

Micro Focus Visual COBOL 2010 R4 for Visual Studio Update 1 Release Notes



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

Copyright © Micro Focus IP Development Limited 2009-2011. All rights reserved.

MICRO FOCUS, the Micro Focus logo and Visual COBOL 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.

2011-09-09

Contents

Micro Focus Visual COBOL 2010 R4 for Visual Studio Update 1 Release Notes	5
System Requirements for Visual COBOL for Visual Studio	6
Installing Visual COBOL for Visual Studio	7
Licensing Information	9
What's New	10
Features Added in Visual COBOL 2010 R4	10
ACUCOBOL-GT Compatibility	10
ADO.NET Runtime Improvements	11
Creating Projects from Selected Files	11
Debugging Enhancements	12
Documentation	12
Embedded HTML	12
HCO for SQL Server (HCOSS)	12
Improvements to the Implements Smart Tag	12
Language Improvements	12
Large Projects Support	12
New Compiler Directives	13
New Samples and Tutorial	13
Project Details Window	13
Project Properties Updates	13
RM/COBOL Compatibility	13
Smart Linkage	14
WCF Services and Service References	14
XML Extensions	14
Features Added in Visual COBOL 2010 R3	14
.NET COBOL Syntax Improvements	14
Creating Projects from Existing Code	17
Add Existing COBOL Items Wizard	18
Override Class Members Dialog	19
Smart Tag for Implementing Interfaces	20
Snippet for Implements	20
Navigate To	20
Find All References	21
Web Application Projects	21
Debugging	21
Samples Browser	23
SQL CLR Integration Support	24
ACUCOBOL-GT Compatibility and RM/COBOL	24
XML Support	24
Features Added in Visual COBOL 2010 R2	25
File Handler	25
Go To Procedure Division	25
OpenESQL Assistant	25
Samples	25
Snippets	25
SQL Support	25
XML Parse/Generate	26
Features Added in Visual COBOL 2010 R1	26
Visual Studio as the Core Integrated Development Environment	26

COBOL 2010	26
COBOL Language Extensions	26
Known Issues	28
Resolved Issues	30
Updates and SupportLine	32
Further Information and Product Support	32
Information We Need	32
Creating Debug Files	33
Disclaimer	34

Micro Focus Visual COBOL 2010 R4 for Visual Studio Update 1 Release Notes

These release notes contain information that might not appear in the Help. Read them in their entirety before you install the product.



Note: This document contains a number of links to external Web sites. Micro Focus cannot be responsible for the contents of the Web site or for the contents of any site to which it might link. Web sites by their nature can change very rapidly and although we try to keep our links up-to-date, we cannot guarantee that they will always work as expected.

System Requirements for Visual COBOL for Visual Studio

Visual COBOL has the following requirements in addition to those of Visual Studio:

Hardware Requirements

Operating Systems	Windows XP Professional SP2 or later.
Disk Space	Approximately 60 Mb for the Sentinel RMS license server Approximately 200 Mb for Micro Focus COBOL integration into Visual Studio. Note: This disk size includes the space needed to cache information locally so that you can modify the installation without the original source media.

Software Requirements

You need to install the following software beforehand:

- Microsoft Visual Studio 2010 (Premium, Professional, or Ultimate)
- or
- Microsoft Visual Studio 2010 Integrated Shell

You can download the Visual Studio 2010 Integrated Shell from the [Microsoft Download Center](#).



Note: If you download the shell you need make sure that you run the installer to complete the installation. To do this run `vsintshell.enu.exe` from the location where you installed the download.

Installing Visual COBOL for Visual Studio

Download the setup file on your machine and install the product as follows:

1. Click on the **Visual COBOL : Click here to download your Integrated Development Environment for Visual Studio 2010 (Full Image)** link in your delivery notice and save the `visualcobolr4_update1.msp` file to your computer.
2. Run the file and follow the wizard instructions to complete the installation.
3. Install the COBOL 2010 Runtime Deployment Environment for Testing Purposes in the same way.



Note:

- If you are installing onto a machine that has an existing Micro Focus product that uses an older Sentinel RMS License Manager, you might be prompted to remove it and install the Micro Focus License Manager. By doing this you will maintain the existing Sentinel RMS license files while adding the Micro Focus License Manager. If you are unsure about existing licenses on your computer or removing the Sentinel RMS License Manager, consult your System Administrator. If you want to proceed, remove Sentinel RMS License Manager by using Windows **Add or Remove Programs** and rerun the installation file.
- Trial licenses cannot be used with remote desktop services. If you want to use your product in this way, please contact Micro Focus SupportLine to obtain a relevant license.
- We recommend that you install any updates for Visual Studio and the .NET Framework that are available at the [Microsoft Download](#) site.
- If you install JDK you might be prompted to install the latest update. The latest update is not required for use with Visual COBOL but you can install it if you wish.

Installing as an Upgrade

This WrapPack updates the existing installation of Visual COBOL R4 for Visual Studio.

- Visual COBOL and COBOL 2010 Runtime (or COBOL 2010 Test Runtime) cannot co-exist on the same machine.
- Make sure that the product is not running on your machine when you start the installation.
- To install the product, you need to be logged in with a user-ID that has write access to the registry structure under HKEY_LOCAL_MACHINE, HKEY_CLASSES_ROOT, and HKEY_CURRENT_USER so the installation software can set the environment appropriately. You also need to be logged on with Administrator privileges.

After Installing

You are now ready to run Visual COBOL. From the Windows taskbar click **Start menu > All Programs > Micro Focus Visual COBOL 2010 > Visual COBOL for Visual Studio**.

Please refer to the *Start Here* and *Product Information* sections in your product Help. Here, you will find information on getting started including tutorials and demonstration programs.

If you have problems trying to view the Micro Focus help, ensure that the Visual Studio Help Library is pointing to local help. From the Visual Studio menu click **Help > Manage Help Systems > Choose online or local help** and check the **I want to use local help** button.



Note: The first release of Visual Studio 2010 has a browser-based help system, Microsoft Help Viewer 1.0, which does not include an index for the locally-installed help. Navigation of the content is only available using the table of contents and Search and the help contents for the Help system does not expand and collapse in the same way as previous Help systems.

Visual Studio 2010 SP1 provides an upgrade of the help system, Microsoft Help Viewer 1.1, which provides a stand-alone help viewer with an index and a fully expandable table of contents.

If you do not wish to install Visual Studio 2010 SP1, you can install some third party tools that enable the index or the fully expanding table of contents. Read <http://kb.microfocus.com/display/4/kb/article.aspx?aid=31484> for more.

Licensing Information

 **Tip:** If you are unsure about what your licensing policy is or what sort of license you require, consult your System Administrator or Micro Focus SupportLine to obtain a valid license.

Windows Use the Authorization Code, supplied with your delivery notice, to license your product. To do this:

1. Click **Start > All Programs > Micro Focus License Manager > License Management System**.

The dialog box for the Micro Focus License System Administration Tool will be displayed.

2. Click the **Authorize** tab.
3. Enter your Authorization Code and click **Authorize**.
4. Close the Micro Focus License System Administration Tool.

If you wish to install the license silently, execute the following after the product has been installed:

```
start /wait <install-dir>\bin\cesadmintool -term activate  
AuthorizationCode
```

License for the Integrated Development Environment for Visual Studio

Your product is supplied with an evaluation license that typically enables you to use your product for 30 days. While you are using the evaluation license, a dialog will display when you start your product prompting you to enter the Authorization Code. This code can be found in the **Requested License Key** email that is supplied when you order your product. Depending on your product order, you might have multiple license Authorization Codes. To obtain a full license for Visual COBOL, enter the code that is printed under **Visual COBOL - Visual COBOL 2010 R4 for Visual Studio**.



Note: If you have purchased a Visual COBOL timed license, the license management system may show these as Trial licenses. This does not affect the license you have purchased.

License for the COBOL 2010 Test Runtime Environment

You need to install this license to run and test your applications in the runtime environment. It is a timed license and expires after one year after the date of purchase. After this time you need to contact Micro Focus to request a renewal.

There is no evaluation period for this license and you need to use your Authorization Code to obtain a full license before you can test your applications. The Authorization Code can be found in the **Requested License Key** email that is supplied when you order your product. To obtain a full license use the code that is listed under **Visual COBOL - COBOL 2010 Test Runtime**.

What's New

The following sections outline the new features that have been added in this release of Visual COBOL for Visual Studio.

Features Added in Visual COBOL 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 routines can now be used with Visual COBOL:
 - 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.

ADO.NET Runtime Improvements

The OpenESQL ADO.NET runtime has been re-engineered, offering several advantages over previous versions, providing:

- Better performance - the new runtime is much faster and can as much as double your performance, depending on the specifics of your application.
- Support for multiple run unit processing.
- Support for both client applications and SQL CLR stored procedures.
- For stored procedure applications:
 - Support for input/output parameters defined as COBOL data types, in addition to the previously supported .NET data types
 - Support that enables stored procedures to return more than one result set.
 - Static SQL Support for the DIALECT=MAINFRAME Compiler directive.
- For client-based applications:
 - Support for dynamic SQL that uses an SQLDA structure to pass parameter and result host variables, specifically including the following embedded SQL statements:

```
EXEC SQL
    PREPARE stmt1 INTO :sqlda FROM :stmt-buf
END-EXEC
```

- EXECUTE - For example, the following statement executes a prepared statement using a previously constructed SQLDA structure:

```
EXEC SQL
    EXECUTE stmt1 USING DESCRIPTOR :sqlda-structure
END-EXEC
```

- OPEN - For example, the following statement opens a cursor using a previously constructed SQLDA structure:

```
EXEC SQL
    OPEN c1 USING DESCRIPTOR :sqlda-structure
END-EXEC
```

For complete information on using these statements, see the [PREPARE](#), [EXECUTE](#), and [OPEN](#) topics.

- Additional support for result set processing that allows you to receive more than one result set back from a stored procedure.
- Support for applications built to target .NET Framework versions 2.0, 3.0 and 3.5, in addition to .NET Framework version 4.0.



Important: To take advantage of these improvements you must recompile your applications with the DBMAN=ADO Compiler directive. However, if you want to continue using the functionality available in previous releases, we now provide the DBMAN=OADO directive to use in place of the DBMAN=ADO directive. DBMAN=OADO provides the same functionality as was available with previous versions of the DBMAN=ADO Compiler directive, and has 100% backward compatibility with applications designed and developed with Visual COBOL R3.

Creating Projects from Selected Files

A new option, Create Project From Selection, is now available for your projects in Solution Explorer. You can select a number of COBOL files and copybooks in your project and opt to create a new project from them in the same solution.

Debugging Enhancements

The ability to load core dump files in Visual Studio has been added. This feature works with native COBOL only.

Documentation

If you are using Visual Studio 2010 Service Pack 1, the help is displayed in a stand-alone help viewer with an index and a fully expandable table of contents.

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 Visual COBOL for Visual Studio. 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.

Improvements to the Implements Smart Tag

The implements smart tag now supports value-types in addition to classes.

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. |

Large Projects Support

Visual COBOL has been optimized to work with bigger, more complex applications. This includes faster processing of multiple files and various IDE features that facilitate the process of developing large-scale project.

You can quickly move existing COBOL code into Visual Studio with the help of various wizards and windows such as the Create Project from Existing Code wizard and the Create Project from Selection wizard. The IDE is preconfigured so that during the file import it automatically scans the files and sets Compiler directives on them as appropriate.

New Compiler Directives

The following new Compiler directives are provided:

- ILCUTPREFIX - removes a specified prefix from the names of the COBOL data items in your source code.
- ILSMARTLINKAGE - exposes the Linkage Section and entry points to managed code by creating types.
- RUNTIME-ENCODING - determines the runtime encoding.
- SOURCE-ENCODING - passes the encoding of the source program to the Compiler.

New Samples and Tutorial

New samples and a new tutorial showing how to create WCF services in COBOL are available.

Project Details Window

A new diagnostic window, Project Details, is available for your COBOL projects and solutions showing a complete list of the files in a project or a solution and various file details. You can open the window from the context menu for a project or a solution in Solution Explorer.

Project Properties Updates

The project properties pages have been restructured to make setting options more intuitive.

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 routines can now be used with Visual COBOL:
 - 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 Visual COBOL, 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.

Smart Linkage

Exposing COBOL group items as managed types You can expose COBOL Linkage sections to other managed languages by using the ILSMARTLINKAGE directive. Smart Linkage saves the need to edit your original COBOL code or write wrapper classes. In this release the feature is available as a technology preview only.

WCF Services and Service References

Support is now available for adding WCF services as service references to your COBOL projects.

Note: WCF is not supported in the Visual Studio Shell but adding service references for client applications is supported.

XML Extensions

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

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.

Features Added in Visual COBOL 2010 R3

.NET COBOL Syntax Improvements

Quoteless syntax	Quotes are not needed when defining types, classes or methods, or when invoking classes and methods.
Construct improvements	The structure of class-id, method-id, enum-id, delegate-id, interface-id, valuetype-id has been improved.
Environment division, Configuration section, Repository	The Environment division, Configuration section and the Repository are no longer needed.
Static, Factory and Object blocks	The Static, Factory and Object blocks are no longer needed.
Attributes, Custom-Attribute and Class-Attributes	CUSTOM-ATTRIBUTE is now replaced by the ATTRIBUTES phrase. You no longer need to define class-attributes. Instead, specify the class custom attributes in the class definition.



Tip:

- Visual COBOL supports the older syntax, so projects that are using it will still compile. However, it is recommended to create applications using the new syntax and adhere to the *.NET COBOL Best Practices*.
- It is recommended to use the COBOL project and file templates, snippets and Intellisense as they use the new syntax. To see the new syntax in action, check the Visual COBOL samples.

The following is a more detailed overview of the changes in the syntax with examples:

Quoteless Syntax

Quotes are no longer needed when you define types, classes or methods, or when you invoke classes and methods. For example:

New Syntax	Old Syntax
01 01 type MyClass	01 01 type "MyClass"
type MyClass::New	type "MyClass"::"New"
set o to new MyClass	set o to new "MyClass"
set class::Property to value	set "class"::"Property" to value
set return-value to class::Method(param1)	set return-value to "class"::"Method"(param1)
invoke class::Method(param1)	invoke "class"::"Method"(param1)
<pre> method-id MyMethod public. local-storage section. procedure division. goback. end method. </pre>	<pre> method-id. "MyMethod" public. local-storage section. procedure division. goback. end method "MyMethod". </pre>

Construct of class-id, method-id, enum-id, delegate-id, interface-id, valuetype-id

The construct of class-id, method-id, enum-id, delegate-id, interface-id, valuetype-id has been improved as follows:

- You do not have to type a period after the declaration (for example, method-id *MethodName*).
- Quotes are no longer required around names.
- You do not need to use the name in the end marker.

New Syntax	Old Syntax
<pre> class-id Namespace.MyClass. object-storage section. method-id InstanceMethod. local-storage section. procedure division. goback. end method. method-id StaticMethod public static. </pre>	<pre> class-id. MyClass as "Namespace.MyClass". environment division. configuration section. repository. static. working-storage section. end static. object. </pre>

New Syntax	Old Syntax
<pre> local-storage section. procedure division. goback. end method. end class. </pre>	<pre> working-storage section. method-id. "InstanceMethod". local-storage section. procedure division. goback. end method "InstanceMethod". end object. end class MyClass. </pre>

Environment Division, Configuration Section, Repository

You no longer need to use an Environment division, a Configuration section or a Repository paragraph. For example:

New Syntax	Old Syntax
<pre> program-id. Program1 as "MyProject.Program1". data division. working-storage section. procedure division. goback. end program Program1. </pre>	<pre> program-id. Program1 as "MyProject.Program1". environment division. configuration section. repository. data division. working-storage section. procedure division. goback. end program Program1. </pre>

Static, Factory and Object Blocks

The Static and Object blocks are no longer used. With the new syntax you need only one working-storage section for items that were defined in a static or object block under the old syntax.

To define a static method, use the STATIC word.

The following example shows how to define static methods with the new syntax and how to avoid using an object block:

New Syntax	Old Syntax
<pre> class-id Namespace.MyClass. working-storage section. 01 my-object-data pic x. 01 my-static-data pic x static. method-id InstanceMethod. local-storage section. procedure division. goback. end method. method-id StaticMethod public </pre>	<pre> class-id. MyClass as "Namespace.MyClass". environment division. configuration section. repository. static. working-storage section. 01 my-static-data pic x. end static. object. working-storage section. </pre>

New Syntax	Old Syntax
<pre>static. local-storage section. procedure division. goback. end method. end class.</pre>	<pre>01 my-object-data pic x. method-id. "InstanceMethod". local-storage section. procedure division. goback. end method "InstanceMethod". end object. end class MyClass.</pre>

Attributes, Custom-Attribute and Class-Attributes

These are the changes for CUSTOM-ATTRIBUTE and class-attributes:

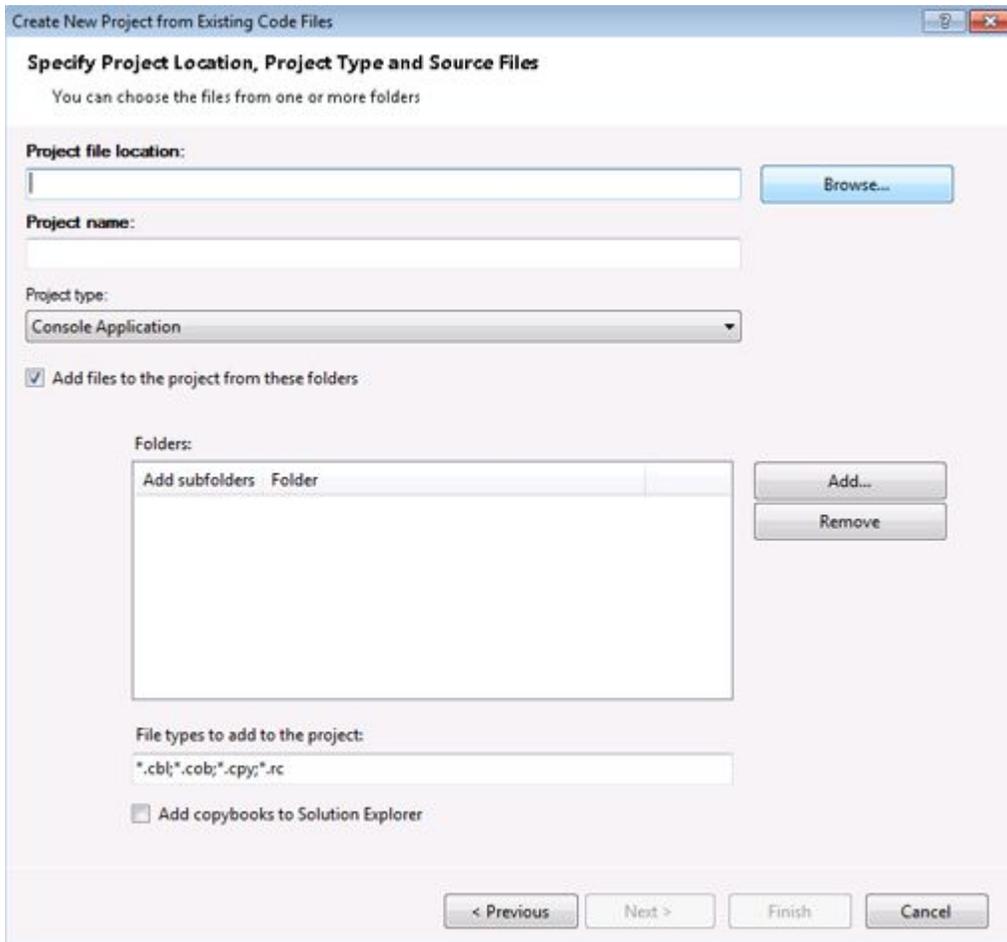
- The CUSTOM-ATTRIBUTE phrase is replaced by the ATTRIBUTE phrase.
- You no longer have to define `class-attributes`. Instead, specify the class custom-attributes in the class definition using the ATTRIBUTE phrase.
- Quotes are no longer needed around the name of the attribute and you can omit the word "Attribute" from the name.

For example:

New Syntax	Old Syntax
<pre>class-id MyNamespace.MyClass. attribute Serializable. working-storage section. ... end class.</pre>	<pre>class-id. MyClass as "MyNamespace.MyClass". class-attributes. custom-attribute is type "SerializableAttribute". object. working-storage section. ... end object. end class MyClass.</pre>

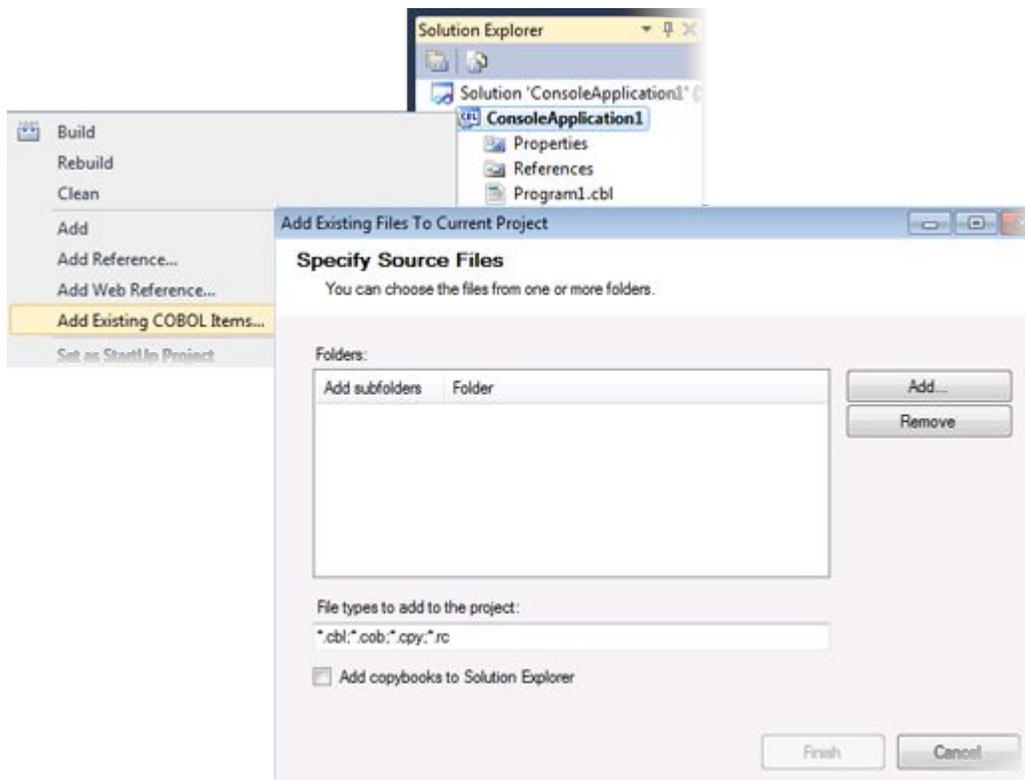
Creating Projects from Existing Code

Now you can create Visual Studio COBOL projects from existing applications using the Create New Project from Existing Code Files wizard. The wizard will create a new COBOL project and automatically add files to it from the specified directories. It will perform an automatic file scanning to identify which files are programs and copybooks, so that they can be correctly added to the project.



Add Existing COBOL Items Wizard

You can add existing COBOL files to your Visual Studio project using the new Add Existing COBOL Items wizard available from the context menu of the project in Solution Explorer. The COBOL files will be scanned to determine which ones are programs or copybooks, and then they will be added to the project.

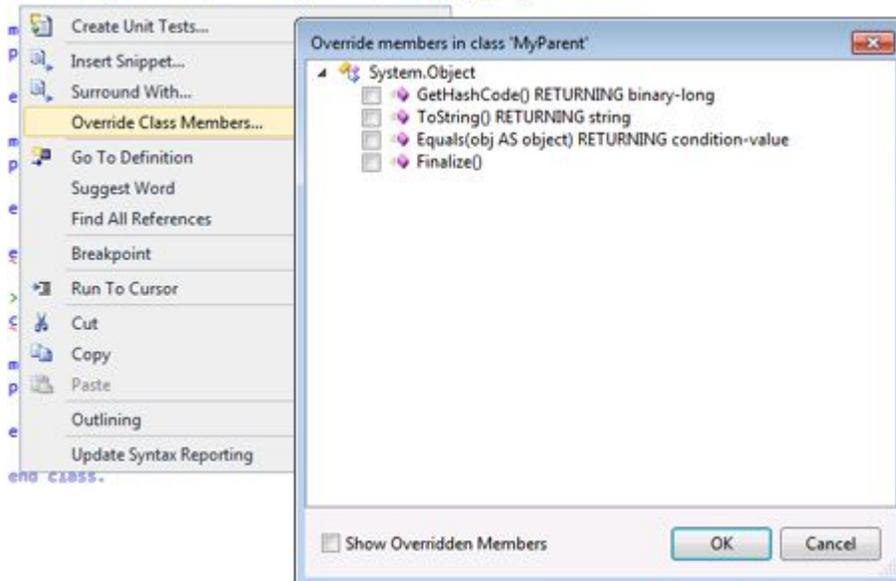


Override Class Members Dialog

 **Note:** This feature works with .NET managed code only.

The new Override Class Members dialog available in the editor enables you to override the members of inherited classes. The dialog helps you see the base classes from which a class inherits, select the members to override and add the construct of the overriding methods to the class.

```
class-id MyParent implements type IDisposable, type IEnumerable.
```



Smart Tag for Implementing Interfaces

 **Note:** This feature works with .NET managed code only.

You can now easily implement interfaces with the help of a Smart Tag. The tag appears underneath at the beginning of the declaration of any interface that is not fully implemented. To implement the interface, you simply need to click the tag.

```
class-id Class1 implements Type IDisposable, type IEnumerator.
```

```
type IEnumerator.
```



Implement interface 'System.Collections.IEnumerator'

Snippet for Implements

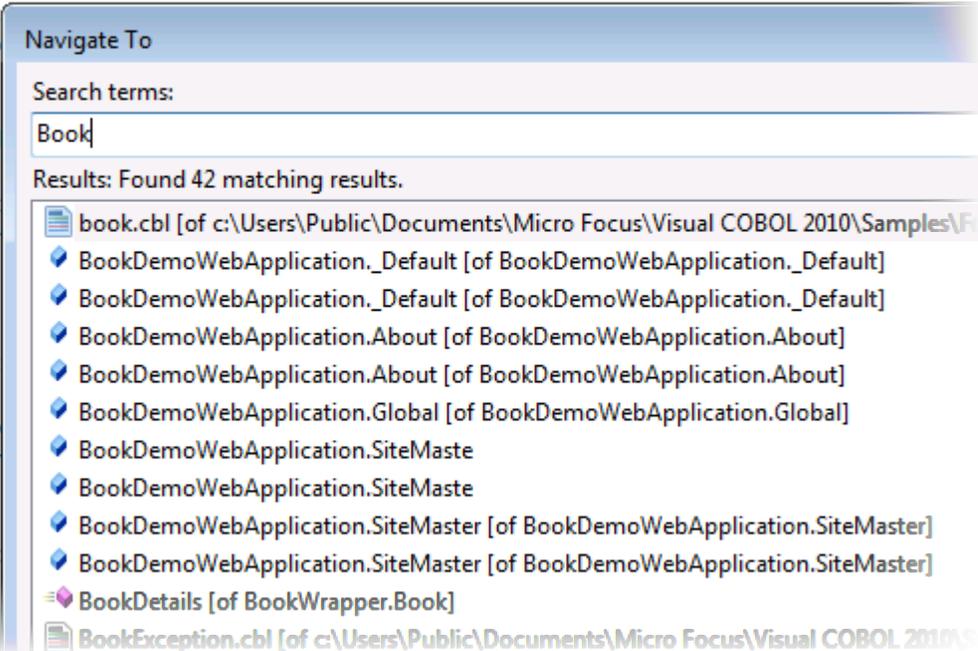
 **Note:** This feature works with .NET managed code only.

The snippet for implements has been improved. It now automatically implements the members from an interface and has improved support for more complex method signatures.

Navigate To

Use the Navigate To option in the Edit menu to search for files, variables and section names in all projects and files in your solution.

```
method-id Page_Load protected.  
loc  
pro  
end  
met  
wor  
01  
01  
loc  
pro  
try  
cat  
end
```



Navigate To

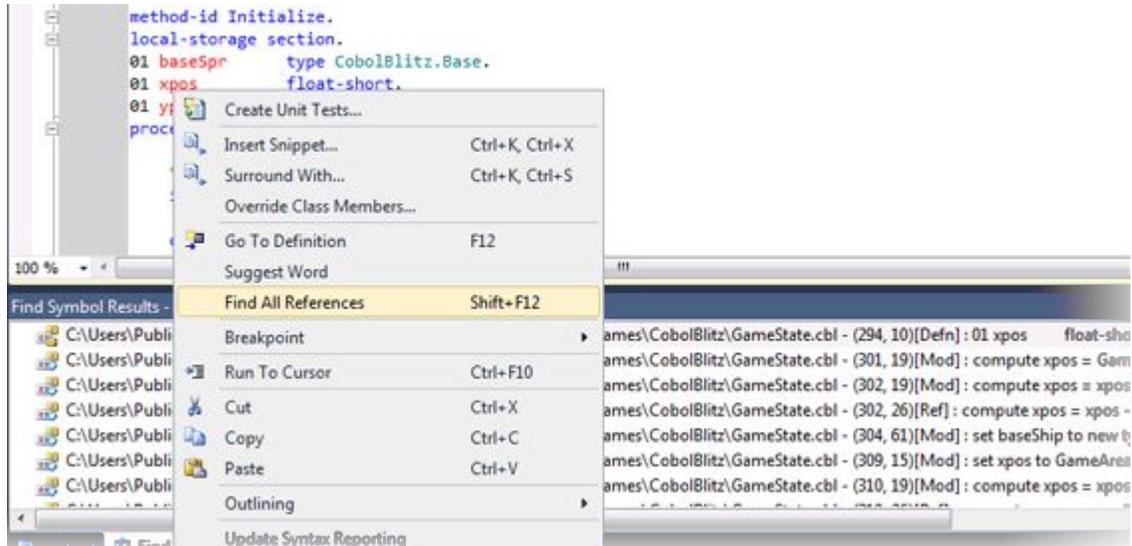
Search terms:
Book

Results: Found 42 matching results.

- book.cbl [of c:\Users\Public\Documents\Micro Focus\Visual COBOL 2010\Samples\F...
- BookDemoWebApplication._Default [of BookDemoWebApplication._Default]
- BookDemoWebApplication._Default [of BookDemoWebApplication._Default]
- BookDemoWebApplication.About [of BookDemoWebApplication.About]
- BookDemoWebApplication.About [of BookDemoWebApplication.About]
- BookDemoWebApplication.Global [of BookDemoWebApplication.Global]
- BookDemoWebApplication.SiteMaste
- BookDemoWebApplication.SiteMaste
- BookDemoWebApplication.SiteMaster [of BookDemoWebApplication.SiteMaster]
- BookDemoWebApplication.SiteMaster [of BookDemoWebApplication.SiteMaster]
- BookDetails [of BookWrapper.Book]
- BookException.cbl [of c:\Users\Public\Documents\Micro Focus\Visual COBOL 2010\S...

Find All References

The Find All References option available from the editor enables you to search for references of COBOL data items, section or paragraph names in your solution.



Web Application Projects

 **Note:** This feature works with .NET managed code only.

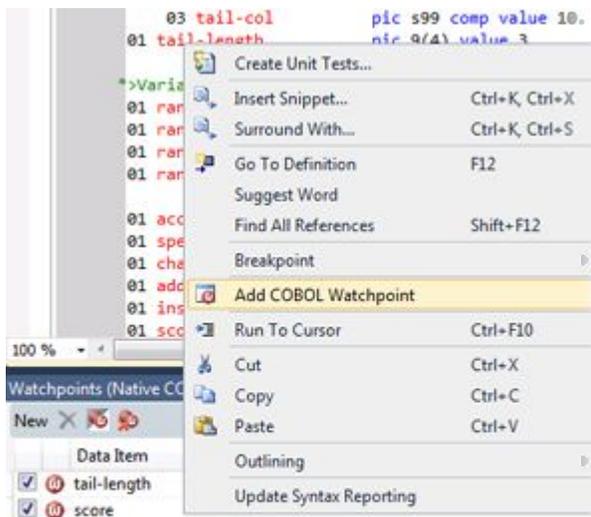
This release offers Web Application Project templates for creating COBOL Web applications and Web sites and applications. The benefits of using a Web Application project include:

- A Web Application project includes a project file which enables you to specify what files are part of the project and should be compiled.
- It adds namespaces for all items of the project.
- The source code is compiled into a single assembly on your local machine and is then deployed to the IIS server. You don't have to deploy the code behind.
- A Web Application project includes a "Publish" option for deploying the compiled assembly to an IIS server directly from the IDE using the automated tools of Visual Studio.
- Supports the Visual Studio Code Analysis feature.

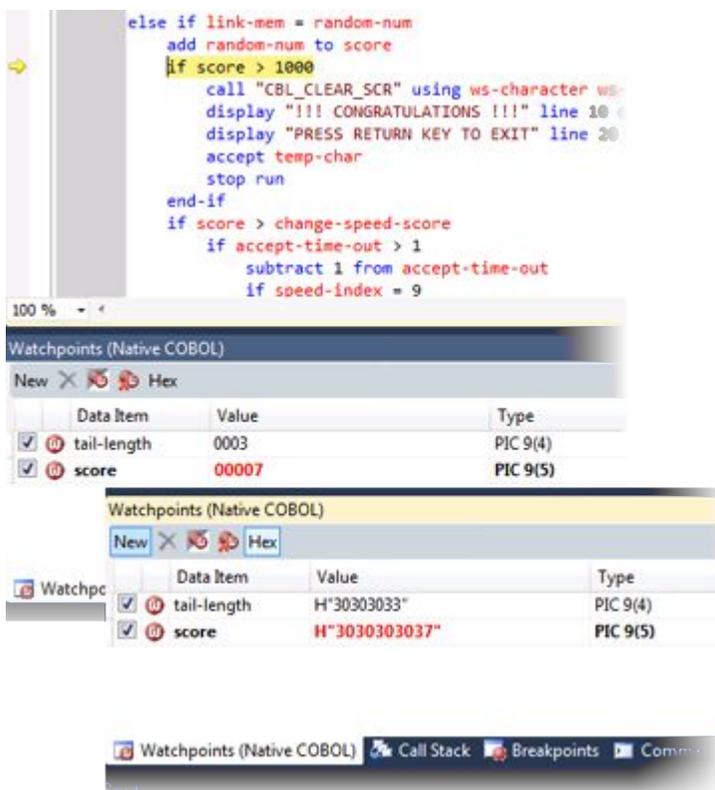
Debugging

The following debugging enhancements have been made:

- COBOL watchpoints and break on data change - you can set COBOL watchpoints on individual data items in native COBOL. COBOL watchpoints enable you to watch the area of memory associated with the particular data item and help track memory corruption. When the memory changes, debugging stops on the line that immediately follows the line on which the data has changed. This feature works with native code only.



- **Watchpoints (Native COBOL)** window - enables you to manage the COBOL watchpoints you add to your applications and view the contents of the memory associated with each watchpoint. This feature works with native code only.



- Simplified remote debugging - a simplified process for setting up remote debugging is provided.
- Attach to 64-bit process and debug - provides the ability to attach to and debug 64-bit COBOL processes.
- Debug tooltip for OCCURS items - you can now specify whether the debug tooltips for OCCURS items should display all items in an array or the value of an expression.

```

working-storage section.
01 var string occurs 5.

procedure division.

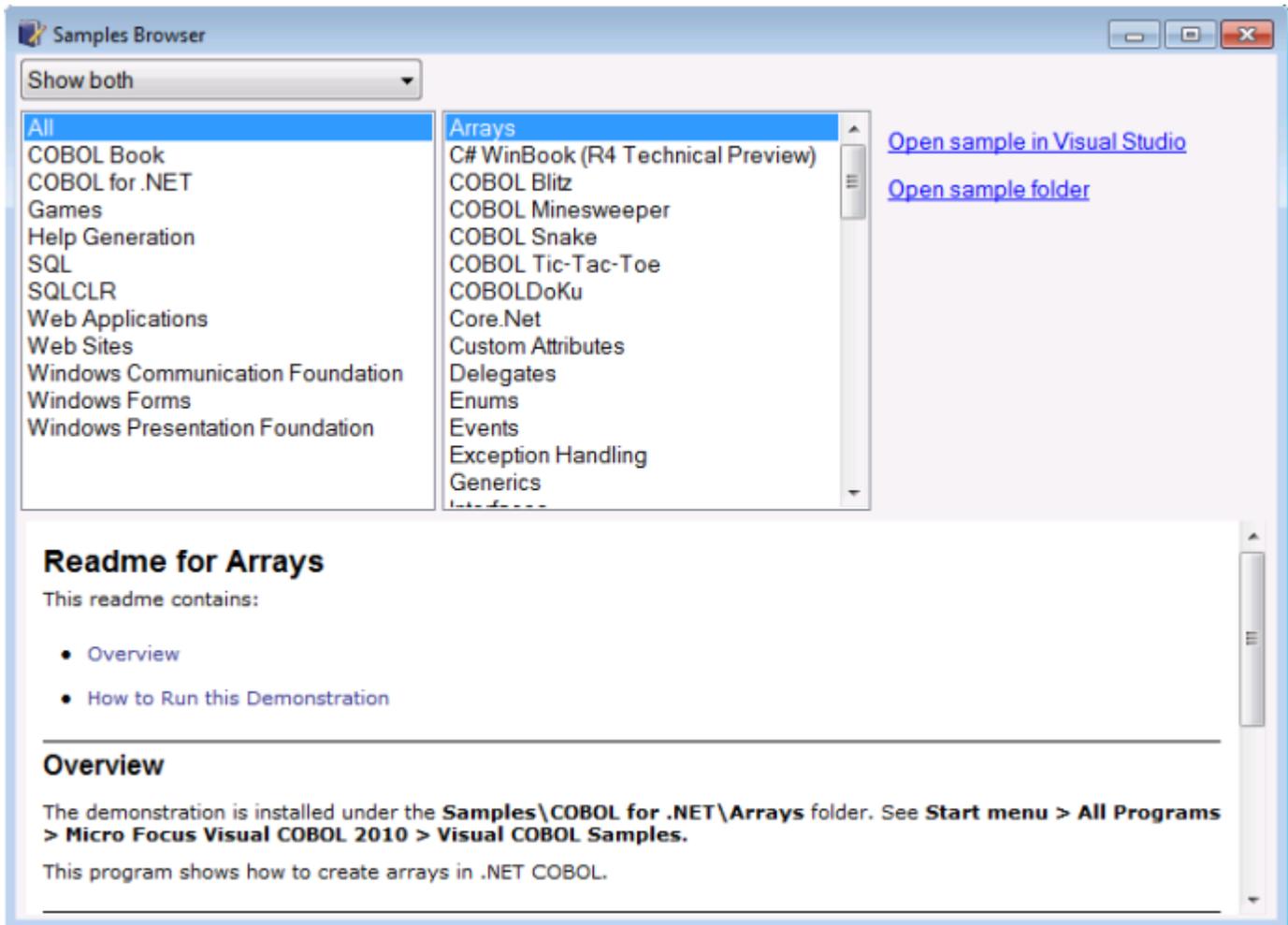
move "first" to var(1)
move "second" to var(2)
move "third" to var(3)
move "fourth" to var(4)
move "fifth" to var(5)

display var (1)
go back.

```

Samples Browser

You can preview the samples and access them more easily with the help of the Samples Browser which is now available from the Start menu. Samples Browser lists the samples by category



SQL CLR Integration Support

The new SQL CLR Integration support enables you to create stored procedures, user-defined functions and types, aggregates, and triggers in managed code, taking advantage of the Microsoft SQL Server 2008 R2 SQL CLR feature.

ACUCOBOL-GT Compatibility and RM/COBOL

The Compiler and run-time now include initial support for ACUCOBOL-GT. This support is enabled by several new Compiler directives. 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:

- Vision indexed file system and utilities (vutil, vio, and logutil) support. Vision support is enabled by the new CALLFH(ACUFH) option.
- ACUCOBOL-GT compiler options. By using the new ACUOPT directive you can specify the same options available in ACUCOBOL-GT.
- ACUCOBOL-GT and RM/COBOL data type support. This enables you to mix ACUCOBOL-GT and Micro Focus Visual COBOL applications via data files or calls.
- Initial ACUCOBOL-GT syntax support. The Compiler now supports some of the ACUCOBOL-GT extensions.
- Interoperability between ACUCOBOL-GT and Visual COBOL components. You can begin to build applications that combine ACUCOBOL-GT with Visual COBOL features.

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

XML Support

Enhancements have been made to XML Parse/Generate to provide compatibility with IBM® Enterprise COBOL for z/OS® v4.2.

New in XMLGENERATE:

- ATTRIBUTES phrase
- NAMESPACE and NAMESPACE-PREFIX phrases
- XML-DECLARATION phrase

New in XMLPARSE:

There are now two modes of XMLPARSE support using the XMLPARSE() compiler directive:

- XMLPARSE(COMPAT) – provides compatibility with IBM Enterprise COBOL for z/OS v4.1 and earlier.
- XMLPARSE(XMLSS) – provides compatibility with IBM Enterprise COBOL for z/OS v4.2.

XMLPARSE(XMLSS) provides:

- ENCODING phrase
- RETURNING NATIONAL phrase
- VALIDATING phrase
- New special registers - XML-NAMESPACE, XML-NNAMESPACE, XML-NAMEPSACE-PREFIX and XML-NNAMESPACE-PREFIX.
- New behaviors – for example, different return codes, different output registers depending on the EVENT.

Note: The ability to parse XML documents one segment at a time with the help of the END-OF-INPUT XML event is not supported yet.

Features Added in Visual COBOL 2010 R2

File Handler

The Micro Focus File Handler is now provided as both verifiable and non-verifiable versions. Compiling your application with the ILVERIFY directive will automatically reference the verifiable File Handler assembly.

Go To Procedure Division

The Go To Procedure Division button is now available on the Go To Location toolbar. Clicking the button positions the cursor on a Procedure Division depending on the current context of the code.

OpenESQL Assistant

Support for the OpenESQL Assistant has been added. The OpenESQL Assistant is an interactive tool that enables you to easily design and build SQL queries and embed those queries into your COBOL code.

Features include:

- Prototype SQL SELECT statements and test them against a database
- Design SQL INSERT, UPDATE, and DELETE statements
- Insert SQL queries into the COBOL code
- Create and insert auxiliary code into your COBOL code

Samples

The following games have been added to the samples:

- COBOL Blitz - A shooter game in which the players use a laser cannon to defend themselves against the invasion of aliens troops. The goal is to destroy the troops and prevent them from reaching the bottom of the screen.

Special Features:

- 2D graphics
- Audio effects
- Snake - An arcade game in which the player navigates a long chain of symbols across the screen and scores by collecting numbers. Numbers add to the overall length and the speed of the snake. The player needs to avoid hitting the borders of the screen or touching the snake's body as this terminates the game.
- Tic-Tac-Toe - The player competes with the PC to place three identical marks in a horizontal, vertical, or diagonal row on the 3x3 board.

Snippets

This release provides new snippets for Attribute, DateTime, Implements and for static methods.

SQL Support

The SQL technology that was present in Net Express is now seamlessly integrated within the Visual Studio 2010 development environment. When you develop COBOL SQL applications in Visual Studio, you can use the same development environment to extend and modernize your COBOL assets.

Features include:

- OpenESQL technology that supports embedded SQL in your COBOL applications
- OpenESQL Assistant wizard that automatically generates embedded SQL in a COBOL program template given basic database information
- DB/2 ECM technology that uses embedded SQL to work with DB2 LUW
- COBSQL processor that provides native DBMS SQL support for Oracle's Pro*COBOL and other vendors

XML Parse/Generate

Visual COBOL now supports the IBM-style XML syntax and enables your applications to process XML data. Support for the XML PARSE and XML GENERATE statements is provided in the Visual COBOL compiler.

Features Added in Visual COBOL 2010 R1

Visual Studio as the Core Integrated Development Environment

The Visual Studio editor has been extended in a number of ways to enhance its support for COBOL, including Standard Visual Studio 2010 features for program navigation are exploited for COBOL applications.

- Fully integrated COBOL development environment delivers high programmer productivity by exploiting Visual Studio tools and providing instant feedback.
- Enhanced COBOL syntax for .NET programmers makes it easier for COBOL programmers to use .NET services or for programmers with .NET experience in other programming languages to be productive with COBOL.
- Visual COBOL supports the development and deployment of both "managed" .NET (with multi-targeting for .NET Framework V4 and earlier versions) and "unmanaged", native code applications.
- Visual COBOL is a part of the Visual Studio 2010 product portfolio from Micro Focus which also includes testing and developer productivity tools.
- COBOL Margins - visual indication of COBOL margins which are sensitive to the COBOL margin directive currently selected for the program – if the setting is changed via an embedded "\$SET SOURCEFORMAT" directive, then the display is immediately updated.
- COBOL sensitivity is extended to support COBOL methods and data items in IntelliSense and preconfigured "code snippets" reduce the effort required to complete code and avoid errors being introduced.
- Background parsing continuously ensures that the code being worked on will compile cleanly.

COBOL 2010

Visual COBOL supports the development of both "managed" code which is fully interoperable with other .NET languages and "native code". It is built on a new Micro Focus COBOL platform "COBOL 2010".

A standalone COBOL 2010 Runtime is available for deploying applications developed within Visual COBOL.

Visual COBOL provides a test license version of the COBOL 2010 Runtime to allow system testing before deployment into production.

COBOL Language Extensions

Historically, COBOL has been case-insensitive which makes interoperation with .NET methods more difficult than it should be. For example, method or member names had to be enclosed within quotation

marks and declare synonyms to refer to external types. With Visual COBOL these restrictions have been removed and the code is more “.NET-like” while still retaining COBOL’s traditional ease of understanding. Unnecessary COBOL elements such as “repository” have been made optional which greatly reduces the size and complexity of a COBOL .NET program. The language changes improve readability and simplify the learning process for existing C# or VB programmers who can easily work on the COBOL code. With this flexibility, teams can be more agile and thus reduce development and maintenance costs.

Known Issues

Please refer to the *Known Errors and Restrictions* topic in the *Product Information* section of your product Help.

In addition, please note the following:

ASP.NET

It is not possible to run ASP.NET Web Sites or Web Services on a production machine (one that has Micro Focus COBOL 2010 Runtime installed) without an additional setup. This is because the production machine does not contain development tools such as the COBOL Compiler. To workaround this, you need to do the following steps:

1. Precompile the site before you deploy it using the **Publish Web Site** command in Visual Studio.
2. Edit the `.asmx` file of the Web service project or the `.aspx` file of the Web site and delete the `Language="COBOL"` statement.
3. Edit the `Web.config` file with a text editor and delete the line which contains: `<compiler language="COBOL" . . .`
4. Ensure that a .NET Server license is installed using Aptrack.

-1080101

Co-existing with Earlier Micro Focus Products

Creating COBOL Projects from Selection

The documentation on creating COBOL projects from selection specifies that the newly created project has the same properties and references as the original project. You should also note that the new project has the default build configuration for the selected project type.

Fileshare

A write of a record with an alternate key where the alternate key is the first value for that key may now return a 0/2 file status rather than a 0/0 status if a record with that alternate key value has been deleted by uncommitted transaction (possibly the same transaction as that performing the write).

Installation

- A bug in 64-bit Windows 7 may cause the display of the Compatibility Assistant dialog box during the installation of the product showing incorrectly that `aslmlocate.exe` and `init2aslm.exe` are not compatible. The two utilities run properly and the installation is successful. To avoid receiving this notification, run Windows update and install update KB978637 before installing this product.
- If, when you start Visual Studio, you receive a message box about a Micro Focus package load failures, check that the Packages folder is on the PATH environment variable. If it is not, click No to disable loading the packages.

If you accidentally click Yes, you need to re-enable loading the packages, as follows:

1. Open a Visual Studio command prompt from the **Start** menu.
2. Enter the command:

```
devenv /ResetSkipPkgs
```
3. Reboot the machine.

This should resolve the problem. The PATH is set up correctly and the packages are found.

- Installing this release as an upgrade to a previous version of the product might take longer compared to installing the product for the first time.

Visual Studio IDE

- In your project properties, the Application page currently allows you to select any of the static methods in the application as a Startup object. This is incorrect. You should always set only the first static method or the program name as a Startup object in order for the project to build.
- In Visual Studio, the controls on the COBOL page in the project properties are not displayed for a WPF project. To workaround this issue, open the Application page in the project properties and from the drop-down list under Output type choose the same output type that was already selected. Close the project property pages and save the project. The next time you open the project properties, the COBOL page is properly displayed.

Visual Studio Shell

The following restrictions apply if you are using the Visual Studio Shell:

- On Windows XP, when building a managed COBOL application in Visual Studio, you may receive a dialog box with error "The application failed to initialize properly (0xc0000005)". If you are running Symantec antivirus runtime protection then this might be the cause for this issue. To resolve it, [click here](#) to download a fix from Symantec.
- WCF is not supported so the WCF demonstration programs do not work and the project templates for WCF are not installed.
- There is no integrated designer for icon and bitmap files.
- The Windows SDK must be installed.
- Examples and demonstration programs that use languages other than COBOL do not work because Visual Studio Shell doesn't support any other languages. This applies to the following demonstrations:
 - CursorDemo
 - InterfacingWithStdCOBOL
 - LobDemo
 - OrderStatusDemo

Resolved Issues

The resolved issues that customers have reported are listed in this section. The numbers that follow each issue are the Reported Problem Incident number followed by the Customer Incident Numbers (in parentheses). RPIs that have numbers only (and no text) are included to confirm that the RPIs have been fixed, since no further information is required.

- [.NET Checker](#)
- [.NET RTS](#)
- [Adis](#)
- [Compiler](#)
- [Vision File System](#)

.NET Checker

- Attributes that use the TYPE OF operand produced invalid MSIL and generated an exception when accessed through reflection APIs.
1079810 (2522328)
- Using an ACCEPT on an item subsidiary to a 01 level screen section record was causing corruption of working storage data referenced in other items subsidiary to that 01 level screen section record.
1080564 (2531039)
- Certain structures of the SCREEN SECTION records could lead to generation of invalid managed code and caused the build to fail with error "COBCH0857: System error - failure during ILASM phase".
1080096 (2524955)

.NET RTS

- The .NET console processed numeric keys incorrectly when Caps Lock was enabled.
1080669 (2532347)

Adis

- An ACCEPT FUNCTION-KEY FROM ESCAPE statement called an incorrect ADIS routine and as a result the application waited for keyboard input.
1080386 (2528881)

Compiler

- The width of Shift Jis characters in the sequence area was determined incorrectly and as a result the source code did not compile in the correct area.
1080341 (2528695)
- Compiling a program containing DBCS characters produced spurious Compiler errors.
1080336 (2528637)

Vision File System

- An alternate key numbering mismatch occurred during the execution of the OPEN statement for an RM indexed file.
1079788 (2521519)

- ACUFH has been enhanced to perform key mapping between the FCD passed to ACUFH and the keys available in the indexed file. This allows programs with differing alternate key orderings to open any indexed file with sufficient matching alternate keys.

1080028 (2524660)

Updates and SupportLine

Our Web site gives up-to-date details of contact numbers and addresses.

Further Information and Product Support

Additional technical information or advice is available from several sources.

The product support pages contain a considerable amount of additional information, such as:

- The WebSync service, where you can download fixes and documentation updates.
- The Knowledge Base, a large collection of product tips and workarounds.
- Examples and Utilities, including demos and additional product documentation.

To connect, enter <http://www.microfocus.com> in your browser to go to the Micro Focus home page.



Note: Some information may be available only to customers who have maintenance agreements.

If you obtained this product directly from Micro Focus, contact us as described on the Micro Focus Web site, www.microfocus.com. If you obtained the product from another source, such as an authorized distributor, contact them for help first. If they are unable to help, contact us.

Information We Need

However you contact us, please try to include the information below, if you have it. The more information you can give, the better SupportLine can help you. But if you don't know all the answers, or you think some are irrelevant to your problem, please give whatever information you have.

- The name and version number of all products that you think might be causing a problem.
- Your computer make and model.
- Your operating system version number and details of any networking software you are using.
- The amount of memory in your computer.
- The relevant page reference or section in the documentation.
- Your Software Support Identification Number (SHIN) if you have one (not used in all countries).
- Your serial number. To find out these numbers, look in the subject line and body of your Electronic Product Delivery Notice email that you received from Micro Focus.

On Windows, if you are reporting a protection violation you might be asked to provide a dump (.dmp) file. To produce a dump file you use the **Unexpected Error** dialog box that is displayed when a protection violation occurs. Unless requested by SupportLine, leave the dump setting as *Normal* (recommended), click **Dump**, then specify a location and name for the dump file. Once the dump file has been written you can email it to SupportLine.

Alternatively, you might be asked to provide a log file created by the Consolidated Tracing Facility (CTF) - a tracing infrastructure that enables you to quickly and easily produce diagnostic information detailing the operation of a number of Micro Focus software components.

Creating Debug Files

If you encounter an error when compiling a program that requires you to contact Micro Focus technical support, your support representative might request that you provide additional debug files (as well as source and data files) to help us determine the cause of the problem.

There are three debug files that the support representative could ask you to provide. The debug files, and the directives you use to create them, are as follows:

Directive	File Created	Contains
CHKECM(CTRACE)	ecmtrace.txt	Pseudo COBOL code that shows the code generated to replace the EXEC SQL statements. This file is equivalent to output out of the IBM COBOL precompiler.
CHKECM(TRACE)	ecmtrace.txt	Detailed information as to what information is passed between the preprocessor and the Compiler. If an error occurs that generates invalid syntax, this file will be required to help isolate where the problem occurred.
DB2(CTRACE)	sqltrace.txt	Detailed list of information passed to IBM precompiler services and the results. This file is very useful if an error may involve a bug in the DB2 system software as well as the preprocessor.

Disclaimer

This software is provided "as is" without warranty of any kind. Micro Focus disclaims all warranties, either express or implied, including the warranties of merchantability and fitness for a particular purpose. In no event shall Micro Focus or its suppliers be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages, even if Micro Focus or its suppliers have been advised of the possibility of such damages. Some states do not allow the exclusion or limitation of liability for consequential or incidental damages so the foregoing limitation may not apply.

Micro Focus is a registered trademark.

Copyright © Micro Focus IP Development Limited 1984-2011. All rights reserved.

Index

.NET COBOL
syntax improvements 14

C

COBOL
.NET managed COBOL syntax improvements 14
contact information 32

D

debugging
error messages 33
downloads 32

H

Hardware Requirements 6
HCO
errors 33

I

Information Sources 32
Installation
After Installing 7

M

managed COBOL
syntax improvements 14

N

new features
.NET COBOL 12
ACUCOBOL-GT compatibility 10
ADO.NET runtime improvements 11
compiler directives 13
create project from selection 11
debugging 12
EHTML 12

large projects support 12
project details window 13
properties 13
RM/COBOL compatibility 13
smart tags 12
tutorial 13
WCF support 14
XML Extensions 14
New Features 10
New Features in the R3 Release 14
New Features in the R4 Release 10

P

product support 32
Product Support 32

R

RM/COBOL compatibility 13

S

serial number 32
Software Requirements 6
support 32
SupportLine 32
System Requirements 6

U

Uninstallation 7

W

WebSync 32
What's New 10
works order number 32

X

XML Extensions 14