

The Development of COBOL

Topic Paper #6

Alex Laird
CS-3210-01

1/30/09

Survey of Programming Languages
Spring 2009

Computer Science, (319) 360-8771

alexdlaird@cedarville.edu

Grading Rubric		Max	Earn
	On Time/Format	1	
	Correct	5	
	Clear	2	
	Concise	2	
	Total	10 pts	

Peer
Reviewer

ABSTRACT

This paper discusses the development of the COBOL programming language.

Keywords

Development, COBOL

1. INTRODUCTION

COBOL, which stands for the Common Business-Oriented Language, is one of the oldest programming languages still in use today. First introduced in 1959, COBOL has been continually developed and most recently updated in 2006 with another release currently pending approval.

2. EARLY HISTORY

On May 28-29, 1959, the first official meeting to plan the development of a common business language was held in the Pentagon; the meeting was sponsored by the Department of Defense. Attendance at the meeting was that of computer people, university representatives, business managers, users, and consultants. It was a well-rounded group to help develop an intuitive and well-rounded language [2].

The specifications, finished in early December of the same year, were largely inspired by the FLOW-MATIC language, often called "the mother of COBOL."

The first compiler was implemented in December 6, 1960. One COBOL program was run on both an RCA computer and a Remington-Rand Univac computer to demonstrate cross-compatibility.

3. DEVELOPMENT OF COBOL

In 1997, 80% of the world's business ran COBOL. It was also reported that 200 billion lines of code of the 310 billion lines that were in use were COBOL; a \$2 trillion dollar investment [3].

Due to its simple syntax, COBOL was readily used. A business manager who didn't have an in depth understanding of programming languages could still proof and understand what was meant by statements in the language. Where a more complex language might say "fruitCount += orangeCount," "ADD ORANGECOUNT TO FRUIT COUNT."

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Alex Laird, Cedarville University, Cedarville, Ohio, 45314

Copyright 2009 Alex Laird

COBOL went through several releases, adding new features each time, including COBOL-68, COBOL-74, COBOL-84, and COBOL 2002. Object-oriented design and many other modern features were implemented in the 2002 release of COBOL.

Upon adding the support for objects in the 2002 release, it was discussed that the name of the language should be changed to poke fun at the C++ programming language by calling the next release of COBOL, "ADD 1 TO COBOL GIVING COBOL" or, more simply, "ADD 1 TO COBOL." Another suggestion was "POSTINCREMENT COBOL BY 1."

The future of COBOL may lie in something called PERCOBOL, a compiler that compiles COBOL applications directly to the Java Execution Environment, allowing COBOL's business model to continue to live on in the portable Java environment [3].

4. DESIGN ISSUES IN COBOL

As is illustrated by the possible naming convection "POSTINCREMENT COBOL BY 1," which is an actual line of COBOL code that could be compiled were the variable COBOL declared, COBOL is quite verbose. There are over 400 reserved words. While the language is very simple to read and interpret, it requires much more effort than many procedural or structural languages to write.

5. CONCLUSIONS

The COBOL language has been around for almost as many years as FORTRAN but is by far more widespread. The simplicity in reading is worth the difficulty in remembering and writing the reserved words. The syntax is simple which essentially makes up for the large amount of reserved words. Due largely to the simplicity of the language and its maintained power throughout the years, COBOL doesn't show any signs of dying off in the foreseeable future.

6. REFERENCES

- [1] Sebesta, Robert W., 2008. Concepts of Programming Languages. Addison-Wesley, Boston. ISBN: 978-0-321-4936
- [2] Sammet, J. E. 1981. The early history of COBOL. In History of Programming Languages I, R. L. Wexelblat, Ed. History of Programming Languages. ACM, New York, NY, 199-243. DOI= <http://doi.acm.org/10.1145/800025.1198367>
- [3] Townsend, Chuck. Future of COBOL. 1998. LegacyJ Corporation. 28 Jan. 2009
<<http://www.legacyj.com/cobol/FutureOfCobol.pdf>>.