

Still another COBOL Crisis and its Impact upon IS Accreditation

Robert F. Roggio, Department of Computer and Information Sciences,
University of North Florida, 4567 St. Johns Bluff Road South, Jacksonville, FL 32224;
Ph: (904) 620-2985. Email: broggio@unf.edu

and

James R. Comer, Department of Computer Science, Texas Christian University,
PO Box 298850, Fort Worth, TX 36129,
Ph: (817) 257-7166, Email: j.comer@tcu.edu

Abstract

Many Information Technology (IT) project managers are seeking Information System/Computer Information System (IS/CIS) graduates/professionals with significant COBOL skills to support billions of lines of legacy applications worldwide. Yet in academe, programs are migrating away from COBOL. This trend may well produce graduates unable to maintain and enhance existing COBOL applications notwithstanding projected new development using COBOL. After examining industry's current and projected dependence on COBOL, concerns in COBOL instruction and its revitalization to include modern topics including object-oriented concepts and practices are presented. The paper concludes with recommendations for two IS/CIS programs - one oriented toward AACSB criteria; a second one oriented toward the new CAC/ABET IS accreditation criteria.

Keywords: COBOL, COBOL's future, traditional COBOL, OO-COBOL, IS/CIS Accreditation

1. COBOL DOMINANCE IN THE CORPORATE IT WORLD

Now that the Y2K transition has concluded, the business community faced with a reassessment (and perhaps retrenchment) of its COBOL dependency. While the corporate world may still be patting themselves on the back in justifiable pride for the successful remediation of billions of lines of operational COBOL code into Y2K compliance, business IT managers and faculty curriculum planners must carefully reconcile the role that COBOL will play for the next ten to fifteen years in their respective worlds. Ed Arranga (Arranga, 2000) cites a Gartner Group estimate that between 150 and 175 billion lines of COBOL are currently in production worldwide and that this base continues to grow. Stern and Stern (Stern and Stern, 2000), authors of one of the most widely used COBOL textbooks estimate there are between 150 and 250 billion lines of production COBOL code. They further assert that new COBOL code is being produced at a rate of approximately five billion lines each year!

2. PREDICTIONS FOR COBOL'S FUTURE

Let us consider a few more facts: the COBOL community invested between \$300 billion and \$600 billion dollars in the Y2K remediation effort. (Arranga and Price, 2000) This exercise revealed that COBOL assets were much more complex than previously assets is imagined, and that corporate dependency upon these also

much higher than previously conjectured. (Bradley, 2000) suggests that 75% of all production transactions on mainframes are done using COBOL programming support, over 60% of all web-access data resides on mainframes, that COBOL mainframes process more than 83% of all transactions worldwide, and that over 95% of finance-insurance data is processed with COBOL. CICS transaction volume grew from 20 billion a day in 1998 to 30 billion a day in 2001. (Ankrum, 2001). Ankrum further cites that this 50% growth has largely been driven by web-based applications that access back-end COBOL applications. "That philosophy of [COBOL] extinction has been replaced with one of extension and inclusion. COBOL applications are, by and large, too critical and too valuable to consider replacing en masse." (Arranga and Price, 2000) If there is but one thing alone that Y2K clearly revealed, it is simply that COBOL is alive, well, and will persist well into the new millennium.

(Carr, 2000) conducted extensive research that provided very illuminating statistics regarding the feelings of business and academic leaders regarding the role of COBOL now and in the future. Approximately 3,000 CIS and IS programs and 5,000 businesses were surveyed regarding the perceived future of COBOL. "In surveying business and academic leaders, the authors found that almost 95% of academic respondents and 90% of IT managers still want IS curricula to offer COBOL instruction. They further discovered that nearly 90% of IT managers want both object-oriented and web-based

features integrated into COBOL instruction in college curricula.” (Carr, 2000)

Another set of related questions can be coalesced into a single question: “Will COBOL continue as a major implementation language for business solutions?” To answer these and related questions, Carr and Kizior’s surveyed business leaders to discover that “over 87% currently develop and maintain code written in COBOL – over 50% of effort solely in maintaining current legacy COBOL applications; 20% of effort developing new applications using COBOL; and 30% of the programming resources expended in a mixture of new application development and maintenance.” (Carr, 2000) That maintenance is a large percentage of effort expended on legacy applications is not terribly surprising. However, that 20% of efforts in COBOL applications are currently expended developing new applications is most revealing and should provide additional insight as to what skill sets IT managers need from future IS graduates

3. PROBLEMS AND TRENDS FOR COBOL IN ACADEMIA

Waning COBOL Importance

Consider the following statistics. The percentage of IS/CIS academic programs teaching COBOL is steadily decreasing, while instruction in languages such as Java and Visual Basic is increasing. According to Hardgrave and Doke (Hardgrave, 2000), COBOL was offered in 90% of undergraduate IS programs in 1995 but has fallen to 53% in 1998, while instruction in C++ has fallen from 60% in 1995 to 53% in 1998. OO COBOL, very slow to gain attention, has risen from 0% in 1995 to 7% in 1998; Java instruction has skyrocketed from 0% in 1995 to 43% in 1998 and instruction in Visual Basic has risen from 59% in 1995 to 70% in 1998.

There are many factors contributing to these figures. Faculty (and students) enjoy teaching and learning the latest technologies. While COBOL 85 has been a significant improvement over earlier standards, COBOL itself is not a glamorous language and is often considered cumbersome. Some feel it lacks “elegance.” The current standard fails to include features commonly available in more modern languages that can be readily used to satisfy current and future industry demands for multi-tiered applications.

Visual Basic and Java Gaining in Importance

GUI interfaces can be readily taught in Visual Basic and Java; networking considerations are facilitated nicely in Java; object-oriented programming encompassing topics such as reusability, component-based software development and much more, are provided in Visual Basic to a degree, more in C++, and in Java extensively. Many features forecast for the new COBOL standard are already available in C++ and Java. Thus, there is little surprise that both faculty and students gravitate toward these technologies, oftentimes at the direct expense of

COBOL, despite its widely recognized strengths in transaction processing systems and its wide appeal to modeling and processing business data.

While some attempts at course modernization have been made, these efforts are presenting very serious challenges to educators. In many cases requirements for courses such as Visual Basic, Internet Programming, Java, e-commerce, client-server computing, and other “newer” courses are replacing some (or all) of the COBOL instruction. In courses where the maximum number of hours is tightly constrained, programs are forced to reduce the role of COBOL in favor of Visual Basic, Java, Internet Programming, and similar courses that appear to many to be responsive to industry demands. As hard decisions are made as to the allocation of language courses to the number of hours available for programming instruction, COBOL continues to lose favor.

Future Trends for COBOL Instruction

Thus, when considering what is taking place within IS/CIS programs nationally, cause for alarm is readily apparent. Instruction in COBOL is becoming less and less prominent in IS/CIS programs. Modern features projected in the new COBOL 2002 standard are not being taught to any extent. Object-oriented design and programming as well as web-based capabilities in COBOL are minimally (if at all) communicated. Faculty interest in teaching COBOL continues to wane, generally in favor of instruction in the more exciting and newer technologies.

Few faculty (15% (Carr, 2000)) have had any formal training or education in the new object-oriented or web-based capabilities of COBOL. This is alarming, since industry seems to be so insistent on this point. Of faculty surveyed, almost 90% cited they would like training in these topics. Of these, 60% indicated that a side effect of receiving this training might be to retard actions in some programs in removing or replacing COBOL from the IS programs. (Carr, 2000)

4. A REVITALIZED COBOL

Modernized COBOL as an Integral Part of IS Program

All evidence points to a definite place for COBOL-based applications in the foreseeable future. IT managers are insistent that COBOL should continue to be an integral part of an IS program and nearly 90% of those interviewed (Carr, 2000) indicated the COBOL instruction should be updated to include object-oriented and web-based features. Further, these same interviewed individuals did not want the traditional topics removed from instruction, but rather felt that instruction in the object-oriented capabilities and other modern features would complement (not replace) the standard structured COBOL typically being taught.

Two Course Sequence is Needed

Using the data from the preceding text, the authors have provided evidence demonstrating that IS/CIS programs need to include at a minimum a two-course sequence in COBOL (one course does not provide the needed expertise). These two courses should provide instruction in the traditional topics including various file organizations and accessing techniques, as these topics are necessary to maintain the huge number of legacy applications. However, COBOL instruction should be enlivened and expanded to include significantly upgraded instruction emphasizing the modern features that are available in the COBOL 2002 Standard (COBOL World, 2002), (Webb, year unknown), and these topics should be integrated into classroom instruction immediately. Topics such as reuse, recursion and dynamic storage allocation must be taught and reinforced with appropriate programming assignments. "Highlight of the new standard is certainly the support of *object-oriented programming*. But beyond this, the new COBOL standard will include a large number of other enhancements, such as support of Boolean and bit data, native binary and floating point data, generalized exception handling, an enhanced CALL mechanism, improved table handling, automatic and based data, user-defined functions, standardized compiler directives and many, many more." (Cobol2000, see references) Web-based features using, for example, COBOL Net Express (by Merant) (Merant Microfocus, see references) requiring students to create COBOL.cgi files on their own personal web servers needs to be presented. Further, coursework must include a discussion of the important object-oriented topics such as encapsulation, inheritance, and polymorphism and how they are implemented in COBOL. Reusability principles must be taught and incorporated into programming assignments. Introducing these and related topics should dramatically stimulate COBOL instruction for both faculty and students. When there is common knowledge that many features and language facilities found in Java Script, Java, Visual Basic, and other more modern technologies can also be accomplished in COBOL – without the need to learn a new syntax – COBOL may once again achieve its place as the best overall language that meets the great functional diversity of business applications.

COBOL should be Threaded throughout the IS Program

However, it is not sufficient to just provide enhanced instruction in the language courses. Rather, IS/CIS curricula (see next section of this paper) must ensure that traditional courses in Systems Analysis, Design and Implementation; (generally a two-course capstone sequence in many IS/CIS programs); affords the students the opportunity, if desired, to develop applications using either the newer technologies by themselves (such as a Visual Basic front end with back end database) or a web-based application based on the COBOL language facilities and accessing a (mainframe) database. Thus, students should be provided a legacy path as well as a

more-modern path through the IS/CIS programs. Teaching a revitalized COBOL course sequence that contains both the traditional topics plus the additional features of the COBOL 2002 standard with special emphasis on the object-oriented topics strengthened through programming assignments at the lower level and team projects at the capstone level will meet the corporate community's concern for maintaining legacy systems while at the same time reaping the benefits of the object-oriented development paradigm.

IT managers overwhelmingly want graduates of IS/CIS programs to have a working knowledge of the OO features of the new standard as well as traditional COBOL topics. While companies are putting browser front ends on existing applications or creating new hybrid applications that tactically leverage legacy system functionality (Hardgrave, 2000), the extensive investments in COBOL suggest that COBOL will survive in an Internet, object-oriented world, as the language continues to evolve to accommodate its new environment. (Glass, 1996)

5. INSTRUCTION AND CCREDITATION ISSUES

COBOL and the Accreditation Landscape

In developing this paper, the authors acknowledge that there are many "IS" programs available. One commonly hears of IT programs, IS programs, MIS programs, IS and CIS programs, within schools of business and IS or CIS programs, and outside schools of business. As it turns out, approximately 50% of IS programs reside outside of schools of business. The authors have seen computer science programs based on COBOL and IS/CIS programs emphasizing C and C++. While one can easily argue the differences between IS and CIS, we have chosen to keep them together because the programs seem more similar than dissimilar.

Given this framework and the preceding text, how should the identified COBOL and modern object-oriented topics be integrated into existing curricula? A two-pronged approach is presented. The first addresses programs in schools of business - in particular, AACSB-accredited schools of business. (AACSB web page, see references) The second approach is to look at IS/CIS programs outside of schools of business. These programs may wish to seek IS accreditation for their undergraduate programs under the new ABET/CAC Criteria. (ABET home page, see references) (AACSB homepage, see references).

It is important to note that the authors are recommending that the COBOL instruction itself is the same in both programs. It is only the accreditation constraints that make the programs different and not necessarily the content of the COBOL courses themselves. Further, the application domains (business for AACSB-accredited schools of business and business or other applied areas for non-AACSB programs) may also differentiate the programs.

IS/CIS Programs in AACSB-Accredited Programs

AACSB-accredited Schools of Business have criteria that emphasize the many disciplines commonly found in Schools of Business (Management, Marketing, Economics, Accounting, Finance, Decision Sciences, Information Systems, and others) and provide a sound basis in these areas. While some of these disciplines are combined in separate departments, such as management and marketing or decision sciences and information systems, others of these disciplines may constitute separate departments. Because satisfaction of AACSB standards provides the graduate with a broad-based foundation in the many 'business' disciplines, which is of vital importance to the graduating business student, there are fewer hours that can be devoted specifically to IS/CIS courses themselves. The number of hours is approximately 24 hours in a 120-hour program. While it is possible to have some computer-related courses taught outside the department (and perhaps not count against this 24), the overall number of hours is still limited by the total number of hours required for the degree.

The suggested program (see Figure 1) assumes that there is an "Introduction to Information Systems" course that is, perhaps, part of the overall general business requirements. By having this course labeled as a general education course or general business course taught outside the department offering the IS program, these instructional hours would not count toward the AACSB permitted hours.

Note in Figure 1 there are two two-course sequences that constitute a Legacy Path and an Object Oriented Path. The first two-course sequence in COBOL is needed to provide the traditional materials necessary for the maintenance and extension of legacy software, such as VSAM files. The two-course sequence is desired by the majority of IT professionals and supports the legacy needs in the corporate community.

The Object-Oriented Path presents a set of courses wherein students are given instruction in object-oriented (object-based) programming and in event-driven programming. COBOL 1 may be the prerequisite to both tracks so that programming principles and problem solving are presented only one time. While OO-COBOL is highly recommended in this second path, courses in the other languages (C++, Java, or others) could provide the student with OO-experience and directly support software development using the OO-paradigm if additional hours are available in the program, and the need for expertise in an additional programming language is desired in order to address other industry needs. It is essential to note, however, that AACSB criteria do not prescribe any specific programming languages. These choices are left to the faculty and designers of the local program. Thus, a revitalized COBOL that contains the traditional topics as well as an emphasis on modern object-oriented topics can be used to satisfy both a Legacy Path and an Object-Oriented Path.

Following successful completion of the legacy and OO paths, students should now be well equipped to continue their instruction by taking a course in Database. While many students typically enroll in a database course with a lesser background, it is unlikely they can fully appreciate the power and flexibility of conducting business using a database system. The prerequisite sequences will provide much needed maturity for students taking Database and pave the way for web-based applications. Lastly, a two-course sequence entitled Systems Analysis, Design and Implementation, or, perhaps, Senior Project 1 and 2, represents the capstone of the program. In this sequence, students use all of their accumulated knowledge to develop an Internet application that includes the commonly-used, multi-tier, web-based architectural approach seen in widespread use today.

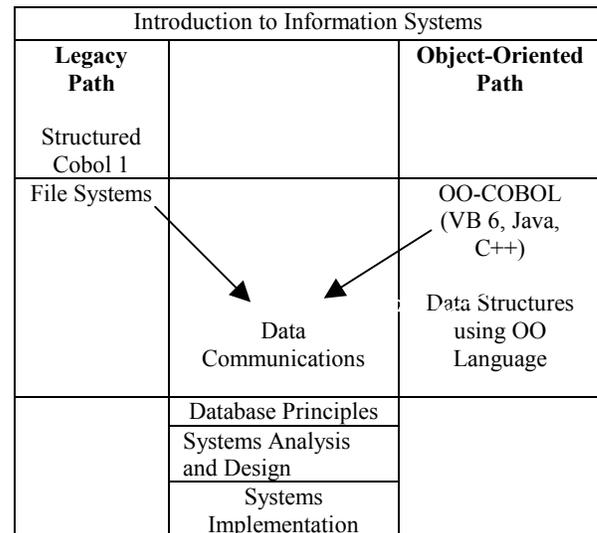


Figure 1. IS/CIS in an AACSB-Accredited Program

Assuming each course in Figure 1 is three semester hours, this schema fits a 24-hour program and is in conformance with AACSB guidelines. Availability of additional hours would make the program more robust. A course in business calculus and discrete mathematics are encouraged. Also, if more hours are available, courses in advanced database processing, or networking would be very beneficial to students.

ABET-Accredited IS Programs

In Fall, 2002, a separate accreditation of IS programs themselves will be available. The Computing Accreditation Commission (CAC) of ABET will begin its first full cycle of accrediting Information Systems programs. (Gorgone, 2000), A number of institutions have already applied to be among the first group to be visited. In breaking new ground, the new IS accreditation presents both opportunities and challenges. (Specifically, one university was visited in the 2001-2002 cycle and approximately eight more will be visited in the 2002-2003 accreditation cycle.)

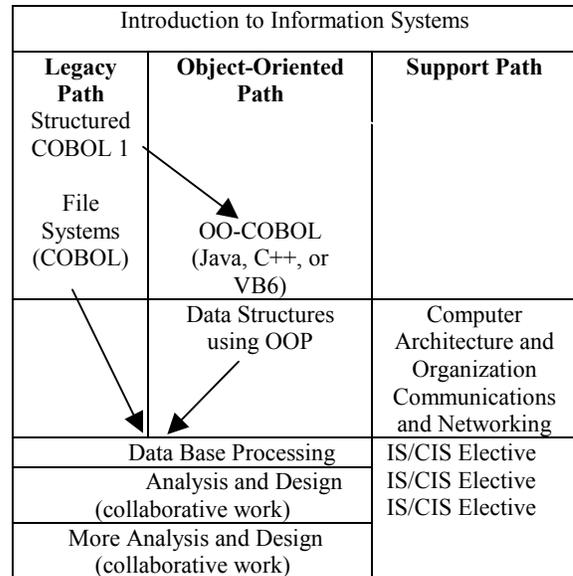
As previously stated, 50% of IS/CIS programs lie outside of schools of business, and it is anticipated that many of these will seek accreditation. In general, these programs require more mathematics courses as well as larger numbers of computing courses. (IS Criteria require at least nine hours of quantitative work including discrete math, statistics, and other mathematics beyond pre-calculus). While this might be an attraction to some, a potential downside is that these programs may be considered to lack some of the fundamental business courses required by IS/CIS programs found in schools of business.

The CAC/ABET Criteria for Information Systems programs has attempted to merge the need for a sound basis in computing skills with the essential broad based information systems courses. ABET requires 30 hours of information systems course work with 15 additional hours of work in an information systems environment (such as business, health sciences, and others). A number of hours in advanced work (building on a core), and specific requirements for collaborative work that is coupled with development and application of both oral and written communication skills encapsulate several of the additional criteria.

The theme of this paper is the compatibility and impacts of what many feel is a new COBOL crisis and the emergence of IS accreditation criteria. We suggest that the immediate needs of IT managers for continued (and significantly upgraded) instruction in COBOL is both necessary and compatible with the new IS Criteria and can be met by revitalized COBOL instruction.

As with AACSB, ABET criteria do not specify a particular programming language. Thus, departmental faculty can decide how best they may satisfy accreditation criteria while responding to present-day industrial needs. Sample programs that include a Legacy Path and an Object Oriented Path might include sequences such as those shown in Figure 2. Note that the COBOL instruction is the same as that suggested in Figure 1.

The major differences between Figure 1 and Figure 2 is the breadth and depth of computer courses that students are required to take. Typically, a minor program of study, (such as a minor in business, mathematics, or some other application area) might also be required. ABET accreditation criteria require 15 hours of coursework in Information Systems environment (which could be business) courses.



* (IS/CIS Electives that provide additional depth Includes: Networks and Distributed Processing, Computer Hardware, Expert Systems and Decision Support, Computer Graphics, Advanced Data Base Processing, others)

Figure 2. IS/CIS in a Non-School of Business Program

6. CONCLUSIONS

The case has been made that COBOL will underpin much of functional business processing for many years to come. IT managers have pointed out the immediate and projected needs for professional COBOL programmers, as they realize that no language can yet match COBOL's file processing capabilities. However, these same managers feel that incorporation of the newer technologies, (e.g., OO programming, COBOL's web-based facilities, and other technologies) must be wedded in order to provide a comprehensive technology base to support critical legacy systems for at least another ten years, while at the same time supporting infrastructure and modern development paradigms to meet business and consumer demands in areas such as web-based applications and e-commerce.

The case for revitalizing COBOL in college curricula has been analyzed and suggestions have been proposed. These suggestions are designed to put the excitement back into COBOL instruction while recognizing that the new capabilities within the COBOL 2002 standard support the need for modern, state-of-the-art software development approaches.

Lastly, a template was offered within which the proposed COBOL sequences might fit into programs currently

accredited by AACSB. A similar template is offered that would seem to satisfy many of the criteria for programs that may be planning to seek Information Systems accreditation under the Computing Accreditation Commission of ABET.

7. REFERENCES

- AACSB web page:
<http://www.aacsb.edu/accreditation.html>
- ABET web page:
<http://www.abet.org/cac>
- Ankrum, Scott, *The COBOL Report*, January 2001
- Arranga, Ed, in Roundtable Discussion, "In COBOL's Defense," *IEEE Software*, 17(2), March/April 2000, pp. 70-72
- Arranga, Ed and Price, W., "Fresh from Y2K: what's Next for COBOL?" *IEEE Software*, 17(2), March/April 2000, pp. 16-20
- Bradley, John, in Roundtable Discussion, "In COBOL's Defense," *IEEE Software*, 17(2), March/April 2000, pp. 70-72
- Carr, Donald and Ronald Kizior, "The Case for Continued Cobol Education," *IEEE Software*, , 17(2), March/April 2000, pp. 33-36.
- "Cobol 2000", Fujitsu Software Corporation,
<http://www.adgtools.com/info/whitepaper/coby2k.pdf>
- COBOL WORLD 2001 – Abstracts Track 06: Cobol 2002
- Coker, Pamela, in Roundtable Discussion, "In COBOL's Defense," *IEEE Software*, 17(2), March/April 2000, pp. 70-72
- Glass, Robert, *The Software Practitioner*, Nov/Dec 1996, p. 8.
- Glass, Robert, "COBOL: Is it Dying or Thriving?" *J. Systems and Software*, 39(3), pp. 197-199.
- Gorgone, John T., Information Systems Accreditation, 2000,
<http://cis.bentley.edu/isa/pages/accreditation.html>
- Hardgrave, Bill C. and E.Reed Doke, "COBOL in an Object-Oriented World," *IEEE Software*, 17(2), March/April 2000, pp. 26-29
- Merant Micro Focus Net Express,
http://www.merant.com/products/microfocus/net_express
- Stern and Stern, *Structured Cobol Programming*, 9th edition, John Wiley, 2000.
- Ulrich, William M., "Remember Cobol? If You Don't, Get Reacquainted," *Computerworld News and Feature Story*, " May 2001
- Webb, Gene, tutorial, "Object-Oriented Cobol,"
http://cobolworld.com/trk6_abstracts.html