Synergy DBL
Version 10.3.3a
Release Notes
July 2016

These release notes describe changes that have been made in Synergy DBL in version 10.3. These notes also describe any last-minute corrections and modifications made after the documentation was published.

CONTENTS

IMPORTANT NOTES AND WARNINGS

-- If you are upgrading from a previous version of Synergy, we strongly recommend that you review the Quick Migration Guide and the Updating Synergy/DE manual for important upgrade information.

-- You should always check the Synergex secured web site (http://www.synergex.com) for any patches that may be available for this version. Log into the Resource Center and then navigate to the Downloads page.

Compatibility

-- Based on recommendations from the NIST (National Institute of Standards and Technology) Computer Security Division, we enhanced the DATA_ENCRYPT and DATA_DECRYPT routines for better security. We recommend that everyone using these routines upgrade to a minimum version of 10.1.1a for improved security. All prior versions of Synergy/DE (9.3.1, 9.5.1, 9.5.3, and 10.1.1) that access data encrypted with 10.1.1a or later will require a hot fix from Synergex. THIS CHANGE MAY BREAK YOUR CODE IF USED ALONGSIDE OLDER VERSIONS. [tr#34363]

-- Version 10 will run traditional code compiled and linked with Synergy/DE 6.1 and higher. You do not have to recompile unless you want to take advantage of performance improvements (or to debug if you’re coming from a version prior to 9).

-- If you have code compiled with a version prior to version 9, you must recompile in order to debug.

-- If you are using non-Synergex-provided classes and you rebuild any object module using a class, YOU MUST REBUILD all object modules using that class.

-- (.NET) If your application uses global commons, global data sections, or public class fields that are accessed across assemblies, any time one of those elements changes, YOU MUST RECOMPILE all projects that reference the assembly containing them. As an example, in version 10, the UI Toolkit globals changed, so therefore any application that uses Synergex.SynergyDE.tklib.dll must be recompiled. To ensure that this occurs, we have changed the assembly version of the tklib.dll to force a recompile. We recommend that you use assembly versioning on your dependent projects as well. (Traditional Synergy does not have this problem because names are resolved at runtime, not at build time.)

Version 10.3.3a notes and warnings

-- Use of the conditional operator could cause an "Invalid class handle" error. This has been fixed. When using -qrntcompat to target a version earlier than 10.3.3a, a PLATSUPT error is now reported at compile time in situations that previously would have generated the "Invalid class handle" runtime error. A runtime change was made for 10.3.3a to allow this to work properly. If you want this to work, you must target at least 10.3.3a (-qrntcompat=10030301). [tr#36557]

Version 10.3.3 notes and warnings

-- Dblproto now creates a single .dbp file containing prototypes for all subroutines, functions, and classes rather than a file for each routine/class. THIS CHANGE MAY BREAK YOUR CODE, as well as highlight coding errors with your imports and cause TOKUDF errors due to duplicate prototypes. See tr#36331 under "Compiler" in the "REVISION 10.3.3" section below for more information. [tr#36331]

-- The IMPORT statement no longer supports a DIRECTORY specification. You should use -qimpdir or SYNIMPDIR instead. THIS CHANGE MAY BREAK YOUR CODE. See tr#36331 under "Compiler" in the "REVISION 10.3.3" section below for more information. [tr#36331]

-- We changed the way namespaces are imported within a compilation unit. Previously, any namespace defined in a compilation unit was available to all source files in that compilation unit. In addition, a namespace that was defined outside the compilation unit and then imported with the IMPORT statement was available to all subsequent source files on the dbl command line. Now, a namespace defined in a file is available only to that file unless it is explicitly imported into other files in the compilation unit (with IMPORT) or implicitly imported with SYNDEFNS. In addition, a namespace imported with the IMPORT statement applies only to the file into which it is imported. THIS CHANGE MAY BREAK YOUR CODE. If your code relies on the old behavior, you will now get compilation errors (NFND, NVTF, NVTP, or NVTPR). To correct this, you must add IMPORT statements to your code (or redefine SYNDEFNS). We made this change so that Synergy behavior matches other .NET languages, and it enables the compiler to resolve symbols that would otherwise be ambiguous and would prevent IntelliSense and QuickWatch features in Visual Studio from working. [tr#36395]

-- Using dblproto to generate prototype files (.dbp) from multiple source files may cause duplicate symbols to be resolved differently than in past versions. THIS CHANGE MAY BREAK YOUR CODE. Previously, by default dblproto generated multiple .dbp files from each source file. If these files were output to the same directory, duplicates were simply overwritten, and the last file processed "won." Now, dblproto generates a single file from multiple source files, and the order in which files are processed means it is unlikely that the same duplicate definition will prevail as in prior versions.

It is possible to encounter a similar situation at compile time if you have prototype files with duplicate definitions and the new qrelaxed:allowdup option is set. You are most likely to encounter this issue if you run dblproto on files generated by multiple runs of gennet40 (which is NOT recommended). When you do this, the generated files include partial classes (for internal use) with the same names as fully implemented classes. The compiler might now use prototypes for partial classes where previously the prototypes for fully implemented classes were used, resulting in prototype mismatch, method missing, or property missing errors. You can avoid this by generating code for all the types you need with a single run of gennet40.

-- (OpenVMS) The minimum supported version of OpenSSL for the HTTP document transport API and DATA_ENCRYPT and DATA_DECRYPT routines is now SSL 1.4. If your OS version is 8.4 (either Alpha or I64), the default version is now the more secure SSL1 1.0-2C. For earlier versions of the OS, which do not support SSL1 1.0-2C, SSL 1.4 will be installed as the default. THIS CHANGE MAY BREAK YOUR CODE. See "OpenSSL Requirements" in the "Requirements and Considerations" chapter of the Installation Configuration Guide and Synergex KnowledgeBase article 100001979. [tr#36282]

-- Previously, if the last data field in a section was not a handle, the compiler failed to report a LBLSCOPE error ("Label %s out of scope"). This has been fixed. THIS CHANGE MAY BREAK YOUR CODE. See LBLSCOPE in the Error Messages chapter of the Tools manual for more information about what causes this error. [tr#36429]

Version 10.3.1b notes and warnings

-- An "Invalid operation" error (INVOPER) is now reported in every case where a READS, FIND(MATCH:Q_SEQ), or WRITES follows a Select, unless context has been reestablished with a READ, FIND, or WRITE statement. (Until READ, FIND, or WRITE is called in this situation, sequential context is undefined.) Previously, there was one case where this error was not reported: after using Dispose with a Select in a FOREACH statement. This has been corrected, but note that THIS CORRECTION MAY BREAK YOUR CODE. Previously, the runtime would allow your program to call READS, FIND(MATCH:Q_SEQ), or WRITES without re-establishing context after a FOREACH Select was Disposed (even though the results were undefined). Now it will report the error. [tr#35907]

-- (Windows 10) When a machine is upgraded to Windows 10, Windows firewall is automatically enabled. If you upgrade a machine that has Synergy installed and is a license server, you may need to reboot after running the license server once to reset the firewall rules. If Windows firewall was not in use before the Windows 10 upgrade, we recommend you disable it after upgrading. [tr#35881]

Version 10.3.1a notes and warnings

-- When using the Synergy HTTP document transport API, note that SSL 2 and SSL 3 protocols are not secure, and TLS 1.0 has known vulnerabilities. We recommend using TLS 1.1 or TLS 1.2 for secure sites. For more information, see Section 3.1 in

nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf

TLS 1.1 and TLS 1.2 are available on OpenSSL 1.0.1.

Version 10.3.1 notes and warnings

-- The following versions are the minimums supported in 10.3.1:

```
   AIX 6.1
   Solaris 10
   HP-UX 11i v3
```

[tr#34993, tr#34994]

-- (.NET) In rare cases issuing a STOP statement in a WinForms or low-level Windows application caused a segmentation fault. This has been fixed. You must recompile your code with 10.3.1, and the "Target Synergy runtime" option (on the Build page of Visual Studio Project Designer) must also be set to "10.3.1." If it’s set to an earlier version, you won’t get this fix. [tr#35491]

-- (Windows, UNIX) We now support a minimum revision of OpenSSL 1.0.1j on Windows and 1.x on all UNIX platforms except HP-UX, where it is 0.9.8. See Synergex KnowledgeBase article 100001979 for detailed information on OpenSSL requirements and installation. THIS CHANGE MAY BREAK YOUR CODE. [tr#33997]

-- (Windows, UNIX) Synergy/DE 10.3 on Linux and Windows 32-bit systems requires a processor with at least SSE2 instructions (i.e., AMD Athlon 64 or higher and Pentium IV or higher). For Windows, the Sysinternals Coreinfo tool can indicate this. Programs like LMU will fail to operate and the installation will fail without this instruction support. [tr#35398]

-- The performance of simple groups has been optimized. THIS MAY BREAK YOUR CODE, as it affects groups in global commons, global data sections, global literals, and global non-CLS structures, and it is not compatible with assemblies built with older versions of the compiler. As a result, all code that exposes groups in global entities across an assembly boundary needs to be recompiled. [tr#31997]

-- The minimum Framework supported is now CLR 4.0.30319. This version or higher is required in order to install Synergy. [tr#34961, tr#34965]

-- Synergy installations require at least Service Pack 1 on Windows 7 and Windows Server 2008 R2. [tr#34849]

-- We now strictly enforce that the xfServer version be at least as high as the client version.

-- The minimum supported Linux version requires a minimum of GCC4.4 compatible libstdc++ and libtinfo.so.5 and OpenSSL 1.03 to use Synergy/DE 10.3. This support was from 2010 onwards, but lower Linux versions may be able to download the required OpenSSL and libncurses (containing libtinfo) packages to enable runtime-only support. [tr#34992]

-- Because prototypes are built differently in the traditional compiler in 10.3.1 and higher, your existing prototypes must be rebuilt, or a PRTLOAD error will occur. [tr#34308]

Version 10.1.1 notes and warnings

-- Doing a FOREACH over a new instantiation of a class that implements IDisposable now causes the class’s Dispose method to be called when the FOREACH loop is exited. THIS CHANGE MAY BREAK YOUR CODE. [tr#34212]

-- (Windows) Synergy/DE is now always installed to the default Program Files\Synergex location, and this path cannot be changed during installation. Existing installations of Synergy/DE can be upgraded to version 10 and will remain in their current location. (If you need to install Synergy/DE to a non-default location, there is an undocumented way to do this via the command line. Contact Synergy/DE Developer Support for more information.)

Note that because Synergy/DE is now always installed under Program Files, the distributed sample repository files, method catalog, system catalog, and synergy.ini files may need to be copied to a writable location outside Program Files and located via the pertinent environment variables (RPSMFIL, GENESIS_HOME, SFWINIPATH, etc.) [tr#33517, tr#33646]

-- (Windows) Visual C++ 2012 libraries are now installed by Synergy/DE Client. If you are currently running a shared installation and upgrading from version 8.1 through 9.5.3, you must also upgrade the clients. If you do not wish to upgrade (or uninstall/reinstall) the clients, and your shared installation has not changed location, you can instead download the appropriate Visual Studio 2012 redistributables (vcredist) from the Microsoft website and install them on each client machine. See the version 10.1 installation instructions for more information. [tr#33272, tr#33243]

-- (Windows, UNIX) ISAMC_REV now defaults to REV 6. All new files created will be revision 6 unless the environment variable ISAMC_REV is set to a lower revision. Also, setting ISAMC_REV to 2, 3, or 5 will no longer allow creation of files at those revisions. [tr#33556]

-- In 9.5.1 and 9.5.3, the // operator was incorrectly evaluated before * and /. This has been fixed, and these three operators now have the same precedence and are evaluated from left to right. THIS CHANGE MAY BREAK YOUR CODE if you rely on // evaluating before * and /. You may need to add parentheses to explicitly specify the evaluation order. For example, if you have

```
var = a * b // c
```

you would need to change it to this to get the same result:

```
var = a * (b // c)
```

[tr#34101]

-- (Windows, UNIX) The functionality for the OPEN options ALLOC and NUMREC has been disabled. The functionality for system option #15 has been removed. [tr#33176]

-- To match behavior of .NET, an unhandled exception now executes all remaining FINALLY blocks in the call stack (including in the same frame). Also, a STOP or CHAIN no longer executes any remaining FINALLY blocks in the call stack (including in the same frame, a FINALLY on the TRY from which the STOP was issued). Note: This parenthetical case is a change in behavior and MAY BREAK YOUR CODE. [tr#33048]

-- For version 10, we have reviewed areas in our source code and published APIs that use version strings, evaluating them to ensure that the buffers used are long enough to store a two-byte version and identifying places where string comparisons are done against the version. (Our version strings are stored left-justified.) To ensure that string comparisons will continue to work in your deployed applications, some existing version strings now use alphabetic letters to represent versions 10 and higher. For example, the SYN_VER define (dbl.def) is "A.1.1" in version 10 so that existing string comparisons against earlier versions will continue to evaluate as you expect. (Otherwise, comparing "10.1.1" to "9.5.3" would result in "9.5.3" being evaluated as higher.) Because you may be passing SYN_VER to some Synergy routines, we have modified those routines in version 10 to handle both "A" and "10".

If you currently use SYN_VER, when you start developing with version 10, we recommend that you modify your code to use the new SYN_VERSION define (in dbl.def), which is "10.0.3". We also recommend that you review all version string logic in your application for version 10.

-- %VERSN returns the runtime version in a text string (e.g., "DBL/MS Windows Version v.r.m "). If you have code that parses the version number out of this string and then uses it in a string comparison against an earlier version, we recommend you review your code’s string comparison logic before you run version 10.

CHANGE HISTORY

REVISION 10.3.3a

Analysis engine

-- An incorrectly formed SET method caused the compiler to crash. This has been fixed. [tr#36525]

-- An incorrectly formed nested BEGIN caused the compiler to crash. This has been fixed. [tr#36526]

-- Dividing by zero in a compile-time expression (for example a1/0 as a type specification) crashed the compiler. This has been fixed. [tr#36527]

Compiler

-- We added support for -qdecargs to dblproto. [tr#36517]

-- Dblproto now honors the DBLMAXERR environment variable. If DBLMAXERR isn’t set, the number of errors that dblproto will generate before abandoning prototype file generation defaults to 20. [tr#36497]

-- Dblproto previously crashed when a field’s type was a%size(field2) and field2’s type was declared as a%size(field3), if field3 was not defined. This has been fixed. [tr#36501]

-- The compiler reported an unexpected TYPPARM warning when using -qrelaxed:paramad on an overloaded routine where one of the overloads exactly matched the call. This has been fixed. [tr#36502]

-- In 10.3.3, the compiler reported an unexpected NFND error when calling a subroutine without an XCALL when SYNDEFNS was set. In addition, default conversions for subroutines in the same source file were sometimes missing (for example, default string to alpha argument conversions). This has been fixed. [tr#36508]

-- Passing an object type, like a string, into an unresolved XCALL or function call could cause unpredictable results at runtime, as no conversions were done on the arguments to match the routine parameter types. The compiler now reports a level 3 W_URCALL warning when objects are passed as argument types to unprototyped routines. Note that this warning won’t appear when -qrelaxed:param is set, as this allows resolution to local external functions. [tr#36515]

-- When displaying the source line for an error from a repository include, the name of the file was not displayed when -qexport (or SLKRUNS=1) was set. This has been fixed. [tr#36519]

-- When a function call had an unresolved cast in front, a BSTMTCH error occurred. The compiler now correctly reports an INVCAST error. [tr#36523]

-- The line number was off by many lines on certain duplicate repository includes. This has been fixed. [tr#36524]

-- We now allow the use of an alpha variable with a format string when qrelaxed:paramad is specified, giving a W_NUMREQ warning. [tr#36530]

-- An INIT of an overlay record seemed to clear out the record after the INIT, giving incorrect results. The compiler now reports a NOTALLOWED error in this case. [tr#36544]

-- Passing a literal integer value to a method that takes a boxed integer no longer gives a runtime error. [tr#36550]

-- (UNIX) The compiler was not importing prototype files with uppercase file extensions. This has been fixed. [tr#36536]

-- In 10.3.3, the compiler reported an unexpected AMBSYM error when SYNDEFNS was set and a source that contained a subroutine and its call (after the routine was prototyped) was compiled. This has been fixed. [tr#36567]

-- In 10.3.3, using a struct field in a conditional operator that targeted a string did not give an error when targeting a version less than 10.3.3a. Now the compiler reports a PLATSUPT error in this situation. [tr#36558]

-- Use of the conditional operator could cause an "Invalid class handle" error. This has been fixed. When using -qrntcompat to target a version earlier than 10.3.3a, a PLATSUPT error is now reported at compile time in situations that previously would have generated the "Invalid class handle" runtime error. A runtime change was made for 10.3.3a to allow this to work properly. If you want this to work, you must target at least 10.3.3a (-qrntcompat=10030301). [tr#36557]

-- Building a traditional Synergy project in Visual Studio with a blank source file correctly reported a NOPROSEC error, but the name of the file causing the error was not included, making it difficult to locate. The name of the first source file is now reported. [tr#36516]

Debugger

-- We fixed the debugger to break at a breakpoint set on a FOREACH statement. As a consequence, an extra STEP will occur on BEGIN statements that have DATA statement(s) with object handles. [tr#36529]

-- The debugger now steps to the correct line with SET STEP OVER and a trapped error. In addition, the error number is output in the message when stepping and an error is trapped. [tr#36533]

-- BREAK <routine> <line#> worked incorrectly if <routine> was not current. This has been fixed so you can set a breakpoint in a routine that is not currently resident. [tr#36532]

Runtime

-- Using a tag key on a Join now correctly returns all possible matches when the On expression is not fully optimized. [tr#36492]

-- An RCBSTACK error previously could cause a stack overflow with TRY-CATCH or $ERR_CATCH. This has been fixed. [tr#36499]

-- When attempting to join one or more remote tables using the Select Join, and the leftmost (driving table) file is not an ISAM file, the creation of the Join object no longer seems to hang. [tr#36545]

-- When using automatic channel allocation, we now start the channel scan at the size of the variable (e.g., 999 for d3) to avoid CHNUSE errors. [tr#36534]

-- (.NET) In 10.3.3, a .NET runtime license did not share a seat with a traditional runtime license. This has been fixed. [tr#36568]

-- (.NET) In 10.3.3, setting SCSPREFETCH while running a .NET client to xfServer hung with a Select or READS. This has been fixed. [tr#36543]

-- (.NET) In 10.3.3, a Synergy .NET app running on the desktop concurrently with a traditional Synergy app consumed one additional RUN license. This has been fixed. [tr#36511]

-- (Linux) In 10.3.3, the SSL linkage was broken for httpslib.so, synssllib.so, and VTXSSL.so. This has been fixed. [tr#36569]

-- (SuSE Linux) The Redhat version of the files httpslib.so, synssllib.so, and VTXSSL.so was being installed on SuSE. This has been fixed. [tr#36570]

Synergy DBMS

-- Isutl no longer reports a "Bad RFA vector" or "snapshot may be corrupted" error when verifying a file with change tracking when a deleted record is compressed with a time key present, or a record is not compressed but follows a free snapshot (isutl -f), and the record was updated and deleted during the same snapshot prior to being freed. [tr#36431]

xfNetLink Synergy

-- In 10.3.3, xfNetLink Synergy failed to connect to an xfServerPlus server using %RX_START_REMOTE with error code RX_NOXFSPL when IPv6 protocol was not available. This has been fixed. [tr#36507]

REVISION 10.3.3

Compiler

-- We added the -qdefns and -quserdef options in the dbl compiler and dblproto to override SYNDEFNS and SYNUSERDEF, respectively. We also added -qimpdir and now allow both -qrelaxed:<option> and -qrelaxed=<option> syntax in dblproto. [tr#36338]

-- We added the -qrelaxed:paramad option, which allows passing an alpha argument to a d parameter whose direction is in or unspecified. [tr#36328]

-- We added the -qrelaxed:paramst option, which allows a non-CLS structure (without objects) to be passed as an argument to a parameter whose type is another non-CLS structure of the same size. [tr#36394]

-- We modified the compiler switch -qdebug=level, where level is the debug level. Level=1 reduces the debug emission output, while level=2 leaves it unchanged. The -d switch now defaults to level=1. With level=1, field references are only forced for those in the scope of a referenced field’s ancestry. That is, if a referenced field is within group G1 within record R1, all of the fields within the scope of R1 will be referenced (i.e., all the R1 fields, all the fields within the R1 groups, all the fields in THOSE groups, and so forth). Finally, with level=1, the source lines in the data division and its include files are no longer accessible via the debug LIST and VIEW commands. All other source emissions are unchanged. [tr#36412]

-- (Windows, UNIX) The dbl compiler, the dblink linker, and dblproto now support the -qrntcompat=<value> switch, added for compiling to a previous version of the compiler. Valid values are 90501 for 9.5.1, 90503 for 9.5.3, 100101 for 10.1.1, 10010101 for 10.1.1a, 10030100 for 10.3.1, 10030101 for 10.3.1a, 10030102 for 10.3.1b, and 10030103 for 10.3.1c. The linker -qrntcompat option ensures that the version of the object files is not greater than the version specified and that it is compatible with the targeted runtime. Note that object files created prior to this release cannot have their version verified. [tr#34357]

-- (Windows, UNIX) The dbl compiler and dblproto now support the -platform=<type> option, which was added to allow specifying 32-bit or 64-bit object file creation. <Type> can be one of the following values:

x86	32-bit. Use a size of 4 for D_ADDR.
x64	64-bit. Use a size of 8 for D_ADDR.

If -platform is not specified, the size of the D_ADDR remains at its current values (i.e., 8 on 64-bit platforms and 4 on 32-bit platforms). [tr#34357, tr#36244]

-- (Windows, UNIX) The -qdefine option from the Synergy .NET compiler is now also supported by the traditional compiler, dblproto, and dbl2xml. [tr#36249]

-- The -qrelaxed:allowdup option from the Synergy .NET compiler is now also supported by the traditional compiler. If -qrelaxed:allowdup is set, the compiler now prevents TOKUDF errors for duplicate subroutines or functions from prototypes. [tr#36314, tr#36252]

-- Dblproto now sets the following -qrelaxed options under the hood: allowdup, deprecate, interop, local, param, path. Additionally, you can set the -qrelaxed options end and extf on the dblproto command line. (Note that setting -qrelaxed=end is the same as the dblproto -qrelaxedend option.) [tr#36312]

-- You can now specify more than one namespace in SYNDEFNS (separated by semicolons) whose prototypes will be implicitly imported. In addition, you can import nested namespaces by adding a wildcard character (*) after the namespace (for example, SET SYNDEFNS=synglobal;mydefaultnamespace.*). We recommend using this format rather than SYNUSERDEF, which has been deprecated. [tr#36264]

-- Dblproto now creates a single .dbp file containing prototypes for all subroutines, functions, and classes rather than a file for each routine/class. Although it improves and greatly simplifies prototyping, THIS CHANGE MAY BREAK YOUR CODE, as well as highlight coding errors with your imports and cause TOKUDF errors due to duplicate prototypes. You must regenerate existing prototypes or you will get an error. Be sure to delete the old .dbp files first.

- The dblproto -out option now specifies a filename rather than a namespace. If not specified, the filename defaults to the first source file name on the command line. If the filename does not end in .dbp, a .dbp extension will be added.

- The dblproto -single option has been deprecated and should not be used. We recommend that you use wildcards or a complete list of source files as input to dblproto. Some customers used the -single option to improve processing time and decrease memory usage, especially on 32-bit machines. You can now use the -platform=type option (see tr#34357 and tr#36244) to target 32-bit from the 64-bit dblproto, making this use of the -single option unnecessary. If you feel you still have a need for -single, please contact Synergex Developer Support.

- Due to corrected error reporting, you may find some files now correctly report an IMPORT error ("Namespace myNamespace not found"). This either means that there’s a prototype import or generation ordering issue that needs to be fixed or the import is actually incorrect.

- Dblproto and dbl compiler performance has been significantly improved when using prototypes in conjunction with wildcard input files to dblproto.

[tr#36331]

-- The IMPORT statement no longer supports a DIRECTORY specification. If included, it will be ignored and generate a warning. Instead, specify the location of prototypes with -qimpdir when you compile or by setting the SYNIMPDIR environment variable. THIS CHANGE MAY BREAK YOUR CODE. This change was a result of a shift in the function of the IMPORT statement. Now it is used solely to enable you to use members of a namespace without having to fully qualify the namespace. IMPORT no longer causes checking of prototypes. Prototype checking now takes place automatically if you have generated prototypes and specified their location with -qimpdir or SYNIMPDIR.

As a result of this change, the -impdir option for the gennet40 utility has been deprecated.

[tr#36331]

-- (OpenVMS Itanium) Dblproto now puts private members into the .dbp file to prevent the I64/OpenVMS linker from generating an INVOVRINI error ("Incompatible multiple initializations for overlaid section"). This was a regression introduced in 10.3.1. [tr#36327]

-- Previously dblproto would not generate any prototypes for the compilation unit if an error was encountered. Instead it now reports errors but goes ahead and generates the corresponding prototypes. Note that if errors are reported, you must fix them, or you will potentially face CRC mismatch errors when linking. [tr#36271]

-- (Windows) TOKUDF errors now provide the location of the original definition of a duplicate symbol as an informational message following the error message. [tr#36073]

-- Unexpected AMBSYM errors previously occurred from imports due to concatenated source file functionality. Therefore, the compiler no longer concatenates files by default. To provide the old concatenating functionality, we added a -qconcat command line option (/CONCAT on VMS). [tr#36389]

-- A string variable passed for a non-string, writable parameter will now always generate an INVPARM error. This replaces a similar error that was previously only issued for .NET compatible compilations. Also, if a runtime-generated, read-only parameter is passed for one of these parameters, a WRTLIT error will be generated on any modification attempt. [tr#36289]

-- If an array used in an INIT has initial values, the first element is no longer simply replicated, and the entire array is directly loaded. In this case, the maximum size of the referenced array is 32767. A NOTALLOWED compiler error will be generated for the INIT if this is not the case. [tr#36008]

-- Previously the compiler crashed when it was running out of disk space while writing to the .dbo file. This has been fixed. [tr#36176]

-- Comparing a boxed boolean variable to an alpha using the .IS. operator incorrectly returned "true." The .IS. operator has been fixed to map boolean to integer. [tr#36201]

-- Prototyping a file with a duplicate structure with one or more unnamed fields caused an unexpected PROTOMISMC compilation error later. This has been fixed. [tr#36318]

-- In 10.3.1, public class records containing fields marked

private would not prototype correctly. This has been fixed.
[tr#36323]

-- The compiler previously reported an unexpected NFND error on an XCALL of a subroutine (not defined locally) within the nested scope of an overloaded method. This has been fixed. [tr#36367]

-- The compiler no longer reports an unexpected AMBSYM error for structure resolution. [tr#36386]

-- We changed the way namespaces are imported within a compilation unit. Previously, any namespace defined in a compilation unit was available to all source files in that compilation unit. In addition, a namespace that was defined outside the compilation unit and then imported with the IMPORT statement was available to all subsequent source files on the dbl command line. Now, a namespace defined in a file is available only to that file unless it is explicitly imported into other files in the compilation unit (with IMPORT) or implicitly imported with SYNDEFNS. In addition, a namespace imported with the IMPORT statement applies only to the file into which it is imported. THIS CHANGE MAY BREAK YOUR CODE. If your code relies on the old behavior, you will now get compilation errors (NFND, NVTF, NVTP, or NVTPR). To correct this, you must add IMPORT statements to your code (or redefine SYNDEFNS). We made this change so that Synergy behavior matches other .NET languages, and it enables the compiler to resolve symbols that would otherwise be ambiguous and would prevent IntelliSense and QuickWatch features in Visual Studio from working. [tr#36395]

-- (Windows) You can now use .IFDEF and .IFNDEF for conditional compilation based on a global enumeration with a particular name. [tr#36072]

-- (Windows) For TOKUDF errors, the location of the original definition of the duplicate symbol is now provided in the information that follows the error message. [tr#36073]

-- Previously, the compiler incorrectly reported an RFAERR when an arrayed parameter was used as an RFA value and when it encountered ^M and function calls. These issues have been fixed. [tr#36174]

-- You can now use duplicated structures when prototyping or when -qrelaxed:allowdup is specified in the "Other options" field on the Compile page of project designer. [tr#36254]

-- Declaring a field of a structure type whose name is the same as a local group no longer causes an AMBSYM error. [tr#36258]

-- Previously, if a field defined using ^SIZE referenced another field in the same structure, the compiler reported an unexpected AMBSYM error when that structure field name was repeated within another structure in the same scope. This has been fixed. The nearest scoped field is now chosen in this situation. [tr#36260]

-- The compiler now handles implicit closures correctly (e.g., for an "ENDSUBROUTINE expected" warning) when it encounters IMPORT, START, and global STRUCTURE statements. [tr#36326]

-- The compiler no longer crashes if a function is used on the left side of an assignment. The compiler now generates an error for this situation. [tr#36388]

-- (Windows, UNIX) -qimpdir="quoted string" did not work from a redirected input file. This has been fixed. If you have multiple paths, either quote the entire string or use multiple -qimpdir commands. [tr#36401]

-- Previously, using a type from the same namespace within another source reported an unexpected NVTF error ("Cannot resolve type for field"). Now, for namespaces declared locally within a source, the compiler automatically imports prototypes from that namespace. [tr#36411]

-- The compiler now reports a W_TYPPARM level 3 warning when passing an argument is satisfied using either the -qrelaxed:paramad or -qrelaxed:paramst option. [tr#36417]

-- Previously, dynamic [#] arrays of an enum type were incorrectly set to i1 rather than i4 values. This has been fixed. [tr#36419]

-- The compiler no longer reports an unexpected NOTCEXP error ("Not a compile-time expression") when using an enumeration value from another source compiled later as the initial value of an enumeration field in an earlier source. [tr#36415]

-- Dblproto no longer crashes if a problem such as an invalid name or a protection issues occurs when attempting to rename the temporary file to the final .dbp filename. [tr#36421]

-- Previously, a runtime error occurred when a simple property was a member of a class. This has been fixed. [tr#36420]

-- An error on the first compiled line following uncompiled lines in an .INCLUDE file was previously reported on the wrong line. This has been fixed. [tr#36375]

-- Previously, if the last data field in a section was not a handle, the compiler failed to report a LBLSCOPE error ("Label %s out of scope"). This has been fixed. THIS CHANGE MAY BREAK YOUR CODE. See LBLSCOPE in the Error Messages chapter of the Tools manual for more information about what causes this error. [tr#36429]

-- Previously, using .NOPROTO around a non-private method within a class caused an unexpected PROTOMISMCH error when compiling source against its prototype. This has been fixed. [tr#36430]

-- The -qrelaxed:EXTF option did not work with prototypes. This has been fixed so that an error is no longer reported. [tr#36391]

-- An unexpected PROTOMISMCH error previously occurred when using a prototype of a class whose member used a dynamic array of type short that was preceded by a fixed array of type short. This has been fixed; dblproto no longer generates an incorrect prototype. [tr#36444]

-- Related classes and strings could report unexpected "Type mismatch between %s and %s on each side of the ':' in a ternary operator" errors (TERNMSMCH). This has been fixed. [tr#36459]

-- An unexpected NOTCALLABLE error occurred when calling a string function without function call syntax. This has been fixed. [tr#36315]

-- A define with a value wasn't getting applied from -qdefine when the define was second or later in the list. For example, -qdefine=DOFLD1,F2TYPE=a10,DOFLD3 didn’t apply F2TYPE as a10. This has been fixed. [tr#36467]

-- Assigning a boxed alpha to a boxed alpha field within a dynamic array of structures returned from a property or method previously caused a segmentation fault in the traditional compiler. The compiler now reports an "Invalid use of type" error (INVTYP) for properties and methods that return an array of a structure that has handle in it. [tr#36486]

-- A structure with a boxed alpha private field previously caused an unexpected PROTOMISMCH error when the source from which the prototype was derived was compiled. This has been fixed. [tr#36487]

-- Previously, a local data field whose type was a fixed array of structures did not apply the structure’s initial values. This has been fixed. [tr#36489]

Debugger

-- After setting a break point in the calling routine, a GO command will now break at the break point when it is reached. [tr#36481]

Installation

-- Upgrading a machine with running dbssvc services previously caused License Manager service to be disabled. This has been fixed. [tr#36321]

Runtime

-- We made several enhancements to Select and related classes to enable you to create simple joins in a way that is similar to SQL92 join syntax. See "Joining data from multiple sources" in the Discussion for Synergex.SynergyDE.Select.Select in your Synergy Language Reference Manual for an overview, as well as the individual syntax sections for the following new classes and methods:

Synergex.SynergyDE.Select.Sparse
Synergex.SynergyDE.Select.On
Synergex.SynergyDE.Select.JoinSelect
Synergex.SynergyDE.Select.RowEnumerator
Synergex.SynergyDE.Select.Rows classes
Synergex.SynergyDE.Select.Select (constructors and Join method)
Synergex.SynergyDE.Select.From (InnerJoin and LeftJoin methods)

[tr#35559]

-- Join allows the use of one or more "tag" fields on the join of an inner table within the On expression. This tag field is a field of any type or size (compared against a value that doesn’t come from the outer table) that when combined with one or more join fields (compared against a value that comes from the outer table) matches a complete key or contiguous part/segment (starting from the beginning of the key) defined in the inner table. In addition to tag fields, multiple join fields specified in the On expression are now combined (if they match a defined key on the inner table), thus optimizing the effectiveness of the join.

[tr#36468]

-- We improved the performance of concurrent read operations when the same ISAM file is accessed across threads and processes. In testing, we have observed between 10% and 50% improvement with some configurations, with even greater improvements when using SSD drives. [tr#36276]

-- (.NET) We now support .NET native release configurations for UWP. [tr#36259]

-- To support large XML documents, we added %HANDLE_FROM_STRING to enable you to create a memory handle from a String object:

```
   <hnd> = %handle_from_string(<str> [, DM_STATIC])
```

and %STRING_FROM_HANDLE to enable you to create a String object from a memory handle:

```
   <str> = %string_from_handle(<hnd>)
```

[tr#34311]

-- (.NET) We now default to an xfServer connection per AppDomain without requiring S_SERVER_THREAD_INIT. We also added the new S_SERVER_THREAD_SHUTDOWN routine. If a program uses S_SERVER_THREAD_INIT, it should now explicitly call S_SERVER_THREAD_SHUTDOWN() to ensure the xfServer connection is closed. [tr#34798]

-- In some cases, the Select class caused a segmentation fault or returned partial results when a Where expression contained one or more OR expressions. This has been fixed. [tr#36127]

-- We corrected an issue that caused the index for a Rev 6 ISAM file to become corrupt after a DELETE or WRITE. A file would be susceptible to this if it had been configured with TBYTE explicitly or implicitly (e.g., with change tracking) and it had a PAGE setting smaller than 2048. [tr#36130]

-- Parentheses are now used correctly in Select.QueryString values. [tr#36143]

-- Attempting to create an ISAM file from an XDL file (using bldism or fconvert) when specifying a key as an autokey (SEQUENCE, TIMESTAMP, or CREATESTAMP) caused a BADKEY error ("AUTOKEYS cannot be segmented") with bldism or an "Invalid XDL/FDL file" error with fconvert if the last key definition was not the autokey but was segmented. This has been fixed. [tr#36144]

-- (Windows, UNIX) Previously, EOF was not trapped for an ACCEPT statement with an I/O error list. This has been fixed. [tr#36157]

-- (Windows, UNIX) Calling READS after opening a pipe with the O mode for OPEN no longer causes a "signal trap/segmentation violation" error. The FILOPT error is now generated in this situation. [tr#36168]

-- Changing the Select locking behavior by using Q_MANUAL_LOCK in the From constructor no longer causes non-matching records to remain locked. [tr#36197]

-- Previously, a Select with Where.In(key1, val1, val2, ...), where the key and the values were alpha, failed to return matches for values with fewer than the maximum number of digits for the key (e.g., "   123"). This has been fixed. [tr#36227]

-- (Windows, UNIX) MERGE now correctly combines remote files. Previously, the first file specified for a MERGE was used, but other remote files for the MERGE were omitted. (For each of these other files, the MERGE added another copy of the first file to the output.) [tr#36291]

-- If the SQL Connection function %SSC_MOVE was called for a cursor in an error state, a segmentation fault sometimes occurred. This has been fixed. An E_TOOFEWVAR error is now reported. [tr#36346]

-- In versions 10.1.1 through 10.3.1b, the runtime leaked memory if an OPEN statement failed with certain errors but the file existed. This has been fixed. [tr#36424]

-- In rare cases, opening an ELB in a STOP-chained program caused a segmentation fault. This has been fixed. [tr#36438]

-- The use of a structfield in local data (DATA statements following BEGIN) where the structure contains one or more object handles no longer unintentionally overwrites data, which sometimes resulted in a segmentation violation at runtime. [tr#36488]

Synergy DBMS

-- We added a Sparse class to the Synergex.SynergyDE.Select namespace that takes a variable number of field arguments and whose objects can be dynamically combined using the .AND. operator. Also, the Select constructors now have an additional overload that includes the Sparse class object as its first argument. This is functionally equivalent to, and an alternative to, the Select.SparseRecord() method, which remains for backwards compatibility. [tr#35021]

-- The fconvert -v option (when <count> is not specified) no longer defaults to 1 but now automatically throttles the display count by a factor of 10 up to 100,000 records. Due to terminal I/O speed, this will significantly improve performance, especially on large files, when using the -v option. [tr#36214]

-- The fconvert utility no longer ignores NOTBYTE, NOROLLBACK, SGRFA, and ERASE_ON_DELETE file options when specified in a PAR file (using -d PAR file). In addition, specifying an option that takes an argument as the last file option in a PAR file no longer causes a Segmentation Violation. Also, TRACK_CHANGES and SGRFA no longer expect an argument. [tr#36131]

-- (UNIX) The fconvert utility no longer generates an "Invalid par file - Bad file option" error when loading an ISAM file using a valid PAR file copied from Windows with CR/LF line terminators. [tr#36453]

-- Read operations (READ/READS) no longer wait for the completion of isutl -v. [tr#36275]

-- (Windows, Linux) The performance of the chklock utility has been improved when accessing large files. [tr#36341]

-- Fcompare crashed when a structure lacked access keys, and the data file had errors. This has been fixed. [tr#36380]

Synergy HTTP document transport API

-- We added support for IPv6. %HTTP_SERVER_CREATE has a new optional argument, <IP_flag>, that determines which IP versions are allowed. Pass 0 to allow IPv4 only; pass any nonzero value to allow IPv4 or IPv6. If <IP_flag> is not passed, it’s equivalent to passing 0. [tr#35565]

-- (OpenVMS) The minimum supported version of OpenSSL for the HTTP document transport API and DATA_ENCRYPT and DATA_DECRYPT routines is now SSL 1.4. If your OS version is 8.4 (either Alpha or I64), the default version is now the more secure SSL1 1.0-2C. For earlier versions of the OS, which do not support SSL1 1.0-2C, SSL 1.4 will be installed as the default. THIS CHANGE MAY BREAK YOUR CODE. See "OpenSSL Requirements" in the "Requirements and Considerations" chapter of the Installation Configuration Guide and Synergex KnowledgeBase article 100001979. [tr#36282]

Synergy socket API

-- We added new routines to the API to support IPv6. If your application will be connecting to an IPv6 socket, use the new SS2_ routines instead of the corresponding SS_ routines. See the "Synergy Socket API" chapter of the Synergy DBL Language Reference Manual for details. [tr#35565]

Utilities

-- We added two new options, -v4 and -v6, to synxfpng for specifying an IPv4 or IPv6 address. By default, synxfpng looks first for an IPv4-formatted address and then for IPv6. Use the command line options to restrict look-up to only one or the other. [tr#35565]

-- We fixed a problem that was causing synxfpng to crash when the host name was invalid. Now you’ll get a "host name not found" error. [tr#36233]

xfNetLink Synergy

-- We added support for IPv6. You do not need to make any changes to your Synergy client for IPv6 unless you use the %RX_DEBUG_INIT routine (for running in debug mode), in which case you will need to pass the <listen_ip> argument as an a16. [tr#35565]

REVISION 10.3.1b

Compiler

-- When trying to make a structure into a typed box, an INVCAST error was previously reported. Boxing structures as alpha types now works correctly. [tr#34560]

-- The compiler now supports INIT on local data fields. Previously, using INIT on a variable defined with the DATA statement caused the compiler to crash. [tr#35957]

-- In 10.3.1 and 10.3.1a, the compiler reported an unexpected PROTOMISM error on a prototyped class with a private constructor. This has been corrected. [tr#36023]

-- If the accessibility of an overridden property method differs from the accessibility of the base method, the compiler now reports an E_BADACCESS error. Previously, the compiler produced code that resulted in a runtime exception for .NET code. [tr#36058]

-- We fixed an issue that caused the compiler to report an invalid E_IVLNG error for a prototyped decimal or implied-decimal field with a negative initial value. [tr#36068]

Installation

-- Previously the links in relnotes.html did not work in Firefox. This has been corrected. [tr#35935]

Runtime

-- We added support for Windows 10. XCALL ENVRN will return 28 (OS_WINDOWS10) when a program is running on Windows 10. [tr#36089]

-- (Windows, UNIX) We added an optional runtime option, /scl=<level>, to the OPEN statement OPTIONS qualifier to allow an xfServer client to specify the security compliance level, which defines which protocols will be used when encryption is enabled. Valid values for <level> are

0 = use the current Synergy default
1 = use protocols TLS 1.0, TLS 1.1, TLS 1.2 (current default)
2 = use protocols TLS 1.1 and TLS 1.2

SSLv3 and lower protocols are no longer supported.

Additionally, the information returned from GETFA("SLC") now includes the full cipher description: <ciphername> <protocol version> Kx=<key exchange> Au=<authentication> Enc=<symmetric encryption method> Mac=<message authentication code>.

(Note: Confusingly, in this output that comes from OpenSSL, TLS 1.0 ciphers are marked as being SSLv3 protocol, but they are really TLS 1.0.) [tr#35922]

-- (Windows, UNIX) We added an optional timeout parameter to the %SS_CONNECT socket routine that specifies the number of seconds to wait for a connection before the routine returns with an SS_ETIMEDOUT error. If the timeout parameter is not specified, the routine will wait indefinitely for either a successful connection or another error to occur. [tr#35931]

-- (Windows, UNIX) When running in debug mode or when BOUNDS_CHECK_FATAL is set, we now check for global symbol references (either common or global data section) that range past the end of the containing psect. If found, a GBLSIZ fatal error occurs. [tr#35992]

-- (Windows, UNIX) We fixed an issue that could prevent a Select from returning all matching records when there was a type mismatch between a numeric field type specified in the Where condition and the defined key type in the file. As a result of this fix, field/key type mismatches can no longer be optimized unless the runtime can determine from the search criteria that no negative values will be returned. This may result in a noticeable decrease in performance unless the type mismatch is corrected or the Where expression is changed. For example, if key1 is a decimal field and it matches an alpha key, the Where expression (key1 < val) will not be optimized. However, if the Where expression is changed to (key1 > 0 .and. Key1 < val), it will be optimized if val is greater than 0. Correctly typing key1 as decimal in the file will also result in optimization. [tr#35651]

-- Stack memory attached to RCB arguments or an SQL bind now gets an RCBSTACK or SQLSTACK error on a THROW/CATCH when compiled with -qcheck. Also, if a routine is built with -qrefresh, an RCBREL or SQLREL error will be signaled. [tr#35966]

-- (Windows) We improved the way the winerrpost.log file is generated and updated. Previously, if an error that was logged to the file was encountered in a loop, the file could grow to an enormous size if the error didn’t terminate the process. Furthermore, the information logged in the file wasn’t sufficient to help diagnose errors. These issues have been corrected. [tr#35882]

-- (Windows, UNIX) In versions 9.5.1 through 10.3.1a, a READS on a channel opened in input mode performed poorly if updates or deletes simultaneously occurred on another channel or by another process. This has been corrected. [tr#35936]

-- We made several improvements to the CurrentRFA method for the Select.AlphaEnumerator and Select.RestrictedAlphaEnumerator classes. This method now returns the correct RFA from a remote server with SCSPREFETCH, and it now returns the correct GRFA when a From class record is smaller than the actual ISAM file record. Additionally, if the "Stored GRFA" attribute of an ISAM file is configured, CurrentRFA now uses this attribute, which reduces CRC calculation overhead for each enumeration. [tr#35915]

-- (Windows, UNIX) The informational error "%DBR-I-OPWFND, Operation was FIND" was reported multiple times for an untrapped I/O error traceback on a Select channel. This has been corrected so that this informational error is no longer reported in this situation. [tr#35895]

-- (Windows, UNIX) In versions 10.1.1 to 10.3.1a, a Select with a Where expression that specified multiple adjacent segments failed to return all records that matched the criteria if it used EQ or EQS for the first segment and GE, GES, LE, or LES for the second and third segments. This has been corrected. [tr#35874]

-- Select now optimizes Where expressions regardless of the way comparisons are ordered. Previously, (Where)(0 < key) would not be optimized, even though (Where)(key > 0) was optimized. Additionally, the order of GE, GES, LE, and LES comparisons would sometimes prevent optimization for segmented keys when multiple adjacent segments were compared. These issues have been corrected. [tr#35871, tr#35623]

-- An "Invalid operation" error (INVOPER) is now reported in every case where a READS, FIND(MATCH:Q_SEQ), or WRITES follows a Select, unless context has been reestablished with a READ, FIND, or WRITE statement. (Until READ, FIND, or WRITE is called in this situation, sequential context is undefined.) Previously, there was one case where this error was not reported: after using Dispose with a Select in a FOREACH statement. This has been corrected, but note that THIS CORRECTION MAY BREAK YOUR CODE. Previously, the runtime would allow your program to call READS, FIND(MATCH:Q_SEQ), or WRITES without reestablishing context after a FOREACH Select was Disposed (even though the results were undefined). Now it will report the error. [tr#35907]

-- (Windows, UNIX) Long exception messages caught by TRY-CATCH are now displayed in full, rather than truncated with the message "*** Warning: Message replacement too long ***." [tr#36045]

-- An exception explicitly thrown or rethrown that is unhandled or covered by an $ERR_CATCH when there’s a TRY with a FINALLY somewhere within scope of the calling chain no longer generates a Signal Trap. [tr#36044]

-- (Linux) In 10.3.1 and 10.3.1a, secure connections failed with "OPENSSL1_0_0 not found" on Redhat and Fedora Linux. This has been corrected. [tr#36066]

Synergy Configuration Program (SynConfig)

-- SynConfig now supports setting the Security Compliance Level. [tr#35953]

Synergy DBMS

-- The isload CLEAR option now supports records larger than 20K. [tr#35970]

-- All output from fconvert (informational and error messages) now goes to STDOUT instead of STDERR. This allows direct use of the fconvert command through an OPEN pipe. [tr#36060]

-- We improved the performance of isutl -f (fastload) and fconvert -ai (append) when loading an existing ISAM file that is empty. [tr#35877]

-- Various issues with ISAM files configured with multiple fixed-length records (i.e., record type "multiple") have been resolved. Isutl -rs (add static RFA) no longer fails, and isutl -rc (add compression) is no longer allowed, as that configuration is not permitted. Also, change tracking is not allowed with multiple fixed-length files and now generates an error if attempted. [tr#36015]

Synergy HTTP document transport API

-- The default protocol is now set to TLS1.0/TLS1.1/TLS1.2 for OpenSSL 1.0.1 and above, and TLS1.0 for OpenSSL versions below 1.0.1. [tr#35823]

-- We added a new encrypting rule set, HTTPDATA_RULES, to %SYN_ESCAPE_HANDLE and %SYN_UNESCAPE_HANDLE that encrypts data according to HTTP RFC 3986. This enables you to encrypt data parts of the URI without affecting unreserved characters in the URI. [tr#35990]

-- Previously, once HTTP_NOESCAPE was set it couldn’t be unset, which prevented automatic escaping from being turned back on. This has been corrected. [tr#35951]

-- In version 10.1.1 on Windows or 10.3.1 and 10.3.1a on UNIX and OpenVMS, running against OpenSSL version 1.0.1n or greater (or a functionally equivalent patch for operating systems that provides OpenSSL) caused the HTTP document transport API to give a "dh key too small" error when using default Diffie-Hellman keys. This has been corrected.

Synergy .NET assembly API

-- Large strings were previously truncated at 65536 bytes on 32-bit systems when returned from the .NET assembly API. This has been corrected. [tr#36052]

Utilities

-- An open error previously occurred with listelb and listdbr when the ELB or DBR file had linked ELBs and the translated environment variables exceeded 32 characters. This has been corrected. [tr#35968]

xfNetLink Synergy

-- We made a number of enhancements to encryption. We added an optional parameter, SCL, to %RX_START_REMOTE and %RX_DEBUG_START to allow you to override the default security compliance level, which determines which protocols will be used for encrypted communication. Valid values for the SCL parameter are

0 = use the current Synergy default
1 = use protocols TLS 1.0, TLS 1.1, TLS 1.2 (current default)
2 = use protocols TLS 1.1 and TLS 1.2

Note that setting 0 means that the supported protocols could change when you upgrade to a new version of Synergy.

SSLv3 is no longer supported.

The security compliance level is also set on the xfServerPlus machine. The setting there should match the one on the xfNetLink machine.

In addition, when there is a protocol handshaking error and secure communication cannot be established between xfNetLink and xfServerPlus, the server closes the socket, logs an error, and RX_SSLCONNECTERR (12323) is returned on the client. Previously, this caused the client to hang. [tr#35923]

REVISION 10.3.1a

Compiler

-- A new global define, _DEBUG, is now supported for compiling with the debug option. This enables you to conditionally include or exclude code based on whether or not the program is compiled in debug mode. [tr#35724]

-- Compiling two sources with separate command lines that both used a built-in enumeration (such as Select’s CTState) caused a SegVio. This has been fixed. [tr#35688]

-- Previously, an INVOPT error occurred when dbl was used in a .proj file with msbuild. This has been fixed. [tr#35421]

-- We made some minor corrections to the compiler when using -qnet to track changes made to the .NET compiler. [tr#35703]

-- A globally declared enumeration followed by an implicit main did not correctly recognize the implicit main. This has been fixed. [tr#35756]

-- A field initializer incorrectly reported an INVINITMBR error if the assignment statement that contained it was the first line in the routine. This has been fixed. [tr#35807]

Installation

-- The Synergy/DE Windows installations now include VC++ merge modules. These merge modules were not included in Synergy/DE 10.3.1 distributions, which prevented Synergy projects from opening in Visual Studio on 64-bit systems that had only Visual Studio 2012 installed. NOTE: To use Synergy DBL Integration on a 64-bit system that has Synergy/DE 10.3.1 and Visual Studio 2012 only, you must either upgrade 64-bit Synergy/DE to 10.3.1a or download vcredist_x64.exe from http://www.microsoft.com/en-us/download/details.aspx?id=40784 and run it on the 64-bit system. [tr#35780]

Linker and librarian

-- Previously, an INVOPT error occurred when dblink and dblibr were used in a .proj file with msbuild. This has been fixed. [tr#35421]

Runtime

-- We added an %ATRIMTOSTRING() function to reduce overhead when creating a string from trimmed alpha variables. [tr#35609]

-- (.NET) We added support for thread-specific xfServer connections for Mono platforms. [tr#35591]

-- When buffering records stored to an ISAM file (using the BUFSTORE OPEN option) and also doing a sequential READS on the same channel, a certain condition sometimes caused a record to be returned out of sequence. Specifying LOCK:Q_NO_TLOCK or LOCK:Q_NO_LOCK increased the likelihood that this would occur. This has been fixed. Also, SCSPREFETCH (input caching) is no longer enabled implicitly on a channel opened with /BUFSTORE. [tr#35568]

-- When a program used IO_HOOKS and was compiled prior to 10.3.1 but run with 10.3.1 or later, a memory leak occurred. This has been fixed. [tr#35579]

-- In 10.3.1, a Select didn’t return any records when the Where expression contained an OR operator with terms that referenced a field contained within a defined key, but the offset of the field didn’t start at the beginning of the key. This has been fixed. [tr#35595]

-- In versions 10.1.1b through 10.3.1, a RTNNF error ("Cannot access external routine") occurred (with no filename) when loading a method when the MAXMEM value was exceeded and the routine being loaded referenced a crushed method name. This has been fixed. [tr#35641]

-- Converting from descriptor types to objects is now prohibited in conditional operators in traditional Synergy, as this could cause an invalid class handle error at runtime. A PLATSUPT compiler error is now reported in this situation. [tr#35679]

-- When using output I/O buffering (/BUFSTORE or OUTPUT sequential WRITES) with xfServer Connection Recovery, and the STORE or WRITES operation was retried with a successful reconnection to the original context using either reconnect_hook() or a retry of the STORE/WRITES on an $ERR_SRVCONRTY error, some records that should have been written to the file were not, due to the cache buffer not being flushed to disk. This has been fixed. [tr#35644]

-- A Select on a non-ISAM file using Orderby.Ascending(fld) or Orderby.Descending(fld) previously skipped the first record when the first record in the ordered selection was also the first record in the file. This has been fixed. [tr#35738]

-- On a big-endian machine, when passing an argument by value to a routine where the ^VAL argument has a specified size of less than the size of an address on that platform, the correct value (instead of 0) is now accessed. [tr#35801]

-- (.NET) We fixed a small memory leak caused by the Select class. [tr#35764]

-- (.NET) We fixed an issue that caused a System.OverflowException ("Arithmetic operand exceeds maximum size") to occur when storing an i8 value greater than 2147483647 on a 32-bit platform. [tr#35767]

-- (.NET) Converting a decimal 18446744073709551615 (2^64-1) to an unsigned long no longer causes a System.OverflowException. [tr#35772]

-- (.NET) We improved the performance of DM_ALLOC for both %MEM_PROC and the use of temporary descriptor types by eliminating the initial value (spaces) for these operations. Consequently, DM_ALLOC is now explicitly uninitialized, as documented. (In previous versions it defaulted to DM_BLANK.) [tr#35782]

-- (.NET) We fixed several issues that had the potential to cause threading problems for Select.OrderBy, record locking for the WAIT I/O statement qualifier, and concurrent opens using the secure sequential or caching open options. [tr#35652]

-- (.NET) Previously, the Select class returned the incorrect number of records in certain cases because the system garbage collector prematurely released Where condition data. This has been fixed. [tr#35695]

-- (.NET) Previously, if an application used multi-threaded access to xfServer with connection recovery but without S_SERVER_THREAD_INIT, an incorrect error code was sometimes returned or the application could lock up. This has been fixed. [tr#35636]

-- (.NET) We fixed a memory leak that occurred if you created an AlphaEnumerator, used From (for the Select class) on the same channel, and then closed the channel. [tr#35663]

-- (.NET) Operations that assign unsigned integers (ushort or uint) to decimals now result in positive numbers. [tr#35658]

-- (.NET) If an ISAM file reaches a limit set with SIZE_LIMIT or RECORD_LIMIT, FileFullException is now thrown. (Previously, SynException was thrown.) [tr#35637]

-- (.NET) Previously, the negative sign (-) was not shown for negative values if you used ToString or an implicit string cast from an implied decimal with 0 decimal places. This has been fixed. [tr#35606]

-- (.NET) Previously, an unhandled exception would in rare cases cause the runtime to hang on shutdown. For example, this could happen if you pressed CTRL+C while xfServer was reconnecting. This has been fixed. [tr#35583]

-- (.NET) Previously, a thread shutdown would sometimes cause a .NET Framework segmentation fault after an I/O operation where the channel that was last used had been closed on another thread. This has been fixed. [tr#35575]

-- (.NET) Characters in the unsigned byte range (128-255) now work on Mono platforms when converting to/from a string. [tr#35570]

-- (.NET) XCALL SETDFN did not support PATHs/logicals or work with XCALL ISAMC. This has been fixed. [tr#35664]

-- (.NET) If you used the Select class for a segmented key where the order of the segments differed, some matching records would not be returned. This has been fixed. [tr#35612]

-- (.NET) Previously, no records would be returned if you used the Select class with an order that was different than the specified key and the comparison spanned more than one segment. This has been fixed. [tr#35573]

-- (.NET) We removed support for MonoTouch, which is no longer supported by Apple. [tr#35667]

-- (.NET) On Android and 32-bit iOS, using %DATE with an i8 variable caused an arithmetic overflow exception. This has been fixed. [tr#35654]

Synergy DBMS

-- We added a new file creation option to the ISAMC routine, bldism, and XDL keyword files. If ERASE_ON_DELETE is specified at file creation, when a record is deleted (or moved internally by ISAM), it will be erased (i.e., nulled out). Without this flag enabled, records that are deleted are only marked deleted; their contents remain in the file until the space is reused, which has been the behavior until now. ERASE_ON_DELETE cannot be used with change tracking. [tr#35481]

-- We removed an unsupported system call to allow the Synergy runtime to run with limited support on Red Hat Enterprise Linux 5. [tr#35712]

-- When isutl -f (fastload) is used without the -k option (counted input file), the operation is aborted if the ISAM file is configured with one or more integer or auto keys, and a message is displayed to indicate that a counted file must be used. [tr#35616]

-- When running isutl -f (fastload) to a file containing autokeys, the autokeys values were not being preserved (regardless of whether or not the -ak option was specified). This has been fixed. [tr#35616]

-- By default (if the -r "input record size" option is NOT specified), when using fconvert with a relative file for input, the OPEN scans the first record to determine the record size (see OPEN option RECSIZ=-1). If the first record contains binary data, it’s possible the open may encounter a premature line terminator or EOF condition. The -r "input record size" specification now correctly overrides the OPEN RECSIZ scan and ignores binary data in the first record. [tr#35805]

Synergy HTTP document transport API

-- OpenSSL version 1.0.1 added support for protocols TLS 1.1 and TLS 1.2, and by default, the HTTP document transport API supports them as well. We’ve also added support to turn off these two new protocols, if desired, by not specifying them as desired protocols in the API. In addition, we added two new defines, SSLVER_TLS1_1 and SSLVER_TLS1_2, to SSLVER_ALL. [tr#35794]

-- The %HTTP_GET, %HTTP_PUT, %HTTP_POST, and %HTTP_DELETE routines previously did not return headers when an error occurred. This has been fixed. [tr#35656]

Utilities

-- Previously, an INVOPT error occurred when dblproto and dbl2xml were used in a .proj file with msbuild. This has been fixed. [tr#35421]

REVISION 10.3.1

Compiler

-- We added support for simple properties. [tr#34689]

-- We added support for STATIC classes. [tr#34691]

-- We added support for READONLY STATIC class fields. [tr#34690]

-- The following operators are now supported for enumerations: bitwise OR (.BXOR.), bitwise NOT AND (.BNAND.), and bitwise NOT (.BNOT.). [tr#34441]

-- We reduced prototype dependencies of private members. Prototypes now don’t include private methods, private properties, and private classes. Due to runtime restrictions, private data members must be loaded (records, groups, fields, structs); however, if a field is marked as private and its type is a handle or array of handles, it is represented as object or array of object in the prototype (to reduce dependencies). The version number and format of the prototype files require that prototypes be rebuilt or a PRTLOAD error will occur. [tr#34308]

-- If the maximum number of files is exceeded when using a command file to redirect input, a MAXIF error is generated. [tr#34444]

-- If a local data field or catch variable hides another local field or catch variable, a level 3 W_HIDEW2 warning is now reported. A level 4 warning was reported in previous versions. [tr#35012]

-- Using an implied-decimal initial value of length 33 or greater could previously crash the compiler. This has been fixed. [tr#35485]

-- Initial values on class fields of type float, decimal, and implied decimal coming from a prototype file were not loaded correctly, causing PROTOMISMCH errors. This has been fixed. [tr#35282]

-- (UNIX) You can now unbox an object variable to a Synergy fixed array ([*]a). Previously, the compiler reported an invalid INVEXPR error. [tr#34627]

-- (UNIX) An "invalid INVNUMDIM" error was reported if ^M was used with an array that had no element number and that was part of a += statement. This has been fixed. [tr#35064]

-- (UNIX) The compiler no longer crashes if NEW is used with a type whose name differs only in case from the name of a local parameter of that type. [tr#35145]

-- (UNIX) In version 10.1.1c, the compiler did not report an INITVERR error as expected when it encountered a mismatched data type for an initial DATA statement value. This has been fixed. [tr#35184]

-- (UNIX) Previously, the compiler did not ensure that required parameters were passed when a system method was called without qualifiers. This has been fixed. [tr#35185]

-- (UNIX) The length method of a property that returned an array of strings gave an incorrect runtime result. This has been fixed. [tr#35229]

Installation

-- Framework CLR 4.0.30319 or higher is required in order to install Synergy. [tr#34965]

-- Synergy installations require at least Service Pack 1 on Windows 7 and Windows Server 2008 R2. [tr#34849]

-- We added experimental support for Mono on 64-bit Linux systems. If a system has Mono, you should add synrnt.dll (included with your distribution) to the global assembly cache. [tr#35292]

-- Dbs services created by dbssvc.exe are now affected by the installation package. On uninstall, all dbssvc services will be removed. On upgrade, all dbssvc services are shut down and restarted after the upgrade is complete. [tr#25302]

-- (UNIX) During the execution of install.sde, choosing to skip a chosen but non-existent installation package such as lm.gz by typing "s" at the prompt has been fixed to properly skip the package and complete the installation of the other selected packages. [tr#34502]

-- Previously, the installation could fail if the license server was selected but no licensee name was provided. This has been fixed. [tr#35106]

Runtime

-- (Windows, UNIX) We added a new ChangeTracking class to retrieve the current change tracking information from an SDMS ISAM file. The class also contains the ChangeTrackingEnabled property, which returns whether or not change tracking is enabled on the file, and the ApplySnapshot method, which applies another snapshot to the file. [tr#34234, tr#34553]

-- We added the String.Replace(str,str) method. [tr#34320]

-- (.NET) We added initial iOS runtime support. [tr#34709]

-- (.NET) The debugger’s SHOW CHANNELS command now shows the open mode. [tr#34636]

-- (Windows, UNIX) We added ISCONFIG, a subroutine that enables you to modify the user-defined text in an ISAM file’s data header. This routine has the following syntax where <channel> is the channel on which the file is open, "TEXT" indicates that the change is for the file’s user-defined text (this is the only option currently available for this argument), and <text_string> is the new text for the file:

```
   ISCONFIG(<channel>, "TEXT", <text_string>)
```

Note that if <text_string> is longer than the space allocated for user-defined data header text, the string will be truncated. [tr#33686]

-- You can now use AS in traditional Synergy to indicate the type of members in a collection. Note that this feature does not yet result in type conversions, so essentially it is useful only for unboxing. [tr#35020]

-- The Where.In selections for the Select statement are now optimized when an ISAM key field is specified by translating to a set of OR conditions. Since IN(F,A,B) results in the equivalent (Where)(F.eq.A.or.F.eq.B), doing both head and tail optimization of each individual part (which would normally not occur because of the .OR.) has greatly improved performance. Also, if the expression contains an additional AND condition (such as IN(F1,A,B).and.(Where)F2.eq.C), each comparison of A and B includes the AND part of the expression, making separate passes for each IN parameter specified. [tr#34453]

-- (.NET) We now support %DLL_CALL on .NET. [tr#35011]

-- (.NET) We changed the default font for W_ in .NET to Consolas to scale better, especially with SYN_RESIZE_SCALE. If a user runs on an unsupported Windows XP or Server 2003 machine, the Consolas font will need to be installed from Microsoft. [tr#34510]

-- We now provide a notification when an invalid service name is specified on the dbssvc unregister command line. [tr#35467]

-- (.NET) Using an environment variable with a trailing backslash (\) and XSUBR in .NET incorrectly generated an "index out of range" exception. This has been fixed. [tr#35429]

-- (.NET) The concurrent use of XCALL GETLOG/SETLOG on multiple threads previously corrupted the environment or caused a segmentation fault. This has been fixed. [tr#35456]

-- (.NET) If %MEM_PROC(DM_RESIZ+DM_STATIC) was used on a local handle, an INVHLD error occurred (which did not happen in traditional Synergy) and memory was leaked. Although this is invalid code, the compatibility should have been the same. This has been fixed. [tr#35457]

-- (.NET) Sometimes a container was too small when SYN_RESIZE_SCALE was set. This has been fixed. [tr#35461]

-- (.NET) Using a System.String type function on a STOP statement is now allowed. A STOP statement cannot be used with the -portable compiler option; an E_NOTSUPT error is reported in this situation. [tr#35492]

-- (.NET) Pressing CTRL+P in Print Preview mode caused a segmentation fault if main() was not marked with the attribute {STATHread}. This has been fixed to report an E_ARGMIS error if the attribute is missing and CTRL+P is pressed. [tr#35500]

-- String.EndsWith and String.StartsWith reported false positives when compared with a longer string. This has been fixed. [tr#35437]

-- Creating an ISAM file with the maximum record size with compression no longer generates an ISAM file where any record stored generates either "Internal ISAM error - Data segment size mismatch with leaf" (in 10.1) or a segmentation fault (prior to 10.1). [tr#35459]

-- (.NET) In rare cases issuing a STOP statement in a WinForms or low-level Windows application caused a segmentation fault. This has been fixed. You must recompile your code with 10.3.1, and the "Target Synergy runtime" option (on the Build page of Visual Studio Project Designer) must also be set to "10.3.1." If it’s set to an earlier version, you won’t get this fix. [tr#35491]

-- (UNIX) The synsslx check for libcrypto.so was wrong for modern 64-bit Linux like Unbuntu and caused SSL failures. This has been fixed. [tr#35168]

-- (UNIX) Using rsynd on 32-bit UNIX with LDAP is unsupported but no longer causes a core dump. Using XCALL GTPPN on 32-bit UNIX with LDAP returns zero for the group and member instead of doing a core dump. [tr#35174]

-- (UNIX) If %DLL_OPEN could not open a file via its full path or find the target file in LD_LIBRARY_PATH, the code appended DBLDIR to the beginning of the path and looked there. Then the runtime reported this updated path in the error message rather than the original path. Additionally, there was no way to get the error text from dlerror() to find out why it failed. These have been fixed, and the <reserved2> argument from XCALL ERROR shows the resulting system error text. [tr#35288]

-- (Windows, UNIX) When storing records to a non-TBYTE ISAM file where the index file (.ism) reached the maximum 2 GB before the data file (.is1) and an $ERR_FILFUL error was generated, it was observed that after running isutl -v on the file, an Index error was generated, indicating an error in the index freelist and unreferenced index blocks. This has been fixed. [tr#35359]

-- (.NET) S_SERVER_THREAD_INIT no longer generates an error in the primary AppDomain. [tr#35396]

-- (.NET) XCALL SETLOG no longer causes unhandled exceptions if two threads use either XCALL SETLOG concurrently or XCALL GETLOG at the same time as XCALL SETLOG. [tr#35396]

-- A signal trap occurred on OPEN when using a search path that contained a remote filename specification. This has been fixed. [tr#35148]

-- (Windows, Unix) Previously, creating an ISAM file from an XDL file would result in a "file not found" error if the OPEN specified "" for the file. This has been fixed. [tr#35146]

-- We resolved an ISAM data corruption issue caused during file update with a non-TBYTE file that resulted in a file exhibiting a data file at or exceeding 4 GB (>=4294967296) in size. [tr#35179]

-- The GETFA subroutine with one of the keywords ALQ, FFB, or HBK now returns the correct size value when the specified file exceeds 2 GB in size. As before, if the size of the field passed to return the size is too small, the value will be truncated to the size of the field. However, if the return field is an integer and it is too small, we now return the largest positive value that will fit in the return field instead of a negative value. So, if the size of the return field is an i2, any file that is greater than 32767 bytes will return the value of 32767 (which is the largest positive value allowed by the i2). [tr#34951]

-- (Windows) Doing a STOP chain to a command file in the form STOP ’@cmdfile’ no longer causes the RUN license and any other license currently logged in by that application to remain logged in. [tr#35131]

-- (.NET) In addition to setting the environment, the logicals RUSER SCSCOMPR and ISAMC_REV are supposed to update the current execution environment. On .NET this did not occur, and only the initial environment was updated. This has been fixed. [tr#34752]

-- (.NET) When CTRL+C was disabled with XCALL FLAGS, ALT+F4 still shut down the app when using W_ routines. This has been fixed. [tr#35071]

-- (.NET) The correct versions of the OpenSSL libraries were not always found, especially when using ClickOnce. This has been fixed. This change affects the HTTP document transport API, xfNetLink Synergy, xfServer client with encryption, and the encryption routines (DATA_ENCRYPT and DATA_DECRYPT). For ClickOnce deployments using these capabilities, OpenSSL must already be installed in DBLDIR on the target machine; you cannot install the OpenSSL libraries via ClickOnce. In addition, other non-.NET windows products no longer look in path and now expect the DLLs to be copied to dbl\bin. [tr#35129]

-- (Windows, UNIX) The parsing of URIs for HTTP functions incorrectly stripped the query from the URI when the URI did not contain a path. This has been fixed. (Note: This was released on OpenVMS in 10.1.1d.) [tr#35327]

-- (UNIX) In versions 9.3.1 through 10.1.1c, the %ABS function returned 0 for user-defined functions. This has been fixed. [tr#34996]

-- (UNIX) Previously, a call to SYN_UNESCAPE_HANDLE in rare cases corrupted the output string and resulted in duplicate characters. This has been fixed. [tr#35176]

-- (UNIX) With versions 9.3.1 through 10.1.1c, variables were not always cleared as expected when running a pre-9.3 .dbr that used INIT. This has been fixed. [tr#35199]

-- (UNIX) %DATETIME_TO_I8 now appropriately adjusts to daylight savings time, and when used in a keyed READ with a timestamp key, it reads the expected record. [tr#35261]

-- (UNIX) If the message argument to the SEND statement has a length of 0, an ARGSIZ error is now signaled. [tr#35289]

-- (.NET) The Select.CTinfo.SnapshotTime property now returns the correct snapshot DateTime. [tr#35533]

Synergy Configuration Program (SynConfig)

-- The Synergy Configuration Program now supports enabling the xfServer Connection Recovery feature. [tr#35404]

-- Adding or modifying an xfServer or xfServerPlus service with a Service name or Display name that is not unique across all services now generates an appropriate message box instead of an empty one. [tr#35026]

Synergy DBMS

-- We added a new file creation option, NOROLLBACK (",[NO]ROLLBACK" or XDL FILE keyword ROLLBACK yes|no), which prohibits the rollback function when change tracking is enabled. Also, the ctutl utility has a new change tracking command option, -n, that prohibits rollbacks on a file. It can be used with or without the -a command. [tr#34450]

-- We added a new file creation option, SIZE_LIMIT (",SIZE_LIMIT=<value>" or XDL FILE keyword "SIZE_LIMIT <value>", where <value> is the maximum number of megabytes that the data .is1 file is allowed to grow). If a STORE or WRITE operation will cause the file to exceed this size, a FILFUL error will be generated and the record will not be stored. [tr#34132]

-- We added a new file creation option, RECORD_LIMIT (",RECORD_LIMIT=value" or XDL FILE keyword "RECORD_LIMIT value", where value is the maximum number of active records allowed in a file). If a STORE or WRITE operation will exceed this number, a RECLIMIT error will be generated and the record will not be stored. [tr#34474]

-- Bldism now recognizes the NETWORK_ENCRYPT option and can create files with network encryption. [tr#35430]

-- We enhanced isutl to better detect and correct the kind of ISAM data corruption with non-TBYTE files (shown in tr#35179) where the data file appears to be at or exceeding 4 GB (greater than 4294967296). However, due to the severity of the corruption, the file header may become damaged, in which case isutl will recommend using fconvert to extract the data to a new file. [tr#35187]

-- ISAM records that exceed the old 64K limit are now allowed when a file is created with ",variable" and 0 for the maximum record size. A file of this type can be created or accessed in 64-bit traditional Synergy or 32- or 64-bit Synergy .NET. [tr#33307]

-- We added error checking when writing the "extend" area of the primary file (for anything not merged at the end of the file) and also when writing the OrderBy intermediate (temporary) file with information for the Select when OrderBy is specified. [tr#35008]

-- Fconvert no longer creates an ISAM file when an exception file exists. [tr#21439]

-- Fconvert now records the number of exception records written when optimized. [tr#34842]

-- Running isutl -r when an old .257 file existed caused unexplained "Sort work file in use" errors. A .257 file results if a failure occurs during the data sort phase of isutl (typically in FILFUL situations). The error situation was possible if the .257 file was not removed by either re-running isutl or physically removing it from the file system. We’ve resolved the problem by automatically removing old .257 files. [tr#34376]

-- (UNIX) An "Internal ISAM error - File may be corrupt" error with the informational text "Unable to locate bucket" no longer occurs on a DELETE operation in very large variable-length records or compressed record files (greater than 4 GB) [tr#35123]

-- (UNIX) With previous version 10 releases, some ISAM files created with version 9.5.3b or earlier and configured with TBYTE could falsely exhibit file corruption. Running isutl -v on such a file reported "Bad reset PAGE offsets." This has been fixed, and the -v option for isutl has been updated to report this situation and suggest using isutl -r to correct it. The fastest way to correct this condition is to patch your file to REV 6 using isutl -p. [tr#35181]

-- (UNIX) Isutl -p previously caused a memory fault error if it patched a pre-REV 6 file that was compressed but contained data that didn’t compress well (i.e., records without much white space or many zeros). This has been fixed. [tr#35212]

-- (UNIX) Running isutl -r or -v no longer fails with "I/O error: No disk space" (when plenty of disk space is available) on a non-TBYTE file when the index file is greater than 2,147,483,647. (A workaround to this is to delete the index file and re-index. If you can add TBYTE, do so now. Otherwise, a higher packing density may suffice.) [tr#35361]

-- (Windows, UNIX) The isutl utility no longer allows an index file to exceed 2,147,483,647 bytes when re-indexing a file that is not configured with TBYTE. It now fails with the error "Index or data exceeds maximum size" and a message indicating that only the data file (.is1) remains, and suggests re-indexing with the addition of TBYTE (-qfile=tbyte). [tr#35361]

-- (Windows, UNIX) The error $ERR_SIZLIMIT (rather than$ERR_FILTUL) is now generated when either part of an ISAM file (.ism or .is1) will exceed 2 GB when updating a file that was not configured with the TBYTE file option. [tr#35361]

-- The index and data headers of an ISAM file pair are now compared on OPEN and will generate an $ERR_NOTISM error ("Not an ISAM file") if they are different. [tr#35360]

-- In version 10.1.1, -ln and -le needed to be at the end of the isutl command string. This has been fixed. [tr#35035]

Synergy HTTP document transport API

-- We increased the maximum HTTP header size to 2048 to better accommodate QAUTH packets. [tr#35080]

-- We updated routines in the Synergy HTTP document transport API so that a variable used for an error argument will now be cleared if the routine completes without error. Variables used for <out_headers> arguments are now also cleared. [tr#34641]

-- The HTTP_ routines caused an "Unhandled exception" error in the .NET runtime when the log file parameter was a zero-length string. This has been fixed to ignore the string. [tr#35117]

-- We changed the stack allocations to use ALLOCA instead of stack allocations for header buffers. [tr#35126]

Synergy .NET API

-- The minimum Framework supported is now CLR 4.0.30319. We no longer distribute gennet.exe, which supports 2.0. [tr#34961, tr#34966]

-- Property initializers used within lambdas reported INVINITMBR errors. This has been fixed. [tr#35257]

Synergy routine call block API

-- (UNIX) A recursive error condition was fixed when doing argument scope checking using the routine call block API and programs built with -qcheck. [tr#34974]

-- (UNIX) When using RCB_SETARG, RCB_SETARGS, or RCB_INSARG with a temporary literal value, the value specified will no longer change after being set. In addition, using literals to RCB_SETARG will no longer cause an F_RCBREL error on segment reclamation of the routine where the literal was set if the RCB handle is still active. [tr#34978]

Utilities

-- (UNIX) Constructor arguments no longer cause an error in dbl2xml. [tr#34968]

-- (UNIX) Functions without an explicitly declared return value where the first FRETURN was another unresolved function call caused dblproto to crash. This has been fixed. [tr#35252]

-- (UNIX) Using dblproto on a parameter whose type was a handle to an unresolved type marked the parameter as optional. When compiled and the type was resolved to class, the parameter was marked as required. Because the prototype didn’t match the declaration, an unexpected PROTOMISMC error was reported. Dblproto now correctly marks the parameter as required when the type is a handle. [tr#35259]

xfNetLink Synergy

-- We added support for the following xfNetLink Synergy routines to the .NET runtime: %RX_START_REMOTE, RX_SHUTDOWN_REMOTE, %RXSUBR, %RX_RMT_TIMOUT, %RX_RMT_SYSINFO, %RX_CONTINUE, RX_GET_ERRINFO, RX_GET_HALTINFO. For more information, see the xfNetLink & xfServerPlus User’s Guide. [tr#34825]

DOCUMENTATION CORRECTIONS AND ADDITIONS

Synergy Tools

-- In the "Building and Running Synergy Applications" chapter, the Runtime compatibility option (-qrntcompat=<value>) has two new values:

10030300	Version 10.3.3
10030301	Version 10.3.3a

RESTRICTIONS

The following restrictions have been identified in version 10.3. See the RESTRICT.TXT file for restrictions identified in earlier versions.

Runtime

-- (.NET) The .NET DrawString function doesn’t support the ATTR_UNDR Windowing attribute underlining an entire untrimmed buffer. [tr#35287]

-- (.NET) On Windows Phone and Windows Store, the ERRMOD routine always returns a line value of 0. [tr#35599]

Synergy Windows printing API

-- Lines drawn by the Synergy Windows printing API’s DWP_LINE function are reduced in thickness after the first page when the pen is not deleted and recreated for each page. [tr#33894]

OPEN INCIDENTS

Compiler

-- Linking in a partial class with a private method causes an unexpected CLSCRC error and a potential dblproto error. The workaround is to combine partial classes in the same source. [tr#36141]

-- For a program that has a non-literal local DATA statement with an initial value followed by a literal local DATA statement with an initial value, the debugger will skip the first local DATA statement and step twice on the second local DATA statement. [tr#36279]

-- Casting ^M of a structure on a handle to @a gives an E_INVCAST error. The workaround is to add an intermediate (a) cast. [tr#34560]

-- (Windows, UNIX) An expected NFND error does not occur when a local variable matches the leading edge of the search path. For example, a method call of variable e of type joe that does not have the method defined matches an e in a higher scope of a different class that does have the method defined. [tr#35469]

-- When building a project, a CLSCRC error can occur when linking an ELB if the base classes aren’t first in the prototyping order. [tr#35752]

Debugger

-- (OpenVMS) A second connection to remote debug port is not producing an error. You cannot have two sessions on the same port. [tr#31408]

-- The debugger does not support multi-level casting of objects. [tr#32571]

Installation

-- (Windows 10) On a 64-bit SDE Client installation, Synergy Control Panel is missing its links. [tr#36436]

-- 64-bit SDE does not properly detect existing licensing if the license server is specified using an IP address. [tr#36413]

-- 64-bit SDE Client is not installing environment variables on Windows 10 correctly unless you log out and restart after installation. [tr#36469]

Runtime

-- (Windows, UNIX) When a stop is called either explicitly or implicitly, open channels in output or append mode are not purged as expected. [tr#36118]

-- Issuing the Count property on a Select that contains an OrderBy.Ascending or OrderBy.Descending takes almost twice as long as iterating through the same Select (with FOREACH or MoveNext). [tr#36148]

-- A Reset() is supposed to set the enumerator to its initial position before the first element in the selection, but when iterating on a Select with Orderby.Ascending or

Orderby.Descending, instead of the first row being returned, the next row is being returned. [tr#36149]

-- Using Where.In within a Select where one or more fields contain the same value results in the same matching record being returned more than once when optimized. [tr#36226]

-- (iOS) A JoinSelect causes a null reference under specific conditions when debugging in iOS. Some workarounds are to make the JoinSelect variable (jobj) be local data instead of a record variable, reference jobj using GetType(), or assign one of the inner joins into a temporary variable. [tr#36423]

-- (UNIX) When you execute an LPQUE command, the underlying shell is emitting escape sequences to reset terminal wrap, turn on the cursor, and reset all attributes. However, the shell should not emit any escape sequences for LPQUE. Note that this only occurs when running a UI Toolkit or low-level windowing application. [tr#23243]

-- %ERRMOD and the stack trace for a system error in UI Toolkit and the Synergy windowing API do not show the unmangled name. [tr#34310]

-- Certain XDL keywords like (NETWORK_ENCRYPT, TRACK_CHANGES, DUPLICATES, etc.) that have yes/no responses are not validated against invalid responses. [tr#34554]

-- Enabling a new network adapter or moving a mobile device that activates a new adapter (e.g., wired to wireless) causes xfServer access to fail with a network error when xfServer is running on the same machine as the client application. The workaround is to change the metric on the adapter in the IPv4 config settings. [tr#34498]

-- SORT/MERGE does not generate an error when one of the keyfields does not belong to the specified record. [tr#34813]

-- (.NET) An OPENELB of an assembly whose name has trailing spaces fails. You can use %ATRIM as a workaround. [tr#34392]

-- (.NET) A dynamic array of Synergy structure types causes a non-blittable error when it contains a boolean type. [tr#31494]

-- (OpenVMS) When calling ISAMC with a file specification that includes a file version number, the version is not honored, and each time ISAMC is called, a new version of the file is created. The workaround is to call XCALL DELET before ISAMC. [tr#35088]

-- (Windows, UNIX) Using a remote spec (@server) in a search list logical doesn’t work. "DAT=@myserver,dat" works only if the file exists on myserver; if the file is only in the dat directory (or any other directories following dat in the search list logical), it won’t be found. "DAT:dat,@myserver" works only if the file exists in the dat directory (or any other directories preceding the remote server spec in the search list logical). [tr#35395]

-- GetCTInfo on a non-change-tracking Select produces an object that generates the wrong information. [tr#35534]

-- (OpenVMS) A pipe command ignores the part of the command after the comma when the command is issued in an OPEN statement. The command will run as expected when the pipe is opened in one statement and the command is written to the channel with WRITES. [tr#35746]

-- Incorrect instantiation of the AlphaEnumerator, RestrictedAlphaEnumerator, and CTInfo classes should generate an error but doesn’t. [tr#36006]

-- (.NET) Using an integer in a (NoCaseWhere) expression causes an InvalidCastException of Select.Where to Select.NoCaseWhere. [tr#36475]

Synergy Configuration Program (SynConfig)

-- Changing the server name for SDE Client in the Advanced License Manager dialog isn’t working. [tr#36437]

Synergy DBMS

-- Attempting to run isutl -r, fconvert, or irecovr at the same time any other instance of isutl -r, fconvert, or irecovr is running generates the error "Operation not allowed due to backup mode." [tr#25195]

-- The isutl utility does not detect .is1 header corruption. [tr#23168, tr#23169]

-- When copying a file using fconvert (without a description file) that contains portable integers (file options ",i=n:m"), the portable integer definitions aren’t transferred to the new file. If a description file is used (-d) that contains the portable integer definitions, they are transferred correctly. [tr#32314]

-- When the ISAM file passed to isutl has a non-.ism extension, .ism is added to the end of the filename reported in the isutl log file (after the existing extension). [tr#35350]

-- When the ISAM file passed to isutl is specified using a logical, the logical is included in the filename reported in the isutl log file. [tr#35351]

-- When specifying a remote description file that is an FDL file, fconvert generates an "%FCONVERT: Invalid par file - Bad file option FDL_VERSION" error and fails. [tr#35868]

-- (Windows) Fcompare incorrectly reports data errors when a record contains a group field. [tr#36443]

-- Executing ipar with -g or -gx and specifying the file with a logical or complete path will result in either a 0-byte file with the name of the logical being created in the current directory, a PAR or XDL file named <logical>:<filename> created in the current directory, or the PAR or XDL file created in the directory in which the file is located. [tr#36495]

-- Running xdlchk on a file with an invalid numeric overlay reports the XDL file is valid. [tr#36551]

Synergy .NET assembly API

-- When using the Synergy .NET assembly API in conjunction with DevExpress 10.1.4, the embedded DevExpress undo/redo controls do not change to an enabled state when the text in the textbox changes. [tr#31554]

-- When an event is based on a delegate whose name is 30 characters or more, gennet40.exe will not include that event in the generated code. [tr#36490]

-- The runtime crashes when the assembly type parameter to DotNetDelegate is null or "". [tr#36520]

Synergy XML API

-- When an invalid XML file is placed into an XML string and this string is placed into a handle and then passed to %XML_PARSER_PARSESTRING, a "Map outside bounds of field or handle" error will occur, but only if the XML has all carriage return/line feeds removed when placed into the dynamic memory handle. [tr#36465]