MICRO FOCUS

extend Interoperability Suite 10.4.0

Release Notes

Contents

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 runtime 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

> **Note:** AcuBench no longer supports Windows XP or Windows Server 2003 (or any Windows platforms that pre-date these). This is because AcuBench utilizes a number of third-party libraries that have ceased to be supported on those platforms.

For this release, if you are running on the AIX 7.1 platform, the minimum requirement is version 7.1 Technology Level 4 (7100-04)), and if you are running on the AIX 7.2 platform, the minimum requirement is version 7.2 SP1.

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

Additional Requirements

General requirements

Linux-based platforms require `glibc` version 2.5 or later.

AcuServer:

- Each server machine must be networked to UNIX, Linux, or 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 runtime. All Windows runtimes Version 5.0 and later are AcuServer-enabled. To verify that your UNIX runtime 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 runtime must have a multiple-user ACUCOBOL-GT runtime license that accommodates each concurrent user that is anticipated. (If you anticipate 100

concurrent users of the Web runtime, you need a 100-user runtime license on the server in addition to the AcuServer license file. Alternatively, runtime users can install a local or network floating license for the runtime 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

AcuToWeb:

gcc versions

AcuToWeb requires the following gcc versions or later on the following platforms:

Platform	Minimum requirement
Aix6.1	GCC 7.1.0
Aix7.1	GCC 7.1.0
Aix7.2	GCC 7.1.0
HP 11.31 PA-RISC	GCC 4.3.1
HP 11.31 IA	GCC 4.2.3
Linux	GCC 4.8.0
Linux PPC	GCC 4.1.2-46
Sun Solaris 10	GCC 3.4.3
Sun Solaris 11	GCC 4.8.2

Linux-based platforms

For Linux-based platforms, the following packages must be added:

Debian-based	libc6:i386
	libstdc++6:i386
RHEL/Centos-based	libstdc++.i686
SUSE-based	libstdc++6-32bit

AIX and Solaris platforms

AIX and Solaris platforms require the Foreign Function Interface Library (libffi) is installed. More information and the installation packages can be found at the following:

AIX platforms

You can download an RPM package from:

http://www.bullfreeware.com/affichage.php?id=3638

And then install it using:

```
rpm -Uvh http://www.bullfreeware.com/download/bin/3638/
libffi-20170516-1.aix6.1.ppc.rpm
```

Solaris platforms

You can download the package from:

https://www.opencsw.org/packages/libffi6/

General information on libffi can be found at:

https://cffi.readthedocs.io/en/latest/installation.html

If you are running on the Solaris 11 platform, the minimum version required to run the AcuToWeb Gateway is version 11.3.

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:

- SQLSetStmtOption
- SQLSetScrollOptions
- SQLExtendedFetch
- SQLSetPos

See A_ODBC_LOCK_METHOD in the extend online help for more information.

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. The location is:

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

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

> **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.llc`. 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 a 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 a 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 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.

Also, 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.

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.

Supported Windows Platforms

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

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. To run the `.msi`, you must run it from a command prompt that has 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` file.

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**.

> **Restriction:** If you specify a mapped drive, it must map to a local directory; remote mapped drives are not supported.

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	Start AcuServer as a service
Start AcuConnect	Start AcuConnect as a service

> **Note:** Starting either AcuServer or AcuConnect as a service will only occur if you have run the `.exe` or `.msi` version of the installer with administrator privileges (see note above). If a previous version of either product is already running on its default port, an additional prompt is displayed, asking you if you want to shut down that service; you must answer **Y** if the new version of AcuServer or AcuConnect is to start - see *Notes on AcuConnect and AcuServer Installation* for more details.

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:

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

Silent Installation

On Windows platforms, you can perform a silent installation of the extend Interoperability Suite using the `msiexec` command, which requires that you use the `.msi` install package that is shipped with your product. This must be run from a command line prompt that has administrative privileges.

The syntax required is:

```
msiexec /i <msi-file> INSTALLDIR=<install-directory> [INSTALLDIR64=<64-bit-
install-directory>] ADDLOCAL=<product1,product2,...> [WINDOWSVERSION=<Win-
version>] /qn [/L*v <log-file>]
```

where:

<msi-file>

> The `.msi` installation file. This could be a 32-bit or a 64-bit version.

<install-directory>

> The directory in which the product will be installed.

> **Restriction:** If you specify a mapped drive, it must map to a local directory; remote mapped drives are not supported.

<64-bit-install-directory>

> The directory in which the 64-bit products will be installed. This is mandatory if you are using the 64-bit installer, because it installs both 32-bit and 64-bit versions of some products, using <install-directory> for the 32-bit versions and <64-bit-install-directory> for the 64-bit versions.

> **Restriction:** If you specify a mapped drive, it must map to a local directory; remote mapped drives are not supported.

<product1,product2,...>

> A list of products and services to be installed; see *Product Variables for Silent Installation* for the comprehensive list of options.

<Win-version>

> This is required for Windows versions 8 and later, in order to create the Extend Start menu. The only permissible value for <Win-version> is **PostWindows7**.

<log-file>

> The path and file name of a log file in which to log the installation details.

Examples

32-bit installation:

For example, the following command silently installs the ACUCOBOL-GT runtime, Acu4GL for MSSQL, and the AcuSQL runtime to the `C:\AcuInstallDir` directory, and (on Windows 8 and later) also creates the extend start menu in the Windows program list:

```
msiexec /i "extend(R) Version 10.2.0 x86.msi" INSTALLDIR=C:
\AcuInstallDir ADDLOCAL=Runtime,Acu4GLMSSQL,AcuSQLRuntime
WINDOWSVERSION=PostWindows7 /qn
```

64-bit installation:

For example, the following command silently installs the ACUCOBOL-GT runtime, Acu4GL for MSSQL, and the AcuSQL runtime to the `C:\AcuInstallDir` directory, and the 64-bit runtime, Acu4GL for MSSQL, and AcuSQL runtime to the `C:`
`\AcuInstallDir64` directory:

```
msiexec /i "extend(R) Version 10.2.0 x64.msi" INSTALLDIR=C:
\AcuInstallDir INSTALLDIR64=C:\AcuInstallDir64
ADDLOCAL=Runtime,Runtime64,Acu4GLMSSQL,Acu4GLMSSQL64,AcuSQLRunti
me,AcuSQLRuntime64  /qn
```

Comments

You can also use the `msiexec` command to run the installation with a user interface: omit the ADDLOCAL parameter and substitute `/qn` for `/qf`.

You can also use the `msiexec` command to install the thin client `.msi` file that is supplied with your product: omit the ADDLOCAL parameter.

Product Variables for Silent Installation

The following table contains a list of possible arguments that you can use with ADDLOCAL when running a silent installation. Most argument names are self-explanatory; the Notes column explains those that are not.

Argument name	Notes
Acu4GLDB2	
Acu4GLDB264	
Acu4GLMSSQL	
Acu4GLMSSQL64	
Acu4GLODBC	
Acu4GLODBC64	
Acu4GLOracle	
Acu4GLOracle64	
AcuBench	
AcuConnect	
AcuConnect64	
AcuConnectDistributedProcessing	
AcuConnectThinClient	
AcuServer	
AcuServer64	
AcuSQLPrecompiler	
AcuSQLPrecompiler64	
AcuSQLRuntime	
AcuSQLRuntime64	
AcuToWeb	
AcuXDBC	
AcuXDBC64	
AcuXDBCEnterpriseEdition	The 32-bit and the 64-bit versions of AcuXDBCEnterpriseEdition are mutually exclusive; you may only specify one of these versions during the installation.

Argument name	Notes
AcuXDBCEnterpriseEdition64	
AcuxdbcsBat	
AcuxdbcsBat64	
AcuXDBCServer	The 32-bit and the 64-bit versions of AcuXDBCServer are mutually exclusive; you may only specify one of these versions during the installation.
AcuXDBCServer64	
BIS	
Compiler	
DevSys	The ACUCOBOL-GT Development System, which includes the following: Compiler, Runtime, WebRuntime, ThinClient, and WebThinClient.
ExtendStartMenu	The entry shown on the Windows program menu for Windows versions 8 and later. The 32-bit and the 64-bit versions of ExtendStartMenu are mutually exclusive; you may only specify one of these versions during the installation.
ExtendStartMenu64	
LicenseActivator	
LicenseActivator64	
OnlineDocumentationCHM	
Runtime	
Runtime64	
ThinClient	
VCRedist	Installs the Microsoft redistributable files, required by the extend products, if they are not already installed.
VortexJar	Required for AcuXDBCEE. The enterprise edition of AcuXDBC needs the vortex.jar file, which enables a Java client application to connect to your Vision database.
VortexJar64	
WebRuntime	
WebThinClient	

Examples

32-bit installation:

For example, the following command silently installs the ACUCOBOL-GT runtime, Acu4GL for MSSQL, and the AcuSQL runtime to the C:\AcuInstallDir directory,

and (on Windows 8 and later) also creates the extend start menu in the Windows program list:

```
msiexec /i "extend(R) Version 10.2.0 x86.msi" INSTALLDIR=C:
\AcuInstallDir ADDLOCAL=Runtime,Acu4GLMSSQL,AcuSQLRuntime
WINDOWSVERSION=PostWindows7 /qn
```

64-bit installation:

For example, the following command silently installs the ACUCOBOL-GT runtime, Acu4GL for MSSQL, and the AcuSQL runtime to the `C:\AcuInstallDir` directory, and the 64-bit runtime, Acu4GL for MSSQL, and AcuSQL runtime to the `C:\AcuInstallDir64` directory:

```
msiexec /i "extend(R) Version 10.2.0 x64.msi" INSTALLDIR=C:
\AcuInstallDir INSTALLDIR64=C:\AcuInstallDir64
ADDLOCAL=Runtime,Runtime64,Acu4GLMSSQL,Acu4GLMSSQL64,AcuSQLRunti
me,AcuSQLRuntime64  /qn
```

Windows 64-bit Installations

There are 64-bit versions of most extend products. These 64-bit versions are installed using a separate 64-bit version of the installer. The installation process follows the same steps as described in *Installation on Windows Platforms*, with the following notable exceptions.

When running the 64-bit installer, if no 64-bit version exists for a selected product (for example, AcuBench), the 32-bit version is installed.

Products such as AcuConnect and AcuServer have 32-bit and 64-bit versions, and both are installed if you select these products during installation. You can also decide which version of the product to start on completion of the installation.

The AcuXDBC product is broken down into three installations: for the Data Interface, you can install both the 32-bit and 64-bit versions; and for the AcuXDBC Server and Enterprise Edition, you must chose which version to install.

By default, all 64-bit product versions are installed in the `Program Files` directory, and 32-bit product versions (and any supporting non-64-bit tools) are installed in the `Program Files (x86)` directory; although, you can change these locations during the installation. All the 32-bit versions are fully supported and functional in a 64-bit environment.

Remember: When running the license activator after the installation, the 64-bit version of the Activator utility is run, which installs license files into both the 32-bit and 64-bit directories. When running the Activator utility from the command prompt, make sure you are using the 64-bit command prompt to ensure the correct licenses are generated and placed in the correct locations; otherwise, if the 32-bit Activator utility is run, only license files for 32-bit products will be generated.

Installation Notes

Debugging files

The Windows distribution media now contains `.pdb` debugging files to assist customer support in solving runtime issues on a customer's machine - `.pdb` files enable runtime stack information to be generated.

To setup this ability, copy the `.pdb` files from the 32-bit or 64-bit `Debug` directories on the distribution media to the 32-bit or 64-bit directory of the installation. 32-bit files on the media are located in `Debug\Win32\bin\*.pdb`, and 64-bit files on the media are located in `Debug\x64\bin\*.pdb`.

Once the `.pdb` files are copied, follow any instructions given to you by Customer Support.

AcuConnect and AcuServer

During the installation process, if you are installing AcuConnect or AcuServer, you can choose to automatically start those services as part of the installation process.

Those services can only be started if certain conditions are met. For example, the services will attempt to start on a default port (AcuConnect 5632, AcuServer 6523). If an existing installation is already using that port, that service must be stopped if the installation is to create and run the new service. If you do not stop the existing service, the installation can only create the new service; it cannot run it.
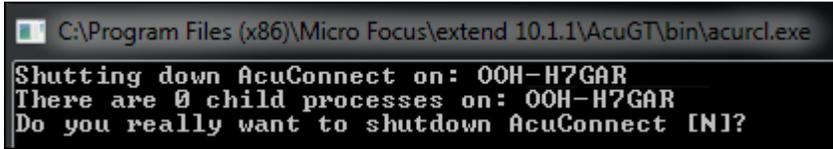
If there is no previous installation, a default `AcuAccess` file is only created (in its default location of `C:\etc`) when the service starts. If you configure the installation to not start the service, no `AcuAccess` file will exist until you start the products from their respective control panels (or command line equivalents). If a previous installation exists, the `AcuAccess` file already located in `C:\etc` is used.

> **Note:** This file and directory are not removed when you uninstall a previous product.

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.
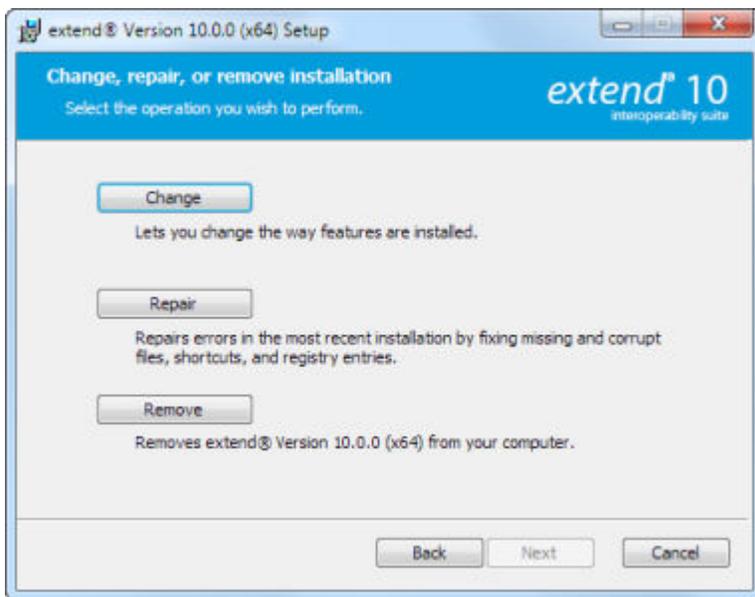
Launching extend Products on Windows 8 or Later

Since the introduction of Windows 8, the Windows program menu does not display programs in a hierarchical way. For the extend Interoperability Suite, this would mean that all the programs relating to AcuBench, ACUCOBOL-GT, AcuConnect, etc... would be displayed as a flat list, and it would be difficult to distinguish which programs were related to which products.

In order to show the programs grouped within the products they belong to, the extend Start Menu has been introduced. After installation of the extend Interoperability Suite, there will be one entry in the Windows programs menu - **extend x.x.x Start Menu** (where *x.x.x* is the version number of the installed product):
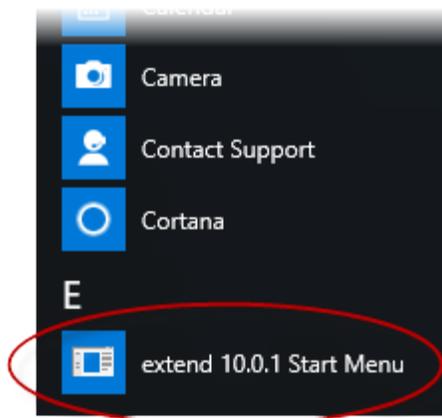
Launch this application to display a hierarchy that looks similar to the Windows program menus prior to Windows 8. Simply navigate the folders displayed in the left-hand pane to display shortcuts to the related programs in the right-hand pane. Select the required shortcut: at which point, the extend Start Menu closes and the program is launched.
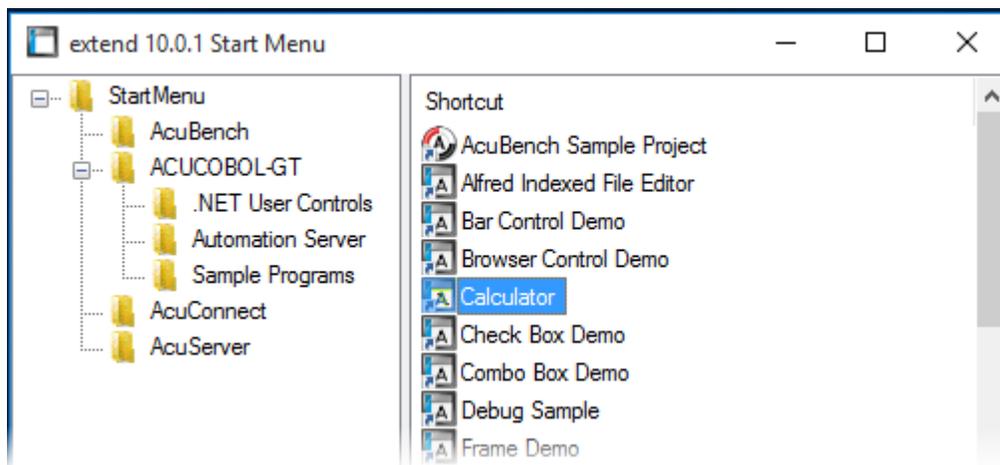
Compiling Your Programs

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` (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:

```
ccbl32 -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:

```
ccbl32 -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 executable 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 `\bin` directory path to the PATH system environment variable (typically, 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 <*printername*>". 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 <printername>

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 \\printername
```

in the configuration file (CBLCONFI). *Printername* is the printer designation as given in the **Devices and Printers** screen. 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 printer name, 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 \\printername" TO WS-PRINTER-NAME.
```

When the runtime opens a file assigned to "-Q <printername>", it sets the Windows print spooler to use this printer. The printer driver must be installed on the computer from which you print. If *printername* 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 (CBLCONFI):

```
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 theses 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 Files

If you need to alter a license file, contact your Micro Focus *extend* representative for updated product codes and product keys. When you receive them, launch the Activator utility 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 are delivered via FTP.

Installation Process

1. Download the product from the Micro Focus SupportLine site.

   Note: Before running the installer, ensure that the `tar` utility in on your PATH.

2. Enter the following:
   ```
   /path/to/installer/installer-name [options]
   ```
   where *installer-name* will be something similar to `setup_acucob1010pmk59shACU`.

   The following options can be included:

Option	Description
`-d installation-path`	Specifies a new default install location offered during the installation. If not specified, the default location is the current working directory (.). Any specified directory must already exist.

Option	Description
	This option can be combined with the `-EULA` option, but if it is, it will specify the location of the extracted EULA, and not the location of the installed product.
`-EULA`	Extracts the EULA, in `.htm` format, to a location of your choosing. The product installation will not continue when using this option.
`-help`	Displays the available options that can be appended to the installation command.

> **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. From the installation directory, enter the following command to run the Activator utility:

```
./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. To start the acushare license manager service, enter the following:

```
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 > Runtime Manual > Shared Memory > 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 `aclnt.so` (or `aclnt.sl`) in case the LIBPATH (or LD_LIBRARY_PATH) variable is not set. libclnt.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 `aclnt.so` (or `aclnt.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.llc` and `/opt/acu/bin/runcbl.llc` 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.llc` and finally `/opt/acu/bin/runcbl.llc`.

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.llc` 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 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:

AcuBench Enhancements

This section includes the enhancements related to AcuBench.

ECN-WB718 New atw-script special properties control

Product: AcuBench

Module: AcuBench.exe

Machines Affected: Windows

Known Versions Affected: All

DESCRIPTION:

The new atw-script control with special properties (see *ECN-4662*) has been added to AcuBench.

The atw-script control does not show to the user; therefore, you can add it anywhere on the screen and make it any size. However, it does need to be visible in the screen designer to enable selection and the setting of properties.

Note: We recommend that you modify and inquire the special properties of the atw-script control in line in your COBOL code.

ECN-WB722 New grid properties and style

Product: AcuBench

Module: AcuBench.exe

Machines Affected: Windows

Known Versions Affected: N/A

DESCRIPTION:

AcuBench 10.4.0 supports the following for Grid controls:

REGISTRY-KEY common property	REGISTRY-KEY enables the runtime to store information in the registry about modifications to the grid via sort-column, physical-columns, and column widths. This information is loaded from the registry when a COBOL program with the same key name is run. For details, see the *Common Properties* topic for Grid controls in your extend documentation.
MOVEABLE-COLUMNS style	Enables the user to move or hide grid columns by placing the cursor over a column header and moving it to a new position. Users can also drag the column to hide the column, leaving a portion of it in the header to show it has been hidden. For details, see the *Common Properties* topic for Grid controls in your extend documentation.

PHYSICAL-COLUMNS special property	The PHYSICAL-COLUMNS property enables programmers to move or hide columns. It is indexed by the logical column number in the original layout. The *logical* column is the column number as the program originally constructed the grid. A *physical* column is the actual column number on the screen. For details, see the *PHYSICAL-COLUMNS (numeric)* special properties topic for Grid controls in your extend documentation.

You can access these from the **Grid Properties** screens.

ACUCOBOL-GT Enhancements

This section includes the enhancements related to ACUCOBOL-GT.

ECN-4658 Create .NET objects on the server in the distributed environment

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

DESCRIPTION:

You can now create .NET objects on the server when executing in a distributed environment such as Thin Client or AcuToWeb.

When you use the CREATE statement to create a .NET object on the server, you must specify a SERVER-NAME. The value of the SERVER-NAME must start with the prefix `Local:` to signify that the object should be created on the application host (server runtime) rather than on the client (display host, or thin client, or web browser). For example:

```
CREATE .NET_object SERVER-NAME "Local:the_server_name"
```

Note: In a non-distributed environment, the server-name property has no effect, meaning you can use this format with server-name for compiling both distributed and non-distributed environments with no change in code required.

ECN-4659 New C$SOCKET server-name parameter and numeric option

Incident: 3114136

RPI Number: 1110307

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: 10.4.0

DESCRIPTION:

The C$SOCKET AGS-CREATE-SERVER interface has been enhanced to enable specification of an optional server name, and to optionally indicate that the server name is a numeric IP address.

```
C$SOCKET using AGS-CREATE-SERVER, port-number [, server-name, [numeric]]
```

Where:

- *server-name* is a `PIC X(n)` data item indicating the network interface address to which the server will be attached
- *numeric* is a numeric data item holding a non-zero value when *server-name* is a numeric IP address

The C$SOCKET AGS-CREATE-CLIENT interface has been enhanced to allow an optional indication that the server name is a numeric IP address.

```
C$SOCKET using AGS-CREATE-CLIENT, port-number, server-name [, numeric]
```

Where *numeric* is a numeric data item holding a non-zero value when *server-name* is a numeric IP address.

ECN-4662 New atw-script control for use with AcuToWeb

Product: ACUCOBOL-GT

Module: Compiler, Runtime

Machines Affected: All

DESCRIPTION:

The new atw-script control, for use with AcuToWeb, enables scripting in the browser used as the user interface for your application.

ATW-SCRIPT is a hidden screen control that can execute Javascript code and can be controlled by the COBOL program just like other screen controls. You can establish links to:

Common Properties

Because ATW-SCRIPT is a hidden control, meaning it is not visible in the user interface, it has no common properties.

Special Properties

You can include the atw-script control in the COBOL program Screen Section, or by using Display syntax in the Procedure Division. This control ignores all common properties, such as position, visible, title, etc. The control is not visible, meaning it has no position, and the user cannot interact with the control directly.

The COBOL program can Modify the control to execute browser supported scripting such as JavaScript. This control has six special properties, two of which can be inquired. The control has an event, NTF-ATW-EVENT. There is a new AcuToWeb Gateway configuration "path_resources", this variable is used to indicate the path used for return all css and js file requested from COBOL code.

For the syntax below, `js1` is defined as the handle for the atw-script control. EVALUATE takes a single string. That script is executed on the browser immediately. For example:

```
        move   "document.getElementById(""demo"").innerHTML = ""Paragr
    -      "aph changed."";" to js_str
        modify js1 evaluate(js_str)
```

ADD takes two strings - an identifier and a script. This associates the script with that identifier, in order to reference that script later. The script is not immediately executed. For example:

```
        move "function myFunction() {document.getElementById(""demo""
    -       "").innerHTML = ""Paragraph changed.""; }" to js_str
        modify js1 add("this-is-my-id", js_str)
modify js1 add("chartsJS", "[SRC]:https://cdnjs.cloudflare.com/ajax/libs/
Chart.js/2.9.3/Chart.js") giving addJSCharts
```

The prefix `[SRC]:` means that the client browser looks for resources. In this case, it describes that the content is a URL and the content is not JavaScript.

```
modify js1 add("myscript", "[SRC]:/resources/js/myscript.js") giving addJSVal
```

In this case, the file requested is a local file in web folder – the gateway web folder. The `resources` matches with the PATH defined in `path_resources`, which is configurable in `gateway.conf`.

✎ **Notes:**

- When you use ADD to add a custom function, you must consider the name translation. For example:

```
modify js1 add("myfunc", "function f1(msg){ alert(msg); }") giving
addJSVal
```

While this is formally correct syntax, due to the architecture of atw-script, the function `f1` won't be reachable from any further CALL statement. The correct format for an inline function, which is a function that is not defined into a script added with the [SRC] sintax, is:

```
modify js1 add("myfunc", "f1:function (msg){ alert(msg); }") giving
addJSVal
```

Then, call the function as follows:

```
modify js1 call("myfunc.f1(""hello world"")") giving addJSVal
```

- The `giving` value is important because `return` provides the number identified by ATW-SCRIPT-EVENT in DATA-EVENT-2 when DATA-EVENT-1 is equal to 1.

VARIABLE can be used to get or set variables in the script. Modify this property to set a variable, and inquire this property to get the value of the variable. For example:

```
        move "var price1; var price2; var total; function myFunction(
-           ") { total = price1 + price2; }" to js_str
        modify js1 add("another-id", js_str)
        modify js1 variable("price1", num1)
        modify js1 variable("price2", num2)
        modify js1 call("myFunction()")
        inquire js1 variable("total") in result
```

CALL takes at least one string – the name of the function to call. If that function takes parameters, you can include them as part of the string you create to modify the CALL parameter. Alternatively, you can specify multiple strings. The extra strings are the parameters passed to the function. For example:

```
        move "function myFunction(name,job) {document.getElementById(
-           """demo"").innerHTML = ""Welcome "" + name + "", the ""
-           "+ job + "".""; }" to js_str
        modify js1 add("this-is-my-id", js_str)
        modify js1 call("myFunction('Harry Potter','Wizard')")
        move "Harry Potter" to par1
        move "Wizard" to par2
        modify js1 call("myFunction()", par1, par2)
```

Alternative ways to use the CALL property:

```
        Move 1 to var1
        Move 2 to var2
        Move "test" to var3

        Modify js1 call("myFunction(1, 2, ""test"");").
```

Or:

```
        Modify js1 call("myFunction()", var1, var2, var3).
```

NTF-ATW-EVENT NTF-ATW-EVENT this event will trigger when there is an event that has been received by the JS, when the JS has been injected via the ATW-SCRIPT. Note that only the code that has been inserted via ATW-SCRIPT will eventually trigger this event.

```
EVALUATE EVENT-TYPE
     WHEN NTF-ATW-EVENT
       EVALUATE event-data-1
         WHEN 1
               modify js1 CALL("googleMaps")
               modify js1 add("script2",
                          "[SRC]:/js/myscript2.js")
```

LAST-ERROR LAST-ERROR can be inquired and contains the last error the Javasctipt engine returned. There could be many errors as these depends on what your JS code does. Normally these errors goes into the JS console of the browser, but this property provide a means to COBOL program to be aware of such error message. The AcuToWeb will queue all the error messages that originate from the code injected by AcuToWeb into a FIFO list and will delete the message once COBOL inquires this property.

```
 inquire js1 last-error value in valuefromJS
```

LAST-RESULT LAST-RESULT can be inquired to get any character result from the last script called, and is available when an EVALUATE returns a positive value.

REMOVE REMOVE takes a single string, which is an identifier used with the ADD property. This removes that script from the browser, so it can no longer be referenced. For example:

```
 modify js1 remove ("this-is-my-id")
```

Events

NTF-ATW-EVENT is triggered when an event is received by the JavaScript, when the JavaScript has been injected via the ATW-SCRIPT.

```
      EVALUATE EVENT-TYPE
   WHEN NTF-ATW-EVENT
     EVALUATE event-data-1
       WHEN 1
             modify js1 CALL("googleMaps")
             modify js1 add("script2",
                        "[SRC]:/js/myscript2.js")LAST-ERROR
```

Note: The only event triggered by AcuToWeb occurs when the ADDed script has finished loading.

If you want to notify to the COBOL program of any other event, you can use the `CustomJSHelper.event` helper in your JavaScript code as follows:

```
getData: function(_url) {     $.ajax({url:_url}).then( function(data)
        { getDataObj.dataReturn = JSON.stringify(data);
        CustomJSHelper.event("getDataObj", CustomJSHelper.DATA_LOADED,
1);      });
        },getReturn: function(){ return getDataObj.dataReturn; }
```

Values for `EVENT-DATA-1` can be:

`ATW_SCRIPT_LOADED = 1`	Script loaded
`ATW_SCRIPT_CUSTOM_EVENT = 2`	Custom event; the JavaScript developer can generate a custom event from JavaScript to COBOL
`ATW_JSON_DATA_LOADED = 3`	JSON is loaded; the JavaScript developer can use it when the ajax function returns a value from the webservice call

`ATW_JSON_ERROR_DATA_LOADED = 4`	JSON is error loaded; the JavaScript developer can use it when the ajax function returns an error from the webservice call

When the value of `EVENT-DATA-1` is `CUSTOM_EVENT` or `DATA_LOADED` or `ERROR_DATA_LOADED`

`EVENT-DATA-2` is a number, defined by the JavaScript developer, that identifies the number of AJAX operations contained in the JavaScript code.

ECN-4673 ECNs removed from 10.4.0 onwards

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

DESCRIPTION:

The correction documented in resolved issues, *ECN-4673*, has eliminated the following ECNs starting with release 10.4.0:

4444
4466
4475
4497
4507
4509

ECN-4678 Two new fields added to C$REDIRECT I/O linkage handlers

Incidents:

3231462
3231463

RPI Numbers:

1120623
1120625

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: N/A

DESCRIPTION:

Two new fields have been added to the linkage item passed to C$REDIRECT I/O handlers:

HANDLER-SEQUENTIAL-TYPE	Contains the type of sequential file. Values are the same as defined for SEQ-TYPE in `def/filesys.def`.
HANDLER-RELATIVE-KEY	Contains the relative key for relative files. Changes made to this field by the handler will be returned to the runtime's key value for the file after the handler returns.

In previous versions, `handler.cpy` only contained the latest version of the current linkage item. `handler.cpy` has been replaced with three versions of the file, each containing one of three revisions to the linkage item.

To access the new fields, do one of the following:

- Set a symbolic link (or similar) from `handler.cpy` to a new version of the file that contains the desired version of the linkage item. This option enables future updates without disturbing existing code that uses an older version of the linkage item.
- Update your source code to reference the desired version of the linkage item.

Example:

1. Include `handler`*n*`.cpy` in the I/O handler program.
2. Set HANDLER-VERSION to *n* before registering the I/O handler with the runtime with C$REDIRECT.

   ⚠️ **Important:** Be sure that HANDLER-VERSION matches the version of the linkage item included in your code.

ECN-4679 New CobolJavaJDBC sample program

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

DESCRIPTION:

An new sample program, `CobolJavaJDBC.cbl`, along with accompanying readme, configuration, and batch script files, has been added to the `%PUBLIC%\Documents\Micro Focus\extend 10.4.0\sample\java` directory. This new program shows how to use C$JAVA and JBDC to access data in a Microsoft Access table. Required software and setup details are included in the `CobolJavaJDBC-README.txt` file.

ECN-4686 New RESTful Web services support in RMNet

Product: ACUCOBOL-GT

Module: rmnet.dll

Machines Affected: All

DESCRIPTION:

The `Http-Get` and `HttpPut` function calls have been added to support RESTful Web services.

`Http-Get`	This function initiates an HTTP GET request and waits for a response. The HTTP GET method is used to request a representation of the specified resource.

	Usage	``` CALL "HttpGet" USING destination-url response-pointer response-length [extra-headers] GIVING status-code. ```

Parameters

destination-url	An alphanumeric item specifying the target URL for the GET.
response-pointer	A pointer to an alphanumeric item specifying the value of the HTTP response. After use, use `NetFree` to return memory used to store response.
response-length	A numeric item returning the length of the HTTP response to which *reponse-pointer* points.
extra-headers	An optional alphanumeric item specifying extra headers to be added to the HTTP header. This argument consists of name/value pairs separated by hex x"00", and ended with two x"00"'s.
status-code	A numeric item that, if nonzero, indicates the request encountered an error. In this case, the string representation of the error may be obtained by calling `NetGetError`.

HttpPut This function initiates an HTTP GET request and waits for a response. The HTTP GET method is used to request a representation of the specified resource.

Usage
```
CALL "HttpGet" USING
        destination-url
        response-pointer
        response-length
        [extra-headers]
        GIVING
        status-code.
```

Parameters

destination-url	An alphanumeric item specifying the target URL for the GET.
response-pointer	A pointer to an alphanumeric item specifying the value of the HTTP response. After use, use `NetFree` to return memory used to store response.
response-length	A numeric item returning the length of the HTTP response to which *reponse-pointer* points.
extra-headers	An optional alphanumeric item specifying extra headers to be added to the HTTP header. This argument consists of name/value pairs separated by hex x"00", and ended with two x"00"'s.
status-code	A numeric item that, if nonzero, indicates the request encountered an error. In this case, the string representation of the error may be obtained by calling `NetGetError`.

ECN-4687 New functions to support HTTP servers that require a login

Product: ACUCOBOL-GT

Module: rmnet.dll

Machines Affected: All

DESCRIPTION:

The `HttpSetUsername`, `HttpSetPassword`, `HttpSetLoginOptions`, and `HttpSetSSLVersion` function calls have been added to support HTTP servers that require a login a login.

HttpSetUsername This function sets the CURLOPT_USERNAME option value to the string specified. See also the *HttpSetPassword* and *HttpSetLoginOptions* function topics.

    Usage
```
CALL "HttpSetUsername" USING
        user-name
        GIVING
        response-status.
```

  Parameters

user-name	An alphanumeric item specifying the user name to use in authentication. Trailing spaces are removed from the value and then the value is zero terminated.
response-status	A status code that, if nonzero, indicates the request encountered an error. In this case, the string representation of the error may be obtained by calling `NetGetError`.

HttpSetPassword This function sets the CURLOPT_USERNAME option value to the string specified. See also the *HttpSetUsername* and *HttpSetPassword* function topics.

    Usage
```
CALL "HttpSetPassword" USING
        password
        GIVING
        response-status.
```

  Parameters

password	An alphanumeric item specifying a password to use in authentication. Trailing spaces are removed from the value and then the value is zero terminated.
response-status	A status code that, if nonzero, indicates the request encountered an error. In this case, the string representation

	of the error may be obtained by calling `NetGetError`.

HttpSetLoginOptions This function sets the CURLOPT_USERNAME option value to the string specified. See also the *HttpSetUsername* and *HttpSetPassword* function topics.

Usage
```
CALL "HttpSetLoginOptions" USING
          login-options
          GIVING
          response-status.
```

Parameters

login-options	An alphanumeric item specifying the login options to use in authentication. This can be used to specify protocol specific options such as the preferred authentication mechanism. Trailing spaces are removed from the value and then the value is zero terminated.
response-status	A status code that, if nonzero, indicates the request encountered an error. In this case, the string representation of the error may be obtained by calling `NetGetError`.

HttpSetSSLVersion This function sets the CURLOPT_SSLVERSION option to the specified value, which represents the preferred TLS/SSL protocol version to use when attempting to connect to the server.

🖉 **Note:** If a call to HttpPost results in an SSL connect error (007 return status), and you suspect that the server does not support TLS negotiation, try adding the following call after calling NetInit and before calling HttpPost.
```
call "HttpSetSSLVersion" using "SSLv3" giving
response-status.
```

Usage
```
CALL "HttpSetSSLVersion" USING
          version-string
          GIVING
          response-status.
```

Parameters

login-options	An alphanumeric item or literal specifying the SSL version to use. Valid values include: SSLv2 SSLv3 TLSv1 TLSv1_0

	TLSv1_1 TLSv1_2 Trailing spaces are removed from the value and then the value is zero terminated. **Note:** If you specify a value other than one of these valid values, the default value of CURL_SSLVERSION_DEFAULT is used, which causes the client to attempt to determine the remote TLS/SSL protocol version.
response-status	A status code that, if nonzero, indicates the request encountered an error. In this case, the string representation of the error may be obtained by calling `NetGetError`.

ECN-4697 Hidden and moveable grid columns

Product: ACUCOBOL-GT

Module: Compiler, Runtime

Machines Affected: Windows

Known Versions Affected: N/A

DESCRIPTION:

ACUCOBOL-GT runtime on Windows now supports hidden and movable columns in a standard grid control via the `ADJUSTABLE-COLUMNS` and `MOVEABLE-COLUMNS` grid styles.

In previous releases, columns could not be completely hidden, and had a minimum width of 1. This update enables the column width to be set to 0 (zero), hiding the column completely from view. In addition, users can now move a grid column from one location in the grid to another.

The `gridctl.cbl`sample program, located in your `%PUBLIC\Documents\Micro Focus\extend 10.4.0\sample` directory, has been updated to show these features.

For complete details on the hidden and moveable grid columns feature, see the *Working with Hidden and Moveable Grid Columns* topic in your ACUCOBOL-GT documentation.

ECN-4698 Increase in size of CJAVA-DBQUERY query string

Incident: 3214795

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

The size of the CJAVA-DBQUERY query string has been increased from 1024 to 32768 bytes.

ECN-4700 New JSON PARSE and GENERATE feature

Product: ACUCOBOL-GT

Module: Compiler, Runtime, JSON

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

This release includes new functionality that supports JSON PARSE and GENERATE for the compiler and runtime.

For complete details, see the following topics in your *ACUCOBOL-GT Reference Manual*:

> *JSON GENERATE Statement*
> *JSON PARSE Statement*

AcuSQL Enhancements

This section includes the enhancements related to AcuSQL for this release.

ECN-SQL162 Support for COBOL-CHARACTER-SET in AcuSQL

Product: AcuSQL

Module: asqlsrvr, esqllib

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

AcuSQL now respects the value of the COBOL-CHARACTER-SET configuration variable for both the SQL Server-specific version (asqlsrvr) and the general ODBC version (esqllib).

AcuToWeb Enhancements

This section includes the enhancements related to AcuToWeb.

ECN-AW157 CSS Theme Generator now supports customized controls

Product: AcuToWeb

Module: AcuToWeb

Machines Affected: All

DESCRIPTION:

A new control on the Themes Generator **Home** tab, **Add Custom**, enables you to create customized controls.

When you select the Add Custom button, you can

1. On the Themes Generator **Home** tab, click **Add Custom**.
2. Enter an ATW-CSS property, which is either ATW-CSS-CLASS or ATW-CSS-ID.
3. Enter a Property Value that matches an entry used with a control in the COBOL program.
4. Click **OK**.

   Controls selected with the **Add Custom** button appear on the **Customs** tab of the Themes Generator.
5. Export the stylesheet.

   Note: To view, edit or add more custom controls to the stylesheet, return to the Themes Generator and import the CSS file.

ECN-AW159 AcuToWeb Connection Panel now detects websockets

Product: AcuToWeb

Module: AcuToWeb

Machines Affected: All

DESCRIPTION:

A circle has been added to the right side of the Gateway Connection in the AcuToWeb Connection Panel. The circle turns green and provides a tooltip when AcuToWeb can establish websocket communication. Otherwise, the circle turns red.

AcuXDBC Enhancements

This section includes the enhancements related to AcuXDBC.

ECN-XD127 KEY columns excluded from "null_alpha_read null"

Product: AcuXDBC

Module: AcuXDBC

Machines Affected: All

Known Versions Affected: 10.4.0

DESCRIPTION:

To ensure that columns that are part of a key return what is in them instead of NULL, add the following to the AcuXDBC configuration file:

```
null_alpha_read        null
```

This nulls out spaces from key columns.

Resolved Issues

The following are resolved issues for the extend products.

Acu4GL ECN List

This section includes the ECNs relating to Acu4GL:

ECN-GL579 Error reading NULL data from national columns

Incident: 3228705

RPI Number: 1120415

Product: Acu4GL

Module: MSSQL

Machines Affected: All

Known Versions Affected: 10.1.0 and later

DESCRIPTION:

When reading national data (nchar, nvarchar), the interface reported a 9D, 00 error when the data was NULL.

ECN-GL580 Unable to use 4GL-COLUMN-CASE

Incident: 3231590

RPI Number: 1120630

Product: Acu4GL

Modules: ODBC, DB2

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

It was not possible to create tables or columns with upper-case or misxed-case names. This has been corrected.

ECN-GL581 Missing records on READ NEXT with split key

Incident: 3232617

RPI Number: 1120752

Product: Acu4GL

Modules: Oracle

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

When using the Oracle configuration variable A_ORA_KEEP_START_CURSOR with a split key and a START EQUAL, the interface sometimes skipped records after a READ NEXT. This has been corrected.

ECN-GL582 No AIX support for Acu4GL for ODBC

Incident: 3230552

RPI Number: 1120624

Product: Acu4GL

Modules: ODBC

Machines Affected: AIX

Known Versions Affected: All

DESCRIPTION:

A problem with Acu4GL for ODBC that caused it to look for the wrong libraries on AIX machines rendered it unusable for AIX. This has been corrected.

AcuBench ECN List

This section includes the ECNs relating to AcuBench:

ECN-WB720 Limitation when searching for regular expressions

Incident: 3227062

RPI Number: 1120195

Product: AcuBench

Module: AcuBench.exe

Machines Affected: Windows

Known Versions Affected: 10.2.0 and later

DESCRIPTION:

When searching source code for a regular expression, clicking the Up radio button find no matches. This is a permanent restriction.

ECN-WB721 AcuBench crash in the ActiveX Component bar

Incident: 3235088

RPI Number: 1121011

Product: AcuBench

Module: AcuBench.exe

Machines Affected: Windows

Known Versions Affected: 10.2.0 and later

DESCRIPTION:

Adding certain ActiveX controls, particularly the Windows Server-specific Group Policy Management control, caused Acubench to crash. This has been corrected.

ECN-WB843 MOVE X TO ACCEPT CONTROL omitted NONE option

Incident: 99-003545

Product: AcuBench

Module: AcuBench80.exe

Machines Affected: Windows

Known Versions Affected: 7.0 through 10.3.1

DESCRIPTION:

Starting with version 7.0, the **Tools/Options/Code Gen** drop-down list was updated with an option that enables users to determine what integer value was placed in the MOVE X TO ACCEPT CONTROL statement (0,1,2,3,4). The value NONE, which enables you to leave the statement out, was erroneously omitted. This has been fixed, and NONE is now a choice in the drop-down.

ECN-WB846 Deleted files were not going to the recycle bin

Incident: 99-028813

Product: AcuBench

Module: AcuBench80.exe

Machines Affected: Windows

Known Versions Affected: 10.3.1 and earlier

DESCRIPTION:

In the **Structural**, **File**, and **Data View** trees, the **Delete From Disk** command on the popup menu deleted the file with no option to undo the action. The **Delete From Disk** behavior has been changed to move deleted files to Windows recycle bin.

ECN-WB848 Parser did not allow BLANK WHEN ZEROS

Incident: 99-024680

Product: AcuBench

Module: AcuBench80.exe

Machines Affected: Windows

Known Versions Affected: 8.0.0 through 10.3.1

DESCRIPTION:

The `BLANK WHEN ZEROS` syntax caused a parser error. This has been corrected to accept `BLANK WHEN ZEROS`, which is equivalent to `BLANK WHEN ZERO`.

ACUCOBOL-GT ECN List

This section includes the ECNs relating to ACUCOBOL-GT:

ECN-4659 Incorrect binding for SERVER_IP and SERVER_HOST

Incident: 3114136

RPI Number: 1110307

Product: ACUCOBOL-GT

Module: Socks

Machines Affected: All

Known Versions Affected: 10.4.0

DESCRIPTION:

The SERVER_IP and SERVER_HOST configuration variables were originally implemented to associate a particular IP address or host name with a server process on a system with multiple network interfaces. However, this behavior was inadvertently changed in a previous release such that these variables enabled the server to be available on all the network interfaces on the system. This change restores the original, correct behavior.

SERVER_IP and SERVER_HOST now identify a server IP address or host name. The specified server binds only to the indicated interface. The server is not accessible via any other network interfaces on the system.

ECN-4660 Thin Client automatic update TC_INSTALLER_TARGET_DIR has no effect

Incident: 3208450

RPI Number: 1118458

Product: ACUCOBOL-GT

Module: Thin Client

Machines Affected: Windows

Known Versions Affected: 10.0.0 through 10.3.1

DESCRIPTION:

The documentation says:

The TC_INSTALLER_TARGET_DIR configuration variable was designed to apply only to the ACUCOBOL-GT Thin Client automatic update feature, and to be used to specify the location where you want the updated thin client to be installed; however setting TC_INSTALLER_TARGET_DIR had no effect, and always used the default installation directory, which, by defaut, is:

```
C:\Program Files (x86)\Micro Focus\extend 10.2.1\
```

In this release, TC_INSTALLER_TARGET_DIR has been updated to work as designed.

ECN-4665 Acuthin ignoring user input during heavy output

RPI Number: 1117080

Product: ACUCOBOL-GT

Module: Thin Client

Machines Affected: Windows

Known Versions Affected: All

DESCRIPTION:

If an application performed a lengthy sequence of screen updates, the user interface was sometimes unresponsive and occasionally entered a "not responding" state if the user tried to interact with it.

This change ensures that the UI remains responsive.

Note: If a long sequence of updates has already been queued, those must be processed before any user input is acted upon. This might appear to slow down processing; however, this is the expected behavior. To increase processing speed, try reducing the frequency of screen updates, for example, by updating a progress bar at moderate intervals, or by counting every 100th record processed instead of every record.

ECN-4666 Bitmap buttons with text issues

Incident: 2844920

RPI Number: 1102392

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: All

DESCRIPTION:

Bitmap pushbuttons with text did not animate when clicked, and the cropping of the image by the text was inconsistent. These have been corrected.

ECN-4671 Acuthin hangs on ACCEPT BEFORE TIME 0

Incident: 3214711

RPI Number: 1119006

Product: ACUCOBOL-GT

Module: Runtime, Acuthin

Machines Affected: Windows

Known Versions Affected: 10.2.1 and later

DESCRIPTION:

When executing ACCEPT item BEFORE TIME 0 with the thin client, the process would sometimes hang. This has been fixed.

📝 **Note:** This problem occurred when the ECN-4576 configuration variable was set to ON. You can set ECN-4576 to OFF.

ECN-4672 "Index out of bounds" error when initializing a var-size array

Incident: 3227763

RPI Number: 1120266

Product: ACUCOBOL-GT

Module: Compiler

Machines Affected: All

Known Versions Affected: 10.3.0 and later

DESCRIPTION:

When executing INITIALIZE on a data item that included OCCURS DEPENDING ON, and had the variable set to something less than the maximum size, the runtime showed an "Index out of bounds" error. This has been fixed.

ECN-4673 AcuThin displays a "Not Responding" message on lengthy processes

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

DESCRIPTION:

While using AcuThin client, and when the server program is performing a lengthy process such as I-O or a called subroutine, a "Not Responding" message sometimes displayed on the client. This has been corrected.

This fix eliminates the following ECNs from version 10.4.0 and later:

```
4444
4466
4475
4497
4507
4509
```

ECN-4674 AcuThin disappears without error

Product: ACUCOBOL-GT

Module: AcuThin, Runtime

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

When a COBOL program flooded the thin client with a large number of DISPLAY or MODIFY statements without asking for input, the thin client sometimes disappeared without warning, causing the resulting COBOL program to shut down. This has been fixed.

ECN-4675 C$REGEXP crash

RPI Number: 645154

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

When the START parameter (parameter 4) for the MATCH operation code was larger than the length of match parameter, which sometimes happened when the variable was not initialized, C$REGEXP could crash the runtime. This has been corrected.

ECN-4676 Runtime memory corruption when calling .NET assemblies

Incident: 3229306

RPI Number: 1120563

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: 10.3.0 and later

DESCRIPTION:

When calling .NET assemblies, the runtime generated BSTR strings to pass to the assembly. Sometimes, these BSTR strings needed to be freed, and sometimes not; however, they were always being freed. This sometimes resulted in the double-freeing of memory, leading to memory corruption. This has been corrected.

ECN-4677 ACCEPT FROM TERMINAL-ABILITIES always returns the same information

Incident: 3232625

RPI Number: 1120755

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: 10.3.1 and later

DESCRIPTION:

When executing ACCEPT FROM TERMINAL-ABILITIES, the runtime always returned the same lines and columns information. This has been corrected.

ECN-4680 Incorrect results from SORT table WITH DUPLICATES IN ORDER

RPI Number: 646539

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

Processing a SORT table that used the WITH DUPLICATES IN ORDER phrase sometimes caused records to be returned in the wrong order, and occasionally resulted in a memory violation. This has been corrected.

ECN-4681 Memory Access Violation (MAV) when creating a sequential file

Incident: 3234224

RPI Number: 1120953

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: 10.2.0 and later

DESCRIPTION:

When creating a sequential file with a file name starting with "pdf" (not intended to be a PDF file), the runtime crashed. This has been corrected.

ECN-4682 Invalid handle reported on a .NET object return

Incident: 3232675

RPI Number: 1120880

Product: ACUCOBOL-GT

Module: CLRControlModule.dll

Machines Affected: Windows

Known Versions Affected: 10.3.1

DESCRIPTION:

When a .NET method returned an object handle, that handle was treated as invalid by the runtime. This has been corrected.

ECN-4683 C$DELETE returns the wrong value

Incident: 3233817

RPI Number: 1120867

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: 10.3.0 and later

DESCRIPTION:

On Windows, C$DELETE failed to return a failure code if the requested file did not exist. This has been corrected.

ECN-4684 Unable to search a paged list for a string starting with a national character

Incident: 3234680

RPI Number: 1120950

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: 10.3.0 and later

DESCRIPTION:

When searching a paged list box for a string starting with a national character, the runtime ignored the character, rending the search impossible. This has been corrected.

ECN-4688 Sample 4 Control Panel is not big enough to span its children

RPI Number: 645024

Product: ACUCOBOL-GT

Module: BIS

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

A small panel on the BIS Sample 4 example did not extend down far enough to cover the Invoice Detail Request line. This has been corrected.

ECN-4689 Sample 4 boxes too small to display contents correctly

RPI Number: 628054

Product: ACUCOBOL-GT

Module: BIS

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

When the BIS Sample4 was run on Chrome, Firefox, Edge, or Internet Explorer 11 without compatibility view enabled, the boxes were not big enough to correctly display the contents. This fix corrects the problem for all listed browsers.

ECN-4690 WHEN directive causes XFD warning

Incident: 3231421

RPI Number: 1120673

Product: ACUCOBOL-GT

Module: Compiler

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

An XFD WHEN condition used in a file definition in a file that was not the first file in the source code sometimes caused the, "No matching field for condition" XFD error, and ultimately prevented the generation of the XFD. This has been fixed.

ECN-4691 Second OPEN on file "NUL" causes 93 file status code

Incident: 3235901

RPI Number: 1121058

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: 10.3.0 and later

DESCRIPTION:

Opening the special file "NUL" a second time sometimes caused a code 93 error: File previously closed with LOCK by this run unit. This has been corrected.

ECN-4692 Exception handled incorrectly, causing a crash in AcuThin

Incident: 3233396

RPI Number: 1121162

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

Occasionally when an exception is generated by a control, the control is destroyed before the exception can be handled, causing a crash. This would sometimes happen, for example, in the AcuThin client when the connection had a high level of latency. The problem has been corrected.

ECN-4693 MAV on entry-field validation on 64-bit Windows

Incident: 3235411

RPI Number: 1121182

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows 64-bit

Known Versions Affected: All

DESCRIPTION:

On 64-bit Windows machines only, when a value provided for an entry field was outside of the bounds set by a MIN-VAL and/or MAX-VAL property on that entry field, the runtime would crash with a MAV. This has been fixed.

ECN-4694 XML-TEXT of 0 length causes XML PARSE to crash

Incident: 3238170

RPI Number: 1121267

Product: ACUCOBOL-GT

Module: Compiler, Runtime

Machines Affected: All

Known Versions Affected: All with XML PARSE

DESCRIPTION:

In cases where XML-TEXT had a 0 (zero) length - for example, when an attribute was empty - and XML-TEXT was referenced when executing an XML PARSE exception procedure, this caused the runtime to

crash with a parameter error. In addition, this scenario prevented the debugger from examining the XML-TEXT variable. These problems have been fixed.

ECN-4696 Double-byte character display misalignment

Incident: 3232115

RPI Number: 1120711

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: 10.1.0 and later

DESCRIPTION:

The UNICODE support introduced in version 10.1.0 modified the way double-byte characters are displayed, after translating them to UNICODE, showing them as a single stream. In cases where a double-byte character was invalid, this caused incorrect spacing of the characters, resulting in the misalignment of text on the screen. This has been corrected with the addition of the WIN32-DOUBLE-BYTE-CHARS configuration variable.

The WIN32-DOUBLE-BYTE-CHARS configuration variable, when set to the non-default value of TRUE, emulates the runtime behavior for double-byte characters in versions 10.1.0 and earlier. This runtime behavior displays double-type characters one by one.

The default value of FALSE sets the runtime to display the entire string at once.

ECN-4699 XML GENERATE crashed with zero-length OCCURS DEPENDING ON

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: All with XML GENERATE

DESCRIPTION:

When the data item being used to generate an XML document contained an OCCURS 1 TO k times DEPENDING ON n, and n is 0, then XML GENERATE crashed with a memory access violation (MAV).

When generating an XML document using a data item described by both of the following:

* OCCURS 1 TO k times DEPENDING ON n
* The value of $n=0$

XML GENERATE crashed with a memory access violation (MAV). This has been fixed.

ECN-4701 Line sequential file error

Product: ACUCOBOL-GT

Module: acme.dll

Machines Affected: Windows

Known Versions Affected: 10.3.0 and later

DESCRIPTION:

When writing to a line-sequential file, end-of-line characters were writtn as on UNIX (LF) instead of what is standard on Windows (CR/LF).

When reading a line-sequential file, the end-of-line characters were erroneously passed to the caller, which normally expects LF characters.

These issues have been resolved.

ECN-4702 PDF corrections

Incidents:

> 3242354
> 3240189

RPI Numbers:

> 1121510
> 1121713

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: All

Known Versions Affected: All with PDF printing

DESCRIPTION:

The following issues with PDF printing have been corrected:

- When using WRITE AFTER *nn* LINES, the PDF printer ignored the count and wrote just the line.
- The second and following pages of a PDF printout each had one additional line as compared to the first page.

ECN-4703 Thin Client automatic update support for older versions

RPI Number: 646569

Product: ACUCOBOL-GT

Module: Thin Client

Machines Affected: Windows

Known Versions Affected: Thin Client earlier than release 10.4.0

DESCRIPTION:

Some previous versions of the Thin Client were not automatically updated during a version upgrade as follows:

```
Upgrade To        Upgrade From Version
Version           10.3.1       10.0.0 - 10.3.0   9.2.5 and earlier
---------------   -----------  ---------------   -----------------
10.4.0            Yes          Yes               Yes
10.3.1                         No                No
10.0.0 - 10.3.0                                  No
```

To update your version of Thin Client:

1. Place the 10.4.0 AcuThin installer file, `extend(R) Thin Client Version 10.4.0.msi`, onto a server machine that is accessible from the AcuThin client machine.
2. Set the TC_INSTALLER_SERVER_FILE runtime configuration variable to the location of the `extend(R) Thin Client Version 10.4.0.msi` file. For example:

```
TC_INSTALLER_SERVER_FILE
C:\acu1040\extend(R) Thin Client Version 10.4.0.msi
```

This updates your version of Thin Client to 10.4.0, which takes effect the next time you run it.

ECN-4704 NETBOJECTS sample fails with "invalid handle"

Product: ACUCOBOL-GT

Module: Runtime

Machines Affected: Windows

Known Versions Affected: 10.2.0 and later

DESCRIPTION:

Using some .NET controls in the SCREEN SECTION resulted in an "Invalid handle" error when they were modified. This has been corrected.

AcuSQL ECN List

This section includes the ECNs relating to AcuSQL:

ECN-SQL163 Numeric data cannot be written when DECIMAL-POINT IS COMMA set

Incident: 3232600

RPI Number: 1120784

Product: AcuSQL

Module: esqllib

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

When DECIMAL-POINT IS COMMA was set in the SPECIAL-NAMES section, the AcuSQL runtime would sometimes try to send numeric data in a format not handled by the database receiving the data. For example, data in the format `123456,78` is not handled by many databases. This has been corrected.

AcuToWeb ECN List

This section includes the ECNs relating to AcuToWeb:

ECN-AW156 Display and Print updates

RPI Numbers: 1120304, 1120332, 1120444, 1120449, 1120499, 1120539, 1120544, 1120582, 1120583, 1120635, 1120681, 1120734, 1120758, 1120779, 1120794, 1120889, 1121109, 1120483, 1120509, 1121471, 1121466, 1121456, 1121514, 1121534, 1121544, 1121590, 1121575, 1121655, 1121679, 1121690, 1121763, 1121810, 1121811, 1121863, 1121895, 1121900, 1121933, 1121971, 1121992, 1122012, 1122024, 1122095

Product: AcuToWeb

Module: AcuToWeb

Machines Affected: All

DESCRIPTION:

The following display and print issues have been fixed in this release:

- When displaying and accepting character screens, the focus was different from the runtime
- C$OPENSAVEBOX failed when the path contained an accented character
- Reconnection splash screen was freezing the program
- The last record on Paged Grid was not loading
- Could not change colors for Push Button
- W$KEYBUF was using an incorrect grid cell
- The space in a Check-box text was being compressed
- Bitmap was displaying incorrectly in grid.
- List-box was not showing the selected item
- Scrollbar did not scroll to last record when using the mouse
- C$LIST-DIRECTORY did not get creation date of file on client
- Incorrect data in EVENT-DATA-2 from MSG-GOTO-CELL
- Paged grid displayed wrong and empty row at the bottom of a grid
- Unable to disable date entry control
- Events were being handled differently when using Safari or IE
- Grid cells, when right-aligned, did not display correctly
- Grid sometimes displayed an empty row at the bottom
- Adding a bitmap to a grid header did not work properly
- Drawing a large window sometimes caused performance issues
- Tree-View was not generating MSG-TV-EXPANDING event correctly
- MSG-BITMAP-CLICKED event was not being returned
- Incorrectly displayed the < and > characters in a grid
- Copy/Paste did not work on a grid cell in edit mode
- Call w$menu using wmenu-popup was not working properly
- Angled fonts did not work properly when using AcuToWeb Desktop
- An alignment issue occurred when using the grid with multiple rows
- A socket disconnect occurred when a program was idle for a period of time
- The gateway.log has been improved, and now contains more information
- The Euro symbol (€) was not supported
- Grid headers with dividers were not displaying properly
- A grid with two row headers did not display the same as when run with Thin Client
- Graphical item are displayed poorly when the window is scrollable
- The Euro symbol (€) was incorrect in a message box
- The grave accent for "a" was inverted in the grid
- Grid header columns were not aligned correctly

- Message box was displaying inaccurate French translation
- Radio Button with bitmap was not displaying correctly
- Program hung when Entry-Field had EXCLUDE_EVENT_LIST
- The Euro symbol (€) was not drawn correctly in a WRITE/READ/DISPLAY sequence
- Entry Field with NUMERIC property did not allow "/", "*" or "+"
- CSS file had no impact on push buttons on message boxes
- A new row was not added automatically when NUM-ROWS was -1
- The MSG-FINISH-ENTRY event sometimes failed to be triggered
- The Comma (**,**) key on the number pad acts like the **Enter** key
- Combo boxes showed inconsistent value highlighting
- The SockJS connection was erratic

ECN-AW158 Error when executing AcuToWeb alias within an iFrame

RPI Number: 1121621

Product: AcuToWeb

Module: AcuToWeb

Machines Affected: All

DESCRIPTION:

Executing the AcuToWeb alias within an iFrame of a main .NET application caused the following error: "Uncaught DOMException: Permission denied to access property "WT_ENABLE_DEBUGGER" on cross-origin object". This has been fixed.

AcuXDBC ECN List

This section includes the ECNs relating to AcuXDBC:

ECN-XD128 AcuXDBC installer does not deploy requisite Visual C++ libraries

Incident: 3233733

RPI Number: 1120865

Product: AcuXDBC

Module: AcuXDBC

Machines Affected: Windows

Known Versions Affected: 10.3.0 and 10.3.1

DESCRIPTION:

After installing AcuXDBC, when attempting to run the `asql.bat` command with the `-s` option, a `DLLLOAD` error sometimes occurred. For example:

```
asql -s <server-name>
***** ERROR: DLLLOAD: acuxdbc03, The specified module could not be found.
```

In this example, the error is somewhat misleading in that `acuxdbc03.dll` does exist, but the Microsoft Visual C++ libraries, on which it depends, do not. Therefore, the AcuXDBC installer did not install the correct version of the Microsoft libraries. This is fixed in the 10.4.0 release.

As an alternative workaround, you can download Microsoft Visual C++ 2012 Redistributable from the Microsoft website, and install the 32-bit or 64-bit version: `vcredist_x86.exe` or `vcredist_x64.exe` respectively.

ECN-XD129 AcuXDBC 64-bit msi silent install not updating the registry

Incident: 3227402

RPI Number: 1120241

Product: AcuXDBC

Module: AcuXDBC

Machines Affected: Windows

Known Versions Affected: 10.3.0 and 10.3.1

DESCRIPTION:

When using the silent install of the AcuXDBC 64-bit msi, it did not create the registry settings. This has been corrected.

Boomerang ECN List

This section includes the ECNs relating to Boomerang:

ECN-BMRG002 Boomerang compiler shows wrong line number or gives a memory fault

Incidents:

    3133321
    3133321
    3230918

RPI Numbers:

    1119524
    1112069
    1120594

Product: Boomerang

Module: Boomerang (server) and Compiler (client)

Machines Affected: All

Known Versions Affected: All

DESCRIPTION:

On some systems, when the Boomerang client sent a file to the Boomerang server to be preprocessed, the Boomerang server produced a memory fault due to small system stack space. Boomerang has been updated to accommodate small stack space.

On other systems, when the original COBOL program contained an error, the compiler didn't report the correct line number of the statement in error. This happened because the Boomerang server did not correctly insert the Boomerang line directives in the preprocessed file ,or because the compiler reported the line prior to the statement in error.

Both of these issues are resolved in this release.

Xcentrisity-BIS ECN List

This section includes the ECNs relating to Xcentrisity-BIS:

ECN-BIS001 Debugger sometimes attempts to start with a Debug tag

RPI Number: 581510

Product: Xcentrisity-BIS

Module: UNIX Request Handler

Machines Affected: UNIX

Known Versions Affected: 9.0.0

DESCRIPTION:

Occasionally, a BIS service program would not run, returninga vague error message about the service failing to run. Also, the debugger would start and then terminate in the X-windows session. This has been fixed.

ECN-BIS002 UNIX Debug tag's TYPE keyword accepted only lower-case values

RPI Number: 581511

Product: Xcentrisity-BIS

Module: Request Handler

Machines Affected: All

Known Versions Affected: 9.0.1

DESCRIPTION:

The BIS Debug tag TYPE keyword accepts one of the following values:

    xterm
    term
    acuthin

The Windows BIS accepted these values in mixed case; however, the UNIX version accepted only lower-case values. The UNIX version has been updated to also accept mixed-case values.

ECN-BIS003 BISMKAPP crashed on startup when decimal point was comma in the current culture

RPI Number: 647277

Product: Xcentrisity-BIS

Module: BISMKAPP

Machines Affected: Windows

Known Versions Affected: All

DESCRIPTION:

BISMKAPP crashed when launched on a machine that used a locale/culture that used a comma instead of a decimal point in decimal numbers. This includes "fr-FR", where the problem was first reported. This has been corrected.

Updates and SupportLine

Our Web site provides up-to-date information 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:

- Product Updates on *Software Licenses and Downloads*, where you can download fixes and documentation updates.

  1. Log into the Software Licenses and Downloads (SLD) site at *https://sld.microfocus.com/mysoftware/download/downloadCenter* .
  2. Select your account and click **Entitlements**.
  3. Search for the product by using any of the available search parameters.
  4. Click **Show all entitlements**.
  5. Click **Get Software** in the Action column for the product you want to download or update.

     In the **File Type** column, you see entries for "Software" for any GA products, and "Patch" for any patch updates.
  6. Click **Download** on the relevant row.
- The *Examples and Utilities* section of the Micro Focus SupportLine Web site, including demos and additional product documentation. Go to *https://supportline.microfocus.com/examplesandutilities/index.aspx*.
- The *Support Resources* section of the Micro Focus SupportLine Web site, that includes troubleshooting guides and information about how to raise an incident. Go to *https://supportline.microfocus.com/supportresources.aspx*

To connect, enter *https://www.microfocus.com/en-us/home/* in your browser to go to the Micro Focus home page, then click **Support & Services** > **Support**. Type or select the product you require from the product selection dropdown, and then click **Support Login**.

> **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, *https://www.microfocus.com/support-and-services/contact-support/*. 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.

Also, visit:

- The Micro Focus Community Web site, where you can browse the Knowledge Base, read articles and blogs, find demonstration programs and examples, and discuss this product with other users and Micro Focus specialists.
- The Micro Focus YouTube channel for videos related to your product..

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 .

Copyright and Disclaimer