



Cumulative Changes from Mainframe Express to Micro Focus Enterprise Developer for Eclipse

Micro Focus
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK
<http://www.microfocus.com>

Copyright © Micro Focus 2018. All rights reserved.

MICRO FOCUS, the Micro Focus logo and Enterprise Developer are trademarks or registered trademarks of Micro Focus IP Development Limited or its subsidiaries or affiliated companies in the United States, United Kingdom and other countries.

All other marks are the property of their respective owners.

2018-06-07

Contents

Cumulative Changes from Mainframe Express to Enterprise Developer for Eclipse

.....	4
About this Guide	4
What was New	4
What was New in Enterprise Developer 4.0	4
What was New in Enterprise Developer 3.0	13
What was New in Enterprise Developer 2.3 Update 2	22
What was New in Enterprise Developer 2.3 Update 1	25
What was New in Enterprise Developer 2.3	31
What was New in Enterprise Developer 2.2 Update 2	39
What was New in Enterprise Developer 2.2 Update 1	44
What was New in Enterprise Developer 2.2	51
What was New in Enterprise Developer 2.1 Update 1	56
What was New in Enterprise Developer 2.1	58
What was New in Enterprise Developer 2.0	61
What was New in Enterprise Developer 2010	67
Significant Changes	71
Significant Changes in Enterprise Developer 4.0	71
Significant Changes in Enterprise Developer 3.0	79
Significant Changes in Enterprise Developer 2.3 Update 2	85
Significant Changes in Enterprise Developer 2.3 Update 1	87
Significant Changes in Enterprise Developer 2.3	90
Significant Changes in Enterprise Developer 2.2 Update 2	94
Significant Changes in Enterprise Developer 2.2 Update 1	98
Significant changes in Enterprise Developer 2.2	101
Significant Changes in Enterprise Developer 2.1 Update 1	104
Significant Changes in Enterprise Developer 2.1	105
Significant Changes in Enterprise Developer 2.0	106
Unsupported or Deprecated Functionality	107
Unsupported or Deprecated at Enterprise Developer 4.0	107
Unsupported or Deprecated at Enterprise Developer 3.0	107
Unsupported or Deprecated at Enterprise Developer 2.3 Update 2	108
Unsupported or Deprecated at Enterprise Developer 2.3 Update 1	108
Unsupported or Deprecated at Enterprise Developer 2010	108
Known Errors and Restrictions	108
Upgrading from Mainframe Express to Enterprise Developer	112
An introduction to the process of upgrading your COBOL applications	113
How to create an application created with Mainframe Express to Enterprise Developer	113
Using Enterprise Developer for Visual Studio	115
Change the Defaults to Replicate Your Existing Project Structure	117
Best Practice in Enterprise Developer Development	118
Modernize Your Applications and Processes	119
Procedural COBOL Compared with Managed COBOL	120

Cumulative Changes from Mainframe Express to Enterprise Developer for Eclipse

Welcome to Enterprise Developer for Eclipse. This document combines information on Enterprise Developer for Eclipse releases from the first release to the most recent release, and on upgrading from Mainframe Express to Enterprise Developer. This information is taken from the various releases' Release Notes and other sources, and brought together here for your convenience.

You can use this document:

- If you are migrating from Mainframe Express, to see everything that has changed in the life of Enterprise Developer for Eclipse.
- If your Enterprise Developer installation is up to date, to see the changes made in the latest release.
- If you already use Enterprise Developer but your installation is not entirely up to date, to see the changes made over several updates, together in one place.

About this Guide

We recommend that you read all the sections of this publication, looking for information on the release that is most relevant to your needs. The main sections are as follows:

What was New	This section describes new functionality that was introduced in each successive release.
Significant Changes	This section describes, for each successive release, the changes in behavior or usage that could affect the behavior of existing applications or impact the way the tools are used.
Unsupported or Deprecated Functionality	This section describes any functionality that was discontinued or deprecated at a given release.
Known Errors and Restrictions	This section describes known errors affecting the latest release, and restrictions in its use.
Upgrading from Mainframe Express to Enterprise Developer	This section gives guidance on upgrading from earlier Micro Focus products.

Within each section, we recommend reading in the reverse chronological order in which it is presented here - that is, reading about the most recent release first, and then going back through the development of the products. That way, you can more easily see if anything was added in an earlier release and subsequently removed.

What was New

This section describes the new features that were introduced in each successive release of Visual COBOL.

What was New in Enterprise Developer 4.0

This release provides enhancements in the following areas:

- [Integration with Eclipse](#)
- [Application Server JCA Support for Enterprise Server](#)
- [Application Workflow Manager](#)
- [AppMaster Builder](#)
- [Build Tools for Windows](#)
- [Code Coverage](#)
- [Codeset support](#)
- [Compiler directives](#)
- [Data File Tools](#)
- [Debugging](#)
- [Docker](#)
- [Documentation on working with large applications](#)
- [Enterprise Server](#)
- [File Handler](#)
- [HCO for DB2 LUW](#)
- [IMS support](#)
- [JCL support](#)
- [Mainframe Access](#)
- [Managed COBOL](#)
- [Library routines](#)
- [Micro Focus Unit Test Framework](#)
- [OpenESQL](#)
- [Platform support](#)
- [PL/I support](#)
- [Web Service Clients](#)

Integration with Eclipse

[Back to Top](#)

This release provides the following general enhancements:

- Support for Eclipse 4.7 Oxygen (64-bit) and 64-bit project templates - the 64-bit Eclipse 4.7 is now installed by default. Project templates are available for both 32-bit and 64-bit applications. You can set a default project template (either 32-bit or 64-bit) to use so that all future new projects will use that as the preference. On platforms only capable of building to 64-bit, the 64-bit templates are the default ones.

The 32-bit Eclipse is still supported on 64-bit Windows platforms, however you need to manually install it and a 32-bit Java.

Support for Eclipse 4.4 and 4.5 has been discontinued.

- Support for SUSE 12 - only the 64-bit SUSE 12 platform is supported and you can only use the 64-bit Eclipse with 64-bit projects on it.
- (Native COBOL projects only) The **Project Settings** page in the project's properties now include a search option. This enables you to find files and build configurations that have settings different from the ones set at project level.
- **Remove File Directives** - a context menu command in the explorer view enables you to reset a file's directives.
- IVP diagnostic tool enhancements - it is now possible to run the server-side IVP diagnostic tool for diagnosing issues with a remote connection from within Eclipse installed at the client side. The IVP tool now also performs a check for whether XTERM is installed on the remote machine.
- IMTK service mappings are now automatically regenerated if you make a change to the interface fields in the underlying program.

The following enhancements are available for JVM COBOL projects and package handling:

- New wizards for creating ENUM, DELEGATE and VALUETYPE types.
- Additional code snippets for Method-Id and Property-Id.
- Showing and grouping packages in the COBOL Explorer view - use the **COBOL JVM Project Presentation** command from the **View** menu in the explorer to either display COBOL categories or packages.
- Refactoring by renaming the package name - available from the context menus in the COBOL Editor and in COBOL Explorer.
- Refactoring by moving source folders, packages or compilation units - available from the explorer context menu.
- Support for switching off the package name mapping using the **Each part of the package name corresponds to a subdirectory** option in the **Build Configuration** page.
- Support for packaging .class files in a .jar file when building COBOL JVM projects.
- **Open Type Hierarchy** and **Open Call Hierarchy** context menu commands - available in the COBOL editor, COBOL Explorer and in the outline views while seeing the packages presentation in the IDE.

The following editor improvements are available:

- A **Properties** context menu command - enables you to access a file's properties directly from the editor.
- A **Show In** context menu command - enables you to locate the file in the COBOL or the PL/I Explorer.
- **Edit > Convert Tabs to Spaces** command - enables you to convert any tabs in your COBOL source files to a specified number of spaces.

Automatic relinking of applications created with Enterprise Developer 3.0:

- Enterprise Developer 4.0 can automatically relink existing projects created with Enterprise Developer 3.0 that have executable link artefacts. Eclipse displays a warning in the Problems view that the project requires relinking. It then offers a Quick Fix action for you to execute that will link your project with the most recent version of the run-time system.

Application Server JCA Support for Enterprise Server

[Back to Top](#)

This release includes the following enhancements:

- COBOL Resource Adapters now support WebSphere 9.0 and WebLogic 12.2.1.
- Tomcat 7.0 support for servlet generation with J2SEBeans.
- NullSearch utility - for COBOL resource adapters, this new utility provides assistance in locating NULL fields in mappings passed to Enterprise Server. When a large number of arguments is provided in the parameters passed to Enterprise Server, it is difficult to locate NULL fields, which are not allowed. The NullSearch utility isolates NULL fields, so the Java application can be corrected.

Application Workflow Manager

[Back to Top](#)

Improvements have been made in the following areas:

- Model editor - now includes:
 - A new **Edit** action. Enables you to edit any model component instead of working with the Properties view. The new action enables you to change the attribute values of several model components at a time.
 - An Outline and a Relationship Hierarchy view.
 - Improved backwards and forwards navigation in the model.
 - Automated sequence numbering in the model.
 - The context menu of the model editor has been restructured for more clarity.
 - Context help.

- Tool creation has been improved. Dependent components are generated from the function package tool definition.
- Development of models:
 - Newly created empty models now contain all categories. The new models also validate without any errors or warnings.
 - A new AWM system type, “Custom System”, is available. This system type supports the model development process by facilitating the creation, change, load and reload of a model.
- New modelling features:
 - The “Return Property Value” tool in the function package now supports several additional use cases.
 - The “Transfer File” tool in the function package offers improved support for mass processing.
 - Support for properties with an application-wide scope.
 - A modelling capability to structure the local file cache for copybooks and include files downloaded by the background parser in the editor. Additional tools are provided to check the existence of or clear the file cache for copybooks and include files.
- The product help includes some tutorials showing how to create new models and extend the functionality of existing models. See the *Tutorials Guide*.
- A number of template models are now available and can be used as a base to develop your own models.

AppMaster Builder

[Back to Top](#)

This release includes the following new features:

- AppMaster Builder is now available in the Team Developer Perspective (Technology Preview) - this perspective provides a development environment for your AMB projects that includes much of the functionality available from the AppMaster Builder perspective plus access to the AMB Endeavor Interface (Technology Preview).
- AMB Endeavor Interface (Technology Preview) - enables you to share and synchronize resources between an AMB project and mainframe Endeavor datasets.

Build Tools for Windows

[Back to Top](#)

This release includes Enterprise Developer Build Tools for Windows, a separately-installable component of Enterprise Developer that has been designed to be used in environments where you want to work with your COBOL and PL/I projects but you don't want the overheads associated with the Eclipse IDE.

Build Tools provide a lightweight, easy-to-install development environment that is well-suited for use in Docker containers and continuous integration or continuous delivery systems.

Code coverage

[Back to Top](#)

This release provides the following enhancements:

- Support has been added in Eclipse for code coverage for procedural copybooks.

Codeset support

[Back to Top](#)

Support has been added to enable codeset mapping to additionally be configured to use IBM's Conversion Tables directly instead of the Micro Focus supplied tables. You need to download IBM's conversion tables from IBM's Web site. Then you can use the MFCODESET environment variable to convert between IBM's CCSIDs.

Compiler directives

[Back to Top](#)

The following Compiler directives are new in this release:

- **DISPSIGN** - determines the display output of numeric fields with included signs, under an IBM mainframe dialect only.
- **GNTLITLINKSTD** - stops the suppression of call-convention 8 when both call-convention 2 and call-convention 8 are in effect for a `.gnt` file in an Intel x86 32-bit environment.
- **ILSMARTTRIM** - trims any trailing spaces from a string item returned by the `get` property associated with an alphanumeric item processed by `ILSMARTLINKAGE`.
- **MAINFRAME-FLOATING-POINT** - specifies the format of a program's floating point data items: either IBM hexadecimal format or IEEE format. This directive is supported in managed code only.

The following Compiler directives contain new parameters in this release:

- **CHECKDIV** - a new parameter 'ACOS' now emulates a divide by zero operation on an ACOS mainframe system: the quotient and the remainder are set to the value of the dividend.
- **OOCTRL** - a new parameter, `L`, specifies whether to include directory location comments in `.cls` and `.ins` inheritance files.
- **NUMPROC** - a new parameter 'ACOS' provides partial compatibility with the behavior of NEC ACOS COBOL processing of invalid data in `USAGE DISPLAY` data items and invalid sign information in `USAGE COMP-3` data items.

Data File Tools

[Back to Top](#)

It is now possible to export any filtered results. When filtering a data file, you can use the results to create a new data file - click **Search > Export Results** when a filter is applied to save the filtered records to a new file. You can:

- Save the filtered records to a new file.
- Save the records that match the specified filter (such as customer information or orders).
- Download a subset of the data from a remote file.
- Save a small portion of the data for testing purposes.

Debugging

[Back to Top](#)

This release includes the following enhancements:

- Debugging CICS channels and containers - you can now debug CICS COBOL applications that use channels and containers. The debugger now enables you to view the contents of any channels and containers, and to change content as you debug.

Docker

[Back to Top](#)

This release provides support to enable you to run your COBOL and PL/I applications in Docker containers, taking advantage of the many benefits offered by the Docker platform such as portability, performance, agility, isolation, and scalability.

Documentation on working with large applications

[Back to Top](#)

The product help now includes a new section, *Working with a Large Code Base*, that includes recommendations and best practices for working with large applications inside the IDE. It includes tips on

how to structure your projects, how to optimize the performance of the IDEs, and step-by-step workflow showing how to move an existing legacy application into Enterprise Developer.

Enterprise Server

[Back to Top](#)

The following enhancements have been made to Enterprise Server:

- Conversation filtering - the Enterprise Server Communications Process (MFCS) can now restrict access to listeners by client address. You can specify any permitted or forbidden addresses either by IP address, network mask, or domain name, and use wildcards. Filters can be applied to individual listeners, communications processes, or to entire regions. More specific filter rules override any general ones.
- The Enterprise Server Security Facility now starts throttling Verify requests when it receives more than 100 requests per second.

This can be used to limit the effectiveness of denial-of-service and brute force attacks. You can configure the value where throttling occurs. See *Verify Request Throttling* for more information.

- (Technology Preview only) Support for adding, deleting, and modifying XA resources in a live Enterprise Server region.

It is now possible to add, edit, or delete XA resources while an enterprise server instance is running. Any changes made come into effect after any in-flight transactions have completed. The ability to make these changes in a live environment comes under the control of the existing enterprise server permissions.

- XA-compliant Resources (XARs) - this release provides enhanced CTF tracing that allows more flexible reporting of warnings and errors on the RM switch module level.
- A new Communications Server resource class - enables you to control the access to the Enterprise Server Console Log and Communications Server Log when external security is in effect for an enterprise server region; see *Resource Classes for Communications Server* for more information.
- Improved catalog availability - there is now an improved resilience to temporary communication issues with the catalog and error reporting enabling a region to stay active if a region has multiple catalogs defined and one of the catalogs is not available.
- Enhanced SSL/TLS certificate support - for communications with TLS (formerly SSL), additional certificate and key file formats are supported. Servers may now be configured with both an RSA and an ECC key and certificate.
- Enhanced SSL/TLS cipher configuration - for communications with TLS (formerly SSL), the permitted cipher suites and their preferred order can now be configured. The minimum size of Diffie-Hellman groups for DH key exchange can also be configured. The defaults have been made more secure.

File Handler

[Back to Top](#)

This release provides the following enhancements:

- The DFSORT and SYNCSORT emulations now support the NULLOFL parameter of the OUTFIL statement.
- The **ASCIISOSI** configuration option is now available. It adds the required SOSI characters to the relevant EBCDIC DBCS character strings in order for them to be displayed or written out correctly.

HCO for DB2 LUW

[Back to Top](#)

This release includes the following new features:

- Support for CHARSET EBCDIC COBOL stored procedures
- Seamless integration with DB2 LUW SBCS databases, providing various EBCDIC collation sequences
- 64-bit HCO command-line tools are now available across all supported platforms
- HCO DCLGEN enhancements that provide more copybook generation options

IMS support

[Back to Top](#)

This release introduces the following new features:

- The IMS Database Forward Recovery feature has been updated to provide point-in-time forward recovery that enables you to recover all database transactions completed up to a specified point in time, including date and time of day.
- IMS support now includes MFS code for Light Pen devices. In your MFS, include a DEV statement with the FEAT=PEN parameter, and set the ATTR property on appropriate fields to DET or IDET to make them detectable by a light pen. See your IBM documentation for details.



Note: MFS that contains light pen code must be regenerated to take effect if the MFS is part of an Enterprise Developer project created using an earlier release.

- Support for export and import of IMS-configured enterprise server regions through Server Explorer.

JCL support

[Back to Top](#)

This release introduces the following new features and enhancements:

- Micro Focus FTP utility - you can now configure MFFTP to not process trailing blank spaces for a GET command, by setting the new control variable MFFTP_PROCESS_TRAILS_ONGET=FALSE and setting the LOCSTAT value TRAILINGBLANKS=FALSE.
- A new API has been introduced - MVSSPLPB - that enables you to search all job information held in the JES spool control files. It is more comprehensive than the existing MVSCASPB API, and should instead be used for all new spool enquiries.
- A number of new environment variables have been introduced that enable you to configure the JES startup behavior with regards to opening user catalog files. You can configure JES to open user catalog files only when they are required (and not during system startup), and you can configure JES to retry opening a catalog file that has failed to open. These changes increase the resilience of the JES system, as you can mitigate against JES startup failures if the user catalog files are not readily available whilst the region starts. See *Configuring Multiple Catalogs* for more information.
- The public catalog API, MVSCATPB, has been enhanced to better report on file errors when opening catalog files. The following new functions have been added to `mfpubcat.cpy`: GCFS, GNFS, STNI, GCFE, and GNFE.

Library routines

[Back to Top](#)

The following library routines contain new functionality:

- **CBL_GET_OS_INFO** - this library routine can now detect if the program is running within a Docker container: `cblte-osi-rts-capabilities` parameter, bit 7.

Mainframe Access

[Back to Top](#)

The installation process has been changed as follows:

- A single installer for both Mainframe Access and the Mainframe Access z/Server support is provided.

- There is now only one XML configuration file for Mainframe Access. There is also a new configuration utility that guides you through the configuration process.
- Specify your own JES console character - it is now possible to configure the Mainframe Access installation to use a different JES console character instead of the default ("\$"). To achieve this, use the new JES_CONCHAR configuration parameter.

Enhancements are available in the following areas:

- The Mainframe Call Generator now runs in an independent address space. Previously, it was possible for code run in the MFA address space to enter a supervisor state. MCGLIB is now the new address space's STEPLIB. This enables other user modules to be called without placing them in MFA's STEPLIB, as was previously necessary. See *Mainframe Call Generator: Calling Programs on the Mainframe* for more information.

Managed COBOL

[Back to Top](#)

Deploying JVM COBOL to an Application Server:

- Running JVM COBOL under WebSphere 9.0 and WebLogic 12.2.1 is now supported.

Micro Focus Unit Test Framework

[Back to Top](#)

This release provides support for the following functionality:

- Generation of unit test stubs for selected entry points within your program.

OpenESQL

[Back to Top](#)

This release provides the following new features:

- Support for SQL Server 2017.
- The SQL(TRANSACTION) compiler directive has been enhanced to clearly define transaction boundaries.
- A new SQL(NOWHERECURRENT) compiler directive that allows you to define updateable cursors that do not do positioned updates or deletes with PostgreSQL or MySQL.
- Larger communication area (PID) that accommodates longer plan and program names.
- SQL(OPTIMIZECURSORS) has been enhanced for consistent and better cursor performance across all OpenESQL backends.

Platform support

[Back to Top](#)

Note the following changes in platform support for this release:

- Windows 8 and Windows Server 2012 are no longer supported for developing applications. They are still supported for deployment.
- SUSE platforms - this release only supports SUSE Linux 12 SP2, 64-bit.

On this platform, you can only build COBOL and PL/I programs to 64-bit executables. This applies when using Eclipse on SUSE or with remote projects when Micro Focus Enterprise Developer UNIX Components is installed on SUSE.

PL/I support

[Back to Top](#)



Note: PL/I applications that were created using Enterprise Developer 3.0 and earlier must be recompiled in Enterprise Developer 4.0 to ensure they will compile and execute properly.

Support has been added for the following features in PL/I debugging:

- Displaying complex data structures in the Expressions and Variables views.
- Viewing the console window on non-Windows platforms so that output and input can be seen or entered.
- The Outline view in the default Debug perspective.
- Watchpoints on variables on the line where the variable has been declared.
- Conditional breakpoints.
- Skipping variables a defined number of times before the execution stops.
- Adding expressions to the Expressions view from the Outline view.
- Inspecting variables when referenced as part of a macro argument.
- Signals and conditions.

PL/I compiler and Run-Time System enhancements:

- Support for the following built-in functions - CHARVAL(), COMPARE(), CURRENTSIZE(), EDIT(), PICSPEC(), SOURCEFILE() and SOURCELINE().
- Support for additional DATETIME() format strings.
- Support for the FLUSH FILE(*), ITERATE, and RESIGNAL statement.
- Support for the XMLIMIT and XMOLATTR attributes on variable declarations.
- Support for the MARGINI compiler option.
- Support for the SIZE and STRINGSIZE prefix conditions.
- Improved support for the repeat factor on arrays of AREA variables.
- Improved pointer arithmetic on 64-bit platforms.
- Where necessary, MFPLX now recognizes selected *PROCESS statements and passes them to the appropriate preprocessors.

PL/I EXEC preprocessor:

- The EXEC preprocessor now honors the -margins setting for the output MDECK in addition to the input file. This provides an improved experience in the Eclipse IDE when using the Outline view and other PL/I features provided by the background parser.

PL/I macro preprocessor:

- The Micro Focus macro preprocessor now provides a new default behavior that mimics IBM's RESCAN(ASIS) behavior. In addition, a new option, rescan_upper, has been introduced to enable you to optionally mimic IBM's RESCAN(UPPER) behavior.

Improvements in compatibility with IBM's Macro Preprocessor:

- Allows OTHER as an acceptable abbreviation for OTHERWISE on a macro SELECT statement.
- Supports the SCAN/NOSCAN option on declaration of macro variables.
- Implements the NOSCAN option on ANSWER statements.
- The %NOTE message format has been modified to match IBM's behavior.
- The SYSVERSION built-in function has been modified to always return a value of 22 bytes in length (space padded, if necessary).
- Relaxed diagnostics for scenarios where Enterprise Developer is generating a warning message and IBM does not.

(Technical Preview) Generating PL/I test coverage reports:

- The PL/I Compiler and Run-Time System now support capturing of information showing which lines in the program have been executed, and the generation of human-readable reports at the conclusion of a run.

(Technical Preview) Interaction with COBOL AMODE and 370 Assembler programs (Windows 32-bit only):

- A new compiler option is available which enables Open PL/I to interact with 370 Assembler, and with COBOL programs compiled with AMODE. See the product help for the current limitations to this support.

Web service clients

[Back to Top](#)

This release provides the following enhancements for Web service clients:

- Improved support for larger, more-complex WSDL files, including:
 - Configurable generation of variable arrays and repeated structures by specifying the Inline-Array-Size-Limit parameter.
 - Generation selection for service, port, and one or more operations.
- Support for SOAP version 1.2 in addition to version 1.1.

What was New in Enterprise Developer 3.0

Enterprise Developer 3.0 provided enhancements in the following areas:

- [Integration with Eclipse](#)
- [Application Server JCA support for Enterprise Server](#)
- [Application Workflow Manager](#)
- [Building applications](#)
- [Character encoding](#)
- [CICS Web services](#)
- [COBOL language enhancements](#)
- [Code analysis](#)
- [Code coverage](#)
- [Compiler control](#)
- [Data File Tools](#)
- [Database access - HCO for DB2 LUW](#)
- [Database access - HCOSS](#)
- [Database access - MySQL](#)
- [Database access - OpenESQL](#)
- [Debugging applications](#)
- [Deployment on multiple platforms](#)
- [Documentation](#)
- [Enterprise COBOL 6.1 compatibility](#)
- [Enterprise Server](#)
- [iFileshare](#)
- [IMS support](#)
- [Mainframe Access \(MFA\)](#)
- [Micro Focus Unit Testing Framework](#)
- [PL/I support](#)
- [Rumba terminal emulator](#)
- [Transaction Classes](#)
- [XML processing](#)
- [z/Server](#)

Integration with Eclipse

Enterprise Developer 3.0 provided enhancements in the following areas:

- Support for Eclipse 4.6 - Enterprise Developer now ships with the 64-bit Eclipse 4.6. By default, Eclipse is configured to compile applications for 32-bit and the default target platform for new projects is 32-bit.



Note: Applications that were created using Enterprise Developer 2.3 Update 2 and earlier must be recompiled in Enterprise Developer 3.0 to ensure they will compile and execute properly.

The 32-bit Eclipse is still supported on 64-bit Windows platforms and you need to manually install the 32-bit IDE and a 32-bit Java

You can install the Enterprise Developer plugin in other instances of Eclipse (32-bit or 64-bit). Support for Eclipse 4.2 and 4.3 has been discontinued.

- Rename refactoring - enables you to rename COBOL elements such as variables and identifiers, section and paragraph names, classes, and methods across a program or a workspace. Renaming helps improve the readability of an item or make its purpose clearer. Two preferences for renaming are supported in Eclipse - directly in the editor or using a **Rename** dialog box that offers a preview.
- Dependency paths - it is now possible to specify dependency paths to folders that are outside the project folder.
- Standalone files:
 - It is now possible to specify the locations where the IDE will search for the program symbol files (.idy) for standalone files: **Window > Preferences > Micro Focus > COBOL > Standalone Files > Program Symbols (.IDY)**.
 - It is now possible to use the Eclipse debug configurations to debug standalone files, including core dump files.

In previous versions of Enterprise Developer, standalone files were known as "single files". References to "single files" in the IDE and the product help have been changed to "standalone files".

- Error reporting - errors reported in the **Console** view now include a link for opening the source file at the line where the error originates.
- Editor improvements - an option for removing trailing whitespaces has been added to the IDE preferences for the COBOL and the PL/I editor.

Application Server JCA support for Enterprise Server

In Enterprise Developer 3.0, EJBGEN has been updated to generate an EAR file as a part of the COBOL deployment process, which enables you to deploy EJBs to Java Application Server.

Application Workflow Manager

Enterprise Developer for Eclipse now provides the Application Workflow Manager (AWM) feature. AWM enables you to customize and extend the Eclipse UI and workbench functionality to create application workflows that meet your requirements.

The features comes with:

- A standard project model for local COBOL development that reflects the features available in COBOL Explorer. You can modify this model by adding or removing project features.
- **Application Explorer**, **Filter Definitions**, **Properties**, and **Table Results** views in the COBOL perspective.

Enterprise Developer 3.0 includes improvements in the following areas:

- Application Workflow Manager model editor:
 - The definition of linked element types and properties has been improved. Some required model objects and relationships are now generated automatically.
 - The editor now detects unused references in a model.
- Application Workflow Manager modeling:
 - Basic resource processing has been extended to support mass processing where appropriate.

- The AWM function package now includes the following new tools:
 - "Return Property Value" - enables you to map the values of input parameters to output parameters.
 - "Validate Value" - enables you to validate a filename against a specified naming convention.
- You can now specify enabling conditions in a more granular way. An additional condition type has been introduced to hide actions or properties. For example, hiding an action from the context menu of an element.
- The Eclipse linked element type now enables the label and label decorator to be automatically set to mirror their representation in other Eclipse views if they are not explicitly modeled.
- Table columns can be modeled as a combination of two or more properties - for example, to combine a date and time property value.
- Additional standard icons.
- Mainframe Access (MFA) integration and the MVS function package:
 - MVS resources are now available as linked element types.
 - The following tools have been added to the MVS function package:
 - Delete MVS data set or member
 - Rename MVS data set or member
 - Copy MVS file to clipboard, Paste MVS file from clipboard
 - Get data set or member attributes
 - Improvements have been made to copy and paste support in the MVS Explorer.
 - File mapping between z/OS and local files now supports additional local code pages.
 - The editor context menu contains a "Submit to Mainframe" action when a file is opened from the MVS Explorer with an extension which is mapped to JCL in the File Mappings view.
 - Some of the new MVS function package tools replace existing ISPF function package tools resulting in improved performance of the tools.
 - The sample model "MVS Projects Sample Application" has been improved by including some of the new AWM MVS function package features.
- AWM is now available as a standalone feature which you can install into a separate instance of the Eclipse IDE installed on your machine.

Building applications

Enterprise Developer 3.0 provided the following improvements:

- Support for faster, parallel building on multi-CPU machines - support has been added for multi-processor compilation of the sources in native COBOL projects on multi-CPU machines.

You can specify the maximum number of concurrent compilations from the IDE preferences - **Window > Preferences > Micro Focus > Builder**.

Character Encoding

A new utility, `cobutf8`, is available. `cobutf8` enables you to seamlessly run applications that require non-UTF-8 character encodings in a UNIX environment that is using a UTF-8 locale.

CICS Web Services

From Enterprise Developer 3.0, CICS Web Services was made generally available; it had previously been available (from Enterprise Developer 2.2 Update 2) as a Technology Preview feature.

From Enterprise Developer 3.0, support also includes the following additional features:

- CICS JSON Web services for request/response and RESTful JSON. CICS Web Services support has been extended to allow JSON/REST requests to be used in both requestor and provider mode.
- Use of container-based mappings

COBOL language enhancements

Enterprise Developer 3.0 includes the following enhancements to the COBOL syntax:

- The DISPLAY-OF and NATIONAL-OF intrinsic functions are now able to process conversions using any IBM CCSID value.

The following enhancements are available in managed COBOL:

- To avoid an exception being thrown if an explicit conversion fails, use the AS IF syntax, which results in the target object being set to null and no exception thrown.

Code analysis

Enterprise Developer 3.0 provided the following improvements:

- A new group of predefined rule sets for 64-bit readiness is now included in Enterprise Developer.
- Support for importing code analysis reports produced with one of Micro Focus's advanced tools for code analysis, Enterprise Analyzer or COBOL Analyzer.

Code coverage

The following improvements are available within the IDE:

- Information about unexecuted programs - the code coverage reports in the **Code Coverage** window now show the unexecuted programs.
- Code coverage support for standalone COBOL files - you can import existing code coverage reports in the **Code Coverage** window and use it to supply code coverage information for standalone files.
- Remote code coverage files - in the **Code Coverage** window in Eclipse, it is now possible to import code coverage report files that are stored in remote locations.
- Test coverage files - it is now possible to use a test coverage file to debug applications.

If you are using Test Coverage from the command line, you can now use the following features:

- A new Compiler directive, COLLECTION - the directive enables test coverage to gather information about unexecuted programs. In the IDE, this directive is automatically set on a project when you enable code coverage for it.
- A new command line utility, tcutil - the utility enables you to convert the test coverage binary results file into XML format.
- It is now possible to integrate test coverage in a Continuous Integration (CI) system. You can use tcutil and an XSLT processor to transform test coverage data into a format suitable for including in a CI.

Compiler control

The following Compiler directive are new in this release:

- COLLECTION - provides a mechanism for code coverage to identify unexecuted programs.

The following Compiler directives have been updated:

- ALIGN - this directive has new parameters (FIXED and OPT) that can be used in conjunction with the integer taken, which can aid performance. The default is ALIGN"8 OPT"; see the Comments section of the *ALIGN* Compiler directive topic for details of its affect on memory boundaries.
- ARITH - this directive emulates the IBM mainframe option of the same name. Defines the maximum number of digits for numeric data items.
- FASTINIT - this directive is now on by default when setting the MF dialect; it remains not set by default for other dialects.
- SSRANGE - this directive now has an additional option (3), which permits zero-length reference modified items at run time when bounds checking.
- XMLPARSE - includes a change in the way entities are processed when XMLPARSE"COMPAT" is set

Data File Tools

The Data File Tools editor previously provided (from Enterprise Developer 2.3) as a Technology Preview item was supported at GA level from Enterprise Developer 3.0.

Enterprise Developer 3.0 provided the following enhancements to Data File Tools:

- Opening files in shared mode - it is now possible to switch between read-only shared and edit modes. While a file is open in shared mode, others users can only open it in shared mode to ensure data consistency between users.
- Enterprise Server-level of security when accessing files - there is an improved level of security when exchanging data between Data File Tools and the targeted enterprise server instance. Users must now provide a user ID, group and a password when they try to access and view datasets in enterprise server instances. These are used for authentication and authorization checks to provide the same access level as Enterprise Server.
- Opening datasets using SSL - communication to a region is now possible using SSL. To enable the SSL communication, you need to provide a Java trust store which contains either a CA root certificate or a self-signed certificate of the region that it is communicating to. Java and the targeted region SSL configurations need to meet each other's standards in order for the communication to succeed.

This feature enables you to secure the information exchange between Data File Tools and the targeted enterprise server.

- Auditing of access and updates on datasets - Audit Manager now audits the access and updates on datasets via Data File Tools.
- Support for existing .pro files - enables you to use your existing editor profiles.
- Support for existing .str files - enables you to use your existing COBOL and PL/I structure files.
- Automatic timeout - if no internal operations or external actions (such as a mouse click) have been detected for 30 minutes, Data File Tools now displays a countdown message. If the user does not take any decision within the specified period, Data File Tools closes all opened files.

Database access - HCO for DB2 LUW

Enterprise Developer 3.0 provided a new DB2"QUALIFY-CALL" Compiler directive that enables stored procedure invocations to include a schema name.

Database access - HCOSS

Enterprise Developer 3.0 provided a new Offline Schema Extract feature – a batch solution that migrates a z/OS DB2 catalog to SQL Server without direct access to z/OS DB2.

Database access - MySQL

Enterprise Developer 3.0 provided support for MySQL with ODBC.

Database access - OpenESQL

Enterprise Developer 3.0 provided the following new features:

- Statement prefixes for the SQL"CHECK" Compiler directive that enable the creation of temporary tables and other SQL objects at compile time, ensuring full SQL syntax checking during compilation.
- SQL"OPTIMIZECURSORS" Compiler directive that enhances processing for traditional embedded SQL cursors that use WITH HOLD and FOR UPDATE clauses.
- SQL"CLOSE_ON_COMMIT" Compiler directive to leave cursors open for further result set processing after a commit.
- SQL"GEN-SQLCA" Compiler directive that generates an SQLCA similar to the z/OS DB2 directive STDSQL"YES".

Debugging applications

Enterprise Developer 3.0 provided the following new features:

- Debugging core dump files without a project.

Deployment on multiple platforms

Enterprise Developer 3.0 provided support for deploying JVM COBOL applications on multiple platforms. You can compile an application on one platform (such as Windows) and then deploy its class files to a different platform (such as Linux or UNIX).

Some features of the COBOL language, however, are platform-specific and their behavior on different platforms might vary. See *Multi-Platform Deployment of JVM COBOL Applications* in your product Help for details.

Documentation

The following new sections have been added to the product help:

- *Where do I start?* - located on the launch page of the product help, this section provides the information you need in order to get started depending on which aspects of the product you need to get to grips with first.
- *Multi-Platform Deployment of JVM COBOL Applications* - includes information about how to ensure the portability of your JVM COBOL applications between Windows and UNIX or Linux platforms.
- *Configuring an SSL Client Connection to a Mainframe* - includes instructions about how to configure client applications to connect to MFA and z/Server mainframe via AT-TLS in a zOS environment.

Enterprise COBOL 6.1 compatibility

The following features have been added to support Enterprise COBOL 6.1:

- A new statement, JSON GENERATE, which enables you to convert data to JSON format.
- A new special register, JSON-CODE, and its resulting exception codes to support the new statement.
- Two further new statements, ALLOCATE and FREE, to obtain and release dynamic storage.
- A new MF level, MFLEVEL"19", which contains the following new additions to the reserved words list:
 - JSON
 - END-JSON
 - ALLOCATE
 - FREE
 - INITIALIZED (context-sensitive reserved word)



Note: These keywords are also in effect under the ENT dialect.

- The SSRANGE Compiler directive has an additional option "3", which permits zero-length reference modified items at run time when bounds checking.
- Support for the INITIALIZE statement under the ENT dialect has been enhanced.

Enterprise Server

Improvements are available in the following areas:

Integration with Eclipse

- Exporting an enterprise server definition from the IDE in XML format.
- Importing an enterprise server into the IDE using its definition file.

EXEC CICS TRANSFORM:

- Enterprise Server now supports the XMLTODATA and DATATOXML options of this API.

JCL spool

The JCL spool repository infrastructure has been redesigned to significantly improve its performance. The improvements include:

- A more disk-efficient infrastructure, reducing the space required by the spool files by up to 75% in some cases.
- Improved internal indexing, resulting in quicker generation of messages and sysout records during processing.
- Improved navigation of spool queues: job status **Complete** replaces the **Output**, **Out Hold**, and **Printed** statuses, and completed jobs with any combination of sysout files can be displayed in a single list.
- Increased range of job numbers, from five digits to seven digits.



Important: These changes are automatically in place for new enterprise servers that you create. To enable these updates to existing enterprise servers, you must run a conversion utility before any JCL processing is permitted. Follow the **To run the spool conversion utility** process, detailed in *Significant Changes in Behavior*, for each existing JES-enabled enterprise server you have.

Long user IDs and passwords:

- Enterprise Server now supports user IDs and passwords of up to 100 characters. It is possible to map IDs from long to short (or vice versa) to enable compatibility with programs that do not support long names.

Multi-factor sign-on:

- Multi-factor authentication enables users to sign on to Enterprise Server CICS without entering a username and password. A secure user identifier, such as a certificate, is mapped to a username and is used to generate a passtoken. It is possible to use smartcards, biometrics, or other authentication systems to identify yourself to the system. The username and passtoken can then be used to log on to CICS in an automated way.

The supported sign-on mechanisms are Express Logon Feature (ELF) and Automated Sign-On for Mainframe (ASM). See the product Help for examples of how to configure a region to use the two mechanisms.

SHA-256 support in DemoCA:

- By default, the Demonstration Certificate Authority (CA) now signs certificates with SHA-256. This ensures that the demonstration or evaluation certificates will be accepted by modern browsers and other software that has enhanced security requirements.

Syslog auditing:

- Enterprise Server now supports auditing using syslog events, which can be consumed by a wide range of Security Information and Event Management (SIEM) products. This replaces the Audit Manager auditing solution. Syslog auditing provides a much more efficient auditing mechanism, with significantly less impact on overall speed.

iFileshare

iFileshare, previously considered a Technology Preview feature, was supported at GA level from Enterprise Developer 3.0. It also contained the following enhancements:

- An improved failover and recovery process. iFileshare now supports full recovery of nodes in the group. For high availability (HA-VSAM) groups, servers can now rejoin the group without the entire group having to be restarted. In addition:
 - A primary failover now results in a takeover from the most suitable node.
 - If configured, external clients will automatically reconnect to the new primary and will issue a notification if the transaction has been lost.
 - A failed node, when restarted, will rejoin the group, recover its files and request a log update from the current primary. Once this task has completed it will be considered an active hot-standby and will continue to process replication requests as normal.

- Users will experience a higher level of uptime/availability with their Fileshare configuration and will be able to recover from errors more easily.
- A new exit procedure, `ifsexitproc.cbl`, can be configured to automate some aspects of iFileshare behavior.
- The iFileshare Control page in ESMAC contains details of the current iFileshare high availability group.
- The following new iFileshare-specific environment variables are available:
 - `FSWRKDIR` - enables you to specify the Fileshare working directory, overriding the default, which is the system directory of the region.
 - `FSCHKLFH` - determines if a check is performed when a high availability group is started, to test the consistency of the data files within the group.
- The database reference file (`dbase.ref`) now supports wildcard matching for filenames, allowing you to perform operations on multiple files at once; for example: `fs /d dbase.ref /f data*` adds the entire contents of the data directory to the database reference file.

IMS support

A new MFSGEN Compiler directive, `CONTMVS`, has been added to enable the compiler to parse MFS continuation lines (those continued without a terminating comma) by treating the line that follows as a comment line, just as is done on the mainframe.

Mainframe Access (MFA)

Enterprise Developer 3.0 provided the following improvements:

- Running MFA client in 64-bit processes - you can now call the MFA client modules from 64-bit applications and take advantage of the wider memory addressing available to those modules.
- MFA client CTF tracing - MFDAS now supports CTF tracing. This provides better diagnostics of issues and helps increase turnaround on issues. CTF is common to a number of Micro Focus modules so cross-component logging can be turned on from a single place.
- MFA and z/Server holder close coupling - MFA server can now start the z/Server holder as a subtask in its own address space. This helps improve the performance as MFA can hand-off AWM sessions directly to the z/Server scheduler without instructing AWM to redirect them.
- Endeavor functions - the following functions are now available to Enterprise Sync and MFDAS: DELETE, DIRX, GENERATE, GETENQI, LIST, MOVE, SIGNIN and TRANSFER.

The Micro Focus Unit Testing Framework

The Micro Focus Unit Testing Framework is now available from within the IDE. It includes much of the architecture you would expect of an xUnit framework to create, compile, run and debug unit tests, including the following features:

- A unit test project template.
- A test creation wizard that enables you to generate tests from your source code.
- Code templates for each element of a test case.
- Support for running tests with Code Coverage enabled.
- The Micro Focus Unit Testing view, where you can manage your test runs and view test output.

There has been a number of enhancements to the command line version of the Micro Focus Unit Testing Framework. Support has been added for:

- Running test fixture files using Apache Ant.
- Applying traits to your test cases, then performing a test run based on those traits.
- Applying a high, medium, or low priority to test cases, which affects the order in which they are run.
- Adding coded command line options directly into your test code.
- Using a test run-specific configuration file, in which you can set environment variables.

PL/I support

Enhancements are available in the following areas:

PL/I debugging

- New Eclipse-based PL/I debugger - the CodeWatch debugger has been replaced with the debugger within the industry standard Eclipse User Interface. New features such as true remote debugging are now available along with a much simplified user debugging experience that works in exactly the same way as the debugger works for COBOL (and Java). This can help improve the productivity of PL/I application developers who are adding new features or migrating existing applications to a distributed world.

A selection of features which are now supported includes:

- Attaching to a running PL/I process, waiting for attachment from a PL/I process
- Breakpoint-setting in the editor
- Stepping through PL/I code and querying the contents of PL/I data items
- Viewing the execution stack, resetting the executions point, and running to a selected line
- Remote debugging of PL/I applications running on Linux or AIX machines

PL/I compiler and run-time system

Support has been added to enable the following:

- The UNSIGNED data type - the PL/I Compiler now supports the UNSIGNED attribute for FIXED BINARY data items. In addition to the basic language support, the SIGNED, UNSIGNED, ISIGNED, and IUNSIGNED new built-in functions have been implemented. These features offer an improved compatibility with other languages such as C or C++.
- New built-in functions - UTF8(), UTF8TOCHAR(), UTF8TOWCHAR(), and SUBTRACT(). These new built-in functions make it easier to modernize migrated PL/I applications.
- The ENTRY LIMITED data attribute - the Open PL/I Compiler now accepts and fully supports the LIMITED attribute on ENTRY declarations. This increases the compatibility of the Micro Focus PL/I Compiler with IBM's PL/I for applications that you are migrating from the mainframe.
- Support for processing of JSON data - a number of new built-in functions are available for processing JSON text. This enables you to modernize your PL/I applications by enabling them to interpret, create and process JSON data.

JSONGETARRAYEND	JSONPUTARRAYEND	JSONGETMEMBER
JSONGETARRAYSTART	JSONPUTARRAYSTART	JSONGETVALUE
JSONGETCOLON	JSONPUTCOLON	JSONPUTMEMBER
JSONGETCOMMA	JSONPUTCOMMA	JSONPUTVALUE
JSONGETOBJECTEND	JSONPUTOBJECTEND	JSONVALID
JSONGETOBJECTSTART	JSONPUTOBJECTSTART	

PL/I macro preprocessor

A number of enhancements have been made to improve the compatibility between the PL/I macro preprocessor and IBM's macro preprocessor which helps ease the effort required to migrate PL/I applications to new platforms. Support is available for:

- RESCAN in the ANSWER statement.
- The shorthand "<>" syntax for indicating "Not Equal To".
- Break characters in hex literal constants to improve readability of code.
- A new option, -tagdbg, that annotates macros and %FILE/%ENDFILE statements with line numbers and the source file which injected them.
- The -caseasis option which allows source code to flow through to the mdeck.pp file retaining the case as entered.

- Macros with a null parameter list.
- The LEAVE statement.
- Improved macro preprocessing speed.
- New built-in functions: SYSENDIAN() and MFPHOSTNAME().

Rumba terminal emulator

Enterprise Developer now ships with Micro Focus Rumba 9.5.

[Back to Top](#)

Transaction Classes

Transaction classes, which provide a mechanism to limit the number of CICS® tasks in your system and were previously a Technology Preview item, was supported at GA level from Enterprise Developer 3.0.

XML processing

XML PARSE now works in a purely managed COBOL environment. It is now supported in JVM COBOL and, in both .NET and JVM COBOL, it has a fully managed implementation. XML PARSE working without calling out to native code ensures it can be used in restricted rights environments.

z/Server

The z/Server configuration has been simplified through the use of a unified scheduler. This replaces the three different types of scheduler, STC, CEA and JOB, that were used previously for starting different types of user servers.

What was New in Enterprise Developer 2.3 Update 2

Enterprise Developer 2.3 Update 2 provided enhancements in the following areas:

- [Integration with the Eclipse IDE](#)
- [Application Workflow Manager \(AWM\)](#)
- [COBOL language enhancements](#)
- [Classic Data File Tools](#)
- [Compiler directives](#)
- [Database access - DSNREXX](#)
- [Editor writing assistance](#)
- [File handling](#)
- [IMS support](#)
- [Interface Mapping Toolkit](#)
- [Library routines](#)
- [PL/I support](#)
- [Tutorials](#)

Integration with the Eclipse IDE

Enterprise Developer 2.3 update 2 provided enhancements in the following areas:

- The following settings have been added to the BMS page in the project's and file properties - Redefine Input, SDF Map Size, and Hyphenate Field Suffix.
- Using the search facility, **Search > Micro Focus**, you can now limit the search to the copybooks used by the current program rather than all copybooks in the project or the entire workspace.
- The editor provides a new context menu command, **Extract COBOL code to copybook**, that enables you to move a selected segment of the code to a new copybook file in your project. The segment of code moved to a copybook is replaced with a COPY statement in the original program.

- Support for Eclipse 4.5 - after installing Enterprise Developer, you can add it as a plugin into a separate instance of Eclipse 4.5. See your product's *Installation* notes.

Application Workflow Manager (AWM)

Enterprise Developer 2.3 update 2 provided enhancements in the following areas:

AWM model editor:

- It is now possible to use drag and drop not only for objects of the same type but in all cases where this is relevant. For example, it is now possible to drag tool descriptors onto an action descriptor which will create an "Action Descriptor has Tool" relationship. Dragging a File_Descriptor_has_Property relationship onto an element list structure will create an Element_List_Structure_has_Column relationship.
- It is now possible to group any major modelling objects such as action descriptors, properties and tool descriptors in the AWM model editor.

AWM modelling:

- The Eclipse function package includes a new tool, "Build Project", and the MVS function package includes a new tool, "Refresh DataSet".
- "Edit MVS File" now provides a parameter to the editor ID.
- The AWM meta model has the following enhancements:
 - New model objects for configuring the decoration of labels displayed in the Team Developer Tree view.
 - Visible conditions for enabled conditions. You can use the visible conditions to specify whether an action should be shown on an element's context menu.
 - A new application-specific attribute, INITIAL_PARSE, enables you to specify whether the background parse process of the editor should start when a file is opened in the editor.
 - The Eclipse function package includes a new category of linked properties that support the Eclipse team properties.
 - The Eclipse linked element types now support the "Run As", "Debug As" and "Profile As" context menu actions.

AWM z/OS extensions:

- The Micro Focus z/Server connection now provides an Owner filter.
- The Micro Focus z/Server connections support case sensitive input of passwords.
- The IDz function package provides a new tool which opens MVS resources depending on the RSED connection defined in the Micro Focus z/Server connection. This now enables you to download copybooks in IDz environments using the Micro Focus z/Server connection.

Micro Focus model extensions:

Support is available in the Micro Focus model extensions (the Enterprise Development Projects model and the SCM models for Endeavor, ChangeMan, SCLM and SVN) for:

- The Eclipse team linked properties
- Grouping modelling objects in the editor
- Visible conditions
- The "Run As", "Debug As" and "Profile As" commands

COBOL language enhancements

Numeric, edited and external floating point items can now specify USAGE NATIONAL when the NATIONAL"2" Compiler directive is in effect. Signed numeric items must be specified with the SIGN IS SEPARATE clause.

Classic Data File Tools


A new command line utility is available which enables you to initiate the following actions: open data files, create or open record layout files, create or open segment layout files, and open IMS databases using a DBD or PSB file. Note that although you can initiate these actions from the command line, you must complete them from within the IDE.

Compiler directives

The following Compiler directives are new in this release:

- **COMMAND-LINE-LINKAGE** - enables you to call a program and pass the command line to the main program as a parameter to be accessed via the Linkage Section. This offers equivalent functionality to the `command_line_linkage` tunable, which has now been deprecated.
- **EBC-COL-SEQ** - controls the behavior of an EBCDIC collating sequence, specified in a `NATIVE"EBCDIC"` program. `EBC-COL-SEQ"1"` (the default) maintains use of the long-standing fixed (platform-independent) EBCDIC collating sequence. `EBC-COL-SEQ"2"` prompts use of the latest CODESET table, which varies according to platform and user-controlled MFCODESET environment variable setting.
- **NATIONAL** - enables you to specify numeric, edited and external floating point items as `USAGE NATIONAL`.

Database access - DSNREXX

 **Note:** This is a technology preview feature only. It is being made available to allow you to test and provide feedback on this new capability; however, this feature is not intended for production use and it is not supported as such.

DSNREXX support is available for use with Enterprise Server using DB2 LUW and SQL Server. See *DSNREXX Support* for more information.

Editor writing assistance

Enterprise Developer 2.3 update 2 provided the following enhancements:

- Colorization of conditional compilation regions - by default, inactive code is now colored the same as Compiler directive elements. Inactive code is defined as code within conditional blocks that do not evaluate with the applied Compiler settings.
- `$REGION` statement - support is provided for the `$REGION` Compiler-control statement. You can use `$REGION - $END-REGION` to surround blocks of code that you want to fold or expand in the editor.
- AutoCorrect - you can configure the editor to automatically fix the most frequently misspelled words. You use the IDE preferences to specify a list of words that you sometimes mistype, and the correct spellings for them. Whenever you misspell that word, the editor automatically replaces it with correct version. This feature is enabled by default and can be configured from a new preference page in **Window > Preferences > Micro Focus > COBOL > Editor > AutoCorrect**.

File handling

MFJSORT ICETOOL now supports the `USING` parameter in the `SELECT` operator.

IMS support

Enterprise Developer 2.3 update 2 supports the `PROMPT` keyword in MFS source files.

Interface Mapping Toolkit

Enterprise Developer 2.3 update 2 supports JSON schemas for the generation of REST Web service clients in the Eclipse IDE.

Library routines

The following library routine contains new functionality:

- `CBL_GET_PROGRAM_INFO` - a new function (function 10) has been added for native COBOL which returns the path and program name, or the program name only of a particular program.

PL/I support

Enterprise Developer 2.3 update 2 provided the following enhancements:

- New built-in functions - support has been added for the `ONCONDID()` and `SAMEKEY()` built-in functions.
- Support for strings declared with the `VARYINGZ` attribute - the Open PL/I compiler and run-time system now support the use of `VARYINGZ` strings as variables, parameters, and targets for built-in functions. This provides an improved compatibility with IBM's mainframe compiler, and the ability to more easily interact with other languages such as ANSI C.
- Improved PL/I macro preprocessor compatibility - the Open PL/I compiler now supports all macro preprocessor built-in functions used by IBM's PL/I Macro Preprocessor.
- Improved compatibility with IBM's Structure Alignment rules - the `-zalign` option of the Open PL/I compiler now maps structure elements of the `BIT` data type that do not consume a full byte to the same boundaries as IBM's PL/I compiler. This helps make the migration of mainframe data files to a distributed platform easier and improves the program execution compatibility.
- Improved floating point support - the Open PL/I Code Generator now utilizes the newer Intel floating point operands for improved speed. In addition, the Code Generator now provides improved compatibility with passing parameters between C and COBOL.
- PL/I editor improvements:
 - A new context menu command in the editor, **Extract PL/I code to an include file**, which enables you to move a selection of code in the editor into a new include file. The extracted code is replaced with an `%INCLUDE` statement which references the include file.

Tutorials

The product help includes the following new tutorials:

- *Tutorial: SQL - Deploying an Enterprise JavaBean Containing JVM COBOL to a JBoss Application Server* - that walks you through the process of deploying an EJB that contains JVM COBOL code.
- *Tutorial: DSNREXX*, that walks you through the process of executing DB2 LUW queries using DSNREXX.

What was New in Enterprise Developer 2.3 Update 1

Enterprise Developer 2.3 Update 1 provided enhancements in the following areas:

- [Integration with the Eclipse IDE](#)
- [Application Server JCA support for Enterprise Server](#)
- [Application Workflow Manager \(AWM\)](#)
- [CICS Web services \(Technology Preview\)](#)
- [Code Analysis](#)
- [Code Coverage](#)
- [Compiler directives](#)
- [Database Access - HCO for SQL Server](#)
- [Database Access - Mainframe Batch Database Tools \(MBDT\)](#)
- [Data File Tools](#)
- [Editor writing assistance](#)

- [Enterprise COBOL 5.2](#)
- [File Handling](#)
- [Library routines](#)
- [Managed COBOL Syntax](#)
- [Native COBOL Syntax](#)
- [PL/I support](#)
- [Rosetta Stone for COBOL, .NET and Java Developers](#)
- [SourceConnect](#)
- [z/Server](#)

Integration with the Eclipse IDE

Enterprise Developer 2.3 Update 1 provides the following enhancements in the integration of Micro Focus COBOL with the Eclipse IDE:

Editors:

- COBOL editor:
 - You can now specify the increment for the COBOL and the standard line numbering from **Window > Preferences > Micro Focus > COBOL > Editor > Line numbering**.
 - You can now toggle single or multiple lines between commented and uncommented states.
- JCL editor - the Unnumber and Renumber commands are now supported for the JCL editor. You can specify the line number increments from the IDE preferences in **Window > Preferences > Micro Focus > JCL > Editor > Line numbering**.
- PL/I editor:
 - The Unnumber and Renumber commands are now supported for the PL/I editor. You can specify the line number increments from the IDE preferences in **Window > Preferences > Micro Focus > PL/I > Editor > Line numbering**.
 - You can now toggle single or multiple lines between commented and uncommented states.

Building applications:

- Environment variables - a new page, **Build Environment**, in the project's properties enables you to specify environment variables for your applications. You can also specify environment variables that only apply at run time on the run or debug configuration for the application.

Application Server JCA support for Enterprise Server

[Back to Top](#)

This release provides support for:

- Automatic connection recovery to an active Java application server when an enterprise server region is restarted. This applies to:
 - COBOL resource adapters
 - CICS resource adapters
 - All CICS API based programs
- The IBM implementation of CICS resource adapter for WebSphere 8.5 and JBoss 7.1.1.

Application Workflow Manager (AWM)

Enterprise Developer 2.3 Update 1 provides enhancements in the following areas:

New features and enhancements are available in the following areas:

- AWM modelling:

- It is now possible to associate keyboard shortcuts with modeled actions. The key sequence is displayed in the corresponding context menu at run time.
Currently, it is not possible to persistently overwrite any modelled key bindings with user key bindings.
- Support is provided for table row updates in modeled AWM dialogs - it is now possible to use table row actions in the modeled AWM dialogs to modify table rows similarly to how this is done in the Team Developer Tree or in the Table views. The supported modifications are copy, delete, insert and update of selected rows.
- It is now possible to hide element properties in an AWM element's property view using the "Element Type has Property" condition.
- The "Get Children" attribute of an element type has been replaced by the "Element Type has Get Children Action" relationship between an element type and an action descriptor which enables you to check whether an AWM container element can be expanded.
- Full support and a seamless integration is now available in the IDE for the AWM SCM models for ChangeMan, Endeavor and SCLM. You can now check out sources from the SCM system directly into your COBOL or PL/I projects in the IDE and vice versa.
- AWM Run-Time:
 - A single port for server connection - starting with this release you now only need to specify one port, the Mainframe Access (MFA) server port, for the **Micro Focus z/Server** connection type in the Remote System view. All other required ports are configured centrally in the server configuration on the mainframe. You must install the latest version of z/Server.
It is still possible to connect to an earlier supported version of z/Server. There is a backward compatibility flag that must be enabled to be able to specify the three ports that are required with the earlier z/Server version.
 - SSL support - it is now possible to specify a SSL/TLS configuration when defining a new **Micro Focus z/Server** connection in the Remote System view. The client then uses the configured SSL/TLS certificates for the communication with the server. The mainframe must support SSL/TLS tunneling (such as AT-TLS from IBM).
 - RACF passticket support - AWM clients now include an extension point which you can use to provide user credentials from an external application. When connecting to a Micro Focus mainframe server system (either MFA or z/Server), the password dialog is then bypassed. This enhancement enables you to integrate a login mechanism that uses RACF passtickets instead of RACF passwords.
 - Support for binary download and upload of files - it is now possible to specify a Transfer Mode (either text or binary) in the file mapping dialog of the File Mappings view when transferring files between the mainframe and the local machine such as when editing files or during a drag and drop operation:
 - Text - the configured code pages are respected.
 - Binary - no code page translation is applied. You must use binary mode when transferring mainframe load modules.
 - The SVN attachment is now an integrated part of the Enterprise Developer and is installed as part of the product. This enables you to use the tools from the SVN function package in an AWM model. For example, you can extend an existing model by adding the check-in and check-out functions for sources in an SVN repository. A sample model that includes a full integration of SVN support in the Enterprise Development Project model is available.
 - A new AWM "MVS Projects Sample Application" is available with the installation of z/Server. The application uses the tools from the MVS and ISPF function package and is designed as a template AWM application for project-based access to MVS data.
- The AWM Enterprise Development Projects Application now provides the following enhancements:
 - Web Services support for COBOL applications in the Team Developer perspective.
 - Context menu commands for performing code analysis on demand of the sources in your projects from the Team Developer Tree and Table views.
 - An **Open in context** action for COBOL copybooks in the Team Developer Tree and Table views.

- An Import and Export are wizards from the context menus of the Team Developer Tree and Table views.
- Error markers in the Team Developer Tree and Table view on project, folder and file level when the sources compile with errors.
- A "Link with Editor" function in the Team Developer Tree view for such files that you viewed when you expanded the tree view.
- Support for using filters (using **Customize View > Filters**) in the Team Developer Tree and Table views to specify what entries should be displayed.
- An improved support for drag and drop between the Team Developer Tree and Table view and the standard Eclipse views (such as Project Explorer and the Navigator view), the Remote Systems view (from the local and the Micro Focus z/Server connections), and Microsoft's Windows Explorer.
- The product help now includes a new section, *AWM API Programming and Extension Points*, that describes the published AWM API classes and interfaces and the extension points provided by the AWM client. Use cases for the usage of the AWM API and extension points and a detailed tutorial showing how to extend the main AWM extension point, `com.microfocus.awm.model.extension`, are also available.

CICS Web services (Technology Preview)



Note: This was introduced as a technology preview feature only. It is being made available to allow you to test and provide feedback on this new capability; however, this feature is not intended for production use and it is not supported as such.

This release provided an initial support for CICS Web services (CWS) using an SOAP-based provider and requester modes and using HTTP or MQ as the transport mechanism. CICS Web services support the Coded Character Set Identifier (CCSID) and configuring the CCSID is a prerequisite for CWS in all environments.



Restriction: Security and Web Service Atomic Transactions (WSAT) are not currently supported.

Code analysis

[Back to Top](#)

This release provides support for performing code analysis at the command line using Ant which enables the integration of code analysis in CI frameworks. Features include:

- Support for performing code analysis at the command line using the project's `.cobolBuild` Ant script and specifying a target.
- New Ant targets for code analysis - `analyze` and `build.and.analyze`. These enable you to only run analysis and produce analysis data or to build and produce build artifacts as well as analysis data.
- New Micro Focus Ant task, `analysis`, for the `.cobolBuild` file. A parameter for this task enables you to specify whether the build fails or continues when code analysis results are received.
- New parameters for code analysis for the cobol Ant task - `analysisData`, `analysisDataDir`.
- New Ant type, `ruleList`, for the `.cobolBuild` file - enables you to specify the rules to execute.
- Support for running analysis using a custom `.cobolBuild` file from outside of the project directory or the workspace.

See the [Micro Focus Ant User Manual](#) in the Micro Focus Infocenter for more details on the new task, types and parameters.

Code coverage

[Back to Top](#)

The code coverage reports are now integrated with the IDE and with the editor. Features include:

- A new Code Coverage view (Eclipse) showing the statistics of what percentage of the code has executed.
- Navigation from the Code Coverage view (Eclipse) to the missed and covered blocks in the editor.
- Colorization in the editor of blocks that were executed (covered blocks) or not (missed blocks).

Compiler directives

The following Compiler directives are new in this release:

- **ILMAIN** - you now specify the main entry point for the executable program, which can be specified either as `class-name::method-name`, or just as `method-name`. For example, `ILMAIN"classA::methodB"` or `ILMAIN"methodB"`. The first format can be used to distinguish between multiple methods with the same name in different classes.

This directive is now available for JVM COBOL.

- **OOCTRL** - a new parameter, `+/-A`, as been added. Set this parameter to `-A` to allow ActiveX controls in your COBOL application to use classes and methods in the OLE class library. The default is `+A`, which does not allow it

Database Access - HCO for SQL Server

[Back to Top](#)

This release provides the following new features:

- End-to-end demonstrations that show several methods for submitting a JCL batch job that calls COBOL stored procedures. To view these demos, see the *SQL* demonstrations in the Mainframe Samples browser.
- The **SPCALLLOCAL** SQL compiler directive option, used to convert an `EXEC SQL CALL` into a call to a local COBOL subroutine instead of a call to a SQL Server `SQLCLR` stored procedure.
- Extended support for conversion of the DB2 `CONCAT` function.

Database Access - Mainframe Batch Database Tools (MBDT)

[Back to Top](#)



Note: This is a technology preview feature only. It is being made available to allow you to test and provide feedback on this new capability; however, this feature is not intended for production use and it is not supported as such.

This release provides MBDT support for DB2 LUW partitioned tables.

It also provides support for 32-bit MBDT for use with DB2 LUW on the following platforms:

- IBM System p running AIX
- x86-64 running SUSE Linux
- x86-64 running Red Hat Linux

Database Access - SQL Option for DB2

[Back to Top](#)

This release provides the following new features:

- Import/export of XML with SQLWizard/DSNUTILB(XUTLB)
- New XDBSetup utility for TS/Citrix environments
- Using Migrate to move XML tables from z/OS DB2 to XDB
- Support for Temporal Tables

Data File Tools

This release provided improved security and increased support for more file types. Features include:

- Certain aspects of Enterprise Server security are honored when you attempt to access data sets. If the Enterprise Server region has security enabled, logon details must be authenticated before you can access the data set. If the details are unable to be authenticated, access is denied.
- When using a record layout, certain data is now validated at field level (to ensure the contents is compatible with its picture string) and record level (to ensure the record length matches the layout size).
- Full editing support has been added for variable block sequential files and relative files. Full editing is also available for line sequential files, as long as they do not contain any binary data

Enterprise COBOL 5.2

[Back to Top](#)

With the introduction of Enterprise COBOL 5.2, the following features were supported:

- The VOLATILE keyword is supported within the data entry description; although, this is treated as documentary. It has also become a reserved word when under the ENTCOBOL dialect.
- Format 2 of the SORT statement no longer treats the COLLATING SEQUENCE clause as documentary-only.
- The SUPPRESS clause of the XML GENERATE statement has been enhanced.
- The IBM z/OS JSON parser API, as documented for the IBM z/OS client web enablement toolkit.

File Handling

- A new indexed file format, IDXFORMAT12, has been introduced to improve file maintenance and recovery procedures when using the rebuild utility. This file format is similar in structure and use to IDXFORMAT8. Where the two formats differ is that an IDXFORMAT12 file has an accompanying side file (.idx file) containing the indexed key information.

You can use this type of file with the new `rebuild /q` option. This rebuild process is considerably quicker than other rebuild processes such as a data scrape or `rebuild /p`.

- Faster SORT operations for fixed block records - when using the DFSORT emulation, the performance when sorting fixed block records has greatly improved.

Library routines

The following library routine were new at this release:

- CBL_CODESET_SET_MAPPING - enables you to change the codeset in effect.
- CBL_RUNTIME_ERROR - forces an application to terminate with a run-time error condition.

Managed COBOL syntax

[Back to Top](#)

The following enhancements have been made to the managed COBOL syntax:

- A new command line utility, `mfjarprogmapp`, is available to allow you to create the necessary Java property file when calling COBOL programs that have been compiled as part of a package.
- You can now create generic iterators.
- You can now use the Profiler utility to obtain detailed statistics on the run-time performance of managed COBOL applications.

Native COBOL Syntax

[Back to Top](#)

The following items are new features of the native COBOL syntax:

Class condition tests New and updated class condition tests are available for DBCS, KANJI, and JAPANESE.

PL/I support

Enterprise Developer 2.3 Update 1 provides the following enhancements:

- FIXED BIN(63) data type - PL/I programs that utilize FIXED BIN data types that are larger than FIXED BIN(31) are now supported which enables you to migrate such programs more easily to distributed platforms.
- TRIM() built-in functions - the Open PL/I macro preprocessor now supports the TRIM() built-in function.
- A 64-bit Open PL/I peephole optimizer - when compiling 64-bit Open PL/I programs optimization for speed, a peephole optimizer is now run against the intermediate code to improve your applications' performance. Any code which has attributes that benefit from peephole optimizations will then run faster.

Rosetta Stone for COBOL, .NET and Java Developers

[Back to Top](#)

The product Help now includes a quick and easy to use syntax guide for developers who need to learn OO COBOL syntax when modernizing COBOL applications for the Java or .NET platforms. The guide includes side-by-side equivalent syntax for COBOL, C#, VB and Java.

SourceConnect

[Back to Top](#)

It is now possible to use SourceConnect to link to mainframe locations and to define mainframe datasets and members. The datasets and members then appear in your project's directory structure within the IDE.

z/Server

[Back to Top](#)

The z/Server Configuration Utility is now installed as part of Enterprise Server and is not a Technology Preview download. The user interface has been streamlined for the creation of a default working configuration.

What was New in Enterprise Developer 2.3

Enterprise Developer 2.3 provided enhancements in the following areas:

- [Integration with the Eclipse IDE](#)
- [General IDE enhancements](#)
- [Building JVM COBOL Projects Incrementally](#)
- [COBOL Editor in Eclipse](#)
- [Code analysis](#)
- [Code coverage](#)
- [Command Line Compilation and Linkage](#)
- [Compiler directives](#)
- [Data File Structure command line utility](#)
- [Data File Tools \(Technology Preview\)](#)
- [Database access](#)
- [File locking](#)
- [File handling](#)

- [Library routines](#)
- [Managed COBOL syntax](#)
- [Micro Focus Infocenter](#)
- [Micro Focus Unit Testing Framework](#)
- [Preprocessors](#)
- [Profiler](#)
- [Remote Connections](#)
- [REST service interfaces](#)
- [RM/COBOL Compatibility](#)
- [Single file support](#)
- [Tunables](#)
- [Updated run-time system](#)

Integration with the Eclipse IDE

This release shipped with Eclipse version 4.4.2.

Support for Eclipse versions 3.7 and 3.8 has been deprecated. This affects any applications that were created using an earlier version of Enterprise Developer that have a JVM COBOL part. Such applications must be rebuilt using Enterprise Developer 2.3 to avoid receiving errors during compilation or execution.

In addition, when Enterprise Developer is installed, you can optionally install the Enterprise Developer plugin in other instances of Eclipse installed on your machine (supported versions of Eclipse are 4.2, 4.3 and 4.4 for the 32-bit IDE).

General IDE enhancements

In Eclipse:

- The **Variables** view now shows the file status of an internal file name.
- The **Find All References** and **Go To Definition** commands are now supported for JVM COBOL applications.
- Support is provided for the SOCKS5 proxy server for debugger communication over SSL.

Building JVM COBOL Projects Incrementally

To minimize the number of modules to compile when building JVM COBOL projects, Enterprise Developer now enables you to configure your projects so that they are built incrementally - the IDE only rebuilds the files that have changed.

To enable incremental builds for JVM COBOL applications that contain namespaces, check the **Use incremental build (Technical preview feature)** option on the project build configuration tab in the project's properties.

For JVM COBOL applications that do not contain namespaces, you can use the **Use dynamic calls** option available on the build configuration tab in the project's properties. When the project is built with this setting, calls to modules are resolved at run time rather than during compilation. This has the effect of not requiring every module to be compiled when rebuilding the application.

COBOL Editor in Eclipse

Content Assist support for COBOL includes various enhancements and is now also available in JVM COBOL:

- Context sensitive proposal - Content Assist only shows proposal that are relevant for the position of the cursor in the code or for the type of project
- Enhanced proposal lists - lists include any relevant COBOL verbs, clauses and words, copybooks, code templates, data items and section and paragraph names

- Intelligent assistance with completing statements - when you have entered a COBOL verb, Content Assist shows proposals for the relevant clauses and identifiers that you can use to complete the statement.
- Automatic completion for items - Content Assist automatically inserts single suggestions in the code.
- Qualifying non-unique names - Content Assist qualifies data items whose names are not unique.
- Configuration preferences for Content Assist - enable you to configure what suggestions appear in the completion lists, whether suggestions are added in insert or overwrite mode, and the case of the inserted words.
- Code templates - code templates are now included in the Content Assist proposals.

Code analysis

Enterprise Developer now offers more advanced code analysis features and enables you to run various analysis queries (rules and groups of rules called rule sets) against your code to ensure adherence to standards such as standards for coding or performance.

You can run analysis rules against programs in a project in the IDE at user request or you can run analysis rules at the end of a project's build.

Code coverage

Enterprise Developer now provides support for code coverage of native COBOL applications directly from within the IDE where code coverage uses the Test Coverage functionality. You can produce code coverage reports for applications running in the COBOL run-time and for applications that run in Enterprise Server.

To produce reports, you need to enable code coverage in a project's, a build configuration's, or a file's properties, compile your application and then run your application with code coverage to produce the relevant reports. For applications that require an Enterprise Server instance, you start the enterprise server with code coverage.

Command Line Compilation and Linkage

When using the `cbllink` command to compile and link, there is a new `-y` option. Use this option to create an executable that includes support to be able to run on Windows XP and Windows Server 2003.

Compiler directives

The following Compiler directives are new in this release:

EOF-1A	Treats a 0x1a character in the source file as the end of file.
JVMDECIMAL	Determines the type in which certain items are exposed. This directive affects COBOL data items of type 'decimal' and non-integral numeric items exposed as a result of either ILSMARTLINKAGE usage or the PROPERTY keyword.
NLS-CURRENCY-LENGTH	Specifies the number of bytes to allocate for the currency symbol in a PIC field.
NULL-ESCAPE	Treats a 0x00 character in the source file as an escape character for other non-printable characters in the source code.

The following Compiler directives contain new parameters in this release:

DBSPACE The new parameter "MIXED" extends the DBSPACE directive to be able to evaluate data items in programs that contain a mix of single-byte and double-byte strings.

Data File Structure command line utility

The Data File Structure Command Line (DFSTRCL) utility is a DOS-based command line utility that enables you to create record layout (`.str`) files from COBOL debug information (`.idy`) files. You can use the utility to process a single `.idy` file or batch process up to 100 `.idy` files.

Data File Tools (Technology Preview)



Note: This is a technology preview feature only. It is being made available to allow you to test and provide feedback on this new capability; however, this feature is not intended for production use and it is not supported as such. Furthermore, Micro Focus does not guarantee that this feature will be delivered at a GA level and if it is, then the functionality provided might differ considerably from this technology preview.

The Data File Tools (Technology Preview) is a new standalone text editor in which you can create and edit data files. By nature of it being a 'technology preview' product, it does not currently include all the functionality that was available in the previous version of Data File Tools - now referred to as Classic Data File Tools. If you require any of the functionality not provided in this version, you can still use the classic version by accessing it in the usual way.

To run Data File Tools (Technology Preview), type `mfdatatools2` from Enterprise Developer's command prompt (Windows) or a terminal (UNIX).

To use the new editor directly from the Eclipse IDE, use the **Open with** option on the shortcut menu when selecting a data file or structure file, and select **Data File Tools**. Eclipse remembers the last tool used for a particular file type, and so will use Data File Tools (Technology Preview) until you select a different editor.

Database Access

Enterprise Developer version 2.3 provides the following enhancements to database access:

COBSQL Enterprise Developer version 2.3 provides:


- Selection and configuration of the Oracle Pro*COBOL preprocessor for compiling COBSQL applications in project properties on the **SQL Preprocessor** tab and in the build configuration settings.
- Support for COBOL directives `SOURCEFORMAT=TERMINAL` and `SOURCEFORMAT=VARIABLE` for Pro*COBOL applications.

HCO for DB2 LUW Enterprise Developer version 2.3 provides:

- GUI versions of data migration tooling (Windows platforms)
- A new tool that enables you to bind or rebind all packages. See *BindAll Packages Tool* for details.
- Support for MFHCO mode across all platforms by default via the new HCO "NOHCO" DB2 compiler directive option. See the *HCO DB2 compiler directive option* topic for details.
- A new DB2 compiler directive option, `OPTPER "NOOPTPER"`, that enhances performance for `CHARSET EBCDIC` processing. See the *OPTPER DB2 compiler directive option* topic for details.
- A new DB2 directive option, `BINDDIR`, which specifies an alternative directory in which to write the DBRM file created during compilation. You can set `BINDDIR` from the command line or specify it in your project properties. See the *BINDDIR DB2 compiler directive option* topic, and the *Binding* topic for details.
- 64-bit support for PL/I on appropriate platforms. See *Additional Software Requirements on Windows* and *Additional Software Requirements on UNIX* for details.

HCO for SQL Server Enterprise Developer version 2.3 provides:

- Support for the `DATA-CHANGE-TABLE-REFERENCE` clause.
- Context Help is now provided for:
 - Options for HCO for SQL Server tools
 - For each tool in the HCO for SQL Server user interface

OpenESQL	Date/Time Processing	This release provides streamlined datetime processing for ODBC and JDBC.
	OpenESQL Assistant	OpenESQL Assistant Options are now set via the Eclipse IDE from Window > Preferences > Micro Focus > Database > OpenESQL Assistant (applies to Windows environments only).
	Performance	This release includes a new SQL compiler directive option, OPTPER "NOOPTPER", that enhances performance for CHARSET EBCDIC processing. See the <i>OPTPER</i> SQL compiler directive option topic for details.
	PL/I	This version provides 64-bit support for PL/I on appropriate platforms. See <i>Additional Software Requirements</i> for details.
	PostgreSQL	<p>In this release, PostgreSQL 9.4 has been tested with OpenESQL and OpenESQL Assistant with the following PostgreSQL software:</p> <p>Server software PostgreSQL EnterpriseDB version 9.4.1-3</p> <p>Client software</p> <ul style="list-style-type: none"> • psqLODBC driver version 09.03.04.00 • JDBC41 PostgreSQL driver version 9.4-1201 <p>PostgreSQL 9.4 has been tested with OpenESQL and OpenESQL Assistant on the following Windows platforms:</p> <ul style="list-style-type: none"> • Windows 32-bit • Windows 64-bit <p>PostgreSQL 9.4 has been tested with OpenESQL on the following UNIX platforms:</p> <ul style="list-style-type: none"> • X86-64 running Red Hat Linux, 32- and 64-bit • X86-64 running SuSE Linux, 32- and 64-bit <p> Note: Micro Focus provides compatibility for PostgreSQL but does not directly contribute to or support the PostgreSQL open source project. Any issues relating to PostgreSQL functionality should be addressed through an open source support vendor.</p>
	SQL Server	Enterprise Developer version 2.3 provides support for the SQL Server OUTPUT clause.
Mainframe Batch Database Tools (MBDT)	Windows only. Enterprise Developer version 2.3 includes mainframe batch utilities that provide the following functionality for Oracle, DB2 LUW, and SQL Server agnostically, as well as specifically for HCO for SQL Server:	<ul style="list-style-type: none"> • DSNTEP2 • DSNUTILB LOAD • DSNUTILB UNLOAD • 32-bit and 64-bit support • New batch configuration utility
SQL Option for DB2	Enterprise Developer version 2.3 provides the following z/OS DB2 version 10 enhancements:	<ul style="list-style-type: none"> • Extended support for implicit casting • Greater timestamp precision for applications

- Support for `TIMESTAMP WITH TIME ZONE`

This release provides initial XML support for the XDB engine.

XA Switch Modules

In this release, the XA interface has been redesigned to provide:

- Consistent look and feel for SQL Server, DB2, and Oracle user personalization
- Consistent look and feel for both RM dynamic and static registration (SQL Server, DB2, Oracle, generic one-phase commit)
- Additional support for two instances of the same switch module using Web Services applications via the new XAID compiler directive
- Using a specified XA resource only with batch applications executing under Enterprise Server

File handling

This release contains the following new configuration options:

ACUFH Enables or disables the use of the ACU file handler (ACUFH), which is required to handle Vision and RM/COBOL indexed files.

ESACUFH Enables or disables the use of the ACU file handler (ACUFH) for file handling operations running under Enterprise Server. ACUFH must also be enabled for this option to take effect.

File Locking

In versions prior to Enterprise Developer 2.3, the semantics of the sharing phrase specified in an `OPEN` statement or used within a call to `CBL_OPEN_FILE` were not correctly applied in some cases on UNIX and Linux platforms. From version 2.3 onwards, the sharing phrase is correctly honored when the tunable `strict_file_locking=true` is set, which is the default setting.

Example of potential changes in behavior:

- *Process-A* opens a file with read-only access and a sharing mode that denies other processes write access (`SHARING WITH READ ONLY`).
- *Process-B* then attempts to open the file with read-only access and a sharing mode that denies other processes read access (`SHARING WITH NO OTHER`).

With `strict_file_locking=true`, *Process-B* is unable to open the file, because *Process-A* has successfully opened the file allowing only read access.

With `strict_file_locking=false`, *Process-B* successfully opens the file.

If your application encounters unexpected `OPEN` conditions or fails to open files, it might be as a result of the new file locking behavior. In such circumstances, we recommend that you review the file locking and sharing requirements of your application and refactor your source code to work with the default setting. The original file locking and sharing behavior can be restored by setting `strict_file_locking=false`.

Library routines

The following library routines are new in this release:

CBL_MANAGED_SESSION_GET_USERDATA Retrieves user data saved in the current RunUnit.

CBL_MANAGED_SESSION_SET_USERDATA Sets user data in the current RunUnit.

The following library routines contain new parameters in this release:

CBL_LOCATE_FILE You can now specify a file name that is a null-terminating string, which has resulted in three new values available for the `user-mode` parameter.



Managed COBOL syntax

The following enhancements have been made to the managed COBOL syntax:

- The `TYPE OF type-name[ANY...]` syntax enables you to obtain the `System.Type` (.NET) or `java.lang.Class` (JVM) object for a generic class, interface, or delegate.
- The `self::` or `super::` syntax is no longer required to access inherited data within a subclass.
- The `ATTRIBUTE-ID` syntax enables you to define new attribute types, which can be used in various contexts.

Micro Focus Infocenter

The Micro Focus Infocenter Web site (<http://documentation.microfocus.com>) has been upgraded and now includes the following improvements:

- Scope being persisted when you select a product documentation in the Product Documentation section on the Micro Focus SupportLine Web site and choose to view the documentation in the Micro Focus Infocenter.
- Updated **Scope** settings - provides the ability to nest four levels deep when setting a scope.
- Scope being persisted between browser sessions once it has been set.
- Creating automatic scopes using the **Search Topics** icon, .
- A link to change the scope from the search results when there are too many results.
- Improved Boolean search expressions.
- Details included with the search results.
- Help on how to use the Infocenter and how to construct search expressions - available using the Infocenter Help button, .

Micro Focus Unit Testing Framework



Note: This is a technology preview feature only. It is being made available to allow you to test and provide feedback on this new capability, but it is not intended for production use and is not supported as such. Furthermore, Micro Focus does not guarantee that this feature will be delivered at a GA level and if it is, then the functionality provided might differ considerably from this technology preview. During the preview, you are encouraged to share your feedback and experiences via the Micro Focus community forum - <http://community.microfocus.com/microfocus/>.

The Micro Focus Unit Testing Framework is an xUnit style testing framework, available from the command line, for procedural COBOL applications.

It includes much of the architecture you would expect in an xUnit framework. The test runner is a 32- or 64-bit executable that you run from an Enterprise Developer command or shell prompt. A test fixture or suite is a COBOL program compiled to `.dll` (Windows) or `.so` (UNIX) that can include the setup, the test case code, and the teardown associated with the test case.

Test results are available in a number of formats. By default, results are displayed to screen and to a `.txt` file, but you can use additional parameters on the command line to produce reports in JUnit format.

Preprocessors

Support has been added in the IDE for enabling and using multiple preprocessors with your projects.

A new page, **Additional Preprocessors**, has been added to the project's and the files' properties of native COBOL applications to enable you to choose one or more preprocessors to use when building your application and to specify their order of execution.

New reporting capability is now available for user preprocessors: `resp-main code 18` indicates that a buffer contains a data name to be marked as modified by the immediately preceding preprocessed line. The data name may be qualified and `resp-more` contains the column information for the reference.

Profiler

Enterprise Developer now provides support for Profiler for native COBOL applications directly from within the IDE. To produce reports, you need to:

1. Enable Profiler in the COBOL property page for a project, a build configuration, or a file.
2. Compile your application to apply the changes.
3. Create a run configuration that has Profiler enabled.
4. Run your application with Profiler to produce the relevant reports.

Remote Connections

This release provided the following enhancements:

- An improved diagnostic tooling to help determine connection problems - enhancements are available for both the client and the server installations.
- It is now possible to specify the Remote System Explorer (RSE) type of connections to create remote mainframe COBOL and PL/I projects. This is to cater for scenarios when it is not possible to use SAMBA or NFS connections within your environment.



Note: The following features and utilities are not supported when remote projects use the RSE connection type:

- The file layout editor and the file editor in the Classic Data File Tools and Data File Tools (Technology Preview) utilities.
 - The BMS compiler and painter.
 - The IMS layout editor and the IMS database editor.
 - Mainframe Assembler.
- Changing the type of a remote connection - it is now possible to change the type of the remote connection from RSE to NFS and vice versa using the **Remote Settings** for remote projects.

REST service interfaces

RESTful service interfaces utilizing JSON as the media type in request and response messages are now supported using the Interface Mapping Toolkit. This enables you to extend COBOL applications using modern transport payloads and protocols.

RM/COBOL Compatibility

This release includes improved support for RM dialect applications. Please consult with Micro Focus before considering a transition from RM/COBOL to Enterprise Developer.

Single file support

The recommended way to work with files within Enterprise Developer is to include them in a project. For situations where you might want to quickly open edit a single file, Enterprise Developer now provides support for native COBOL files in the IDE when the file is not opened as part of a project. There is limited support for the IDE editing, compiling and debugging features as full support requires a project file.

Tunables

Enterprise Developer version 2.3 contains the following new tunables:

putenv_interface Provides backward compatibility for UNIX systems in which the operating system's putenv() function is required when setting environment variables.

strict_file_locking Enables a new, more reliable method of file locking for UNIX systems. See *File Locking* for more information.

Enterprise Developer version 2.3 contains the following updates to tunables:

- default_cancel_mode** A new parameter, and default, has been introduced for this tunable; see *default_cancel_mode* for more information.
- subsystem_cancel_mode** A new parameter has been introduced for this tunable; see *subsystem_cancel_mode* for more information.

Updated run-time system

Enterprise Server has been updated to provide an execution environment capable of running applications that were each built using different development products. A consequence of this is that if your application has a main COBOL executable (.exe) that was built with a previous version of Enterprise Developer, you should ensure that the executable is rebuilt and packaged with the new run-time system. This does not affect mainframe subsystem projects deployed to Enterprise Developer. You can rebuild from the IDE or the command line.

Other COBOL subprograms built with previous versions of Enterprise Developer are not required to be rebuilt.

What was New in Enterprise Developer 2.2 Update 2

Enterprise Developer 2.2 Update 2 provided enhancements in the following areas:

- [Eclipse](#)
- [Character Set Enhancements](#)
- [Code Analysis](#)
- [AppMaster Builder](#)
- [Database Access](#)
- [Enterprise COBOL 5.1 compatibility](#)
- [External Call Interface \(ECI\)](#)
- [Micro Focus COBOL enhancements](#)
- [External Security Facility \(ESF\)](#)
- [Enterprise Server MQ-IMS Bridge](#)
- [IMS Connect](#)
- [PL/I General Enhancements](#)
- [PL/I Editor Enhancements](#)
- [RUMBA 9.2](#)
- [Transaction Classes \(Technology Preview\)](#)
- [Tunables](#)

Eclipse

Enterprise Developer 2.2 Update 2 provides the following new functionality and improvements:

- Support for Eclipse 4.2 and 4.3 - Enterprise Developer 2.2 Update 2 ships with Eclipse 3.8 but also supports Eclipse 4.2 and 4.3 (the 32-bit IDE only). To use Enterprise Developer with a newer version of Eclipse, you need to install Enterprise Developer first and then use the `installeclipseplugins.bat` script in `%ProgramFiles(x86)%\Micro Focus\Enterprise Developer\installer` (UNIX). See *Installing into other instances of Eclipse* for more details.
- Remote JVM COBOL projects - this release provides enhanced support for remote JVM COBOL projects.
- Remote project connections - the diagnosis tool for remote connection issues has been improved. There is now a client-side diagnosis tool and a server-side diagnosis tool for diagnosing connection problems to remote projects and connections to your Micro Focus Enterprise Developer UNIX Components. You should run both tools for a complete diagnosis.

Character Set Enhancements

The following character sets, available using the MFCODESET environment variable, have been enhanced or added in this release:

- Thai Extended (0066) - new
- Korean (0082)
- Simplified Chinese (0086)
- Traditional Chinese (0886)

There are also a number of double-byte character sets that are now capable of mixed single-byte and double-byte character conversion; see the definition of MFCODESET in *Environment Variables in Alphabetical Order* for more information.

Code Analysis

Enterprise Developer version 2.2 Update 2 provides Dead Code analysis for COBOL programs that enables you to find unreferenced items or any piece of code that can't be reached .

AppMaster Builder

The AppMaster Builder Database View and maintenance support (known as Data Views) is now provided within the Eclipse framework for Enterprise Developer, rather than relying on invoking the AMB database view interface that was part of Mainframe Express Enterprise Edition. You must install the AMB Eclipse plug-in to be able to use this feature.

Data Views enable you to create and maintain database views, as well as add and compile PSB and DBD sources in a COBOL project, all while working within the Eclipse framework.

Database Access

The following new features are available in database access support:

- | | |
|-----------------------------------|--|
| COBSQL | In Eclipse, the new KEEPCOMP directive resolves COMP/COMP-5 issues with Oracle applications on little-endian platforms. |
| HCO for DB2 LUW | Enterprise Developer version 2.2 Update 2 introduces GEN-HV-FROM-GROUP - a new DB2 ECM compiler directive option, that generates host variables for all elementary data items when a multiple-level group variable is used in a FETCH or singleton SELECT DB2 statement. |
| HCO for SQL Server (HCOSS) | Enterprise Developer 2.2 Update 2 provides the following new HCOSS features: <ul style="list-style-type: none">• DATE/TIME formatting - the OpenESQL Configuration Utility now supports DATE/TIME formatting for HCOSS batch utilities.• Multi-row INSERT - supports a multi-row INSERT statement. |
| OpenESQL | This version provides the following new OpenESQL features: <ul style="list-style-type: none">• Support for SQL Server 2014.• New SQL Compiler directive options:<ul style="list-style-type: none">• DETECTDATE=SERVER - resolves host variables alignment with column data types in an SQL table.• GEN-HV-FROM-GROUP - generates host variables for all elementary data items when a multiple-level group variable is used in a FETCH or singleton SELECT SQL statement.• Sample applications - the following native COBOL SQL sample applications are new with this version: |

- Get Diagnostics - demonstrates how to use GET DIAGNOSTICS EXEC SQL calls to get diagnostic information from various DBMSs.
- LOB Data Types - Demonstrates how to INSERT and SELECT LOB data in a native application using various DBMSs.

SQL Option for DB2

Support for the following has been added in this version:

- DSNUTILB LOAD LOG=NO option
- RID scalar function
- XML data type via XDB Link to mainframe DB2

XA switch modules

The following XA switch module updates are available in this version:

- Oracle switch module - Windows and UNIX platforms :
 - Supports User Impersonation when statically registered.
 - Enables you to specify which XA resource definitions use User Impersonation.
 - Now compiled with one source file, rather than two.
- SQL Server switch module - Windows platforms only:
 - Enables you to specify which XA resource definitions use User Impersonation.
 - Now compiled with one source file, rather than two.

Enterprise COBOL 5.1 compatibility

There have been a number of enhancements to make this version of Enterprise Developer more compatible with Enterprise COBOL 5.1.



Note: These enhancements are available under both DIALECT(MF) and DIALECT(ENTCOBOL).

- The following phrases have been added to the XML GENERATE statement:
 - NAME
 - SUPPRESS
 - TYPE
- The following intrinsic functions have been added:
 - ULENGTH
 - UPOS
 - USUBSTR
 - USUPPLEMENTARY
 - UVALID
 - UWIDTH
- A number of reserved and context-sensitive reserved words have been added; see the topics *Reserved Words Table* and *Context-sensitive Words Table* for complete lists.

External Call Interface (ECI)

Enhancements to the ECI include:

- The ECI for Java clients is no longer restricted to the 32K Commarea. You can now transfer virtually unlimited containers of any size in a channel to a CICS server program. See *ECI Java Interface* for more information.
- Java support for IBM's implementation of ECI, which, in turn, uses the Micro Focus implementation.
- A ECI RA for WebSphere; however, in this release, no IMTK tooling is provided for servlets or JSP.

Micro Focus COBOL enhancements

The following enhancements have been made to Micro Focus COBOL:

- The following phrases have been added to the XML GENERATE statement:
 - NAME
 - TYPE
 - SUPPRESS
- The following intrinsic functions have been added:
 - ULENGTH
 - UPOS
 - USUBSTR
 - USUPPLEMENTARY
 - UVALID
 - UWIDTH

External Security Facility (ESF)

The Enterprise Server External Security Facility (ESF) now supports caching the results of some security queries. This can improve the performance of enterprise server instances and of the MFDS when they are configured to use external security.

To enable caching, you need to set non-zero values for the **Cache limit** (maximum size of the cache) and **Cache TTL** (Time To Live, or how long before a cached result expires) settings on the **MFDS Security** tab, the **Default ES Security** tab, or on the **Security** tab for an individual enterprise server. (Currently, the cache settings for Security Managers have no effect; you need to set cache parameters on one of the three Security pages mentioned earlier.)

For more information, see <http://supportline.microfocus.com/examplesandutilities/doxygen/caching.html>.

Enterprise Server MQ-IMS Bridge

At Enterprise Developer 2.2 Update 2 the Enterprise Server MQ-IMS Bridge was supported at GA level. It had previously (from Enterprise Developer 2.2 Update 1) been available as a Technology Preview item only.

IMS Connect

Visual COBOL 2.2 Update 2 now supports Java IMS Connect messages except when also using the IBM-supplied HWSJAVA0 I/O exit, excluding customizations to that exit.

PL/I General Enhancements

Enhancements are provided in the following areas:

- Attributes - the PL/I DECLARE statement now supports the OPTIONAL attribute as part of the parameter-descriptor list or as an attribute in a parameter declaration. This provides an improved functionality and a greater language compatibility for customers looking to migrate applications from z/OS environments.
- Built-in functions - the PL/I Compiler and run-time system now support the PRESENT and OMITTED built-in functions which provides an improved functionality and a greater language compatibility if you are looking to migrate applications from z/OS environments.
- CodeWatch - this release includes the following enhancements:
 - CodeWatch is now supported in native 64-bit environments which ensures that the CodeWatch debugger runs properly in native 64-bit mode. This enables the PL/I debugger to take advantage of faster speeds and to debug larger programs which have access to larger amounts of storage.

- Notifications for the CodeWatch conditions - you can use the command line and the UI to turn notifications on or off for all of the currently documented PL/I conditions. The use of notifications enables you to investigate the state of programs and optionally set breakpoints before continuing program execution.
- Compound operators - the PL/I macro preprocessor now supports the compound operators += , -=, *=, and /= within macro assignment statements.
- Error handling - you can now use the EXEC CICS HANDLE ABEND check which provides a consistent error handling technique for PL/I programs running under CICS that is consistent with error handling on the mainframe. EXEC CICS HANDLE ABEND determines error handling for programs running under CICS. Conditions detected by CICS cause an abend if established using EXEC CICS HANDLE ABEND, and PL/I ON units do not get control. If EXEC CICS HANDLE ABEND is not established, CICS defines that the action taken will be as defined by CICS, which normally leads to a transaction abend.
- PL/I data structures - this release provides enhanced support for PL/I data structures in the Data File Tools. You can now easily generate a PL/I structure map (.str) from within the IDE, and the Data File Tools can then use to generate a formatted layout for a PL/I data file. This enables you to see what is in each field within a PL/I Data File Record, and to modify the records in a logical way without having to know the offset, length and raw format of the data item.

PL/I Editor Enhancements

Enterprise Developer 2.2 Update 2 provides a new and improved PL/I editor within Eclipse. Features include:

- Configurable background syntax parsing.
- Syntax error reporting - uses red squiggles to denote errors in the code.
- Configuration settings for the editor - you can hide or show the horizontal ruler, turn on or off visible lines for the margins (at columns 2 and 72 for Mainframe Subsystem projects, and only at column 72 for native projects), and configure the syntax coloring and the smart edit mode.
- Smart edit mode - defines the word wrapping behavior when breaking lines or around margins, moving the cursor using the **Home** and **End** keys.
- Folding - support for collapsing and expanding regions for procedures and blocks within procedures.
- Content Assist - provides assistance with inserting PL/I keywords, variables and syntax templates.
- Syntax templates for the most commonly used PL/I statements.
- Block selection - use the button from the toolbar to toggle the mode.
- Breadcrumbs navigation - enables you to quickly navigate between source programs and the dependent include files.
- Outline view - shows a structure of the procedure and structure declarations within a program. Clicking on an item in the Outline view positions the cursor on that item in the code and vice versa.
- Include Dependency View - shows where in the source file's structure the INCLUDE statements appear, and which include files they reference.
- Opening include files directly from the editor using **F3**.
- Task creation - inserting a TODO or a FIXME in a comment in the code automatically creates a task that appears in the Task view.

Rumba 9.2

Visual COBOL 2.2 Update 2 of Enterprise Developer includes Micro Focus Rumba 9.2 which you can optionally install as part of the setup process. The license for Enterprise Developer will license all components of Rumba.

Transaction Classes (Technology Preview)

Transaction classes provide a mechanism to limit the number of CICS® tasks in your system. By spreading your tasks across a number of transaction classes and configuring the maximum number of tasks that can be dispatched within each transaction class, you can control resource contention between tasks and limit the number of tasks that CICS considers eligible for dispatching at task attach.

Tunables

Enterprise Developer 2.2 Update 2 includes the following new tunable:

- `reduce_java_signals` - specifies the options that are passed to a JVM when mixing Java and COBOL.

What was New in Enterprise Developer 2.2 Update 1

Enterprise Developer 2.2 Update 1 provided enhancements in the following areas:

- [Micro Focus Heartbleed Update](#)
- [ACUCOBOL-GT Compatibility](#)
- [AppMaster Builder](#)
- [Assembler Support](#)
- [Btrieve Support](#)
- [COBOL Source Information](#)
- [Compare and Synchronization Monitor](#)
- [Compiler Directives](#)
- [Database Access](#)
- [Eclipse IDE](#)
- [Enterprise Server Integration in the IDE](#)
- [Enterprise Server MQ-IMS Bridge \(Technology Preview\)](#)
- [Environment Variables](#)
- [Fileshare Recovery](#)
- [Line Numbering for COBOL Programs](#)
- [Local Catalog](#)
- [IMTK](#)
- [Working with the Mainframe](#)
- [Mainframe Emulation](#)
- [Mainframe Subsystem](#)
- [Managed COBOL](#)
- [Managed COBOL Syntax](#)
- [PL/I](#)
- [Preprocessors](#)
- [Project and Item Templates](#)
- [Remote Connection](#)
- [Rumba Integration with Eclipse](#)
- [Run-time Launch Configuration Files](#)
- [Terminfo Files](#)

Micro Focus Heartbleed Update

The OpenSSL library used in this product was updated to version 1.0.1g to fix the "Heartbleed" vulnerability with TLS heartbeat requests.

ACUCOBOL-GT Compatibility

The following ACUCOBOL-GT support has been added in this release:

- **-Di compiler option** The `-Di` compiler option, which initializes Working-Storage data items based in their type, is now supported.

AppMaster Builder

Enterprise Developer 2.2 Update 1 provided updated AppMaster Builder support that integrates more fully into the Eclipse IDE. In addition, the following enhancements have been added:

- Code generation and compilation in an Eclipse COBOL project
- Enhanced Program Painter
- Enhanced Job Queue

Assembler Support

Enterprise Developer 2.2 Update 1 provided Assembler support from within the IDE. Features include:

- Support for creating Assembler programs (.mlc extension), macro files (.mac or .cpy extensions) or Assembler linker files (.lin) from within the IDE.
- Support for building Assembler programs and for linking Assembler linker files from within the IDE. You can choose to exclude certain files from the build.
- Assembler editors for Assembler and macro files - includes content assist for reserved words and colorization.
- Support for using Ant for building .asm, .mlc and .cap files and for linking .lin files.
- IDE support for configuring the Assembler compiler and linker options on project, build configuration and file levels.
- Configuring the macro paths.

Btrieve Support

Support for the Btrieve file handling system from Pervasive Software Inc. has been added into Enterprise Developer.

Support is restricted to native COBOL, in a Windows environment.

COBOL Source Information

The **Quick Browse** option is now available as a context menu command in the editor.

Compare and Synchronization Monitor



Note: This feature is only available in a Windows environment.

With the release of Enterprise Developer 2.2 Update 1, the Compare and Synchronization Monitor has been updated to version 2.

Version 2 is greatly improved in terms of performance, especially during initial checkout of partitioned data sets or when synchronizing a large number of members. Also, the user interface has been improved, and some of the functions available in the old version have now changed or become obsolete.

Compiler Directives

The following Compiler directives have been added in this release:

- ILPARAMS** Determines the way in which you call a method that contains an array as its last receiving parameter.
- INIT-BY-TYPE** Initializes Working-Storage Section data items to a default value, according to their type.
- Alphabetic, alphanumeric, alphanumeric edited, and numeric edited items are initialized to spaces.
 - Numeric items are initialized to zero.
 - Pointer items are initialized to null.

- Index items are initialized to the value 1.

Database Access

The following new features have been added as part of database access support:

HCO for DB2 LUW

- Support added for DB2 LUW version 10.5.
- Enhanced RETURN-CODE processing.
- The HCO DDL processor now accepts column names with a length greater than 18 characters.
- The HCO Import Data tool now processes CLOB data less than 32K in size.
- The HCO Import Data tool can now convert DSNTIAUL import files to delimited format, enabling the actual import to be done by DB2 LUW IMPORT and LOAD utilities. This is done using the new SAVEDEL and CHARDEL parameters.

HCO for SQL Server (HCOSS)

- Support for the mainframe SET CURRENT SQLID statement.
- Support for the mainframe LEFT and RIGHT scalar functions.
- Data Migration tools now support the mainframe ALIAS and SYNONYM objects.

OpenESQL

- Enhanced internationalization support for UNICODE, DBCS and MBCS.
- Enhanced GET DIAGNOSTICS statement support.
- Enhanced LOB support for CLOB, BLOB and DBCLOB data types.
- Enhanced IDE support for OPTION directives.
- Now provides support for the creation of save points and rolling back to save points.

XA Switch Modules



Restriction: This feature applies only when the Enterprise Server feature is enabled.

- New two-phase commit module for SQL Server based on Microsoft's XA switch. This provides support for xa_recover.
- Support for DB2 LUW version 10.5.
- Support for Oracle version 12.1.

Eclipse IDE

Assigning memory to Linkage Section items

If the debugger steps on a line with an unassigned linkage item (for example, if you are debugging only a part of your application and no memory has been allocated to that linkage item), debugging terminates. To assign linkage to that data item and continue debugging, you need to select the data item, right-click it and click **Inspect COBOL**. When prompted, confirm and assign a value to the data item. Alternatively, to assign linkage, you can right-click the data item in the **Variables** view and click **Change Value**.

Debugging Windows Services

It is now possible to debug Windows services. You must be logged on to the console of the computer running the service and can debug either using just-in-time debugging and a CBL_DEBUGBREAK call, or using library routines and a "COBOL Wait for Application Attachment" debug session having added a call to CBL_DEBUGBREAK or CBL_DEBUG_START to the application.

Indicating that a file is a copybook or a COBOL program

In COBOL Explorer, you can now use two new file context menu commands to indicate that a COBOL program is a copybook (**Transform Program to Copybook**) and that a copybook is a COBOL program (**Transform Copybook to Program**). You may need to use these in situations when you imported existing COBOL source code

in the Eclipse IDE and some of the files were incorrectly identified as either a copybook or a COBOL program.

Enterprise Server Integration in the IDE

You can now use the context menu for the servers in Server Explorer to enable the display of the Enterprise Server log information in the Console view.

Enterprise Server MQ-IMS Bridge (Technology Preview)



Note: At Enterprise Developer 2.2 Update 1 this was provided as a technology preview feature only. It was made available to allow you to test and provide feedback on this new capability; however, this feature was not intended for production use and was not supported as such.

Enterprise Developer version 2.2 Update 1 provided support that enables WebSphere MQ applications to communicate with IMS applications in an Enterprise Server region.

Environment Variables

The following environment variable has been added in this release:

strictvsam strictvsam enables strict mainframe emulation when processing VSAM files.

When set to ON and running under mainframe emulation, file status 37 is returned for an existing VSAM file when opened for OUTPUT if the file has data or previously had data written to it, or if the file is of a different format to the file on disk. When set to OFF, file status 0 is returned and a new file is created when an existing VSAM file is opened for OUTPUT. This variable is set to OFF by default.

Fileshare Recovery

Recovery of Fileshare data files has been enhanced.

Rollback recovery is a faster process that aims to fix the files from their failed state.

This process cannot be used in all scenarios, but a new user exit has also been introduced that allows you to programmatically control which files you wish to recover with this process.

Hot backups are also a new introduction, which allow you to perform a backup without having to shut down Fileshare.

Line Numbering for COBOL Programs

Enterprise Developer version 2.2 Update 1 provided options for auto-inserting or removing line numbers in source files open the editor. Features include:

- COBOL numbering - line numbers are inserted in the sequence area of the code (columns 1 - 6), starting by default at 000100 at the first line, incrementing by 100 by default.
Micro Focus recommends that you use COBOL numbering only if your files are in fixed or variable source format.
- Standard numbering - line numbers are inserted immediately to the right of area B, in columns 73 - 80, starting by default at 00000100 at the first line, incrementing by 100 by default.
Micro Focus recommends that you use Standard numbering only if your files are in fixed format.
- The **Renumber** and **Unnumber** commands available from the context menu in the editor.

Local Catalog

From within Server Explorer in the IDE, you can add one or more local catalogs, and then use them in conjunction with Drag and Drop to drag down data set items from your mainframe, edit them and then move them back to the mainframe or to your testing environment.

IMTK

You can now use the Interface Mapping Toolkit to create Web Services and Java Interfaces for remote COBOL projects.

Working with the Mainframe

Enterprise Developer version 2.2 Update 1 includes the “Launch ISPF” functionality as a technical preview. This feature is supported only under z/OS 1.13.

Mainframe Emulation

Previously, MGMTCLAS was supported by simply saving it on the catalog record when a dataset was allocated in JCL using the MGMTCLAS parameter on the DD statement. Starting with this release, you can use batch jobs to define MGMT classes and to specify the EXPIRE AFTER time (in days).

Mainframe Subsystem

New editors Enterprise Developer now provides special editors for REXX, TSO CLIST and Easytrieve programs. The editors include colorization and content assistance. Syntax checking is not available.

Managed COBOL

Enterprise Developer now provides support for Java managed beans (MBean) in JVM COBOL code that enable you to manage and monitor RunUnits, and to identify certain issues such as leaks and long-running RunUnits.

- You can enable an MBean only for a particular RunUnit level or for all RunUnits you create.
- You can view and use MBeans from programs such as Oracle's Java Mission Control or JConsole.
- MBeans include the `LogicalRunUnitCount` and `LiveRunUnitCount` attributes that enable a visual indication of how many RunUnits are live. If the values of these two attributes are different, this might indicate some issues.

Visual COBOL 2.2 Update 1 also includes the following tutorials for JVM COBOL:

Deploying JVM COBOL to an Application Server Using some ready-made sample projects, this tutorial guides you through implementing your JVM COBOL code into an Enterprise JavaBean (EJB), then deploying it to a JBoss application server. Instructions are also included on how to deploy the application to WebSphere and WebLogic application servers.

Managed COBOL Syntax

Enterprise Developer version 2.2 Update 1 includes the following enhancements to the managed COBOL syntax:

Specifying parameters in the method signature You can now specify passing parameters and returning items in the method signature, instead of using a Procedure Division header. This applies to methods, indexers, iterators, constructors and delegates.

CONSTANT keyword Use the CONSTANT keyword on a field to protect it from being altered.

Operations on string fields You can now use the STRING, UNSTRING and INSPECT statements on fields of type string.

PL/I

PL/I support within the IDE now includes the following enhancements:

- Support for new built-in functions

The PL/I Compiler and Runtime now support the following new built-in functions:

DAYS
DAYSTODATE
SECSTODATE
SECS
WEEKDAY
DAYSTOSECS
ONSUBCODE
REPATTERN
IAND
IOR
INOT
OMITTED
XMLCHAR

This provides an improved functionality and a greater language compatibility when you migrate applications from a z/OS environment.

- Support for %OPTION in the PL/I macro preprocessor

You no longer need to modify application code that uses %OPTION in order for it to compile successfully.

- Support for QUOTE() and COMMENT() built in functions.

This enables the use of PL/I macros that generate code using these built-in functions.

- Support for IBM Style %INCLUDE(s) that utilize PDS name as part of include file names.

The PL/I macro preprocessor now parses the %INCLUDE syntax of the form shown below and utilizes the –isuffix and –ipath Compiler directives to locate the “member”.

```
%INCLUDE MYPDS(MYMEMBER) ;
```

This provides an improved language compatibility with the application code without the need to modify the original underlying source code.

- Support for PL/I AREA data items up to 2Gb.

Open PL/I now supports the declaration and usage of variables of type AREA up to 2Gb in size. This provides you with improved functionality and greater language compatibility when migrating applications from a z/OS environment as there is no need to modify the code in order to deal with a maximum area size of 32Kb.

- Enhanced compatibility with the DISPLAY() REPLY() syntax.

The PL/I Compiler and runtime now provide a DISPLAY() REPLY() syntax that is no longer dependent upon the use of the SYSIN and SYSOUT DD's and that no longer causes behavioral differences if stream IO to SYSIN/SYSOUT is intermingled with programs using DISPLAY() REPLY().

- Support for big-endian style bit strings on little-endian platforms.

Using a new Compiler directive –bitsltr to compile programs that have logic that is dependent on big-endian bitstring ordering enables you to utilize the code unchanged on Open PL/I little-endian platforms (such as Windows, Red Hat Linux, SUSE Linux). This enables you to migrate applications that make an assumption about bit ordering on the underlying operating system or hardware and to run them unchanged.

- Support for the –initcall Compiler directive.

Compiling a PL/I program with the –initcall directive causes the specified program to be invoked prior to any user logic. This enables you to customize your environments and do things such as tracking which routines have been invoked, detect memory corruption prior to routine startup, establish database connectivity.

- Support for the “F” option in PLIDUMP.

A PLIDUMP which was generated using the “F” option will now attempt to list all known information about files which have been accessed during the execution of the program and includes information about the last IO operation, key values, and a dump of buffers, if available. This enables you to identify more easily the cause of file-related failures and what file operations were in progress at the time of failure.

- Technology Preview support for PL/I data structures Data File Tools.

In the IDE, you can now generate a PL/I structure map (.str) which can be utilized by the Data File Tools utility to generate a formatted layout for a PL/I Data File. The PL/I data structure must be compiled in advance with the -adt Compiler directive. This enables you to see what is in each field within a PL/I data file record, and to modify them in a logical way without having to know the offset, length and raw format of the data item.

- Support for PL/I debugger notification for ON Unit invocation.

You can now request that the debugger stops when a PL/I ON Unit is about to be triggered. This enables you to locate the ON Unit, establish appropriate breakpoints within the ON Unit and determine the cause of the ON Unit being triggered.

- Support for PL/I debugger notifications for the load of shared libraries and .dll files.

You can now request that the debugger issues a notification when a shared object or a .dll file has been dynamically loaded. This enables you to stop a program when a dynamic program has been loaded and to establish the appropriate breakpoints, watch lists, etc.

- Alternative Index Performance Improvements for the CASSPOOL command.
- Support of the SHAREOPTIONS property in a VSAM cluster.
- Support for setting directives automatically for PL/I files.

In Eclipse, click **Window > Preferences > Micro Focus > PL/I > Directives Determination**.

- A new page in the project and file properties enables you to set SQL directives on PL/I source files.

Preprocessors

A guide for third-party vendors and administrators showing how to integrate additional preprocessors with Enterprise Developer is now available in the product help in the *General Reference > Integrated Preprocessor Interface* section.

Project and Item Templates

Support is now available for using existing projects and files as custom templates to create new projects and files. You create and configure projects that include the files and settings you would like to use as templates.

Remote Connections

Enterprise Developer now provides a new connection type, **Micro Focus DevHub using SSH**, that uses a Secure Shell daemon process to launch a server on the remote host.

You can use this type of connection when the UNIX machine you are connecting to uses LDAP authentication which is not supported by the DevHub daemon.

Using this connection also means you do not need to run the DevHub daemon process with root privileges. It also gives you greater flexibility in setting environment variables needed for building or debugging on the remote server.

Rumba Integration with Eclipse

Enterprise Developer 2.2 Update 1 provided enhanced integration with the Micro Focus Rumba application for running and debugging Mainframe Subsystem applications that require a TN3270 emulator to run. You can now configure Eclipse to launch a Rumba mainframe display embedded in the IDE or a mainframe session of Rumba Desktop.

Run-time Launch Configuration Files



Note: This feature is only supported in a Windows environment.

Use a run-time launch configuration file to ensure an application can be launched when it is deployed in a separate location to the run-time system (in the case of dynamically bound applications), or when the licensing daemon is not already running.

Terminfo Files

The following terminfo files have been added:

- *ansi80x25* - this is based on the old *ansi* file. A newer version of *ansi* exists in this release that has no function key support, which is consistent with *ansi* terminfo files on various other UNIX platforms. If you currently use *ansi* and require function key support, you should instead set the TERM environment variable to *ansi80x25* to continue previous behavior.
- *xterm-color* and *kterm-color* - these are now available on all UNIX platforms - previously, they were only available on Linux.
- *aixterm-old* (AIX systems only) - this has similar capabilities to the AIX OS terminfo file of the same name. It differs from the existing *aixterm* file, because it has no line drawing capability. Line drawing is only possible with *aixterm* if it is displayed on an appropriate display (X server).
- *vt220-w* - this is the wide (132-column) version of the vt220 file, and is based on the vt100-w file. For more information, see *Wide Terminal Mode* in the documentation referenced at the bottom of this section.

There have been a number of additions and fixes to existing terminfo files; refer to the *Terminfo Database and Terminal Devices* section of the documentation for full details.

There have also been a number of terminfo files that have been removed; refer to the *Backward Compatibility* section for a complete list.

What was New in Enterprise Developer 2.2

Enterprise Developer 2.2 provided enhancements in the following areas:

- [ACUCOBOL-GT Compatibility](#)
- [RM/COBOL Compatibility](#)
- [Application Configuration](#)
- [COBOL Source Information \(CSI\)](#)
- [Compiler Directives](#)
- [Consolidated Tracing Facility](#)
- [Enhanced Accept and Display Statements](#)
- [Debugging](#)
- [Grouping Files in Virtual Folders in Solution Explorer](#)
- [File Handling](#)
- [Interface Mapping Toolkit](#)
- [Managed COBOL](#)
- [Problems View](#)
- [Upgrading from Net Express to Visual COBOL](#)
- [XML Extensions](#)

ACUCOBOL-GT Compatibility

The following enhancements are applicable to Enterprise Developer:

- Accessing data files through AcuServer - You can now access your ACUCOBOL-GT data files, both sequential and Vision files, through AcuServer.

- Standard library routines - Support for the following library routines has been added:
 - C\$GETPID
 - C\$JUSTIFY
 - C\$LIST-DIRECTORY
 - C\$LOCKPID
 - C\$REGEXP
 - C\$RUN
 - C\$SLEEP
 - C\$SYSTEM
 - C\$TOLOWER
 - C\$TOUPPER
 - I\$IO
- Using Vision files with Micro Focus Data File Tools - You can now use some of the Data File Tools functionality with Vision files. You can:
 - Convert Vision files to Micro Focus format using the Data File Converter and the DFCONV command line utility.
 - Edit Vision files using the Data File Editor.



Note: For more information about the **Data File Tools** utility, see *Data Tools*.

RM/COBOL Compatibility

The following support has been added to Enterprise Developer in this release:

- Subprograms - Support for the following subprograms (referred to as library routines in Enterprise Developer) has been added:
 - C\$OSLockInfo
 - C\$SecureHash
- recover1 - The recover1 utility, RM/COBOL's indexed file recovery utility, is now distributed with Enterprise Developer. Refer to the *RM/COBOL File Handling* section of *RM/COBOL Compatibility* for details of its use.

Application Configuration

You can now set environment variables for when you run native projects from within the IDE from the project's properties - click **Environment** on the **Application** tab in the project properties.

COBOL Source Information (CSI)

COBOL Source Information (CSI) provides a quick and easy way of providing you with information about your program when you are working on it. You enter a query in the **Quick Browse** dialog box and CSI returns the results of the query in the Search view.

Compiler Directives

The following Compiler directives are new:

- | | |
|-----------------------------|--|
| ACU-UNDERSCORE | This directive treats underscores in COBOL words as hyphens. |
| ILSHOWPERFORMOVERLAP | This managed COBOL-only directive generates a warning when an overlapping PERFORM range is detected in the program. |
| ILEXPONENTIATION | This managed COBOL-only directive enables you to optimize exponential arithmetic operations by specifying the calculation method used. |

EXITPROGRAM This directive determines how the EXIT PROGRAM statement is executed.

The following Compiler directives have changed

CHANGE-MESSAGE The scope of this directive has been widened to allow you to change the severity of different types of error messages, not just syntax checking messages.

DIALECT"RM" DIALECT"RM" now sets PERFORM-TYPE"RM". If you recompile an application that uses DIALECT"RM", the behavior may change for nested PERFORM statements. If that is the case, explicitly set PERFORM-TYPE"MF" after DIALECT"RM" to continue with the previous behavior.

HIDE-MESSAGE The scope of this directive has been widened to allow you to hide different types of error messages, not just syntax checking messages.

PRESERVECASE This directive now defaults to PRESERVECASE when compiling native COBOL; managed COBOL compilation already defaults to PRESERVECASE. This results in externally visible identifiers preserving their case instead of being converted to uppercase.

Consolidated Tracing Facility

The following changes have been made to the Consolidated Tracing Facility (CTF):

CTF for JVM COBOL application CTF tracing is now supported in JVM COBOL applications.

New emitters A new emitter, JVALOGGER, is available for JVM COBOL web applications. This emitter passes details to the Java logging API that is available through your web server. Use this emitter if you are unable to configure the TEXTFILE or BINFILE emitters due to web server permission restrictions.

New properties and variables for existing emitters The following support has been added to existing emitters.

Properties The following property has been added to the BINFILE emitter:

Property	Description
RunUnitID	Controls whether the RunUnit information is included in the trace.

Variables Four new pseudo-variables for the FILE property have been added to the BINFILE and TEXTFILE emitters:

pseudo-variable	Description
\$(PLATFORM)	A platform specific constant, useful when two run-time systems are in the same process, and you require separate trace files
\$(RUNUNIT)	A unique number that represents the managed RunUnit ID
\$(RUNUNIT_SESSIONNAME)	The session name passed to the managed RunUnit

pseudo-variable	Description
\$(RUNUNIT_GUID)	The globally unique identifier associated with the managed RunUnit

Enhanced Accept and Display Statements

Two of the existing Enhanced ACCEPT and DISPLAY settings available through Adis have additional values, which are aimed at RM/COBOL users migrating their source code to Enterprise Developer. The new values are:

- Emulation of RM/COBOL-85 style data entry for numeric data entry on ACCEPT statements.
- Emulation of an RM/COBOL backspace in free format fields when in replacement editing mode, in that deleted characters are removed and characters to the right are shifted left, the same as when in insertion editing mode.

For more information on how to set these values, refer to *Configuring Enhanced ACCEPT and DISPLAY*.

Debugging

Displaying debug information for managed applications

You can set the DEBUG constant for managed COBOL projects on the **COBOL** tab in the project properties. This enables you to use the System.Diagnostics.Debug class in your applications to ensure they write diagnostic information in the Output window for projects compiled for Debug but not for projects compiled for Release.

Changing the display format for individual items in the Watch window

It is now possible to change the display format for individual items in the Watch window in COBOL. To do this, click a row, press **F2**, and type: *Variable,h* or *Variable,x* to always display the values in hexadecimal format; *Variable,d* to always display the values of variables in decimal format, and of strings - as text.

Grouping Files in Virtual Folders in Solution Explorer

Enterprise Developer now provides a Virtual View of a project within Solution Explorer. In the Virtual View you use virtual folders to improve navigation by logically grouping the files that make up the project. You can also create your own virtual folders to group files of your choice (a file can only belong to one virtual folder). The files can be of different file types.

File Handling

New features include:

- Converting and editing Vision and RM/COBOL indexed data files using the Data File tools is now supported.
- Access to data files (either sequential or indexed) through AcuServer is now supported.
- Access to Vision and RM/COBOL indexed data files through Enterprise Server is now supported.

Interface Mapping Toolkit

Enterprise Developer now supports the creation and deployment of COBOL program-based services using the Interface Mapping Toolkit (IMTK).

Managed COBOL

Documentation A guide that provides a basic introduction to Object-Oriented Programming (OOP) for COBOL developers, *An Introduction to Object-Oriented Programming for COBOL*

Developers, with examples is now available from the *Product Documentation* section on the Micro Focus SupportLine Web site - [click here to download it](#).

Named and optional parameters

Two new types of parameter have been introduced for use during method invocation:

Named parameters

As part of the invocation expression, you can define a value for a parameter named in the method definition. The named argument must be specified after any positional arguments, and must not correspond to any of those preceding arguments.

Optional parameters

Optional parameters are parameters defined with a default value in the procedure division header of the invoked method. If none of the arguments passed in during invocation correspond to this parameter, the default value is used in the method; if an argument does correspond, the value that was passed in is used.

Delegates and events

A number of new features have been added that relate to delegates and events:



Note: Some of these features were also available in previous versions of Enterprise Developer.

The ATTACH and DETACH statements

Use these statements to attach or detach a delegate, method group or an anonymous method to or from an event.

The RUN statement

Use this statement to invoke a delegate once it has been created.

Combining delegates

Use the '+' operator to add a method group, anonymous method or another delegate to a delegate, and use the '-' operator to remove a method or another delegate from a delegate.

Method groups conversions

Use the METHOD keyword to specify a compatible method from a method group, and convert it to a delegate.

Problems View

The **Problems** view now has a 'Program' column that displays the name of the program in which the problem occurred. Click this column heading to sort the errors by program.

Program	Description	Resource	Path
	Errors (200 of 262 items)		
ZBNKE15.CBL	COBCH0008S Unknown copybook CABENDD spec...	ZBNKE15.CBL	/Mainframe1
SBANK00P.CBL	COBCH0008S Unknown copybook CBANKDAT spe...	SBANK00P.CBL	/Mainframe1
SBANK00P.CBL	COBCH0217S Preceding item at this level has zero ...	SBANK00P.CBL	/Mainframe1
SBANK00P.CBL	COBCH0008S Unknown copybook CBANEXT spe...	SBANK00P.CBL	/Mainframe1
SBANK00P.CBL	COBCH0217S Preceding item at this level has zero ...	SBANK00P.CBL	/Mainframe1
SBANK00P.CBL	COBCH0008S Unknown copybook CABENDD spec...	SBANK00P.CBL	/Mainframe1
SBANK00P.CBL	COBCH0012S Operand BANK-ENV-CICS is not dec...	SBANK00P.CBL	/Mainframe1
SBANK00P.CBL	COBCH0012S Operand BANK-COLOUR-SETTING i...	SBANK00P.CBL	/Mainframe1

If the column is not present in the view by default, you can add it through the **Configure Columns** dialog box in the **View Menu**.

Support for SOA



Restriction: This topic applies only when the Enterprise Server feature is enabled.

Enterprise Developer now includes support for creating Web service and Enterprise Java Bean applications using the Interface Mapping Toolkit (IMTK) in conjunction with Enterprise Server. If you are upgrading to this release from an earlier version of Enterprise Developer, you may need to apply for a new authorization code in order to access the functionality - please contact Micro Focus SupportLine to receive an updated authorization code. Note that the Enterprise Developer Personal Edition license does not support the IMTK functionality.

Upgrading from Net Express to Enterprise Developer

A new section in the product help, *Upgrading from Net Express to Enterprise Developer for Eclipse*, provides guidance on how to move existing applications either developed or debugged in the Net Express IDE into the Eclipse IDE.

XML Extensions

You can now use XML Extensions in your managed COBOL projects.

Use XML Extensions to import and export XML documents to and from COBOL working storage. Specifically, XML Extensions allows data to be imported from an XML document by converting data elements (as necessary) and storing the results into a matching COBOL data structure. Similarly, data is exported from a COBOL data structure by converting the COBOL data elements (as necessary) and storing the results in an XML document.

While importing or exporting data to or from XML documents, you can apply XSLT transforms to the data by using XSLT stylesheets.

For more information, refer to the XML Extensions User's Guide, available from the product documentation section of the SupportLine website (<http://SupportLine.MicroFocus.com/ProductDoc.aspx>)

What was New in Enterprise Developer 2.1 Update 1

Enterprise Developer 2.1 Update 1 provided enhancements in the following areas:

- [Compiler Directives](#)
- [DB2 ECM](#)
- [Enterprise Server](#)
- [HCO for DB2 LUW](#)
- [HCO for SQL Server](#)
- [Mainframe Compatibility](#)
- [OpenESQL Assistant](#)
- [PL/I Support](#)
- [SQL Option for DB2](#)
- [Debugging Enhancements](#)

Compiler Directives

You can now set SQL and PL/I-related Compiler directives and their values more easily, using a table of tick boxes in a project's Properties dialog box.

DB2 ECM

- Support for 64-bit DB2 ECM
- Support for 64-bit compile and runtime
- Support for DB2 10.1
- New DB2 SQL compiler directive option, BGP, to enable background parsing

Enterprise Server

The following new features and enhancements are available:

Clustering	Enterprise Server Clustering allows the scaling-out of work units, so that an increased number of operating system images can share the workload, resulting in high-performance, multi-system data sharing across all platforms.
Historical Statistics Facility	The Historical Statistics Facility has been extended to include the generation of JCL file records, increasing the amount of information customers have available to assist them in monitoring and tuning their Enterprise Server installations.
Recovery of in-doubt XA transactions	Some events in XA environments can result in 'in-doubt' transactions, where all parts of a composite transaction are not committed through all participating resource managers. The recovery of such in-doubt transactions is now supported.
SSL Support for the CICS Web Interface	Enterprise Server now allows clients and servers to identify themselves through X.509 certificates and participate in SSL-enabled conversations.

HCO for DB2 LUW

- Support for 64-bit DB2 ECM
- Support for 64-bit compile and runtime
- Support for DB2 10.1
- New DB2 SQL compiler directive option, BGP, to enable background parsing.

HCO for SQL Server

- Data Transfer tool enables you to:
 - Specify the number of error rows to identify before committing them to the log file.
 - Transfer tables in one SSIS package instead of one package per table (optional).
 - Execute DDL for tables
 - Transfer DDL
- The Generate DDL tool provides you the option of using SQL Server database collation or preserving mainframe DB2 collation with SQL Server.
- Migration of DB2 temporary tables has been enhanced.

Mainframe Compatibility

Enterprise Developer for z Systems is compatible with IBM System z9 mainframe hardware and later.

OpenESQL Assistant

Support for generation of PL/I SQL queries and copybooks.

PL/I Support

This release of Enterprise Developer includes support for developing, compiling and debugging PL/I applications, including PL/I-specific project types, file categories, and editors.


- Support for executing PL/I programs using IBM MQ Series software
- DEFINE ALIAS support for the Open PL/I Compiler for XML
- Support for TSO/IDCAMS
- Improved PLISRTD performance on UNIX/Linux
- Improved file I/O speed when using the PL/I language to read or write sequential, indexed and relative record files

- To enable PL/I for CICS, JCL, IMS you only need to set `ES_PLI_SUPPORT=Y`. You no longer need to set `ES_PL1_MFFH_JCL=Y`.
- Support for character picture data types A,X
- Support for `%XINCLUDE` (macro preprocessor and compiler)
- Support for `%OPTION` (macro preprocessor)
- Tolerance for source code containing `INLINE` and `ABNORMAL` attributes.
- Support for the `TINY` and `HUGE` built-in functions
- New `PLICTF` and `PLICTFHX` built-in functions
- Support for the `PARAMETER` attribute
- Support for the `NONINIT` attribute

SQL Option for DB2

- Background parsing support
- Enhanced XDB engine LOB support
- UTF-8 support for client-side expressions
- Support for indices on expressions

Debugging Enhancements

You can create a breakpoint for any program in the workspace that your application uses by clicking **Run > Add Program Breakpoint**, or clicking the  icon in the Breakpoints view, and entering the name of the file.

What was New in Enterprise Developer 2.1

Enterprise Developer 2.1 provided enhancements in the following areas:

- [ACUCOBOL-GT Data Types in Managed Code](#)
- [ACUCOBOL-GT Library Routines in Managed Code](#)
- [Associating file extensions with the COBOL language](#)
- [Compiler Directives](#)
- [.int, .gnt and .lbr File Types Support](#)
- [IMS Support](#)
- [Just-in-time Debugging](#)
- [Managed code enhancements](#)
- [OpenESQL](#)
- [UNIX Platforms Support](#)
- [Automatic Directives Detection and Setting](#)

ACUCOBOL-GT Data Types in Managed Code

ACUCOBOL-GT data types and `sign()` variants that were previously only available in native code are now supported in managed code. Use the Compiler directives `COMP1` and `COMP2` to set ACUCOBOL-GT behavior for those particular data types.

ACUCOBOL-GT Library Routines in Managed Code

ACUCOBOL-GT library routines that were previously only available in native code are now supported in managed code.

Compiler Directives

The following new Compiler directives are now available:

DISPLAY	Defines the default behavior of standard DISPLAY statements.
COMP1	Specifies the behavior of a COMP-1 data item.
COMP2	Specifies the behavior of a COMP-2 data item.
RESTRICT-GOTO	Generates a syntax error for GO TO statements that transfer control to outside of the current section.
ILSMARTRESTRICT	Limits the generation of properties in ILSMARTLINKAGE classes to non-redefining elementary items.

The following Compiler directive has changed:

- DATAMAP - Two new parameters allow you to display either the address or offset values for data items in your program.

.int, .gnt and .lbr File Types Support

Support has been added within the IDE for compiling native COBOL applications to the Micro Focus legacy formats `.int` and `.gnt`, and to package these files as a Micro Focus library file (`.lbr`). Improvements include:

- An option to compile all native COBOL projects to `.int` and `.gnt` code. You can set this in your project's properties.
- An option to package the `.int` and `.gnt` files produced by the project as a Micro Focus `.lbr` library files.
- Improvements to the Net Express Project Import wizard that enable you to convert existing Net Express projects to Enterprise Developer projects that compile to `.int` and `.gnt` code.

IMS Support

Mainframe projects contain categories for IMS resources (`.dbd`, `.mfs`, and `.psb` files), in which files with these extensions are automatically put when you add them to the project. Each file type has its own editor that colorizes tokens to aid development. Building a project containing these file types automatically generates DBD, PSB, and MFS files.

Just-in-time Debugging

Enterprise Developer now supports "Just-in-time" debugging: when a run-time error occurs, or an application calls `CBL_DEBUGBREAK`, the IDE can start automatically with the debugger attached to the failed process.

Managed code enhancements

Delegates and Events	Delegates and events are now implemented on the JVM platform. This release provides support for combining delegates, using the <code>METHOD</code> keyword to specify method groups, and implicit conversion from a method group or an anonymous method to the suitable delegate type.
Handling Invalid Numeric Data	The handling of invalid numeric data is controlled by a number of Compiler directives: <code>HOSTNUMMOVE</code> , <code>HOSTNUMCOMPARE</code> and <code>SIGNFIXUP</code> . These directives were previously only available in native code but are now supported in managed code.

Resolving Types

In this release, the Compiler attempts to resolve types to those defined in the current compilation unit wherever possible. The Compiler will attempt to resolve such types to an external name only if no suitable type exists in the current compilation unit. For example:

```
$set ilusing"System"  
class-id MyNamespace.EventHandler.  
01 o type EventHandler.  
end class.
```

In this release, `01 o type EventHandler.` resolves to `MyNamespace.EventHandler` and not to `System.EventHandler`.

Specifying Properties

In previous versions of the products, properties declared using the PROPERTY keyword on a data item were generated as final properties. Starting with this release, they are generated as virtual properties by default. In order to make the properties final, you need to specify the word FINAL following PROPERTY. This change may affect the generation of Proxy classes, for example, if you are using WCF.

OpenESQL

JDBC

JDBC has been enhanced to support two new directives:

JNDI Enables you to specify a JNDI class that looks up connection strings.

JNDIENC Enables applications to use the JNDI Environment Naming Context (ENC) when looking up JDBC data source names using JNDI.

ODBC

Added support for a generic one-phase commit for ODBC XA switch module.

SQL Compiler Directive Options

OpenESQL has been enhanced to support the the following new SQL compiler directive options:

DATE Controls the reformatting of date values in output parameters and in input parameter character host variables when DETECTDATE is also specified.

TIME Controls the reformatting of date values in output parameters and in input parameter character host variables when DETECTDATE is also used.

DATEDELIM Specifies a single character as the delimiter between the year, month, and day components to override the default delimiter determined by the HCOSS DIALECT or DATE directive specification.

TIMDELIM Specifies a single character as the delimiter between the hour, minute, and second components to override the default delimiter determined by the HCOSS DIALECT or TIME directive specification.

TSTAMPSEP Specifies a single character as the separator between the date and time parts of timestamp and date/time data.

OpenESQL Assistant

OESQL Assistant now supports updateable cursors.

SQL Server

We now support Microsoft SQL Server 2012.

UNIX Platforms Support

Support for remote development and deployment of projects is now available on UNIX and Linux.

Automatic Directives Detection and Setting

The IDE automatically determines and sets the COBOL dialect, and the CICS and SQL directives on local or remote native COBOL projects. You can also start a directives scan from within COBOL Explorer - select

Determine Directives from the context menu for the projects or the COBOL source files. This triggers a scan to determine the COBOL dialect, the CICS and SQL settings, and sets them as Compiler directives on a file or project level respectively. At the end of the scan, you view the results and choose to apply the changes.

What was New in Enterprise Developer 2.0

Enterprise Developer 2.0 provided enhancements in the following areas:

- [Automatic Directives Detection and Setting](#)
- [COBOL Explorer view](#)
- [COBOL File Search](#)
- [Compiler Directives](#)
- [Compiling Single Files](#)
- [Enhancements to Developing Applications on a Remote Machine](#)
- [Converting Projects](#)
- [Copybook Context](#)
- [Debugging Enhancements](#)
- [Eclipse 3.7](#)
- [JVM COBOL File Handler](#)
- [Library Routines](#)
- [Managed COBOL Language Features](#)
- [Data Access](#)
- [Remote COBOL JVM Projects - Early Release](#)
- [Run-Time Tunables](#)
- [Samples](#)
- [Vision Data File Searching](#)

Automatic Directives Detection and Setting

It is now possible to automatically determine and set the COBOL dialect and SQL directives on native COBOL source code. In the COBOL Explorer, select **Determine Directives** from the context menu for the projects or the COBOL source files. This triggers a scan to determine the COBOL dialect and SQL settings and sets them as Compiler directives on a file or project level respectively. At the end of the scan, you view the results and choose to apply the changes.

This feature works with local or remote native COBOL projects only.


COBOL Explorer View

Enterprise Developer now includes the COBOL Explorer view in the Eclipse IDE. This allows you to navigate around COBOL projects in a more useful and convenient way than Eclipse's own Navigator view.

The COBOL Explorer view includes the following features to help you manage your projects:

- For COBOL Project, Remote Project, and Mainframe COBOL Project types, COBOL Explorer adds category folders that automatically group together your project's COBOL programs, copybooks, and output files. (These folders are not physical folders on the disk, but effectively headings for certain file types.)
- To simplify navigation around your project, you can hide some files that are part of the project, such as the `.cobolBuild`, `.cobolProj`, and `.project` files, and the `.settings` folder and its content in JVM projects. You do this by opening the view menu, selecting **Customize View** and choosing from the options.

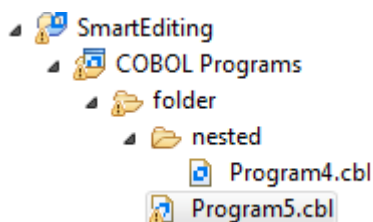
Filters Choose types of content to hide in COBOL Explorer.

Option	Description
.*resources	Any files of type . * such as .cobolBuild, .cobolProj
Closed projects	Resources in closed projects never display, but you can choose to hide all closed project icons  too
Empty folders inside COBOL categories	Folders mapped to the COBOL Programs and Copybooks that contain no COBOL program or copybook files
Non-COBOL projects	Projects of types other than those under Micro Focus COBOL - for example AspectJ, Plug-in, or Update-site projects
RSE Internal projects	Projects generated by the RSE plug-in

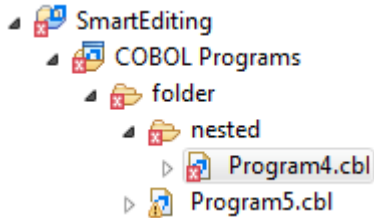
Content Choose types of content to show in COBOL Explorer.

Option	Description
Working Sets	Working sets are subsets of workspace resources you can choose to show or perform options on. To define a working set, click Project > Build Working Set > Select Working Set > New .
COBOL Elements	Non-resource types that are COBOL-specific: <ol style="list-style-type: none"> 1. x 2. category folders, including those for COBOL programs and copybooks, and BMS macro source files 3. icons for the different types of COBOL files. 4. overlays for build errors and warnings 5. some context menu items
Resources	COBOL projects, COBOL programs and copybooks. Has effect only if the COBOL Elements option is unchecked.

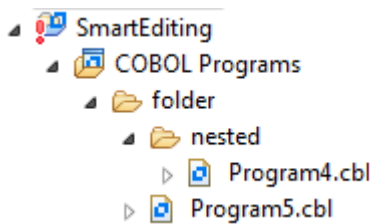
- COBOL Explorer helps you fix problems by using icons to identify files and containers that cause build errors and warnings.
- A file that generates a warning, and any containing folders and categories, is marked with a yellow warning sign. For example:



- A file that causes an error, and any containing folders and categories, is marked with a cross. In this example the icons indicate the most severe problem is the error caused by `Program4.cbl`, and the folders are marked with error icons despite `Program5.cbl` generating a warning:



- A project that suffers from a dependency error is marked with a red exclamation mark; its contents remain unmarked. In this example the project depends on another project that is closed, causing a build path problem:



Also, the context menus are reorganized and include some additional tasks such as the **Determine Directives** command.

You access COBOL Explorer in the same way as other Eclipse views, by selecting **Window > Show View**.

COBOL File Search

Enterprise Developer now includes the COBOL File Search feature in the Eclipse IDE.

You can use the Micro Focus search feature to make it easier to find files within your projects:

- Select **Search > Micro Focus**, and type the file search pattern.

You can filter results based on:

- File type: choose from COBOL programs, copybooks, PL/I files, JCL files, and BMS and macro source files.
- Only those files that cause build errors.
- Only those files that cause build warnings.

You can control the scope of your search:

- Workspace - searches entire workspace.
- Selected resources - searches only the resource currently being edited or highlighted in the tree view.
- Enclosing projects - if the editor is focussed on a resource or resources are highlighted in the tree view, then the search is within the entire project containing the resources.
- Current program - searches the resource currently being edited, as well as within any copybooks or include files used by the resource.
- Working set - see *Working Sets* for more information.
- Select **Search > Run Stored Micro Focus Search Query** and choose to find either:
 - Only those files that cause build errors.
 - Only those files whose properties override the project's properties (for example **Language dialect** or **Compile for debugging** settings).

Whichever way you use the Micro Focus search the results are shown in the **Search** view.

After you have performed a search you can edit the properties of any of the program files listed in the search view. You can even simultaneously edit the properties of multiple program files which have compilation settings available on them, that is program files rather than copybooks or include files. Highlight the files you want to change, right-click **Properties**, and then check **Enable file specific settings**. Change the file properties as required and then click **OK**.

You can save your search criteria and give it a label which is added to the list of stored queries available when you select **Search > Run Stored Micro Focus Search Query**:

1. In the Search tab, click  **Save the Current Search**.

This opens the **Add Micro Focus Search** dialog box.

2. In the **Search label** field, type a search label.

- You can assign a keyboard shortcut by typing the required keystrokes in the **Binding** field.
- Optionally, you can modify the current search criteria.

3. Click **OK**.

The search label is added to the list under **Search > Run Stored Micro Focus Search Query**.

You can also add, edit, and remove your searches from **Window > Preferences > Micro Focus > Search**.



Note: Do not open the **Keys** preferences page (**Window > Preferences > General > Keys**) while editing the binding in the **Search** preferences page. Opening the **Keys** preferences page will prevent any changes you make to the search bindings from being saved.

Compiler Directives

The following new directives are now available:

- **COPYSEARCH** - enables you to specify how copybooks are located. You can choose between usual Micro Focus COBOL behavior or usual RM/COBOL behavior.
- **ILSMARTNEST** - enables you to nest **ILSMARTLINKAGE** classes inside the program class in which they are defined. This makes it possible to have multiple programs in a single compilation unit that include linkage records with the same name.

The following directives have been changed:

- **DIALECT(RM)** - now accepts a new parameter, **RM**, which enables the RM-compatible functionality that the **RM** directive used to enable.
- **ILREF** - can only specify a **.class** as a parameter, and not a **.jar** file or other file types.
- **ILUSING** - when set on a single file using the **SET** statement, `$set ilusing`, the directive only affects that file.

Compiling Single Files

It is now possible to compile individual COBOL source files without rebuilding your entire project. **Build Automatically** on the **Project** menu must be turned off.

Enhancements to Developing Applications on a Remote Machine

Previously, this product depended on Samba or NFS to transfer and manage the project files on a remote UNIX machine. Remote Server Explorer (RSE) was only used to build the applications on remote machines.

"Remote file system (RSE)" is now the default option in the Create Remote COBOL project wizard for managing the files of remote COBOL projects. RSE establishes a connection with the remote machine and is used to transfer and manage the project files on it.



Important:

- You must use SSH connection and not DStore for remote COBOL projects created with Remote file system (RSE).
- You can still use Samba and NFS as file system providers. On system where SSH is not allowed, you can use SAMBA and DStore connection to transfer and manage the files on the remote system.

Converting Projects


It is now possible to convert Net Express projects into Eclipse projects and use them with Enterprise Developer. To do this, use the Net Express Project Import and Convert Wizard available from **File > Import > Convert NetExpress Projects to COBOL Projects**. The wizard analyzes the Net Express project file and its configuration settings, creates Eclipse projects based on this information, imports the existing source code into them, and sets the requisite project and file properties from the original Net Express project.

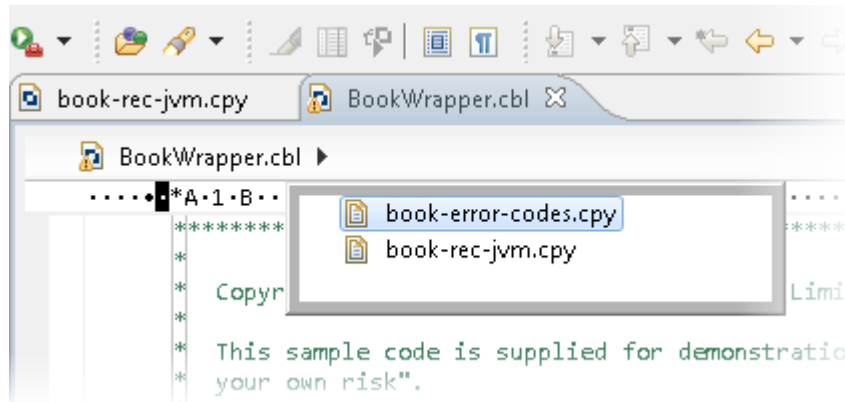


Note: If you import a Net Express project that uses functionality that is not supported in Enterprise Developer, the wizard will still produce an Eclipse project. In some cases you might be able to perform additional steps (such as editing source files, installing AddPacks, or reworking parts of the application) in order to successfully compile and run it. See *Converting Net Express Projects to Eclipse Projects* for more information about the limitations of the wizard.

Copybook Context

When you view or edit a copybook, you need to see it in the context of the program that references it, as its appearance and usage can change depending on the program. For example, data items that are not used in the program are struck through, and horizontal lines indicating the start and end of code lines can show depending on the source format of the program. The Program Outline view can differ between contexts too.

A breadcrumb trail at the top of the Editor indicates which program and copybooks are the current context - click Toggle Context Breadcrumbs  in the Eclipse toolbar and use the drop-down list to view and open the dependent copybooks.



Debugging Enhancements


Enterprise Developer 2.0 provides the following enhancements to debugging:

Program Breakpoints



Note: Program breakpoints are supported in native COBOL only, and are not supported with nested programs.

You can now set program breakpoints, which break into a program whenever it or one of its entry points is called.

To set or toggle a program breakpoint, double-click in the left margin, next to the Procedure Division heading. The  icon appears when the program breakpoint is set.

Step out of OSVS perform statements You can now step out of a perform statement when PERFORM-TYPE(OSVS) or DIALECT(OSVS) is set.

Eclipse 3.7

Enterprise Developer 2.0 uses Eclipse 3.7.1 Indigo.

JVM COBOL File Handler

Use the JVM COBOL File Handler, a File Handler written in purely JVM COBOL managed code, when you are deploying to environments that do not allow the use of native code such as the default Micro Focus File Handler.

Library Routines

The following CTF library routines are now available in COBOL for JVM:

```
CBL_CTF_COMP_PROPERTY_GET
CBL_CTF_TRACE
CBL_CTF_TRACER_LEVEL_GET
CBL_CTF_TRACER_GET
CBL_CTF_LEVEL
```

The following routine has been enhanced:

- The CBL_SEMAPHORE_ACQUIRE routine now accepts a `timeout` parameter.

Managed COBOL Language Features

The following new syntax elements are now available in managed COBOL:

Local Variables	In managed COBOL, Data items can now be declared in the procedure division, using the DECLARE statement. In addition, they can be declared inline as the iterator in a PERFORM statement, or as an exception message in a TRY ... CATCH ... FINALLY statement block.
Collections	There are two new collection types in managed COBOL: LIST and DICTIONARY. For a LIST, you can add elements to a list, retrieve the nth element of the list, replace the nth element, iterate through the list and clear the list. For a DICTIONARY, you can add key value pairs, retrieve a value corresponding to a key, to replace the value corresponding to a key, iterate through the dictionary and clear the dictionary.
Properties	In managed COBOL, a property can now be defined using PROPERTY-ID and GETTER and SETTER phrases to access to the property. The previous technique of specifying the keyword PROPERTY on a data declaration is still available.
Indexers	In managed COBOL, an indexer can now be defined using INDEXER-ID and GETTER and SETTER phrases to access the indexer value. Indexers are similar to properties, except that their accessors take parameters. Indexers allow instances of a class or valuetype to be indexed just like arrays.
Zero-based Indexing	The managed COBOL syntax for arrays now uses zero-base indexing to access arrays when square brackets are specified. For backward compatibility, one-base indexing is used when round parentheses are specified.

The Help now contains code examples comparing managed COBOL and Java.

Data Access

Enterprise Developer version 2.0 provides the following enhancements:

- Improved IDE integration with SQL directives - now supports handling of deprecated and removed directives. Also supports filtering of the choices offered to the user by product type, project type, and platform.
- OpenESQL has been enhanced and it now:
 - defaults to optimal performance
 - supports 64bit ODBC across all platforms
 - OpenESQL now supports JDBC across all platforms

Remote COBOL JVM Projects - Early Release

You can use Eclipse to develop COBOL JVM projects on a remote UNIX/Linux machine. The source code resides and is being compiled and debugged on the remote machine.

Run-Time Tunables

Enterprise Developer 2.0 provides the following new tunable:

- `subsystem_cancel_mode` - use this to override the default cancel mode when you use the `CBL_SUBSYSTEM` library routine to cancel a subsystem.

Samples

The following new samples are now available:

- Collections - demonstrates the managed COBOL collections syntax
- Local Variables - shows how to declare data items in the procedure division in the `DECLARE`, `PERFORM` and `TRY` statements
- The code in the Properties sample has been enhanced to use the new `PROPERTY-ID` syntax. The sample also includes a sample program for Indexers which illustrates the new `INDEXER-ID` syntax.

Vision Data File Searching

Enterprise Developer 2.0 provides the following new ACUCOBOL-GT compatible environment variables to help search for Vision data files at run time:

```
APPLY_FILE_PATH  
FILE_CASE  
FILE_PREFIX  
FILE_SUFFIX
```

What was New in Enterprise Developer 2010

Enterprise Developer 2010 provided enhancements as part of the following releases:

- [New Features in Enterprise Developer 2010 R4 Update 2](#)
- [New Features in Enterprise Developer 2010 R4](#)

Features Added in Enterprise Developer 2010 R4 Update 2

New Platforms Support

Support for Enterprise Developer for Eclipse has been added for the following platforms:

- x86-64 running Red Hat Enterprise Linux 5.7/6.1

- x86-64 running SuSE SLES 11 SP1

Support for Enterprise Developer Development Hub has been added for the following platforms:

- x86-64 running Red Hat Enterprise Linux 5.7/6.1
- x86-64 running SuSE SLES 11 SP1

Support for COBOL for JVM has been added for the following platforms:

HP IA 11.31 - 32/64-bit
 x86-64 running Red Hat Linux 5.6/6.1 - 32/64-bit
 SPARC running Solaris 10 - 32/64-bit

OO COBOL Class Library Reference

On Windows, Help for the following OO COBOL class libraries is available:

Base class library
 GUI class library
 OLE class library
 OLE Automation class library

The Help is available in the file `nrxclr.chm`, which is installed in the Help folder of your installation. The default location is `%ProgramFiles(x86)%\Micro Focus\Enterprise Developer\Help`.

To open the help, double-click `nrxclr.chm` in Windows Explorer.

Help for the OO COBOL class libraries are available from the Micro Focus SupportLine Web site, as follows:

1. Go to the Server Express documentation, at <http://supportline.microfocus.com/documentation/books/sx51ws02/sx51indx.htm>.
2. Click *Reference > OO COBOL*.
3. Expand *OO COBOL Class Library Reference*.

OpenESQL

OpenESQL now includes the JDBC preprocessor option that you can use to access databases for applications running under the Java Virtual Machine (JVM).

Features Added in Enterprise Developer 2010 R4

ACUCOBOL-GT Compatibility

The Compiler and run-time continue to provide support for ACUCOBOL-GT. The directive ACU is the main switch for turning on ACUCOBOL-GT compatibility. The ACU directive enables various ACUCOBOL-GT syntax extensions and other language elements. Additional ACUCOBOL-GT compatibility features include the following:

- When using a CALL statement, the USING and GIVING/RETURNING phrases can now appear in either order.
- The following ACUCOBOL-GT standard library routines can now be used with Enterprise Developer in native code:
 - C\$CALLED BY
 - C\$CALLERR
 - C\$CHDIR
 - C\$MAKEDIR
 - C\$MEMCPY

- C\$MYFILE
- C\$PARAMSIZE
- C\$RERR
- M\$ALLOC
- M\$FREE
- M\$COPY
- M\$FILL
- M\$GET
- M\$PUT
- WIN\$VERSION
- The following ACUCOBOL-GT 'ccbl' compiler options can now be used with Visual COBOL:
 - -E, -V
 - -Cv
 - -Da, -Db, -Dd31, -DL1/2/4/8, -Dq, -FpRounding
 - -La, -Li, -Lc, -Lf, -Ll, -Lo, -Ls, -Lw

Note: The output that these list options provide differs in Visual COBOL.

 - -Qm
 - -Rc, -Rn, -Rw
 - -Sa, -St, -Sd, -Sp, -S1...-S9
 - -noTRUNC, -truncANSI, -Dz
 - -Td, -Te
 - -Vc
 - -Za, -Zc, -Zl, -Zn, -Zs, -Zi, -Zr1, -Zy, -arithmeticVSC2

Full ACUCOBOL-GT compatibility is documented under the *Programming* section in the product help.

COBOL for Java Virtual Machine (JVM) Support

In this release you can compile COBOL applications to JVM byte code (.class files) so that they can be run on a JVM. There is support in the IDE to edit, compile and debug JVM COBOL applications. This release includes Managed COBOL - COBOL with extensions to support the JVM framework plus OO syntax support.

Features include support for

- New managed COBOL syntax:
 - The SYNC statement, which marks a statement block as a critical section by obtaining the mutual-exclusion lock for a given object, executing a block of statements, and then releasing the lock.
 - Extension methods, enabling you to extend an existing class with new methods without the need to recompile the existing class
 - Java style inner classes, which define a nested class within another class. These follow all the methods belonging to the containing class.
 - The STATIC keyword, enabling you to mark methods and data as static.
 - Enumerators, which represent a list of constant values. You can declare an enum type that defines the values and symbolic names for them, and refer to the values by name in your code.
- Seamless interoperation between JVM COBOL and native COBOL
- The ability to add Java projects to the classpath for COBOL JVM projects, and COBOL JVM projects to the classpath for Java projects
- Red Hat, AIX and SUSE platforms

Embedded HTML

We now support the use of Embedded HTML (EHTML) in COBOL CGI programs, which enables you to output HTML directly from your applications.

HCO for SQL Server (HCOSS)

Host Compatibility Option for SQL Server (HCOSS) is a new component in Enterprise Developer for Eclipse. However, it was previously available in the Visual COBOL 2010 Enterprise Option product. The following are HCOSS features that are new since its previous release in Visual COBOL 2010 Enterprise Option:

- HCOSS supports client-based applications compiled with the DBMAN=ADO and DIALECT=MAINFRAME compiler directives when the application targets .NET Framework versions 2.0, 3.0 and 3.5, or .NET Framework version 4.0, but only when the application does not contain dynamic SQL that uses an SQLDA structure.
- HCOSS supports dynamic SQL that uses the SQLDA structure in client-based applications compiled with the DBMAN=ADO and DIALECT=MAINFRAME compiler directives, but only when the application is built to target .NET Framework version 4.0.

Improved Usability

The following improvements to COBOL development in Eclipse have been made:

- | | |
|--|--|
| Dialog box improvements | The dialog boxes governing build configuration, debug configuration, and launch configuration are redesigned to make them easier to use. |
| Autocompletion and Content Assist | The IDE now includes autocompletion and content assist editing features. |

Language Improvements

The following improvements have been made to managed COBOL:

- | | |
|--|--|
| Extension methods and extending operators | Managed COBOL now supports extension methods. This feature enables you to add methods to existing types without the need to edit or recompile the code. You can also extend operators. |
| The SYNC modifier for methods | The SYNC modifier locks the values of the arguments sent to the method, so that they do not change while the method is processing. |
| Nested classes | In managed COBOL, a nested class can now be defined so that it can access the instance fields, properties and methods in its containing class. To allow this, you add the optional SHARING PARENT phrase to the nested class definition. |

Renamed Color Preferences

The COBOL Editor syntax-coloring scheme called "Micro Focus Net Express" is now called "Micro Focus Traditional".

The syntax-coloring element called "Identifiers" is now called "Identification Division Names".

Any existing settings for these options will not change.

Reporting of Linker Errors

Errors relating to linking, such as undefined entry points and libraries not being found, are now logged in Eclipse's Problems view.

RM/COBOL Compatibility

The Compiler and run-time continue to provide support for RM/COBOL. Additional RM/COBOL compatibility features include the following:

- The following RM/COBOL standard library routines can now be used with Enterprise Developer in native code:

- C\$Century
- C\$ConvertAnsiToOem
- C\$ConvertOemToAnsi
- C\$DARG
- C\$Delay
- C\$GetEnv
- C\$GetNativeCharset
- C\$LogicalAnd
- C\$LogicalComplement
- C\$LogicalOr
- C\$LogicalShiftLeft
- C\$LogicalShiftRight
- C\$LogicalXor
- C\$NARG
- C\$SetEnv
- C\$RERR
- DELETE
- RENAME
- The RM/COBOL file handler can now be used with Enterprise Developer, enabled by using the CALLFH"ACUFH" Compiler directive, and then configuring an add-on to the Vision file handler.

Full RM/COBOL compatibility is documented under the *Programming* section in the product help.

XML Extensions



Note: This functionality is supported in native COBOL only.

You can now use XML Extensions, the system that enables your COBOL applications to interact with XML documents, with Enterprise Developer.

XML Extensions has many capabilities. The major features support the ability to import and export XML documents to and from COBOL working storage. Specifically, XML Extensions allows data to be imported from an XML document by converting data elements (as necessary) and storing the results into a matching COBOL data structure. Similarly, data is exported from a COBOL data structure by converting the COBOL data elements (as necessary) and storing the results in an XML document.

For more information about XML Extensions, refer to the *XML Extensions User's Guide*, available from the RM/COBOL product documentation set, in the SupportLine section of the Micro Focus Web site.

Significant Changes

This section describes significant changes in behavior or usage in each successive release. These changes could potentially affect the behavior of existing applications or impact the way the tools are used.

Where present, the numbers that follow each issue are the Support Incident Numbers followed by the Reported Problem Incident (RPI) number (in parentheses).

Significant Changes in Enterprise Developer 4.0

This section describes significant changes in behavior or usage. These changes could potentially affect the behavior of existing applications or impact the way the tools are used.

Where present, the numbers that follow each issue are the Support Incident Numbers followed by the Reported Problem Incident (RPI) number (in parentheses).

- [Application Server JCA Support](#)
- [Application Workflow Modeller](#)
- [CICS Support](#)
- [Code Coverage](#)
- [Codeset Support](#)
- [Common Communications Interface](#)
- [Communications Server](#)
- [Compiler](#)
- [Data Tools](#)
- [Documentation](#)
- [Enterprise Server](#)
- [Enterprise Server Auditing](#)
- [Executables require relinking](#)
- [File Handling](#)
- [HCO for Microsoft SQL Server](#)
- [IDE](#)
- [Interface Mapping Toolkit](#)
- [JCL Support](#)
- [Mainframe Access](#)
- [MF Server Administrator \(GUI\)](#)
- [Micro Focus Directory Server](#)
- [PL/I Compiler](#)
- [Run-Time System](#)

Application Server JCA Support for Enterprise Server

- With this release, all CICS-related Java artifacts have been delivered using a new directory structure under the `javaee-ccl` directory of your Enterprise Developer installation directory. The `javaee-ccl` directory structure mimics the structure of the `javaee` directory. For example, `mfcc1.jar` is no longer located in the `bin` subdirectory, but can be found in the `javaee-ccl\common` directory.

Application Workflow Modeller

[Back to the list](#)

- Source or listing files are now opened in read-only mode when a browse action is executed in the ChangeMan attachment model.
3138354 (1112546)
- A new function package, Micro Focus Background Parser, enables you to define the SYSLIB or PROCLIB concatenation in which to search for COBOL copybooks, PL/I or JCL include files on the mainframe or on network drives. The package also enables you to define any identifiers which determine the structure of the local cache for COBOL copybooks, PL/I or JCL include files.
3132249 (1111922)
- The Endeavor attachment application now validates the input of a CCID or a comment. They must not be blank.
- Added the new "Modeled Edit Sessions Only" AWM model attribute to control whether modeled editor actions should be available in non-modeled edit sessions.

CICS Support

[Back to the list](#)

- Client ECI timeouts now propagate CTG requests to ABEND AITH the ECI server program.

- The update resource definition file utility, CASRDTUP, now supports the DELETE command. Note that the REMOVE option is not currently supported.

2885020 (1108004)

Code Coverage

[Back to the list](#)

- Schema changes that affect the test coverage results generated from the `tcutil` utility mean that if you propagate the results to a third-party application (for example, an XSLT processor), and rely on the `<copyFileCoverage>` element, you need to alter your transformations to focus on `<sourceFileCoverage>` instead. The element was renamed to more appropriately reflect its contents, as `tcutil` now gives global coverage for all source files (not just copybooks).

Codeset Support

[Back to the list](#)

- Code-set mappings between ASCII and EBCDIC have been updated when Simplified Chinese is the language in effect. ASCII table 5210 now maps to EBCDIC CCSID 836 for SBCS conversions. This replaces the previously conversion (where ASCII table 1042 was used), which would convert the “\” character to “\$”.

3124321 (1111464)

- New single-byte character set tables for MFICODESET have been added in order to improve support for DB2 LUW - both for off-mainframe databases and for access to z/OS DB2. A number of existing MFICODESET mappings have also been updated. See 'Supported Country Codes' for a full listing of ASCII/ANSI <-> EBCDIC mappings.

3111843 (1109984)

Common Communications Interface

[Back to the list](#)

- You can now configure the Micro Focus Directory Server and enterprise server region's listeners to only use the server's configured SSL and TLS protocols and define a priority ordered cipher suite collection. This forces connecting clients to use the server's preferred ordered list of cipher suites when using the specified protocols.

2866265 (1105526)

- In some circumstances it was possible for a connection to incorrectly accept the identity of an SSL/TLS peer and allow a connection to complete when the connection should have been denied. This occurred due to a failure to check the peer's entire identity certificate chain. This has now been fixed. NOTE: You might need to correct your system's configured certificate chains that fail verification checks at secure connection creation time.
- In some circumstances it was possible to crash the CCITCP module when it was attempting to obtain detailed error information about a closed connection. This has been fixed.

Communications Server

[Back to the list](#)

- TN3270 conversations to Enterprise Server now correctly handle the receive (idle) timeout setting configured for the listener. There are also two new settings for configuring TN3270 timeouts, "Printers time out" and "Output resets timeout". See the online product documentation for more information.

3144133 (1113024)

- Web access to the Enterprise Server Console Log and Communications Server Log is now restricted when the region is secured using external security. Users will be required to provide a valid username

and password in order to view either log file. Web access to the logs can be controlled using the standard ACL definitions under the new "Communications Server" Resource Class, with resources "Enterprise Server Console Log" and "Communications Server Log". If these resources exist, users require read access to be allowed to view the logs. If they do not exist, the default behavior is to allow read access. The new resource class and resource definitions can be found in the "es_default_ldap(_msuser/_unix).ldf" file in the bin or etc sub-directory of your product install directory.

3113539 (1110155)

- MFCS listeners can now be SSL-enabled without the need to have DemoCA installed.

2868627 (1105777)

- MFCS no longer initializes the Security Facility if there are no External Security Managers defined for the region.

Compiler

[Back to the list](#)

- Programs containing EVALUATE statements of the form: EVALUATE true | false WHEN conditional-expression where conditional-expression included inline method invokes would give an RTS 114 error when run as .int code, and an "Illegal .int code" error when generated. This has been fixed. Also, short circuit evaluation is now correctly observed, such that when evaluating condition-1 AND condition-2, if condition-2 contains an inline method invoke and condition-1 is false, then the inline method invoke is not executed. Previously, despite being correctly evaluated, the inline method invoke in condition-2 was being executed. Similar behavior relating to OR evaluations has also been corrected.

3138510 (1112492)

- During compilation, characters within literals that are unknown in the current locale are now less likely to cause spurious errors. However, the correct (and safest) solution is to ensure that the locale has been set correctly, to match the source encoding of these characters. On UNIX, this means setting LANG, LC_CTYPE, or LC_ALL appropriately; each of these variables takes precedence over the former. On Windows, this means setting the system locale in the 'Region and Language' section of Control Panel.

3123935 (1111148)

- The Compiler now produces an E level message - COBCH1888 Typedef is defined differently in another external program - if different external programs have conflicting definitions of the same typedef name. (To restore the previous behavior, where the earlier definition was ignored, use the directive HIDEMESSAGE"1888".)
- An issue with the Compiler has been fixed so that in the RECORD VARYING clause, if the minimum and maximum lengths are specified, the maximum length must be greater than the minimum length.

Data Tools

[Back to the list](#)

- Records with an invalid value for a conditional field will no longer result in a match for that conditional layout.

2853226 (1103406)

Documentation

[Back to the list](#)

- You can use 'byte' or 'BYTE' as a synonym for the binary-char unsigned data type. As a result, 'byte' and 'BYTE' are now reserved words in Managed COBOL. Use the REMOVE"BYTE" Compiler directive to prevent an error being produced for existing programs that use the reserved word as a user-defined word.

3147576 (1113323)

- As of version 3.0, references to types within an assembly other than mscorlib need to be explicitly referenced. You can achieve this by using the ILREF Compiler directive. (Previously, in certain circumstances, the Compiler would allow access to types within the System.dll assembly without the need for an ILREF"System" directive.

3121002 (1111373)

Enterprise Server

[Back to the list](#)

- The External Security Facility (ESF) can now be configured to throttle large volumes of incoming Verify (user authentication / signon) requests to improve resilience to denial-of-service and brute force attacks. See "Verify Request Throttling" for more information.

3113639 (1110160)

- The LDIF files used to create the sample configuration for Enterprise Server LDAP-based security no longer create an empty "PHYSFILE" resource class. Changes in the JCL engine as of ES 3.0 caused most jobs to fail when submitted to a security-enabled region using such a configuration. See the product help for more information.
- The MLDAP ESM Module, part of the Enterprise Server External Security Facility, now supports the Argon2 hash algorithm for creating password verifiers. See MLDAP ESM Module Custom Configuration Information in the product help for more information. NOTE: The Argon2 hash is optional and not enabled by default.
- The MQ pages in ESMAC are now controlled by a new security resource, MQL. This enables you to either restrict or grant users access. 'MQL*' is a new resource that needs to be added under MFESMAC similar to existing resources such as 'PCT*' or 'XAT*'. The following is a sample export of the LDAP repository:

```
*****
# Sample security definitions for ESMAC MQ Listeners/Writers pages

#####
##### MQL*          ##
#####
dn: CN=MQL*,CN=MFESMAC,CN=Enterprise Server Resources,CN=Micro
Focus,CN=Program Data,DC=X
changetype: add
cn: MQL*
objectClass: microfocus-MFDS-Resource
microfocus-MFDS-Resource-Class: MFESMAC
microfocus-MFDS-Resource-ACE: allow:SYSADM group:alter
microfocus-MFDS-Resource-ACE: deny:*:execute
microfocus-MFDS-UID: mfuid
#description: Allow full access any ESMAC MQ Listeners/Writers Screen

*****
```

3143258 (1112990)

- You can now use the ECIResponse.getReturnCode() method to obtain the return code for any errors from Enterprise Server.

3142092 (1113248)

- Communication with the console daemon has been improved. Messages are displayed more quickly and requests are being processed more efficiently and, as a result, times for initialization and shutdown might be reduced.

3136867 (1112483)

- Administrators can now add, delete or modify XA resources in the Enterprise Server Administration Web UI while a region is running.

2589624 (1085625)

- CICS Web services has been updated to include the ES_TRANCLASS_CWI environment variable, which enables you to prioritize and limit Web requests in a TRANCLASS-enabled region. When the ES_TRANCLASS_CWI environment variable has been set to either TCPIPSERVICE (default) or URIMAP, a TRANCLASS-enabled region processes the elected transaction according to its priority and associated TRANCLASS.



Note: When using CICS® Transaction Server for z/OS®, Version 3 Release 1, and IBM CICS initiates the ATTACH transaction followed by the ALIAS transaction for a single Web request, this is intended to enable the application of different limitations and priorities on these transactions. Enterprise Server does not support this. Instead, a TRANCLASS-enabled region processes the requests as a single ATTACH transaction, and uses the ALIAS transaction for security checks.

3126381 (1111349)

- Web access to the Enterprise Server Console Log and Communications Server Log is now restricted when the enterprise server region is secured using external security. Users will be required to provide a valid username and password in order to view either log file.

Web access to the logs can be controlled using the standard ACL definitions under the new **Communications Server** resource class, with resources **Enterprise Server Console Log** and **Communications Server Log**. If these resources exist, users require 'read' access to be allowed to view the logs. If the resources do not exist, the default behavior is to allow 'read' access on entry of a valid username and password.

An example of the new resource class and resource definitions can be found in the `es_default_ldap_msuser.ldf` (Windows), or `es_default_ldap_unix.ldf` (UNIX) file, which is located in the `bin` (Windows), or `etc` (UNIX) sub-directory of your product directory.

- In UNIX environments, DB2 and ODBC switch modules can now be enabled together in the same region if the DB2 switch is built with the "-o" option.

3137455 (1112398)

Enterprise Server Auditing

[Back to the list](#)

- The maxRetryTime value in the audit configuration file now treats 0 as a no timeout time, and any negative number as an infinite timeout.

3150566 (1113592)

- Any extra information that was added to the syslog messages will now correctly appear in the structured data items.

Executables require relinking or recompiling



Note: Windows-only.

- Due to an internal change in version 4.0 of your product, you must at least relink any executable programs compiled prior to this version, to make them compatible with the latest run-time system. However, a full recompilation of your source code is the recommended action, to allow your executables to benefit from the product's latest programming and performance enhancements.

Relinking an executable without recompiling means using the original object code with the `cbllink` utility. Original object code is typically the binary file output (usually containing the `.obj` extension) produced during the original compilation process. An application can contain one or more binary files.

File Handling

[Back to the list](#)

- In some cases the ESF LDAP Security Administration Web Interface inadvertently removed users from groups when changing their password. This has been fixed.

3124294 (1111259)

- The ESF LDAP Security Administration Web Interface can now filter by Class and Resource name, description, and ACL. Previously, you could only filter on Class name.

2871549 (1106119)

- Setting the configuration option ASCIISOSI=ON will add the required SOSI characters to the relevant EBCDIC DBCS character strings, in order for them to be displayed or written out correctly.

3113802 (1110183)

- The OPEN mode of SYSOUT files now honors the DISP specified in the JCL.

3109432 (1109745)

- OPEN I-O of a virgin ESDS file now correctly returns a file status of 35, as it does on the mainframe.

2887724 (1108443)

- The TO parameter is now supported for TOT|TOTAL, MIN, MAX, and AVG in the TRAILER3 trailer item.

3135298 (1112194)

- The OUTFIL statement of a SORT operation now supports the NULLOFL parameter.

3110364 (1109822)

- The MFJSORT OUTFIL emulation now supports the VLTRIM parameter.

3107105 (1109820)

HCO for Microsoft SQL Server

[Back to the list](#)

- The HCOSS Manage Connections tool was incorrectly limited to using the default CCSID for mainframe DB2 connections, causing problems when mainframe database did not use the default CCSID of 37. This has been corrected by the addition of a text box that enables the setting of a specific CCSID value for the connection.

3130137 (1112180)

- When using the DIALECT=MAINFRAME setting, the OpenESQL STRICTPARSE SQL compiler directive option is now turned on by default for ODBC (DBMAN=ODBC). This ensures strict compliance with z/OS DB2 syntax.

If your applications require the OpenESQL preprocessor to behave as it did in an earlier release, compile using NOSTRICTPARSE.

IDE

[Back to the list](#)

- New functionality to support the debugging of CICS channels and containers has been added to this release.

2810448 (1098047)

- Code blocks in PL/I editor can now be folded. You can configure the PL/I editor to fold various block types. To do this, click Window > Preferences > Micro Focus > PL/I Editor > Folding, and check the options that you require.

2867993 (1105654)

Interface Mapping Toolkit

[Back to the list](#)

- Refresh Resources functionality has been added to regenerate existing service interfaces for COBOL programs whose linkage section has been updated after the service interface was initially created.

2849403 (1103036)

- Due to changes made to the **Configure Runtime Environment** dialog box in this release, after you upgrade your system, you must reenter all previously set values on this dialog box before running a service.

JCL Support

[Back to the list](#)

- You can now configure MFFTP to not process trailing blank spaces for a GET command, by setting the new control variable MFFTP_PROCESS_TRAILS_ONGET=FALSE and setting the LOCSTAT value TRAILINGBLANKS=FALSE.

3138375 (1112486)

- ICEGENER now does not copy the RECFM from SYSUT1 to SYSUT2 if SYSUT2 RECFM is not specified in JCL and when SYSUT1 is VSAM. It defaults to F/FB if SYSUT2 is non-VSAM, or V/VB if SYSUT2 is VSAM.

3131268 (1111805)

- An error in identifying the IDENT in IEBGENER when the record included high values at the start of the record has been corrected.

3109443 (1109759)

- FOR ED 2.3.2 HF7: The following changes have been made to the MF_UCC11 environment variable: MF_UCC11=Y - when running a job with DISP=(NEW,CATLG), it will not result in job failure caused by the data set already being cataloged. MF_UCC11=YR - when restarting a job with DISP=(NEW,CATLG), it will not result in job failure caused by the data set already being cataloged. For ED 3.0 HF1: The following changes have been made to the MF_UCC11 environment variable: The default behavior of MF_UCC11=Y has changed: DISP=NEW is changed to DISP=MOD only when the job is restarted. MF_UCC11=YA reverts to the old behavior of changing DISP=NEW to DISP=MOD even when the job is first run.

3107409 (1109722)

- The introduction of a new environment variable, ES_OLD_DYN_PDS, has changed the resulting behavior of dynamic PDS members when specified in a JCL DD statement. Default behavior (ES_OLD_DYN_PDS=N) no longer creates physical files or catalogs the PDS members when specified in a JCL DD statement. Set this variable to Y to enable the previous default behavior of creating the physical files and cataloging PDS members when specified in a JCL DD statement.

2845176 (1102528)

- To prevent unintended deletion of a GDG Bias, a GDG in use by one job may not be accessed by a second job until the first job has completed.

2816745 (1101569)

Mainframe Access

[Back to the list](#)

- Calling mfmon2.exe with the following options has the given effect:
 1. If you do not specify "EXTENSION" or specify "EXTENSION=" it will result in files downloaded without an extension.
 2. If you specify "EXTENSION=*" you will get an extension that matches the last part of the data set name. For example, if the source data set is a.b.c.d then the file extension will be .d.
 3. Everything else results in the extension specified. For example, "EXTENSION=txt" will result in files downloaded with a .txt extension.

MF Server Administrator (GUI)

[Back to the list](#)

- The total number of active sessions or clients in MFDS is now limited to 2000.

Micro Focus Directory Server

[Back to the list](#)

- The mfdss -g options D, O, and S have been added to the product Help.
2848627 (1102864)
- On UNIX, specifying an invalid user ID in the MFDS "General" options value for "Default process user ID" no longer causes the child process (such as a region start or stop request launched from the Web interface) to fail. The user ID under which the MFDS process was started will be used instead.
- MFDS now disables and limits the scope of Web listeners on add. It also emits a warning if any insecure Web listeners are displayed in the validate and listener tables.

PL/I Compiler

[Back to the list](#)

- Previously, storing PL/I named constants in big-endian form was incorrect and impaired some instances of restricted expression evaluation. In ED 4.0, named constants are stored in native form when the -bigendian compiler option is in effect.
3150931 (1113613)
- Open PL/I now supports the FLUSH statement.
3134294 (1112117)
- The %PROCESS statement now includes new compile-time options, -pporder and -options. See the product help for more information.
3107667 (1109861)
- Pointer subtraction in 64-bit mode now works as expected.

Run-Time System

[Back to the list](#)

- The run-time system now produces a more precise error message if a shared object of the wrong bitism is loaded.
- scan64 is no longer available. This has been superseded by the COBOL Analysis functionality in the IDE.

Significant Changes in Enterprise Developer 3.0

Enterprise Developer version 3.0 includes significant changes in the following areas:

- [AppMaster Builder](#)
- [CICS Support](#)
- [Compatibility AddPack](#)
- [Compiler](#)
- [Documentation](#)
- [Enterprise Server](#)
- [File Handling](#)
- [IDE](#)
- [IMS Support](#)
- [JCL Support](#)
- [JES-enabled Enterprise Server Regions](#)
- [Micro Focus Batch Scheduler Integration](#)

- [Micro Focus Directory Server](#)
- [OpenESQL](#)
- [PL/I Support](#)
- [Reserved words](#)
- [SQL: HCO for SQL Server](#)
- [HCO for DB2 LUW](#)

AppMaster Builder

- A new parm "STS" can be passed into the Program Generator. This parm tells the Program Painter to issue an error when it encounters a STUB keyword and the stub program is not specified in the application definition. This new parm is specified in the ISPSLIB\SSMPCOM file tailoring skeleton. Do a search for "STUBSCAN" to see where this parm is specified. By default, this new option is not in effect. To turn it on uncomment the line where the value is set to "YES". This feature is only available when generating programs via the IDE. When using the Distributed Generation Server (DGS) this setting is ignored.

2988372 (1108572)

CICS Support

- The value specified in the LENGTH field of the SEND TEXT command is now honored when the target terminal has its SOSI or PS attributes switched on. This means that trailing spaces will no longer be removed for SOSI/PS terminals.

2884877 (1107983)

Compatibility AddPack for Visual COBOL

Compatibility AddPack for Visual COBOL is now deprecated and will not be available with release 3.0 and later.

The Dialog System GUI and run-time components and Dialog System Character Mode (on Windows and UNIX) which were part of the AddPack are now installed as part of Enterprise Developer for Visual Studio. The run-time components are installed as part of Enterprise Server. These are only included for backward compatibility and Micro Focus does not recommend that you use them for new development.

The other components which were part of the AddPack, the Character-Based Data File Editor, CSBIND and Screens, will be available upon request from Micro Focus SupportLine.

Compiler

- Replacing a partial token no longer causes the second part of the token to appear on a new line. This could happen if the new text was larger than the text being replaced.

2869185 (1105763)

Documentation

- There have been a number of new reserved words added to the language in this release; these are all in effect under MFLEVEL "19", which is the default level when running under the MF dialect. Any of the following words are now not allowed under default conditions, and you will need to remove/rename them, or specifically configure your environment to allow them: ALLOCATE FREE JSON END-JSON

Enterprise Server

- WEB CONVERSE now supports a value of 0 for the USERLEN and PASSWORDLEN options which matches the behavior on the mainframe. There is no change to the behavior of WEB SEND (client) which is to return LENGERR 139/140 when USERLEN or PASSWORDLEN are 0.

2989188 (1108602)

IDE

- You can now open data files in RSE configured remote projects using the mfdatatools2 on the remote machine displaying back to an X display. You can configure the remote tools display by clicking Window > Preferences > Micro Focus > X Display. In the X Display (DISPLAY) field, type the display details. In addition, COBOL Explorer displays the Open With Remote Data File Editor context menu item for remote data files.

2852872 (1104085)

- Enterprise Server sign on credentials which were created on Windows when using the 32-bit versions of Enterprise Developer or Visual COBOL for Eclipse are not compatible with the 64-bit versions of those products, and vice versa. Users are required to re-enter credentials which are not compatible with the current version of the product. You only need to do this the first time they are used in the current product.

(625819)

File Handling

- Under certain circumstances, retry-lock requests on UNIX systems were sleeping for a second before re-attempting to acquire the lock. This no longer happens.

2988222 (1108521)

- A problem that generated a 39 error when attempting to access a VSAM file via an alternate index PATH element has been fixed.

2874622 (1106562)

- Under Mainframe emulation, when an empty virgin VSAM file is opened for input or for input/output, a file status of 35 is returned, regardless of the access mode specified.

2692826 (1093604)

- New PARM option SIGN-EBCDIC documented for MFJSORT.

2880037 (1107256)

- The MFJSORT ICETOOL emulation now supports the USING parameter in the SELECT operator.

2821176 (1099529)

- An empty, space-filler record is no longer produced at the end of an ICETOOL listing file.

(622789)

IMS Support

- Inconsistent results returned to a GET call issued from a PROCOPT=GO application when segments within the same database record were updated and locked by another session. Symptoms included an invalid (GE) status code or the wrong segment returned to a GN call. This has been fixed.

2862879 (1104899)

- The IMS control page is now secured according to the IMS* entity in MFESMAC.

2877405 (1107099)

JCL Support

- An emulation of the LE run-time option CBLQDA has been implemented in the form of the environment variable MF_CBLQDA. The default is OFF, but when set ON, QSAM files can be dynamically created.

3104665 (1109251)

- A new event call to the JCL user exit (MFJUXIT) has been included. The event 'job-queued' (event code 27) occurs when a JCL job has been parsed and put on either the Input Ready queue or the Input Hold queue.

2990395 (1108734)

- When reading commands, MFFTP now checks the charset (EBCDIC or ASCII) of each concatenated INPUT dataset to determine if it should be translated. The output is written in the charset which appropriate for the dataset.

2876407 (1106795)

- The PHYSFILE security setting will now be checked when a dataset entry is inserted into the catalog, or when a dynamic PDS member is created.

2857819 (1104844)

JES-enabled Enterprise Server Regions



Important: Existing JES-enabled Enterprise Server regions must perform this upgrade to continue using JES functionality. If you start a region without upgrading, the following message is displayed in the console log, and JES functionality is disabled:

```
JES000058E JES validation - Product requires new spool control files. Run
spool conversion utility, splconv, to create them.
```

In this release, the infrastructure of the JES spool repository has been upgraded in order to improve performance (see the *Enterprise Server* section of *What's New* for more details). For each JES-enabled region, before you can submit any jobs to JES, you must run the spool conversion utility; any newly created regions will automatically be configured to use the new spool files.

The conversion process converts the information in the existing spool files to a set of new files required under the new system. There should be no loss of data in moving to the new system.



Important: After converting to the new JES spool infrastructure, we recommend that you do not employ the increased range of job numbers until you have thoroughly tested the new infrastructure. Reverting back to the previous JES spool files is possible, but requires that you remove all job entries using more than 5 digits, for which there is no easy process.

To run the spool conversion utility

1. At the Enterprise Developer command prompt, set the following environment variables, as required:

Variable	Description
MFSYSCAT	The location and name of the current JES catalog. This is mandatory. If multiple catalog files are being used (that is, one or more user catalogs are defined), this environment variable should point to the catalog where JES spool datasets are cataloged.
MVSSPOOLDIR	The location of the current JES spool files. This is only mandatory if the location is different to the location of the JES catalog.
NEWSPOOLDIR	The location for the resulting JES spool files (that is, the SPL* files if running the conversion, or the old-style spool files if running the regression). This is only mandatory if the location is to be different to the location of the existing spool files.
MFSYSCAT_ERRS	The number of catalog errors that will be tolerated during the conversion process before the conversion is terminated. This is optional, and defaults to 100. If the process is terminated, the following error is produced:

Variable	Description
	MVSCATIO Error limit exceeded = <i>nnnn</i> (where <i>nnnn</i> is the error count)

2. Type `splconv`, then press **Enter**.

The conversion process runs, and details of the files and records converted are displayed to screen.

3. If MVSSPOOLDIR and NEWSPOOLDIR were set to different locations prior to running the conversion, you must now set MVSSPOOLDIR to the value of NEWSPOOLDIR.



Note: If you do not perform this step, you will receive the following error when you start your region, and JES fails to initialize:

JES000058E JES validation - Product requires new spool control files. Run spool conversion utility, `splconv`, to create them.

4. You can now start the region identified by the variables, and begin to submit jobs.

Reverting to the previous JES spool repository

If you plan to downgrade to an older version of your current product, any enterprise server region that has been converted or created with the current product will not be compatible. Before you can use them, they must be reverted to using the old JES spool repository.

Use this process if you are wanting to use a JES-enabled region in a previous version of the product:

1. At the Enterprise Developer command prompt, set the environment variables listed in the *To run the spool conversion utility* process, as required.
2. Set an additional variable, `SPLREGR_TARGET`, to `PRE30`.
3. If the region has started to use 6- or 7-digit job numbers (an enhancement new to version 3.0 of the product), then before you run the regression, you must remove the jobs that are using these numbers, and ensure that the `MF_MVSJOB` environment variable (or the `SPLJNO.dat` file if the variable is not used) does not specify an upper limit greater than 5 digits.



Note: If using the `MF_MVSJOB` environment variable, to revert to 5-digit job number support, set `MF_MVSJOB=<low number>+<high number>`, where *high number* is less than 0099999, and ensuring that the + sign is used to separate the low and high values.

4. Type `splregr`, then press **Enter**.

The regression process runs, and details of the files and records converted are displayed to screen.

You can now use the region identified by the variables in previous versions of Enterprise Developer.

Exporting the spool files to Enterprise Server for .NET

If you plan to export a region to Enterprise Server for .NET, any enterprise server region that has been converted or created with the current product will not be compatible. Before you can use them, they must be reverted to using the old JES spool repository.

Use this process if you are exporting a JES-enabled region to Enterprise Server for .NET:

1. At the Enterprise Developer command prompt, set the environment variables listed in the *To run the spool conversion utility* process, as required.
2. Set an additional variable, `SPLREGR_TARGET`, to the appropriate value:

DOTNET5

Create spool files for use on Enterprise Server for .NET, using a maximum of 5 digits for job numbers.

DOTNET7

Create spool files for use on Enterprise Server for .NET, using a maximum of 7 digits for job numbers.

3. Type `splregr`, then press **Enter**.

The regression process runs, and details of the files and records converted are displayed to screen.

You can now use the region identified by the variables in Enterprise Server for .NET.

Micro Focus Batch Scheduler Integration

- MFBSI no more requires that the name of the last folder specified in the path in MFBSI_DIR matches the name of the target Enterprise Server region. It is now possible to specify any directory path in MFBSI_DIR. You must use the same value for MFBSI_DIR in both the Enterprise Server configuration and for MFBSIJCL execution. When the name of the last folder specified in the path in MFBSI_DIR does not match the name of the target Enterprise Server region, you need to specify the region name in one of the following ways: 1. From the mfbisjcl command line, using the parameter REGION=ESRegionName. 2. In an MFBSI configuration file, in a line containing REGION=ESRegionName. You can specify this either in the default configuration file, mfbsi.cfg located using the path specified by MFBSI_DIR, or using a custom location or configuration file and the following parameter: CFG=[path]/name[.ext] See your product help for more details.

2869805 (1105939)

Micro Focus Directory Server

- In the Enterprise Server Administration HTML GUI, the "Scripts" page functionality is only available if administration access is restricted and the logged on user has sufficient authority.

3101625 (1109025)

- Some additional CSRF security measures have been added to the Enterprise Server Administration HTML GUI.

3101205 (1108916)

OpenESQL

The new OpenESQL OPTIMIZECURSORS SQL compiler directive option is turned on by default for ODBC (DBMAN=ODBC). This ensures that embedded SQL cursors that use WITH HOLD and FOR UPDATE clauses have the same data integrity across all databases.

If your applications require the OpenESQL preprocessor to use the behavior provided in an earlier release, compile them using OPTIMIZECURSORS=NO.

PL/I Support

- PUT output-items of type POINTER are now output as a hexadecimal string rather than a fixed bin integer value.
- Enterprise Developer 3.0 includes some changes that result in a significant performance improvement when assigning a null character string " to arithmetic data items, pointers, and structures. In versions of the product earlier than 3.0, pointer = "; assigned the value sysnull(). This has been fixed and null() is now assigned. See the product help for details about the -setnull compiler option.

(623184)

(627517)

- Previously, if writing output to a CTLASA (PRINT) file with the optional LINE(x) parameter having a value of x that was less than the current line, PL/I did not generate ASA Control characters to end of page and emit the output on the first line of the new page. This no longer occurs.

2871278 (1106114)

Reserved words

- There have been a number of new reserved words added to the COBOL language; these are all in effect under MFLEVEL"19", which is the default level when running under the MF dialect. Any of the following words are now not allowed under default conditions, and you will need to remove/rename them, or specifically configure your environment to allow them:

ALLOCATE
FREE
JSON
END-JSON

SQL: HCO for SQL Server

- The HCOSS Assess Application Migration tool now enables you to view the original DB2 SQL and the converted TSQL together in both the HCOSS tool itself and in any exported CSV files.

HCO for DB2 LUW

- Help buttons previously available on the XDB Server Configuration Utility, XDB Service Controller, Options Dialog, Bind Utility, and Linker Config (Link Profile) UIs have been removed with the exception of error messages in the SQLWizard, Migrate, and Declaration Generator.

Significant Changes in Enterprise Developer 2.3 Update 2

Enterprise Developer version 2.3 update 2 includes significant changes in the following areas:

- [AppMaster Builder](#)
- [Enterprise Server](#)
- [Compiler](#)
- [Eclipse IDE](#)
- [MF Directory Server](#)
- [Monitoring and Management](#)
- [JCL Support](#)
- [PL/I Support](#)
- [Run-Time System](#)

AppMaster Builder

- To correct a problem with generating programs when no APRPROG member is present, a new Output Generation preference has been added that, when enabled, creates a temporary APSPROG member at generation time if none is found. In addition, a new customizable APRPROG template is available to enable customization of the temporary APSPROG member.

2849347 (1103014)

Compiler

- Replacing a partial token no longer causes the second part of the token to appear on a new line. This could happen if the new text was larger than the text being replaced.

2869185 (1105763)

- There is no longer a problem opening an RM/COBOL indexed file when the program has a RECORD CONTAINS n CHARACTERS clause and there are record descriptions with lengths less than n. This situation previously caused a 39 error on the OPEN (other than OPEN OUTPUT) because there was a mismatch in the minimum record length.

Eclipse IDE

- This update modifies any existing connections defined in an Eclipse workspace. If the connections had any non-default values, those values could revert to their original default setting. After installing this release, before you use any remote connections for the first time, you need to check the settings and amend them as necessary.

2852872 (1103699)

Enterprise Server

- Previously, it was possible to install groups that should not have been installed. If a group name, as defined in the Startup List, did not exist in the list of Groups then the next Group in the alphabetical order would be loaded instead. Now, if a Group is not defined in the list of Groups, a warning that the Group could not be loaded is issued.

2869848 (619107)

- On UNIX, if the "File Path" setting was not specified in the configuration of an Enterprise Server, the environment variable TXFILEP was defaulting to \$COBDIR/etc/cas. This has been changed and TXFILEP is not populated when the "File Path" is not specified.

(618668)

MF Directory Server

- The "-n" option for the mfd command now supports hostnames as the network addresses in addition to IPv4 addresses.

2816871 (1099564)

Monitoring and Management

- Messages that are written to the console log by applications that perform "display upon console" now contain a standard message ID (CASM0001).

2854207 (1103659)

JCL Support

- A printer exit can now evaluate the spool copybook field lk-dest-source-is to find the source of the value in the spool DEST-NODE and DEST-USERID fields, to determine if it is from a SYSOUT 'DEST=' value or from a /*ROUTE PRINT value.

2852302 (1103304)

PL/I Support

- A problem with PUT STRING when compiled with the -bigendian option in effect has been fixed. Note that when upgrading to ED 2.3 U2 you must recompile any program using PUT SPRING.

2857374 (1104039)

- Previously, if using the -incafter option of the macro preprocessor to infuse a %INCLUDE with macro logic, you were required to schedule a second macro pass to preprocess the contents of the %INCLUDE. This is no longer necessary and the macro logic will be preprocessed on the first macro pass as if it was part of the original source.

- Previously, if writing output to a CTLASA (PRINT) file with the optional LINE(x) parameter having a value of x that was less than the current line, PL/I did not generate ASA Control characters to end of page and emit the output on the first line of the new page. This no longer occurs.

2871278 (1106114)

- Previously, if SIGPIPE was triggered when a PL/I program was in the callstack, then PL/I error handling got control and processed the ERROR. This no longer occurs. Users who rely upon catching SIGPIPE can re-enable it by adding a call to cobpostsighandler() to their code to turn it back on.

2859980 (1104399)

Run-Time System

- The `command_line_linkage` tunable has been deprecated; equivalent functionality can be achieved by using the `COMMAND-LINE-LINKAGE` Compiler directive instead.

2838118 (1101539)

Significant Changes in Enterprise Developer 2.3 Update 1

Enterprise Developer version 2.3 update 1 includes significant changes in the following areas:

- [Application Workflow Modeller](#)
- [AppMaster Builder](#)
- [Communications Server](#)
- [Data Tools](#)
- [Editor Writing Assistance](#)
- [Enterprise Server](#)
- [IBM Language Environment for OS/390 & VM Support](#)
- [IDE](#)
- [JCL Support](#)
- [Run-Time System](#)
- [SQL: OpenESQL](#)
- [SQL Option for DB2](#)

Application Workflow Modeller

- Tools used in the `File_Descriptor_Has_Action` relationship now support Resource Processing. This means tools defined within a dialog table action can modify properties of the selected table rows.
2835290 (1101493)
- The standard ED project model now contains additional context menu actions under "Open With..." for IMS DBD and PSP files to open either the DBD, the PSP or the IMS Database editors.
- You no longer receive a REXX error when recompiling a component from a baseline.

AppMaster Builder

- A new checkbox, "Use File Tailoring for Dataview generation", has now been added to the Database View dialog box in the AppMaster Builder (AMB) preferences. When checked, the Database Views will be generated using File Tailoring files and the AMB Builder Server.

2820090 (1099354)

Communications Server

- A new option, `[Operation] synchronous=yes` (or 1, or Yes, or y or Y) is now supported in the Fileshare listener configuration.

Data Tools

- When filtering a data file, if there is no valid temporary directory set, you are prompted to set one using the option in the Preferences dialog box.
- The editor no longer allows you to open a file if the file size (without header size) is not a multiple of the record size on disk; an error is produced instead.
- The editor no longer allows you to open a file if the file size without header size is not a multiple of the record size on disk; an error is produced instead.

- The level numbers displayed in a record layout correspond to the levels used in the .idy file that was used when the structure file was created.

Editor Writing Assistance

- IntelliSense (Visual Studio) or Content Assist (Eclipse) suggestions are no longer offered if you start typing numbers and automatic triggering of suggestions is enabled.

Enterprise Server

- In situations where the crossregion option of the shareoptions parameter is either 1 or 2, both a CICS and a JCL job could have the same file open for IO. This has been fixed. In addition, the '-oo' (and the synonymous '-o ') option of the casfile command has been deprecated. If '-oo' is used, you will now receive a return code 4 (warning). Instead of '-oo', you need to use the '-ooi' option (open immediate). If '-oo' is used with a region configured for VSAM shareoption use, the '-ooi' option is now forced.

2839272 (1101672)

- When using an external security manager to secure a region, the EXEC CICS START TRANSID API now works as expected and honors the USERID option. Previously, the started transaction would run under the authority of the user that executed the EXEC CICS START API.

2835112 (1101201)

IBM Language Environment for OS/390 & VM Support

- When calling SETENV, SYSOUT(?) can be specified. This designates the file as a spool file but will not update its CLASS.

2841220 (1101955)

IDE

- When compiling to multiple executables from the command line, you must specify the -logger parameter to enable the correct log to be output to the console.

2848855 (1102932)

- For a file added to a JVM COBOL project by adding the folder that stores the file (using the **Source** tab on the **Micro Focus > JVM Build Path > Source page** in the projects' properties), there is now a new context-menu command, **Copy to Output Directory**, in COBOL Explorer. Choosing this command on a file triggers a build of the project and copies the file into the output directory.

2699374 (1094326)

JCL Support

- The return code from a JCL printer exit will now be checked. If it is non-zero, the dataset that was sent to the printer exit will be moved to the OUT-HOLD queue rather than to the PRINTED queue. To release the dataset, click "Release" in the ESMAC view - this will also present the dataset again to the printer exit.

2834206 (1101079)

- Under Enterprise Server, flushing an active job previously resulted in all associated spool records for the job being deleted, but left the job running. A subsequent attempt to kill the running job using casout/seeout failed due to the job spool record no longer existing. This change causes the flush request to fail if the job is active. You must kill the active job first before attempting to flush it.
- The catalog search facility has been enhanced to provide a quicker return when using wildcards in the dataset name selection.

2698700 (1094266)

- REXX can now be executed in VSE via IKJEFT01 and IRXJCL using in-stream data.

2831984 (1100883)

Run-Time System

- The Audit Manager contains a new TIMEOUT option. When a client sends an audit event using the 'CBL_AUDIT_EVENT' API, the event gets placed in the next available slot in a shared memory block. If shared memory is full (i.e. no slots are available), the event is re-tried until a slot becomes available.

If no Audit Manager is running, no events are removed from shared memory, and no slots will ever become available. Therefore, use the new TIMEOUT option so that a client will only retry sending until the TIMEOUT duration is reached; after which, it will stop sending audit events. If Audit Manager is recycled, events will start to be sent again.

To set the TIMEOUT for all Audit Manager clients, specify the following line in the Audit Manager configuration file:

```
mfaudit.timeout = n
```

Where n is the timeout value in milliseconds.

To set the TIMEOUT for an individual Audit Manager client, use the 'CBL_AUDIT_CONFIG_PROPERTY_SET' API. It takes an integer property-value, which should be the timeout value in milliseconds.

If TIMEOUT is set using both methods, the client property TIMEOUT takes precedence, unless this property is set to zero; in such cases, the TIMEOUT in the configuration file is used. If you use the 'CBL_AUDIT_CONFIG_PROPERTY_GET' API on the 'TIMEOUT' property, it only returns the TIMEOUT value for the client property; it does not return the value set in the configuration file.

2838689 (1101685)

- Several changes have been made to the implementation of IS DBCS, IS KANJI and IS JAPANESE class condition tests:

- IS [NOT] DBCS

When CHARSET"EBCDIC" is in effect, the IS DBCS test returns true when each character in the string is deemed to be a valid DBCS character. A valid character has its first byte in the range 0x41 through 0xFE, and the second byte in the range 0x41 through 0xFE, or the character is an EBCDIC space (0x4040). When CHARSET"ASCII" is in effect, the DBCS test uses an OS call to determine if the string contains only valid double-byte character, and returns true if valid.

- IS [NOT] KANJI

When CHARSET"EBCDIC" is in effect, the IS KANJI test returns true when each character in the string is deemed to be a valid Kanji character. A valid character has its first byte in the range 0x41 through 0x7F, and the second byte in the range 0x41 through 0xFE, or the character is an EBCDIC space (0x4040). When CHARSET"ASCII" is in effect, the IS KANJI test uses an OS call to determine if the string contains only valid Kanji character, and returns true if valid.

- IS [NOT] JAPANESE

When CHARSET"EBCDIC" is in effect, the IS JAPANESE test is not supported, and will generate a COBCH1806 Feature not supported in selected charset message on compilation.

When CHARSET"ASCII" is in effect, the IS JAPANESE test returns true when the string contains only double-byte Japanese characters or single-byte Japanese Katakana characters, and returns true if valid. When NSYMBOL"NATIONAL" is in effect, these class tests are not supported, and will generate a COBCH0303 Operand has wrong data-type message on compilation.

2812895 (1098401)

SQL: OpenESQL

- The DB2 CONCAT function and operator now convert to SQL Server using the HCOSS-supplied dbo.CONCAT for character, numeric and datetime data. If you are using BINARY or VARBINARY data,

you must apply the HCOSS-supplied `dbo.CONCAT_BINARY` function. HCOSS applications deployed with earlier versions of Enterprise Developer are affected, if they use string or binary concatenation. The mainframe dialect `DB2 ||` operator and `CONCAT` function now call a new SQL Server scalar function `dbo.CONCAT()`. All existing programs with `dialect=mainframe` that use DB2 concatenation syntax should be recompiled. All existing SQL Server databases that are accessed by these programs must have `dbo.CONCAT` installed. To create the new function in your application's SQL Server database, you can either:

- Run a DSN bind against the customer database. Or:
- Execute the `%ALLUSERSPROFILE%\Micro Focus\Enterprise Developer\hross\InstallDigitsFunction.sql` script.

This is a one-time only change to the database.

2843818 (1102248)

SQL Option for DB2

- Spurious errors were sometimes returned while querying using an ALIAS.

2830383 (1100609)

Significant Changes in Enterprise Developer 2.3

Enterprise Developer version 2.3 includes significant changes in the following areas:

- [AMB](#)
- [Assembler Support](#)
- [CAS Security](#)
- [CAS XA Switch modules](#)
- [Compiler](#)
- [Data Tools](#)
- [Discontinued Support for cwgui](#)
- [File Handling - External File Handler](#)
- [File Locking](#)
- [IDE](#)
- [J2EE Connector](#)
- [JCL Support](#)
- [Mainframe Access](#)
- [MF Server Administrator \(GUI\)](#)
- [PL/I Support](#)
- [SQL - HCO for SQL Server](#)
- [Updated Run-Time System](#)

AMB

- Loading PSBs sometimes caused an error that prevented the building of DDI symbols required for generation.

2813119 (1098471)

Assembler Support

- Previously, the Assembler run-time error SOC4 (Invalid 370 address abend) was mapped to the COBOL run-time error 200 (Internal logic error) which was misleading. It is now mapped to the COBOL run-time error 205 (COBRT205 Invalid mainframe pointer value (Fatal)). Note that, in most cases, you receive an Assembler SOC4 abend if the calling COBOL program was compiled without the `AMODE"24"` or the `AMODE"31"` Compiler directives.

2806239 (1097738)

CAS Security

- The Enterprise Server External Security Facility now includes MLDAP ESM Module 2.0, with a new algorithm for identifying the best-matching resource-access rule and ACE for resource-access security checks. This algorithm is faster and matches most customers' expectations. The new algorithm also provides an optional "username substitution" feature. It can be enabled by setting "rule substitutions" to "yes" in the [Operation] section in the Security Manager configuration text area. When this is enabled, the string "\${user}" in a resource-rule name will be replaced with the name of the user that makes the request. For example, a DATASET rule named "USERS.\${user}.*" would apply to datasets with the requesting user's name as the second qualifier. In rare cases, customers with complex, ambiguous resource-access security rules might see experience changes in behavior as a result of the new algorithm. The old algorithm is still supported and can be enabled by setting "version 1 authentication" to "yes" in the [Operation] section of the Security Manager configuration.

2807531 (1097783)

CAS XA Switch modules

- The XA switch modules now support dynamic registration.

2682101 (1092325)

- The XA switch modules now support batch-only operations when multiple XA Resource Managers have been defined.

2664675 (1091082)

- In Enterprise Developer 2.2 update 2, Micro Focus identified undefined run-time behavior when the following combination of directives was specified: SIGN"EBCDIC", CHARSET"ASCII", and one of the following: HOST-NUMMOVE, HOST-NUMCOMPARE or SIGN-FIXUP. Previously (Enterprise Developer 2.2 update 1 and earlier), if this combination was specified, the SIGN"EBCDIC" directive should have been ignored, to avoid a mixture of ASCII and EBCDIC characters; however, SIGN"EBCDIC" was still being honored, resulting in undefined run-time behavior. Therefore, this combination of directives is now invalid for Enterprise Developer 2.2 update 2 or later, and if specified, will be rejected at compile time.

2786397 (1095265)

Compiler

- For native COBOL, the size limit of the Data Division now stands at 2GB -1.

2796076 (1096384)

- COBDATA has no effect on compilation. The output of the Compiler is the same location regardless of whether COBDATA is set.

Previously, it was not possible to specify sign(EBCDIC) with sign-fixup, host-num-move or with host-num-compare. This combination is now supported in native COBOL but remains invalid for managed COBOL code. This is applicable to version 2.2 U2 HotFix 10 onwards.

2824577 (1100823)

Data Tools

- DFCONV now returns the correct return-code; previously, it would always return 0.

Discontinued support for cwgui

The cwgui debugger is no longer available. You can now use the cw_java debugger which provides a richer graphical debugging experience.

File Handling - External File Handler

- Custom file handlers (using DYNREDIR) are now called for each part of a concatenated file.
2795077 (1096322)

File Locking

- In versions prior to Enterprise Developer 2.3, the semantics of the sharing phrase specified in an OPEN statement or used within a call to CBL_OPEN_FILE were not correctly applied in some cases on UNIX and Linux platforms. From version 2.3 onwards, the sharing phrase is correctly honored when the tunable `strict_file_locking=true` is set, which is the default setting.

Example of potential changes in behavior:

- *Process-A* opens a file with read-only access and a sharing mode that denies other processes write access (SHARING WITH READ ONLY).
- *Process-B* then attempts to open the file with read-only access and a sharing mode that denies other processes read access (SHARING WITH NO OTHER).

With `strict_file_locking=true`, *Process-B* is unable to open the file, because *Process-A* has successfully opened the file allowing only read access.

With `strict_file_locking=false`, *Process-B* successfully opens the file.

If your application encounters unexpected OPEN conditions or fails to open files, it might be as a result of the new file locking behavior. In such circumstances, we recommend that you review the file locking and sharing requirements of your application and refactor your source code to work with the default setting. The original file locking and sharing behavior can be restored by setting `strict_file_locking=false`.

IDE

- Enterprise Developer for Eclipse now ships with Eclipse 4.4.2 (Luna). If your applications contain JVM COBOL code that was built with a previous version of the product, those parts of your application must be rebuilt; otherwise you will experience errors at either compilation or run time.
(609469)
- A project can have one of two connection modes: NFS/Samba, where the target location is mounted as a local drive, and RSE, which is a purely remote connection to the target location. For project types such as Mainframe COBOL, NFS/Samba is required to use some tools which do not support RSE. For most other projects, it is possible to switch between connection modes using the **Remote Settings** context menu option. In the dialog box, there are radio buttons allowing for selection of connection modes. On changing mode, you must select an existing connection of the appropriate type, or create a new one. If switching to NFS/Samba mode, you must specify the local path to the project.
2792882 (1096196)
- By setting "-Denable.projectrepair=true" in the eclipse.ini file, .cobolProj and .pliProj files will be repaired to reflect the workspace on project refresh.
2696707 (1095994)

J2EE Connector

- Enterprise Developer version 2.3 provided a new command-line argument to Java, `mf.ssl.algorithm`, which can be set to an appropriate algorithm.
2799213 (1096684)

JCL Support

- When allocating a dataset to a step using MVSCTLBP, the values returned for the dataset attributes are those defined in the catalog. If these are unset in the catalog, then the values defined in the MVSCTLBP call are returned.
2816179 (1098882)
- The program attributes for the SSTM job have been changed to default to a mainframe dialect, AMODE 31 program. These attributes will be used when constructing the MVS Control Blocks for the job, which means that the pointers to the sub-block areas will be in mainframe format rather than native format.
2815695 (1098825)
- Any OUTPUT statement information passed to the printer exit now includes the correct class of that statement.
2814249 (1098636)
- In ESMAC, when deleting the Spool, the physical spool files and the entries in casspool.dat are deleted.
2812899 (1098412)
- New functionality has been added to the printer exit interface that enables you to delete the printer spool file after a file has been printed. Set (ws)-prn-file-disp, which is defined in cascbprn.cpy, to 1 to delete the spool file after printing. Additionally, the environment variable ES_JESYSMSG_OUTPUT=Y now correctly routes spool files to the Output queue. Previously, ESMAC still showed the job on the HELD display, even when there were no spool files HELD.
2812899 (1099966)
- A new environment variable has been introduced - ES_JESYSMSG_RESTRICTED. JESYSMSG spool files are normally routed to HELD. This can be overridden by setting the environment variable ES_JESYSMSG_OUTPUT=Y, which causes all JESYSMSG spool files to be routed to OUTPUT. If ES_JESYSMSG_OUTPUT=Y is set but you require specific classes to be held, set the environment variable ES_JESYSMSG_RESTRICTED to a string containing the classes to be HELD.
2812878 (1098417)
- The IEBDG PICTURE value length must match the length specified. The value can now contain any characters including quotes as on the mainframe.
2794881 (1096221)
- Changes to the IDCAMS ALTER processing prohibit applying MGMTCLASS changes to GDGs and to a VSAM element other than a Cluster. Shareoption settings in a SMS DATACLASS are now applied to the VSAM elements correctly.
2789939 (1095803)

Mainframe Access

Mainframe Access is now built using the IBM XL/C compiler rather than the SAS/C compiler that was used previously. This change requires you to update your production JCL procedures with reference to the supplied samples MFA, MFAS, and MFAAS. In each JCL procedure, change the STEPLIB DD statement and add a new one, CEEOPTS.

The web administration feature is no longer available in Enterprise Developer.

To use the Mainframe Access support for Endeavor you must be using Endeavor R14 or later. CA stabilized the Endeavor API at R14, so if you upgrade from R14 to a later version you will not need to update the support module that Mainframe Access uses.

MF Server Administrator (GUI)

- Passwords that entered through either the MFDS or the ESMAC interface now use the same encoding.
2792382 (1096011)

PL/I Support

- For BASED variables declared using the form BASED (ADDR(x)), storage checking is now performed on the base reference “x”. This behavior requires the compiler option `–nolaxbased`. See the Open-PLI User guide for more information on the options `–[no]laxbased` and `–[no]laxdefined`.

2819663 (1099293)

- The behavior of the `–zp1` option has been reverted to that of before Enterprise Developer 2.2 Update 1, with an additional correction relating to Char Varying data items. For a full description of the `–zp1` option, refer to the Open-PLI User’s Guide in your product help. If you are installing ED 2.2 U1 HF5 as an upgrade to ED 2.2 U1, you must rebuild any applications that are compiled using the `–zp1` option after the upgrade.

2789213 (1095636)

- Previously, when using any of the compiler options `-mvs`, `-dli`, `-ims`, or `-cics`, the BINARY attribute was incorrectly applied if the FIXED attribute is specified without BINARY or DECIMAL. The default attribute DECIMAL is now applied.
- The BIGENDIAN attribute, when explicitly applied to a structure, is now propagated to its CHAR VARYING and WIDECHAR VARYING members.
- Declarations using DEFINED storage are now checked for compatible types between the DEFINED variable and its base variable. This behavior requires the compiler option `–nolaxdefined`. See the Open-PLI User guide for more information on the options `–[no]laxbased` and `–[no]laxdefined`.
- The PL/I Macro preprocessor has been enhanced to support the use of the ANSWER statement from within a PL/I Macro.

2698537 (1094243)

- The macro preprocessor has been enhanced to support arrays of CHAR and FIXED variables. New macro builtin functions have been implemented (HBOUND, LBOUND, DIMENSION).

2696129 (1094011)

SQL: HCO for SQL Server

- The return type for the HCOSS DB2 CHAR() function has changed from CHAR(255) to VARCHAR(255).

2804069 (1097253)

Updated Run-Time System

- Enterprise Server now provides an execution environment capable of running applications that were each built using different development products. A consequence of this is that if your application has a main COBOL executable (.exe) that was built with a version of Enterprise Developer prior to version 2.3, you should ensure that the executable is rebuilt and packaged with the new run-time system. This does not affect mainframe subsystem projects deployed to Enterprise Server. You can rebuild from the IDE or the command line.

Other COBOL subprograms built with previous versions of Enterprise Developer are not required to be rebuilt.

Significant Changes in Enterprise Developer 2.2 Update 2

Enterprise Developer version 2.2 update 2 includes significant changes in the following areas:

- [Compiler](#)
- [Compiler Front-end](#)
- [Documentation](#)

- [Enterprise Server](#)
- [File Handling - External File Handler](#)
- [J2EE Connector](#)
- [MTO: CICS Communications](#)
- [MTO: IMS DB](#)
- [MTO: IMS MFS](#)
- [MTO: JCL MVS](#)
- [MTO: JCL Utils](#)
- [Open PL/I Compiler](#)
- [Open PL/I Debugger](#)

Compiler

- When using the HOSTRW directive with the mainframe dialect, Report Writer will now produce the full range of ASA control characters and will emulate mainframe print files.

2697615 (1094527)

Compiler Front-end

- Fixed Binary (p<=7) is now an 8-bit, signed, 2's complement binary integer by default.

Documentation

- The default setting for the MFALLOC_PCFILE environment variable has changed; the default is now set to Y, which means that when cataloguing a file that has a DCB attribute of DSORG=PS, a physical file is created for it if one does not exist. Previously, the default was set to N, which meant that a file was not created.

2697571 (1094370)

Enterprise Server

Starting with this release, IMSCONFIG.DAT is no longer used for configuring enterprise servers. Instead, you need to use environment variables to control the following fields:

Field	Environment Variable
LANG=PL/I PCB address lists	ES_IMS_PLI_INDIRECT_PCBADDR=D Y N Where: D Dynamic (default) use indirect PCB address list when PSB language is PL/I and main program is PL/I. Y Always use indirect PCB address list if PSB language is PL/I. N Never use indirect PCB address list.
IBM Platform	ES_IMS_IBMPLATFORM=M D Where:

Field	Environment Variable
Secondary Index Sparse exit language	M MVS (default)
	D DOSVS Required by some DOS/VS customers.
	ES_IMS_SPARSE_EXIT_LANG=C A Where:
	C COBOL
	A Assembler
	Not set (default) Sparse exits disabled

File Handling - External File Handler

- The ES_IMS_TLOG_FLUSH environment variable is now deprecated. To control TLOG flushing, use the following environment variable: ES_IMS_DB_TLOG_WRITETHRU=0|1

0 Forces the flushing of TLOG buffers to disk on COMMIT only (default)

1 Forces the flushing of TLOG buffers to disk on all database I/O

To control database flushing, use the following environment variable: ES_IMS_DB_COMMIT_FLUSH=0|1

0 Forces the flushing of database buffers to disk on CLOSE only (default)

1 Forces the flushing of database buffers to disk on COMMIT only

Depending on the number of database updates, the frequency of COMMITs, and other concurrent computer activities, use of these environment variables could cause significant performance degradation.

2784949 (1095190)

- The use of the environment setting MFJ_INPUTDS_ERROR=N has been extended so that an input file for a JCL step is now considered as optional and you no longer receive an error when the file is missing.

2784622 (1095971)

J2EE Connector

- The listSystem.properties file in package com.ibm.ctg.client was missing documentation for some sections.

(606556)

MTO: CICS Communications

- In CRTE mode, the transactions are now run using the correct userid if the user has signed on after running CRTE.

2663890 (1091979)

MTO: IMS DB

- The NODCX mfims dbdgen directive has been added to enable you to compile DBD source without executing data capture exit routines contained in the DBD source. Previously, data capture exit routines contained in the DBD source but that were not found during compilation were ignored. The new default behavior is to process all data capture exit routines unless NODCX has been specified.

2579600 (1084675)

MTO: IMS MFS

- All existing IMS Global Physical Terminal edit routines (DFSGPIX0) in use must be recompiled with charset EBCDIC instead of charset ASCII.

(606142)

MTO: JCL MVS

- Following the detection of an unrecoverable file status error on the casspool file, a message is displayed on the console and an ACCEPT statement is issued. This stops processing, allowing you to fix the underlying file problem, before continuing the job.


2651654 (1090287)

MTO: JCL Utils

- You can now activate support for the VSAM Shareoption for batch jobs and for CICS files that use a catalog entry. For more details, see your product help.

2660651 (1091290)

Open PL/I Compiler

-  **Important:** If you are installing this release as an upgrade to Enterprise Developer 2.2 Update 1, after the upgrade you must rebuild any applications that are compiled using the `-zp1` option.

The behavior of the `-zp1` option has been reverted to that of versions of Enterprise Developer earlier than 2.2 Update 1, with an additional correction relating to Character Varying data items. For a full description of the `-zp1` option, refer to the Open PL/I User's Guide in the product help

This fix restores the behavior in Enterprise Developer versions earlier than 2.2 where, when compiling with `-zp1`, all parameters are treated as unaligned. (In Enterprise Developer 2.2 Update 1, the behavior when compiling with `-zp1` was to not treat parameters as if unaligned).

When using the `-zp1` compiler option, all Character Varying data items are now treated as if unaligned. In previous versions of Open PL/I, for Character Varying data items, the `-zp1` unaligned requirement was applied only to structure members and parameters.

To illustrate the change, consider the following example:

```
zptest: proc options(main);  
  
    dcl 1 st1,  
        2 c char,  
        2 x(4) char(7) var init ('a', 'xx', 'yyy', 'zzzz');  
  
    dcl y(4) char(7) var init ('a', 'xx', 'yyy', 'zzzz');  
  
    dcl sub entry ((4) char(7) var);  
  
    call sub (x);  
  
    call sub (y);
```

```

end;

sub: proc (z);

    dcl z(4) char(7) var;

    dcl i fixed bin(31);

    do i = 1 to hbound(z);
        z(i) = 'x';
    end;

end;

```

Where:

- For `x` and `z`, each `char (7) var` item is 7 plus 2 bytes which equals 9 and then multiplied by 4 equals 36.
- If `y` were aligned on half-word by default, each array element is half-word aligned and each equals 10 bytes (9 + 1 pad byte), and the total size equals 40 bytes.
- At `call sub (x)`, the calling argument and parameter are matched.
- At the `call sub (y)`, the `y` element size (10 bytes) is mismatched against the parameter `z` element size (9 bytes) due to `-zp1`. This is incorrect and causes unexpected program behavior.

Due to this correction of treating all Char Varying data items as if unaligned when using `-zp1`, the size of CHAR VARYING arrays now differs from previous versions of Open-PL/I. For example:

```

dcl X(4) char(7) var;

    Put skip list (size(X)) /* size is 36 bytes vs. 40 bytes in previous
versions of Open-PL1 */

```

2789213 (1095636)

Open PL/I Debugger

- Previously, the SRC list in the debugger only included the source filename without the path to the file. If a source file did not exist in the current directory, the debugger could not find it using the filename as it always expects a fully qualified filename. The SRC list now contains the fully qualified file name which includes the path. Also, when changing the SRC or ENV values the source is updated correctly.

2783734 (1094988)

Significant Changes in Enterprise Developer 2.2 Update 1

Enterprise Developer version 2.2 update 1 includes significant changes in the following areas:

- [MTO - JCL ESMAC](#)
- [MTO - JCL MVS](#)
- [MTO - JCL System Catalog](#)
- [MTO - JCL Utils - IDCAMS](#)
- [Open PL/I Compiler](#)
- [Open PL/I Debugger](#)
- [Open PL/I RTS](#)
- [SQL: COBSQL](#)

MTO - JCL ESMAC

- As part of an enhancement to improve performance of the spool display function, several new indices have been added to the `casspool.dat` file. If you are upgrading from a previous release, you must convert the `casspool.dat` file of any enterprise server created before Enterprise Developer 2.2 Update 1 to the new format before starting any JES-enabled enterprise server and submitting jobs to it.

Enterprise Developer version 2.2 update 1 provided the following scripts that convert the `casspool.dat` file to the new format:

- On Windows: `fixcasspool.bat` in `%COBDIR%\bin` or `%COBDIR%\bin64`
- On UNIX/Linux: `fixcasspool.sh` in `$COBDIR/bin`

To convert the `casspool.dat` file, run the appropriate script from an Enterprise command prompt using the following syntax:

- On Windows:
`fixcasspool.bat <old-location>casspool.dat <new-location><backup_filename>`
- On UNIX/Linux:
`fixcasspool.sh <old-location>casspool.dat <new-location><backup_filename>`

Where:

- `<old-location>` - the fully or partially qualified location of the existing `casspool.dat` file, if not in the current directory. Optional.
- `casspool.dat` - the spool file that will be updated. The file is located in the same directory as the enterprise server's system catalog.
- `new-location` - the fully or partially qualified location where the backup file will be created, if not in the current directory. Optional.
- `backup_filename` - the name of the file that will be a back-up of the old `casspool.dat` file. If the file name already exists, the script terminates without converting the `casspool.dat` file.

If you do not convert `casspool.dat`, attempts to start the enterprise server fail with the following messages:

```
JES000011E Batch Spool File Error; Status [39]
      JES000130E Note JES000011 Error - New casspool indexes may need
generating. See error
      description in product docs
```

2654417 (1090421)

MTO - JCL MVS

- The final disposition processing of temporary datasets has been corrected to be in line with the mainframe processing, with a default value of PASS unless DELETE has been specified in the DD card. Previously, the default value was DELETE.

2671175 (1091627)

- When running batch programs, receiving an "Out of Space" error requires operator intervention to continue processing.

2642115 (1089055)

MTO - JCL System Catalog

- You can now use MGMTCLAS to provide an EXPIRE AFTER number of days. This is used to calculate an expiry date for datasets allocated using the MGMTCLAS parameter.

2664639 (1090992)

- MFLEX*.DAT files were not being moved with spool files. Spool housekeeping now deletes LEX files when it deletes JESYSMSG. Also, to identify any remaining orphan LEX files, the MF_SPOOL_HK_LEX_SCAN environment variable sets spool housekeeping to scan the data directory for LEX files older than the max retain period.

2659555 (1090759)

- Spool housekeeping has been rewritten to provide a number of new features. These will be described in the product documentation.

(592452)

MTO - JCL Utils - IDCAMS

- The state of VSAM virgin files is now being correctly processed.

2686149 (1093587)

- A REPRO COUNT of zero is now processed as zero records to be included in the REPRO operation.

2665006 (1091034)

Open PL/I Compiler

- On Intel platforms, the `-bitsltr` Compiler option can be used to store bit strings left-to-right within each byte. This is similar to Big Endian bit ordering on Intel platforms. The Intel default is to store bit strings right-to-left within each byte similar to the Intel hardware bit ordering.

2675860 (1092098)

- The libraries `libmf.so`, `libespli.so` and `libespliz.so` are no longer delivered. This is transparent if linking PL/I objects using `mfplx` or `ldpli`, and no changes are needed to your build process.

However, if linking PL/I objects using either the UNIX `ld` command or the Enterprise Developer `cob` command, you must include the libraries `libmfpli.so` when linking stand-alone Open PL/I objects (executed outside of Enterprise Server), or `libmfpliz.so` when linking PL/I objects executed under the control of Enterprise Server.

To do this, use the option `-lmfpli` for stand-alone Open PL/I objects or `-lmfpliz`, respectively, for PL/I executed under Enterprise Server. These options replace the `-lmf` option.

- Open PL/I applications must be recompiled and re-linked in order to use them with this release.

Open PL/I Debugger

- The debugger now stops before an ON UNIT triggers. This lets you perform debugging on the ON UNIT itself. You can configure the ON UNITS the debugger stops on.

2674783 (1092319)

Open PL/I RTS

- Writing to `SYSPRINT` when running outside of Enterprise Server now drives output directly to the terminal and does not buffer the I/O.

2477031 (1076203)

SQL: COBSQL

- COBSQL now displays appropriate COBOL syntax errors after encountering EXEC SQL statement errors.

2673619 (1093197)

Significant changes in Enterprise Developer 2.2

Enterprise Developer version 2.2 included significant changes in the following areas:

- [ASM Compiler: HLL Macro Assembler](#)
- [CAS \(COBOL App Server\) General](#)
- [CCI Session Layer Code](#)
- [Compiler](#)
- [Eclipse IDE](#)
- [File Handling - Sort / JCL Sort](#)
- [Interface Mapping Toolkit](#)
- [JCL System Catalog - Dataset Allocation](#)
- [MF Directory Server](#)
- [MF Server Administrator \(GUI\)](#)
- [MFBSI](#)
- [MTO - IMS Other](#)
- [MTO - JCL MVS](#)
- [MTO - JCL System Catalog](#)
- [Open PL/I Compiler](#)
- [Open PL/I Debugger](#)
- [Open PL/I RTS](#)
- [Request Handler](#)

ASM Compiler: HLL Macro Assembler

- A "." (dot) specification is no longer ignored in the SYSIN environment variable or in the mf370ctl.cfg file when used to specify an output directory, and indicates the current directory. This is a change in behavior from Mainframe Express, where the "." (dot) character in the SYSIN environment variable or in the mf370ctl.cfg file is still ignored and assembler output files are placed in the directory specified by the SYSIN environment variable.
- Assembler Build Output will now be sent to the location specified in the COBOL output path project setting. Previously, all Assembler build output went to the project directory.

CAS (COBOL App Server) General

- Previously, on cancelling an active JCL job on UNIX (after you have used STOP on the ESMAC > SEPs page to stop the INITIATOR that it is running in), the INITIATOR was terminated and a new one was restarted in its place. Now, the new INITIATOR will not be started, as the STOP request was received before the CANCEL.

2559087 (1084226)

CCI Session Layer Code

- A new option, use_global_namespace, is available for the cci.ini file in the Windows %SystemRoot% folder. If use_global_namespace is set, all the ccishared memory objects are created in a system-wide address space, and the applications hosted by different users, including system services, can communicate. To use this facility, edit the cci.ini file and ensure use_global_namespace is set to "yes".
[ccismem-base] # Allow interaction between users on a # single system. Using this option reduces security as # all users will have access to the same name space. use_global_namespace=yes If the value of this option is anything other than "yes", or if the option is missing, no change is made to the existing behavior.

2195519 (1062800)

Compiler

- The default for the NSYMBOL directive under DIALECT(ENTCOBOL) has been changed to NSYMBOL(NATIONAL) to emulate the equivalent IBM default.
2657471 (1090355)
- To improve RM/COBOL and ACUCOBOL compatibility, the SIGN clause at a group level is no longer applied to non-DISPLAY usage signed numeric data items within the group, just as it is not applied to unsigned numeric data items and non-numeric data items within the group.
2549904 (1082171)
- Previously, even though no code was generated, the Compiler allowed the ON EXCEPTION and NOT ON EXCEPTION phrases in the DISPLAY statement in formats that do not allow these phrases. As a result, if the DISPLAY statement was in the ON EXCEPTION phrase of another statement, the NOT ON EXCEPTION phrase would bind incorrectly with the DISPLAY statement instead of with the intended containing statement - for example, ACCEPT or CALL.

Eclipse IDE

- The Problems view now has a 'Program' column that displays the name of the program in which the problem occurred. If the column is not present in the view by default, it can be added through the Configure Columns dialog box in the View Menu.
2608496 (1088530)
- The outline of the ruler in the COBOL editor now changes on modelChanged depending on the current source format.
2488419 (1077143)
- Deleting a Web service or a Java interface mapping now deletes the files relating to it from the repos directory and, if it is empty after the files have been deleted, will remove the repos directory itself.
- The IDE now performs an automatic directive determination when files are added to a project. You can also use a command to perform directive determination of the project manually.

File Handling - Sort / JCL Sort

- When used with MFJSSTRICTSORT=ON, MFJSORT now errors (return code 16) if the input RECFM is different to a SORTOUT RECFM. It also errors if the input RECFM is different to an OUTFIL RECFM when FTOV or VTOF/CONVERT has not been specified.
2643384 (1088993)

Interface Mapping Toolkit

- For program-based Service Interfaces, if the program-id name in the COBOL source is in lowercase and is not surrounded by quotes, its corresponding entry-point name is now forced to uppercase when used in a Service Interface Operation. Previously, the case was preserved. As a result of this change, existing Service Interfaces will become invalidated if you refresh their program's annotations because of the new spelling of the entry-point name. To avoid this, you need to surround the program-id name in the COBOL source with quotes before you refresh the annotations.
- The default for previous releases was to create a catalog entry with no physical file until data was actually written to the catalog. Now, a PC file is allocated for non-VSAM files by default. You can emulate the previous behavior if you set the environment variable MFALLOC_PCFILE to N.

MF Directory Server

- The mfd command line option for exporting registered Enterprise Server definitions to an XML file now supports the "*" option. This exports all registered servers rather than a specified server. Multiple server definitions are now exported into the target directory and saved into a file with the default name

ALLSERVERS.xml. The import option now also supports the import of multiple server definitions from a single XML file.

2641890 (1088838)

- mdump now supports a new option, -e, to help you query the Security Manager configuration details. The possible values of the option are: "1" - shows security configuration that applies to any returned enterprise servers; "2" - shows security configuration for MFDS and the default Enterprise Server security configuration. This requires MFDS version 1.15.00 or higher; "3" - returns the properties of all configured external Security Managers.

2487164 (1081693)

MF Server Administrator (GUI)

- When adding a user to an external security manager, you can now include a password expiry time in the Advanced Configuration section of the Add New User wizard in Enterprise Server Administration. The field value is specified using generalized time format (YYYYMMDDHHMMSS.0Z), and can be used by the MLDAP ESM for calculating whether a user's password has expired and requires updating. This value may only be specified using this page when adding a user. You need to use an external directory services configuration tool to edit it.

2562118 (1083203)

MFBSI

- casout now correctly sets return-code to the job status for the sysout retrieval when the file is less than 65K. Also, MFBSI now considers return codes of 0, 7 or 8 to be successful for JESYSMSG retrieval where, previously, it only considered 0 as successful.

2655482 (1090444)

MTO - IMS Other

- The DBDCPY, MFSCPY, and PSBCPY environment variables are no longer required and have been deprecated. Use the IMSCPY environment variable to specify copybook locations.

MTO - JCL MVS

- Empty values for the USER or NOTIFY parameters on the JCL JOB card no longer cause an error when parsing the JCL.

2646770 (1089380)

- The disposition processing of PASSEd datasets has been revised to be consistent with mainframe processing. Particularly, received datasets are removed at the end-of-step processing, unless they are to be passed on to another step in the job. Any passed datasets that were not received at the end of the job may be removed if they conform to the mainframe conditions for this.

2610362 (1087601)

MTO - JCL System Catalog

- The value for the JCL DD UNIT=device parameter is now stored in the system catalog entry for the dataset.

2619050 (1089095)

Open PL/I Compiler

- Open PL/I applications must be recompiled and re-linked in order to use them with this release.

2675860 (1092098)

- If using the `-cics`, `-ims`, or `-mvs` options when compiling on a little endian platform, the possibility for errors existed. Starting with this release, using the `-cics`, `-ims`, or `-mvs` options automatically uses `-bigendian`, thereby eliminating the possibility for these errors. Evaluate any applications that were built with a prior release to determine any impact on any data files being used, or where programs compiled with the `-mvs` option pass data to sub-programs compiled without it. The `-bigendian` option can be suppressed by adding the `-native` option after the `-cics`, `-ims`, or `-mvs` options.

Open PL/I Debugger

- CodeWatch no longer attempts to use an incorrect directory when starting. Also, CodeWatch is now the default debugger when no other debugger is specified, and is the only debugger available when using `-optexec plitest`.

2655906 (1090193)

- Support for the evaluation of X, NX, Z, A, and E constants has been added to CodeWatch.

Open PL/I RTS

- A problem with possible truncation or trailing zeroes within PICTURE assignments has been fixed. As the fix depends on changes to both the Compiler and the run-time, you must perform a complete rebuild of all your source code (recompile and relink) in order to apply it.

2662261 (1090768)

- Previously, when taking a PLIDUMP, if the .PDB file needed for a PLIDUMP was not in the same directory as the .DLL and the .STB files, and they were not in the first CAS_BATCH_PATH subdirectory, then the PLIDUMP was not successfully generated and an S0C4 abend could occur.

2615412 (1090483)

- The Open PL/I debugger no longer loses its place and behaves erratically when debugging code containing the same PL/I macro multiple times within the same source routine, even if the `-nodebuginfo` and `-pp` options are not specified.

2612530 (1087809)

Request Handler

- A problem that caused BIS to create log files in a directory named `C:\ProgramData\AcuCorp\BIS\LogFiles` was fixed. BIS no longer creates log files unless specified and the BIS logging service is now disabled by default. To enable it, you need to use the following global environment variable:
`BIS_LOG=[OFF | ON | <directory>]` Where the values are:

- `OFF` - disables logging (the same as if `BIS_LOG` is not specified or is left blank)
- `ON` - enables logging and directs the log files into the default location, which must not be read-only.
- `<directory>` - enables logging and directs the log files into the specified directory. The user must ensure that the BIS request handler has write rights for this directory. The directory must be an absolute path or network path. If the specified directory does not exist, BIS will attempt to create it. The containing directory must exist.

The `BIS_LOG` variable is only examined when the BIS application pool is started or recycled. After setting or changing `BIS_LOG`, IIS must be restarted in order for the variable to take effect.

Significant Changes in Enterprise Developer 2.1 Update 1

Enterprise Developer version 2.1 update 1 includes significant changes in the following areas:

- [Documentation](#)
- [IDE](#)

- [JCL Support](#)

Documentation

- To ensure no loss of functionality when accessing Vision and RM/COBOL data files, you should use the appropriate IDXFORMAT Compiler directive setting or file handling option, and not use the CALLFH(ACUFH) Compiler directive. See 'Configuring Access to Vision Files' and 'Configuring Access to RM/COBOL Data Files' for more information.

IDE

- You can now set program breakpoints as follows - click "Add Program Breakpoint" in the Run menu or on the Breakpoints view toolbar, and enter the program name in the dialog that is displayed.
- Local Enterprise Server regions without secure user credentials are now started with casstart and stopped with casstop commands.

JCL Support

- Input files are no longer optional so you can receive errors if you catalog a file without creating the physical file. To help avoid failures of JES job steps because of a missing input file, you can use one or both of the following environment variables: 1) MFALLOC_PCFILE=Y - results in the creation of a catalog file if DSORG=PS is set in the DCB parameters. 2) MFJ_INPUTDS_ERROR=N - results in IEBDG, IEBCOMPR and IEBCOMPR utilities to treat missing input files as though they were present but empty files.

Significant Changes in Enterprise Developer 2.1

Enterprise Developer version 2.1 includes significant changes in the following areas:

- [CICS Support](#)
- [IDE](#)
- [Run-Time System](#)
- [Vision File System](#)

CICS Support

- If external security is configured (MLDAP) and the default ESMAC user is not allowed to login, the region does not startup.

2550800 (1082337)

IDE

- You can now specify in the IDE whether directories that you add to the project should be added to the copypath. You configure this in the project's properties under Micro Focus COBOL > Build Paths, and on the Copybook Paths tab.

(589096)

Run-Time System

- When running a full-screen application inside a terminal emulator on Linux, the actual size of the terminal is read at startup and reread when the terminal is resized. This behaviour is also supported on AIX, HP/UX, and Solaris. The Micro Focus vt220 terminfo entry now correctly describes a 24-line display. A vt220-25 terminfo entry is included for compatibility with the previous behaviour.

2579335 (1084817)

Vision File System

- When you configure your application to return RM/COBOL file status codes, by setting COBFSTATCONV=rmstat, the codes returned are ANSI'85 codes.

2553438 (1082469)

Significant Changes in Enterprise Developer 2.0

Enterprise Developer version 2.0 includes significant changes in the following areas:

- [Compiler](#)
- [HCO for DB2 LUW](#)
- [File Handling](#)
- [IDE](#)
- [Run-Time System](#)

Compiler

- The scope of the ILUSING Compiler directive when used in a \$set command has changed. The scope of the directive is now limited only to the source file it is set in, and not globally. This new behavior may mean that your source files no longer compile. To resolve this, add the required ILUSING statements to the required individual source files, or add the ILUSING directive on the command line. Alternatively, use the IDE to achieve the required behavior: in Visual Studio, use the Namespaces tab; in Eclipse, set the directive in the Additional Directives field.

HCO for DB2 LUW

- The DB2 ECM has been updated to resolve run-time errors returned when compiling against mainframe databases in 64-bit mode.

2549058 (1082441)

File Handling

- When reading a file cataloged as DISP=SHR the file handler now buffers the read for better performance.

2518330 (1079491)

- Using SKIP=nL in the SECTIONS section of SORT will now cause the following behavior: 1) Under a mainframe sort, (n/3 rounded up) blank lines (when n>0) will be skipped and the appropriate carriage control character will be placed in column 1 of each blank line. 2) For sequential file organizations, the same behavior will occur but no carriage-control characters will be printed. 3) For line sequential files, n blank lines will be skipped.

2525670 (1080092)

IDE

- Typing in the sequence area moves text after the cursor to the right and doesn't delete the characters at area end. To overwrite characters in the area, switch to insert mode by pressing the Insert key. Pressing the Tab key in the sequence area indents the text if the option "Pressing Tab in the sequence number area moves caret to area A" is not selected. Pressing the Tab key when there is a multiline selection aligns the text to the area margins.
- Typing in the sequence area now works like typing in a regular text editor - it moves to the right all the text after the cursor and doesn't delete the characters at area end. If you want to overwrite characters in the area, you can switch insert mode to overwrite by using "Insert" key.

Run-Time System

- On Windows 7, building 64-bit native COBOL applications always rebuilt the entire project. This was caused by the Microsoft FileTracker tool, used by the COBOL projects to track dependencies, not working because of Windows 7 security updates. This product now includes a fix to work around this issue.

Unsupported or Deprecated Functionality

The following topics describe functionality that was removed or deprecated at each product release.

Unsupported or Deprecated at Enterprise Developer 4.0

The following features or functionality are no longer supported or are deprecated at version 4.0:

- Support for Eclipse 4.4 and 4.5 has been discontinued.

Unsupported or Deprecated at Enterprise Developer 3.0

The following features or functionality are no longer supported or are deprecated at version 3.0:

- The HOSTSIGNS Compiler directive is no longer supported. Micro Focus recommends that you use the following Compiler directives instead: SIGN-FIXUP, HOST-NUMMOVE, and HOST-NUMCOMPARE.
- Compatibility AddPack for Visual COBOL - this is now deprecated and will not be available with release 3.0 and later.

The Dialog System GUI and run-time components and Dialog System Character Mode (on Windows and UNIX) which were part of the AddPack are now installed as part of Enterprise Developer for Visual Studio. The run-time components are installed as part of Enterprise Server. These are only included for backward compatibility and Micro Focus does not recommend that you use them for new development.

The other components which were part of the AddPack, the Character-Based Data File Editor, CSBIND and Screens, will be available upon request from Micro Focus SupportLine.

- Audit Manager is deprecated and provided for backward compatibility only. We recommend that you use syslog events instead. See *Enterprise Server Auditing* for more information.
- The following DB2 environment variables are deprecated at version 3.0, and provided for backward compatibility only.
 - HCOBND - Micro Focus recommend you use either the BIND or the BINDDIR compiler directive option.
- The following compiler directives are deprecated at version 3.0, and provided for backward compatibility only.
 - CONVERTRET
 - IDYSRCPATH
 - ILOBJECTIFY
 - OPTION
 - SPZERO
 - TRICKLE
- The following file handling options are deprecated at version 3.0, and provided for backward compatibility only. Micro Focus recommend you use IDXFORMAT"8" instead:
 - STRIPING

- MAXSTRIPEDIGITS
- MAXSTRIPEFILES
- MAXSTRIPESIZE
- STRIPE-X
- STRIPENAMETYPE -

Unsupported or Deprecated at Enterprise Developer 2.3 Update 2

The following features or functionality are no longer supported or are deprecated at version 2.3 Update 2:

- The `command_line_linkage` tunable has been deprecated; equivalent functionality can be achieved by using the `COMMAND-LINE-LINKAGE` Compiler directive instead.

Unsupported or Deprecated at Enterprise Developer 2.3 Update 1

The following features or functionality are no longer supported or are deprecated at version 2.3 update 1:

Unsupported or Deprecated at Enterprise Developer 2010

The following Net Express features or functionality were not supported or were deprecated from the first version of Enterprise Developer:

These Net Express compiler directives are not supported by Enterprise Developer :

01SHUFFLE
64KPARA
64KSECT
AUXOPT
CHIP
COBIDY
DATALIT
EANIM
EDITOR



Note: The directive `SPZERO` was already deprecated from Net Express 5.0 onwards, and is provided for backward compatibility only. You should instead use `SIGN-FIXUP`.

and the pseudovariables of the following Net Express environment variables are obsolete and cannot be used:

BASENAME
FILENAME
PATH
TARGETDIR

Known Errors and Restrictions

Errors and restrictions in previous versions are not listed in this document; those that still apply in version 4.0 are included here, while any that have been fixed may occur in the *What was New* or *Significant Changes* sections.

Refer also to the *Known Issues and Restrictions* topic in the *Product Information* section of your product Help.

The following errors and restrictions are known to apply to Enterprise Developer version 4.0:

- [Application Workflow Manager](#)
- [CICS](#)
- [Commuter licenses](#)
- [Debugging](#)
- [Database Access](#)
- [Docker](#)
- [Enterprise Server](#)
- [imtkmake](#)
- [Installation](#)
- [Micro Focus Rumba](#)
- [OpenSSL](#)
- [PL/I Support](#)
- [Resource Adapters](#)
- [REST/JSON IMTK implementation and the same-origin policy \(SOP\)](#)
- [Viewing the Product Help](#)

Application Workflow Manager

- In an Application Workflow Manager modelled action, you retrieve a file from the remote zOS system directly into a COBOL project. You must set the text file encoding of the COBOL project to UTF-8 in the project properties in order for the Compiler to correctly process the source code. To set the encoding, in the project properties of the COBOL project and on the **Micro Focus > Project Settings > COBOL**, specify the `source-encoding"utf8"` Compiler directive in the **Additional directives** field.

CICS

EXEC CICS VERIFY PASSWORD has a failure scenario that doesn't exist on the mainframe. It is currently not possible to distinguish this failure from an incorrect password on the command itself via the returned EIBRESP and EIBRESP2.

Under Enterprise Server, the following options mean that an admin LIST request to ESF is necessary to acquire the appropriate information:

```
CHANGETIME
DAYSLEFT
EXPIRYTIME
INVALIDCOUNT
LASTUSETIME
```

If no users are signed on to CICS, then this admin request will be made as the default CICSUSER which will be rejected.

The VERIFY PASSWORD command will return NOTAUTH 2, "The supplied password is wrong".

This is misleading because it implies that the value for PASSWORD on the command itself is incorrect. Note that in that case, CICS returns NOTAUTH 2 as per the mainframe.

The aggregated return code from ESF to CICS for the admin request is identical in both cases and so CICS has no way of distinguishing between these two failure scenarios. We therefore cannot return a new EIBRESP or EIBRESP2 for this particular case.

To allow CICSUSER to make admin LIST requests, the anonymous list option needs to be enable in the region's security manager configuration (not the Security Manager):

```
[Admin]
allow-list=yes
```

Commuter licenses

If there are a number of entries displayed in the License Administration tool, these might be either for different products or for the same product but for different serial numbers, for example.

When a commute request is made using the license manager, that request is serviced from a pool of licenses that could be provided by any number of Micro Focus product licenses stored on that server. The commute request cannot target a particular RMS license by product name, serial number, or number of cores.

The products to which you apply the commuter licenses will be licensed correctly but the number of licenses left in the License Administration tool might appear to be less than expected. This issue is more likely to occur with multiple entries for the same product.

If you are using a number of different products, Micro Focus recommends that you use a different licensing server for each product in order to help minimize the issue.

Debugging

- When debugging `.gnt` code on early builds of Windows 10 and Windows Server 2016, execution fails to stop at COBOL breakpoints. To resolve this, you should install the most recent Windows updates for your operating system, as this problem is only known to be present with the earlier versions.
- When debugging a remote program, the performance of the network, specifically the latency between the local and remote machines, will have an effect on the responsiveness of Eclipse. Whenever Eclipse has to show the new execution position, a number of messages are sent from the remote machine, over the network, to get the required information. The effect of this is that it can take a few seconds after a Step command is issued before Eclipse has updated the screen and is ready for the next command.

You can mitigate this delay by closing any unnecessary debug views. For instance, by closing the Expressions view and the Variables view, the number of messages that are sent is reduced, which allows Eclipse to update quicker.

- Large programs can suffer from a large delay the first time that the program is displayed in the debugger and an expression is evaluated. You can reduce this delay by following these steps:

1. Create a text file called `debugconfig.xml`.
2. Add the following lines, and then save the file:

```
<?xml version="1.0" encoding="utf-8" ?>
<NativeDebuggerOptions>
  <DebugOption Option="REMOTECHECKER" Value="yes"/>
</NativeDebuggerOptions>
```

3. At a command or shell prompt, set the environment variable `DEBUG_CONFIG` to the full path name for `debugconfig.xml`.
 4. Start Eclipse from the same command or shell prompt.
- Remote debugging does not work for programs running on AIX or HP machines, if you are trying to debug using Enterprise Developer installed on a Linux machine.
 - `core_on_error=3` does not work correctly on AIX 7.2.

Database Access

- If you have a remote COBOL project under Eclipse that uses DB2 ECM in Visual COBOL or HCO for DB2 LUW in Enterprise Developer and the DB2 software is not installed on the client machine where

you are using the Eclipse IDE, you receive background parsing errors in your application. To resolve the issue, you can do either one of the following:

- Disable background parsing **Window > Preferences > Micro Focus > COBOL > Editor** and disabling the checkbox for **Background parsing**.
- Install the IBM DB2 client-side software on the machine on which you are running Eclipse. Go to the IBM Support Home and locate the page entitled "IBM Download Fix Packs for IBM Data Server Client Packages" .

Docker

- Using the Micro Focus SupportLine Support Scan Utility, `MFSupportInfo`, in a container is not supported on Windows.
- When using the Docker support provided in the Enterprise Developer Build Tools or Visual COBOL Build Tools, on Windows Server version 1803 or Windows 10 version 1803, the following errors might be encountered:
 - Windows Server version 1803: container "xxxx" encountered an error during CreateContainer: failure in a Windows system call: The operating system of the container does not match the operating system of the host. (0xc0370101)
 - Windows 10 version 1803: container "xxxx" encountered an error during Start: failure in a Windows system call: The compute system exited unexpectedly. (0xc0370106)

The problem is encountered only when creating containers which are intended to build COBOL applications. The creating of containers for application execution is not affected.

Please, contact Micro Focus SupportLine for further information.

Enterprise Server

- The Historical Statistics Facility may generate incorrect records for SSTM-enabled enterprise servers.
- On Windows 10, if you are using Microsoft's EDGE browser to access the Enterprise Server Administration GUI, issues with EDGE can cause the automatic refresh feature to display a dialog asking whether you want to resubmit a form. To work around this issue, cancel the resubmit request and then refresh the server list page or the Home page of Enterprise Server Administration. You can also turn off the automatic refresh by setting the **Auto-refresh interval** setting on the Home page of Enterprise Server Administration to 0.
- In Enterprise Developer 2.3, some job scheduling tools from third-party vendors might stop communicating with Enterprise Server. This might have an impact on our external partners who integrate with Enterprise Server such as job schedulers which do not use MFBSI. The issue might occur with job schedulers such as Zena, CA, ASG or UC4, if these are run outside of a COBOL environment with the path set. [Click here](#) for more information about the issue and how to work around it.

imtkmake

- On AIX v7.2, you must install IBM's patch APAR IJ04075 in order for `imtkmake -genclient` to work correctly. See http://www.microfocus.com/docs/links.asp?vc=IBM_APAR_IJ04075 for details about how to install the patch.

Installation

- Installing this product fails if Micro Focus License Manager version 10000.2.02070 (April 2015) or older is already installed on your machine. The setup files does not upgrade the License Manager automatically to the latest version. You must uninstall the existing version before you start the installation.

Micro Focus Rumba

- On versions of Windows Vista and later, Enterprise Server listens only on the IPv4 loopback address (127.0.0.1). As a result, an attempt to connect to localhost with a TN3270 emulator such as Micro Focus Rumba may fail. To work around this issue, in your emulator's configuration use 127.0.0.1 in preference to localhost or your host machine's name.

OpenSSL

The version of OpenSSL included in Enterprise Developer and Enterprise Developer 4.0 is intolerant of expired X.509 certificates in the CA Root Certificate collection in some circumstances. Specifically, if you configure a PEM file that contains multiple certificates as the collection of trusted CA certificates, and one or more of the certificates in that file have expired, OpenSSL might not be able to load and use other certificates in the file which are still valid. This can cause certificate validation errors and connection failures for SSL and TLS connections.

To work around this, you need to remove any expired certificates from such collections. If necessary, replaced them with any valid certificates that have the same Distinguished Name and Modulus (key). This is a known issue with OpenSSL. Micro Focus is investigating for other possible workarounds.

PL/I Support

- The Micro Focus PL/I Macro Preprocessor supports the majority of the IBM PP(MACRO) functions. The only support for PP(PLX) is for the deprecated KEYS option. There are no plans to further extend the existing support for PLX.
- On SUSE, the PL/I CodeWatch debugger does not display output with MicroFocus ViewNowX. To resolve this issue, you need to install a HotFix of ViewNowX - contact Micro Focus SupportLine for more details.

Resource Adapters

- Trying to deploy the local resource adaptor `mfcobol-localtx.rar` to WebLogic may fail with a `ClassCastException`. To work around this issue, you need to deploy `mfcobol-xa.rar` first, then need to undeploy this file and deploy the local one, `mfcobol-localtx.rar`. If there are issues deploying using the WebLogic GUI, you can use the command line. If there are issues with this as well, try reducing the length of the command (for example, by moving the file to a location with a shorter path).

REST/JSON IMTK implementation and the same-origin policy (SOP)

- HTTP requests sent from scripts within a web browser to REST services deployed on an enterprise server might fail due to the same-origin policy (SOP). Some browsers might implement (or support plugins that implement) techniques for relaxing SOP such as cross-origin resource sharing (CORS) that enable sending cross-origin requests successfully. For REST services, Enterprise Server does not implement a method for relaxing SOP. This means that browsers that implement CORS (or any other SOP relaxation technique) might still forbid requests made from scripts due to Enterprise Server not implementing the equivalent technique on the server side.

Viewing the Product Help

- A known issue in JRE 1.7.0_45 (issue JDK-8028111) causes problems with searching and indexing the help in the Eclipse help viewer if you are running Eclipse on a machine that has this version of JRE installed.

Upgrading from Mainframe Express to Enterprise Developer

An introduction to the process of upgrading your COBOL applications

Enterprise Developer enables you to develop applications that run on the mainframe or offload applications from the mainframe and develop them locally on Windows. If you are upgrading from Mainframe Express, you can move your existing applications to Enterprise Developer and continue to develop them in the new environment.

Once you have familiarized yourself with the basics of the Enterprise Developer IDE, you can import your existing code in Enterprise Developer.

See the *Upgrading to Enterprise Developer* guide for full details on upgrading your applications from Mainframe Express to Enterprise Developer, the changes between the two products and what technologies are not supported or have been superseded.

How to create an application created with Mainframe Express to Enterprise Developer

The following provides an outline of the incremental approach to upgrading a Mainframe Express application to Enterprise Developer where you use both development products to maintain the application while gradually moving the sources to the new IDE.

1. Review your Mainframe Express application:

- Create an inventory of the sources it uses, the copybook dependencies, data files, SQL, JCL and IMS files, third party utilities, integrated preprocessors.
- Check what workgroups are used and what the workflow processes are, what repositories are used to store the sources, what the build scripts and command line parameters are, how the application accesses the mainframe.



Note: Use the MFECCL command line utility of Mainframe Express to export the project details to an .xml file which will include all project details at a glance - files in use, workgroups, output and dependency folders, Compiler directives, project and file options. The command to use is:

```
mfecl /exportxml project-name.mvp [ xml-filename.xml ]
```

- Review the XML file to see what information it includes:

```
2 <project name="bankdemo.MVP" />
3 <projectsettings>
83 <workgroups>
84 <level description="System">
94 <level description="Production">
101 <level description="Development">
118 </workgroups>
119 <component name="UDATECNV.MLC" type="source assembler">
125 <component name="UTWOSCMP.MLC" type="source assembler">
131 <component name="MBANK10.BMS" type="source bms">
139 <component name="MBANK20.BMS" type="source bms">
147 <component name="MBANK30.BMS" type="source bms">
155 <component name="MBANK40.BMS" type="source bms">
156 <folder location=".\\BMS\\MBANK40.BMS" type="source bms" />
157 <settings>
```

The file provides details about the project and file settings, as well as of the workgroup levels. You are going to use this information when you are configuring the new projects in Enterprise Developer.

- Analyze the strengths and weaknesses of your application in the Mainframe Express environment and to what extent it meets the business requirements. The application might be a good candidate for being modernized by upgrading it to Enterprise Developer.
2. Define a small selection of the source files of your Mainframe Express application that you will move across to Enterprise Developer as a start – this could be the files responsible for a single operation of the application.

See *Considerations* for whether the files need to be recompiled in the new environment.

3. Prepare your original application:

- Replace any hard-coded paths in your Mainframe Express project with concatenated ones – to do this, in Mainframe Express, click **Project > Convert Paths to Relative**.
- Backup the sources.

4. Create mainframe subsystem projects in Enterprise Developer.

In Eclipse:

- a. Click **File > New > Mainframe COBOL Project**.

To emulate workgrouping behavior in Eclipse, you can use copybook projects that will define the copybook paths for each level, and you can add these projects as dependency projects to the ones for the source files. See *Emulating workgroups in Eclipse* in this guide.

5. Configure the projects in Enterprise Developer.

In Eclipse, to specify project properties:

- Navigate to the project properties, expand Micro Focus and set the project properties as taken from the *xml-filename.xml* file.

6. Import the source files and establish a product baseline – use linking to add the files to the new projects in order to store the files in their original location:

In Eclipse:

- a. Right-click the project, and click **New > Folder** if you wish to add an entire folder, or **New > File**.
- b. Click **Advanced**, and click **Link to alternate location (Linked Folder)** or, respectively **Link to File in the file system**.
- c. Browse to the folder in your Mainframe Express project that includes the files you wish to add, select the folder or the file to add, and click **Finish**.

7. Set individual file properties:

- Right-click a file in your project, and click **Properties**.
- Set individual file properties as taken from the *xml-filename.xml* file.

8. Compile the sources in Enterprise Developer.

In Eclipse:

- By default, Eclipse performs a build automatically as you are adding the source files. Check the **Console** view to see if there are any errors.

9. Create and configure an enterprise server instance for the projects.

Check the *xml-filename.xml* file to see what regions the original application uses, and what its settings are.

10. Run the application.

- Run JCL applications by submitting the JCL file – right-click the JCL file in your project, and click **Submit JCL** (in Visual Studio) or **Submit to Enterprise Server** (Eclipse).

Use the Micro Focus Rumba emulator supplied with Enterprise Developer to connect to the enterprise server instance you have created. Run the application in Rumba.

11. Debug and perform thorough testing.

12. Configure MFA and the **Mainframe Explorer** window or the **Remote Systems** view.

13. Plan for rolling out Enterprise Developer project and servers to your development and QA teams. Also, organize relevant training for them.

- a. As a first step, ensure your project is configured as desired and that it builds successfully.
- b. Distribute the project:

In your Eclipse workspace, copy the project folder with all the files. You can store this on a shared network location. Users need to import the existing project and sources locally in their workspaces.

Enterprise Server templates:

Create your own templates using the following command at an Enterprise Developer command prompt:

```
mfds -x [repository type] [repository address] [server name] <options>  
<user id> <password>
```

Distribute the server .xml files across your development teams. The file should be saved in the %ProgramFiles(x86)%\Micro Focus\Enterprise Developer\etc\ServerTemplates subfolder in the local Enterprise Developer installation folders.

Using Enterprise Developer for Visual Studio

Understanding the structure of Enterprise Developer solutions

On creating a new project, the following files are created in the file system with the following structure:

```
...  
|_Location  
|_Solution  
|_Name  
|_bin  
|_x86  
|_Debug  
|_Release  
|_x64  
|_Debug  
|_Release  
|_obj  
|_x86  
|_Debug  
|_Release  
|_x64  
|_Debug  
|_Release  
|_Properties  
_Solution.sln  
_Name.cblproj  
_Name.dep  
_Program1.cbl
```

If you select the Create Directory for Solution option when creating a solution, the structure is slightly different.

In the *Solution* folder:

- **Solution.sln** - a description of the solution and what it contains.
- **Name.cblproj** - the project file that is opened in Visual Studio, which holds the description of the project and all its related configuration and directives information.

- COBOL source files - when you create a project, a skeleton COBOL source file `Program1.cbl` is added for most of the project templates.

In the *Name* folder:

- `..\bin` - this is the default location of build artefacts. With this folder are the subfolders `x86\Debug` that contains the executables or libraries, and `.idy` file for each of the project's COBOL source files. The `.idy` files contain information required for debugging your application. When you use the Release build configuration, build output goes to a subfolder `x86\Release` and no `.idy` files are created.

Debug and Release are standard build configurations that you launch from the Visual Studio task bar. They use a different set of compiler directives as well as outputting different files. You can create your own build configurations by clicking **Build > Configuration Manager** and choosing **New** from the **Active solution configurations** drop-down list.

The `x86` folder exists because the default output platform is 32-bit. If you change this to be 64-bit, you will instead find your output in an `x64` folder.

- `..\obj` - this also has `x86\Debug` subfolders, and contains an `.obj` file for each source file, used in intermediate build stages. The `obj` folder also holds supporting information such as logs and file lists.



Note: The project file `.cblproj` is an msbuild file, much like a makefile but consisting of XML that you can extend and modify to customise your builds. You can use this directly from command line, as it uses the same build environment as the IDE, and behavior is identical. This means you can have a single source of configuration information that makes your build process easier to maintain.

If you open a command prompt and change to the *Location* folder, you can execute the `msbuild` command, without needing to specify the `.cblproj` file.

Finding your way around the IDE's features

- **Solutions and projects**

A solution is a container holding one or more projects that work together to create an application. The solution has the extension `.sln`. A COBOL project has the extension `.cblproj` and a C# project has the extension `.csproj`.

Solution Explorer shows the solution that is open and the projects therein.

You can use the project's properties pages to display a list of the files in your solution with file details like output file type and location, COBOL dialect, and the number of errors generated by the file. To display the properties, click **Project > Name Properties**.

- **COBOL editor**

The COBOL editor provides help such as column cut and paste, and background syntax checking, which underlines errors with red wavy lines (also known as "squiggles"), which you can then hover over to display details of the syntax error.

When you are editing, you can insert code snippets and navigate forward and backward quickly, and the **Find All References** option enables you to search for references of any COBOL data items, section and paragraph names in the solution.

You can customize the editor to display line numbers, adjust colorization, tabs, and margins, from the **Text Editor > Micro Focus COBOL > Advanced** page in **Tools > Options**.

When developing code, the editor provides IntelliSense that helps you write syntactically correct code and, in managed code, helps when you need to type more complicated constructs, such as the code to override the members that a class inherits from a base class or the code for implementing an interface.

The feature for implementing an interface helps complete incomplete interface declarations. A appears at the beginning of the declaration: click it and choose the missing member(s) of the inheriting interface.

When you encounter a COPY statement, or data item that is defined in a copybook, if you put your cursor on that code and press F12 the appropriate copybook opens in the editor at the relevant line. You can also do this by right-clicking the line and selecting **Show copybook name**.

- **Setting Compiler directives**

Many Compiler directives are set automatically by certain configuration options in the IDE, but you can explicitly add directives to your project. Right-click on the project in Solution Explorer and choose **Properties**. In the COBOL tab, you can see directives that are set by the IDE in the **Build Settings** text box. Enter others in the **Additional Directives** text box as a space-separated list.

If you use a separate text file to manage your directives, you can reference this instead by entering the USE "*directives file*" directive. You should enter a relative path.

- **Build Tools, the Output Pane and the Error List**

Build configurations define how to build a project or solution. There are default configurations of Debug and Release for each project type, and you can create your own specific configurations.

The Output window shows the results of your build together with errors. You can double-click an error and navigate directly to the appropriate line in the source code. You can do the same from the Error List.

- **Debugging**

When you debug the application, you can step through the code, hover over a data item to see its value, and watch data item values in a variety of ways. You can specify breakpoints on a range of conditions, such as when an expression is true or changes, or when a line is hit a specified number of times.

In native code, you can set COBOL watchpoints on data items and watch for changes in the area of memory associated with the watchpoints. When the memory changes, the debugger breaks on the line that follows the line on which the data change occurred.

Also in native code, you can use the Memory window to watch the contents of the memory that is associated with data items or expressions.

Change the Defaults to Replicate Your Existing Project Structure

Change the location of source files

To add an existing COBOL source file to your project, right-click the project in Solution Explorer and choose **Add > Existing item**. You can then browse to the sources you want to add.

- If you click **Add**, Enterprise Developer makes a copy of the file, which it saves in the project folder. Any edits you make to this file do not get applied to the original.
- If you click **Add As Link**, a reference to the original file, rather than a copy of it, is added to the project in Solution Explorer. If you then open the file in Visual Studio, any edits are applied to the file in its original location.

You can also drag files from Windows Explorer and drop them into your project in Solution Explorer. This also makes a copy of the file and leaves the original in place.

To remove a file from your project, but not delete the file on disk (whether added as a link or not), right-click the file in Solution Explorer and choose **Exclude From Project**.

Change the location of built files

By default, built artefacts for the Debug configuration are created in the `..\Location\Solution\Name\bin\x86\Debug` folder.

You might want to change this, so that several developers can save built items in the same folder for example. To do this, right-click the project in Solution Explorer and choose **Properties**. In the COBOL tab, change the value of the **Output path** field to the preferred folder. (We recommend you always use relative

paths when entering this value.) When the project builds, the output files will be saved in this folder, and the folder created if it doesn't already exist.

To change the output path for the Release configuration, select **Release Configuration** in the COBOL property page and change the value of the output path.

Change the type of built files

The default output and target types when you create a project depend on the project type. You can change these settings on the project **Properties** page. Use the following table to show the default output and target types for each project and the possible changes once the project has been created :

Project type	Output type	Target type	Possible output types	Possible target type	
Native	Console application	.exe	single	.dll, .exe	single, multi
	Windows application	.exe	single	.dll, .exe	single, multi
	Link library	.dll	single	.dll, .exe	single, multi
	Enterprise Server application	.dll	multi	.dll, .exe	single, multi
	INT/GNT application	.int	multi	.int, .gnt	multi
Managed	Console application	.exe	single	.dll, .exe	single
	Windows application	.exe	single	.dll, .exe	single
	Link library	.dll	single	.dll, .exe	single
	Procedural multi-output project	.dll	multi	.dll, .exe	multi

Best Practice in Enterprise Developer Development

Break down large projects

Projects with a large number of source files and build artefacts can be hard to navigate and slow to build. If you find this the case, we recommend that you review the contents of large projects and split them into separate projects (and possible separate solutions) in which you group items that are logically related. These projects can still be built in the same output folder if required.

For example:

- If you have different versions of a product for different customers, keep common source in one project and a separate project for each customer. You could also have a master solution into which you add projects from other solutions by right-clicking a solution and selecting **Add > Existing Project**.
- If you have core code that is rarely changed or recompiled, keep that in one project and have separate projects for those areas that change regularly.

Referencing common sources

To avoid repetition and reduce maintenance effort, you should consider keeping all your Compiler directive settings in a directives file and reference this file in each project. Similarly you should keep copybooks in a single project and add this project as a dependency to your COBOL projects.

If using managed code and multiple projects, use project references rather than file references.

Create templates

After creating and configuring a project, you can save the settings as a template that can be reused and distributed to other users. It can be added to the list of project types available when clicking **File > New > Project > COBOL**.

To create a template of the open project, click **File > Export Template** and follow the steps explained in the Export Template Wizard.

Use relative paths

Keep source relative to a base path and avoid full paths so that code is portable and easy to use with source control systems. You should also avoid using network shares or drives.

Modernize Your Applications and Processes

Following industry standard development practises

Many source code control systems and Agile tools can be integrated into the Visual Studio IDE.

You should also consider using continuous integration, which involves the automatic building and testing of an application after a change occurs to the source code. This method traps errors sooner in the development life cycle and can greatly improve efficiency and reduce costs.

Interface modernization

Enterprise Developer enables you to use Visual Studio's built-in design tools to create more intuitive user interfaces. By wrapping existing procedural COBOL in an wrapper class you can integrate your code into Windows Forms (WinForms) and Windows Presentation Foundation (WPF) technology, and WebForms for ASP.NET browser-based applications.

Multi-user applications

Enterprise Developer includes a Run Unit API to enable multiple users to simultaneously use an application based on COBOL code that was designed originally for a single user.

Developing Web-based applications

You can use Enterprise Developer to migrate existing, core applications to a service oriented architecture as Web services, and deploy them using Micro Focus COBOL Server and Enterprise Server, so that you can develop COBOL-based software components to be invoked across the Web.

You can do this by creating an Enterprise Server application

Developing .NET applications

Both new and existing COBOL can be compiled as .NET managed code. This enables you to:

- Reuse existing COBOL business logic and data access across the .NET environment
- Access .NET Framework classes and features from COBOL applications including Windows Forms and Web Forms
- Create and extend composite applications consisting of COBOL, C#, VB.NET, C++ and ASP.NET
- Reuse and extend Open ESQL applications

Both procedural and OO COBOL are supported within the .NET framework. OO COBOL classes can inherit classes written in other Microsoft .NET languages and vice versa.

The managed COBOL syntax includes many extensions to the COBOL language to support .NET features; for example, the TRY ... CATCH syntax to enable exception handling in COBOL.

There are also certain directives that help integrate your managed COBOL with other languages in the .NET environment. For example, you can now expose the Linkage section and entry points in your COBOL to other managed languages by compiling with the ILSMARTLINKAGE directive.

Modernizing Dialog System applications

Enterprise Developer provides the following support for Dialog System applications:

- Dialog System run-time system and run-time components.
- Panels V2.
- Dialog System painter.
- GUI class library and OLE class library. These libraries are needed if you migrate an existing Dialog System application that was extended using those libraries.

Projects for building the GUI and OLE class libraries from source are also supplied. Additionally, a project file for the Base class library was added in Enterprise Developer 2.0.

- Visual Studio plug-in to associate screensets in Visual Studio with Dialog System. Double-clicking a screenset in Solution Explorer in Enterprise Developer starts the Dialog System painter.
- Sample applications demonstrating a range of modernization techniques.
- Supporting documentation in this Help explaining the significant elements of the sample code.

You can modernize Dialog System applications within Enterprise Developer. You migrate an application to Enterprise Developer and from there you can run the application without change, or modernize it over time.

Modernization techniques include:

- A Windows Forms form replacing a Dialog System dialog, where the form can contain .NET controls. See the Customer + .NET WinForm sample `CustomerWinForm.sln`.
- A Windows Forms control wrapped as an ActiveX control and used on a Dialog System dialog. See the Customer + .NET GridView User Control sample `custgrid.sln`.
- A WPF user control hosted by a Windows Forms user control, which is then exposed as ActiveX ready for use by Dialog System. See the Customer + .NET WPF GridView User Control sample `CustGridWPF.sln`
- A .NET managed code application interacting with Dialog System as native COBOL `.dll`. See the Managed Customer sample `ManagedCustomer.sln`.

Data File Tools

Enterprise Developer comes with two versions of the Data File Tools utility: Data File Tools and Classic Data File Tools.

The Classic Data File Tools is the utility that was previously available in Net Express. It includes the Data File Converter, Data File Editor, IMS Database Editor, Segment Layout Editor and the Record Layout Editor. This utility is only available on Windows.

Data File Tools is a new version of the utility and comprises the Data File Editor and the Structure File Editor. This utility is available on both Windows and UNIX.

Procedural COBOL Compared with Managed COBOL

Procedural COBOL is regular COBOL without any of the new syntax that has been added for .NET and JVM. This is the COBOL that will have been used to write Net Express, Server Express and Mainframe Express applications, and it is still actively supported today.

You can compile to native or (in most cases) managed code. The core COBOL syntax is supported in managed code. However, there are some features that are not supported (for example Panels V2, Dialog System and ACUCOBOL-GT). This means that you can take most existing COBOL applications and recompile to create managed applications.

Managed COBOL

Managed COBOL is the collective term for .NET COBOL and JVM COBOL.

Managed COBOL is COBOL with extensions to support the .NET and JVM frameworks. It offers OO syntax support and syntax to allow access to the available class libraries.

When you compile managed COBOL the compiler generates managed code: `.class` or `.jar` for JVM which will run on the Java Virtual Machine.

Managed Code and Native Code

You can compile your COBOL program to managed code using the `jvmgen` compiler directive. From within the IDE this happens automatically if you are using a managed COBOL COBOL JVM project.

The compiler has now created an intermediate language (JVM byte code `.class/.jar`).

COBOL and all other JVM languages (for example Java, JRuby and Jython) compile to this format, which makes mixed language applications easy to write.

You can also create native code applications. In Eclipse the default COBOL project compiles to native code.

The compiler generates `.exe/.dlls` as the result of a native compilation.

The native COBOL application has to call the appropriate management services available for the operating system, whereas a managed application can take advantage of the management services provided by the run time such as exception handling, garbage collection, and thread management.

Run Time

The JVM byte code (`.class/.jar` files) can be deployed to a JVM for execution.

The JVM's just-in-time (JIT) compiler compiles the byte code into code native to the operating system. The JVM provides additional services including memory management, exception handling, garbage collection and thread management.

Developing Native and Managed Applications

You use the IDE to develop, compile and debug both native and managed applications. You can write new COBOL code or you can recompile existing COBOL applications to managed or native code, potentially without any code changes.

You can deploy and further debug the application under the run-time system provided by Enterprise Server.

JVM COBOL applications are deployed to a Java Virtual Machine for execution.