-to-day transactions.

Almost 70 to 80 percent of worldwide present-day business transactions are done on IBM iSeries. A 2017 research study uncovered billions of lines of COBOL codes, which are still in use today.

But there is a gap that needs to be bridged. Legacy AS/400 systems that run COBOL applications need **COBOL developers**. for regular maintenance and new application development. Existing legacy server infrastructures that run on AS/400 are irreplaceable. Government institutions, large banks, public services companies, insurance

companies, etc. – all of them are unwilling to stop [AS400](#) systems and upgrade to the newer IBM Power Systems.

Their argument is simple – when something is running super-fine, why change just for the sake of change?

Why cannot COBOL be Replaced?

AS/400 COBOL applications are used all over the world, and there is always a need for COBOL developers. Considering that more than 100,000 AS/400 customers exist, one can imagine the enormity of COBOL application deployments and the increasing need for COBOL developers.

- More than 60 million patients are treated and cared with applications developed on AS/400 COBOL apps. Thanks to efforts of **COBOL programmer**.
- Most ATM transactions, at least 90% of them are powered by COBOL application.
- Every year, almost 96% of vacations are booked using systems built on COBOL developers of all ages.
- Social Security Administration has more than 60 million lines of code written in COBOL.
- IRS uses COBOL applications; about 50 million lines of COBOL code are used at this single entity.

Has Something Better Come Along?

The answer is – it is not required. For so many business applications, COBOL has been a rock-solid foundation. COBOL applications haven't broken, are resilient, and secure. So, the need to replace it does not arise because if a system is working fine, then there is no need to tinker with it.

Another point of consideration is that COBOL has explicitly been designed for business applications. Other low-level programming languages such as Fortran and LISP are better suited to solve scientific problems and create artificial intelligence applications.

Grace Hopper's Graceful Language Has Come a Long Way

At the beginning of the computer age, banks, government organizations, and insurance companies started creating machine-specific programming languages with the help of COBOL developers.

Due to the high cost and time involved in running and maintaining these applications, a universal solution was needed in place. And Grace Hopper was instrumental in bridging, this gap by creating COBOL. The programming language was universal and capable to run on all business systems.

COBOL's syntax is an output of Grace's reasoning that the syntax of programming languages should be close to spoken languages. That's why COBOL syntax was considered a wordy one by COBOL developers. The era of verbose English-like programming languages has dawned again, starting from Python. Grace's reasoning of yore is now very relevant because the humanization of computing increases the possibilities of unlimited programmatic expressions.

What Differentiates COBOL from Other Programming Languages?

Unlike general-purpose languages, COBOL is best suited for business applications programming. The following are some of the characteristics of business applications that makes COBOL different from other programming languages:

- Business-oriented programming languages should be equipped to manipulate, manage, and declare heterogeneous data. COBOL developers write highly dynamic code. A program file means that one could find a mix of data types and complex data or record structures. Floating-point data types, integers, strings of fixed and variable lengths, etc., are used abundantly and sometimes randomly. The need to map such data models to RDBMS by way of object-relational mapping tools is challenging.

- Only valid decimal data types can do full justice to high-fidelity business applications such as accounting software built by COBOL developers. An entry in a ledger must be correct up to the last digit; likewise, accounting software must accurately record a line item up to the previous digit. Financial data is highly reliant on the accuracy of these numbers.
- Data in the form of record structures that are maintained externally can store large amounts of information. Business applications need to access and manipulate these structures. COBOL developers could easily access and use such data with COBOL.
- Although general-purpose programming languages can resolve the points mentioned above, but the point is that COBOL fulfils all these requirements with its native capabilities for the same. That could be the reason why COBOL developers thrived in an era where large-scale business applications that we're opposed to changes needed to be built and maintained.

The fact of the matter is that billions of lines of COBOL continue to exist. Any attempt to transition the lines of code to newer programming languages hasn't been successful.

Read More: <https://www.integrativesystems.com/cobol-developers/>