FREE MX COBOL COMPILER AND POSSIBILITIES OF USING IT

Vlastimil Čevela
Modřice, http://www.volweb.cz/cevelavl/, cevelavl@vol.cz

Motto:
„I need to process 2 million to 3 million records in my program .. etc.“
(from a topic in the IBM COBOL/OS390 Expert Forum – 04 Jan 2006)

ABSTRACT:
From the world's point of view the current status of COBOL consist of many enterprise programm-systems in a large organizations first of all on IBM mainframe computers which run under CICS (Customer Information Control System). These applications process billions of online transactions every day and provide the bussiness logic and database processing for most e-bussiness sites. Alongside there is a push to migrate these applications to new plattforms as Unix, Linux, Windows or Internet (.NET). These facts involved the need of the Cobol programming qualification not only for this time but for the future too.
Free MX COBOL compiler is user friendly, simple and easy to obtain compact tool which can be used in MS Windows for small tasks and training of basic programming skills in classical Cobol concepts. This text describes characteristics of the compiler and some possibilities of using it.

KEY WORDS:
Free COBOL compiler, Cobol language, IBM CICS, Cobol concepts, mainframe, training of programming skills, IBM Cobol Family, HP Open VMS Cobol, Acucobol-GT, Fujitsu NETCobol, Fujitsu Siemens Cobol, Micro Focus Net Express, Čevela MX Cobol,

1. Big players in the world of COBOL

COBOL is the programming language not only with the past but also with the future. In classical dialects it provides a striking contrast to Java, C-family, and Visual Basic. Structured programming vs. Object-oriented programming. Self-documenting code vs. Cryptic code. One ***read*** *s*tatement vs. no easy way to read a record in a file. A few dozen COBOL statements vs. hundreds of functions or thousands of classes, methods and properties. These and other apposite characteristics are very interestingly described in report [8]. But COBOL isn't classical language only. Let's put some contemporary facts about the selected important Cobol providers. They participated in preparing of the new ISO/IEC COBOL 1989:2002 norm too [7].

1.1 IBM Cobol Family [4]

provides a complete offering of compatible, cross-platform, cross-product compilers that support programming in all IBM operating systems include Object-Oriented concept, Report Writer and WebSphere. Language Environment provides a common run-time support which estabilishes the same way of processing for C, C++, COBOL, PL/I, Fortran and Assembler applications.

1.2 Hewlett-Packard [3]

HP COBOL (formerly known as Compaq or DIGITAL (DEC and VAX) COBOL) is a high-level language for business data processing that operates on the OpenVMS (VAX, Alpha, and I64) and Tru64 UNIX (Alpha) platforms. It is a high-performance, optimizing compiler environment that is based upon the 1985 ANSI COBOL Standard X3.23-1985 as modified by the X.23a-1989 amendment. Support for ORACLE CDD/Repository and some X/Open features are also provided.

1.3 Acucobol [1]

ACUCOBOL-GT lets to write a program once, and run it on virtually any operating system platform without recompiling. This platform independence is achieved through an efficient pseudocode system. Rather than producing native code for a single platform, compiler produces a small, independent object that can be placed on any open-systems machine. At runtime, the object is executed by a native runtime executable COBOL Virtual Machine.

1.4 Fujitsu / Siemens [2]

provides two NetCOBOL product packages for Windows, three for .NET, two for UNIX (HP-UX and SPARC Architecture), one for Linux and an integrated environments for developing and deploying client/server and standalone applications on Solaris, Reliant UNIX and BS2000/OSD platforms.

1.5 Micro Focus [5]

Micro Focus Net Express with .NET is the COBOL development environment for extending core business processes to the .NET Framework and other distributed platforms. Powerful integration with Visual Studio .NET and the .NET Framework plus direct COBOL Web services capabilities, J2EE connectivity and XML support allow easy integration of existing and new COBOL applications with leading enterprise technologies. In MF Studio there are other Server and Minframe Express and Revolve tools.

2. Classical COBOL concepts in the free MX Cobol compiler

Free MX Cobol compiler has got the long history. As the Czech-original mainframe tool in seventies and eighties of last century – operation system MOS and DOS 3 etc. (KS Brno + VUMS Praha) – some hundreds installation in EC1021-27, EC1030 and 40 in former Czechoslovakia and East Germany [9], [10],[12] - was transfered into PC platform in nineties. From 1994 its owner (with all rights) is Cobol specialist and fan – Vlastimil Čevela. He placed it on the internet for free use now (March 2006) [13].

MX Cobol is user friendly simple compact compiler for small tasks and training of basic programming skills in Cobol language. It can run in 16-bit MS DOS and in command-mode in 32-bit MS Windows too – it was succesfully checked for 2000 and XP. It is based on ANSI-74 COBOL with some features from ANSI-85 (***initialize, end-if, end-perform***), ISO/IEC 1989:2002 (***screen section***) and many own system library routines (math-functions, work with bits etc.). Its output is source in C with following free Borland TC 2.01 compilation recommended [13].

2.1 Source program structure is partly fixed.
Ther are some standard division, section and paragraph headers that must be used in prescribed order when the relevant parts of program are desired. The basic principle is separation of common program options, data descriptions and procedural statements.
e.g.

program-id, special-names, file-control, file section, working-storage section, linkage section, report section, screen section, procedure division, declaratives.

This historical concept not only considerably increases simplicity, clarify and maintainbility of programs but also enables to join the most modern qualities to the classical ones. As examples we can give ***class-id, method-id, function-id*** etc. from [7].

2.2 Data items (structures, records) have to be explicitly described
in sections (see above) of the ***data division***. Each data description entry begins with a level-number followed by the programmer's data-name, and a sequence of independent clauses descriptive of the item. Main data categories are numeric (***pic 9***) and alphanumeric/non-numeric (***pic x***) elementary data items which can create group items in different levels – ***1*** (or ***01***) thru ***49*** and ***77*** for noncontiguous elementary item.

2.3 Program control flow in style of top-down structured programming
can be realized with help of exceptional statement ***perform*** for called (paragraph-name) or embedded (terminated with ***end-perform***) internal subprograms. Statement ***if .. else .. end-if***, logical expressions and conditional names allow create all algorithms as three basic structures - sequence, decision and iteration or as decision-tables-like ones [11].

2.4 I/O handling via keybord and screen
Statements ***accept, display, exhibit*** and ***stop „literal“*** give the possibility of the on-line interactive dialog with program. This dialog can be programmed not only in line-mode but with ***screen section*** declaration in positioned text-mode full-screen too.

MX Cobol compiler provides ***ufd*** and ***ufo*** statements for combination of Cobol with source in C language. With them there is possible input of options-data-string from command line i.e. from batch or command files.

2.5 Classical domain for Cobol is file handling.
Organization of files can be ***sequential*** (include ***line sequential*** for simple text files) or ***indexed*** with direct access to record with key-item.

2.6 Standard arthmetic
is realized with basic operations (***add, subtract, multiply, divide***) and universal ***compute*** [***rounded***] statement for arthmetical expressions as desired.

2.7 Very interesting is Cobol strings-character handling.
It's possible to string them from different parts, unstring from free format with separators to fix structure, analyse, replace and transform according to any code-table include hexadecimal one.
e.g.

string X-1 ***delimited*** Y-1 … ***into*** Z-1,
unstring A-1 ***delimited*** [***all***] B-1 ***or*** B-2 …
  into C-1 ***delimiter*** D-1 ***count*** E-1

```
      …,
  inspect F-1
    tallying G-1 for all/leading/ending [before/after [initial] H-1]
    replacing
      all/leading/ending/first/last I-1 by J-1
      …,
  transform K-1 from „1234567890“ to #f1f2f3f4f5f6f7f8f9f0#.
```

2.8 Subscripting and table handling
is important for saving of data fields (Dat-1 ***occurs*** Integer-1 [***depending*** Dat-2]) in memory and searching with „***search*** Dat-1“ statement in them.

2.9 Modular structure of project
is possible to be realized with help of external subprograms using of ***call*** and ***cancel*** statements.

2.10 Report writer for level-controlled listings
is traditional Cobol-tool included not only in MX Cobol but also in most modern ISO/IEC COBOL 1989:2002 [7].
e.g.
report section (with "***rd***" = report group descriptions)
initiate, generate and ***terminate*** statements

3. What is MX Cobol like in brief

3.1 Environment

MX COBOL is not the mouse oriented grafical environment with particularly specialized windows, menus, and buttons.
You write your source programs in the simple MS Windows Notepad editor, for the compilation control is associated standard command file and for program executing you have to double click on the respective icon. But in every time you can see what are you doing.

3.2 Applications

With MX COBOL you can't develop spectacular GUI applications.
But your programs can read files or generate static outputs in standard HTML, JavaScript, CSS, XML and other modern text oriented formats without any problems . And so you have quite wide area for thinking and working in current state of branch.

3.3 OO and RAD

MX COBOL is neither Object Oriented nor Rapid Development oriented.
But with help of it you can taste beauty and charm of good structured classical procedural programming with your own clear and self-documented control on algorithmic flow of statements.

3.4 Training Tool

It is very important to say - MX COBOL isn't the professional executive tool at all.
It's purpose is to help you to learn basic skills in classical Cobol concepts and discover the power of its natural way of expression. Just this last quality perhaps was the main for its longevity and useful stability in stormy waves of IC technologies.

3.5 Intellectual entertainment

And if you have a sense for creative thinking and invention you maybe can appreciate that MX COBOL doesn't force you to work with drags, drops and clicks to create the blinking, moving and noising wonders.
But you can try your intellectual ability to algorithm problems in the same programming conditions which had prepared Grace Hopper and her co-workers in historical sixties of previous century.

Well, is MX COBOL attractive for you ? - I put it to you, and I leave it to you.

LITERATURE:

[1] Acucorp – ACUCOBOL-GT
http://www.acucorp.com/

[2] Fujitsu NETCobol and Fujitsu Siemens COBOL
http://www.netcobol.com/
http://www.fujitsu-siemens.com/products/software/compiler/languages/cobol.html

[3] Hewlett Packard OpenVMS COBOL
http://www.openvms.compaq.com/commercial/cobol/index.html

[4] IBM COBOL Family (with Library) and IBM Mainframes Expert Forum
http://www-306.ibm.com/software/awdtools/cobol/
http://ibmmainframes.com/index.php

[5] Micro Focus and Net Express 4.0 Language References, Programming Guides etc.
http://www.microfocus.com/
http://supportline.microfocus.com/supportline/documentation/books/nx40/nx40indx.htm

[6] Microsoft – Mr. Gates and Mr. Kadhim remarks about COBOL in .NET
http://www.microsoft.com/BillGates/Speeches/2000/07-12pdc.asp

[7] COBOL Standards and ISO/IEC 1989:2002 COBOL
http://www.cobolstandards.com/
http://www.iso.ch/iso/en/CombinedQueryResult.CombinedQueryResult?queryString=COBOL
http://honor.fi.muni.cz/tsw/2003.HTM#009

[8] Murach M., The Future of COBOL
http://www.infogoal.com/cbd/cbdz040.htm

[9] Vérosta V., Překladač COBOL EC-1021,
in „Používání jazyka COBOL“ (page 82-85), DT ČSVTS Pardubice 1975

[10] Lacko B., Pravidla pro používání jazyka COBOL,
in „Používání jazyka COBOL“ (page 173-186), DT ČSVTS Pardubice 1977

[11] Lacko B., COBOL a rozhodovací tabulky,
in „Základy technologie programování v jazyku COBOL“ (page 134-140),
DT ČSVTS Pardubice 1981

[12] Drbal P., Vaníček J., Metodika programování v operačním systému DOS-3/JS
in „Programování '81“ (page 198-207), DT ČSVTS Ostrava 1981,

[13] Čevela V., Some topics about COBOL with Free MX Cobol and other interesting links
http://www.volweb.cz/cevelavl/cobol/