

スクールCOBOLユーザーの方のための情報誌

# スクール COBOL Report

Contents

巻頭寄稿

プログラムを書かない  
プログラマー

株式会社 CSK  
黒川利明

ユーザー校事例

秋田商業高等学校  
上杉清博

COBOL最新企画

新製品

「スクールCOBOL2002」の  
ご紹介

「COBOLって時代遅れなの?」



All Rights Reserved, Copyright ©2003, Hitachi, Ltd.

Vol. **5**  
2003.9

## ◇基本情報技術者試験対策に◇



— 経済産業省基本情報技術者試験テキストシリーズ —

1. **ハードウェア・ソフトウェア** 改訂版  
B5判 272p. 定価 2,000円
2. **システムの開発と運用** 改訂版  
B5判 168p. 定価 1,500円
3. **情報化・経営・セキュリティ** 改訂版  
B5判 144p. 定価 1,500円
4. **COBOL** B5判 288p. 定価 1,900円
5. **COBOL問題集** 改訂版  
B5判 256p. 定価 1,900円



- 合格のための総仕上げ  
2003年度版 **速攻基本情報技術者 午前**  
B5判 216p. 定価 2,205円
- 合格のための総仕上げ  
2003年度版 **速攻基本情報技術者 午後**  
B5判 176p. 定価 2,048円

## ◇初級システムアドミニストレータ試験対策に◇

2003年度版  
— 合格のための総仕上げ —  
**速攻シスアド**  
A5判 256p. 定価 2,415円

**シスアドテキスト1** 三訂版  
エンドユーザーコンピューティング  
B5判 224p. 定価 1,880円

**シスアドテキスト2** 三訂版  
システム環境の管理・運用  
B5判 192p. 定価 1,850円

\* 定価はすべて5%税込みです

実教出版株式会社

<http://www.jikkyo.co.jp/>

本社 千代田区五番町5  
TEL 03(3238)7777 FAX 03(3238)7755

大阪 大阪市淀川区宮原5-1-3 新大阪生島ビル  
TEL 06(6397)2400 FAX 06(6397)2402

九州 福岡市博多区博多駅前3-10-24 藤井ビル  
TEL 092(473)1841 FAX 092(471)7529

# プログラムを書かないプログラマー



株式会社CSK

黒川利明 Toshiaki Kurokawa

東芝、ICOT、IBMワトソン研究所、日本IBMで、第5世代コンピュータ、人工知能、プログラム言語などの研究や金融部門のシステム構築に従事。LISP、Prolog、Java、C#、UMLなどの標準化にも参画。現在はCSKフェロー。

著書：ソフトウェアの謎（岩波新書）、Prologのソフトウェア作法（岩波書店）、LISP入門（培風館）、プログラミング言語の仕組み（朝倉書店）、作品としてのプログラム（岩波書店）など多数。

「計算しない計算機」という題名の見出しがある本（作品としてのプログラム、岩波書店、1990）をもう十年以上前に書いたことがある。そのときは、付録として「長生きできるプログラマーの条件」を述べた。その中では、プログラムばかりが人生ではないとそぶいていたものである。

今回は、「プログラムを書かないプログラマー」について書いてみようと思う。白状すると、10年くらい前から書くプログラムがなくなっている。

ところが、最近、ばりばりの若手の仕事を見ている、私どもの若い頃とは大違いで、ほとんどプログラムを書いていないのである。この「スクールCobol Report」の第1巻（2001年9月号）に早稲田大学の寛先生が、プログラミングを教えない日本の「情報」教育を嘆かれ、プログラミングの重要性を認識していない日本という国家の先行きを危惧されている。これは、私のようなロートル・プロ

グラマーにとっても深刻な問題だと理解できる。

しかし、この現場でのプログラミングのこの惨状をご覧になると、寛先生、卒倒されてしまうのではないかと心配になってしまう。しかも、こういう若手には、今や、プロジェクトマネージャ（PM）などというカタカナの肩書きがついていて、もてはやされている。弊社のようなシステムインテグレーターを含め世では、優秀なプロジェクトマネージャがいなくなると、これからの日本のシステム開発は大変なことになると喧しいことさかりで、各社ともPMの早期養成に血道を上げているらしいのである。

さて、「プログラムを書かないプログラマー」を取り上げようと思った背景には、やはり十年くらい前に、どこかのシンポジウムで電気通信大学の竹内さん（当時は、まだNTT通研ではなかったか）とダベっていた際に、「将来のプログラミングでは、

プログラムを書くよりも読むことの方が大事になるのではないかと議論していたことをふと思い出したためだ。そのときには、今流行の言葉(もう古くなったか?)で言うなら、コンポーネントとかソフトウェア部品の再利用などということが頭にあった。実際、全世界で見ると、今なお、何千万というプログラマーが日夜プログラムを書いているのである。

一人一人が新しいプログラムを一から書くよりは、良いプログラムを広めて、それより少しでも良いプログラム(がもしも存在するならば)を作るよう工夫するか、それともその良いプログラムを写す方が良さだろう、ということであった。実際、気の利いたJava™プログラマー(Java™に限る話ではあるまい、きっとすぐれたCobolプログラマーもそうしているに違いない、今城\*1さん、そうじゃありませんか?)の要件は、必要なクラスをどこから見つけてくるかという能力である。

若干脇道に逸れるが、情報処理技術者試験にJava™を科目として取り上げるかどうかという議論をしていた折、そのような、必要なクラスを探し出しては、それを適切に修正して使いこなすという問題が、通常の穴埋め方式のペーパー試験では実現できないということが問題になった。極端な言い方をすれば、そういう能力が、「情報処理技術者」では一切評価されないとしたら、現場では困ったことになる。現実には、問題を解決し、役に立つ技術者をこそ、評価し、養成しなければならないのだから。

この辺りの呼吸は、いつだったか明治大学で「Cプログラミング」のコースを受け持ったときにも、随分気になってしまった。大学でのプログラミングコースは、建前としては、自分でアルゴリズムからプログラム書法を考えて、プログラムを完成するよう期待するのだが、どう見ても、気の利いた連中は、できるクラスメートのプログラムを適当に写しているのである。別のある大学での特別講義では、試験の最中に隣の人の解答を写しているとんでもない学生がいたが、プログラムを「適切に」写すことは、プログラマーの能力の一つである。もちろんそこには、知的所有権の尊重という基本的な礼儀作法が伴わなければならない。さらに、元のプログラムがまともなものかどうかを素早く評価できなければならない。

そう言えば、同じく十年ほど前に、(こういう昔の話ばかりが出てくるのは、やはり馬齢を加えてしまったと言うことなのか)「ソフトウェアの話」という新書を書いては、ソフトウェア文化の到来を夢想したことがあった。そこでは、文芸評論家ならぬプログラム評論家が出て、様々なソフトウェアを俎上に載せて、世上の紙価を高からしめるはずであった。ACMのSIGPLAN Noticesには、いまだにそのようなコーナーがあるのだけれど、邦家の新聞紙上には、IT早わかり風のコーナーはあっても、プログラム評論のコラムは無い。(Wall Street JournalにもFinancial Timesにもないからしかたがないか。)

まじめな話、ソフトウェア評論はまだ無くても

いいが、「システム評論」は、これは何とかしなければいけないかなという気分になっている。理由は、昨年来話題になっている大型システムのトラブルである。

このような「不祥事」についての報道で気になったのが、原因として担当者の「気のゆるみ」や「不注意」などといった精神的なものが上げられていたことである。それが、現場の担当者の判断なのか、記事を書いた新聞記者なり、まとめた編集者の判断なのかは分からない。私にとって問題だったのは、原因が本当に「作業の心構え」の問題なのか、それとも、システム構築運用の方法論なり、ツールなり、技術的な問題なのかということである。

例えば、みずほ銀行での障害問題に関しては、昨年11月の日本信頼性学会創立10周年記念 緊急シンポジウムで、日経BP社コンピュータ第一局編集委員 谷島 宣之さんが、「経営トップの問題」と切り捨てている。それはそうに違いないが、「技術的には」何の問題もなかったのだろうか?

この辺りの議論を聞いていると、第二次世界大戦で日本が勝っても負けても、その原因を「精神」に帰着して、敗戦後は、その「精神」をうっちゃって、「物質(金銭?)」に邁進した悪夢がよみがえってくる。

国内外を問わず、何人かの信頼できる知り合いにこの問題に対する意見を聞いてきたのだが、どういう訳か、私の問いに答えてくれた人は皆、それは、「精神的な」問題ではなく、「技術的な」問題だということに合意している。十年前とは、「システ

ム」の中身も使い方も全く変わってしまったんだもの、昔ながらのやり方が通用するはずはないと。

本稿で、私が提案したいのは、箕先生の「プログラミング教育」と並行して、「プログラムを書かないプログラマー」教育を徹底することである。それが、まともなシステム評論家の育成や、まともなプロジェクトマネージャ(PM)の育成につながるはずなのだ。

現在、文部科学省科学技術政策研究所科学技術動向研究センター(なんて長い名前!)で、システム構築の上流工程について調査を開始している。始めたばかりでがっかり来ているのは、この分野での研究開発や専門家の育成、あるいはベンチャー企業など、日本の状況は何もないに等しいことである。

まるで、神風と大和魂と竹槍に最後を託した(運を天に任して、自分は責任を取らないという)日本軍の末路を見るようで、背筋が寒くなる。原因は、いろいろあるが、システムやプログラミングに関わる技術者としての自己反省を兼ねて言えることは、信頼に足るシステムを安価に短期間に開発する方式やツールをまじめに作らなければならない。作ったシステムの評価をまじめにやっていかなければならないという当然のことを私たちがやってこなかったということである。

と言うわけで、まじめにやろうよというSupervisor Callを起動したつもりなのだが、うまくプロセスを割り当ててもらえるものだろうか。

\*1 今城哲二: COBOLコンソーシアム会長、COBOL JIS規格作成委員会委員長、日立製作所ソフトウェア事業部 勤務

# スクール COBOL REPORT

ユーザー校  
事例

## 秋田市立秋田商業高等学校



秋田市立秋田商業高等学校

商業科主任 **上杉 清博**

### 本校の概要

本校は、大正9年に秋田市制30周年記念事業の一環として設立され、県内の商業高校の中ではもっとも古い歴史を有している。現校舎は昭和53年、秋田市西部にある勝平の丘に新築された。約88,000m<sup>2</sup>の校地には、教室棟をはじめ、さまざまな学習施設と運動施設が建ち並んでいる。

校訓である「感謝」「勤勉」「鍛錬」のもと、学習はもとより、実社会に即応できるよう、しつけ教育を重視し、また、心豊かな人間形成に向け部活動への全員参加を奨励、積極的な活動を援助している。

平成14年度からは、国際的なビジネス感覚を身につけた人材を育成するため、学校全体を「模擬株式会社」にみたと、会社経営を通して、ビジネス教育、国際交流とボランティア活動に関する実践教育に力を入れている。

学校規模は、全校18学級ですべて商業科である。平成14年度入学生までは、2年次より会計コース、流通経済コース、情報コースの3コース制をとっていたが、平成15年度入学生（新教育課程）より総合選択制を導入している。近年の進路状況は、進学者（4年制大学・短期大学・専門学校）が5割強、就職者が5割弱で推移している。

### 実習環境の推移

本校の情報教育は、昭和49年8月に汎用小型コンピュータ（HITAC8150）を導入した時より始まる。COBOLによる言語教育を中心に情報処理教育を行ってきたが、平成6年の教室内LANシステム導入の際に、就職動向を勘案し、ワードプロセッサ、表計算のアプリケーション活用学習を並行して行うようになった。平成10年に教室内LANシステム実習室を増設するとともに、言語またはアプリケーションの学習を生徒の選択にて行うこととした。

平成13年度より年次計画で現在の実習環境整備に取り組み、平成13年に教室内LANシステム実習室を増設（計3実習室）、平成14年9月に校内LANシステムを導入、平成15年4月にそれまでの2実習室をまとめて総合情報処理室に更新、現在プログラミング実習室新設の予定である。校内LANの概要は、別表の通りであり、各クライアントすべてからインターネット接続が可能である。

### COBOL実習の概要

汎用端末、パソコン端末、DOS版とCOBOLを使用してきたが、平成14年の校内LAN構築の際にクライアントOSをすべてWindows® 2000Proで統一

したのに伴い、スクールCOBOL85GUIパックを導入した。GUI環境で操作できるのと、付属の開発マネージャーからほとんどの基本操作がコントロールできるので、生徒には好評である。また、エラー修正、デバッグ作業にかかる時間が短縮できるようになったため、その分発展学習に取り組める生徒が増えた。フローチャートに工夫を加え、ソースの修正 コンパイル 実行とその場で試行できるので、学習に幅が広がっている。また、入力ファイルは“ファイル名.TXT”、印刷ファイルは“OUTFILE”

として学習しているので、学習の初期の段階で、入力装置（ファイル） 入力領域 作業領域 出力領域 出力装置（ファイル）の概念が理解しにくかった生徒も、入力データを変化させることによって、視覚的にデータのトレースができるので、全体的に理解度が高まっている。

主な学習内容は別表の通りであり、学習到達度は全商1級におき、発展的に全経1級や基本情報技術者に挑戦させている。

### 校内LANの概要

ネットワークサーバ
管理サーバ、ファイルサーバ（業務系、学習系）、インターネット用各サーバ（Express5800）
業務系（VLAN）
クライアント（Mate MA24V）6台、（VersaPro VS16S）24台、（Mate MA25V）4台、レーザープリンタ3台
学習系（VLAN）
総合情報処理室（同一教室に合併・分離可能な2系統設置）
制御/ワークファイルサーバ（Express5800）2台、クライアント（Mate MA25V）94台、17インチTFTディスプレイ94台 CAI用センターモニタ17インチTFT48台、ネットワークスキャナ&カラーレーザープリンタ2台、大判プリンタ1台 レーザープリンタ4台、インクジェットプリンタ4台、書画カメラ2台、DVD/VHSレコーダ2台
情報処理室（現在プログラミング実習兼用室）
制御/ワークファイルサーバ（Express5800）1台、クライアント（Mate MA86T）49台、17インチCRTディスプレイ49台 CAI用センターモニタ15インチTFT24台、ネットワークスキャナ&カラーレーザープリンタ1台、レーザープリンタ2台
プログラミング実習室（新設準備中）
無線アクセスポイント（各普通教室および特別教室）
クライアント（VersaPro VS16S）45台

### 学習内容

編集・見出し・印字
合計・平均・条件判定（IF）・最大値・最小値
一定回数の繰り返し・ページコントロール
複合条件・グループータル
条件名条件・多方向への分岐
テーブル（集計・線形探索・二分探索・順位付け）
内部分類
2次元テーブル
ファイル処理（データチェック・照合）
ファイル処理（整列・併合）
ファイル保守
ファイル処理（索引編成ファイル）
ファイル処理（相対編成ファイル）

### 学習の流れ



## 新製品

# 「スクールCOBOL2002」のご紹介

2002年11月制定のCOBOL第4次国際規格[ COBOL2002規格 ]は、オブジェクト指向などのプログラミング言語の進歩を吸収し、開発生産性の向上に役立つ新機能を導入しています。今回ご紹介します「スクールCOBOL2002」では、COBOL2002規格の主な新機能をサポートしました。これらの機能は、プログラミング実習にも有益だと考えます。

「スクールCOBOL85」の後継製品であり、Windows®環境でご利用いただけます。

## 第1章 COBOLの将来性

COBOL( COmmon Business Oriented Language )は、メインフレーム時代から40年近くも使われてきたプログラミング言語です。名前のとおり、会計処理・帳票処理などのビジネスアプリケーションの開発が主たる利用分野です。

そして今でも、COBOLは、コンピュータ利用の大半を占めるビジネスアプリケーションの開発で最もよく使われています。世界では、200万人近くも現役COBOLプログラマーがいます。オープン系COBOLの出荷数も上昇傾向が続いています。企業のeビジネスを支えるWebシステムの構築においても、メインフレーム等で使用実績のあるCOBOL既存資産を活用することの有効性が再認識されているからです。

誤差のない10進演算や多彩な数値文字列操作などのCOBOLの機能は、ビジネスロジックを記述するのに適しています。誰にでも読みやすいプログラム、保守性の高いプログラムが作れます。また、各COBOLベンダが提供する開発ツールも充実しています。

さらに、COBOLおよびCOBOL関連技術は、ITの発展に適応進化しながら、確実にその活躍の場を広げています。例えば、各種のデータベース技術や新しいプログラミングパラダイムに対応してきました。また、先端Web技術であるJava™やXMLにも対応しています。\*

COBOLは、地味だが安心して使えるITコア技術のひとつ

だと言えます。パソコン雑誌などは目新しさを追いがちですが、実際のアプリケーション開発では安心して使える技術こそがキーになります。それが、COBOLです。Java™などの新しいプログラミング言語と適材適所で役割分担しながらも、ビジネスロジックの核は保守性のよいCOBOLで開発しているというのが開発現場の実状です。

COBOL2002規格は、COBOL85規格から17年ぶりの大改訂です。COBOL85規格との互換性を保持しながら、近年のプログラミング言語の進歩を積極的に取り入れています。COBOLが長寿なのは、公的機関による拡張・標準化の長年にわたる努力があるからとも言えます。今後も、国際規格があるCOBOLなら、安心してお使いいただけます。

## 第2章 スクールCOBOL2002の概要

### 2-1 COBOL2002規格対応の機能

スクールCOBOL2002は、オブジェクト指向機能をはじめ、以下に示すCOBOL2002規格の主な新機能をサポートしています。読みやすく保守性のよいプログラムを効率よく作成するのに役立つ機能です。

オブジェクト指向機能	利用者定義のデータ型
利用者定義の関数	共通例外処理
翻訳指令(条件翻訳等)	自由形式の正書法

COBOL2002規格のオブジェクト指向機能は、代表的なオブジェクト指向言語( C++やJava™ )と同等以上です。また、オブジェクト指向の枠組みの中で、既存のCOBOLプログラムをCALL文で呼び出すことも可能です。オブジェクト指向プログラミング機能は、スクールCOBOL85からの構造化プログラミング機能と同様、プログラミング実習でもお役に立ちます。ここでご紹介する「利用者定義のデータ型」と「利用者定義の関数」は、プログラミング実習でも容易に活用できると考えます。

### (1)利用者定義のデータ型

同じデータ構造を複数個宣言する場合、利用者が定義するデータ型を使ってデータ定義部分の行数を削減できます。その結果、重複記述の不一致のようなケアレスミスも回避できます。

利用者定義のデータ型機能を活用：  
TYPEDEF 句で宣言したデータ型を、TYPE 句で利用してデータの実体を定義する。

```
01 日付 IS TYPEDEF.
05 年 PIC 9(4).
05 月 PIC 9(2).
05 日 PIC 9(2).
01 帳票
05 受注日 TYPE 日付
05 納入日 TYPE 日付
```

従来のコーディング：  
同じ構造の受注日と納入日も、繰り返し定義しなければならない。

```
01 帳票
02 受注日
05 年 PIC 9(4) .
05 月 PIC 9(2) .
05 日 PIC 9(2) .
02 納入日
05 年 PIC 9(4) .
05 月 PIC 9(2) .
05 日 PIC 9(2) .
```

### (2)利用者定義の関数

モジュール化は、読みやすいプログラムを作成するための基本技法です。COBOLでは、副プログラムの利用によってモジュール化を実現できます。

COBOL2002規格では、新たに、利用者が関数を定義できるようになりました。定義方法は、副プログラムの定義と同様ですが、PROGRAM-IDがFUNCTION-IDに替わり、関数の戻り値を示すために手続き部見出しにRETURNING指定が必要です。

利用者定義関数は、従来の組込み関数と同じように式中でも記述できます。戻り値のあるケースでは、副プログラム呼び出しに比べて記述量も減り、簡潔なコーディングになります。

最大公約数を求める利用者関数の定義：

```
IDENTIFICATION DIVISION.
FUNCTION-ID. 最大公約数_FUNC.
DATA DIVISION.
LINKAGE SECTION.
01 データ1 PIC S9(9).
01 データ2 PIC S9(9).
01 結果 PIC S9(9).
PROCEDURE DIVISION.
USING データ1 データ2 RETURNING 結果
(結果が、この利用者定義関数の戻り値)
...
【最大公約数を求める処理】
END FUNCTION 最大公約数_FUNC
```

最小公倍数を求めるプログラムで  
利用者定義関数(最大公約数\_FUNC)を参照する：

```
01 整数1 PIC S9(9).
01 整数2 PIC S9(9).
01 最小公倍数 PIC S9(9).
COMPUTE 最小公倍数
= 整数1 * 整数2 / FUNCTION 最大公約数_FUNC(整数1 整数2)
```

最小公倍数を求めるプログラムから、  
副プログラム(最大公約数\_SUB)を呼び出す：

```
01 整数1 PIC S9(9).
01 整数2 PIC S9(9).
01 最大公約数 PIC S9(9).
01 最小公倍数 PIC S9(9).
CALL 最大公約数 SUB USING 整数1 整数2 RETURNING 最大公約数
COMPUTE 最小公倍数 = 整数1 * 整数2 / 最大公約数
```

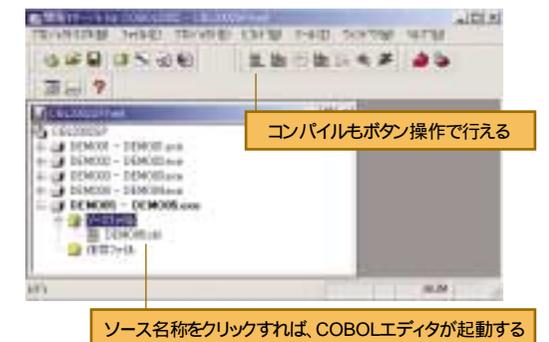
### 2-2 スクールCOBOL2002の開発環境

COBOL2002規格対応の最適化コンパイラやCOBOLエディタを装備しています。\*2 その他、スクールCOBOL85と同様に操作性のよい開発ツールやオンラインマニュアルも用意しています。

以下では、開発マネージャ、COBOLエディタおよびテストデバッガを説明します。

### (1)開発マネージャ

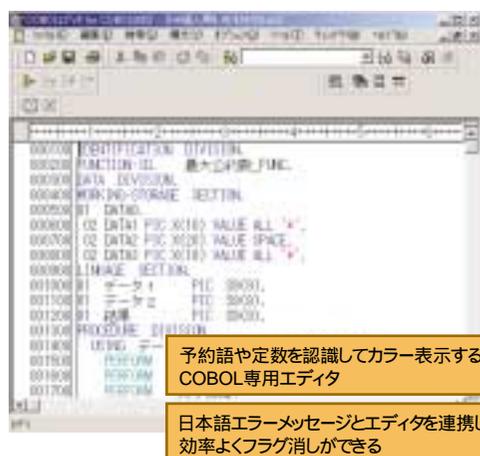
すべての開発ツールは、開発マネージャを中心に統合され



ています。COBOLソースなどの開発リソースを管理して依存関係を表示するとともに、それに基づくコンパイルの自動化機能を提供しています。

### (2) COBOLエディタ

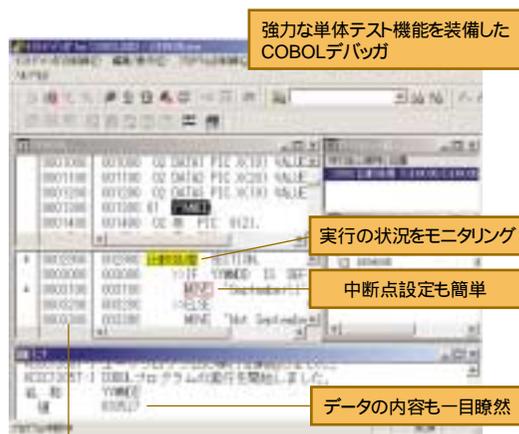
予約語・定数の色分け表示や一連番号の自動付与、構文チェック機能など、COBOLの文法に対応した各種機能が利用可能です。また、COBOLの全構文を網羅したテンプレートを標準装備しているため、デバッグ、翻訳時のエラーとの連携も可能で、効率よくソース編集ができます。



予約語や定数を認識してカラー表示するCOBOL専用エディタ  
日本語エラーメッセージとエディタを連携して効率よくフラグ消しができる

### (3) テストデバッグ

シミュレーション(メイン/サブプログラム、ファイル、データベースなど)をはじめとする強力な単体テスト機能を用意しています。



行番号をクリックすればCOBOLエディタが起動する(デバッグ中のプログラムの修正も容易)

またバッチ実行によって、大量のテストデータを効率よく実行できます。データ表示・代入、常時監視機能によってデバッグ効率が向上しプログラム不良の早期発見に貢献します。

## 第3章 おわりに

初級プログラミング教育では、プログラム書法の躰が大切です。コーディング規則が厳格なCOBOL言語は、その目的に適しています。さらに、スクールCOBOL2002は、オブジェクト指向機能など、プログラミング教育に効果的な言語要素を新たに取り込んでいます。基礎的な言語要素の理解は、他のプログラミング言語や最新ITの学習にも有益です。

まずはCOBOLプログラミングを十分に学習し、その後にはCやJava™などのプログラミング言語、あるいはWeb技術にもチャレンジされることをお勧めします。

ワープロソフトや表計算プログラムなどの利用技術の修得も欠かせませんが、「プログラム作成こそがコンピュータを理解し、コンピュータ技術を修得する中核である。プログラム作成の訓練なしにはコンピュータの理解はない…」という西村先生のご指摘は、正鵠を射ていると思います。\*3

現在、各企業は膨大なCOBOLプログラム資産を保有しています。4年前に騒がれた「コンピュータの2000年問題」の際の見直し作業では、10年以上も前に作成されたCOBOLプログラムもたくさんあったそうです。しかも、今日でも立派に働いているのです。

使用実績のあるCOBOLプログラムは、信頼できる経営資産です。適切に保守し、有効に活用すべきです。そのための若手技術者育成は緊要の課題です。

最後に改めて、COBOLによるプログラミング教育の実施・継続をお願いいたします。今回ご紹介しましたスクールCOBOL2002が、少しでもそのお役に立てば幸いです。

\*1:Java™とCOBOLの連携機能ご紹介(本誌Vol.3)  
COBOLにおけるXML連携のご紹介(本誌Vol.2)  
\*2:現在、情報処理試験などではCOBOL85規格がベースです。COBOL85規格言語仕様の範囲内に制限するためのコンパイルオプションも用意しています。  
\*3:巻頭寄稿(本誌Vol.4 西村 忍彦)

・Java及びすべてのJava関連の商標及びロゴは、米国及びその他の国における米国Sun Microsystems, Inc.の商標または登録商標です。  
・Windowsは、米国及びその他の国における米国Microsoft Corp.の登録商標です。

# COBOLって時代遅れなの?

「時代遅れなので、COBOL教育はやめました。」  
最近、顧客リサーチにおいて、こんな回答が目につくようになってきました。「西暦2000年問題」以降、COBOLというプログラミング言語は「時代遅れ」のイメージが付きまっています。  
でも、これって本当なのでしょうか?

Q:もう使われていないから時代遅れなのでは?

A:いいえ。きっと、あなたもよく使ってますよ。

『2005年までに新たに作られるすべてのアプリケーションのうち、15%(50億行)がCOBOLで開発される。』\*1という予測をご存知でしょうか。COBOLが「時代遅れ」の言語であるなら、新たに作られるアプリケーションに使うことは、ちょっとためらわれると思います。世の中ではそんなことを気にしない無頓着な開発が行われているのでしょうか?当然、違いますよね。

COBOLは今でもプログラム開発の中核を占める言語なのです。

『すべてのビジネスデータの75%はCOBOLで処理される。』\*1  
『世界中で使用されているCOBOLは、ソースコードにして1800億~2000億行である。』\*1という情報からも、COBOLの重要度は明らかです。

あなたは気づかないでしょうが、COBOLは日常生活のあちこちに使われています。多量のお金を扱う必要がある場所、大量の物品管理や帳票を作成する場所など、「大量のデータを安全かつ正確に処理する」ことを要求される場所で休むことなく使われています。

きっと、あなたもよく使ってますよ。

Q:言語仕様が古臭いから時代遅れなのでは?

A:いいえ。COBOLは時代に合わせて変化していますよ。

COBOLの最初の言語仕様は1960年に決められました。40年前のプログラム言語と言われるとさすがに古臭く思われるでしょう。(なんてたって、集積回路を使っていないコンピュータの時代の話ですから。)でも、考えてみてください。こんな長い間、第一線で働いているプログラミング言語が他にありますか?使い続けられているということには、何か理由があるはずですよ。

事務処理用共通プログラミング言語(Common Business-Oriented Language)の名前の通り、COBOLは、史上初の共通言語仕様として設計されました。その共通言語という設計思想は、「古い規格で書かれていたプログラム」や「異なるプラットフォームで動いていたプログラム」などの既存ソースが使える高い互換性として、今も生き続けています。高い互換性は、既存ソースが持つ、「何年も動かして得られた信頼性」を活かせることに繋がっているのです。

\*1: <http://www.cobolwebler.com/cobolfacts.htm> (英文)より \*2: <http://www.eweek.com/article2/0,3959,1098146,00.asp> (英文)より これらのURLは2003.08.05現在、参照できる情報です。

しかも、COBOLは高い互換性に加えて、40年間立ち止まっていたわけではありません。『産業界の言語』として、時代の変化に合わせて、規格の改訂を行ってきました。2002年にもISO(International Organization for Standardization)で第4次規格が制定されたばかりです。

この規格(COBOL 2002規格)では、クラスやメンソッドといった『オブジェクト指向』など、最先端のプログラミング技術に対応した言語仕様も含まれています。数年の内にJIS規格としても制定されることですよ。

COBOLは時代に合わせて変化していますよ。

Q:PCを使った運用環境に対応していないから時代遅れなのでは?

A:いいえ。ようやく、PCでCOBOLが使える時代になったんですよ。

「COBOLをPC(パーソナルコンピュータ)上で使いたい!!」と思っているユーザーは、昔からたくさんいました。しかし、従来はPCでは処理能力的に基幹システムなどを構築するのは難しく、COBOLが十分活躍できるような環境ではなかったため、開発環境など、限定的に使うのみでした。近年、PCの能力向上により、ようやくCOBOLも活躍できるような環境になってきました。

『うそでしょ?PCのプログラムをわざわざCOBOLでなんか作らないよ。』って言いましたか?

そうですね。わざわざ作らなくても、メインフレームのソースを使えばいいんですから。COBOLの高い互換性により、再利用は容易で、しかも信頼性の高いソースです。使わない手はありませんよね。

また、分散オブジェクト技術の進歩により、メインフレームで稼働しているCOBOLアプリケーションそのものを、ネットワークでつなげて利用できるようになってきました。WebネットワークでのCOBOLアプリケーションの利用が拡大していくことは、

『次の10年で、最も高給を得られるプログラマは「インターネットを知っているCOBOLプログラマ」である。』\*2  
という予測からも裏付けされます。

すなわち、それは間接的なPCからのCOBOL利用の増加を意味しているのです。

ようやく、PCでCOBOLが使える時代になったんですよ。

この記事を読んでどうお考えになりましたか?

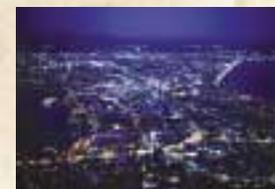
『COBOLって時代遅れなの?』

答えはあなたが決めてください。

## 編集後記

今回のレポートは、2002年11月に制定されたCOBOL第4次国際規格(ISO/IEC 1989:2002)に対応した「スクールCOBOL2002」を中心に紹介しています。一度、お手にして「スクールCOBOL2002」を体験していただきたいとお思います。また、今回は学校の先生方から寄せられる「COBOLは時代遅れなので!」のご意見や疑問

を正面から受け止め、実際にCOBOLがどのように使われているか、今後も使われ続ける言語である内容を掲載しているホームページがあり、皆様に紹介しています。少しでも「COBOL教育」の重要性を再認識していただければと思います。お読みになった皆様の感想は、いかがでしょうか…。



実教出版株式会社 東京都千代田区五番町5 Tel.(03)3238-7777 <http://www.jikkyo.co.jp/>  
発行元: 株式会社 日立製作所 ソフトウェア事業部 販売推進部 東京都品川区南大井6-26-2 大森ベルポートB館  
Tel.(03)5471-2592 <http://www.hitachi.co.jp/soft/>

掲載されている会社名、製品名は、それぞれ各社の商標または登録商標です。

日立スクールCOBOL85  
リピータ特別価格のご案内

既に「日立スクールCOBOL85スクールパック」をご導入されている場合、最新の「日立スクールCOBOL2002スクールパック」をリピータ特別価格でご提供させていただきます。

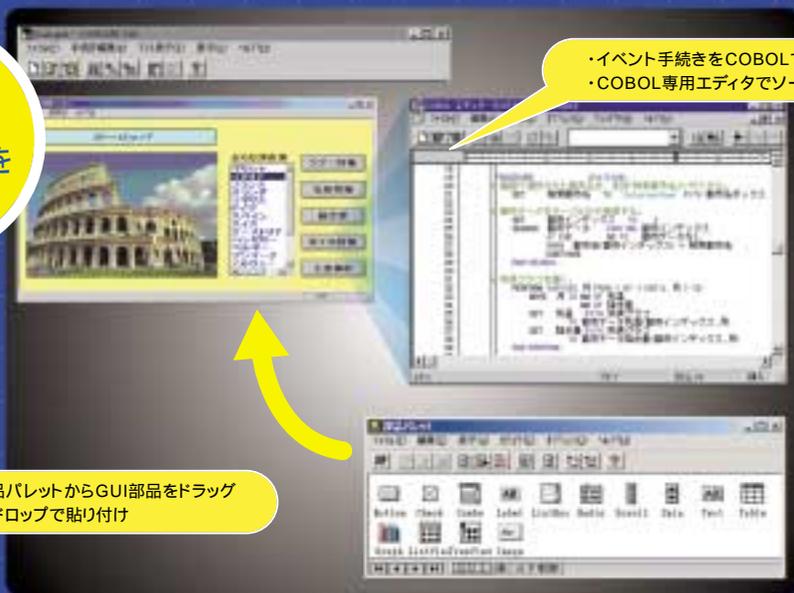
情報処理教育用COBOLコンパイラの決定版  
Windows®対応  
スクールCOBOL2002シリーズ

**HITACHI**  
Inspire the Next

使いなれた  
COBOLで  
GUI構築技法を  
効率よく学習

- ・イベント手続きをCOBOLで記述
- ・COBOL専用エディタでソース編集

- ・部品パレットからGUI部品をドラッグ  
&ドロップで貼り付け



スクールCOBOL85 GUI構築パック画面

## 情報処理教育に最適。

高信頼性と豊富な実績のある「スクールCOBOL85」の後継製品、  
「スクールCOBOL2002シリーズ」です。

操作性のよいWindows®に完全対応した高性能コンパイラと、今までにない効率的な実習環境を提供。

### スクールCOBOL2002

最新の第4次国際規格に対応しています。  
日本語メッセージを表示するなど、エラー修正も容易です。  
エラーメッセージをクリックすると、自動的にソースプログラムの該当行にカーソルが位置付きます。  
プログラムを実行しながら、どの命令を実行しているか画面に表示するデバッグツールを内蔵しているため、デバッグ作業も容易です。

### スクールCOBOL85GUI構築パック

従来のCOBOL言語仕様に若干の言語仕様を加えるだけで、イベント駆動型GUIアプリケーションの学習ができます。  
スクールCOBOL85の機能は、そのまま使えます。  
部品パレットからGUI部品をドラッグ&ドロップするだけでGUI画面が作成できます。  
GUIで発生するイベントに対する手続きをCOBOLで記述できます。

GUI: Graphical User Interface

### 価格(税別)

スクールCOBOL2002		スクールCOBOL85 GUI構築パック	
1セット	58,000円	1セット	88,000円
21セット	940,000円	21セット	1,408,000円
41セット	1,725,000円	41セット	2,584,000円

対応OS: Windows®2000、WindowsNT®4.0、Windows® Millennium Edition、Windows®98、Windows®XP  
Microsoft、Windows、WindowsNTは、米国およびその他の国における米国Microsoft Corp.の登録商標です。

販売 実教出版株式会社 〒102-8377 東京都千代田区五番町5  
本社/TEL(03)3238-7777 FAX(03)3238-7755 大阪/TEL(06)6397-2400 FAX(06)6397-2402 九州/TEL(092)473-1841 FAX(092)471-7529  
開発元 株式会社 日立製作所 ソフトウェア事業部