IBM COBOL V4.2 END OF SERVICE

Position Paper

Marble Computer, Inc.

6416 Via De Albur Court
El Paso, TX 79912

Phone 800-452-1400
Info@marblecomputer.com

What End of Service for IBM COBOL 4.2 Means for Your Organization

IBM is Providing Extended Service for COBOL V4.2

According to the IBM website, IBM is offering Extended Service for an additional fee. There is no Extended Service pricing on the IBM website. There are other IBM Enterprise Compilers going EOS on April 30, 2022, besides V4.2.

What does the COBOL EOS date mean for your organization according to IBM?

Withdrawing service for Enterprise COBOL 4.2, 5.1, 5.2 and 6.1 means customers can no longer open cases or get APAR fixes for the COBOL compiler after the end of service dates become effective. Code compiled with these compilers will continue to run. In addition, the COBOL runtime, which is part of the z/OS Language Environment, will continue to be supported.
MLC customers can continue to use COBOL 4.2 without support as long as the monthly charge is paid.
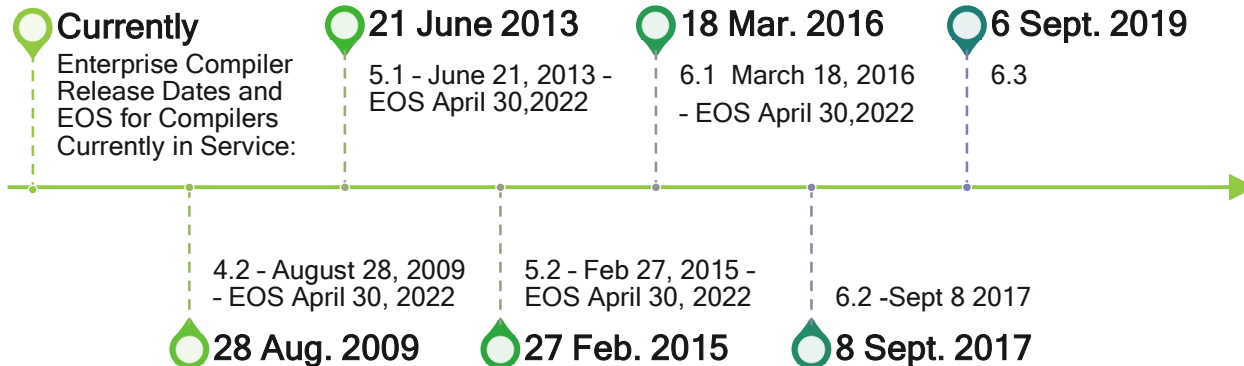
Best Options for Migrating to COBOL 6.

IBM is offering its customers a way to optimize COBOL V4.2 Code with the use of the Automatic Binary Optimizer for z/OS. IBM claims the ABO will improve the performance of older compiled code. ABO optimizes existing compiled binaries for the latest IBM Z hardware.

IBM suggests using COBOL 6 for new development, maintenance, and modernization, and use ABO to improve the performance of the COBOL V4.2 modules without a recompilation plan.

Customers who are currently on COBOL 4.2 are encouraged to plan and start their migration to COBOL 6 as soon as possible. Plan your migration by attending a no-charge COBOL Migration and Performance webinar. In addition, you can find all the COBOL migration-related information in the COBOL Migration Portal.

COBOL Compiler Update 04-01-22

Currently
Enterprise Compiler Release Dates and EOS for Compilers Currently in Service:

21 June 2013
5.1 - June 21, 2013 – EOS April 30,2022

18 Mar. 2016
6.1 March 18, 2016 – EOS April 30,2022

6 Sept. 2019
6.3

4.2 – August 28, 2009 – EOS April 30, 2022

5.2 – Feb 27, 2015 – EOS April 30, 2022

6.2 -Sept 8 2017

28 Aug. 2009

27 Feb. 2015

8 Sept. 2017

The Enterprise 6 compilers focus on producing optimized code. The optimized code has produced workflow productivity by 10-20% according to IBM. However, the new compilers have created migration issues the most glaring of which is Invalid Data. By using Marble Computer software, Control/DCD, our customers have avoided many of the migration pitfalls.

COBOL 6.2 and 6.3 compilers will be the only supported Enterprise compilers after April 30, 2022. Yet, COBOL Version 4.2 is still the most widely used of the Enterprise Compilers.

IBM, in our opinion, is putting their mainframe customers in a position of moving entirely to COBOL 6.2 or 6.3 or paying a monthly license charge for a compiler that is no longer supported.

Knowing that COBOL Version 4.2 was going to go EOS at some point, Marble Computer's technical staff developed enhancements for the standard COBOL Listing for each COBOL Compiler, past and present.

Referred to as, Enhanced Compile Mode within our Control/DCD software, these enhancements provide significant information regarding the logic of an individual COBOL program. The COBOL programmer now has additional narrative information and warnings referred to as, the "Special Narrative", available to them within the COBOL V4.2 compiler listing.

Enhanced Compile Mode adds value to the last release of COBOL V4.2, while providing features for migrating to COBOL 6.2 or 6.3.

Below is some of the functionality that provides enhancements to the standard IBM COBOL Compiler Listing.

- <u>Forward Tracing Chart</u>
  - o Added to show a hierarchical indented chart of Performed Routines *(Hierarchical Tracing is unique to Control/DCD.)*

- <u>Perform Analysis</u> is provided to show:
  - o Existing PERFORM Errors that were introduced at some point in development
  - o Top-down Structure of PERFORMs for use in COBOL Refactoring to make the COBOL program more maintainable
    1. This includes showing backward PERFORMs for possibly restructuring program code into only forward PERFORMs

- <u>Special Narrative</u> for each data field. The three features of this Special Narrative are shown here:

  - o **DATA DIVISION** information including PIC, USAGE, VALUE, From-To positions, 01 record information, SECTION, any COPY member information, and a sequence number <u>preceded with an &</u>, to allow an immediate FIND in ISPF that goes directly to this field in the DATA DIVISION in the source code
  - o **PROCEDURE DIVISION** narrative relating to all activity is very different from the older narrative in Control/DCD that was appended on the right side of to the data field in the DATA DIVISION
    1. The <u>Performed Routine Name</u> (Paragraph or Section) that the activity belongs to has been added to allow the tracing activity to be separated by Performed Routine Name
    2. Activity is indented underneath the corresponding Routine Name for understanding what Routine the activity belongs to (see example below)
    3. The SORT sequence is changed by necessity to fit this new visual look
    4. The sequence number in parentheses at the end of any PROCEDURE DIVISION statement referring to the sequence number in the PROCEDURE DIVISION source code may be

used with the addition of an & character, so that an ISPF FIND will go right to this PROCEDURE statement

- o **Routine Tracing (<u>Abbreviated</u> Forward Tracing Chart)** for each data field
    1. This abbreviated chart references every <u>Performed Routine Name</u> also referenced in above PROCEDURE DIVISION narrative
    2. The Routine Names are indented for a visual look at the program's flow

Migration to COBOL 6.2 or 6.3 provides several challenges, the most critical issue is the creation of Invalid Data by the new Enterprise Compilers. Much has been written about the need to plan a migration project carefully. There are six critical issues to be concerned with:

1. Invalid Data due to past programming conventions that no longer work in COBOL 6.2 or 6.3
2. Call Parameter Arguments
3. Misuse of the Occurs Clause
4. Packed Decimal and Binary data definitions and usage
5. Perform Errors
6. Perform Structures.

To overcome these problems IBM has set up a unique set of compiler options for testing and then production. The most useful compiler option is NUMCHECK. NUMCHECK, when on, adds an IF Statement in object code to every numerical instruction. If NUMCHECK is left on in production, it can cause an overhead of up to 50%. Because every case of Invalid Data cannot be detected future production stoppage can occur months or years later according to IBM.

Control/DCD provides an ability to be used as a pre-compiler for COBOL 6.2 or 6.3. Control/DCD will provide information on all 6 major obstacles to migrating to a COBOL 6 compiler.

Control/DCD provides compiler like warnings on invalid data, uninitiated display fields, perform errors, invalid use of the occurs clause, poorly defined packed

decimal and binary data, invalid parameter arguments, poorly structured perform statements, dead code, and more.

Please visit our website for more detailed information at www.marblecomputer.com. While on the website please look in both the Articles and Download section for a myriad of brochures and whitepapers on the subjects discussed within this paper.

You may also want to visit our Video Gallery for more information on Enhanced Compile Mode.