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
プログラミングは面白い
拓殖大学 工学部 情報工学科 教授 工学博士
岩澤 京子

●「COBOL 最新企画」
COBOLとクラウドコンピューティング
ー 計算機の今昔・50年を越える
COBOLと最新流行のクラウドと… ー

●ユーザー校事例
福島県立喜多方桐桜高等学校
渡部 和博

●企業におけるCOBOL活用
ジャパネットたかた殿
再構築事例

●スクールCOBOL Reportの歴史

Vol.14
2011.3

情報教育をサポートするじっきょうのテキスト

◇経済産業省基本情報技術者試験対策に◇

基本情報技術者テキスト

Ⅰ. ハードウェア・ソフトウェア
B5判 304p. 定価2,100円

Ⅱ. データベースとアルゴリズム
B5判 184p. 定価 1,890円

Ⅲ. システム開発とその運用
B5判 168p. 定価 1,680円

Ⅳ. ネットワーク技術
B5判 144p. 定価 1,680円

Ⅴ. セキュリティと標準化・情報化と経営
B5判 200p. 定価 1,890円

基本情報技術者学習テキスト
— 基本情報技術者試験シラバス準拠

1. テクノロジ系 B5判 264p. 定価 1,995円

2. マネジメント・ストラテジ系
B5判 208p. 定価 1,785円

◇経済産業省ITパスポート試験対策に◇

ITパスポート学習テキスト —ITパスポート試験シラバス準拠—
B5判 216p. 定価 1,890円

平成23年度版 ITパスポート試験問題集 B5判 344p. 定価 1,680円

ファーストステップ ITパスポート 合格テキスト＆問題集
B5判 256p. 定価 1,200円

実教出版株式会社
http://www.jikkyo.co.jp/
※Webに最新情報を掲載

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3-2-1 日本生命博多駅前ビル
TEL 092(473)1841 FAX 092(471)7529

プログラミングは面白い

拓殖大学 工学部情報工学科 教授 工学博士
岩澤京子 *Kyoko Iwasawa*
1981　東京農工大学工学部数理情報工学科卒業
1981-1992　(株)日立製作所中央研究所勤務
1992-1998　東京農工大学工学部電子情報工学科勤務
1998-現在　拓殖大学工学部情報工学科勤務
プログラム解析やコンパイラの最適化、並列化処理に興味を持つ。
主な著書・訳書
「コンパイラとバーチャルマシン」(オーム社)、「データ構造とCプログラム」(丸善)

最近の私たちの生活や仕事を見渡すと、コンピュータ無しでは何もできない。複雑な処理に用いるばかりでなく、電卓が普及して私たちは計算能力が衰え、ワープロが普及して漢字を書けなくなり、今インターネットのサーチエンジンによって記憶力や調査能力が覚束なくなっているのではないかと、とそんな話をよく耳にする。そのうち、ＸＸＸが普及したので、プログラムが作れなくなった、ということになる可能性もあるかもしれない。

もっとも、ハードウェアの開発手法がCADなどを中心として飛躍的に進化したのに比べて、ソフトウェアの開発手法は、約70年前にコンピュータの先駆者たちが行ってきたのとあまり変わっていないようにも思える。そこで、当時の数少ない天才だけが行っていたのと同様の方法で、私たち凡人がプログラムを作らなければならないのだから、大変である。見方を変えれば、昔は天才にしか許されなかったことを、今は私たちもトライすることができるのである。わくわくと楽しむべきだと思う。

1．創世記のプログラム

広く使われている、フォンノイマンアーキテクチャ（プログラム内蔵方式stored program）の本格的な最初のコンピュータは、ケンブリッジ大学のWilkes教授のチームで開発したEDSACだと言われている。[1][2]1949年のことである。私たちが学生のころ、このコンピュータを例に取り、プログラム内蔵方式がどういうことか、電卓と何が違うのか、ということを教えられた。

- （名前のとおり、）実行前にすべてのプログラムが計算機のメモリに格納されている。
- プログラムカウンタは自動的に一つずつ進み、逐次的に命令を実行する。
- 命令もデータも同じ、０と１の並びで構成される。プログラムカウンタが指せば、それは命令であり、ほかの命令により演算の対象となれば２進数のデータである。

1ワード17bitで、1024ワードのメモリと、アキュミレータという名前のレジスタとアドレス修飾用のレジスタの２つのレジスタから構成される。

今、数万円購入できるPCと比べると非常に貧弱であるが、驚くほどの成果をあげた。量的な制約が多い分、プログラムは洗練され凝ったものとなった。おそらく最初のOSであるイニシャルオーダーは、芸術的である。わずか40命令弱（プログラムの実行文ではない！）で、入力されたプログラムをデコードして、指定されたメモリアドレスに格納する。このプログラムは、実行しながら自分の一部の命令を書き換えてしまう。読むには面白いが、開発できるのは天才だけであろう。

しかも残念なことに、こんな離れ業は、アセンブラ（命令語）でコーディングする人にのみ許された特権である。C、Java、Fortran、Cobol、という私たちになじみのプログラミング言語（コンパイラを用いるコンパイル言語）では、まずできない。

命令語によるプログラミングから、記述性も可読性も優れている多くのプログラミング言語が提案され、広く使われている。手続き型の言語では、フォンノイマン型計算機のアーキテクチャ（命令語）と対応をつ

けられている。基本は、制御を指示する「逐次・条件分岐・繰返し」であり、データを扱う算術演算や論理演算と代入処理、そしてこれらの処理のまとめた関数があるだけである。それだけなのに、驚くほどなんでもできるのは、アルゴリズムとデータ構造のおかげかもしれない。

PascalをつくったWirthの有名な本のタイトル「アルゴリズム＋データ構造＝プログラム」にもあるように、プログラムはデータ構造とアルゴリズムから成り立つ。処理手順を示すアルゴリズムに着目して進化してきたのが構造化プログラミングで、データの方に着目して、データとそのアクセスモジュールをまとめたのがオブジェクト指向プログラミングともいえる。

2．10倍遅かったらプログラムではない？

「10倍遅かったらプログラムではないですよ。」これは、あるプログラムの課題で「再提出」を食らった学生の、「正しい結果を得られるのに」という抗議にたいする大教授の答えである。EDSACのイニシャルオーダーのプログラムのように複雑で洗練されたものばかりでなく、単純な例でもいくらでも面白いアルゴリズムがたくさんある。わかりやすい例をいくつかあげよう。

■ 素数の判定

暗号に用いる素数を乱数から探し出すプログラムで、与えられた自然数が素数か否かを判断する処理を考える。引数で与えられた自然数nが、他のいかなる自然数でも割り切れないことを確かめる必要がある。単純に考えれば、2からnまでの数で割ってあまりの有無を調べればよさそうである。そのとおりにプログラムを作成した前述の学生が、これは「プログラムではない」と言われたわけである。そこで、最少の判定回数となるようにアルゴリズムを考える必要がでてくる。

2で割り切れないものは、4でも8でも割り切れないし、3で割り切れないものは、6や9でも割り切れない。そもそもnまで調べる必要はない。ルートnまでで十分である。そこでプログラム開発の困難さ、バグが入り込みやすくなったり、保守が困難になる複雑さ、実効効率とのトレードオフになる。

図1のプログラムで、ループの繰り返し回数をn回からsqrt(n)回（ルートn）にすることによって、判定回数は1/nになる。nが大きくなるほど速度向上の効果が大きくなる。nが100を越えると大教授のいう「プログラムではない」領域に入ってしまう。一方、すべての倍数をはねて、ルートn以下の素数だけで判定するようなアルゴリズムは、プログラムが複雑になり、よほど工夫をしないと、その処理のためにメモリや時間を使うことになる。

一般に、プログラムが複雑にならず、速度効果が認められるところで手を打つと図1のようなプログラムになると考えられる。2の倍数だけは、ループの初期値と増分値を変更することによって無駄な判定を避けることができる。この結果判定の回数（素数ではない場合のループの繰返し回数）は、1/(2*n)になる。プログラムの複雑さを問わずに、最短のプログラムを作ろうとすると、難しさは、ここから急に増加する。アプリケーションの性質よっては必要になろう。

```
int Sosu (int n) {
    int i, flg=n%2 ;
    for (i=3; (i <= sqrt(n) ) && (flg ! = 0) ; i = i+2) {
        flg = n%i;
    }
    return flg;   /*  n が素数ではない場合は 0 を返す  */
}
```

図1　そこそこの素数判定の関数

■ ホーナーの方法

ホーナーの方法は、n次の多項式を計算するアルゴリズムである。これは、n進数の変換や、文字列と数値の変換などにも用いられたりする。

$f(x)=a_0+a_1*x+a_2*x^2+\cdots+a_{n-1}*x^{n-1}+a_n*x^n$

これをそのまま計算すると、乗算の回数がn*(n+1)/2回で、加算の回数がn回である。

同じ計算を

$f(x)=a_0+x*(a_1+x*(a_2+\cdots+x*(a_{n-1}+x*a_n)\cdots))$

拓殖大学 工学部情報工学科 教授 工学博士
岩澤 京子

というように、括弧の内側から順に計算すると、乗算も加算もn回である。n^2のオーダーの計算を、オーダーnの計算にすることができるわけである。図２のようなプログラムになる。

```
double fx (double a[ ], double x, int N) {
    double y=a[N] ;
    int i;
    for (i=N-1; i>=0; i--) {
        y=y*x+a[i];
    }
    return y;
}
```

図２　ホーナーの方法によるｎ次多項式の計算

■べき乗の計算

a^nを計算するのも、必ずしもn-1回乗算を必要としない。たとえば、100乗の場合は、$((((a^2\times a)^2)^2\times a)^2)^2$ と計算すると、99回乗算を行わなくても、8回の乗算で結果が得られるはずである。図３のように二進数に変換して、上位桁から対応させて計算する方法である。しかしこれを実装するプログラムは、99回繰り返すループよりずっと複雑になる。そこで、トレードオフである。対象とするｎの範囲がいくつであることが予想できるのか、どれくらい速いことが要求されているのか、オーバーフローの可能性はどうなのか、開発時間はどれくらい許されているのか、といったことから、アルゴリズムを選ぶことになる。

$f = a^{100}$の計算手順

$100_{10} = 1100100_{2}$

$f = 1$　初期値
$f = f \times f \times a = a^{1}$
$f = f \times f \times a = a^{3}$
$f = f \times f = a^{6}$
$f = f \times f = a^{12}$
$f = f \times f \times a = a^{25}$
$f = f \times f = a^{50}$
$f = f \times f = a^{100}$

図３　べき乗の計算

人によっていろいろな見解があると思うが、コンピュータを使う一番の利点は、いつの時代でも、はやり速さである。余計な仕事をさせてその利点を相殺させるプログラマは一流とはいえまい。プログラマにとって、速さを左右するのはアルゴリズムということになる。状況によっては、計算機アーキテクチャやそのハードウェア特性を有効に使うことができるか否かが重要になる。コンパイラにしかできない最適化、プログラマにしかできない最適化の両方がある。そして、プロとしては、開発の容易さ、保守性、可読性、拡張性、などとのトレードオフを考慮して、その場で最適なものを作る必要がでてくる。非ノイマンと言われるさまざまな種類の並列処理や並行処理をいかに効率よく使うかが今後の課題ではないかと思う。

３．「銀の銃弾などない」からプログラミングは面白い

ソフトウェアという怪物に対する銀の銃弾などない。ほとんどの場合に、いわゆる唯一無二の正解がない、ということには多くの同意が得られると思う。でも、だからこそプログラムを作るのは面白い。「ソフトウェアの構築には、抽象的なソフトウェアの実態を構成する複雑な概念構造体を作り上げること、そうした抽象的事実をプログラミング言語で表現し、それをメモリスペースとスピードの制約内で機械語にマッピングすることが含まれている。」[3]　そして、凡人は凡人なりに考えれば考えただけの成果が得られるのではないかと思う。

天然資源の乏しい日本でも、プログラム開発は、人間の知恵で切り開いていくことができる分野の一つだと思う。かまえずに、プログラミングを楽しんでほしい。ある天才ゲームプログラマは、「ゲームばかりやっている人間は決してゲームを作る側にはなれない」と言っていた。さらに私は付け加えたいと思う。ゲームを作る方がゲームをやるより、ずっと大変だけどずっと面白い。これからの若い世代と、彼らを教育する方々に大いに期待したい。

＜参考文献＞
[1] Maurice V. Wilkes and David J. Wheeler and Stanley Gill,The Preparation of Programs for an Electronic Digital Computer, ADDISON WESLEY PRESS INC（1951）
[2] https//www.cl.cam.ac.uk/conference/EDSAC99
[3] Frederick P. Brooks, Jr.著, 滝沢, 牧野, 富澤訳, 人月の神話, Addison-Wesley Japan

COBOL最新企画

COBOLとクラウドコンピューティング

－計算機の今昔・50年を越えるCOBOLと最新流行のクラウドと・・・－

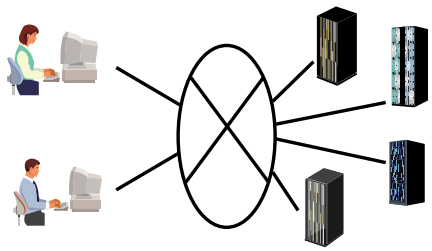
はじめに

ドッグイヤーと言われるITの世界において、齢50年を超えたCOBOLは「レガシー」と呼ばれ「古臭い」というイメージが定着してしまった。一方、この対極として、最近のIT業界で最も頻繁に話題となるのが「クラウドコンピューティング」（以下「クラウド」）である。今回は、このクラウドについてCOBOL進化の歴史も振り返りつつ解説する。

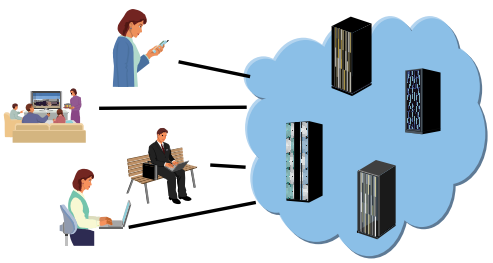
クラウドコンピューティングとは

日本では、システム構成図などにおいてインターネットを⊗の記号で表すことが多いが、米国では雲の形で表すことが習慣だったという。クラウドコンピューティングとはITサービスがインターネットを経由して提供されることから、さながら、様々なIT資源が雲の中にありサービスを提供されるように見えることから名づけられた名称である。

“Cloud computing”という用語は2006年の「検索エンジン戦略会議」（Search Engine Strategies Conference）においてEric Emerson Schmidt氏が提唱したのが最初だとされている。ちなみに、GoogleのCEOとして著名な同氏だが、プログラミング言語の分野ではUnixの字句解析器生成プログラム「lex」の開発者の１人として知られている。



ネットワークを介したサーバ利用



全ては“雲”(インターネット)の中から
サービスとして提供されるクラウド

COBOLのように“規格”が定まったものと異なり、色々な会社がいろいろなものを「クラウド」と称している。COBOLの標準化でもおなじみのNIST（National Institute of Standards and Technology：米国国立標準技術研究所）は、「The NIST Definition of Cloud Computing」[注1]において、クラウドの本質的特質（Essential Characteristics）として以下の５点を挙げている。

#	特　性	内　　容
1	On-demand self-service	必要とする時にセルフサービスで利用できること
2	Broad network access	ネットワーク経由でどこからでも使えること
3	Resource pooling	利用者共用の資源プールから動的に資源を利用できること
4	Rapid elasticity	即座にかつ弾力的に資源を使えること
5	Measured Service	使用する資源量に応じてサービスが使えること

ネットワーク越にデータセンタにある計算機資産を必要な時に利用できるのが「クラウド」の要諦だといえる。従来のホスティング、ハウジングと異なるのは、仮想化技術の発達により、1つの物理サーバを複数の利用者がシェアして利用できる点にある。（マルチテナント性）

クラウドの社会へのインパクトは、しばしば電気や水道などの社会ユーティリティ（Social Utility）に例えられる。電力が社会ユーティリティ化される前は、工場は自前で発電設備を保持しなければならず、そのコストも大きかった。巨大な発電設備、給配電網をもつ電力会社の登場により、電力はユーティリティ化され、利用者は料金を払うだけで電力を利用することが可能となった。ITも同様に、クラウドにより演算装置、ディスク装置、通信機器などを自前で維持することなく、ネットワーク経由で利用可能となるユーティリティ化されるというものである。[注2]多くの企業がクラウドを検討しているのは、この「変革」に期待しているためである。

この道はいつか来た道　ーCOBOLの発達を振返るー

「計算機センタにある巨大な計算機を回線経由で使う」というクラウドの利用方法は、PCやワークステーション以前の世代のCOBOLプログラマにとっては、むしろ“懐かしい”使い方といえる。

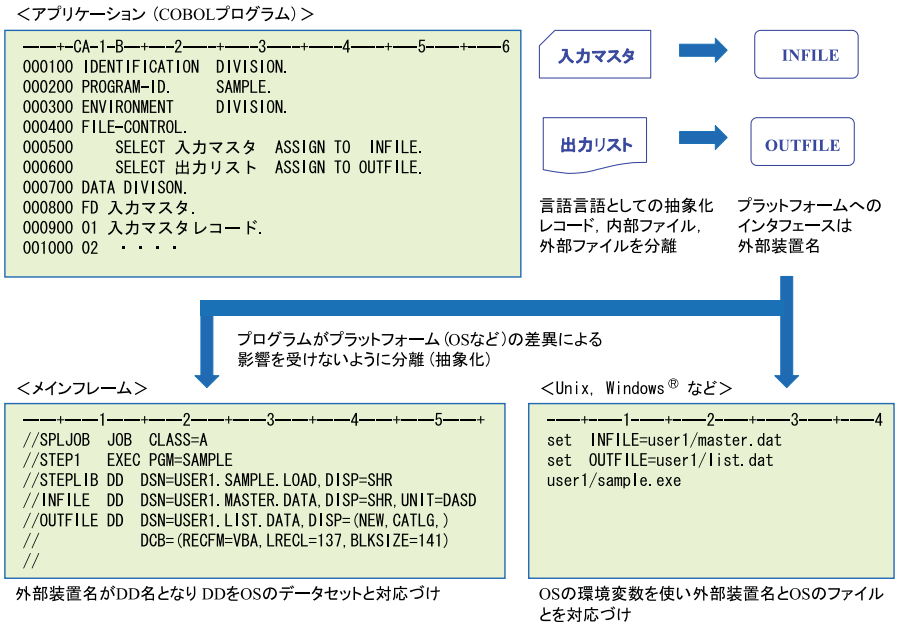
メインフレームと呼ばれた大型計算機が登場した頃は、計算機は大きなスペースを必要とし、重く、高価であったため計算機センタに置かれ、プログラマや利用者は事務所などの端末からアクセスしていた。複数の利用者に対応するため、メインフレームではTSS（Time Sharing System: 時分割）と呼ばれる方式などが採られた。

メインフレームは更に進化し、現在でもなお銀行の基幹システム、交通機関の座席予約システムなどの中核として使い続けられている。クラウドを可能にした、仮想化技術、ネットワーク技術などは、その内容は異なるものの、メインフレームで生まれた技術の相似形であると言えなくもない。ワークステーションやPCなどの「オープンシステム」の登場からユーティリティを標榜するクラウドに至る潮流は、計算機の誕生から社会基盤となったメインフレームの発達の過程に酷似している。

パラダイム	メインフレーム/ COBOL	オープンシステム / クラウド
スタンドアロン	バッチ処理	表計算など1台の中で利用 （CP/M, MS-DOS®）
マルチユーザ化	TSS, DCモニタ	マルチユーザOS （Unix, Windows NT®など）
ネットワーク	HNA, SNA, HDLC	NetWare®, LAN-Manager等を経て TCP/IPなど
仮想化	VM	Hyper-V, VMware®, Xenなど

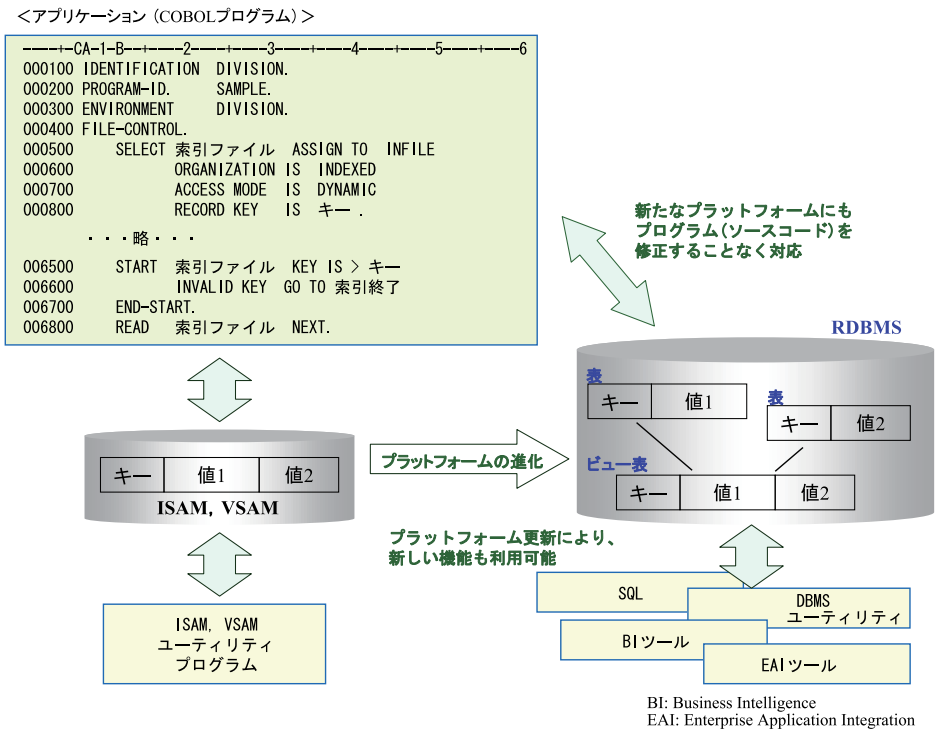
プラットフォームの変化を乗り越えるCOBOL

COBOLは、その名が表すように「共通の事務用プログラム言語」である。「共通」とは、本来は言語仕様について提供者を問わず仕様が共通であることを目的としていた。この基盤があったことから「どのような計算機の上でも共通の言語が使える」ようにするため、COBOLの言語要素とプログラムが動作するプラットフォームの要素とが抽象化され分離されるに至った。例えば、COBOLプログラム中ではI/Oの対象は外部装置名として抽象化されている。実行するOSの仕様/機能により外部装置名とOS上のファイルの実体との対応が行われることにより、OSの違いがプログラムの記述に影響を及ぼさないように工夫されている。



クラウド化されたプラットフォームでは、分散ファイルシステムなど新たな概念と機能が登場することが予想される。リレーショナルデータベースが重要なプラットフォームとなった際、表の操作を行うSQLをプログラムの中に記述できる「埋め込みSQL」をサポートした。しかし、「キーによるファイルの検索」という概念はISAM,VSAMなどそれ以前のプラットフォームにもありCOBOLでは索引編成ファイルとしてアクセスすることができた。この概念によりRDBの検索も可能なことから、RDBを索引編成ファイルとしてアクセスする方式も提供され、それ以前のプログラムを書き換えることなく、プラットフォームをISAMなどからRDBに置き換え、新たなプラットフォームの機能を利用することを可能とした。

また、ベンダによっては、CSV形式やXML形式のファイルについても、COBOLのレコードとして扱える機能を提供している。



このような抽象化により、COBOLはプラットフォームから独立してアプリケーションを記述でき、既存プログラムを新たなプラットフォームに対応させることも容易になっている。このため、クラウド時代においても、事務処理分野における主力言語としての地位を失うことはないと思われる。

変化に対応するCOBOL

他方、COBOLは時に応じ新たなITの概念を吸収し変化を遂げてきた言語でもある。種々の言語が栄枯盛衰を繰り返してきた、ALGOL,Pascalの構造化記述、Smaltalkに源を発し、C++やJava™により普及したオブジェクト指向、人工知能（AI）のアプローチを持つLisp, Prologなど枚挙に暇が無い。このような中にあって、COBOLのように1つの言語が50年もの長期にわたり改良を加えられ続けて来た例は他にない。現在でも、なお第5次COBOL国際規格の制定を目指して検討が進められている。第4次COBOL国際規格では「オブジェクト指向」が取り入れられたが、他にも「利用者定義のデータ型」「アドレス付け機能」「プログラムの再帰呼出」などが加った。構造化記述、オブジェクト指向などCOBOLより後に登場した言語の概念と文法が取り込まれたことにより、記述性・利便性においても、プログラミングの新たな手法の実装においても、他の言語に遜色の無い仕様を備えている。

また、規格化はされていないが、.NET Frameworkへの対応など、新たなIT技術への対応が行われている。COBOLコンパイラの提供者は、根幹を守りつつ、柔軟に変化に対処し、COBOLの利点を活かしながら、最新のIT技術の利用を可能とすべく不断の工夫を続けている。

おわりに

現在、非常に大きな変化をITに与えているクラウドではあるが、これとてもITのパラダイムシフトの１つの段階に過ぎないだろう。しかし、COBOLがこれまでのITの変化に対応してきたように今後も変化を遂げ、使い続けられる言語として最有力な候補であることに疑いはない。半世紀に渡るCOBOLの蓄積は他の言語には無いものであり、今後の激しい変化にも対応できる底力を備えているのである。

注[1]：http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc

注[2]：ニコラス・G・カー著『クラウド化する世界』村上彩訳，翔泳社，2008年

＜参考文献＞
・COBOLコンソーシアム/日経システム構築編集部共著
『オープンシステムに生かす実践COBOL資産移行ガイド』，日経BP社，2003年

スクール COBOL REPORT

ユーザー校 事例

福島県立喜多方桐桜高等学校

福島県立喜多方桐桜高等学校
渡部 和博

本校の概要

本校は今年度、喜多方商業高校と喜多方工業高校が統合し、喜多方桐桜高校として新たに創立された学校である。「士魂商才・勤労誠実」の校訓のもと「ビジネス」と「ものづくり」をともに学ぶ専門高等学校をコンセプトに、「新しい時代を担うスペシャリストの育成を目指し、主体的に進路実現が出来る人材の育成」を教育目標としている。そのための新たな取り組みとして、学科を超えて自由に科目を選択することが出来る総合選択制を取り入れている。設置している学科は工業科３学科に商業科３学科（エリアマネジメント科・ビジネス実務科・情報システム科）である。全国初となるエリアマネジメント科では「町育て」をコンセプトに地域の文化や産業を学び、体験・実践を通して地域貢献できる人材の育成を目指している。他の学科も将来のスペシャリストや起業家を育成するための「専門性の深化」を目指し、特色ある学科・カリキュラム編成となっている。

情報処理科の教育課程

本校の情報システム科の目標は「商業の基礎的科目の学習のうえに、高度情報通信ネットワーク化の進展に対応することができるよう、従来のプログラム言語教育に加えて、システム全体の設計・構築や管理・運営の技術を習得し、情報処理技術者として情報化社会に対応できる人材の育成を目指す」としている。

【1年】
[単位]　関係科目のみ
情報処理　3単位
プログラミング　4単位
[資格等]
全商情報処理検定ビジネス部門2級
全商情報処理検定プログラミング部門2級

【2年】
[単位]
プログラミング　3単位
OA演習（学校設定科目）　3単位
データベース（学校設定科目選択）　2単位
文書デザイン（学校設定科目選択）　2単位
[資格等]
全商情報処理検定ビジネス部門1級
全商情報処理検定プログラミング部門1級

【3年】
[単位]
ビジネス情報　2単位
マルチメディア応用（学校設定科目選択）3単位
[資格等]
経産省ITパスポート試験
経産省基本情報技術者試験

コンピュータの施設と設備

今年度から新たに商業実習棟が建設され、棟内にはパソコン室・情報処理室・総合実践室・プログラミング室・エリアマネジメント室が設けられた。実習用サーバシステムはドメインサーバがRX200　S2（RAID1）、ファイルサーバがRX300 S2（RAID5）、メタフレームサーバがRX300 S2（RAID1）を使用しており、LANによりどの教室からも生徒IDにてログオンできるようになっている。また、各教室には授業支援ソフト（SKY MENU）が導入されており、一斉にファイルや画面転送が可能で、授業や生徒のパソコン管理に役立っている。

COBOL学習の進め方

プログラミング学習は、実教出版の教科書「最新プログラミング21」を使用し、これに従い下記のような流れで学習に取り組んでいる。また１クラス教員２名、実習助手１名の計３名で指導を行い、生徒個々の理解力向上および知識の深化を図っている。

～学習の流れ～

①問題の分析と考え方　②文法の説明　③実習問題の説明　④領域図作成　⑤流れ図作成　⑥コーディング作成　⑦プログラム入力　⑧コンパイル（ビルド）　⑨デバッグ　⑩実行結果の出力　⑪教員による検印

授業内で終了できない生徒は、昼休み時間・放課後を使用し終了させる。

～実習内容～

①四則演算　②合計と平均　③見出しと編集　④条件の判定　⑤最大値と最小値　⑥一定回数の繰り返し　⑦ページコントロール　⑧複合条件　⑨グループトータル　⑩他方向への分岐　⑪テーブルの作成　⑫逐次表引き　⑬非逐次表引き　⑭順位付け　⑮内部整列　⑯二次元テーブル　⑰ファイル・データチェック　⑱ファイルの照合　⑲ファイルの更新・保守・併合

資格取得への取り組み

本校における資格取得は、全商協会主催の各種検定試験をベースに、1級取得者の中から希望者が経産省国家試験へ挑戦する形で進めている。

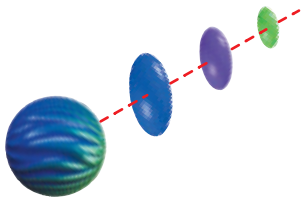
最後に

普通科に教科情報が導入されて８年になる。普通科の生徒も全商情報処理試験を受験する生徒が増え、上位級にも合格している。また、経産省の国家資格にも合格者を出している。このような現状で、商業科の情報教育と他学科との差別化をどのような点で見出していくべきかが、今商業教育に問われているのではないだろうか。

全商情報処理検定試験で出題されるCOBOL言語は平成26年度までとなり、27年度以降は廃止という流れにあるようである。本校でもこのままCOBOL言語を継続して学習するか、他の言語に移行するかが今後の検討課題となっている。しかし、当然ながら検定のための言語教育ではないので、このままCOBOL言語をやめてしまっていいものかどうか一抹の不安がある。プログラミングで生徒がはじめに躓くのはアルゴリズムである。COBOLは他の言語と比較してもアルゴリズムの基礎を身につけるには適していると思われ、またプログラムの構造も理解し易い言語であることから、COBOL言語（手続き型プログラム）で基礎を学び、オブジェクト指向やイベント駆動型の言語を習得させる方法が生徒にとって取り組み易いのではないかと考える。ただし、情報技術者に求められる能力や資質も時代とともに変化しており、プログラミング教育の学習方法や内容も時代に沿った見直しも当然必要になるであろう。しかし、どの言語を学習するにしてもプログラミングは商業科の情報教育において中核となる教育であるべきではないかと考えている。

企業におけるCOBOL活用

ジャパネットたかた殿 再構築事例



高田社長による
TV通販の様子

長崎県佐世保市に本社を置く通販大手「株式会社ジャパネットたかた殿」は、商品の魅力を最大限に表現することをモットーとし、「快適ライフのパートナー」をキャッチフレーズにメディアミックス戦略を展開。商品仕入れから、メディア制作、コールセンター・物流センター運営、アフターフォローに至るまで、自社で一貫した管理を行なう「自前主義」により常にお客さまの満足を追求して急成長を続けています。

2010年4月16日、東京コンファレンスセンターにおいて、COBOLコンソーシアム/日経BPセミナー事業センター主催COBOL誕生50周年記念セミナーが開催されました。

今回は、このセミナーの基調講演「ジャパネットたかたが COBOLを選択した理由」と題した再構築事例をご紹介します。

背景

通信販売では、顧客価値を高めるサービスを提供し続けることが重要で、新しいサービスが頻繁に追加されます。サービスを追加するたびにシステム構築が必要になりますが、仕様を外部のベンダーと決め、見積り、発注するサイクルはビジネスのスピードに追いつかず、競争に打ち勝つことができません。このため、Windows®上で構築していた情報系のシステムは、2005年頃から徐々に自社開発へと切り替えていきました。

一方、汎用機で稼動している基幹システムは、業務の拡大に伴い処理能力が限界に達し、CPU利用率が90%を超えるまでになりました。さらに、約6,000万件のデータをインターネット通販のEC（Electronic Commerce）サイトから利用したかったことも重なり、「マイグレーション」を決断。2008年1月、基幹システムのオープン化を通じて自社開発体制を確立させるプロジェクトをスタートさせるに至りました。ジャパネットたかた殿がめざすスピード経営には、業務全体を把握するために基幹システムをも自社開発することが重要と判断したのです。今後も長期にわたって継続利用できる開発・実行環境を整備するためには、自社開発によりシステムを理解し、トラブルが起こっても自力で復旧できる体制を作ることが重要と、専務執行役員 星井龍也氏は語ります。

＜これまでの開発体制＞

■システム構築をベンダーに依頼
- 汎用機上にメディアミックス通販システムの基礎を構築
- 4GL（第4世代言語）でCOBOLプログラムを生成
- 顧客との接点である顧客対応画面はVB開発
- VBによる顧客画面は汎用機上のCOBOLプログラム経由で基幹データベースと連携

■情報系はWindowsで開発
- ホストデータベースとSQL Serverの同期連携ツールを導入
- GUIをWebブラウザー上で表現するためASP言語で開発

■問題点
- 現場の改善スピードに追いつけない。

開発言語には「COBOL」を選択

情報系のシステムでは、Visual Basic®やC#、Java™などの開発言語を使用してきました。しかし、基幹システムを自社開発するには、ほとんどの技術者がこれまで経験のないCOBOL言語を理解しなければなりません。それでも、COBOLにこだわったのは、「信頼性と互換性の高さを評価したからだ」といいます。実際、日立COBOL2002は信頼性や互換性が高く、メインフレームで使用していた第2次規格版COBOLはほとんど変換する手間なく、オープン上でそのまま安定稼動させることができ、マイグレーションを成功させる大きな要因になりました。星井氏は、COBOLを選択した理由として以下を挙げています。

- ビジネスロジックを記述する言語としてCOBOLは最適。
- バッチ処理をC#やVBで書き直しても生産性は上がらない。
- データベース操作をSQLにさえすれば、言語はシンプルな構文を備えるCOBOLがわかり易い。
- これまでのビジネスロジックを活かせる。

COBOLは初めてという若手技術者も、データベースアクセス部分の変換ツールの開発を通して、実務でCOBOLを学んだそうです。COBOLは記述が非常に容易なことから、予想以上に短期間で戦力になったといいます。また、すべてのシステム開発に責任を持ち、運用していかなければならないという自覚と基幹業務の全容を知りたいという積極性も後押ししました。

自社内でシステムを再構築

今後システムを自社で運営していくためには、自社内でマイグレーションを完結させることが重要でした。実現するには開発要員の確保が必要です。このため中途経験者を募集し、更に不足人員分はベンダーから派遣することで補ったそうです。つまり、不足人員は外部から確保するけれども、自社内でマイグレーションを行うことでノウハウを蓄積し、自社内で運営できるようにしていくという考えです。そのため、社員とベンダーの混合チームで開発管理を仕組み化し、業務習得およびプロジェクトの効率と品質の向上を図ったといいます。

特に、データベース、プログラムの書き換えは、確実に自社内で行うという決意を持って望みました。これは、純粋なビジネスロジックとデータベース入出力の部分を把握することがスピード経営に深くかかわり、その他の部分はベンダーに任せる方が最適なIT環境がスピーディに構築できると考えられるからです。現行COBOL資産は、約700本/50万ステップ。データベース操作は、シンプルな構文を備えるCOBOLが解り易いことから、COBOL言語を使用してSQLに書き換えたと、星井氏はいいます。

マイグレーションを成功させるための重要な課題となった事項として以下を挙げています。

■パフォーマンス
データベースハードウェアとデータベースソフトウェアの効率性、安全性、拡張性にこだわって次期システム構成を選択。また、1000台の同時受注トランザクション処理を可能にするQueue操作と実地試験で性能評価を実施。

■基幹システムのビジネスロジックの正確な移行
今後、長期に渡って耐え得る、開発環境、実行環境を整備することを念頭に置いて推進。

■シンプルな技術を選択
自社で理解でき、自社で復旧できるものに絞る。

■マイグレーションの期間
混合チームのため、業務知識がなく、開発スキルにも差がある中で、汎用機の処理能力が限界があり、1年半というタイムリミットで推進。

結果と今後

本プロジェクトでは、現場担当者の協力も得ながら、コンバート後のプログラム品質向上を徹底的に追求。テストと検証に多くの時間を費やし、近年ほとんど例のない、自社内でのマイグレーションを2009年8月に予定通り完了させました。

今回のプロジェクトで最も重要な役割を果たしたツールが、既存のビジネスロジックをオープン環境へ容易に移行できる日立の「COBOL2002」でした。「COBOLは歴史が長く、現在も国際規格委員会の手で保守され続けています。ビジネスロジックの書きやすさにおいても右に出る言語はありません」と明言されています。

また、ジョブコントロールには日立の統合システム運用管理「JP1」を適用し、JP1/ScriptとJP1/AJS2（JP1/Automatic Job Management System 2）を組み合わせることで、従来のジョブ制御機能をほぼそのままの形で移行できました。帳票系でも日立のミドルウェアは驚くほど手間がかからず、効率的に作業を進められたといいます。

今回のマイグレーションにより、情報システム部門の全員がCOBOLを自由に読み書きできるスキルを身につけました。基幹系の開発スピードはマイグレーション前に比べて3 倍にも高まったということです。複数言語を操れるようになった自信と、業務知識の共有が急速に進んだことはマイグレーションプロジェクトの大きな成果です。

今後は6,000万件のデータを活用し、マーケティング担当者に役立つ機能追加などを計画しているとのことです。また、更なる攻めの経営を目指し、先進的なクラウドサービスの活用も星野氏は視野に入れています。

「自前主義」の精神を貫き、全員一丸となって取り組んだマイグレーションプロジェクトは大成功に終わりました。

[取材：（株）日立製作所 ソフトウェア事業部]

会社概要

株式会社 ジャパネットたかた

[本　　社] 長崎県佐世保市日宇町2781
[設　　立] 1986年1月
[資 本 金] 1億円
[従業員数] 501名（内パート・アルバイトを含む）
（2010年4月1日現在）
[業務内容] 家電、雑貨、健康食品などを扱う通信販売

スクール COBOL Report の歴史

本冊子は、1997年より、COBOL実習で使用されていた「スクールCOBOL85」を全国のCOBOL授業を行っている先生方に紹介する冊子として、2001年9月にVol-1を発行し、約10年間にわたり今回（2011年3月）のVol-14を発行することができました。

◆本冊子の内容

- 大学教授による巻頭寄稿
- COBOLの最新情報
- 「スクールCOBOL85」のCOBOL授業事例校の紹介
- 企業におけるCOBOL活用の紹介

10年間を一区切りとして、「スクールCOBOLレポート」形式での発行は今回をもって終了とさせていただきます。長い間のご愛読ありがとうございました。

次回より、COBOLも含めた新しい学校向け冊子を計画しています。ご期待ください。

（注）詳細は実教出版のURLをご覧ください。（http://www.jikkyo.co.jp/index.jsp）

編集後記

今回の本冊子（Vol-14）をもって、「スクールCOBOL Report」の発行は終了となります。

巻頭寄稿に執筆いただいた大学の先生、ユーザ事例校に執筆いただいた教育現場の先生方のご支援と、多くの皆様のご協力により、10年間も発行することができ感謝しております。

「COBOL言語は、古くて新しい進化する言語です。」これからも教育現場で活躍されることを願っています。

今後は、速い流れのIT業界に翻弄されることなく、教育現場でゆっくりと先生と生徒さんが一緒になって読める新しい冊子ができることを夢に見ています。

（写真は、昨年8月の大正池からの焼岳です）

発行元：**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/
株式会社 日立製作所 ソフトウェア事業部 販売推進本部　東京都品川区南大井六丁目27番18号　日立大森第二別館
Tel.（03）5471-2336　http://www.hitachi.co.jp/soft/

※Javaは、Oracle Corporation及びその子会社、関連会社の米国及びその他の国における登録商標または商標です。
※Microsoft、Windows、WindowsNT、およびHyper-Vは、米国MicrosoftCorporationの米国およびその他の国における登録商標または商標です。
※Microsoft.NETは、お客様、情報、システムおよびデバイスを繋ぐソフトウェアです。
※MS-DOS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。
※NetWareは、米国Novell、Inc.の登録商標です。
※SQL Serverは、米国法人Sybase、Inc.の商標です。
※UNIXは、The Open Groupの米国ならびに他の国における登録商標です。
※VisualBasic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。
※VMwareは、VMware、Inc.の米国および各国での登録商標または商標です。
※Xenは、Citrix Systems、Inc.の米国およびその他の国における商標です。
※その他記載の会社名・製品名は、それぞれの会社の商標もしくは登録商標です。

日立スクールCOBOL
リピータ特別価格のご案内

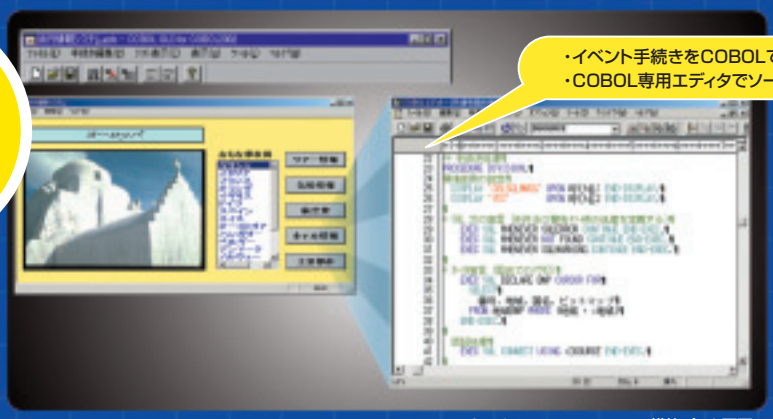
すでに「スクールCOBOL2002スクールパック」、「スクールCOBOL2002GUI構築スクールパック」、「スクールCOBOL85スクールパック」、「スクールCOBOL85GUI構築スクールパック」のいずれかをご導入されている場合、最新の「スクールCOBOL2002スクールパック」、あるいは「スクールCOBOL2002GUI構築スクールパック」をリピータ特別価格でご提供させていただきます。

「スクールCOBOL2002シリーズ」

●情報処理教育用COBOLコンパイラの決定版

■使い勝手が良く、実習を助けるツールが揃っている。
■最新の第4次COBOL国際規格に対応
■高信頼性と豊富な実績ある「スクールCOBOL85」の後継製品

使いなれたCOBOLでGUI構築技法を効率よく学習



スクールCOBOL2002 GUI構築パック画面

価格（税抜）

スクールCOBOL2002		スクールCOBOL2002 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、Windows®XP、Windows Vista®、Windows®7

※ 対応OS：Windows®2000、Windows®XP、Windows Vista®、Windows®7

「スクールメディアガード」

●個人情報漏えい対策ソフト

■重要なデータを暗号化することにより、『盗難』や『紛失』やメールの誤送信や盗聴から、重要なデータを守ることができます。
■暗号化したファイルを、USBメモリーに保存して運搬・送付するのに安心です。
■暗号化したファイルを、メールに添付して送信するのに安心です。

「個人情報漏えい対策ソフト」

価格（税抜）

●基本セット	7,600円	（媒体＋ライセンス2）
●20セット	75,000円	（媒体＋ライセンス20）
●50セット	186,000円	（媒体＋ライセンス50）
●1ライセンス	3,800円	（媒体＋ライセンス1）

※ 対応OS：Windows®2000、Windows®XP、Windows Vista®、Windows®7

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

PRINTED WITH SOY INK
本カタログは環境に配慮し、植物性大豆油インキを使用しています。
再生紙を使用しています