



Evolution, without revolution

Solution Overview

Migrating from Micro Focus COBOL to isCOBOL Evolve

© 2017 Veryant. All rights reserved.



Table of contents

- 3. Introduction
- 4. Introduction to isCOBOL Evolve
- 6. Compiler options
- 10. Data access
- 11. User interfaces
- 13. Interoperability
- 14. Getting started with a move to isCOBOL
- 15. Conclusion

Optimise IT resource Comprehensive COBOL Clear future

With isCOBOL technology from Veryant, you can move to an innovative COBOL platform with lower TCO, a solid roadmap, and numerous options for GUI maintenance and modernization, all backed by a dedicated global support team.

INTRODUCTION

Veryant delivers comprehensive COBOL application development, maintenance and modernization technologies that optimize IT resources, improve business performance, and dramatically lower cost.

Organizations are converting applications from Micro Focus™ COBOL platforms such as Micro Focus Server Express™ and Net Express™ to Veryant technology for many reasons:

- **To gain the advantage of a COBOL development and Java deployment model without rewriting or retraining.**
- **To improve ROI and dramatically lower COBOL licensing fees.**
- **To deploy to multiple platforms using a single set of source code.**
- **To modernize applications with a graphical thin client and web client options and natively integrate with a leading Web and application server technologies.**
- **To support a wide range of data access options and capabilities.**

isCOBOL Evolve from Veryant offers a seamless transition plan and a stable path forward for Micro Focus COBOL users. Backed by a dedicated team of COBOL experts and over 100,000 production installations, isCOBOL protects existing COBOL investments and offers the perfect blend of COBOL and Java for comprehensive, cost-effective development, deployment and modernization of COBOL applications.

This quick reference guide highlights the benefits of the isCOBOL platform, gives you advice on planning for the move to isCOBOL, and helps you understand similarities and differences between isCOBOL and Micro Focus COBOL.

Multi Platform

Compelling
cost-effective
alternative

Web
Enablement

Java Integration

The guide includes:

- [An introduction to the isCOBOL Evolve platform](#)
- [xxxx](#)
- [xxxx](#)
- [xxxx](#)
- [Details on how to get started with a move to isCOBOL](#)

INTRODUCTION TO ISCOBOL EVOLVE

isCOBOL Evolve offers a compelling and cost-effective alternative to distributed COBOL platforms such as ACUCOBOL-GT and RM/COBOL. isCOBOL technology enables customers to retain and enhance valuable COBOL application and development assets, while taking full advantage of the flexible Java platform in deployment.

Core components of the isCOBOL suite include a 100% portable COBOL compiler and runtime environment; an Eclipse-based Integrated Development Environment (IDE) with a real-time syntax checker; and a 100% portable, graphical debugger that facilitates remote debugging.

With isCOBOL technology, all development and debugging tasks are performed in a familiar, flexible and user-friendly COBOL environment -- no retraining or rewriting code required. The isCOBOL Compiler translates COBOL source code into Java classes that are then executed with the Java Virtual Machine (JVM). Because the isCOBOL Runtime Environment is implemented entirely in Java, the result is an extremely portable, robust solution that simplifies development, administration and implementation tasks. Thin Client and distributed processing capabilities are included in the isCOBOL Runtime Environment, enabling developers to maintain one graphical user interface (GUI), regardless of platform choice or deployment model. Application modernization efforts, such as improved integration and interoperability are also made easier.

The modular design of isCOBOL Evolve supports a wide range of data access options. All common COBOL file organizations and record types are supported, including Indexed, Relative, Binary Sequential, and Line Sequential file organizations, with fixed and variable length records. isCOBOL has a built-in ESQL compiler that translates embedded SQL (EXEC SQL) statements to JDBC calls, and also supports Pro*COBOL and DB2 Precompilers.

isCOBOL Evolve supports ANSI standards and legacy COBOL dialects, so application code can quickly be replatformed to an isCOBOL environment with typically little or no change. With isCOBOL Evolve, 'compile once, run anywhere' is truly delivered and new application modernization features and data options can be rapidly introduced.

PHASED TRANSITION OPTION

With Veryant, organizations have the choice of either migrating an entire application codebase at once or of taking a phased approach which sets the pace of a conversion according to resource availability and desired timeframe. During a phased migration, an organization's development team can continue to maintain an application with the Micro Focus COBOL compiler, while simultaneously working on the same set of source code to compile and execute with isCOBOL compiler. Veryant provides compiler and runtime compatibility settings to keep the overall level of effort required for conversion to a minimum.

MIGRATION PROCESS OVERVIEW

The key steps involved in a conversion to isCOBOL Evolve are:

- **Set up the isCOBOL SDK Evolve**
- **Recompile the application; review compiler output and make minor syntax modifications where necessary**
- **Migrate data using Veryant-supplied conversion utilities**
- **Set up the target platform runtime environment, including environment variables and data connections**
- **Integrate with external software libraries and routines such as those written in the C programming language**

The remainder of this document examines how Micro Focus COBOL programs and data files can be rapidly converted to isCOBOL Evolve. The paper specifically looks at compiler options, data access, user interfaces, interoperability requirements and the phased transition approach

Java Sources
generation

C memory
model

Full Pointer
support

COMPILER OPTIONS

isCOBOL Evolve include a 100% portable COBOL compiler, which supports the latest ANSI standards as well as common legacy dialects and XML data exchange. Embedded SQL (ESQL) is supported either through database-specific pre-compilers, such as Oracle Pro*COBOL or DB2 precompiler, or through Java Database Connectivity (JDBC). In the latter case, no ESQL pre-compiler is required. Developers do not need to be familiar with the Java programming language in order to work with isCOBOL compiler, all development and debugging is done in the COBOL language.

isCOBOL compiler operates as command line utilities, just like Micro Focus COBOL. Existing Micro Focus build scripts can be used in an isCOBOL environment by changing relevant compiler executable names and command line options. Veryant offers a high degree of compatibility with Micro Focus COBOL syntax, and for those parts of an application that rely on Micro Focus-specific semantics, compatibility options for compilation and runtime behavior are provided.

Command line options

Veryant and Micro Focus largely use different conventions for specifying compiler options at the command-line and in application source code. Veryant technology uses standard command-line style options, whereas Micro Focus COBOL uses its own compiler directive style.

With Veryant, command line options start with a hyphen followed by an abbreviation, and then sometimes an equal sign '=' followed by a value. These command-line style options can be specified in a file using the "iscobol.compiler.options" property or in the COBOL source code using the ">> IMP OPTION" directive.

For example, to provide compatibility with the mainframe behavior of the PERFORM statement such as on OS/VS COBOL:

- **Micro Focus COBOL requires the compiler directive PERFORM-TYPE"OSVS"**
- **isCOBOL requires the compiler command-line option -pt2**

Language syntax

Small code changes may need to be made to items such as inline comments, for example:

```
01 WS-VAR PIC X.    // Micro Focus inline comment
01 WS-VAR PIC X.    | Veryant inline comment
```

Runtime configuration variables (properties)

Small code changes may need to be made to items such as inline comments, for example:

At runtime, there are a number of isCOBOL properties that affect application behavior. These properties can be adjusted to obtain the same behavior as Micro Focus COBOL.

For example, a default behavior of Micro Focus COBOL is to strip trailing spaces from line sequential file records during WRITE operations. The same behavior can be obtained in isCOBOL by setting the runtime property “iscobol.file.strip_trailing_spaces=true”.

Another example is accommodating the Micro Focus RTS Switch “N”, which is default behavior in Micro Focus COBOL for including computational data in line sequential files. This behavior is obtained with isCOBOL by setting the property “iscobol.file.linesequential_N=true”.

Environment variables

No change is required to handle environment variables when moving from Micro Focus COBOL to isCOBOL Evolve. The same COBOL syntax is used to get and set environment variables.

ACCOMMODATING VARIATIONS

Veryant's support team has considerable experience assisting customers in migration off the Micro Focus platform. In areas where Veryant does not support specific Micro Focus compiler syntax, the effect of a Micro Focus compiler directive can be obtained using a recommended isCOBOL compiler option or runtime setting.

The following compiler options need to be used when migrating from MicroFocus COBOL:

- **-cm to use set the MicroFocus compatibility**
- **-dcm to use the MicroFocus sign encoding**

In addition, the following list maps commonly used Micro Focus directives to the appropriate isCOBOL Evolve alternative:

Table 1. compiler directives mapping

Micro Focus directive	isCOBOL directive
ADDSYN	-rm=
ALIGN	-align=
ANIM	-d
APOST	-apost
ARITHMETIC=OSVS	-cva
ASSIGN(EXTERNAL)	-cax
CHANGE-MESSAGE	iscobol.compiler.messagelevel
COBOL370	-cv
CONSTANT	iscobol.compiler.constant
CONVERTRET	-d5
COPYEXT(cpy)	-ce=cpy
COPYLIST	-lf

Micro Focus directive	isCOBOL directive
DATAMAP	-ld
DEFAULTBYTE(OO)	-dv=0
DOS/VS	-cv
ERRLIST	-ef
FOLD-COPY-NAME	-scnl or -scnu
FOLD-CALL-NAME	-ssnl or -ssnu
HIDE-MESSAGE	iscobol.compiler.constant
IBMCOMP	-dcmi
LIST, LISTPATH, LISTWIDTH, LW	-lf -lo=
MAKESYN	-rm
NOTRUNC	-dz
OVERRIDE	-rc
OSVS	-cv
PERFORM-TYPE	-pt0 or -pt1 or -pt2
PRINT	-lo=
REMOVE(NULL)	-rw=NULL
SEQUENTIAL	-cfl
SOURCEFORMAT	-sa or -st or -sf

MF-ISAM
Connector

JDBC
Integration

Flexible
Database
Model

DATA ACCESS

Veryant' innovative and modular design approach supports a wide range of All common COBOL file organizations and record types are supported by Veryant, including indexed, relative, binary sequential, and line sequential, with fixed and variable length records. isCOBOL compiler has a built-in ESQL compiler that translates embedded SQL statements into JDBC calls. Veryant software also supports database specific pre-compilers such as Oracle Pro*COBOL and DB2.

Most data files will need to be converted to an equivalent Veryant format before they can be used. A data migration utility is included with isCOBOL Evolve

Sequential files

Micro Focus and Veryant employ the same techniques to determine record lengths, so neither fixed length nor variable length sequential files require conversion.

Relative files

Relative files are converted with a utility provided by Veryant support.

Indexed files

Under Linux/UNIX platforms, a File Connector solution (named MFC Micro Focus File Connector) is available. This allows you to maintain your Micro Focus indexed files.

In addition an utility named ISMIGRATE (with the specific class ScanMF) allows to easily migrate them to any supported isCOBOL file system.

Relational databases

No data migration is required for relational databases when moving to isCOBOL Evolve -- current databases can continue to be accessed through existing interfaces without COBOL code changes. Veryant supports access to any database that offers a JDBC driver or a COBOL/ESQL pre-compiler. Databases can be accessed through standard COBOL file I/O statements or by including embedded SQL in the COBOL source code.

External File System

Veryant technology can be configured to use external file systems that contain indexed, relative or sequential files via an EXTfH (external file handler) interface. Developers can set the property, "iscobol.extfh.libname" to specify the name of the EXTfH library. I/O statements remain the same in the COBOL source whether using native or EXTfH file system libraries.

To further simplify the transition to isCOBOL Evolve, developers can configure the runtime to generate the same file status codes used by Micro Focus COBOL. This is accomplished by setting the property "iscobol.file.status=com.iscobol.io.FileStatusMF".

USER INTERFACES

Veryant supports a range of user interface options and access methods. The majority of user interfaces commonly associated with Micro Focus COBOL move smoothly to isCOBOL Evolve with a basic recompile.

IsCOBOL Evolve can work with a Character or GUI interface implemented in the COBOL language itself or it can also work with external interfaces like Web or Java Server technology, or a third- party screen package. The user interface may also be written in Microsoft Visual Basic™, C# (or other Microsoft .NET™ language). Or the user interface could be a C routine interfacing with a terminal. In all of these cases, the user interface is external to the COBOL applications themselves and would be supported 'as-is' without any migration requirements.

Character-based user interfaces

IsCOBOL Evolve enables character-based user interfaces to be deployed on dumb terminals and terminal emulators, locally on a graphical Linux, Unix or Windows desktop, or by using isCOBOL software's thin client technology. isCOBOL Application Server offers a thin client character or graphical user interface implemented with Java Swing, and is thereby completely portable. The isCOBOL platform also offers numerous compile options and properties to fine tune appearance, including fonts and colors.

isCOBOL supports the following character-based user interface methods:

- **Simple ACCEPT and DISPLAY**
- **Micro Focus enhanced ACCEPT and DISPLAY**
- **Windowing Syntax**
- **SCREEN SECTION**

Screen Section code comes over as is and isCOBOL offers additional controls and properties to update and modernize such user interfaces.

The following user interface methods require transformation in order to be migrated to isCOBOL Evolve:

- **Dialog system for character user interfaces**
- **Panels**

Extended
GUI Interface

Powerful
Reporting
Capabilities

Multi Platform

Graphical user interfaces

IsCOBOL Evolve delivers innovative, next-generation graphical user interface (GUI) programming capabilities directly in the COBOL language, no need to learn .NET or any other language. Organizations updating legacy COBOL applications with modern Web 2.0 user interfaces can build dynamic GUIs with the graphical isCOBOL Integrated Development Environment (IDE). Developers can also transform a green-screen into a GUI or add JavaBean graphical controls directly in the COBOL language.

The following graphical user interface methods require transformation when moving to isCOBOL Evolve:

- [Graphical User Interface API](#)
- [Panels V2 \(Version 2\) Windowing Syntax](#)
- [Dialog system for graphical user interfaces](#)
- [NET programming using WinForms or WebForms](#)

Screen Section code comes over as is and isCOBOL offers additional controls and properties to update and modernize such user interfaces.

The following user interface methods require transformation in order to be migrated to isCOBOL Evolve:

- [Dialog system for character user interfaces](#)
- [Panels](#)

Similar to Micro Focus COBOL, isCOBOL supports third-party screen I/O packages.

Function Keys

IsCOBOL Evolve supports all function keys received as value in the crt status definition.

INTEROPERABILITY

Programs compiled with isCOBOL can call external language routines without the need for code changes. Runtime framework properties are used to point to the external libraries. Also the opposite interoperability is supported, so an isCOBOL program can be executed from C or Java through a specific isCOBOL library, see isCOBOL documentation for details.

Library routines

IsCOBOL implements many library routines that provide equivalent functionality to the Micro Focus “call by name” routines (CBL_*).

The following Micro Focus library routines are supported by isCOBOL:

- **CBL_AND**
- **CBL_CHANGE_DIR**
- **CBL_CHECK_FILE_EXIST**
- **CBL_CLEAR_SCR**
- **CBL_COPY_FILE**
- **CBL_CREATE_DIR**
- **CBL_DELETE_DIR**
- **CBL_DELETE_FILE**
- **CBL_DIR_SCAN_END**
- **CBL_DIR_SCAN_READ**
- **CBL_DIR_SCAN_START**
- **CBL_EQ**
- **CBL_ERROR_PROC**
- **CBL_EXIT_PROC**
- **CBL_GET_SCR_SIZE**
- **CBL_IMP**
- **CBL_JOIN_FILENAME**
- **CBL_NOT**
- **CBL_OR**
- **CBL_READ_SCR_CHARS**
- **CBL_READ_SCR_CHATTRS**
- **CBL_RENAME_FILE**
- **CBL_SPLIT_FILENAME**
- **CBL_WRITE_SCR_CHARS**
- **CBL_WRITE_SCR_CHATTRS**
- **CBL_WRITE_SCR_N_CHAR**
- **CBL_WRITE_SCR_N_CHATTR**
- **CBL_XOR**
- **SYSTEM**

Where necessary, Veryant’s support team can write or assist in writing required subroutines, C routines or Java programs using the equivalent isCOBOL syntax.

Mobile solution

REST, SOAP
Services

JSON, XML
HTML
Integration

GETTING STARTED WITH A MOVE TO ISCOBOL

Veryant's innovative isCOBOL technology is highly compatible with today's common COBOL dialects, enabling quick and efficient migration from Micro Focus COBOL to the isCOBOL platform.

As with other COBOL-to-COBOL transitions, the key steps involved in a conversion to isCOBOL are:

- **Set up the isCOBOL Evolve Software Development Kit (SDK)**
- **Recompile the application with the isCOBOL Compiler; review the compiler output and make minor syntax modifications where necessary**
- **Migrate data using conversion utilities included with isCOBOL (if necessary)**
- **Set up the isCOBOL Runtime Environment, including environment variables and data connections**
- **Integrate with external software libraries and routines such as those written in the C programming language**

With isCOBOL, organizations have the choice of either migrating an entire application codebase at once, or of taking a phased approach which sets the pace of a conversion according to resource availability and desired timeframe. During a phased migration, an organization's developers continue to maintain an application with the ACUCOBOL compiler, while simultaneously working on the same set of source code to compile and execute with isCOBOL. With isCOBOL software's compiler and runtime compatibility settings, the overall level of effort required for conversion can be kept to a minimum.

When considering a move to isCOBOL, the Veryant support team can help assess the level of effort required. If desired, Veryant also offers services to build a limited representative prototype of an application running in an isCOBOL environment. This exercise includes a thorough analysis process that helps determine how straightforward the transition to Veryant will be for an organization.

With assistance from Veryant, it typically takes customers less than two days to get a clean compile of their application source code with isCOBOL. With an additional two days, a working prototype of an application running with isCOBOL can be produced.

Evolution not
Revolution

Application
Modernization

ROI
Improvement

CONCLUSION

Organizations are converting applications from Microfocus COBOL to the isCOBOL platform for many reasons:

- **to gain the advantage of a COBOL development and Java deployment model without rewriting or retraining**
- **to improve ROI and dramatically lower COBOL licensing fees**
- **to deploy to multiple platforms using a single set of source code**
- **to modernize applications with graphical thin client and web client options**
- **to natively integrate with leading Web and application server technologies**
- **to support a wide range of data access options and capabilities**
- **to implement mobile application**

With hundreds of thousands production installations worldwide, isCOBOL protects existing COBOL Investments and offers the perfect blend of COBOL and Java for comprehensive, cost effective development, deployment and modernization of COBOL applications.

Veryant welcomes the opportunity to work together with you to develop a plan to protect and enhance your valuable COBOL-based assets.

For more information on migrating to isCOBOL Evolve, visit us online at www.veryant.com or email info@veryant.com.



Corporate Headquarters

4455 Murphy Canyon Road, Suite 209
 San Diego, CA 92123 - USA
 Tel: +1 619 453 0914
 Fax: +1 858 569 4243
info@veryant.com

European Headquarters

Via Pirandello, 29
 29121 - Piacenza - Italy
 Tel: +39 0523 490770
 Fax: +39 0523 480784
emea@veryant.com

veryant.com

© 2017 Veryant LLC. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution and recompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Veryant and its licensors, if any.

isCOBOL and Veryant are trademarks, or registered trademarks of Veryant in the United States and other countries. All other marks are the property of their respective owners.