



# What's New in the Enterprise Product Suite Version 5.0

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

© Copyright 2019 Micro Focus or one of its affiliates.

**MICRO FOCUS**, the Micro Focus logo and Enterprise Developer are trademarks or registered trademarks of Micro Focus or one of its affiliates.

All other marks are the property of their respective owners.

2019-06-19

# What's New

This *What's New?* document covers some of the new features and functions in the latest release of the Micro Focus Enterprise Product Suite. Updates apply to the following products:

- **Micro Focus Enterprise Developer** which provides a contemporary development suite for developing and maintaining mainframe applications, whether the target deployment is on or off the mainframe.
- **Micro Focus Enterprise Test Server** which provides a comprehensive test platform that takes advantage of low cost processing power on Windows environments, to supply scalable capacity for testing z/OS applications without consuming z/OS resources.
- **Micro Focus Enterprise Server** which provides the execution environment to deploy fit-for-purpose mainframe workload on Linux, UNIX and Windows (LUW) environments on IBM LinuxONE (IFLs), standalone servers, virtual servers, or the Cloud.
- **Micro Focus Enterprise Server for .NET** which provides the execution and modernization platform to deploy fit-for-purpose mainframe workload on a scale-out .NET infrastructure and the Azure Cloud.

This document helps you to quickly understand the new capabilities within the 5.0 release.

Enhancements are available in the following areas:

- [.NET Core Support](#)
- [Amazon Web Services Quick Start for Enterprise Server](#)
- [COBOL Formatting](#)
- [Code Analysis](#)
- [Enterprise Server and Scale Out Architecture](#)
- [Enterprise Server Common Web Administration](#)
- [Application Workflow Manager](#)
- [AppMaster Builder](#)
- [CICS Support](#)
- [COBOL Language Enhancements](#)
- [Compiler Directives](#)
- [Consolidated Trace Facility](#)
- [Data File Tools](#)
- [Database Access](#)
- [Database Connectors](#)
- [Debugging](#)
- [Eclipse Integration](#)
- [Enterprise Server for .NET](#)
- [Enterprise Server Security](#)
- [IMS Support](#)
- [JCL Support](#)
- [Mainframe Access](#)
- [Micro Focus Native Database File Handler](#)
- [Micro Focus Rumba](#)
- [Micro Focus Unit Testing Framework](#)
- [PL/I Support](#)
- [Platform Support](#)
- [Product Documentation](#)
- [Visual Studio Integration](#)

## .NET Core Support

[Back to Top](#)



**Note:** .NET Core on Linux is provided as an Early Adopter Product (EAP) feature only.

.NET Core is an open-source development platform that supports multiple operating systems. You can now use Enterprise Developer to develop a COBOL application then publish it to be deployed on .NET Core.

The following project templates for .NET Core applications are available - Class Library (.NET Core) and Console App (.NET Core).

## Amazon Web Services Quick Start for Enterprise Server

[Back to Top](#)

Micro Focus has worked with Amazon Web Services (AWS) to create an AWS Quick Start for Enterprise Server. The Quick Start includes AWS CloudFormation templates and a deployment guide with step-by-step instructions that enable you to deploy Enterprise Server into a new or existing virtual private cloud (VPC) on the AWS Cloud in about 1 hour and 15 minutes.

For full information see *Micro Focus Enterprise Server on AWS* ([click here](#)).

## COBOL Formatting

[Back to Top](#)

You can now reformat your COBOL code in the editor according to your preference and standards.

Formatting sets the indentation based on the configuration options which you can set in **Tools > Options > Text Editor > Micro Focus COBOL > Formatting > Indentation** (for Visual Studio) and **Windows > Preferences > Micro Focus > COBOL > Editor > Formatter** (for Eclipse).

You can find the COBOL formatting features on the **Edit > Advanced** menu (for Visual Studio) and editor context menu (for Eclipse) menu.

## Code Analysis

[Back to Top](#)

You can now create the following views for your code:

- Program Flow Graph - enables you to view the overall structure of the COBOL program with nodes representing sections and paragraphs. PERFORM statements between them which affect the program flow are represented by links joining the nodes. External calls to other programs are shown but cannot be expanded in the Program Flow Graph view. Hovering over the nodes shows a tooltip containing the code representing the respective sections or paragraph. Clicking a node or a link joining said nodes positions the code editor to that point in the source, allowing quick and easy navigation of the program.
- Data Flow Analysis Tree - enables you to select a data item and then either statically trace what might change its value or what other data items it might affect.

## Enterprise Server and Scale Out Architecture

[Back to Top](#)

 **Note:** This feature is in Early Adopter Product (EAP) release status.

This release offers an enhanced and improved availability, and serviceability of Enterprise Server through the ability to administer and maintain a cross-system group of regions as a single system image with the new Performance and Availability Cluster (PAC).

In order for different Enterprise Server instances to be able to work together, they need to be able to share data. This is achieved through the use of a Scale Out Repository (SOR). All ES instances in a PAC will have a common SOR (PSOR) which is used to store CICS resources (limited to PCTs, PPTs, FCTs, DCTs and TSTs in this release) as well as internal system data to facilitate synchronisation between instances. Temporary Storage Queues and Transient Data Queues can also be shared between Enterprise Server instances by directing them to a SOR. Benefits of using PACs and SORs in this way include:

- Reduced hardware costs - taking advantage of the PAC to scale-out rather than scaling-up for more efficient use of processor resources.
- Easier maintenance - Dynamically adding or removing of regions to the PAC for system maintenance.
- Increased availability - in the event of the failure of an Enterprise Server instance, the PAC can continue to operate with reduced capacity. Enterprise Server instances can reside on different machines, improving availability further.
- Dynamic scaling - Enterprise Server instances can be added to, or removed from, the PAC depending on demand.
- Better performance - throughput is no longer restricted by the resources on a single machine (scale-up).
- Improved serviceability - you can now administer the PAC and any associated regions as a single image from a new contemporary web-based administration interface (ESCWA).

## Enterprise Server Common Web Administration

[Back to Top](#)

 **Note:** This feature is in Early Adopter Product (EAP) release status.

Enterprise Developer now includes a new Enterprise Server Common Web Administration interface (ESCWA). ESCWA is a web user interface and server for modern administration, monitoring and control of Enterprise Server. It offers improved usability that consolidates the different Enterprise Server user interfaces so that native and managed regions, and security stores can be managed in one place. Features include:

- Administering directory servers across multiple hosts.
- Monitoring and control of CICS and JES Enterprise Server instances.
- Configuring and administering a security store, defined in an LDAP-compatible security manager such as Microsoft Active Directory or OpenLDAP.
- Administering the Scale-Out features - enable you to specify logical groups of Enterprise Server instances, and configure and run Performance Availability Clusters (PACs) and their related Scale-Out Repositories (SORs).
- Administering, monitoring and control of Enterprise Server for .NET regions and listeners.
- The use of current web frameworks that have a greater focus on security.

## Application Workflow Manager

[Back to Top](#)

Improvements have been made in the following areas:

- Tools from the AWM, Eclipse, and ISPF Function Packages have been restructured and improved:

- Tool parameters are now named so that the position of the parameter is not significant anymore.
- You can only specify parameters which are supported by the tool.
- Only the parameter types String and File are now supported.
- Mass Processing is now an attribute of a tool parameter and is no longer a special input/output parameter type.
- The tool relationship "Resource Operation Value" replaces the parameter type "ToolDefinedResourceOperation".
- A new REST Function Package has been added. It provides support for the following functions - DELETE, HEAD, GET, PATCH, POST, and PUT.

The new JSON File Descriptor has been provided to support REST payloads. This function package enables you to integrate RESTful API with your AWM workflow. See *REST Function Package* in your product Help.

- New tools in the Eclipse Function Package enable you to store and retrieve information from the Eclipse secure store:
  - Save To Secure Storage; Read From Secure Storage; Delete From Secure Storage

You can use these tools to securely store login credentials to a remote system and reuse the information for subsequent access.

- A new attribute is now available to enable you to add comments to every model component in the AWM model editor. See *Comments in the AWM Model Editor* in your product Help.

## AppMaster Builder

[Back to Top](#)

The AppMaster Builder Data View Explorer tab has been redesigned in this release make it easier to create and maintain data views. Features include:

- Creating SQL data views directly from the mainframe DB2 catalog
- Creating IMS and VSAM data views without the requirement for a global DDIFILE
- Generating data views using new DDIMETA files located in the DDIMETA folder in the AMB project
- A data view import process is provided to create DDIMETA files from existing DDISYMB files

## CICS Support

[Back to Top](#)



**Note:** This feature is in Early Adopter Product (EAP) release status.

This release includes the following enhancements:

- CRCN CICS transaction - this new in-built CICS transaction monitors the state of XA connections for each resource manager (RM) entry defined in the system. If CRCN finds that connections have become unavailable, it will attempt to reconnect at regular intervals. (The frequency of the monitoring is controlled by the ES\_XA\_RECONNECT environment variable.) CRCN will emit messages to the console to warn of connections lost and regained.

## COBOL Language Enhancements

[Back to Top](#)

The following enhancements have been made to the COBOL syntax:

- ISO2002 conditional compilation - an additional mechanism for conditional compilation, the ISO2002 Conditional Compilation method, is now available. This has been implemented as part of the support for Enterprise COBOL 6.2.

The following compiler directing statements are supported:

```
>>DEFINE
```

```
>>IF/>>ELSE/>>END-IF
```

```
>>EVALUATE/>>WHEN/>>WHEN OTHER/>>END-EVALUATE
```

- The JSON PARSE statement enables you to convert JSON text into COBOL data formats. This has been implemented as part of the support for Enterprise COBOL 6.2.
- Async and await syntax (.NET COBOL only) - support is available for asynchronous programming in .NET COBOL using the async and await syntax.
- INSPECT statement - the performance of INSPECT... CONVERTING has been improved, particularly for cases where `source-alphabet` is long.
- Support has been added for the portable syntax for SET - `set[string]`.

## Compiler Directives

[Back to Top](#)

The following Compiler directives are new in this release:

<b>JVMTARGET</b>	Specifies the JVM version number targeted by the class files generated in this compilation unit.
<b>ILNORMALIZENAMES</b>	Determines the program elements that are normalized when a program is compiled to .NET or JVM COBOL. Normalization in this context results in replacing all \$ and - (hyphen) characters with _ (underscore) characters in the name of the generated class file.

## Consolidated Trace Facility (CTF)

[Back to Top](#)

This release provides the following enhancements:

- Availability of CTF from the Eclipse interface - a new property tab, CTF, in the Eclipse launch configuration dialog box helps to enable CTF tracing and specify a CTF configuration file. By default, the trace files are then saved in a **Logs** subfolder within the project's folder.
- Dynamic CTF - it is now possible to configure the CTF dynamically from outside the process being traced. This feature is only currently available on Windows platforms.

Using a new command line utility - `cblctd` - you can alter the tracing events of running applications that already have CTF tracing enabled. You can alter trace levels, add or remove components to or from the trace, and also configure the emitters in effect.

Additionally, there is also a new component that you can trace - `mf.mfdbfh` enables you to trace activity of the Micro Focus Native Database File Handler.

## Data File Tools

[Back to Top](#)

This release provides the following enhancements:

- The Data File Editor now includes a Compare Files tool that enables you to compare the contents of two data files side-by-side.
- Structure files, and the layouts within them, can now be created within the Data File Editor; you no longer need to use the Classic Data File Tools utility to manage your layouts.
- When connecting to a VSAM dataset stored in an enterprise server region, you can store any passwords required for access, for the duration of your current session.
- You can view archived JES spool jobs that have been merged into one spool file using the merging archived spool files process.
- You can now quickly duplicate records in non-indexed files, using the Duplicate Record option.

## Database Access

[Back to Top](#)

Enhancements are available in the following areas:

### SQL Option for DB2

- XML data type support - the following XML scalar functions are now supported - XMLATTRIBUTES, XMLCOMMENT (add), XMLCONCAT, XMLDOCUMENT, XMLELEMENT, XMLAGG(delete), XMLNAMESPACES, XMLPARSE, XMLPI, XMLSERIALIZE, XMLTEXT, XMLQUERY.
- Initial XDB Server (**add**) support for z/OS DB2 V11 - includes support for archive tables and ALIAS on sequences (delete).
- XDB Link support for z/OS DB2 V10/V11/V12 (**add**)

### OpenESQL ADO.NET Connection Editor Toolbar

- A new "i" (information) button is available that provides all the details of any installed .NET provider. Provides additional information mitigating resource definition and use within the ADO.NET Connection editor
- Static XA switch modules "Reconnect" - At the beginning of a global transaction, the XA switch will attempt to reconnect temporarily broken connections to the database. If there is a temporary disconnect to the database, there is no need to manually re-enable the XA switch to run transactions.

## Database Connectors

[Back to Top](#)

You can now create an identity column in your database by using a new XFD directive - IDENTITY\_COLUMN. This must be specified on a numeric field.

## Debugging

[Back to Top](#)

This release includes the following enhancements:

- Animate - Eclipse now includes an additional menu option when debugging a COBOL program for setting the speed and toggling Animate mode. When active, the program will automatically step through each line of source code, updating the source code in the editor, the Debug Stack, Outline and Variables Views. See *Debugging Commands and Options* in your product Help.
- COBOL Visualizer - a new window in Visual Studio enables you to view and edit a data item during debugging in either text or HEX forms. You invoke the window from the DataTip for the data item.

- Conditional watchpoints - you can now specify conditions and hit counts for COBOL watchpoints. A condition can be in the form of an expression or it can be a hit count (for the number of times memory associated with a data item has been updated). The condition is evaluated each time the data item being watched changes. If the condition evaluates to true, then program execution stops. This enables more fine-tuning of the current watchpoint support and is useful when debugging large, complex programs.
- Dynamic core dumps - you can now invoke a core dump programmatically and continue execution of your program. A new library routine, `CBL_CREATE_CORE`, enables you to produce a core file for the current process or a process owned by the same user that owns the current process.
- Improved visibility of the PERFORM stack - the PERFORM stack and PERFORM range names are visible in system debuggers such as `gdb` in many cases on 32-bit Linux with Intel architecture and Solaris with Intel architecture. This was already the case on AIX, 64-bit Solaris-Intel, and all other Linux platforms (including 64-bit Linux-Intel).
- Live Recording - a new environment variable, `COB_LIVE_RECORD_SIZE`, is now available to aid performance during the creation of a live recording file. Use it to specify the amount of memory (in MB) that is to be used to store events that are to be written to the live recording file. When the limit is reached, the buffer is cycled so that the oldest events are removed to make space for the most recent events.
- In order to include source line numbers in run-time error messages, which makes the diagnosis of issues in your program easier, the COBOL compiler now produces `.idy` files by default when building from the command line, unless the `NOANIM` directive is specified. You can safely remove `.idy` files after compilation if you do not need them; they do not need to be distributed with your application unless you want to debug your code. However, unless you specify the `ANIM` directive, generated executables such as `.dll` and `.gnt` will not be debuggable, which is consistent with prior behavior.



**Note:** Behavior has not changed if you are using an IDE or build scripts generated by an IDE.

- Stackdump files (Linux Intel 64-bit, Solaris Intel 64-bit, and Linux/390 platforms only) - you can produce a stack trace for applications in a production environment that encounter a Run-Time system error. Use the `stackdump_on_error` and `stackdump_filename` tunables to configure the production of a stackdump file when an error occurs, from which you can locate the exact line of failing code in your program.
- A new configurable tool is now available that dumps to a file stack traces of all threads in process on any Run-Time System error. [Linux Intel, Solaris Intel, zLinux only – requires `ANIM` on 32-bit Linux Intel]. This is only supported for native COBOL.
- Line number information is now being output for optimized builds (on Linux Intel 64-bit, Solaris Intel 64-bit, PowerLinux, zLinux, and Solaris SPARC platforms only). This is only supported for native COBOL. Because of what optimization does, the information is imperfect, but it remains highly indicative. This gives the following advantages:
  - Enables the use of the `stackdump` utility (where available) for optimised builds.
  - Provides location information should a signal occur when running in a system debugger such as `gdb`.
  - Provides some limited ability to step through optimized code in system debuggers.

## Eclipse Integration

[Back to Top](#)

This release includes enhancements in the following areas:

- Supported Eclipse versions - this release supports Eclipse 4.7 (Oxygen) and 4.8 (Photon). The setup file installs Eclipse 4.8.  
Eclipse 4.6 (Neon) is no longer supported.
- Java support - the 64-bit Java 8 is required to run the Eclipse IDE. The setup file installs Adopt OpenJDK 8 (u202) with Hotspot. Java 8 (32 or 64-bit versions), and Java 11 (64-bit version only) are

supported for executing JVM COBOL code and for native COBOL and Java interoperability. For full details on the supported Java versions, see *Software Requirements* in your product Help.

- COBOL editor:
  - Automatic insertion of END statements - Eclipse now automatically inserts the end clause for IF, EVALUATE, PERFORM, EXEC and TRY statements.
  - Automatic insertion of closing symbols - by default, Eclipse now automatically adds closing quotes and brackets where required.
  - Search text positioned between specified columns - the **Micro Focus Find/Replace** dialog box now includes options to search between specified columns in the code.
- Directive sets - Compiler directives for your projects are now saved in language-specific `.mfdirset` XML files stored within a `.settings` subfolder in the project's folder. You can create your own versions of `.mfdirset` files and verify them against an XML schema that Micro Focus supplies.

Previously, all settings were stored in a single, proprietary IDE project file. Existing projects which currently specify directives and other similar settings in the `.cobolproj` file will be upgraded to use the new `.mfdirset` files when imported or loaded in Enterprise Developer 5.0. If you are storing your source code in a source control system, ensure that you add the `.mfdirset` files to source control as well.

- Remote connections - support is available for automatic SSH tunnelling of Micro Focus Enterprise Developer UNIX Components network traffic. You can route most Micro Focus Enterprise Developer UNIX Components network traffic using SSH port forwarding. This makes developing on a remote UNIX host running a firewall easier to configure. In addition, the network traffic is encrypted as it is being sent within an SSH tunnel. See *To create a connection to a remote UNIX host running behind a firewall* in the product Help.
- User Interface:
  - The **COBOL Explorer** view uses an icon overlaid with P to indicate COBOL programs that will not be built because they are preceded by another of the same name on the build path.
  - Improved settings UI - the COBOL, IMS, and the Run-time Configuration settings pages now use a property grid that also shows the definitions of the settings.

## Enterprise Server for .NET

[Back to Top](#)

This release includes the following enhancements:

- Improved CICS compatibility - CICS EIBRCODE handling in Enterprise Server for .NET now matches the mainframe specification. CICS BMS support in Enterprise Server for .NET now matches the behavior on the mainframe and in native Enterprise Server more closely.
- MMC administration UI support for ESF security - the MMC snap-in for Enterprise Server for .NET administration now supports the External Security Facility. This means you can use LDAP-based or custom security solutions to provide administrator authentication and access control.
- TLS (SSL) support - the `seelistener` utility now supports TLS (also known as SSL) for its TCP communications channels. These include all channel types except MQ. The Enterprise Server for .NET client classes also have been enabled for TLS. For example, the `seeout` utility can now be used with a TLS-enabled MFBINP channel.
- Performance improvements - this release comes with an improved performance in WCF, system database operations, and VSAM database operations.

## Enterprise Server Security

[Back to Top](#)

This release includes the following enhancements:

- Security - security features can now be employed when developers and administrators install new COBOL services (web services and EJBs) into an enterprise server instance over the network. There are a number of authentication and authorization options that can be enabled. See *Deployment Listeners* and *The .mfdeploy File* in your product Help.
- Vault Facility - a new security feature has been added that enables some Enterprise Server components to keep certain sensitive information in a form of storage defined as a vault, accessible via a configurable vault provider. The default vault provider stores data in encrypted format on disk.
- OpenSSL 1.1.1 - the OpenSSL security provider has been updated to OpenSSL version 1.1.1.

This is the stable Long Term Support version of OpenSSL.

- Added support for the ratified TLS protocol version 1.3. TLS 1.3 benefits include:
  - Much shorter initial connection negotiation sequence. This reduces the time taken to establish a link before starting to transmit data.
  - Using only the most secure ciphers and hash methods.
  - TLS 1.3 will be negotiated in preference to the older TLS protocols.
- Added support for new Ciphers and Key Exchange groups in line with TLS 1.3 requirements.
- The default security level for previously configured endpoints has been moved from Security Level 0 to Security Level 1. This removes the ability to accidentally make use of known-weak elements such as SSL3 and MD5. Similar changes to the default Security Level have recently happened to Java, Chrome, Firefox, and other systems providing secure connections.
- Fileshare Security - the Fileshare Secure TCP/IP transport provider now supports the trusted use of X509 certificates bearing the name of the Fileshare service as the Common Name element of the certificate.

In previous releases, a secure connection to a Fileshare server was made using a certificate that represented the network location upon which the Fileshare service was located. This method is still supported, but does not distinguish between the exact Fileshare server that is being connected to when more than one service can exist on a single host system. With this change, individual Fileshare services can identify themselves by using a unique certificate. While running on the same host and registered with the same network endpoint.

- Support for Active Directory user groups and group name mapping - the Enterprise Server External Security Facility's MLDAP ESM Module can now use Active Directory user group objects for Enterprise Server user groups. Also, the module can now map long group names to the 8-character-maximum names required for mainframe emulation.
- Selective auditing - administrators can audit only security activity of particular interest, reducing audit overhead and the volume of events. The Enterprise Server External Security Facility's MLDAP ESM Module can now enable ESF Audit events only for particular users, groups, and resources.
- Improved interaction with LDAP client libraries resulting in fewer LDAP-related issues and easier diagnostics - the Enterprise Server External Security Facility's MLDAP ESM Module has improved interoperation with LDAP client libraries:
  - The client library vendor and version information is logged after the library is loaded
  - The module has better heuristics for loading the correct library supplied by the OS vendor, so the "provider" configuration option can generally be omitted
  - For OpenLDAP, the module sets its proprietary "connect timeout" option

## IMS Support

[Back to Top](#)

This release includes the following new feature:

- Checkpoint and restart are now available for output GSAM files in addition to input GSAM files.

## JCL Support

[Back to Top](#)

This release includes the following enhancements:

- Spool file housekeeping - you can now merge your archived spool files to a central location where they can be viewed using the Micro Focus Data File Editor. This copies the archived spool files and merges them into a new, combined archive. It does not change the original archive, so the audit trail remains untouched, but the combined archive allows viewing of all the jobs that have been merged. The merge process is carried out using a JCL job or by using a command line utility.

## Mainframe Access

[Back to Top](#)

This release provides enhancements in the following areas:

- Full support for the z/Server functionality - MFA Server now has the capability to service all z/Server requests. This means it is no longer necessary to configure z/Server, and the number of started tasks running on the mainframe is reduced. Installing and configuring MFA Server becomes easier as z/Server is no longer required.
- Browse dataset using IDCAMS dump - when using the MFDAS BROWSE command, it is now possible to view the output of the IDCAMS DUMP command on the specified dataset in hex and printable characters side-by-side.
- Improved JES support for executing jobs and input datasets - it is now possible for MFA Server to interact with JES jobs that are in executing phase, and to list and download input datasets.
- Retrieve binder information for load modules - when executing the MFDAS BINDER command, it is now possible to view the binder information from a load module.
- TSO command server - MFA Server includes a new type of application server to support executing TSO commands as the current user. The address space is spawned as your user id and any valid TSO command can be executed.
- UNIX sub-system (USS) support - MFA Server can now operate on files stored on a USS file system. The available functions are - edit, browse, rename, delete, and copy USS files and directories.

## Micro Focus Native Database File Handler

[Back to Top](#)

 **Note:** This feature is in Early Adopter Product (EAP) release status.

The Micro Focus Database File Handler (MFDBFH) is a file handler that enables your native CICS and batch enterprise server applications to interact with data files and certain infrastructure files stored within a database.

VSAM files may experience logical and/or physical corruption in the event of a hardware failure. This is due to a single COBOL I-O operation consisting of multiple physical I-O operations. If all the physical I-O operations that make up the COBOL I-O operation are not performed, the file can be left without integrity. In previous releases you could guard against such occurrences with Fileshare and logging, but they required some manual intervention and came with performance cost.

In this release, configuration options have been added that allow the I-O operations to be routed to an RDBMS to take advantage of the availability and performance characteristics of the database. This can be achieved with no changes required to the application. The file will be routed to the SQL database simply by changing the filename prefix to `sql://`.

Features include:

- CICS and batch Cobol file I-O can now be routed to an RDBMS with only simple configuration changes. No application changes are required.
- Supports all Micro Focus file formats
- Files can be routed to an RDBMS on a file by file basis.
- Utilities are provided to enable data to be moved between disk files and RDBMS.
- Support for PostgreSQL 10.x, Amazon Aurora PostgreSQL, and MS SQL Server (on Windows or Linux Intel)

## Micro Focus Rumba

[Back to Top](#)

This release comes with Micro Focus Rumba+ Desktop version 10 which you can choose to install with Enterprise Developer. This will enable embedded Rumba support within the IDE.

## Micro Focus Unit Testing Framework

[Back to Top](#)

This release provides support for the following functionality:

- Data-driven unit tests - a new type of test suited to testing data where values read from a source are passed through the same tests.
- JCL unit tests - two types of test that interact with the JES engine on an enterprise server region. These tests are run from the command line.
- Test output colorization - test output can now include basic colorization for supported terminal emulations. This feature must be run from the command line.

## PL/I Support

[Back to Top](#)

Enhancements are available in the following features:

- Code coverage in Eclipse - code coverage data can now be generated for PL/I programs in the Eclipse IDE using the same mechanisms as for COBOL programs.
- Compiler:
  - Full support for GET DATA.
  - Built-in functions:
    - New built-in functions - ALLOCATE, INDEXR, MEMCONVERT, MEMINDEX, ONCONDCOND, PLIFREE, REM, REPLACEBY2
    - The HEX built-in function now matches its z/OS version.
    - The PROCNAME and PROCEDURENAME functions used without trailing open/close parents.
  - Allow the use of VALUE to initialize members of a PL/I Data Structure.
  - Support for INONLY, NONASSIGNABLE attribute.
  - Diagnostics when a supplied parameter to a call is larger than the entry declaration.
  - Diagnostics when the Compiler detects that a loss of precision occurs as a result of operations on FIXED DECIMAL operands.
  - Compiler listings now provide additional information about arrays.
  - OPTIONS(REORDER) is now parsed but ignored.

- \*PROCESS OPT(TIME); is now supported.
- Support for larger concatenations when initializing variables or making assignments.
- Support for named constants of numeric PICTURE type in static structure dimensions and initializations.
- Diagnostics when the concatenation of two strings results in a string that is larger than the target variable.
- Support for the DUMMY ALIGNED and DUMMY UNALIGNED arguments.
- Support for the XML case asis and XML case Upper.
- Diagnostics if code attempts to use bitwise operators on non-bit variables.
- PL/I debugger in Eclipse:
  - Variables surrounding current point of execution now displayed in Variables view.
  - A Console view for PL/I programs that are being remotely debugged outside the control of Enterprise Server, and in Eclipse.
  - Data watchpoint setting using a variable declaration in the debugger.
  - PLIDUMP now emits STATIC variables in addition to AUTOMATIC and Parameters when documenting the callstack.
- The Macro preprocessor now supports %INSCAN and %XINSCAN.
- The EXEC preprocessor now supports the following:
  - SQL statements embedded within PL/I packages
  - The -maxmsg option for setting the maximum number of errors to be generated before terminating the compilation
  - Up to 4000 include files in a signal program
- PL/I Run-Time System - support is now available for:
  - Optionally generating Windows "Mini-Dumps" or a Linux coredump off the back of a call to PLIDUMP.
  - Dynamic allocation of datasets using the TITLE option(s).
  - Raising an ERROR condition when a called subroutine raises a COBOL RTS error.
  - FILE variables can have differing DCB when used in differing contexts.
  - STATIC FIXED BIN(31) variable accessed at open within the RECSIZE() attribute of the ENVIRONMENT.

## Platform Support

[Back to Top](#)

Support is now available for the following additional or updated operating systems:

- SUSE (Power) 11 SP3, 12
- Red Hat (Power) 7.2, 7.3
- Windows Server 2019

For a full list of supported platforms, check <https://supportline.microfocus.com/prodavail.aspx>.

## Product Documentation

[Back to Top](#)

The following is new in this release:

- Online context help in Visual Studio - online context Help is now the default setting for Enterprise Developer and opens the documentation available on Micro Focus SupportLine.

Enterprise Developer local help is still supported and can be installed manually, if required. See *After Installing* in the product Help for details.

- The *Comparison of COBOL with Other Managed Languages* document has been updated to reflect the recent enhancements in the .NET and JVM COBOL support.

## Visual Studio Integration

[Back to Top](#)

This release includes enhancements in the following areas:

### COBOL editor:

- Code snippets - new snippets are available for `$if` and `$region` statements, and for specifying `*cblformat off` which indicates the area of code that is unaffected by formatting.
- Collapsible regions - outlining is now available for IF, EVALUATE and PERFORM statements.
- **Peek Definition** context menu command - this invokes a pop-up window, embedded in the editor, showing the data item definition. You can use the feature both while editing and while debugging.
- IntelliSense suggestions - IntelliSense suggestions for the COPY statement now include directories to add.
- Find All References - the **Find All References** output window for COBOL has been enhanced, so you can now filter results, group results, order results and search results.
- Quick actions - the editor now displays light bulbs for creating a missing section or a paragraph, for adding and sorting \$ILUSING statements, and for removing or commenting out dead code.

### Visual Studio tool windows:

- The **Error List** window now includes a column for the **Sequence Number**.
- You can use Visual Studio's **Find All References** tool window to search for references in your COBOL projects.

### New settings in the project property pages:

- The **Debug** project properties page for managed projects now includes settings for either generating a portable symbol file (.PDB) or for embedding the portable symbol information directly in the assembly.
- The **Smart Linkage** section in the properties of managed projects now offers two new settings for enabling the ILSMARTTRIM and ILSMARTANNOTATE Compiler directives.

### New project template categories:

- The COBOL project templates are organized in a new set of categories that better represent the tasks and workloads used in modern development. Some of the categories, such as Azure or .NET Core, only show if you have installed the support for these features in Visual Studio. For more information, see *To install missing features from the IDE* in your product Help.

## Copyright and Disclaimer

© Copyright 2019 Micro Focus or one of its affiliates.

The only warranties for this product and any associated updates or services are those that may be described in express warranty statements accompanying the product or in an applicable license agreement you have entered into. Nothing in this document should be construed as creating any warranty for a product, updates, or services. The information contained in this document is subject to change without notice and is provided "AS IS" without any express or implied warranties or conditions. Micro Focus shall not be liable for any technical or other errors or omissions in this document. Please see the product's applicable end user license agreement for details regarding the license terms and conditions, warranties, and limitations of liability.

Any links to third-party websites take you outside Micro Focus websites, and Micro Focus has no control over and is not responsible for information on third party sites.

This product includes software developed by Computing Services at Carnegie Mellon University (<http://www.cmu.edu/computing/>).