Retail passion. Innovative solutions.

Retek® Extract Transform and Load™ 11.2.1.1

Release Notes

Corporate Headquarters:

Retek Inc.
Retek on the Mall
950 Nicollet Mall
Minneapolis, MN 55403
USA

888.61.RETEK (toll free US)
Switchboard:
+1 612 587 5000

Fax:
+1 612 587 5100

European Headquarters:

Retek
110 Wigmore Street
London
W1U 3RW
United Kingdom

Switchboard:
+44 (0)20 7563 4600

Sales Enquiries:
+44 (0)20 7563 46 46

Fax:
+44 (0)20 7563 46 10

Customer Support

Customer Support hours

Customer Support is available 7x24x365 via email, phone, and Web access.

Depending on the Support option chosen by a particular client (Standard, Plus, or Premium), the times that certain services are delivered may be restricted. Severity 1 (Critical) issues are addressed on a 7x24 basis and receive continuous attention until resolved, for all clients on active maintenance. Retek customers on active maintenance agreements may contact a global Customer Support representative in accordance with contract terms in one of the following ways.

Contact Method	Contact Information
E-mail	support@retek.com
Internet (ROCS)	rocs.retek.com Retek's secure client Web site to update and view issues
Phone	+1 612 587 5800

Toll free alternatives are also available in various regions of the world:

Australia	+1 800 555 923 (AU-Telstra) or +1 800 000 562 (AU-Optus)
France	0800 90 91 66
Hong Kong	800 96 4262
Korea	00 308 13 1342
United Kingdom	0800 917 2863
United States	+1 800 61 RETEK or 800 617 3835

Mail

Retek Customer Support
Retek on the Mall
950 Nicollet Mall
Minneapolis, MN 55403

When contacting Customer Support, please provide:

- Product version and program/module name.
- Functional and technical description of the problem (include business impact).
- Detailed step-by-step instructions to recreate.
- Exact error message received.
- Screen shots of each step you take.

Overview

The RETL 11.2.1 product release includes enhancements to performance, performance reporting, and logging as well as bug fixes to improve stability.

RETL 11.2.1 has been certified with the following applications:

- RMS 11.0.2
- RPM 11.0.1
- ReIM 11.0.2
- RDW 11.0

Note: RDW 10.2.2.1 is not certified with RETL 11.2.1. The certification process is planned for the first quarter of 2005.

It is strongly recommended that before any installation or development work is performed that the RETL 11.2.1 Programmer's Guide, provided with the release, is read cover to cover. In particular, read and understand “Chapter 2 – Installation and system configuration” as well as the “Known Issues” section of this document. Additionally, product groups should take care to extensively complete all certifications and regression/volume testing with RETL 11 versions before deployment to customer sites.

Clients that took delivery of the RETL 11.2 release, prior releases and/or subsequent patch releases, are strongly encouraged to upgrade to the RETL 11.2.1 release once a full regression test has been performed. Issues identified in RETL 11 releases and prior releases will be resolved with enhancements and fixes being delivered through RETL 11 patch releases. No additional releases of RETL 10.x are planned.

Note: An enhancement made in the RETL 11.2 release produces a warning about the output schema length for a field being less than the input schema length. For example:

WARN - W115: export:1: Output schema length for field LOC_IDNT (10) is less than the input schema length (25).

Data WILL NOT be truncated to match the output schema and the output data file may be improperly formatted.

This warning is saying that the entire range of values that can be stored in the RETL data type requires more characters than are specified in the export schema. To prevent data loss, the entire value will be written to the file, even if it requires more than the specified number of characters and thereby breaking the file’s fixed-length format.

This warning can be safely ignored if you know that the data length will never exceed the length specified in the export schema. This will commonly be the case when a NUMBER column is read from a database and later exported because the associated RETL data type’s range of values is larger than the database column’s range. In the example above, LOC_IDNT is defined in the database as NUMBER(10) but is processed in RETL as a dfloat, which requires 25 characters.

See the “Troubleshooting” section of the RETL 11.2.1 Programmer’s Guide for more details.

RETL 11.2.1 Improvements

This upgrade is recommended to take advantage of several improvements:

- Operator Enhancements
  - RETL 11.2.1's PARSER operator has been enhanced with the capability to send output to different datasets. This allows the PARSER operator to function like a FILTER, a SWITCH, or a COPY operator, or all of these within the same expression. Using the PARSER operator improves performance by reducing the number of operators and scales better when used with data partitioning.
  - Two new properties, progind and progfreq, have been added to each operator to allow the display of a progress indicator.
- Logging Enhancements
  - The performance monitoring of RETL 11 has been enhanced to create graphs showing processing time spent per operator. This provides a powerful tool for determining bottlenecks in flows.
  - RETL 11.2.1 now uses the log4j logging facility for all terminal and performance log output. The log4j facility is an extremely flexible open-source package maintained by the Apache Software Foundation. Refer to http://logging.apache.org/log4j/docs/documentation.html and the RETL 11.2.1 Programmer's Guide for more information on log4j.
- Stability: 20 bugs have been fixed since RETL 11.2 including:
  - Bugs were fixed where specifying a different number of partitions in different operators resulted in an exception in a FUNNEL operator or in a failed assertion.
  - A bug was fixed where the lookup table was not being loaded correctly in a LOOKUP operator when the second INPUT was not partitioned using a HASH operator.
  - The LOOKUP operator was enhanced to allow multiple threads to load the shared lookup table.
- Performance Enhancements
  - Processing of database DATE fields was enhanced to work around a single-threaded block of code in a Java library.
  - The DBLOOKUP now supports data partitioning.
- Platform certifications
  - HP-UX 11.23 Itanium 64bit was added as a supported platform in RETL 11.2.1.

RETL 11.x Improvements

- Partitioning/Sizing implementation - The concept of partitioning is to parallelize as much as possible, which in RETL, ideally results in a flow uninhibited by bottlenecks. The goal is to code flows so that they can be partitioned to perform within a customer's batch window.
- Major Performance Enhancements
  - RETL 11 releases include a new architecture that optimizes pipeline parallelism by consolidating connected operators into the same pipeline where possible. This reduces the number of threads required, minimizing context switching and thrashing among threads, and thereby allowing a framework that supports improved performance.
  - Enhanced funneling algorithm results in significant speed increases for flows that contain the funnel operator.
  - Internal types are now represented in native byte format, which saves space and results in increased performance from not having to convert to strings/etc.
  - Increased performance when handling rejects from the import operator. Users can now turn off 'verbosity' of printing rejects to the screen, which can result in vast improvements to RETL modules that reject records in import.
  - Significant improvements to performance in oraread and in the implementation of datasets that connect operators.
- Error handling
  - RETL 11 releases introduce better flow debugging by giving line and column specific error messages when RETL encounters problems reading an XML flow definition. Additionally, there is exception handling in RETL 11 that allows for better error handling and debugging.
  - Configuration of error threshold in orawrite. RETL can be configured to abort when reaching the maximum # of rejected records from SQL*Loader.
  - Overall improved error messages vs. 11.1 and 10.x versions.
- Debugging and Logging
  - RETL Visual flow graphs. Beginning with RETL 11.2, a command-line option can be used to generate a visual graph of a RETL flow, complete with operator names, link names, and thread identifications. See the RETL Programmer’s Guide for more info.
  - Enhanced performance logging. RETL 11.2 will now log significant events such as query execution, sorting via gsort, loading data via native database load utilities (e.g. SQL*Loader), and more.
  - Performance hotspots/bottleneck graph. RETL will now export an HTML graph of performance activity. This graph uses visual cues to point out bottlenecks in the flow so that flow developers, performance analysts, and services consultants can diagnose the cause of slow-performing flows.

- New Operators
  - PARSER operator. Business logic can be coded in a java-like code script, which eliminates proliferation of too many operators, improves performance by reducing the # of operators (e.g. FIELDMOD, BINOP), reduces maintenance costs by decreasing complexity of flows, and aids in a better sizeability when partitioning flows
  - DBLOOKUP operator. Allows the flow to join records directly to a table in the database.
  - CHANGECAPTURELOOKUP operator. Identical in functionality to changecapture, but can take unsorted input, which lends itself to better partitioning and sizing.
  - SPLITTER operator. A round-robin-based partitioning operator that allows equal distribution of records to each partition.
- Documentation
  - Much-improved Programmer's Guide
  - Performance Tuning Guide
  - Best Practices Guide
- Online Help - A command-line option has been added to RETL 11 that allows flow developers to view operator syntax and usage via the command-line without having to refer to this document.
- Simplified installation and configuration - RETL 11 releases consolidate code into a single binary rather than the 26 binaries of 10.x release. This considerably eases installation. In addition, there is less environment setup that needs to be done in order to complete an installation.
- True Platform Independence - The new architecture of the RETL 11 releases has been written entirely in Java. As a result, RETL may now run on platforms that previously would have required significant efforts to port to. See the Programmer's Guide and the Compatibility Matrix for supported platforms. As a direct result of the platform independence, significant RETL resources in platform maintenance will be freed up to develop new features going forward.
- True Database Independence - With the RETL 11 releases, generic 'dbread'/'dbwrite' operators have been introduced. These operators use JDBC-compliant technology to connect to databases in a truly database-independent manner. Additionally, previous 10.x releases were unable to connect to different database types in the same flow. RETL 11 releases allow full interoperability among different databases in the same flow (e.g. among DB2, Oracle, Teradata all in the same flow). See the programmer's guide for more information on this topic
- Backwards compatibility with previous releases - A requirement of RETL 11 versions is that they be backwardly-compatible with the 10.x versions. However, RETL 11 versions are stricter on enforcement of valid XML flow interfaces, input schemas, and data fields. In some instances, the RETL 11 product will find data and/or flow errors that may have previously been unreported. The result may be more discarded or rejected records and/or error messages than previously identified.

Compatibility Matrix

The following represents OS/Database combinations that are supported and have been certified to work properly:

OS	Version	Arch	DB	Version
AIX	5.1	32/64	none	none
AIX	5.1	64	Oracle	9.2
AIX	5.1	64	DB2	7.2.3
AIX	5.1	64	DB2	8.1.3
AIX	5.2	32/64	none	none
AIX	5.2	64	Oracle	9.2
AIX	5.2	64	DB2	7.2.3
AIX	5.2	64	DB2	8.1.3
HP-UX	11.11	32/64	none	none
HP-UX	11.11	64	Oracle	9.2
HP-UX	11.11	64	Oracle	9.0.1
HP-UX	11.11	32	Oracle	8.1.7
HP-UX	11.11	32	Teradata	2r4.2
HP-UX	11.23	IA64	none	none
HP-UX	11.23	IA64	Oracle	9.2
Solaris	8	32/64	none	none
Solaris	8	64	Oracle	9.2
Solaris	8	32/64	Oracle	9.0.1
Solaris	8	32	Oracle	8.1.7
Solaris	8	32	Teradata	2r4.2
Solaris	8	32	DB2	7.2.3
Solaris	8	32	DB2	8.1.3
Solaris	9	32/64	none	none
Solaris	9	9	Oracle	9.2
Solaris	9	32/64	Oracle	9.0.1
Solaris	9	32	Oracle	8.1.7
Solaris	9	32	Teradata	2r4.2
Solaris	9	32	DB2	7.2.3

OS	Version	Arch	DB	Version
Solaris	9	64	DB2	8.1.3

Note: RETL 11 only requires one binary and installation for all database/platform combinations listed above.

Installation

See the Programmer's Guide for installation and configuration instructions.

Package Contents

The following is a brief description of the contents of retl.11.2.1.tar.gz:

bin/

Includes executables required for running retl on different platforms.

File	Description
gsort.SunOS	gsort for SunOS
gsort.HP-UX	gsort for HP-UX
gsort.AIX	gsort for AIX
retl	RETL executable
rfx	Symbolic link to retl provided for backwards compatibility
verify_retl	The RETL verification tool
README.verify_retl	The README for verify_retl
install_graphing.ksh	Installs the GraphViz graphing package
install_jre.ksh	Installs the Java Runtime Environment
test_funcs.ksh	Used by verify_retl to run test flows

Note: There is only one binary (retl/rfx) required for all database and platform combinations

lib/

These are jar files that are needed in order to run retl.

File	Description
retl.jar	RETL runtime library
db2java.zip	DB2 JDBC Driver
log4j-1.2.9.jar	log4j Logging Facility
ojdbc14.jar	Oracle JDBC driver
teradata.jar	NCR Teradata JDBC driver
xercesImpl.jar	Xerces XML Parser
xml-apis.jar	XML APIs
xmlParserAPIs.jar	XML APIs
tools.jar	Future enhancements
activation.jar	Future enhancements

docs/

This directory contains the documentation associated with RETL.

File	Description
retl-1121-rn.pdf	This file.

etc/

This directory contains the documentation associated with RETL.

File	Description
rfx.conf	The default retl configuration file.
logger.conf	The default logger configuration file.

samples/

This directory contains the samples packaged with RETL. See the README in the samples/ directory for more information about the samples packaged with RETL. These are the samples that are run after the install, by verify_retl

graphing/

This directory contains the Graphviz application suite for graphical visualization of RETL flows. See the "Producing graphical output of flows with RETL" section of the Programmer's Guide.

JRE/

This directory contains the Java Runtime Environment (JRE) required by RETL.

Known Issues

General Known Issues

- RETL does not handle arbitrary precision math.
- The export operator will default to use '|'-delimited fields if there isn't a 'schemafile' property specified. This can cause problems if the incoming data contains the '|' character. The recommendation is to use an export schemafile if it is possible for data to contain a '|' character.
- Join operators may use a large amount of memory when the distribution of key-equivalent records is low and the number of records is high. It is important to test join operations with appropriate data volumes on each input side to verify the maximum amount of memory your application may require.

Differences between 10.x and 11.x versions of RETL

- RETL 11 versions are stricter on enforcement of valid XML flow interfaces, input schemas, and data fields. In some instances, the RETL 11 product will find data and/or flow errors that may have previously been unreported. The result may be more discarded or rejected records and/or error messages than previously identified.
- An unintended feature of 10.x allowed input data files to contain more fields than are specified in the schemafile. RETL 11 does not allow more fields to exist in the data file and will throw an exception should this happen. This is logged as bug#632 and will be fixed in a future version of RETL 11
- Output field order between 10.x and 11.x may be different for those flows that don't specify a 'schemafile' property in the export operator.
- RETL 11.x versions require additional properties for dbread/dbwrite operators. 'hostname' and 'port' are now required properties and may be specified in rfx.conf for convenience. See the programmer's guide for more information on syntax and usage.
- When upgrading an oracle database, a separate instance of RETL should be used in order to test the new database. This is because only one port can be specified in rfx.conf at a time. The other option is to use RETL with the '-c' option to use a different config file. See the programmer's guide for more info.
- RETL 11.x versions may require more physical memory for certain flows than 10.x releases.

Backwards-Compatibility Issues with 10.x versions

- It is not recommended to upgrade to RETL 11 unless the product using RETL has been certified on RETL 11.

Known bugs in RETL 11.2.1

- Bug#511 - Port is not configurable for db2 operators. This prevents RETL from running on a different server than the database.
- Bug#611 - Tests for table existence can return false matches if the user doesn't have sufficient privileges on a table.
- Bug#636 - The funnel operators use the schema of the first INPUT without checking for matching nullability. This may result in null values in non-null fields, which could result in breaking of export files, etc.
- Bug#646 - terawrite performs a 'delete' on a table when TERAWRITE's 'mode' property == 'truncate'. This results in slow truncation on large tables.
- Bug#652 - NOT can only be used once in filter expression. A workaround is to negate logic in the expression (e.g. NE instead of EQ)
- Bug#659 - A pipe delimiter is used when paging datasets to disk. This will cause problem re-importing if there are pipes in the data.
- Bug#693 - terawrite doesn't handle timestamp fields
- Bug#808 - Database table creation errors messages are lost when table creation fails. Generic error message is displayed instead.
- Bug#820 - Export file schema can be broken when the input data is cached for numeric values and the export field length is less than the imported field length.
- Bug#840 - Cliprows displays incorrect message about possible incorrect results when the input data is sorted in descending order.
- Bug#856 - If a record is altered by a binop operator after the record has been exported and then it is re-exported, the second export will have incorrect fields because the record's cache is not cleared by the binop.
- Bug#916 - If an integral field is multiplied by a constant right floating point value in a binop operator, the result will always be 0.
- Bug#1105 - fieldmod duplicate does not check for pre-existing field.

Changes

Changes since RETL 11.2

See above for enhancements in the 11.2.1 release

- Fixes Bug#434 - Allow logging to log actual times spent per operator
- Fixes Bug#544 - Missing CONVERTFUNCTION Results in No-op Instead of Error
- Fixes Bug#860 - 4 RETL Instances Much Quicker than 4 ORAREADs
- Fixes Bug#905 - Progress indicators
- Fixes Bug#906 - Performance statistics show operator initialization time under framework
- Fixes Bug#934 - dblookup does not partition
- Fixes Bug#936 - Partitioning error when funneling multiple partition groups with different numpartitions
- Fixes Bug#942 - Incorrect schema resolution in generator after inline funnel
- Fixes Bug#945 - negative dfloats can break fixed-length output schema
- Fixes Bug#946 - Occasional NullPointerException in funnel
- Fixes Bug#953 - Join key datatypes must match
- Fixes Bug#955 - Enhanced logging
- Fixes Bug#957 - Assertions should be enabled when RFX_DEBUG=1
- Fixes Bug#961 - Partitioned lookup is incorrect when lookup table is non-keyed partitioned
- Fixes Bug#963 - Mismatched numpartition settings cause assertion error.
- Fixes Bug#964 - Date comparison failure in parser
- Fixes Bug#966 - Should be able to specify multiple outputs in parser
- Fixes Bug#967 - IndexOutOfBoundsException in orawrite
- Fixes Bug#969 - Groupby errors on null value for min/max
- Fixes Bug#970 - Double formatting with magnitude < 0
- Fixes Bug#971 - first field nulls not handled properly in comparison and assignments.
- Fixes Bug#972 - Enhance graphing with performance measurements
- Fixes Bug#973 - Verify graphing performance measurement accuracy
- Fixes Bug#974 - Add and adjust dataset monitoring capabilities
- Fixes Bug#975 - Improve the tests surrounding the new parser functionality
- Fixes Bug#981 - Partitioned lookup may not scale
- Fixes Bug#982 - Delimited orawrite should specify length for CHAR
- Fixes Bug#984 - DEFAULTS in rfx.conf do not override operator properties

- Fixes Bug#985 - NullPointerException from StreamRedirector
- Fixes Bug#986 - Comparison between different field types gives incorrect error
- Fixes Bug#990 - dblookup sets incorrect SQL to look up in database
- Fixes Bug#994 - -g must be last option on command line
- Fixes Bug#995 - Convert can invalidate sort order on presorted data

Changes since RETL 11.1

- Fixes Bug#407 - In framework we tread all int and uint type as long in Java version
- Fixes Bug#421 - rejected records get printed to the screen which can produce too much output
- Fixes Bug#441 - Support for properly-formatted precision in dfloat/sfloat
- Fixes Bug#445 - parallel file reads
- Fixes Bug#461 - Include Specific Error in Import Rejects
- Fixes Bug#464 - determine if we really need hostname/port for oraread/write
- Fixes Bug#484 - Need Build Information Option
- Fixes Bug#488 - Better exception handling in Operator base class
- Fixes Bug#493 - Record Key Hash Non-Unique with Multiple Key Fields
- Fixes Bug#499 - Import Doesn't Use nullvalue Property to Detect Null Values
- Fixes Bug#503 - Fixed Width of Zero Not Allowed
- Fixes Bug#507 - Schema.parseKeyFields(...) should be more robust
- Fixes Bug#508 - Number Format is not Standardized on Export
- Fixes Bug#510 - userid/password properties don't exist for db2read/teraferead operators
- Fixes Bug#515 - jdbcwrite should set proper type in constructor
- Fixes Bug#516 - Implement native load mechanisms for orawrite in 11.x versions
- Fixes Bug#517 - Implement native load mechanisms for terawrite in 11.x versions
- Fixes Bug#522 - generator doesn’t implement partnum_offset or partcount_incr
- Fixes Bug#532 - final_delimiter not Implemented
- Fixes Bug#534 - Missing len From Fixed Schema File Causes Delimited Export
- Fixes Bug#541 - Incorrect Handling of Too Few Fields in Import
- Fixes Bug#543 - Multiple Use of Same Dataset as Input Throws Null Pointer Exception
- Fixes Bug#547 - JDBCWrite Should Check Schema Owner
- Fixes Bug#548 - port is not configurable for Teradata (terawrite)
- Fixes Bug#553 - checks if there is sqlldr
- Fixes Bug#555 - Strip Schema Attribute Not Handled

- Fixes Bug#557 - fieldmod doesn't check if fieldnames actually exist in incoming dataset
- Fixes Bug#558 - createtablemode should only allow recreate/create
- Fixes Bug#561 - RETL doesn't abort gracefully if dbwrite password not specified
- Fixes Bug#564 - should modify maxsize if database column maxsize < field maxsize
- Fixes Bug#572 - Merge jdbcread methods with DatabaseUtils and ColumnInfo
- Fixes Bug#574 - logical error in changecapture
- Fixes Bug#579 - Fatal error trapping when using SQL-loader
- Fixes Bug#606 - Handling environment problems with retl executable/library
- Fixes Bug#610 - In the fixed import, we should check if the total len of the record equals sum of the len of fields
- Fixes Bug#632 - Incorrect Handling of Too Many Fields in Import
- Fixes Bug#635 - Generator Doesn't Create Null Values Same as 10.x
- Fixes Bug#641 - Generator Sequence Doesn't Respect Destination Field Datatype
- Fixes Bug#643 - Convert Can't Handle Duplicate Source Fields in One CONVERTSPEC
- Fixes Bug#647 - Reject File Incorrect Format
- Fixes Bug#648 - Incorrect Contents of Reject File
- Fixes Bug#675 - Sort of Generated String Fails on Re-Import
- Fixes Bug#678 - Multibyte Fixed Width Export Incorrect/Slow
- Fixes Bug#685 - funnel shouldn't be waiting when there is not data in any one of inputs
- Fixes Bug#686 - Dataset should check with deadlock listener always even though the dataset has been expanded.
- Fixes Bug#694 - Non-UTF Encodings Cause Exception in Import
- Fixes Bug#696 - 11.x can break the output schemafile definition
- Fixes Bug#698 - sortfunnel outputs wrong data --- logical problem
- Fixes Bug#700 - Generator Sequence Doesn't Work With Doubles
- Fixes Bug#701 - Binop Incorrect When Destination Field is a Source Field
- Fixes Bug#703 - Improved performance when handling errors
- Fixes Bug#705 - Performance bottlenecks in orawrite.getRecordStringBuffer(...)
- Fixes Bug#708 - Performance Improvement in jdbcread
- Fixes Bug#709 - Performance Degradation in jdbcread Due to String Comparison
- Fixes Bug#710 - sort doesn't handle non-existent 'key' gracefully
- Fixes Bug#713 - changecapture Codes Incorrect in Documentation
- Fixes Bug#714 - Error message bad when null value is in non-null field
- Fixes Bug#716 - Delimiter Property of Sort Not Working Correctly

- Fixes Bug#718 - NullPointerException in Lookup
- Fixes Bug#719 - rightouterjoin Invalid Schema
- Fixes Bug#721 - Import of Multi-Byte Fixed-Width Files Fails
- Fixes Bug#722 - ConvertByteToBytes Incorrect
- Fixes Bug#723 - LONGVARCHAR Only Partially Supported
- Fixes Bug#724 - cannot handle "select NULL something from test_jdbc"
- Fixes Bug#725 - Generator Backwards Compatibility Issue with Field Length
- Fixes Bug#726 - ArrayIndexOutOfBoundsException when Missing Field Type in Schema
- Fixes Bug#727 - NullPointerException when numloader > 1 and RFX_SHOW_SQL=1
- Fixes Bug#728 - I18N and Fixed Width Incorrect
- Fixes Bug#729 - The dataset hang when generator-->noop with independent thread with very small buffersize and huge data
- Fixes Bug#734 - NullPointerException when Paging
- Fixes Bug#736 - ArrayOutOfBounds in export
- Fixes Bug#737 - Rejects File Not Flushed When Exiting Due To Too Many Rejects
- Fixes Bug#738 - parallel orawrite doesn't alternate TEMPDIRs
- Fixes Bug#739 - tempdir property should override setting in rfx.conf
- Fixes Bug#740 - intra-operator parallelism properties should have standard names
- Fixes Bug#743 - temporary directories should be checked
- Fixes Bug#745 - Schema Assembler Should Ensure Field Delimiter != Record Delimiter
- Fixes Bug#746 - Orawrite Should Ensure Output Delimiter != Record Delimiter
- Fixes Bug#747 - orawrite cannot handle preload and postload for stored procedure
- Fixes Bug#748 - Convoluted Error Output When Setting Properties
- Fixes Bug#749 - Issue Warning Instead of Error for Unrecognized Properties
- Fixes Bug#751 - Combine DataSet, SDataSet, DataSet_Simple, and DataSet_RawBytes
- Fixes Bug#752 - add --graphviz-files
- Fixes Bug#753 - sp_prequery should be executed BEFORE the query
- Fixes Bug#754 - heterogeneous output schemas
- Fixes Bug#756 - Add 64bit JVM support in RETL 11
- Fixes Bug#757 - Convert doesn't handle TYPEPROPERTY properly
- Fixes Bug#758 - Int32/Int64 Field type rollover
- Fixes Bug#760 - Binop odd division rounding error
- Fixes Bug#761 - Lookup Returns False Matches

- Fixes Bug#762 - Unsigned Integer issues
- Fixes Bug#764 - NumberUtils.ConvertIntToBytes Poor Algorithm
- Fixes Bug#769 - Sort Failure on Date without Century
- Fixes Bug#771 - Sort Isn't Removing Temporary Files
- Fixes Bug#772 - RETL 11.x Doesn't Support Partitioning
- Fixes Bug#774 - Lookup ArrayIndexOutOfBoundsException with Multiple Lookup Tables
- Fixes Bug#776 - Out of Memory Exception Should Abort Flow
- Fixes Bug#779 - Fieldmod Loses SortKeys
- Fixes Bug#780 - Schema Assembler Sets Sort Keys Incorrectly
- Fixes Bug#782 - Default Value for Properties Not Set for Join
- Fixes Bug#784 - graphviz cleanup
- Fixes Bug#786 - Should error when record len != sum(field lens) in schemafile
- Fixes Bug#788 - Forwards compatibility for operator properties
- Fixes Bug#790 - Join with database operator
- Fixes Bug#791 - Groupby Requires reduce Property before Min, Max, ... Fields
- Fixes Bug#792 - Verify retl.jar during retl startup
- Fixes Bug#793 - Verify retl.jar during retl startup
- Fixes Bug#796 - enhanced performance logging
- Fixes Bug#799 - Deadlock Detection not Working for Pipelines with More than Two Outputs
- Fixes Bug#800 - Occasional Errors when Paging
- Fixes Bug#801 - Paging May Result in Too Many Files Open
- Fixes Bug#802 - DataSet Paging Files May Not Be Created
- Fixes Bug#803 - Unneeded check for duplicate tables in existence
- Fixes Bug#811 - leftouterjoin Doesn't Support Spaces in nullvalue Property
- Fixes Bug#812 - fieldmod blows away sorted order
- Fixes Bug#813 - rightouterjoin Invalid Sort Order
- Fixes Bug#816 - Record.Join uses incorrect calculation to get # of fields
- Fixes Bug#818 - Poor Message Displayed when Output Record breaks Schema
- Fixes Bug#821 - Delimited Import Strips Trailing Whitespace in 11 But Not in 10
- Fixes Bug#822 - Package debug/non-optimized version of retl.jar
- Fixes Bug#823 - Programmer's Guide Description of ifnotfound Property is Incorrect
- Fixes Bug#824 - Improve logging information with flow name
- Fixes Bug#825 - Paths in retl Script Get Fouled up on Windows

- Fixes Bug#826 - Add 'loader_options' parameter to orawrite
- Fixes Bug#829 - Maxlength of 0 Messes up Orawrite
- Fixes Bug#830 - slsildmdm runs faster in 10.x than 11.x
- Fixes Bug#832 - Incorrect Null in Fullouterjoin
- Fixes Bug#833 - Error in Double to Byte
- Fixes Bug#834 - dotty files removed with -g option
- Fixes Bug#835 - need specific error message in oraread query
- Fixes Bug#836 - Missed Record in Multi-Threaded Import
- Fixes Bug#838 - Stack Overflow in Generator if limit < init
- Fixes Bug#839 - Copy Doesn't Perform Deep Enough Copy
- Fixes Bug#841 - dfloat fields can break fixed output schema
- Fixes Bug#842 - Lookup Table Loading is Dependant on Data Record Production
- Fixes Bug#843 - assertion is fired in allocation size_profile script
- Fixes Bug#845 - Assertion in Parser_ByteSequence.readFixedField in Allocation size_profile
- Fixes Bug#847 - Pipeline Initialization and Schema Resolution is Serial
- Fixes Bug#848 - ArrayIndexOutOfBoundsException when 1 INPUT Provided to lookup
- Fixes Bug#852 - Operator jdbcwrite Needs parallel Property
- Fixes Bug#853 - Check for WHERE in dblookup select_sql Property Incorrect
- Fixes Bug#854 - Partitioned Orawrite Runs preload and postload for each Partition
- Fixes Bug#855 - db2write deletes log files even after a fatal error
- Fixes Bug#859 - Monitor Method of Funnel Doesn't Work for Empty Datasets
- Fixes Bug#860 - 4 RETL Instances Much Quicker than 4 Orareads
- Fixes Bug#862 - Adding batched funnel algorithms causes dataset._pblock to be null
- Fixes Bug#864 - should print a warning if orawrite can't be partitioned
- Fixes Bug#867 - verboserejects should be a visible property in import
- Fixes Bug#868 - retl doesn't print any warning if rejects + verboserejects=false
- Fixes Bug#869 - Errors not caught in DatabaseUtils.Connect()
- Fixes Bug#871 - ImportRejectSupporter throws an error on processing rejected records
- Fixes Bug#872 - Multiple-query JDBCRead is slower than Multiple JDBCReads
- Fixes Bug#873 - jdbcread needs to partition
- Fixes Bug#874 - Import partitioning doesn't calculate number of partitions correctly
- Fixes Bug#875 - Import partitioning doesn't need to end thread
- Fixes Bug#877 - verify_retl chokes on space between -d and database type

- Fixes Bug#881 - Schema compatibility checking in export is too strict
- Fixes Bug#883 - Partitioning fails in aipt_future_delivery
- Fixes Bug#887 - Execute jdbcread subthreads in parallel
- Fixes Bug#888 - Dataset-less Funnel
- Fixes Bug#889 - Check for retl.jar in retl script doesn't check RETL_DIST_LIB variable
- Fixes Bug#890 - Funnel should have 11.1 algorithm in 11.2
- Fixes Bug#891 - Generated HTML for graphs contains incorrect message about dotty file
- Fixes Bug#892 - Funnel should drop sort order
- Fixes Bug#895 - Add create_options property to orawrite
- Fixes Bug#901 - Include 1.4.1 AIX JRE in package rather than 1.4.0
- Fixes Bug#903 - Inlining lookup may improve performance
- Fixes Bug#904 - Splitter needs to end thread when partitioning
- Fixes Bug#907 - Non-deterministic output if debugging is set on
- Fixes Bug#910 - Incorrect schema resolution for fullouterjoin
- Fixes Bug#911 - Binop produces incorrect field type when multiplying int by dfloat
- Fixes Bug#912 - NullPointerException in binop when left or constleft not specified
- Fixes Bug#913 - Conflicting versions of rfx.conf num partitions and -n option do not seem to resolve properly
- Fixes Bug#919 - Import partitioning doesn't work with multiple files
- Fixes Bug#920 - Poor error message when nullable missing in import schemafile
- Fixes Bug#921 - Hash partitioning after splitter uses incorrect dimensions
- Fixes Bug#922 - Import - Multi-Threaded imports are losing records
- Fixes Bug#923 - Export should partition
- Fixes Bug#924 - FileUtils concatenation() errors when output file doesn't exist yet
- Fixes Bug#925 - Misleading error message "record was too long" in import
- Fixes Bug#927 - Changecapture algorithm doesn't allow for partitioned imports
- Fixes Bug#929 - install_jre fails on some servers
- Fixes Bug#930 - Cannot override default logger file from config file
- Fixes Bug#931 - Log file should be in temp directory if directory not specified
- Fixes Bug#932 - Unable to write to log file
- Fixes Bug#935 - changecapturelookup interferes with lookup's tablekeys property

Changes since RETL 11.0

- Implements native database load utilities for orawrite and terawrite
  - orawrite - implements SQL*loader
  - terawrite - implements mload/fastload
- Adds increased parallelism & configurability for the orawrite operator
  - Properties NUMLOADER, ROWS, SINGLEROW, SORTEDINDEXES have been added. See the Programmer's Guide for more information on syntax usage
- Fixes Bug#396 - generator doesn't handle delimiter when defining fields
- Fixes Bug#406 - changing input on jdbcwrite will cause errors
- Fixes Bug#426 - give better error message when connecting to database fails
- Fixes Bug#465 - Sort Occasionally Hangs.
- Fixes Bug#496 - Merge backward bug
- Fixes Bug#525 - Generator Appends Fields at End Instead of at Beginning
- Fixes Bug#529 - Export File Format Inconsistency with 10.x
- Fixes Bug#530 - Import Inconsistency with 10.x with Trimming Whitespace
- Fixes Bug#531 - Assignment of Null Value to a Non-Nullable Field
- Fixes Bug#539 - Blanks Not Allowed for Null Dates
- Fixes Bug#545 - Exception when writing to table with more columns than input
- Fixes Bug#546 - Filter errors on checking IS_NULL for non-nullable fields
- Fixes Bug#552 - Logger level 2 is broken
- Fixes Bug#556 - outerjoin doesn't rename incoming duplicate fields correctly
- Fixes Bug#559 - Allow configurability of orawrite parameters
- Fixes Bug#567 - "Migrate jdbcwrite, orawrite, terawrite to use common jdbc utilities"
- Fixes Bug#568 - RDW flow slsildmex produces incorrect data
- Fixes Bug#569 - 10.x versions write latest duplicated column to db but 11.x
- Fixes Bug#570 - Funnel Does Not Allow for Equivalent Schemas
- Fixes Bug#571 - Bug in AreColumnInfosCompatible
- Fixes Bug#575 - backwards compatibility problem when truncating existing table
- Fixes Bug#576 - Binop Doesn't Set Max Length of New Fields
- Fixes Bug#577 - Orawrite writes out multiple NULLIF statements to the SQL*Loader
- Fixes Bug#578 - Binop Doesn't Handle -1 Left Hand Side Constants
- Fixes Bug#581 - Problems with New Fields Generated by Binop
- Fixes Bug#582 - columnInfo doesn't get populated for orawrite/jdbcwrite

- Fixes Bug#584 - Number * Null Gives Exception
- Fixes Bug#585 - First Record Containing Null Field Causes NullPointerException
- Fixes Bug#586 - Null Exception in Filter
- Fixes Bug#587 - Import Error in slsildmex
- Fixes Bug#589 - Convert Doesn't Handle Multiple CONVERTFUNCTION
- Fixes Bug#590 - orawrite doesn't drop data from INFILE
- Fixes Bug#592 - orawrite control file len is off by one
- Fixes Bug#593 - 'len' in orawrite control file should be based on actual field
- Fixes Bug#594 - changecapture incorrectly drops fields in certain cases
- Fixes Bug#596 - There are new line in value of orawrite tablename property in RDW flow
- Fixes Bug#597 - CompareDifferentSchemas fails comparing records with null fields
- Fixes Bug#600 - leftouterjoin Drops Record(s)
- Fixes Bug#602 - Dataset should generate unique temp file.
- Fixes Bug#603 - sortfunnel doesn't work if the field order is different in input schemas
- Fixes Bug#604 - more problems with column->field mappings in orawrite
- Fixes Bug#607 - error when re-importing from a sort
- Fixes Bug#608 - Backwards-compatibility problem in groupby
- Fixes Bug#627 - Numeric Database Type Not Converted to RETL Type Correctly
- Fixes Bug#629 - Bug in checking actual RETL types
- Fixes Bug#630 - sortfunnel should use schema/fields from LHS
- Fixes Bug#631 - orawrite doesn't handle null fields
- Fixes Bug#633 - 10.x funnel allows conversion nullable->non-nullable
- Fixes Bug#634 - Regression introduced in checkin to bug#589
- Fixes Bug#637 - "leftouterjoin, rightouterjoin, and fullouterjoin have to change output schema the nullable=false to true if the null will be set to the field."
- Fixes Bug#644 - Binop Destination Field Nullability Incorrect
- Fixes Bug#645 - orawrite does an unnecessary delete when truncating table
- Fixes Bug#650 - thread corruption of date/time/timestamp fields
- Fixes Bug#653 - Sort Not Working in slsmkdnildex.ksh
- Fixes Bug#654 - string cache for numeric fields can be invalid
- Fixes Bug#656 - binop always uses Field_Double to add constright/constleft
- Fixes Bug#657 - Adding space to field in binop throws NullPointerException
- Fixes Bug#663 - RDW factopendm.ksh Has Null Pointer Exception in Fieldmod

- Fixes Bug#667 - JDBCRead Doesn't Allow userid and password Properties
- Fixes Bug#668 - JDBCRead Doesn't Support schemaowner Property Required
- Fixes Bug#669 - Exception Handling Dates in Teradata
- Fixes Bug#670 - Exception in Teradata JDBCRead
- Fixes Bug#671 - orawrite supports multiple SQL*Loaders
- Fixes Bug#672 - RDW Load Fails with Japanese Characters
- Fixes Bug#677 - Terawrite Unable to Create Tables
- Fixes Bug#679 - NullPointerException in Terawrite
- Fixes Bug#680 - Exception in Funnel tsildqdm.ksh
- Fixes Bug#681 - "Terawrite ""Operation not allowed"" Exception"
- Fixes Bug#682 - slsmkdnilddm Fails on Load
- Fixes Bug#683 - Incorrect Column Datatypes in DWI in pre_dwi_temp.ksh
- Fixes Bug#687 - Incorrect Datatypes Created by JDBCRead
- Fixes Bug#689 - Incorrect Max Field Length in JDBCRead
- Fixes Bug#691 - Join should buffer records on RHS instead of LHS
- Fixes Bug#416 - -sSCHEMAFILE option doesn't print actual datatypes
- Fixes Bug#533 - Switch Doesn't Handle Quoted Case Values
- Fixes Bug#540 - Nulls Not Detected Correctly on Input
- Fixes Bug#551 - Datasets that page to disk will never re-use memory
- Fixes Bug#580 - Funnel Transforms Records Unnecessarily
- Fixes Bug#583 - Sortfunnel Does Not Allow for Equivalent Schemas
- Fixes Bug#599 - terawrite Doesn't Drop Data
- Fixes Bug#601 - Schema is Not Shown Unless All Schemas Are Resolved
- Fixes Bug#609 - orawrite/jdbcwrite can't obtain table information in certain
- Fixes Bug#651 - Default output delimiter should be pipe '|'
- Fixes Bug#655 - Sum in Groupby Fails on Null Field
- Fixes Bug#661 - sort operator doesn't delete the temp files
- Fixes Bug#664 - AreSchemasCompatible Is Too Strict On Float Data Types
- Fixes Bug#588 - The '|' or other delimiter can not be content of data field
- Fixes Bug#619 - We may need to add preload or postload in orawrite.
- Fixes Bug#620 - "need ""arraysize"" property in jdbcread like C++ does"
- Fixes Bug#621 - We should be able to let user specify the control file name
- Fixes Bug#691 - Join should buffer records on RHS instead of LHS

Changes since RETL 11.0 BETA

- Fixes Bug#405 - recreating a table has problems
- Fixes Bug#409 - IS_NULL operation doesn't check for null
- Fixes Bug#410 - Filter doesn't work on dates
- Fixes Bug#412 - switch doesn't work without ifnotfound
- Fixes Bug#413 - Innerjoin is broken
- Fixes Bug#414 - groupby requires key property
- Fixes Bug#415 - Sortfunnel doesn't handle null records
- Fixes Bug#417 - jdbcread doesn't handle fields w/ null values correctly
- Fixes Bug#418 - join shouldn't rename fields to include "left" and "right" prefixes
- Fixes Bug#419 - output schemafile cannot reorder or drop fields
- Fixes Bug#420 - Fieldmod performs operations sequentially which causes an error in schemas
- Fixes Bug#423 - Copy operator doesn't clone schemas
- Fixes Bug#424 - unable to compare float and int in changecapture
- Fixes Bug#425 - jdbcwrite should check if getField is null before writing to db
- Fixes Bug#427 - verify_retl on RETL 11.0
- Fixes Bug#429 - i18n problem with import
- Fixes Bug#430 - i18n orawrite problem
- Fixes Bug#431 - outerjoin flavors don't support nullvalue property
- Fixes Bug#432 - lookup mishandles null records
- Fixes Bug#433 - Join hangs
- Fixes Bug#435 - export should allow implicit conversion from non-nullable to nullable
- Fixes Bug#437 - static final configuration variables are not configurable
- Fixes Bug#438 - temporary flow files created by retl should be removed upon exit
- Fixes Bug#444 - Copy after Generate Results in Incorrect Results
- Fixes Bug#448 - Need Time/Timestamp datatypes for 11.0
- Fixes Bug#449 - JDBCWrite doesn't work with dfloat
- Fixes Bug#450 - JDBCRead doesn't read dfloats properly
- Fixes Bug#451 - removedup doesn't handle null records properly
- Fixes Bug#452 - switch doesn't allow multiple values to be on same output
- Fixes Bug#453 - Collect operator isn't Implemented
- Fixes Bug#454 - numeric overflow in jdbcread

- Fixes Bug#455 - RETL hangs on certain RDW flows
- Fixes Bug#457 - Copy doesn't clone record in certain cases
- Fixes Bug#458 - export operator doesn't support 'mode' property
- Fixes Bug#459 - dbwrite operators don't have 'primaryindex' property
- Fixes Bug#460 - Date, Time, Timestamp in JDBCRead / Write
- Fixes Bug#462 - Need to propagate schemas through each operator
- Fixes Bug#463 - Backwards-compatibility bug in JDBCWrite
- Fixes Bug#466 - Problem with JDBCWrite table with schema owner attached
- Fixes Bug#467 - Date, Time, Timestamp in JDBCWrite
- Fixes Bug#468 - Support Date, Time, Timestamp in JDBCRead
- Fixes Bug#469 - JDBCRead Doesn't Support TO_CHAR Conversion
- Fixes Bug#471 - Generator has problems being both source and serial operator
- Fixes Bug#472 - Required nullvalue isn't enforced
- Fixes Bug#473 - Null Value with Table Recreate Causes Exception in Orawrite
- Fixes Bug#474 - Null pointer exception in convert when making field nullable
- Fixes Bug#475 - Dfloat nullvalue doesn't work properly
- Fixes Bug#476 - Lookup fails with an ArrayIndexOutOfBoundsException
- Fixes Bug#477 - JDBCWrite Throws Exception with Dates and createtablemode="create”
- Fixes Bug#478 - In the lookup the key fields in the "tablekeys" are separate
- Fixes Bug#485 - Lookup Gives Incorrect Results using Multiple Keys
- Fixes Bug#486 - NullPointerException in leftouterjoin variants
- Fixes Bug#487 - ArrayOutOfBounds exception in join variants
- Fixes Bug#490 - ArrayIndexOutOfBounds in binop operator
- Fixes Bug#491 - filter does not require second output when rejects=true
- Fixes Bug#497 - Null Date Truncated on Fixed Export
- Fixes Bug#498 - Null Value Length Not Enforced for Fixed Width Schemas
- Fixes Bug#500 - NullPointerException in Parser_CharSequence
- Fixes Bug#501 - NullPointerException in doResolveSchemas()
- Fixes Bug#502 - NullPointerException in Record.hashKey()
- Fixes Bug#504 - Chained Lookups Do Not Work
- Fixes Bug#505 - Import Rejects Record if Numeric Value Has Trailing Spaces
- Fixes Bug#506 - Lookup 'tablekeys' property should accept comma-delimited list
- Fixes Bug#509 - OpProperties are shared among operators

- Fixes Bug#520 - Operator_Join can not run if both input have the same field
- Fixes Bug#521 - JDBC write can not run if no. of import is more than no. of output