Liant v12 WS2 (v12.06) Supplement

This document describes, for each component of the Liant product set, the changes made in the v12 Web Sync 2 (WS2 or v12.06) of the product. In most cases, the changes were the result of a reported product incident (RPI) and in those cases the RPI number is provided for reference.

BIS+WOW Thin Client Accept/Display

No changes.

Windows 7 Support

Version 12 WS2 (v12.06) has been successfully tested on Windows 7. Some Windows 7 related issues remaining in the v12.05 product have been fixed in v12.06.

InstantSQL for RM/COBOL

RPIs:

1072711 — Fixed an InstantSQL defect that caused a runtime crash when using bound columns or parameters. The defect used the wrong CodeBridge conversion handle (conversion buffer pointer) for conversion of bound columns and parameters. The defect was masked by the v11 and earlier implementation of CodeBridge, but was revealed by v12 changes to the CodeBridge implementation.

1073475 — Fixed an InstantSQL defect that sometimes caused a runtime crash when using the SQL BIND COLUMN or SQL BIND PARAMETER statements with a large number of columns or parameters. The implementation failed to provide a GIVING or RETURNING argument in a generated argument list. When the generated list was allocated at the beginning of a memory page, a memory access violation could occur when CodeBridge tried to save the non-existent GIVING/RETURNING argument.

Relativity Data Client

RPIs:

573539 — Fixed a problem with the 64-bit data client installation where the data server defined during installation was not available to the server administration control panel applet, thus requiring the data server to be defined again after installation.

581779 — Fixed an installation problem for the Relativity Data Client when the installation is invoked in maintenance mode.

Relativity Data Manager

RPIs:

559726 — Fixed a problem with the global configuration control panel applet on Windows Vista and later versions of Windows that caused the "Program Compatibility Assistant" to ask "Did this control panel work correctly?" The question is no longer asked because a manifest has been added to the control panel applet.

573730 — Fixed an installation problem with the global configuration control panel applet on 32-bit Windows Vista (and later versions of 32-bit Windows) where the applet was not available in v12.05.

1069809 — Fixed a problem in computing the record length when updating a row based on a variable length record. The incorrect record length would cause a 44,7 error or would silently change the record length on relative or indexed files to a potentially incorrect value. The error has been eliminated by correctly calculating the length for the rewrite of the record connected to the updated row.

1075580 — Fixed a byte-order issue that caused big-endian machines (non-Intel memory byte order, LSB in low-order byte) to handle the IF scalar function incorrectly. This affected such big-endian memory architecture platforms as AIX and HP; Intel memory architecture machines (LSB in high-order byte) were not affected by this defect.

1075794 — Fixed a defect that caused an INNER JOIN of two tables, both of which are connected to sequential files, to return a single row when more rows should be returned by the query. Now all the expected rows in this case are returned.

1075939 — Fixed a defect that caused incorrect results from a GROUP BY clause in version 12.05 where version 12.04 obtained the correct results. An oversight in the fix for RPI 1067089 in WS1, another GROUP BY defect, caused this regression.

Relativity Data Server

RPIs:

573352 — The Relativity Data Server does not support using a driver name other than "Relativity Server", but the installation incorrectly allowed the user to change it. There is no evidence that customers actually want this feature, so the Data Server installation was changed to prevent the situation.

581770 —Fixed an installation problem that caused "New Feature" to be displayed where "Query Plan Viewer" should have been displayed. The correct feature name is now displayed.

1069809 — Fixed a problem in computing the record length when updating a row based on a variable length record. The incorrect record length would cause a 44,7 error or would silently change the record length on relative or indexed files to a potentially incorrect value. The error has been eliminated by correctly calculating the length for the rewrite of the record connected to the updated row.

1075580 — Fixed a byte-order issue that caused big-endian machines (non-Intel memory byte order, LSB in low-order byte) to handle the IF scalar function incorrectly. This affected such big-endian memory architecture platforms as AIX and HP; Intel memory architecture machines (LSB in high-order byte) were not affected by this defect.

Relativity DBA

No changes.

Relativity Designer

No changes.

RMNet

Updated RMNet to use more recent cURL and OpenSSL libraries. The cURL library is now 7.21.2 and OpenSSL is 0.9.8o on all platforms on which RMNet is supported.

The TempConvert.cbl example program for RMNet has been slightly enhanced to be a better example.

RPIs:

1077493 — Added two new functions, NetSetSSLKey and NetSetKeyPassword, to RMNet.

RM/COBOL

Language Reference Manual erratta:

On page 204, Arithmetic Expressions, Table 20, the first row is incorrect as shown. The first row should contain hash, solid, hash, hash, solid. That is, the second symbol after an operand cannot be an operand, can be an arithmetic operator, cannot be a unary + or -, cannot be an opening parenthesis and can be a closing parenthesis.

RPIs:

578683 — Fixed a license management issue that caused out-dated updates to be left in the license vault when a new update is added to a base license. The license management code was changed to remove existing matching UPDATE amendments from the license vault before inserting new ones.

1067087 — Fixed a CodeWatch editor problem when using "Replace In" set to "Selection". Text on the remainder of a line at the end of the selection is deleted when text in the selection is replaced. This was a regression from v11.01 and is now fixed.

1072668 — CodeWatch editor has been fixed for a problem with the 'cut' operation when characters are selected past the last actual character in the line. .

1072678 — Fixed a CodeWatch editor problem that caused incorrect coloring of literals in the COBOL source program.

1072930 — Fixed a CodeWatch problem that made opening copy files inconvenient because of the File Tab jumping to the expanded Program Files node for a program that uses the copy file and then selecting the copybook in that location. Now the File Tab will remain in the "Copy Files" area for the double clicked copy file name as it did in version 11. In addition, the Source Files and Copy Files lists have also been moved to precede, instead of follow, the Program Files and Library Files lists to reduce shifting of the window under most circumstances.

1074065 — Fixed a problem where correct data item value queries either from CodeWatch or the built-in debugger, erroneously resulted in error messages for some identifiers.

1074083 — Fixed an installation problem that caused an error when the Windows firewall was turned off in Windows Vista or later. For Windows Vista or later a new error code was returned in this case and was not properly handled by the installation process. The error code is now interpreted properly and the installation continues instead of failing.

1074438 — Fixed a problem that caused a compiler core dump when an eight-bit (non ASCII) character was used as the currency symbol. An assertion failed in this case when scanning a PICTURE character-string where the leading character was a 'B' followed by the currency symbol

character. The test in the assertion has been fixed to handle the case of eight-bit characters.

1075726 — Fixed incorrect use counting on Windows XP Home, Windows Vista Home and Windows 7 Professional. The runtime counted uses in these operating systems as if they were multi-user server operating systems instead of the single-user workstations they are. In this case, uses were over counted, which caused legitimate (per the license) uses to be incorrectly denied.

1075652 — Fixed a CodeWatch problem that caused the Character Set setting not to be saved when choosing Apply or OK and thus not persisting between sessions. The Character Set setting is now saved and persists between CodeWatch sessions for the same workspace.

1076103 — Improved the behavior of two writers writing to the same file opened EXTEND by reducing the time delay before checking to see if the EOF has been moved by another writer. Since the RM/COBOL file manager is not part of the operating system, multiple writers to the same file opened EXTEND need to cooperate somewhat by not writing records as fast as possible, which effectively locks out other writers because the EOF keeps moving. The COBOL programs need to use C$DELAY after every write, or after a small number of writes, to the file used by other writers to give the other writers a chance to do their writes.

1076502 — Fixed an installation problem that caused an error when the Windows firewall was turned off in Windows Vista or later. Duplicate of RPI **1074083**.

1077514 — Improved the Administrative installation option of LiantInstall to pre-create a LiantInstall.ini configuration file in the target directory. This new configuration file pre-initializes the components to install, disables retrieving of activation codes and disables the License Certificate, Product Selection and Stale Certificates dialogs during subsequent Network installations.

1077559 — Fixed a compiler defect that caused incorrect code generation for an INITIALIZE statement when the subject of the statement was a group that contained two or more variable occurrence data items. The incorrect code would cause a data reference error 104 at runtime when the INITIALIZE statement was executed. The correct code is now generated in this case.

Compiler – Fixed a defect that could cause an erroneous data overflow error when nested OCCURS DEPENDING ON clauses are specified in the COBOL source program. The defect could also have caused an error in the calculation of the minimum record size for a variable length record containing nested OCCURS DEPENDING ON clauses in its data description entry.

CodeBridge – Fixed a defect in the CobolWindowsHandle data conversion function that treated the Windows handle as a signed quantity. In rare cases where the Windows handle value

included a high order bit, an incorrect SIZE ERROR condition would happen in CobolWindowsHandle.

PDFlib support has been migrated to support for PDFlib 8 (version 8.0.1). Support for PDFlib 6 and PDFlib 7 has been dropped, since licenses for those older versions are no longer available from PDFlib GmbH. On Windows, a version 8 pdflib.dll and license must be obtained from PDFlib GmbH; an existing license for PDFlib 7 and a version 7 pdflib.dll will still work on Windows using rmpdflib8.dll. On UNIX, PDFlib 8 is statically linked into the installed rmpdflib8.so in v12.06, but a license must still be acquired from PDFlib GmbH to use the full capabilities of PDFlib. There are about 18 new functions in PDFlib 8 that were not available in PDFlib 7; these new functions are not available when using rmpdflib8.dll or rmpdflib8.so. The rmclock.cbl example program has also been slightly enhanced in v12.06. PDFlib support is deprecated and will be eliminated from future releases.

RM/InfoExpress Server

No changes.

WOW Extensions for RM/COBOL

The WOW Designer was fixed to eliminate truncation of some structured comments after column 72 even when the "Generate Short (72 column) Lines" checkbox in Options/Preferences/Code tab was not selected. The designer was failing to check this setting when generating some structured comments and thus always truncated them after column 72. Now the comments will be truncated after column 72 only when the short lines option is selected. Note that the maximum line is 240 characters and thus structured comments longer than 240 characters will be truncated.

RPIs:

1066776 — The incorrect statement in the WOW Extensions User's Guide that said the TEXT-METRIC structure is defined in **windows.cpy** when it is actually defined in **txtmetric.cpy** was noted in the supplement document for Web Sync 1.

XML Extensions for RM/COBOL

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

XML Extensions has been enhanced with an XML TRANSFORM TEXT statement that transforms a document in memory to another document in memory. The three parameters for this statement are the input document pointer, the XSLT stylesheet file name and the output document pointer, in that order (the **lixmldef.cpy** file briefly documents this new statement in comments). No temporary files are written during the operation of the XML TRANSFORM TEXT statement. The memory area for the output document is allocated by the XML TRANSFORM TEXT statement and must be freed by using the XML FREE TEXT statement.

XML Extensions no longer supports raw output of XML special characters from FILLER data items; XML special characters in FILLER are now escaped as character references. This was a necessary compromise in the elimination of a temporary file for the XML EXPORT TEXT statement. XSLT stylesheets are a better method of inserting markup text into a result document. The verify sample provided with BIS has been modified from using XML special characters in FILLER to using a stylesheet and is instructive in showing the improved power of using a stylesheet over using XML special characters in FILLER. Associated with this removal of raw XML output, the output of a document prefix feature in XML EXPORT FILE and XML EXPORT TEXT was also removed.

Some examples of XML Extensions, in addition to the previously supplied samples, have been added to the installation. The examples are accompanied by a PDF document describing the examples.

A defect in detecting a "literal result element as stylesheet" (term defined in XML standards) on Windows was fixed that could have caused errors in transcoding the transformed output. The defect assumed that only stylesheets that began with an "xsl:stylesheet" or "xsl:transform" were not literal result element stylesheets. This was the wrong assumption because "xsl" is not required as the namespace prefix for stylesheets. If a different namespace prefix were used as allowed, the wrong decision would be made about the encoding of the result. MSXML6 provides literal result stylesheet transform results in UCS-2 encoding, but normal stylesheet results in the encoding specified by the output element of the stylesheet (a literal result stylesheet cannot have an output element).

A defect in transcoding characters between the Latin 9 local encoding and UTF-8 was fixed. The local character code point 0xA8 was transcoded as if it were Latin Capital U with Macron (code point 0x016a in UCS-2) when it should be Latin Small Letter S with Caron (code point 0x0161 in UCS-2). The defect was symmetric for round trips, but would give the wrong result when a round trip is not done.

RPIs:

> **578144, 578145 –** Fixed problems with an internal data structure size growing wildly large when the COBOL program defined large arrays (COBOL tables), particularly when OCCURS clauses are nested in a data structure. The memory used for the XML Extensions internal data structure in this case has been greatly reduced. The reduction was done in such a way that performance of XML Extensions should also be improved in many cases.

1073539 —Fixed a problem when exporting arrays (COBOL tables) that caused one empty occurrence to be exported when the COBOL table is empty and attributes are enabled. XML Extensions v11 did not export any elements for an empty array when attributes are enabled. The fix restored v11 behavior. Environment variable RM_XML_EXPORT_EMPTY_ARRAY was also introduced to normalize the export behavior for empty arrays:

- When set to the value 'Y', 'y' or '1', an empty array is considered non-optional in the context of the exported document and the empty first element is exported, independently of whether attributes are enabled or not enabled.
- When set to any other single character value, an empty array is considered optional in the context of the exported document the empty array is not exported at all, independently of whether attributes are enabled or not enabled.
- When not set or set to a multiple character value, the v11 behavior is implemented; that is, when attributes are not enabled, the array is non-optional and the first occurrence is exported even if empty; otherwise, when attributes are enabled, the array is optional and an empty first occurrence is not exported.

Regardless of the above, when all occurrences are enabled, all occurrences are exported whether they are empty or not empty and whether attributes are enabled or not enabled.

1074419 – Fixed a defect in XML Extensions that could cause an incorrect error 44, "wrong cobtoxml revision", when XML INITIALIZE and XML TERMINATE pairs are repeated three or more times in a loop within one run unit.

1074421 – Fixed a V12 regression in XML Extensions where a sparsely populated non-empty, group in a COBOL table would be considered empty and thus not exported unless exporting all occurrences. XML Extensions has been corrected to properly check groups to make sure they are empty before suppressing export of an empty group.

Xcentrisity® Business Information Server (BIS)

XML Extensions on Windows writes temporary files to the temporary directory defined by the GetTempPath Windows function. GetTempPath checks for the existence of environment variables in the following order and uses the first path found:

1. The path specified by the TMP environment variable.
2. The path specified by the TEMP environment variable.
3. The path specified by the USERPROFILE environment variable.
4. The Windows directory.

Note that the GetTempPath function does not verify that the path exists. If the path points to a symbolic link, the temp path name maintains any symbolic links.

XML Extensions on UNIX or LINUX writes temporary files to the temporary directory defined by the `TMPDIR` environment variable and must have file creation, write and delete permissions for this directory. If the `TMPDIR` environment variable is not defined, the temporary directory defaults to `\tmp` or `\var\tmp` depending on the specific operating system.

XML Extensions is frequently (almost always) used by COBOL service programs run by Xcentrisity® Business Information Server (BIS). Thus problems with using BIS might be the result of permissions issues with the temporary directory used by XML Extensions.

Error messages produced when running BIS applications are frequently from XML Extensions and are documented in the XML Extensions User's Guide rather than the Xcentrisity® Business Information Server User's Guide. Also, some problems in using BIS are actually problems with XML Extensions use of temporary files.

BIS has been changed to take advantage of the fact that in WS2, XML Extensions does not write temporary files when the XML EXPORT TEXT and XML IMPORT TEXT statements are used. The BIS request and response documents were previously passed in a file called the "exchange file". BIS now allows the request and response documents to be passed in in-memory strings to avoid writing sensitive information to disk, such as credit card numbers or other personal information. The XML EXPORT TEXT statement can be used to produce a response document in the COBOL service program in memory; note that the COBOL program is responsible for freeing the memory allocated by an XML EXPORT TEXT statement by using a XML FREE TEXT statement, after the response is written to BIS with B$WriteResponse. The XML IMPORT TEXT statement can be used to process an in-memory request document in the COBOL program; BIS is responsible for freeing the memory used by a request document. Passing the BIS response and request documents in memory is accomplished by changes to the B$WriteResponse and B$ReadRequest statements, respectively. A single COBOL pointer data item can be used, much like the single exchange file, but two appropriately named pointers make the program easier to read and maintain.

The B$WriteResponse BIS service function supports in-memory exchange of the response document by specifying an optional identifier of a COBOL pointer data item in the USING list of the CALL statement. The new calling sequence is as follows:

```
CALL "B$WriteResponse"
    [ USING ProgramDisposition [ ResponseDocumentPointer ] ]
    GIVING BIS-Status
```

When the ResponseDocumentPointer argument is omitted, B$WriteResponse assumes the response document is in the exchange file, as before. When the ResponseDocumentPointer argument is specified, the response document is assumed to be in the memory area referred to by the pointer value. The pointer value is normally obtained before the B$WriteResponse call by using an XML EXPORT TEXT statement to create the response document from a COBOL data structure in the COBOL service program; in that case, it is the COBOL program's responsibility to free the response memory area by using an XML FREE TEXT statement that specifies the ResponseDocumentPointer argument after

B$WriteResponse returns to the calling COBOL program. When specifying the ResponseDocumentPointer for the default program disposition, the ProgramDisposition argument may be specified as OMITTED since the ResponseDocumentPointer argument must be the second USING argument to B$WriteResponse.

The B$ReadRequest BIS service function supports in-memory exchange of the request document by specifying an optional identifier of a COBOL pointer data item in the USING list of the CALL statement. The new calling sequence is as follows:

```
CALL "B$ReadRequest"
    [ USING TimeoutInSeconds [ RequestDocumentPointer ] ]
    GIVING BIS-Status
```

When the RequestDocumentPointer argument is omitted, B$ReadRequest writes the request document to the exchange file, as before. When the RequestDocumentPointer is specified, B$ReadRequest puts the request document in memory and the RequestDocumentPointer value is set to point to that memory, replacing whatever value the pointer data item had before the call. The pointer value is then normally used in an XML IMPORT TEXT statement to obtain the request information into a COBOL data structure for further processing by the COBOL service program. The memory area referred to by RequestDocumentPointer after the call to B$ReadRequest belongs to BIS and the COBOL program must not attempt to free that memory. When specifying the RequestDocumentPointer for the default timeout value, the TimeoutInSeconds argument may be be specified as OMITTED since the RequestDocumentPointer argument must be the second USING argument to B$ReadRequest.

Note: The B$Exchange BIS service function has been deprecated and has not been enhanced to support in-memory document exchanges. Calls to B$Exchange can be replaced by a call to B$WriteResponse followed by a call to B$ReadRequest. B$Exhange can still be used, but only when using an exchange file containing the response and request documents. B$Exchange will be removed in a future release.

The BIS installation now (as of WS2) includes a tutorial and accompanying examples that are instructive in how BIS can be used. It is strongly recommended that users new to BIS review the tutorial, but even experienced BIS users will find the tutorial helpful and give them new insights into BIS.

The BIS samples have been modified to demonstrate the in-memory exchange of response and request documents. Further, the verify sample has been changed to demonstrate choosing the display language based on the browser preferred language setting (only English and Spanish are demonstrated to keep the sample brief enough for easy understanding). The ASP sample client for driving sample 4, a web services example, has been enhanced to have better error processing and provide an improved user interface experience, particularly when the session is terminated.

The stylesheets provided for web services have been changed to use the more popular Document/literal wrapped style of SOAP, where they previously used the RPC/encoded style of SOAP. The RPC/encoded stylesheets are still provided, but the names of those stylesheets have been changed to include "_rpcencoded" in the name. The stylesheets have also been simplified through the use of XSLT

parameters that can be set from the COBOL service program because of the v12 addition of the XML SET XSL-PARAMETERS statement in XML Extensions. The BIS web services samples (samples 3 and 4) have been modified to work with the new stylesheets.

Some examples of BIS usage, in addition to the previously supplied BIS samples, have been added to the installation of XML Extensions for RM/COBOL, including some purely XML Extensions examples. The examples are accompanied by a PDF document describing the examples.

RPIs (also see RPIs for XML Extensions):

580776 — The version 12.06 BISMKDIR now properly configures application pool identities in IIS v7.5 (the IIS version delivered on Windows 7 Professional and Windows 2008 R2).   BISMKDIR now has a new option for setting the application pool name: /POOL32 *poolname*. If "-" is specified as the *poolname*, then "BISv12-Application-Pool" is used as the application pool name. For IIS 7.5, credentials for a user named "AppPool*poolname* are added to the list of users that are permitted to run BIS applications and access the BIS files. Previously, only IIS_IUSERS and IUSR were granted these permissions.   A new button to add application pools is now on the "Create Xcentrisity BIS 12 Virtual Directory" dialog box; an existing application pool may be selected from the drop down list. The dialog varies depending on the version of IIS on the system where BIS is installed.
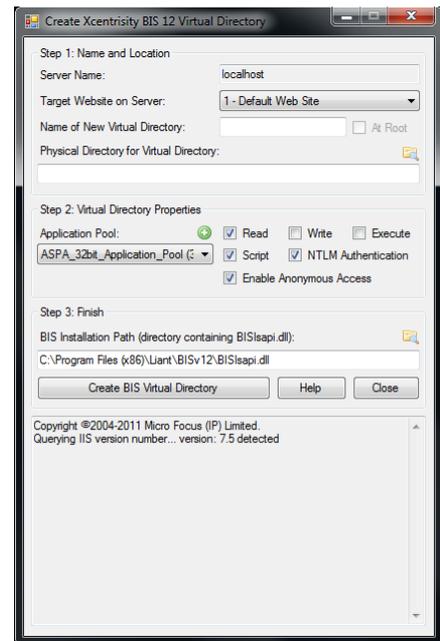
1077423 -- Fixed a problem in the **cobol_to_wsdl.xsl** and **cobol_to_wsdl_rpcencoded.xsl** stylesheets that caused the generated WSDL to contain "--name" items as output parameters of a method that should not be included. The naming convention intended such "--name" items only as a way to re-name items.

Liant v12 WS1 (v12.05) Supplement

This document describes, for each component of the Liant product set, the changes made in the v12 Web Sync 1 (WS1 or v12.05) release of the product. In most cases, the changes were the result of a reported product incident (RPI) and in those cases the RPI number is provided for reference.

Windows 7 Support

Version 12 WS1 (v12.05) has been successfully tested on Windows 7.

BIS+WOW Thin Client Accept/Display

RPIs:

1069650 —Version 12 WS1 (v12.05) BIS+WOW has removed several dlls from the sample cabinets due to problems with missing Microsoft "c" libraries used by those dlls, which implement Thin Client Accept/Display. The BISplus_HOWTO.doc/pdf contains details of this change.

InstantSQL for RM/COBOL

RPIs:

1069429 —Fixed a defect in InstantSQL, when using a MySQL database on Linux, that caused "glibc detected free(): invalid pointer: 0x08086f2e". There was a conflict between external symbol names in the RM/COBOL runtime executable and the myodbc shared object. The problem has been resolved by changing the conflicting symbols in the RM/COBOL runtime to be local (static) symbols.

1069445 — Added the InstantSQL documentation, a ZIP file containing the HTML help, to the RM/COBOL development system InstantSQL directory. The file can be unzipped and then browsed by double clicking the `index.htm` file in the directory where the zip file is expanded. (NB: The zip file is also available at, and can be downloaded from, the following URL: http://downloads.microfocus.com/liant/download/isqlhelp.zip).

1072711 — Fixed an InstantSQL defect that caused a runtime crash when using bound columns or parameters. The InstantSQL defect used the wrong CodeBridge conversion buffer pointer for conversion of bound columns and parameters. The defect was masked by the v11 and earlier implementations of CodeBridge, but was revealed by v12 changes to the CodeBridge implementation to fix memory leaks caused by CodeBridge.

Relativity Data Client

RPIs:

559654 — (not fixed yet; Relativity Data Client segmentation fault when called from InstantSQL on HP-UX PA-RISC systems. On systems other than HP-UX PA-RISC, RPI **1066803** fixed problems with using the Relativity Data Client from InstantSQL).

559679 — Fixed the installation of the Relativity Data Client on Windows to have digitally signed executables.

559727 — Fixed a problem with the server configuration control panel applet on Windows Vista and later versions of Windows that caused the "Program Compatibility Assistant" to ask "Did this control panel work correctly?" The question is no longer asked because a manifest has been added to the control panel applet.

559728 — Created a 64-bit version of the server configuration control panel applet for 64-bit versions of Windows. A 64-bit version is necessary to create the correct registry entries for the 64-bit Data Client introduced in this release.

1066803 — Fixed `SQLDriverConnect` in the Relativity Data Client on some UNIX and Linux platforms to not return "success" without doing any work unless `SQL_DRIVER_NOPROMPT` is specified. The data client would crash when a later operation tried to use the uninitialized connection. (NB: SQLConnect worked in the data client, but InstantSQL uses `SQLDriverConnect` with a default of `SQL_DRIVER_COMPLETE` and thus demonstrated the data client failure on the platforms where the problem existed.)

1066963 —Created a 64-bit version of the Relativity Data Client for Windows, which can then be used with 64-bit versions of other database software, such as 64-bit SQLServer, on 64-bit versions of Windows. (The Relativity Data Server remains a 32-bit application and thus no additional database capacity is gained by using the 64-bit client to talk to the server.)

1072464 —Fixed a license certificate issue with UNIX Data Server licenses. The issue incorrectly prevented using LiantInstall to install the Relativity Data Client on Windows. The fix has been available in licenses issued since January 5, 2010, when the certificate on the license server was corrected. For UNIX licenses issued prior to January, 5, 2010, the Relativity Data Client could be installed on Windows simply by opening the MSI for the data client.

Relativity Data Manager

RPIs:

559675 — Fixed the installation of the Relativity Data Manager on Windows to have digitally signed executables.

559726 — Fixed a problem with the global configuration control panel applet on Windows Vista and later versions of Windows that caused the "Program Compatibility Assistant" to ask "Did this control panel work correctly?" The question is no longer asked because a manifest has been added to the control panel applet.

559729 — The Relativity control panel icon for the DBMS configuration control panel applet is hidden on 64-bit versions of Windows XP, Windows Vista and Windows Server 2008 in a 32-bit Control Panel. On 64-bit XP, the 32-bit Control Panel is opened from the 64-bit Control Panel with the "View x86 Control Panel Icons" icon. On 64-bit Vista and Windows Server 2008, the 32-bit Control Panel is opened from the 64-bit Control Panel with the "View 32-bit Control Panel Items" icon. This release has no 64-bit Data Manager and thus no 64-bit DBMS configuration control panel applet. (NB: See Relativity Data Client in this document for information about the 64-bit Data Client and 64-bit Server Administrator control panel applet.) On 64-bit versions of Windows 7, the 32-bit control panel items are no longer hidden and are shown in the Control Panel with the designation "(32-bit)" appended to the name.

1067065 — Fixed a defect from both v11 and v12 where repeating a certain query multiple times could crash the data manager (or the data server).

1067089 — Fixed a defect with the implementation of the SQL statement **SELECT … GROUP BY** when applied to a Relativity table based upon multiple COBOL files where an incorrect, and sometimes infinite, result set was returned for the query.

1070016 — Fixed a defect in installation of the data manager that caused the installation to fail when the Micro Focus file manager library (DLL) and its default version could not be found. The error is now simply logged and the installation continues. If the user then attempts to enable Micro Focus file support during installation, the user can locate the library manually with the browse capability on the enablement page.

Relativity Data Server

RPIs:

559039 — Fixed the installation to create registry value named **SvcConfigDir** instead of the misspelled registry value **SvnConfigDir**.

559676 — Fixed the installation of the Relativity Data Server on Windows to have digitally signed executables.

559726 — See discussion regarding this RPI under Relativity Data Manager.

567488 — Fixed server administrator to accept a network name containing a hyphen.

1067065 — Fixed a defect from both v11 and v12 where repeating a query multiple times could crash the data server (or the data manager).

1067089 — Fixed a defect with the implementation of the SQL statement SELECT … GROUP BY when applied to a Relativity table based upon multiple COBOL files where an incorrect, and sometimes infinite, result set was returned for the query.

1070016 — Fixed a defect in installation of the data server that caused the installation to fail when the Micro Focus file manager library (DLL) and its default version could not be found. The error is now simply logged and the installation continues. If the user then attempts to enable Micro Focus file support during installation, the user can locate the library manually with the browse capability on the enablement page.

Relativity DBA
RPIs:

559677 — Fixed the installation of the Relativity DBA on Windows to have digitally signed executables.

Relativity Designer
RPIs:

559678 — Fixed the installation of the Relativity Designer on Windows to have digitally signed executables.

Relativity JDBC Client
RPIs:

3288 (Tracker) — Fixed a problem in the Relativity JDBC Client where the PreparedStatement.setObject method would generate the error "Dynamic parameter #1 type mismatch".

RMNet

RPIs:

1066670 — Added a programatic way to configure proxy server settings for RMNet. See the description of the NetSetProxyServer function in the rmnet.pdf documentation for further information. Also added a function to disable (and enable) SSL peer certificate verification. See the description of the NetSSLVerifyPeer function in the rmnet.pdf documentation for further information.

1069427 — Included the RMNet documentation in the product as rmnet.pdf. Previously, the documentation was only available on the Micro Focus web site.

RM/COBOL

RPIs:

568496 — Fixed a problem in the implementation of the RUN-FILES-ATTR configuration record USE-PROCEDURE-RECORD-LOCK-TIMEOUT keyword that could cause a file manager hang when a non-zero value is specified for the value of the keyword.

1065891 — Fixed a CodeWatch defect in the Workspace Properties dialog. When an Overriding or Supplemental Configuration file option specified a file path with a space in the name, CodeWatch could not find the specified configuration file.

1066104 — Fixed a CodeWatch defect that caused an unhandled exception crash when Print Preview was used.

1066606 — Fixed a CodeWatch defect introduced in version 12 where a workspace with more than 511 file entries could not be opened.

1066611 — Fixed a COBOL runtime assertion failure when a LIKE relation condition specified the `CASE-INSENSITIVE` option and the regular expression contained a counted character match, that is, an *{m,n}* quantifier on a character match.

1066705 — Fixed a Linux installation defect where cpio returns "Premature end of file." on some Linux distributions.

1066788 — Fixed a `LicVerifyAll` failure when merging a runtime license into a development system installed license vault on HP/UX Itanium systems. Prior to the fix, the message *LicVerifyAll: License install of 'RM/COBOL Runtime System' failed 0xc1000011: The serial number in the license does not match the serial number of the original license.* could be

incorrectly returned.

1066808 — Fixed a COBOL runtime defect when detecting `SIZE ERROR` condition on binary data items where the `PICTURE` character-string specified more digits than supported by the allocation specified by the `USAGE`, as in `COMP-1` data items or other binary data items with a USAGE that specified an allocation override. The `SIZE ERROR` condition could be raised when it did not apply or not raised when it did apply for arithmetic results stored in such binary data items. This defect has existed since v7.

1066815 — Fixed a defect on UNIX and Linux systems that prevented loading the termcap version of the COBOL runtime because of a spelling error in the function name RM_TCSETATTRNOW.

1066909 — Fixed a COBOL compiler problem in the `REPLACE` statement and the `REPLACING` phrase of the `COPY` statement when replacing a single character (for example, an equal sign) that could cause incorrect replacement actions, including replacing that character at the end of a comment line. This also fixed a problem with multiple replacements on the same line, where a latter replacement on the line might not be done when it should, and a problem with incorrect handling of a period following `END-COPY` or `END-REPLACE` (the period is not part of the `COPY` or `REPLACE` statement when the `END-COPY` or `END-REPLACE` phrase is specified, but rather is a period following the statement; v12.02 incorrectly treated the period as part of the statement).

1066914 — Fixed a defect in the administrative install for RM/COBOL to allow installation from a Windows client onto a mapped network drive.

1072048 — Fixed a compiler defect that produced an incorrect warning (0029: W DATA RECORDS data-name not defined for file: <file-name>) about the record-name specified in the DATA RECORDS clause of a file description entry. The problem was evident when the file is properly described in two or more nested programs (for example, by using COPY statements to copy the same file control and file description entries into the two or more nested programs). There was a compiler error in resolving the file-name that implicitly qualifies the data record-name when the file-name exists in two or more nested programs. The error caused the incorrect compilation warning for all but the first file with the same DATA-RECORDS clause.

RM/InfoExpress Server

RPIs:

1066624 — Fixed a problem with I/O errors 30,21 and 90,07 being returned inappropriately when using two InfoExpress servers on the same local area network.

WOW Extensions for RM/COBOL

RPIs:

1065933 — Fixed `WowGetFocus` to allow the user to obtain the handle of an ActiveX child control.

1066530 — Fixed the WOW runtime to work on Windows 2000 systems.

1066812 — Fixed a WOW runtime defect where the scroll bars and `BackColor` on controls were not redrawn for a WOW application when the whole screen was minimized and restored.

1067022 — Fixed a WOW designer defect introduced in version 12. The CONTROL key could cause selected controls to grow larger and repeated use could cause the designer to crash.

1067027 — Fixed a WOW runtime defect with modal forms created with the click event for an edit box or static label not being modal.

1067053 — Fixed a WOW designer defect in the editor that prevented `CTRL+L` and `CTRL+D` from working in version 12 when they previously worked in version 11.

Documentation Correction — The GetTextMetrics documentation states that the TEXT-METRIC structure is defined in windows.cpy. This is inaccurate, it is defined in txtmetric.cpy.

XML Extensions for RM/COBOL

XML Extensions on UNIX or Linux writes temporary files to the temporary directory defined by the `TMPDIR` environment variable and must have file creation, write and delete permissions for this directory. If the `TMPDIR` environment variable is not defined, the temporary directory defaults to `\tmp` or `\var\tmp` depending on the system. Note that XML Extensions is used by Xcentrisity® Business Information Server (BIS). Thus problems with BIS may result from permissions issues with the temporary directory.

XML Extensions statements `XML GET TEXT` and `XML PUT TEXT` were fixed to allow binary data on Windows. Previously, binary data could possibly be truncated, for example, at a 0x1a character in the "text"). UNIX and Linux systems already allowed binary data without problems.

RPIs:

1067093 —Fixed a problem with XML EXPORT attempting to allocate more than 2 GB of memory and crashing the runtime. The data structures in XML Extensions were revised to significantly reduce the amount of memory allocated when large occurrence tables (arrays) in the COBOL program are exported.

1069607 — Fixed output of arrays to suppress empty occurrences when attributes are enabled and all occurrences are not enabled. Version 11 suppressed empty occurrences correctly, but version 12 incorrectly output all occurrences whenever attributes were enabled.

1069720 – Fixed problems with algorithms and data structures that caused excessive memory use and slow execution for both XML EXPORT and XML IMPORT when large occurrence COBOL tables (arrays) were exported or imported. The problem was particullarly evident when nested tables (tables containing tables) were used in the COBOL program.

Xcentrisity® Business Information Server (BIS)

Xcentrisity® Business Information Server (BIS) makes extensive use of XML Extensions for RM/COBOL. Error messages produced when running BIS applications are frequently from XML Extensions and are documented in the XML Extensions User's Guide rather than the Xcentrisity® Business Information Server User's Guide. Also, some problems with BIS are actually problems with XML Extensions use of temporary files, as explained in the XML Extensions for RM/COBOL topic of this document.

RPIs (also see RPIs for XML Extensions):

559705 — Fixed a request handler crash that occurred if the **IUSR** security principal is selected to serve anonymous requests, but does not have access to the directory containing the service engine programs. The crash is fixed, and the **IUSR** identity is now granted access to the service engine programs during installation and registration. In addition, other enhancements have been made to clearly explain when the **IUSR** identity should and should not be selected during installation or registration.

559768 — Fixed an installation problem on 64-bit systems running IIS 7 that prevented the application pool from being usable after the installation of BIS on those systems. IIS 7 required the application pool to be marked as allowing 32-bit applications on 64-bit systems. The installation now marks the application pool appropriately when necessary.

567867 — Fixed a problem that caused BIS to hang if a malformed IP is specified in the {{Trace}} tag of an SRF file.

1066093 — Fixed an intermittent problem where after a few days the UNIX or Linux BIS request handler would refuse incoming connections and a restart was required to allow new incoming connections to be processed.

1066143 — Fixed an "access denied" issue when executing BIS applications on non-English Windows systems such as Windows Vista and Windows Server 2003 where the "**AUTHENTICATED USERS**" security ID is translated to the locale of the system (BIS installation incorrectly used the untranslated name "**AUTHENTICATED USERS**").

1066570 — Fixed the Windows utility **BISMKDIR** to support IIS web site identifiers with a value outside the range 1 – 99. This was a prior restriction, but this release allows site ID values in the range $1 - 2,147,483,647 \ (= (2 ** 31) - 1)$.

1066938 — Fixed the BIS Windows installer to create the **XBIS12-Application-Pool** as a 32-bit application pool on 64-bit versions of Windows Vista and Windows Server 2008/2008R2 systems, which allows BIS to run successfully on these systems. The Microsoft KB article http://support.microsoft.com/?id=895976 has additional information about the issue resolved by this fix.

1067102 — Fixed the Windows utility **BISMKDIR** to properly create virtual directories for a 32-bit compatible application on 64-bit Windows systems that run IIS version 7.

1071467 — Modified xbisctl utility to provide information for BIS error 29 on a LAN (too many IP address blocks). The extra information reflects the kind of license (LAN versus WAN) and, for LAN licenses, the IP address range for the server to aid administrators in diagnosing the error 29.

1072047 —Modified the b*is/samples/common/soap_request_to_cobol.xsl* stylesheet to correctly support web service array input for Java and PHP. Previously, the stylesheet supported only the Microsoft ASP method of array input for web services as demonstrated in Sample4 of the BIS samples.