



## **extend Interoperability Suite 10.0**

---

A decorative graphic consisting of several overlapping, wavy blue lines that create a sense of motion and depth, positioned in the lower half of the page.

**Release Notes**

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

Copyright © Micro Focus 2009-2015. All rights reserved.

**MICRO FOCUS**, the Micro Focus logo and extend 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.

2015-09-18

# Contents

<b>extend Release Notes</b>	<b>5</b>
<b>extend System Requirements</b>	<b>6</b>
<b>Windows Installation</b>	<b>9</b>
License Files for Windows	9
The Activator Utility	9
Changing or Updating the Windows License File	10
Installation on Windows Platforms	10
Installation Steps	10
Windows 64-bit Installations	13
BIN-REDIST and REDIST Installation Directories	13
Uninstalling or Modifying your Installation	13
Compiling Your Programs	14
Running Your Programs	15
Printing and Spooler Issues	15
Spooler Formatting	16
Direct Control	17
Printing Multiple Jobs Simultaneously	17
<b>UNIX Installation</b>	<b>19</b>
License Files for UNIX	19
The Activator Utility	19
Changing or Updating Your UNIX License File	19
Installation Under UNIX	19
Installation from FTP Site	19
Installation from CD-ROM	20
SHARED_LIBRARY_PREFIX Configuration Variable	21
Configuring Your Terminals	21
<b>What's New</b>	<b>23</b>
ECN-4252 IPv6 addressing support	23
ECN-4271 Vision 6 support	23
ECN-4279: SORT TABLE collating sequence enhancement	24
ECN-4304: Dynamic runtime tracing	24
ECN-4313: Transaction management and C\$RECOVER changes	25
ECN-4317: ActiveX and .NET run-time license auto-generation	25
ECN-4320: SYSTEM-INFORMATION now includes bittedness of the runtime	26
ECN-4324: Sort column indicator	26
ECN-4328 USE_EXECUTABLE_MEMORY	27
ECN-4336 SORT changes in the runtime	27
ECN-4339 Thin client can disappear during long file I/O	27
ECN-4341 Entry Field now supports suggestions	28
ECN-4348 Compiler can detect when verbs are terminated implicitly	29
ECN-4353 new vutil -unload option	30
ECN-4359 Setting grid cells programmatically	30
ECN-AS158 AcuServer Control Panel help	31
ECN-AC097 AcuRCL Control Panel help	31
ECN-GL545 SQL Server support for TIME data type	31
ECN-GL546 Informix support for LVARCHAR data type	32
ECN-WB917 COPY files no longer store full path names	32
ECN-WB932 AcuBench improvements	33
ECN-WB940 Font and bitmap handles that are auto-created	33
ECN-WB941 AcuBench code skeletons	34

ECN-XD101 Different character sets showing incorrectly in a query .....	34
BIS for extend enhancements .....	35
XML Extensions for extend enhancements .....	35
<b>Resolved Issues .....</b>	<b>37</b>
Acu4GL ECN List .....	37
ECN-GL544 Acu4GL for MSSQL fails to read on duplicate primary key .....	37
ECN-GL547 READ PREVIOUS fails on duplicate keys .....	37
ECN-GL548 DATETIME column sorting failure .....	38
ECN-GL549 Error 9D - Cursor already exists .....	38
ECN-GL550 Acu4GL Sybase reads on alternate keys .....	38
AcuBench ECN List .....	39
ECN-WB918 Unable to resize File Open dialog boxes .....	39
ECN-WB934: Save As dialog box changes .....	39
ECN-WB935: Create Alias is not working .....	40
ECN-WB936: AcuBench fails in Replace in Files .....	40
ECN-WB939 Graphical Vutil - Define Keys Information option .....	40
ACUCOBOL-GT ECN List .....	41
ECN-4265: Discontinued support for .NET framework 1.1 .....	41
ECN-4270: Incorrect CMD-GOTO message sent to thin client .....	41
ECN-4301: Windows Installer XML Toolset (WiX) for extend 10 .....	42
ECN-4330: DG ICOBOL file status change for READ when undefined record .....	42
ECN-4334: Threads and event procedures .....	42
ECN-4342 Accept recursion limit .....	43
ECN-4343 Large programs reported as corrupt .....	43
ECN-4344 Copy/Replacing issue with spaces .....	44
ECN-4345 COPY-SELECTION in a grid .....	44
ECN-4346 Windows CLOSE button fails .....	44
ECN-4347 Grid column and row incorrectly displayed .....	45
ECN-4349 CXML-MODIFY-ATTRIBUTE-VALUE length .....	45
ECN-4350 Writing an XML file can cause corruption .....	46
ECN-4351 TREE-VIEW control paint issues .....	46
ECN-4352 Change in floating window size .....	46
ECN-4354 C\$JAVA CJAVA-LOADCLASS not loading user-supplied classes .....	47
ECN-4355 C\$JAVA CJAVA-DBCONNECT .....	47
ECN-4356 Web thin client screen flicker .....	48
ECN-4357 C\$JAVA issues trace messages .....	48
ECN-4358 Vision file corruption in key parameters caused memory error .....	48
AcuSQL ECN List .....	49
ECN-SQL148 AcuSQL preprocessor support for long ANSI format source .....	49
AcuXDBC ECN List .....	49
ECN-XD102 Records not correct in Microsoft Access .....	49
ECN-XD103 SELECT DISTINCT returns duplicate results .....	50
<b>Updates and SupportLine .....</b>	<b>51</b>
Further Information and Product Support .....	51
Information Needed by Micro Focus SupportLine .....	51

# extend 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 links to external web sites. Micro Focus cannot be responsible for the contents of the website or for the contents of any site to which it might link. Web sites by their nature can change very rapidly and although every attempt is made to keep links up-to-date, Micro Focus cannot guarantee that external links will always work as expected.

# extend System Requirements



**Note:** This product includes software developed by the University of California, Berkeley and its contributors.

## Hardware Requirements

extend software has the following requirements:

### For Windows:

- The amount of disk space needed to install the ACUCOBOL-GT development system is typically less than 35 MB.
- AcuBench® requires at least 20 MB for installation.
- You need an additional 40 MB to install all of the other extend products.
- Use of .NET controls with the Run time and Thin Client requires .NET Framework 4.0.

### For all other platforms:

- The amount of disk space needed to install all extend products is typically less than 35 MB.

## Supported Operating Systems

For a list of the supported operating systems, check the Product Availability section on the Micro Focus SupportLine Web site: <http://supportline.microfocus.com/prodavail.aspx>.

## Additional Requirements

### AcuServer:

- Each server machine must be networked to UNIX, Linux, and Windows clients with TCP/IP. TCP/IP is not sold or supplied by Micro Focus.
- All servers must have a copy of the AcuServer license management file.
- Windows clients can run any TCP/IP software that uses a WINSOCK2 compliant `ws2_32.dll`.
- Unless you have an unlimited license for AcuServer, all UNIX servers must run the current version of `acushare`, which is included on the AcuServer distribution media.
- All servers must have a copy of the license file activated by the product installation script. This file is named `acuserve.alc`.
- Client machines must have an ACUCOBOL-GT AcuServer-enabled run-time. All Windows run-times Version 5.0 and later are AcuServer-enabled. To verify that your UNIX run-time is AcuServer-enabled, type `runcbl -v` in a Command prompt and look for this line:  

```
AcuServer client
```
- Servers being accessed by the ACUCOBOL-GT Web run-time must have a multiple-user ACUCOBOL-GT run-time license that accommodates each concurrent user that is anticipated. (If you anticipate 100 concurrent users of the Web run-time, you need a 100-user run-time license on the server in addition to the AcuServer license file. Alternatively, run-time users can install a local or network floating license for the run-time themselves.

### AcuBench:

- Intel Pentium III CPU, 300 MHz; Intel Pentium IV, 2 GHz recommended
- 128 MB of RAM recommended
- 120 MB of available hard disk space recommended
- mouse

- 800 x 600 VGA display or better; 1024 x 768 VGA display recommended

### AcuSQL:

- Your COBOL application must run on a Windows system or a UNIX system supported by Micro Focus. Unless otherwise indicated, the references to Windows in this manual denote supported Windows operating systems. Where necessary, individual versions of those operating systems are referred to by their specific version numbers.
- AcuSQL must be installed with the ACUCOBOL-GT development system on your Windows or UNIX system.
- If using a database other than Microsoft SQL Server, you must have a working ODBC level 2 API connection to your database, including any required networking software support.
- For SQL Server, if running the AcuSQL interface to Microsoft SQL Server, you must have the SQL Server client software from Microsoft. Use the Query Analyzer to see if the SQL Server client software from Microsoft is on your system. For information on opening the Query Analyzer, see the SQL Server client documentation. If the Query Analyzer opens and you are able to connect to the database, the client libraries are most likely all present. Your SQL Server data source may be hosted on one or more of the supported server operating systems.
- If you are running the AcuSQL interface to MySQL, you must have the following software:
  - MySQL 5.0 Database Server Version 5.0.18 or later (Generally Available release). Testing was done with MySQL 5.0.18 Standard.
  - MySQL Connector/ODBC Version 3.51.11 or later (Generally Available release). Testing was done with the `libmyodbc3-3.51.12.so` library. This file is available from <http://dev.mysql.com>.

You can check the version of your server by connecting using `mysql`. The version prints upon connection. For example:

```
[testing ]: mysql
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 29 to server version:
5.0.18-standard
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
```

Once in MySQL, you can also use the following:

```
mysql> select version();
+-----+
| version() |
+-----+
| 5.0.18-standard |
+-----+
1 row in set (0.09 sec)
```

- If your application accesses DB2 data, IBM's DB2 Connect™ software is recommended. Access to DB2 databases has been tested with DB2 Connect. However, any vendor's properly configured ODBC level 2 API connectivity software should work. Your DB2 data source may be hosted on one (or more) of the supported operating systems.

### Acu4GL (for ODBC) driver requirements:

Your ODBC driver must include the following functions:

- all Core ODBC driver functions
- the Level 1 function `SQLColumns`
- the Level 1 function `SQLTables`

Depending on the method of record locking you choose, your driver may also need to support some of the following function calls. See `A_ODBC_LOCK_METHOD` in the extend online help for more information.

- `SQLSetStmtOption`
- `SQLSetScrollOptions`
- `SQLExtendedFetch`

- SQLSetPos

To test the capabilities of your ODBC driver, we have included a driver test program on your Acu4GL for ODBC installation disks. You can also consult your driver documentation to ensure that it meets these requirements.

# Windows Installation

## License Files for Windows

You may request a license file for one or more users. The number of users (user limit) is set in your license agreement with Micro Focus.

When you receive your products the package includes product codes and product keys for every product you ordered. You must have the product codes and product keys to create the license file.

When you install or update your license file, place it in the appropriate directory for your version of Windows.

On Windows Vista and later, the location is:

```
C:\ProgramData\Micro Focus\extend\x.x.x\x86\product-license.alc
```

(Where *x.x.x* is the product version number.)

On Windows XP, the location is:

```
C:\Documents and Settings\All Users\Application Data\Micro Focus\extend\x.x.x\x86\product-license.alc
```



**Note:** This location is new to extend 9.0 and later, and differs from past versions of extend products. See *Appendix C - Changes Affecting Previous Versions* in the *ACUCOBOL-GT Appendices* manual for details.

When you have multiple users or products, you may copy the license file onto each machine containing the corresponding product or place the products and license file on a shared drive. Each product must be able to locate its license file in order to function.



**Note:** The ACUCOBOL-GT Transaction Server runtime license file is named `wrun32.11c`. If you are using other *extend* products, the license files must be concatenated into a special file.

## The Activator Utility

The Activator Utility automates the process of creating a license file.

During installation, select **Install License Activator** on the Installation Settings page to install the Activator Utility (`activator.exe`). It is installed in the `\AcuGT\bin` sub-directory of the installation directory.



**Note:** Always use the version of the Activator supplied with the version of the product you have installed.

To create the required license files during installation, select **Launch License Activator** on the Installation Settings page. This will launch the Activator Utility when the selected products have installed, enabling you to enter the product code and key pairs required to create the license files.

You can also create license files after the installation by running the Activator Utility (`activator.exe`) from the location detailed above or from the Start menu.

During product installation, if the Activator detects the presence of an existing license file, the extension of the existing file is changed before a new license file is created. For example, `runcbl.alc` is renamed `runcbl.al!`. If the Activator is unable to rename the existing license file, it quits with an error message, and no new license file is created. If a license file with the back-up extension already exists, the Activator

attempts to overwrite it. If that fails, the Activator quits with an error message and no new license file is created. On Windows platforms, file attributes such as Read Only are also preserved.

## Changing or Updating the Windows License File

If you need to alter your license file information, contact your Micro Focus *extend* representative for updated product code(s) and product key(s). When you receive them, launch the Activator and enter the new information as prompted.



### Note:

If you have copied the license file to a non-default directory, remember to replace that file with a copy of the updated license file.

## Installation on Windows Platforms

The Windows installation requires little interaction; the setup program copies the files into a directory you designate, or `C:\Program Files\Micro Focus\extend x.x.x` (where `x.x.x` is the version number) by default.

On 64-bit machines, 32-bit executables are installed to `C:\Program Files (x86)`.

Before you run the installation, you should ensure that you do not have another version of the extend Interoperability Suite referenced in the PATH system environment variable, as having more than one version specified may cause unexpected results.

After the installation is complete, if you have installed both the compiler and runtime, you can begin to compile and execute your COBOL programs. Basic compilation and execution techniques are described in *Compiling Your Programs* and *Running Your Programs*.



**Note:** If you move or delete any `.dll` files that have been installed, your products may not run as expected.

Before you run an installation, you should ensure that you have your product codes and product keys, and the product media to hand. Refer to the appropriate installation instructions below.

AcuBench and AcuXDBC™ must be installed locally (on the client). Server-side products such as AcuServer™, AcuXDBC™ Server and AcuConnect® are to be installed only on server machines.

## Installation Steps



**Attention:** The installation is supplied in two formats: `.exe` and `.msi`. On the installation CD, the `.exe` is located in the top level folder, and the `.msi` is located in the `msi` folder. If you plan to install Xcentrisity Business Information Server, or start AcuServer or AcuConnect from the installer, you must run the installation with administrator privileges, from an account that is in the Administrator group.

If you install from the CD, it will automatically run the `.exe` version with administrator privileges. If you run the `.msi`, it does not automatically run with administrator privileges. To run the `.msi` with administrator privileges, click **Start**, and in the **Search programs and files** field, type `cmd.exe`, then in the list displayed, right-click **cmd.exe** and select **Run as administrator**: this opens a command prompt with administrator privileges, where you can run the install by typing the full path name of the `.msi`.

Follow these steps to install your products.

1. Insert the product CD, or use the provided link to download the installation package from the Micro Focus website.

If you are using the CD, it runs the `.exe` version automatically.

2. If the `.exe` version does not start automatically, or you are using the downloaded installation package, do one of the following:

- Navigate to the CD drive, then double-click the `.exe`.
- Navigate to the location of the downloaded file, then double-click it. (See the above note if you are running the `.msi` version.)

The installation starts.

3. On the Welcome page, click **Next**.

4. On the **End-User License Agreement** page, select **I accept the terms in the License Agreement**, then click **Next**.

5. On the **Select Installation Folder** page, click **Browse** and select installation directories for 32-bit and 64-bit (if applicable) products. Alternatively, you can accept the default location(s), then click **Next**.

6. On the **Product Selection** page, select the required products, then click **Next**.

You must have product codes and keys to activate each product; however, you can install all products now, and activate those products for which you do not have a license at a later date. (Be aware that if you attempt to use products for which you do not have a license, you may receive error messages indicating that no license file is available.)



**Note:** Ensure you select **Documentation** on this page to install and access the product user guides.

7. On the **Installation Settings** page, select one or more of the following, then click **Next**:

Select	To
Install License Activator	Install a copy of the License Activator
Launch License Activator	Launch the License Activator at the end of the installation process
Start AcuServer 32/64-bit	Start your version of AcuServer as a service
Start AcuConnect 32/64-bit	Start your version of AcuConnect as a service



**Note:** Starting either AcuServer or AcuConnect as a service will only occur if their pre-requisites are satisfied before you run the installer, and only if you have run the `.exe` or `.msi` version of the installer with administrator privileges (see note above).

If you selected Launch License Activator on the **Installation Settings** page, the **Activator Wizard** appears.



8. Type your first product code and key in the appropriate fields.

The License Activator is case-insensitive and displays only uppercase characters. It also ignores embedded spaces and separating characters. Product codes and keys do not contain the letters "O" or "I".



**Caution:** If you have a license for both the Windows runtime (`wrun32.exe`) and an Alternate Terminal Manager (ATM) runtime (`run32.exe`) for the same machine, be aware that the Activator Utility creates a license file named `wrun32.alc` for each of them. To avoid a situation in which the Activator Utility overwrites the license file for the second runtime:

- Create the ATM runtime license and rename it to match the executable (change `wrun32.alc` to `run32.alc`) before creating the Windows runtime license.
- Make a backup copy of the Windows runtime license file prior to creating (and renaming) the ATM runtime license.

9. If you have more than one code and key pair to enter, select **More** after typing the first code/key pair. Repeat this process until you have entered all code and key pairs, then click **Finish**.

Each time you press **More**, the License Activator creates a separate license file for the product code and key you entered and returns you to the code and key entry screen.

10. Click **Finish** on the **Installation Complete** page to complete the installation.



**Note:** If license activation was successful, but you get a message during product startup indicating that the license file cannot be found, the license file may not be in the correct directory. The License Activator determines where to place the license file based on entries in the Windows registry. If no registry entry is found, the license file is placed in the same directory as the License Activator executable file, which is the `\AcuGT\bin` sub-directory of the default installation directory. If this is not the location of the product's executable file, move the license file to the directory containing the corresponding executable file; for example, move `wrun32.alc` to the directory containing `wrun32.exe`.

## Windows 64-bit Installations

There are 64-bit versions of most products. These 64-bit product versions install from an installation that is separate and unique from the standard 32-bit Windows version of *extend* software. The installation process follows the same steps as described in [Installation on Windows Platforms](#). Note that the 64-bit installation also installs 32-bit versions of those products that do not have 64-bit equivalents. This includes certain program executables, such as AcuBench, as well as non-executables, such as sample programs and tools.

When selecting products to install on Windows x64 platforms the 32-bit products will be installed into a special Windows folder reserved for 32-bit programs called "Program Files (x86)". The Windows **Start** menu items will use these 32-bit versions. Some of the 32-bit products have 64-bit equivalents and are automatically installed into the Windows "Program Files" directory if the 32-bit product was selected to be installed. The 64-bit "Program Files" directory only contains the executables and does not contain all of the support files. The support files are found in the "Program Files (x86)" directory along with all the 32-bit executables.

An example of a 32-bit product that does not have a 64-bit equivalent is AcuBench. When you compile and run programs from AcuBench the 32-bit versions of the compiler and runtime are used. AcuBench does not require any path changes for this to occur. All the 32-bit versions are fully supported and functional in the 64-bit environments.

When running the license activator the installer runs the 64-bit version of the Activator utility so that it can install the license files into both the 32-bit and 64-bit directories. If you run the Activator utility from the command prompt make sure you are using the 64-bit version in the "Program Files" directory because the 32-bit Activator utility will generate license files only for the 32-bit products.

## BIN-REDIST and REDIST Installation Directories

The extend Windows distribution contains two directories: `BIN-REDIST` and `REDIST`.

`REDIST` contains thin client files that should be distributed along with the thin client.

`BIN-REDIST` contains Microsoft Redistributable files. These files are required in cases where the `ACUCOBOL-GT bin` directory and runtime are placed on a shared drive and users then map to that drive. The `BIN-REDIST` directory should be placed inside the shared `bin` directory.

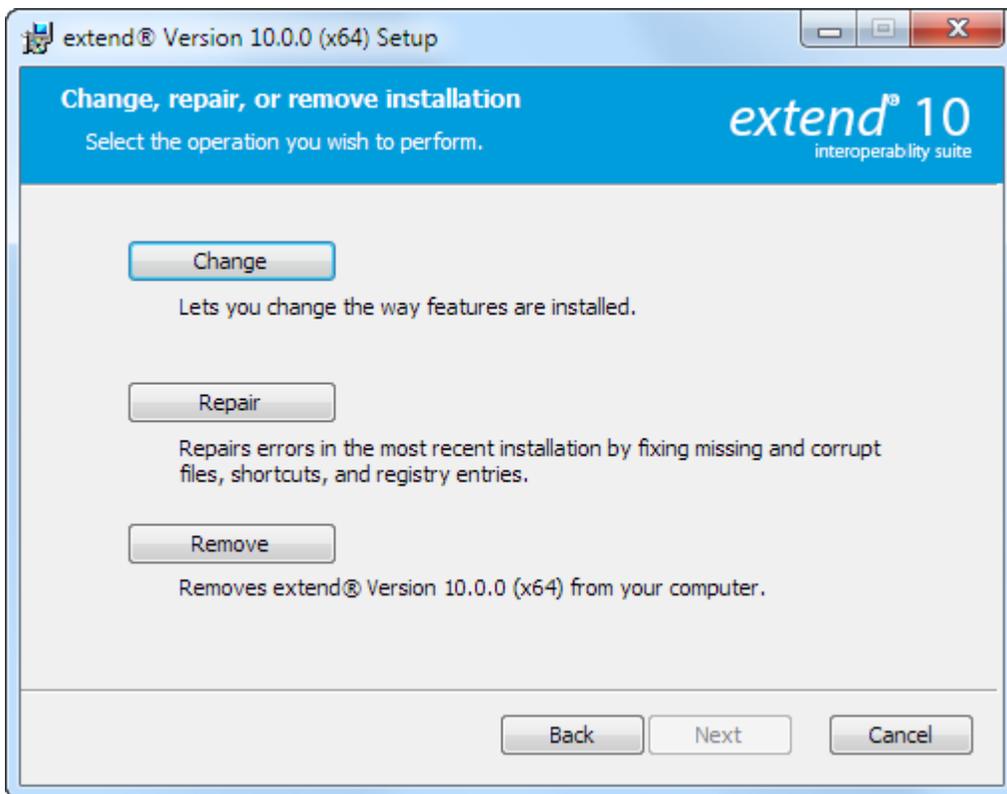
## Uninstalling or Modifying your Installation

You can uninstall or amend your current installation in one of two ways: you can run the installer again, or use the Programs and Features section in Windows. Both enable you to run the Installation Wizard, where you can perform the following program maintenance:

**Change** Enables you to add or remove products to and from your current installation. Any products that were already checked that you uncheck are uninstalled. Any additions are installed using the default installation path (`C:\Program Files\Micro Focus\extend x.x.x` - where `x.x.x` is the version number). There is no option to change to a non-default location, but you can overcome this with some products by copying the installed files from the default location to your preferred location. Note that this method will not work for AcuXDBC, AcuBench, and any server products that are registered as services.

**Repair** Enables you to reinstall the currently installed products.

**Remove** Enables you to remove all products of your installation.



## Compiling Your Programs

The ACUCOBOL-GT compiler is most easily used via AcuBench. However, it is also possible to compile from the command prompt or the Windows Start menu, to establish an association between your source files and the compiler executable, or to set up one or more icons to compile individual files. For information regarding creating file associations and setting up icons, see your Windows documentation.

When you compile, if the system returns "Bad command or file name," you did not add the location of the compiler to the PATH environment variable, or you have not rebooted since installing the software.

You can check the current definition of PATH by typing `path` and pressing **Enter** at the command prompt. The default path is `C:\Program Files\Micro Focus\extend x.x.x\AcuGT\bin\ccb132.exe` (or `c:\Program Files (x86)\...` for 64-bit installations). If you do not update PATH with this entry, you must type the full path and program name to the compiler each time you compile; alternatively, you can create a `.bat` file.

Once you have verified that the compiler is in your PATH (or you have decided that you will specify the full path every time you compile), you are ready to compile your programs. Refer to *ACUCOBOL-GT User's Guide > Compiler and Runtime > Using the Compiler* for a complete list of compile options. You can also display a complete list of options by running the compiler command:

```
ccb132 -help
```

One commonly used option is `-o`. This option is used to specify the name of the output object file. Note that ACUCOBOL-GT uses the naming convention `.acu` to indicate an ACUCOBOL-GT object file. If `-o` is not used, the compiler will name the file `source-name.acu`. If your source includes COPY files, you can copy the COPY files to your current working directory, specify their location with the `-Sp` option followed by the complete path of one or more directories, or set the `COPYPATH` environment variable. A typical compilation command might look like:

```
ccb132 -Sp c:\work\lib -o sample.cbx sample.cbl
```

## Running Your Programs

After a successful compilation, the resultant object file is ready for immediate execution (no link step is required). To run your program, make a note of the full path to your object file and return to the Windows desktop.

There are a variety of ways to run your program under Windows. The basic methods are:

- Running a command from the **Start** menu.
- Placing an icon for each program in a Program Group or folder, and then starting the program by double-clicking its icon.
- Naming your COBOL object files with a common file extension and then associating the extension with the runtime. After the association is established, you can run a program by double-clicking on the name of the file as it appears in Windows.
- Using the **wrun32** (or **crun32** if you are using the console runtime) command at the command prompt.

This document describes how to run a program from the **Start** menu only (although, the command line option is almost identical). The advantage of this method is that it requires no special setup. However, it is more time-consuming (than other methods) because you must type the command line options every time you execute the program. For greater convenience, it is best to set up a program icon: to do this, or to create a file association to the runtime, please refer to your Windows documentation.

1. Click the **Start** button.
2. On the Start menu, in the entry field, type the name of your ACUCOBOL-GT Windows runtime.

The default name is `wrun32.exe`. If you changed the name of the runtime executable, enter that name.

3. Following the name of the runtime, enter any runtime options required, followed by the path and name of the COBOL program you want to run.
4. After you have entered the complete command line, press **Enter** to execute the program.



**Note:** If you did not add the path to the `\bin` directory to the PATH environment variable (that is, `c:\Program Files\Micro Focus\extend xxx\AcuGT\bin`), you will need to specify the full path to the runtime (`c:\Program Files\Micro Focus\extend xxx\AcuGT\bin\wrun32.exe`) each time in the command (and not simply `wrun32.exe`).

You can use the COBOL configuration variable `DEFAULT_PROGRAM` to specify the name of the program to execute when no program is specified on the command line. See *Appendix H* in the ACUCOBOL-GT documentation set for more configuration information.

## Printing and Spooler Issues

ACUCOBOL-GT includes extended support for printing under Windows. In addition to the basic print spooler access procedures described below, the `WIN$PRINTER` runtime library routine provides easy access to extended Windows print spooler capabilities. See the entry for "WIN\$PRINTER" in *ACUCOBOL-GT Appendices Guide > Appendix I*. Rules for printer handling are illustrated in the *ACUCOBOL-GT User's Guide > Compiler and Runtime > Filename Interpretation > Assigning Files to Printers*.

Under Windows, you may print directly to the printer by defining `PRINTER` in the configuration file as `"-D PRN"`. Be aware that this does not prevent other programs from printing at the same time and as a result you may get intermixed pages.

You may also print using the Windows spooler, even if your reports have embedded control codes. The spooler allows many programs to create print files at the same time, and also allows the user to do other tasks while the report is being printed.



**Note:** The configuration variable WIN\_SPOOLER\_PORT allows you to divert printer output to a file or port through the Windows print spooler. For more information, see *ACUCOBOL-GT Appendices Guide > Appendix H* of the ACUCOBOL-GT manual set.

Before you assign your print file to the Windows spooler, you must decide whether you want to control the format of each page directly (with embedded control codes) or whether you want the print spooler to format the pages.

## Spooler Formatting

There are two ways to use the Windows spooler to format your print file: "-P SPOOLER" and "-Q <prntername>". See [Direct Control](#) for information on controlling the formatting yourself.

### -P SPOOLER

If you want to use the default printer and font, simply assign your print file to "-P SPOOLER". For example, to assign "PRINTER1" to the spooler, enter the following line in your COBOL configuration file ("CBLCONFI"):

```
PRINTER1 -P SPOOLER
```

By default, the runtime system assigns the "PRINTER" device to the spooler. You may change this in the configuration file by assigning "PRINTER" to some other name.

When the runtime opens a file assigned to "-P SPOOLER", it automatically initiates a job with the Windows spooler and constructs print pages in accordance with your program. The runtime uses the default printer and font. If the user looks for the job in the spooler, it is named with the current title of the ACUCOBOL-GT window.



**Note:**

The Windows spooler operates by drawing your report on each page. It constructs its own control codes to handle formatting. If you assign your print file to "-P SPOOLER" and your file contains device-dependent control sequences (such as those used to shift to a condensed font, or to print a form and then fill it in), the codes will be passed to the spooler as data and thus will not be interpreted correctly. If you have reports that depend on embedded control codes, you should print those directly to the device, or assign the print file to "-P SPOOLER-DIRECT," as described below.

### -Q <prntername>

If you want the Windows spooler to format the pages of your report, but you want to use a particular printer, assign your print file to:

```
PRINTER1 -Q \\prntername
```

in the configuration file ("CBLCONFI"). *Prntername* is the printer designation as given in the Printers folder under **Settings** in the **Start** menu. The name may be up to 80 characters long and contain embedded spaces. The name may not include the semicolon character (;) or be surrounded by single or double quotes. The pages are printed in the manner described in "-P SPOOLER", above. The sample programs "graphprn.cbl" and "prndemox.cbl" contain examples of these functions.

To determine a valid prntername, use the WIN\$PRINTER library routine to obtain the name of the desired printer. (This is described in *Appendix I* under the WINPRINT-SET-PRINTER operation code in "Specifying a Printer.") Then add the following line to your code:

```
MOVE "-Q \\prntername" TO WS-PRINTER-NAME.
```

When the runtime opens a file assigned to "-Q <prntername>" it sets the Windows print spooler to use this printer. The printer driver must be installed on the computer from which you print. If *prntername* is not recognized by the runtime, a dialog box allows you to choose a printer manually.



**Note:**

If you want to access a printer using a UNC path, you have to print directly to the printer by defining PRINTER as "-D PRN". If you use the UNC path, Windows formatting is not supported.

## Direct Control

If you want to control the format of the printout yourself using embedded control codes, simply assign your print file to "-P SPOOLER-DIRECT" or to "-Q <printername>" using the "DIRECT=ON" option. For example, to assign the print job "PRINTER1" to the spooler and retain direct control over formatting, enter the following line in your COBOL configuration file ("CBLCONF1"):

```
PRINTER1 -P SPOOLER-DIRECT
```

Or, use the following command to assign PRINTER1 to the spooler for printing to a specific printer while retaining direct formatting control:

```
PRINTER1 -Q printername;DIRECT=ON
```

Both of these methods cause the print job to be sent to the printer via the Windows spooler, but the program does not use the spooler to format the pages. You must use embedded control codes to handle formatting (much as you would under UNIX if you used the UNIX spooler).

When using the "-P SPOOLER-DIRECT" option, you may use the WIN\$PRINTER library routine to choose a printer. But because you completely control the printer, the various options provided by WIN\$PRINTER are ignored. For example, WIN\$PRINTER does not set the page size, page orientation, or font. Information returned from WIN\$PRINTER, such as number of lines and columns on the page, may not be accurate and should not be used. This subject is discussed in detail in Appendix I "*Library Routines*" of the ACUCOBOL-GT manual set.

Because some print drivers do not flush the last page, be sure to end your last page with a form-feed (for example, WRITE ... BEFORE ADVANCING PAGE). This ensures that all pages are printed. The ACUCOBOL-GT runtime ensures that no extra blank pages are printed at the end.

If you code WRITE...AFTER ADVANCING PAGE instead of WRITE...BEFORE ADVANCING PAGE, you might receive a blank last page. This is because a blank line written on the new page causes the Windows subsystem to flush the page for some print drivers. ACUCOBOL-GT ensures that entirely empty lines are not sent to the device (only the form-feed will be sent). But it is essential that:

- You have specified trailing space removal in your COBOL code (the default for print files).
- You have set the configuration option MIN-REC-SIZE to "0".

Your other option is to specify WRITE... BEFORE ADVANCING PAGE to avoid this potential problem.

If the user looks for the job in the spooler, it is named with the current title of the ACUCOBOL-GT window.

## Printing Multiple Jobs Simultaneously

If you need to print multiple jobs at the same time, you must open multiple File Descriptors that point to "-P SPOOLER" or "-P SPOOLER-DIRECT" simultaneously. For example, you may have two simultaneous print jobs:

```
SELECT FIRST-FILE
    ASSIGN TO PRINTER "-P SPOOLER" .

SELECT SECOND-FILE
    ASSIGN TO PRINTER "-P SPOOLER" .

..PROCEDURE DIVISION.

..

    OPEN OUTPUT FIRST-FILE .
    OPEN OUTPUT SECOND-FILE .
```

and both will print to the default Windows printer without interfering with each other. You can call WIN\$PRINTER USING WINPRINT-SETUP before one or both of the OPEN statements. Each file may have individual file status variables or may refer to a common file status variable.

This does not mean that you can open a single File Descriptor multiple times. For example, the following will return file status indicating that the file is already opened:

```
SELECT FIRST-FILE
      ASSIGN TO PRINTER "-P SPOOLER" .

. .

PROCEDURE DIVISION.

. .

      OPEN OUTPUT FIRST-FILE .
      OPEN OUTPUT FIRST-FILE .
```

This is normal behavior and is consistent with the way file handling is implemented in COBOL and in other programming languages.

If you are using only the verbs OPEN, CLOSE, and WRITE, no further changes to your code are needed. If you are using WIN\$PRINTER functionality (other than WINPRINT-SETUP) you will need to specify which print job is affected. This can be done in two ways:

1. The simplest way is to execute the WIN\$PRINT operation immediately after an OPEN or WRITE statement on the intended job. Every execution of OPEN and WRITE sets the current job as the default so that subsequent activity using WIN\$PRINTER is automatically directed to the job that was last accessed with an OPEN or WRITE statement.

In this situation, if you have multiple jobs running, and you close one of them, the runtime switches to the next job in the list. For example, if you are printing jobs 1, 2, and 3, and you close job 2, the close command sets the current job to 3. If there is no job 3, the runtime attempts to set to the job that preceded the closed job (which in this case is job 1). If there are no jobs, the current job is initialized.

2. The other method is to use the WINPRINT-SET-JOB operation of the WIN\$PRINTER library routine. This operation is described in *Appendix I* of the ACUCOBOL-GT manual set.

# UNIX Installation

## License Files for UNIX

UNIX users may request a license file for one or more users.

When you receive your product(s), the package includes product codes and product keys for every product you ordered. You require these product codes and product keys to create license files.

## The Activator Utility

The Activator utility automates the process of creating a license file. On UNIX platforms the Activator utility operates through a command-line interface.

By default, the Activator utility program (`activator`) is placed in the same directory as the runtime and other binary executable files.

If you did not install the Activator utility with your other products, simply copy the file onto your computer and run it as you would any other executable.

## Changing or Updating Your UNIX License File

If you need to alter your license file, contact your Micro Focus *extend* representative for updated product codes and product keys. When you receive them, launch the Activator and enter the new product codes and product keys as prompted.

## Installation Under UNIX

To install *extend* products on UNIX or Linux systems, you must have the product media, and the product codes and product keys for the products you intend to install. Your products may be delivered via FTP or CD-ROM.

## Installation from FTP Site

1. Download the product from the Micro Focus SupportLine site.
2. Using a `tar` command, extract the files in the archive to your installation directory. For example:

```
tar -xvf archive-name.tar /install/path
```



**Note:** If you install ACUCOBOL-GT as a shared object library and you don't install to the default location, you need to set an appropriate library path variable specifying the location of the shared objects. For example, on an AIX system, you would need to set the `LIBPATH` environment variable. Note that if you log in as root or a superuser, this variable must also be set in root's environment for ACUCOBOL-GT to start. Additionally, see [SHARED\\_LIBRARY\\_PREFIX configuration variable](#).

After installation, you must use the Activator utility to license the products installed.

3. Enter the following command to run the Activator utility from the installation directory:

```
/bin/activator
```

4. At the prompt, type the product code and product key pairs that came with your product package, pressing **Enter** after each pair: this updates the license file. Repeat this cycle until the code/key pairs for each product you have ordered are entered.

 **Note:** Each product searches for its license file in the same directory in which its executable resides. If you move the product's executable to a new directory, you must move its license file to the same location.

5. Enter the following on the command line to start the acushare license manager service:

```
acushare -start
```

6. Navigate to the `sample` sub-directory of your installation directory and try compiling and running the **tour** program, using the following commands:

```
ccbl tour.cbl  
runcbl tour.acu
```

7. If you get the message `Can't find entry for 'terminal' in 'term-lib'`, you need to configure your terminal for ACUCOBOL-GT. See [Configuring Your Terminals](#).
8. Once you have the sample program running, we recommend that you edit the `cblconfig` file supplied with ACUCOBOL-GT to meet the needs of your site. In particular, you should configure it to support the printers you have attached to your system.
9. If you are using shared memory, see the instructions for configuring acushare in *ACUCOBOL-GT User's Guide > Compiler and Runtime > Acushare Utility Program*.

## Installation from CD-ROM

1. Insert the CD-ROM containing your products into the appropriate drive and mount it using the appropriate `mount` command for your operating system.

 **Note:** If you are mounting a CD-ROM on HP/UX 11.0, use the following mount command:

```
mount -F cdfs -o cdcase,nosuid /dev/dev mountpoint
```

where `dev` is the CD-ROM device assignment.

2. Change to the mount directory, then type the following and press **Enter**:

```
./install
```

Follow the instructions on the screen to select the specific platform, installation directory, and products to be installed. If you select a directory that contains a previous version of *extend* products, you are prompted to view a list of files that will be overwritten, and if you want to proceed with the overwrite. If you decline, the installation stops, which enables you to go back and select a different directory.

 **Note:** For AIX Version 5.1 and later, HP-UX Version 11 and later, and Solaris Version 7.0 and later, ACUCOBOL-GT products may be distributed as shared object libraries. If you install ACUCOBOL-GT as a shared object library and you don't install to the default location, you need to set an appropriate library path variable specifying the location of the shared objects. For example, on an AIX system, you would need to set the `LIBPATH` environment variable. Note that if you log in as root or a superuser, this variable must also be set in root's environment for ACUCOBOL-GT to start. Additionally, see the topic [SHARED\\_LIBRARY\\_PREFIX configuration variable](#).

 **Note:** If you install to the default installation directory (`/opt/acucorp/xxx`), we recommend you create a symbolic link to a version-independent directory such as `/opt/acu`, to allow you to use `opt/acu/bin` in your `PATH` setting and `opt/acu/lib` in your shared library path setting.

3. After your products are installed, the Activator utility starts automatically: at the prompt, type the product code and product key pairs that came with your product package, pressing **Enter** after each pair, which updates the license file. Repeat this cycle until the code/key pairs for each product you have ordered are entered.

 **Note:** Each product searches for its license file in the same directory in which its executable resides. If you move the product's executable to a new directory, you must move its license file to the same location.

4. Enter the following on the command line to start the acushare license manager service:

```
acushare -start
```

5. Navigate to the `sample` sub-directory of your installation directory and try compiling and running the **tour** program, using the following commands:

```
ccbl tour.cbl
runcbl tour.acu
```

6. If you get the message `Can't find entry for 'terminal' in 'term-lib'`, you need to configure your terminal for ACUCOBOL-GT. See [Configuring Your Terminals](#).
7. Once you have the sample program running, we recommend that you edit the `cblconfig` file supplied with ACUCOBOL-GT to meet the needs of your site. In particular, you should configure it to support the printers you have attached to your system.
8. If you are using shared memory, see the instructions for configuring `acushare` in *ACUCOBOL-GT User's Guide > Compiler and Runtime > Acushare Utility Program*.

## SHARED\_LIBRARY\_PREFIX Configuration Variable

If you install ACUCOBOL-GT as a shared object library and you don't install to the default location, you need to set an appropriate library path variable (`LIBPATH` or `LD_LIBRARY_PATH`) specifying the location of the shared objects. The `SHARED_LIBRARY_PREFIX` variable helps the runtime find `libacInt.so` (or `libacInt.sl`) in case the `LIBPATH` (or `LD_LIBRARY_PATH`) variable is not set. `libcInt.so` (or `.sl`) is needed for `AcuServer` and `AcuConnect` support.

If a shared library name is specified without any directory information and the system call fails to load the shared library, the runtime will try to load the shared library from each of the directories specified in the `SHARED_LIBRARY_PREFIX` configuration variable.

The default value for `SHARED_LIBRARY_PREFIX` is `/opt/acucorp/xxx/lib:/opt/acu/lib`. The format of the value of `SHARED_LIBRARY_PREFIX` is the same as `FILE_PREFIX`. You can set `SHARED_LIBRARY_PREFIX` in the configuration file or environment, or programatically with the `SET` verb. Note that the runtime searches for and loads `libacInt.so` (or `libacInt.sl`) using the default value of `SHARED_LIBRARY_PREFIX`. This happens before reading the configuration file, environment, or running any COBOL code.

You can set `SHARED_LIBRARY_PREFIX` to an empty value if you do not want to use it.

Also, if the license is for `AcuTSL`, in a transactional server environment such as `CICS`, the runtime will add `/opt/acucorp/xxx/bin/runcbl.11c` and `/opt/acu/bin/runcbl.11c` to its list of license files to check. First the runtime checks `$ACUCOBOL/etc/license.acu`, then `/etc/license.acu`. If neither exists, the runtime will check `/opt/acucorp/xxx/bin/runcbl.11c` and finally `/opt/acu/bin/runcbl.11c`.

For cases in which users install ACUCOBOL-GT in the default location, `/opt/acucorp/xxx` or `/opt/acu`, and they have a license file, `runcbl.11c` in their `bin` directory, they will not need to copy the license to `/etc/license.acu`.

## Configuring Your Terminals

ACUCOBOL-GT requires data about the video environment it is running in. On Windows machines, it directly examines the hardware and configures itself appropriately. On UNIX, MPE/iX, and VMS machines, you must provide a description of the terminal you are using. This section describes briefly how to provide that information. Additional details are provided in the *Terminal Manager* section of the *ACUCOBOL-GT User's Guide*.

On systems that do not configure themselves automatically, describing the terminal to ACUCOBOL-GT involves two steps:

1. First, identify the terminal by setting the "TERM" variable.
2. Second, ensure that the terminal's characteristics are accurately described in the terminal database file.

## TERM Variable

ACUCOBOL-GT determines the type of terminal you are using by looking at the setting of the "TERM" variable. On UNIX and Linux machines, TERM is an environment variable; on VMS machines, it's a symbol. Samples for both UNIX and VMS are presented below.

TERM should be set to the name of one of the entries in the terminal database. You can examine the database file for valid names. The first field of each entry consists of a list of accepted names. Some common names are "vt100", "tv925", and "wy50" for VT100, Televideo 925, and Wyse 50 terminals, respectively.

Note that on most UNIX systems, the TERM environment variable is initialized as part of the login procedure. You will need to change this only if the name used is not one listed in the terminal database. On VMS systems, the TERM symbol defaults to "vt100" if it is not defined. Thus, you need to define the TERM symbol only if you want to use a terminal that is not VT100 compatible or if you want to use some advanced features of your terminal.

As an example, suppose you want to use a VT220 terminal. In the database, "vt220" is one of the accepted names for this type of terminal. On a VMS system, you would use the following command

```
TERM == "vt220"
```

On UNIX systems with the Bourne or Korn shell, the command would be:

```
TERM=vt200; export TERM
```

Using the C shell, the equivalent command is:

```
setenv TERM vt220
```

You may want to leave the TERM variable at its current setting to maintain compatibility with other software. If the setting is not correct for ACUCOBOL-GT, you can set the "A\_TERM" variable instead. If both the A\_TERM and TERM variables are set, ACUCOBOL-GT uses the definition of A\_TERM. This allows you to have different settings for ACUCOBOL-GT and your other software.

## Terminal Database

ACUCOBOL-GT comes with a database of terminal descriptions. On UNIX machines, this is called "a\_termcap". On VMS machines it is called "A\_TERMS.DAT". This database contains encoded descriptions of many types of terminals. You need to select the terminal type in the database that most closely matches the terminal you are using. If you need to, you can add your own entries in the database.

By default, the terminal database should reside in a pre-selected directory on your machine. On UNIX machines, this is the "/etc" directory; on VMS machines, it is the "SYS\$LIBRARY" directory. If you want to place your terminal database somewhere else, then you must define the variable "A\_TERMCAP" to be the full name of the database file. For example, on a VMS system, you could place the database in the "SYS\$LOCAL" directory with the following command:

```
A_TERMCAP == SYS$LOCAL:A_TERMS.DAT
```

On UNIX systems (using the Bourne shell), you might use the command:

```
A_TERMCAP=/usr/local/etc/a_termcap; export A_TERMCAP
```

The *ACUCOBOL-GT User's Guide* contains more information about setting up terminals and making full use of their capabilities, and selecting terminal types.

# What's New

The following items are new for this release:

## ECN-4252 IPv6 addressing support

Type of Change: Enhancement

Product: ACUCOBOL-GT

New Version: 10.0

Machines Affected: All

### Description of enhancement

extend software products that use networking have been updated to allow usage of IPv6 addressing.

You can turn on IPv6 addressing support by setting the AGS\_USE\_IPV6 configuration variable to 1. By default, this variable will be set to 0, and IPv4 addressing will be used.

No changes are required in order to use IPv6 host names, instead of IPv4 host names.

However, additional syntax may be necessary in some cases when specifying IPv6 addresses. If the IPv6 address is specified stand-alone, without any additional notation, it is sufficient to supply just the address itself. For example:

```
SERVER_IP 1111:2222:3333:4444:5555:6666:7777:8888
```

However, if additional notation is added to the IPv6 address, square brackets must be used to identify the IPv6 address. For example:

```
CODE_PREFIX *[1111:2222:3333:4444:5555:6666:7777:8888] : /usr/acu/code
```

## ECN-4271 Vision 6 support

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Vision

New Version: 10.0.0

Machines Affected: All

Known Versions Affected: N/A

### Description of enhancement

Vision 6 is a new indexed file format, similar to Vision 5, except that it uses a 64-bit offset file I/O. Where available, Vision 6 is the default, but for systems that do not have 64-bit offset file I/O, Vision 6 is unavailable, and Vision 5 is the default.

With Vision 6, maintaining multiple 2GB file segments for very large files is not required, which reduces the potential for file breakage due to mishandled file segments. It also reduces the number of OS file handles required to open a very large Vision file. This is because Vision 6 files only have one data and one index segment, each of which may grow to nearly 128TB, due to the use of 48-bit file offsets.

No special action is required to start using Vision 6. To create indexed files with an earlier file format, set the `V_VERSION` environment variable to the appropriate value.

See the extend help topic *Indexed Files - Vision* in the *ACUCOBOL-GT User's Guide* for more information.

The introduction of Vision 6 has also affected the following:

- The UNIX configuration variable `USE_LARGE_FILE_API` default setting is now ON. This is necessary to support the large file offsets of Vision 6 files.
- The UNIX configuration variable `LARGE_FILE_LIMIT_1TB` default setting is now ON for HP-UX ports only.
- `vutil -rebuild` and `vio` now have new `-6` options to select the new file format.
- The `vutil -info` output has been widened to allow for larger file segment sizes, which produces accurate node usage figures when looking at the `vutil -info -x` output.
- The `v6_sync()` Vision API function flushes files opened with either BULK-ADDITION or MASS-UPDATE when called with a non-zero parameter. In previous Vision versions, only files opened with MASS-UPDATE would be flushed.
- For record data caching of Vision 2 or 3 files, you must now set `V_NO_ASYNC_CACHE_DATA`, as well as setting `V_BUFFER_DATA`. See `V_NO_ASYNC_CACHE_DATA` for more information.

## ECN-4279: SORT TABLE collating sequence enhancement

Type of Change: Enhancement

Incidents: 2686519

RPI Number: 1093237

Product: ACUCOBOL-GT

Module: Compiler/Run time

New Version: 10.0.0

Machines Affected: all

Known Versions Affected: all

### Description of enhancement

SORT TABLE now uses the specified collating sequence rather than ignoring it.

## ECN-4304: Dynamic runtime tracing

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0.0

Machines Affected:

All Known Versions Affected: N/A

### Description of enhancement

A new method of obtaining diagnostic information about a runtime is now available.

See *Dynamic Runtime Tracing* in the *ACUCOBOL-GT Version 10.0 Run-Time Manual* for full details on how to make programs running within the runtime compatible with the tracing, and how to configure (via a new set of variables) the type of information that is traced.

## ECN-4313: Transaction management and C\$RECOVER changes

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0.0

Machines Affected: All

Known Versions Affected: All

### Description of enhancement

A number of changes have been made to transaction management and C\$RECOVER.

- It is possible to specifically exclude or include files from transaction management. Use the configuration variables LOG\_INCLUDE\_PATTERN or LOG\_EXCLUDE\_PATTERN to determine the file transactions written to the log file.
- When calling C\$RECOVER, the actions performed can be written to a report file using the configuration variable LOG\_RECOVER\_REPORT.
- The recovery report now includes more detail than was previously included in the file trace. This includes information about the location in the transaction log from which the run time is reading the current transaction log record, and whether a full record was read. This will help with diagnosing errors caused by corrupt transaction logs.



**Note:** Everything written to the recovery report is also included in the run-time trace file, if file tracing is in effect. Of course, the trace file will include much more information, from other aspects of the run time.

- C\$RECOVER now sets the special register RETURN-CODE. This will be set to 0 if C\$RECOVER executed successfully, or set to a transaction error value if unsuccessful.

## ECN-4317: ActiveX and .NET run-time license auto-generation

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: axdefgen, compiler

New Version: 10.0.0

Machines Affected: All

Known Versions Affected: All

### Description of enhancement

Some ActiveX controls have dual licenses - a development license for creating an application, and a run-time license for distributing that application. It is the job of the development tool to create and embed a run-time license when creating an application that uses such ActiveX controls, which AcuBench can do.

If you are not using AcuBench, or are trying to use 64-bit ActiveX controls (which AcuBench is unable to load), then there is no way to create a run-time license. Similarly, there are some .NET controls that have a dual-license mechanism, and AcuBench cannot create the run-time license for these.

A better way to perform the additional licensing is to use the AXDEFGEN utility, which has been enhanced to add the LICENSE-KEY to the CLASS description created in the COPY file.

Use the COPY file in your application and the compiler has been enhanced to use this license key automatically. However, if your COBOL program explicitly specifies the LICENSE-KEY phrase for the ActiveX control, it will override the automatically generated and embedded license; we recommend that you remove such instances and let the COPY file license the control.

## **ECN-4320: SYSTEM-INFORMATION now includes bittedness of the runtime**

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: runtime

New Version: 10.0.0

Machines Affected: All

Known Versions Affected: All

### **Description of enhancement**

ACCEPT ... FROM SYSTEM-INFO now includes information about whether the runtime is 32-bit or 64-bit. If the variable being accepted is 68 bytes or more, the 68th bytes will be Y if the runtime is 64-bit, and N if not.

The standard COPY file `ACUCOBOL.DEF` includes a FILLER item with a level 88, IS-64-BIT, which you can use to query this information.

## **ECN-4324: Sort column indicator**

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: 9.2.0 and later

### **Description of enhancement**

Previously, when a grid had sortable columns, there was no indication that a user had clicked a column head to sort the grid by that column.

Now, if the user clicks a sortable column, the run time draws a small triangle on the column header to indicate that the grid is sorted by that column. If the sort is currently ascending, the triangle will point down; otherwise, it will point up.

The triangle takes up a little space on the grid column header (about 14 pixels); you should make sure there is enough space on any sortable column headers to account for this triangle.

You can also configure the indicator to be an arrow (as opposed to a triangle), but triangle is the default. To change to use an arrow, set the new configuration variable GRID-SORT-INDICATOR with the value of ARROW. The value of TRIANGLE keeps the indicator as a triangle. This variable can be dynamically set and queried.

## ECN-4328 USE\_EXECUTABLE\_MEMORY

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: 8.1.0 to 9.2.x

### DESCRIPTION of enhancement

The default value for the USE\_EXECUTABLE\_MEMORY runtime configuration variable, introduced in ECN-3726, has been changed from FALSE to TRUE.

This change will allow users to run native objects on UNIX or Windows systems that have executable memory protection enabled, without having to change the value of this configuration variable.

## ECN-4336 SORT changes in the runtime

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

### DESCRIPTION of enhancement

The internals of the runtime SORT verb have been updated to provide a new mode of operation for SORT that avoids a significant amount of file I/O and may provide better performance for large SORT operations. A new method of managing the temporary files used by SORT means that fewer files are used in most cases.

The previous configuration, using the SORT\_FILES configuration variable, is no longer used by a runtime in extend 10.0 or later. Instead, use two new configuration variables A\_SORT\_REGIONS and A\_SORT\_REGIONS\_FINAL.

Another new configuration variable, A\_SORT\_FILE\_MEMORY, may be used to specify an amount of memory to be used for buffering file operations on the temporary files used in the SORT operation.

These enhancements have also resulted in the 6-byte internal minimum record size for SORT, documented in ECN-1514, to be removed for extend 10.0 or later.

## ECN-4339 Thin client can disappear during long file I/O

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Run Time

Incidents: 2808259

RPI Number: 1098232

New Version: 10.0

Machines Affected: Windows

### **Description of enhancement**

When a COBOL program performs a lot of file I/O without interacting with the user, in a thin client environment, and if that user attempts to interact with the COBOL window (by trying to activate a control, for example), the thin client could just disappear after a few minutes.

### **Instructions for use**

In general, this is correct behavior. A well-written COBOL program will allow interaction by the user at any time, and will handle it accordingly. For those misbehaving COBOL programs that don't respond to user interaction in a timely manner, there is a new command-line switch to acuthin, which instructs acuthin to wait forever for a run time when it is trying to get a response to an event (such as CMD-GOTO).

In order to use this new switch, add --notimeout to the acuthin command line, as in:

```
acuthin --notimeout server alias
```

Note that setting this switch may cause the thin client to wait for a run time that has died for some reason, without having notified the thin client. Therefore, it is better to rework your application and to not use this switch if possible, so that your application will respond correctly to user interaction.

## **ECN-4341 Entry Field now supports suggestions**

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: 10.0.0 and later

### **Description of enhancement**

The Entry Field control now supports some additional properties, which result in the user being presented with suggestions for completing the entry; these suggestions are based on previous input to that entry field.

The following new properties are available for entry fields:

#### **AUTOFILL-ID (numeric)**

This property identifies the entry field as an autofill entry field. The ID identifies the values that should be presented to the user as possible suggestions. Only previous entries from an entry field with this ID will be presented. If this property is not specified, then the other AUTOFILL properties are ignored.

Note that use of this property prevents use of NOTIFY-CHANGE, and your COBOL program will not receive any NTF-CHANGED events for any controls with this property set.

#### **AUTOFILL-FILENAME (alphanumeric)**

This property identifies the data store. The data store is where to look for suggestions, and where to store suggestions for future use. If all users have access to this file, then suggestions will be made from all users who have entered data into the entry field with this ID. If only a subset of users have access, then only

those users will get and store suggestions for entry fields with this ID. If this file does not exist, it will be created on the first use. All configuration file translating will be done on this filename, including which file system to use (e.g. MSSQL), with a default file system of VISION. If no filename is given, a filename will be constructed from *program-name* and *ID*.

If you want this data file to be a database table (for use with Acu4GL), an XFD file can be created automatically, before the file is opened, by setting the configuration variable [AUTOFILL-NEEDS-XFD TRUE](#); without this variable, the XFD is not created.

#### **AUTOFILL-MAX-LINES (numeric)**

This is the maximum number of suggestions that a user will be presented with. The default value is 10.

## **ECN-4348 Compiler can detect when verbs are terminated implicitly**

Type of Change: Enhancement

Incidents: 2817156

RPI Number: 1099132

Product: ACUCOBOL-GT

Module: Compiler

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

### **Description of enhancement**

Using ANSI-85 constructs can make COBOL programs much more readable. Unfortunately, there has not been an easy way to detect when such constructs are missing.

For example, consider the following code:

```
IF a = b
    display message box "a = b"
ELSE
    display message box "a NOT= b"
| an END-IF is here missing

display message box "Run is completed"
```

Based on the indentation, this has a different meaning than is intended.

### **Instructions for use**

The compiler can now report when an END construct is missing. In order to get this information, you must use the `-wu` compile option (check unterminated verbs). The message reported will be:

```
my-test.cbl, line 40: Warning: Implicitly terminated verb: IF found with no
END-IF.
```

The verbs that are checked for a missing end construct are ACCEPT, ADD, CALL, COMPUTE, DELETE, DISPLAY, DIVIDE, EVALUATE, IF, MODIFY, MOVE, MULTIPLY, PERFORM, READ, RECEIVE, RETURN, REWRITE, SEARCH, START, STRING, SUBTRACT, UNSTRING, USE, WAIT, WRITE, and XML GENERATE and PARSE.

Note that only complex cases are reported. It would be unhelpful to see a warning for every simple move, such as `MOVE A TO B`, for example.

## ECN-4353 new vutil -unload option

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: vutil

New Version: 10.0

Machines Affected: All

Known Versions Affected:

### Description of enhancement

A new option for `vutil -unload` allows reading the records from the source file without using the index. By default, the records are read in primary key order as they are unloaded to the destination file. In a large file, reading all the records by the primary key can be slow if the actual record data is not generally ordered by the primary key.

Specifying the `-r` sub-option to `vutil -unload` will cause the records to be read in the order in which they occur in the data file, without using the index; the order of the records in the destination file is undefined. This option is not effective when unloading Vision 2 or Vision 3 files.

### Instructions for use

Add the `-r` option to your `vutil -unload` command to read the keys directly, avoiding the index.

## ECN-4359 Setting grid cells programmatically

Type of Change: Enhancement

Incidents: 2826469

RPI Number: 1100186

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: 9.2.2 and later

### Description of enhancement

ECN 4248 in extend 9.2.2 attempted to enforce data in grid cells based on the datatype property of the cell, so that even a `MODIFY` of a cell would fail to set the data as requested. This has turned out to be inconvenient for many users.

### Instructions for use

There is a new configuration variable, `GRID_ENFORCE_DATATYPE` which, when set to the value `TRUE`, will cause a `MODIFY` statement to use valid characters for the datatype of the cell. The default value of this variable is `FALSE`, so that version 10.0 acts like version 9.2.1 and prior in this respect.

You may want to set this variable to `TRUE` if you want to make sure that your COBOL code does not do anything that a user would not be able to do.

## ECN-AS158 AcuServer Control Panel help

Type of Change: Enhancement

Product: AcuServer

Module: acuserve

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

### Description of enhancement

The AcuServer graphical control panel has a Help button that previously opened `.html` help included with the product. It now opens `.chm` help included with the product, which includes context-sensitive help links. These links mean that if you click Help from a particular section (for example, the Info tab), help for that particular section is displayed.

## ECN-AC097 AcuRCL Control Panel help

Type of Change: Enhancement

Product: ACUCOBOL-GT

Module: AcuRCL

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

### Description of enhancement

The AcuRCL graphical control panel that previously opened `.html` help included with the product. It now opens `.chm` help included with the product, which includes context-sensitive help links. These links mean that if you click Help from a particular section (for example, the Info tab), help for that particular section is displayed.

## ECN-GL545 SQL Server support for TIME data type

Type of Change: Enhancement

Product: Acu4GL

Module: MSSQL

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

### Description of enhancement

SQL Server 2012 introduced a new data type called TIME, which can efficiently store time data. Acu4GL for SQL Server now supports this data type.

### Instructions for use

In order to use this data type, you must use the correct SQL Server ODBC driver. The only version known to support the TIME data type (at the time of writing) is the SQL Server Native Client 11.0 (the driver for SQL Server 2012). This driver is installed with SQL Server 2012 or later, or you can download it from the Microsoft web site, free of charge.

To specify that a column should be a TIME column, use the DATE XFD directive, only specifying the time format characters (H, N, S, T).

See *A\_MSSQL\_ODBC\_DRIVER\_NAME Configuration Variable* for instructions for using different ODBC drivers.

## ECN-GL546 Informix support for LVARCHAR data type

Type of Change: Enhancement

RPI Number: 1098158

Product: Acu4GL

Module: Informix

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

### Description of enhancement

The maximum size of a VARCHAR field in Informix is 255 bytes. extend now supports the additional data type LVARCHAR, which can store items up to 32,739 bytes. The new data type will be used for fields larger than 255 bytes that have the VAR-LENGTH directive applied to them.

## ECN-WB917 COPY files no longer store full path names

Type of Change: Enhancement

Product: AcuBench

Module: acubench.exe

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

### Description of enhancement

When linking a COPY file into the Working-Storage Editor, AcuBench no longer stores the full path to the linked file. Previously, this could be a problem if the project was opened on another machine and the path to the COPY file was different.

For example, when linking one of the COPY files distributed with ACUCOBOL-GT, AcuBench would store a path something like:

```
C:\Program Files\Micro Focus\acucbl920\AcuGT\sample\def\opensave.def
```

If the project was opened on a 64-bit machine, the path to the COPY file was slightly different, and so caused an error:

```
C:\Program Files (x86)\Micro Focus\acucbl920\AcuGT\sample\def
```

# ECN-WB932 AcuBench improvements

Type of Change: Enhancement

Product: AcuBench

Module: acubench.exe

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: All prior versions

## Description of enhancement

There have been many improvements to AcuBench. Some of these were required due to upgrades to third-party components; others were done by request.

- When in the **Tools > Customize** dialog box, you can now press **F1** to access context-sensitive help on customizing your AcuBench toolbars.
- The ACUCOBOL keyword list, available from **Tools > Options > Code Editor > Keyword**, now contains a complete list of ACUCOBOL-GT reserved words.
- In previous releases, when a read-only file was opened in the editor, you could edit the file and save, as long as you saved with a different file name.

In the current release, when a read-only file is opened and you first attempt to edit it, a message is displayed asking you to confirm the edit of a read-only file. If you answer No, the file remains read-only, and only when you close and reopen the file do you receive this message again if you attempt to edit it. If you answer Yes, you are able to modify the file, and if you have sufficient permissions (i.e. to change the read-only permission), you can save the file with its original file name (the read-only permission is re-applied after it is saved). If you do not have sufficient permissions, you must save the file with a different file name.

- When editing an ANSI format file, any text that begins in the IDENTIFICATION area stays in that area, unless that is not possible without losing data. This is not configurable.
- Some conversion .dll files have been removed from the current installation. If you are upgrading from AcuBench 4.3 or earlier, you will need to request a temporary version of AcuBench 9.2, so that you can upgrade your project files, after which you can use AcuBench 10.0 with your projects.
- On Windows, The ACUCOBOL-GT sample programs and COPY files are now located at %PUBLIC%\Documents\Micro Focus\extend x.x.x\sample (although, there is still a copy of these files at their previous location - the extend installation directory). If your project files use the COPYPATH environment variable to specify the previous location (using %acudir%), the variable will be automatically modified to use the new location (%sampledir%) when they are loaded for the first time.

# ECN-WB940 Font and bitmap handles that are auto-created

Type of Change: Enhancement

Product: AcuBench

Module: Acubench.exe

New Version: 10.0

Machines Affected: All

## Description of enhancement:

Many situations cause AcuBench to generate code that loads a bitmap or creates a font. In order to not leak resources, it was necessary for the COBOL programmer to hand-code lines that will destroy these handles. Now, AcuBench generates and performs paragraphs that destroy any fonts it created or bitmaps it loaded, when a program exits.

## ECN-WB941 AcuBench code skeletons

Type of Change: Enhancement

Product: AcuBench

Module: acubench.exe

New Version: 10.0

Machines Affected: All

### Description of enhancement:

It is now possible to modify the skeletons used in AcuBench to generate your COBOL code.

### Instructions for use



**Important:** It is very easy to make modifications in such a way that AcuBench will no longer be able to generate COBOL code correctly, so we advise you to make a backup copy of `AcuBench.Bin` before attempting this procedure.

Firstly, extract all of the skeletons from `AcuBench.Bin` by executing `acubench` with a new `/extract` option:

```
acubench /extract AcuBench.Bin
```

This results in a number of `.asp` files, which can be modified. After modifying them, you need to rebundle the files back into `AcuBench.Bin`, with a new `/bundle` option:

```
acubench /bundle AcuBench.Bin
```

Also, we are not providing information about the syntax of these files, except to say that there are two types of information in them: literal text and keywords. Keywords are surrounded by at signs (`@`), these are meant to be (mostly) self-documenting, and misspelling them can cause severe problems. Literal text will be copied directly from the skeletons to the generated COBOL programs (with possible changes in the spacing of the text).

## ECN-XD101 Different character sets showing incorrectly in a query

Type of Change: Enhancement

Product: AcuXDBC

Module: AcuXDBC

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

### Description of enhancement

Previously, if an application such as Microsoft Query attempts to read data files containing OEM characters, the data would not display correctly.

A new configuration variable has been introduced that will translate entered data to and from the ANSI and the OEM character sets.

OEM\_CHARACTER\_SET

Setting this variable to ON TRUE, or YES causes XDBC to call a function that will translate the ANSI character set to the OEM character set. Data being read will be translated from the OEM character set to the ANSI character set.

## BIS for extend enhancements

The following enhancements have been introduced in this release of BIS for extend:

- **This version no longer requires IIS 6 metabase compatibility**

Previous versions of BIS required that, for IIS 7 and later, **IIS 6 Metabase Compatibility** be enabled when IIS is installed. This is because the installer and the `BISMkDir` program used the `Active Directory Services Interface (ADSI)` to configure IIS and create virtual directories.

Starting with extend 10, BIS includes a new program called `BISMkApp`, which natively creates IIS applications without using ADSI. This means that **IIS 6 Metabase Compatibility** is no longer required, and other problems that were caused by the IIS 6 metabase translation layer, such as the IIS configuration files becoming larger with each installation or configuration, are no longer an issue.

Both `BISMkApp` and `BISMkDir` are installed with BIS for extend, and both provide similar functionality, which may be used if IIS 6 metabase compatibility is installed on IIS 7 and later. However, `BISMkDir` is deprecated and will be removed in a future release.



**Note:** IIS 7 was introduced with Windows Vista and Windows Server 2008.

- **Apache 2.4 support**

BIS for extend 10 now supports Apache 2.4 on UNIX, but is limited to an Apache 2.4 linked with the Prefork MPM. Support for Apache 2.2 continues.

## XML Extensions for extend enhancements

The following enhancements have been introduced in this release of XML Extensions for extend:

- **Temporary files during statement execution**

XML Extensions previously utilised temporary files during the execution of the `XML EXPORT TEXT`, `XML IMPORT TEXT`, `XML TEST WELLFORMED-TEXT` and `XML VALIDATE TEXT` statements. These temporary files are no longer used, as the operation of these statements is entirely in memory (except for the external stylesheet input file, if one is specified).

- **The XML TRANSFORM TEXT statement**

A new statement `XML TRANSFORM TEXT` has been added that transforms a document in memory to another document in memory.

The three parameters for this statement are the input document pointer, the XSLT stylesheet file name, and the output document pointer, in that order - the `lixmldef.cpy COPY` file briefly documents this new statement in its comments. No temporary files are written during the operation of the `XML TRANSFORM TEXT` statement. The memory area for the output document is allocated by the `XML TRANSFORM TEXT` statement and must be freed by using the `XML FREE TEXT` statement.

- **Support for raw output of XML special characters**

XML Extensions no longer supports raw output of XML special characters from FILLER data items; XML special characters in FILLER items are now escaped as character references. This was a necessary compromise in the elimination of a temporary file for the `XML EXPORT TEXT` statement.

XSLT stylesheets are a better method of inserting markup text into a result document. The verify sample provided with BIS has been modified from using XML special characters in FILLER to using a stylesheet and is instructive in showing the improved power of using a stylesheet over using XML special characters in FILLER.

Associated with this removal of raw XML output, the output of a document prefix feature in XML EXPORT FILE and XML EXPORT TEXT was also removed.

# Resolved Issues

The following are resolved issues for the extend products.

## Acu4GL ECN List

This section includes the ECNs relating to Acu4GL:

### **ECN-GL544 Acu4GL for MSSQL fails to read on duplicate primary key**

Type of Change: Correction

Incidents: 2816221

RPI Number: 1098845

Product: Acu4GL

Module: MSSQL

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

#### **Description of problem**

A recent feature of ACUCOBOL-GT is to allow duplicate primary keys. This is not recommended, but it was implemented for compatibility with another COBOL system.

This feature now works when using Acu4GL for SQL Server; previously, this feature returned an error whenever you tried to READ a record.

### **ECN-GL547 READ PREVIOUS fails on duplicate keys**

Type of Change: Correction

RPI Number: 1099197

Product: Acu4GL

Module: Oracle

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

#### **Description of problem**

The correct record is now returned when the program changes direction whilst reading alternate keys that allow duplicates.

## ECN-GL548 DATETIME column sorting failure

Type of Change: Correction

Incidents: 2807387

RPI Number: 1099527

Product: Acu4GL

Module: MSSQL

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: 9.2.0 and later

### Description of problem

When you use a non-default ODBC driver (and in particular, SQL Server Native Client 11.0), and a date is part of a key, then executing START LESS on HIGH-VALUES no longer generates the following error:

```
22008 0 - Datetime field overflow. Fractional second precision exceeds the scale specified in the parameter binding.
```

## ECN-GL549 Error 9D - Cursor already exists

Type of Change: Correction

Incidents: 2807691

RPI Number: 1099428

Product: Acu4GL

Module: MSSQL

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: 9.2.2 and later

### Description of problem

Acu4GL for MSSQL no longer produces the following error when reading to the end of a file and then reading previous records:

```
9D - A cursor with that name already exists
```

## ECN-GL550 Acu4GL Sybase reads on alternate keys

Type of Change: Correction

Product: Acu4GL

Module: Sybase

RPI Number: 1100232

New Version: 10.0

Machines Affected: Windows

### Description of problem

A program performing a START and READ NEXT operation on an alternate key with duplicates now returns the records in the correct order, based on the alternate key. Previously, when the configuration variable A\_SYB\_FORCED\_INDEX was set, the records were returned in order based on the primary key.

## AcuBench ECN List

This section includes the ECNs relating to AcuBench:

### ECN-WB918 Unable to resize File Open dialog boxes

Type of Change: Correction

Incidents: 2647228

RPI Number: 1089374

Product: AcuBench

Module: acubench.exe

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: 9.0.0 and later

#### Description of problem

A recent change has meant that within AcuBench, you could not resize the standard Microsoft File Open dialog box; therefore, AcuBench now uses the Explorer-style File Open dialog box that can be found in Windows Vista, and later.

### ECN-WB934: Save As dialog box changes

Type of Change: Correction

Incidents: 2793859

RPI Number: 1096140

Date: 2014-10-01

Product: AcuBench

Module: AcuBench.exe

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: 8.0.0 and later

#### Description of problem

The **Save As** dialog no longer displays some strange behavior when dealing with file extensions that do not match known extensions, or files that have no extension at all. Previously, this was evident, especially when changing the filter to include "All Files (\*.\*)".

#### Instructions for use

Version 10 has a new feature in which files with unknown extensions are treated as COBOL or not, depending on the response to a new prompt. Once answered, files with this extension will always be treated as either COBOL or not, depending on the answer to the first prompt.

With files that have no extension, you will be prompted each time. If a file is to be treated as a COBOL file, then the **Save As** dialog box will default to use the `.cbl` extension for the file, appending that extension to the full original filename. Selecting a different filter will change that extension, but selecting the **All Files (\*.\*)** filter will result in the original filename being used. (Note that the **Save As** dialog box will not allow you to overwrite the file with the same name; use the Save function for that.) If a file is to be treated as non-COBOL, then the **Save As** dialog box defaults to use the **All Files (\*.\*)** filter, and leave the file name as it is. Again, you will not be able to save the file without giving it a different name.

## ECN-WB935: Create Alias is not working

Type of Change: Correction

RPI Number: 606437

Product: AcuBench

Module: acubench.exe

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: 9.2.0 and later

### Description of problem

When a project is using thin client mode, you can now create an alias from within AcuBench. Previously, you could not; in fact, when looking at Project Properties, it was not possible to test the server/port/ directory.

## ECN-WB936: AcuBench fails in Replace in Files

Type of Change: Correction

Incidents: 2800306

RPI Number: 1096830

Product: AcuBench

Module: AcuBench

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: All known

### Description of problem

The Replace in Files functions no longer fail in AcuBench if one of the files modified by a function is open in AcuBench. Previously, AcuBench would prompt to reload the modified file, but because of the wait cursor created by the Replace in Files function, you were unable to interact with the prompt dialog box.

## ECN-WB939 Graphical Vutil - Define Keys Information option

Type of Change: Correction

RPI Number: 612268

Date: 2015-06-16

Product: AcuBench

Module: WVutil

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: 9.2.0 and later

#### **Description of problem**

The correct screen is now shown when you click **Define Key Information** in the Create File tab of the graphical Vutil application. Previously, clicking this option would do nothing.

## **ACUCOBOL-GT ECN List**

This section includes the ECNs relating to ACUCOBOL-GT:

### **ECN-4265: Discontinued support for .NET framework 1.1**

Type of Change: Correction

Product: ACUCOBOL-GT

Module: Compiler

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: 7.0 and later

#### **Description of problem**

Version 7.0 of ACUCOBOL-GT introduced support for creating a .NET .dll or .exe.

Support started with the then-current .NET Framework version 1.1. ACUCOBOL-GT Version 8.0 introduced support for .NET 2.0; this support is made possible by the use of the .NET module `wrunnet.dll`. It has been increasingly difficult to maintain compatibility with .NET 1.1 in this module; and so, in extend 10.0, .NET 1.1 support is dropped, allowing for improvements to `wrunnet.dll`.

The compiler no longer accepts the `:1.1` modifier to the `--netdll`, `--netlld`, or `--netexe` options. It now supports `:2.0`, `:3.0`, `:3.5`, and `:4.0`; although, all of these are equivalent to using `:2.0`, or using no modifier at all, (since the applicable parts of the .NET framework did not change between these versions).

### **ECN-4270: Incorrect CMD-GOTO message sent to thin client**

Type of Change: Correction

Incidents: 2673484

RPI Number: 1092380

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: 8.1.3

### **Description of problem**

Logic errors no longer occur when attempting to enter data into a grid cell.

Previously, in some situations, a spurious CMD-GOTO would get sent to the COBOL program, causing the logic errors. This was seemingly specific to extend 8.1.3, and occurred when the COBOL program performed extra logic (with user interaction) during a MSG\_FINISH\_ENTRY event, and occurred when attempting to enter data into a grid cell.

## **ECN-4301: Windows Installer XML Toolset (WiX) for extend 10**

Type of Change: Correction

Incidents: None

RPI Number: None

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0.0

Machines Affected: Windows

Known Versions Affected: None

### **Description of problem**

Visual Studio 2012 no longer supports the Visual Studio "setup" projects available in previous versions. As some of the extend Interoperability Suite installers used these projects, the extend 10 installation has now switched to using WiX installers.

## **ECN-4330: DG ICOBOL file status change for READ when undefined record**

Type of Change: Correction

Incidents: 2806405

RPI Number: 1097532

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0.0

Machines Affected: All

Known Versions Affected: All

### **Description of problem**

After a failed START, leaving the file position undefined, a READ operation now returns the documented DG file status of 92, and not DG file status 10, as it was doing previously.

If you require the previous behavior, set the configuration variable: `FILE-CONDITION 27=10`

## **ECN-4334: Threads and event procedures**

Type of Change: Correction

Incidents: Many

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0.0

Machines Affected: All

Known Versions Affected: 8.1.0 and later

### **Description of problem**

Previously, ECN-3671 introduced a change that causes the run time to automatically lock a thread in response to certain events, in particular PAGE events for grids and lists. After much consideration, we have determined that this behavior is wrong, and are removing it.

The default value of the configuration variable that enabled this behavior has changed from TRUE to FALSE, thus turning off the change. If you need to continue using this change, you can do so by setting ECN-3671 to TRUE.

A better solution (than ECN-3671=TRUE) is to wrap any event procedures that handle PAGE events with a thread lock. For technical reasons, this can not be done in the run time, evidenced by the fact that more bugs resulted from this change being on, than were caused by having this change turned off.

## **ECN-4342 Accept recursion limit**

Type of Change: Correction

Incidents: 2811181

RPI Number: 1098239

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: 9.2.4 and later

### **Description of problem**

The run time no longer fails with the above error when processing an ACCEPT statement. Previously, if a thread sent a message to another thread, but the receiving thread did not execute a statement that would receive the message, the run time would fail on the next ACCEPT statement.

## **ECN-4343 Large programs reported as corrupt**

Type of Change: Correction

Incidents: 2814467

RPI Number: 1098721

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

### **Description of problem**

The run time no longer produces a "Corrupt Program" message when it attempts to access a large file, with large records (such as when it is loading a COBOL object that is very large). It will now read the file correctly.

## ECN-4344 Copy/Replacing issue with spaces

Type of Change: Correction

Incidents: 2815906

RPI Number: 1098839

Product: ACUCOBOL-GT

Module: Compiler

New Version: 10.0

Machines Affected: All

Known Versions Affected:

### Description of problem

You can now compile a program that uses COPY/REPLACE to replace a string with zero or more spaces.

Previously, the compiler would treat the spaces in the COPY file as non-existent; for example, trying to replace `==(S)==` with `== ==` in a file with a line:

```
03 W(S) PIC X.
```

would result in a compilation failure, because not only the (S), but all following spaces would be replaced, resulting in :

```
03 WPIC X.
```

which of course is not valid, and what caused the compilation error.

## ECN-4345 COPY-SELECTION in a grid

Type of Change: Correction

Incidents: 2504792

RPI Number: 1082754

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

### Description of problem

Format 13 of the SET statement, which allows certain exceptions to be assigned to various actions, such as copying the current selection to the clipboard, when your COBOL program is executing an ACCEPT statement that has an ON EXCEPTION clause, now works correctly when the active control is a grid.

## ECN-4346 Windows CLOSE button fails

Type of Change: Correction

Incidents: 2807799

RPI Number: 1098696

Product: ACUCOBOL-GT

Module: Run Time 9.2.4 and later

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: 9.2.4 and later

#### **Description of problem**

The Windows CLOSE button now operates correctly when clicked. Previously, if a COBOL program reacted to the CLOSE button, but failed to shut down, and failed to set EVENT-ACTION to EVENT-ACTION-FAIL, the run time would not send MSG-CLOSE messages to the COBOL program, and this could cause the COBOL program to misbehave.

Note that the best practice is to close the current window when a user clicks the CLOSE button.

## **ECN-4347 Grid column and row incorrectly displayed**

Type of Change: Correction

Incidents: 2813537

RPI Number: 1098527

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: 9.2.3 and later

#### **Description of problem**

ECN-4310 attempted to correct an error when setting cell headers in a grid when the column type was numeric. This ECN amends that fix, by now taking into account the possibility of the number of column headers being larger than 1.

## **ECN-4349 CXML-MODIFY-ATTRIBUTE-VALUE length**

Type of Change: Correction

Incidents: 2819194

RPI Number: 1099242

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

#### **Description of problem**

When modifying an attribute using C\$XML op-code CXML-MODIFY-ATTRIBUTE-VALUE, the length of the attribute passed as the fourth parameter is no longer ignored. Previously, it was, causing the resulting XML file to have incorrect data.

## **ECN-4350 Writing an XML file can cause corruption**

Type of Change: Correction

Incidents: 2819535

RPI Number: 1099282

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

### **Description of problem**

When writing an XML file (using any of the XML technologies in ACUCOBOL-GT, such as C\$XML or AcuXML), processing instructions (such as xml-stylesheet) are now written correctly. Previously, they were not, resulting in corrupt XML files.

## **ECN-4351 TREE-VIEW control paint issues**

Type of Change: Correction

Incidents: 2819369

RPI Number: 1099315

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

### **Description of problem**

Strange behavior that was noticed during a TREE-VIEW reset-list action, responding to TV-SELCHANGE and TV-SELCHANGING messages has been corrected. We noticed this in version 5.2 of the thin client, and in version 9.2 when using native controls.

### **Instructions for use:**

No action is needed. By default, in both thin client and the local run time, when resetting the list of items, the run time will not send any notifications to the COBOL program until the RESET is complete.

You can turn off this new behavior, for example, if you notice that there are TREE-VIEW events that you are no longer receiving during a RESET, by setting the configuration variable BLOCK-TV-MSG-DURING-RESET to FALSE.

## **ECN-4352 Change in floating window size**

Type of Change: Correction

Incidents: 2821147

RPI Number: 1099502

Product: ACUCOBOL-GT

Module: Web thin client, Web Run Time

New Version: 10.0

Machines Affected: All

Known Versions Affected:

#### **Description of problem**

When running under Internet Explorer 11.0, the web thin client now displays its windows at the correct size. In the previous release, it would display slightly smaller than it did when run under previous releases.

#### **Instructions for use**

The calculation of size is configurable: you can turn off this change by setting the configuration variable CHECK-WINDOW-SIZE to FALSE (default is TRUE).

## **ECN-4354 C\$JAVA CJAVA-LOADCLASS not loading user-supplied classes**

Type of Change: Correction

Incidents: 2822696

RPI Number: 1099878

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

#### **Description of problem**

A call to C\$JAVA using the CJAVA-LOADCLASS op-code now loads user-supplied classes.

## **ECN-4355 C\$JAVA CJAVA-DBCONNECT**

Type of Change: Correction

RPI Number: 1100184

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

#### **Description of problem**

The C\$JAVA CJAVA-DBCONNECT op-code now calls DriverManager.getConnection() using only one parameter.

Previously, it used three parameters, the last two of which were empty strings. This prevented MySQL from using the user and password specified in the connection string.

If you do not want to use the new behavior, you can turn it off by setting the configuration variable ECN-4355 to 0.

## ECN-4356 Web thin client screen flicker

Type of Change: Correction

RPI: 1099993

Product: ACUCOBOL-GT

Module: Run Time

New Version: 10.0

Machines Affected: All

### Description of problem

ECN-4273 fixed a problem where the Web thin client screen was occasionally not getting updated in some cases; in some cases this fix introduced some screen flickering. This ECN enables you to turn off that previous fix if you find the screen flickering unacceptable.

### Instructions for use

Set the ECN-4273 configuration variable to OFF, which will remove the fix applied by ECN-4273.

## ECN-4357 C\$JAVA issues trace messages

Type of Change: Correction

Product: ACUCOBOL-GT

Module: Run Time

RPI : 1100291

New Version: 10.0

Machines Affected: All

### Description of problem

The following trace statements for the C\$JAVA interface are now only issued if A\_TRACE is set to a value greater than zero:

```
linkage signature: ...
```

```
linkage section string length: ...
```

## ECN-4358 Vision file corruption in key parameters caused memory error

Type of Change: Correction

Product: ACUCOBOL-GT

Module: Vision

New Version: 10.0

Machines Affected: All

### Description of problem

File corruption causing an unexpected value to appear in the key parameter section of the Vision header during an OPEN operation, has been fixed. Previously, this caused a memory error, but now the invalid

value is detected, and a file broken status is returned. The secondary code for this status is 23 - An invalid value was found while reading the key parameters in the header.

## AcuSQL ECN List

This section includes the ECNs relating to AcuSQL:

### **ECN-SQL148 AcuSQL preprocessor support for long ANSI format source**

Type of Change: Correction

Incidents: 2807907

RPI Number: 1097698

Product: AcuSQL

Module: AcuSQL

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

#### **Description of problem**

Support for long ANSI format source code (compiled with the `-s1` option) has been added to the AcuSQL preprocessor. Previously, the lack of this preprocessor support may have caused a crash in the compiler.

#### **Instructions for use**

Use the `-s1` option from the compiler as usual. A matching `-s1` option has been added to the AcuSQL preprocessor for stand-alone execution.

## AcuXDBC ECN List

This section includes the ECNs relating to AcuXDBC:

### **ECN-XD102 Records not correct in Microsoft Access**

Type of Change: Correction

Product: AcuXDBC

Module: AcuXDBC

RPI Number: 1098184

New Version: 10.0

Machines Affected: Windows

Known Versions Affected: All

#### **Description of problem**

Microsoft Access no longer returns incorrect data for a file that has a key containing a signed numeric field.

Previously, incorrect data was returned due to the way that COBOL treats the sign on numbers. With numeric fields, unless you specify otherwise, the sign gets encoded into the number. In a particular

instance, there were two numeric fields in a record - one was part of a key and the other one is not. These fields contain the same values. If you selected the field that is not part of the key, things worked as expected. If you selected the field that is part of the key, you did not get the expected results.

If you read the program in COBOL, on is something like 114, you would see:

```
00000011500  
0000001150}
```

and you would convert the second number which would come back as -115.00.

The AcuXDBC interface expects the numbers to come back sorted when reading on an index, so it stops reading.

The Access data engine will query the database catalog to find a unique index. It will then build up a sequence of queries involving this primary key. If the scanning stops due to the encoding of the numeric fields, the result set will be incorrect.

The utilities xdbcutil and mfdbcutil will now issue a warning of this situation with an information message such as: "Caution: Index BSP0900 non-unique, field P09\_POSNR is NumSigned" and the key will be flagged as non-unique in the database.

## **ECN-XD103 SELECT DISTINCT returns duplicate results**

Type of Change: Correction

RPI Number: 1099803

Product: AcuXDBC

Module: AcuXDBC

New Version: 10.0

Machines Affected: All

Known Versions Affected: All

### **Description of problem**

Executing a query that contains a SELECT DISTINCT, including a WHERE clause, now returns the correct results. Previously, the DISTINCT keyword was being dropped, resulting in an incorrect number of rows being returned.

# 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](http://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 Needed by Micro Focus SupportLine

When contacting Micro Focus SupportLine, please include the following information if possible. The more information you can give, the better Micro Focus SupportLine can help you.

- The name and version number of all products that you think might be causing an issue.
- Your computer make and model.
- System information such as operating system name and version, processors, and memory details.
- Any detailed description of the issue, including steps to reproduce the issue.
- Exact wording of any error messages involved.
- Your serial number or works order (WO) 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, or on the box in which the product was supplied, and on the red card supplied in the DVD case .

# Index

-P spooler 16  
-Q spooler 16

64-bit Installations 13

## A

A\_TERMCAP  
    database of terminal descriptions 22  
A\_TERMS.DAT  
    database of terminal descriptions for VMS 22  
Activator  
    back-up of existing license file 9  
    installing on UNIX 19  
    launching in Windows 9

## B

BIN-REDIST 13

## C

changing license files on UNIX 19  
compiling  
    under Windows 14  
configuration variables  
    SHARED\_LIBRARY\_PREFIX 21  
configuring terminals 21  
contact information 51  
Customer Care 51

## D

default directory  
    License Activator 9  
documentation  
    installing 11  
downloads 51

## F

file extensions  
    and execution under Windows 15  
flushing last page of printout 17  
font  
    condensed 15  
format of printout  
    controlling 17

## I

installation  
    configuring terminals 21  
    from Windows CD-ROM 10  
    UNIX CD-ROM 20  
installing ACUCOBOL-GT as a shared object library 20, 21

## L

launching the Activator (Windows) 9  
library path variable for a shared object library 20, 21  
License Activator  
    adding product code and key 9  
    default directory 9  
license files  
    Activator back-up of existing file 9  
    changing or updating on Windows 10  
    concatenating into special file 9  
    installing (activating) 9  
    UNIX 19  
    Windows 9

## M

multiple print jobs 17  
multiple-user license file management 19

## P

PATH environment variable 10  
print file  
    formatting with the spooler 16  
print format  
    controlling 17  
printing  
    and spooler issues with Windows 15  
    multiple jobs at once 17  
    problems with device-dependent control sequences 15  
product support 51  
Product Support 51

## R

REDIST 13  
reinstalling products 13  
running programs  
    Windows 15  
runtime execution for Windows 15

## S

serial number 51  
shared memory and acushare 20, 21  
shared object library 20, 21  
SHARED\_LIBRARY\_PREFIX configuration variable 21  
spooler  
    and Windows 15  
    using to format print files 16  
starting the Activator (Windows) 9  
SupportLine 51

## T

- TERM variable 22
- terminal database 22
- terminal manager
  - installing 21
- troubleshooting
  - compile errors 14
  - printout, flushing last page of 17

## U

- uninstalling products 13
- UNIX
  - installing the Activator 19
  - license files 19
  - updating license files on 19

## V

- Vision file system

## W

- WebSync 51
- WIN\$PRINTER library routine 17
- Windows
  - executing programs 15
  - launching the Activator 9
  - printer and spooler issues 15
- Windows icons
  - using to execute programs 15
- Windows product installation 10
- Windows spooler
  - using to format print files 16
- works order number 51