RiverWare
Releases, Software Maintenance, and Development Environment Work Summary – Fiscal Year 2010

Work accomplished by CADSWES in these categories from October 1, 2009 through Sept 30, 2010 is summarized below by major topic.

Releases, Patches and Snapshots

There were eleven patch releases (5.1.6 – 5.1.7 and 5.2.1 – 5.2.9) as well as one major release (5.2) of RiverWare generated this year. The work involved in generating a patch release is very similar to a major release, except that a major release typically has an extended period dedicated to bug fixing prior to the release and a more extensive set of release notes. The tasks involved in generating releases includes organizing and writing the release notes, fixing bugs and testing, managing the transition of code from the development area to the release area, administering and updating the configuration files for building the software, compiling and verifying the builds, setting up a regression test area for release test runs, creating the release package, testing the installation, and making the executables available on the CADSWES web site and ftp site.

Patch Release 5.1.6 was released October 15, 2009
Patch Release 5.1.7 was released November 23, 2009.
Release 5.2 was released December 30, 2009
Patch Release 5.2.1 was released January 22, 2010.
Patch Release 5.2.2 was released February 5, 2010
Patch Release 5.2.3 was released April 8, 2010
Patch Release 5.2.4 was released April 28, 2010
Patch Release 5.2.5 was released May 21, 2010
Patch Release 5.2.6 was released June 16, 2010
Patch Release 5.2.7 was released August 12, 2010
Patch Release 5.2.8 was released August 31, 2010
Patch Release 5.2.9 was released Sep 14, 2010

Thirteen snapshots of the builds development area were made over the course of the fiscal year to allow users to test newly implemented features before they became part of an official RiverWare release.

Development Environment

The Development Environment category includes all of the associated software and tools used for developing RiverWare as well as their setup and interaction. Work completed this year in this category is presented below by major topic.

64 Bit Windows Port and New Third-Party Libraries

This work involves two commingled tasks:

- 64-bit port of RiverWare on Windows
- Migrating the development environment to new versions of third-party libraries used with RiverWare, including Qt (4.6.3), Qwt (5.2.1), Tcl (8.5.1) and CPLEX (12.2)

Research was done into all the pieces required to generate a 64-bit version of RiverWare on Windows. These included hardware, Visual Studio compilers, RiverWare libraries, 3rd-party libraries, and software tools. A document summarizing the information was generated

(/projects/riverware/doc/simlib/64BitPlan.fm). Based on the information collected, no show stoppers were identified for the porting of RiverWare to a 64-bit executable, so it was decided to move ahead with this port and at the same time use this opportunity to upgrade the third-party libraries used in RiverWare.

At CADSWES, we compile Qt, Qwt, Tcl, ClientServer and qtservice. ClientServer and qtservice aren't new versions, but do need to be recompiled with new versions of Qt. The newer 5.2.1 version of Qwt, the plotting package utilized in RiverWare, was downloaded and built for Windows (32 and 64 bit) and Solaris. Because of renaming of some Qwt methods called by RiverWare, some minor changes needed to be made to RiverWare code to build RiverWare with the new version of Qwt. The newer version of Qt (Qt 4.6.2) was also acquired, configured and built on 32-bit and 64-bit Windows only. ClientServer was then built with the new version of Qt on Windows.

The arduous task of compiling Qt on Solaris was then started. There were compilation errors, which fortunately weren't in Qt proper, but instead were in various Qt add-ons (although this was not apparent at first). Previous versions of Qt required code changes to compile on Solaris so this approach was taken first. Eventually compilation errors were encountered that couldn't be resolved (missing definitions in #include files, for example). At this point, a different path was taken of trying to determine if the code that wouldn't compile was in add-ons that could be removed from the configuration. This was very time consuming because running the configuration script took several hours, as the script compiles code snippets to determine which features the compiler supports. Because Qt was reconfigured for each compilation, it was best to do a clean compile, and compiling Qt took several more hours before encountering the next compilation error. Qt 4.6.3 was eventually successfully compiled on Solaris, and Qt 4.6.3 versions of Qwt and ClientServer were successfully compiled on Solaris.

The effort switched back to 64-bit Windows, where 64-bit Tcl 8.5.1 was compiled (which required several Tcl code changes) and 64-bit CPLEX 12.1 was installed. 64-bit RiverWare was compiled with these, which required numerous code changes caused by the Tcl code changes. Tcl was then recompiled with a different set of Tcl source code changes, greatly reducing the RiverWare code changes needed. CPLEX 12.2 was installed, which required resolution of licensing issues that prevented the Optimization regression tests from running.

At this time, a 64-bit RiverWare was ready to run, except it wouldn't run, due to problems with the Microsoft VC90 Debug side-by-side assembly. Lengthy investigation revealed that the problem was caused by how Visual Studio was installed. Although the Visual Studio installation procedure indicates components can be installed piecemeal, in reality it doesn't necessarily work that way. Reinstalling Visual Studio in one fell swoop fixed the problem.

Finally a 64-bit RiverWare executable was created, which successfully ran several test models to completion. The next step was to run the regression tests to check if model results were correct with the new executable. However, the regression test scripts wouldn't run, due to lack of support for the Tk module in more recent versions of Perl. Older versions of Perl, which support the Tk module, have been effectively erased from the Internet. (ActiveState, which provides the most recent version of Perl for free, provides older versions of Perl for a fee, hence the erasure.) As luck would have it, the Perl installer used at CADSWES was on a non-backed-up partition that was on a disk that crashed earlier in the year. (Large files which theoretically can be re-downloaded from the Internet are sometimes kept on non-backed-up partitions, as CADSWES is always facing a space crunch on the backed-up partitions.) Fortunately, combing though several hundred gigabytes of disk uncovered a copy of the Perl installer, and installing the earlier version of Perl allowed the regression test scripts to run.

So finally in June the regression tests ran successfully, and it was possible to say that there was a 64-bit version of RiverWare on Windows.

In July, the effort returned to creating a development environment on Solaris that incorporated the new libraries. Compiling RiverWare against Qt 4.6.3 immediately generated STL-related compilation errors. Qt was configured with the --stl option, which provides STL support for STL compatible iterators and convenience methods such as QString::toStdString(). The configuration script, which itself takes several hours to run, compiles code snippets to determine the compiler's support for certain features. The compilation of an STL code snippet failed and the configuration script concluded that the Solaris compiler's default STL implementation wasn't adequate and disabled STL support in Qt. Without STL support, compiling RiverWare produced many compilation errors although most, if not all, would have been easy to fix. (One example - because a Qt header file no longer included <vector> a RiverWare source file might have to include <vector>.)

Researching the problem revealed that the Solaris compiler included a second STL implementation option - STLport. Changing Qt's make specification from solaris-cc to solaris-cc-stlport (followed by another day-long configuration and compilation) allowed Qt to be compiled with STL support.

RiverWare was then compiled against the STLport version of Qt, but crashed on startup (before executing any RiverWare code). The stack trace showed the crash was in STL code. This issue wasn't investigated any further in July.

Back on 64-bit Windows, the homegrown Perl scripts for overnight builds (overnight.pl, cvs-update.pl and devenv.pl) weren't capturing output correctly, and consequently they sent email with empty bodies. This was traced to a difference between Windows XP and Windows 7, which was resolved with code changes to the scripts. The scripts were also modified to copy the RiverWare executables to the win32 and win64 directories on animas, so that the 64-bit executables didn't overwrite the 32-bit executables.

The new environment with the new libraries (Qt 4.6.3, Qwt 5.2.1, Tcl 8.5.1, etc) was then compiled on 32-bit Windows and RiverWare was successfully built against it. At the end of July the new environment and RiverWare were compiled and running on 32-bit and 64-bit Windows, but compiled but not running on Solaris. Riverware 5.3 on 64-bit was tested by completing the three training class tutorials. No major problems were found, but three non 64-bit related bugs were filed (4927, 4932, and 4936).

In August, the issue of RiverWare on Solaris crashing on startup with an error in STL code was addressed. An educated guess was that a third party library was compiled with the Solaris compiler's default STL implementation, and the conflict between this and Qt's compilation against the STLport STL implementation was causing the crash. Qt 4.6.3 was then configured and compiled with the default STL implementation, but as mentioned last month the Qt configuration script disables STL support when using the default STL implementation. Without STL support, compiling RiverWare required many code changes, although most were straightforward such as adding a #include or coding around an STL-related method such as QString::toStdString(). After these RiverWare code changes there was finally a functioning Solaris version of RiverWare based on Qt 4.6.3.

At this point there were three distinct RiverWare development areas with three sets of code changes:

- Windows 32-bit with the code and project file changes necessary for the new versions of Qt, Qwt, Tcl, CPLEX, ClientServer and qtservice, as well as the new Reprise library.

- Windows 64-bit with the same changes as for Windows 32-bit, as well as the code chnges necessary for 64-bit.

- Solaris with the code and makefile changes necessary for the new version of Qt.

Maintaining the three areas was becoming time consuming and many of the changes overlapped, so changes made in one area frequently had to be made in other areas. Nothing was checked in, so CVS couldn't be used to synchronize the three areas, and nothing could be checked in until there was a common code base that compiled and ran on all three platforms.

The next step was to synchronize the areas. First Solaris had to be brought up to date with Qwt, Tcl, CPLEX, ClientServer and qtservice. Fortunately compiling these libraries is straightforward, and this was a relatively simple task. Next a three-way merge was done, guaranteeing a common code base on all three platforms. The code was compiled and regression tests were run on all three platforms. Finally a common code base with all of the changes could be checked in.

With the check in, the 32-bit server (scuba) and the CADSWES development machines were transitioned to the new development environment. As expected with a transition of this magnitude, miscellaneous problems were encountered that required several days to iron out. With this step, the tasks of creating 64-bit RiverWare on Windows and migrating the development environment to use new versions of third-party libraries was completed.

Build Tools

To force a relink during an incremental build, a clean-exec target was added to the Solaris makefiles. Due to the relink, the timestamp on the overnight build will now reflect if the build was successful.

The following fixes and enhancements were made to the perl script used to run the RiverWare suite of regression tests (rw-rt.pl):

- A "Y2010" bug was fixed that prevented saved regression tests run in 2010 from appearing in the --view option. This option presents a list of saved test results by date for the user to choose from.
- The perl script was modified to correctly collect system and user times in Windows batch mode (the --run option).
- Filtering of optimization diagnostic messages was modified to permit valid comparisons between optimization models.

The IBM Rational Purify and Quantify tools provide memory and performance profiling for an executable and are used to analyze RiverWare. Because of conflicts between these tools and the FlexLm library used by RiverWare, a special RiverWare executable must be created for use with these tools. The process for building Purify and Quantify executables on Windows has been simplified. A line in the riverware.pro file is now uncommented and the qmakeRational.pl perl script is run to rebuild two Visual Studio project files. Rebuilding the riverware project will then produce executables compatible with the Rational Purify and Quantify tools.

Active Directory

The impact of using the CU Active Directory network file system on the software development environment was tested. Active Directory will be used by CADSWES to provide access to shared files, especially documents from the Windows operating system. Using Active Directory creates a separate AD account on the user's machine. The impacts of this on development are as follows:

- To access the existing user's development areas, the AD account needs to be given full access to the c:\riverware and c:\temp directories.
- Customizations of Visual Studio 2008 need to be transferred to the new user.
- Licensing for the IBM Rational PurifyPlus software needs to be reapplied to include the new user.

Software Maintenance

The Software Maintenance category includes work required to maintain the RiverWare code that is apart from work funded to implement specific new features and new functionality. This can include, for example, removing obsolete code, porting code to use newer versions of libraries (like Qt) or to use alternate libraries, addressing compiler warnings and memory leaks, or changing existing code to improve maintainability and efficiency.

Replacement of RogueWave C++ Library

For many years, RiverWare has used a third party software library created by RogueWave Software. This is a large C++ library, but RiverWare used only a few of its classes, primarily RWCString. The RWCString class is designed to represent text strings and provides a rich set of string processing functionality. CADSWES staff decided to eliminate use of this library for several reasons. First, to facilitate future development and software maintenance, it is best to have the entire code base use a single string class. The interface code uses a different string class, QString from the Qt library, introducing complexity and inefficiency converting between the two representations. Second, licensing the RogueWave library for 64-Bit Windows operating systems is prohibitively expensive. Third, RWCString does not provide good support for use of multiple languages in the user interface (internationalization), whereas QString does. For these and other reasons, the decision was made to discontinue RiverWare's use of the RogueWave library, and instead to meet those needs using Qt functionality.

Replacing the RogueWave library was a large and challenging task because string processing affects most areas of RiverWare. The primary data structure being replaced was used in tens of thousands of locations scattered throughout RiverWare's approximately 1.5 million lines of code. CADSWES staff first created a detailed plan for the transition away from RogueWave (see /projects/riverware/doc/simlib/RogueWaveReplacement.fm), and then carried out this plan in several phases. Most of the numerous changes were necessarily made using automated software development tools.

After extensive testing to verify correctness of RiverWare behavior after the changes, the final phase consisted of performance analysis. The goal of this analysis was to identify and mitigate any performance problems introduced by the RogueWave replacement process. Some impacts on total process virtual memory size and working memory size were anticipated because the new string data structure uses a unicode text representation. While the unicode representation provides better support for internationalization, it also requires more memory than the previous 8 bit ascii representation. In addition, there was now the possibility that in some instances the software was converting between unicode representations and 8 bit representations, and this unnecessary processing could have a direct impact on run time. The performance analysis was conducted by applying several tools to many models, including models of the Colorado, Rio Grande, Arkansas, San Juan, and Truckee river basins.

None of the models analyzed showed a significant increase in the process total working memory size or the total working memory size, probably because memory usage is typically dominated by floating point values rather than text strings. The model load and run times for many models were in fact slower than before the RogueWave replacement, in some cases by a significant amount. Performance bottlenecks were identified and addressed by software changes. After this phase was completed, all of the analyzed models were as fast as or were faster than they were before the replacement of the RogueWave library. The following table presents results for some of the larger models whose run times improved. Note that many additional models were analyzed, but they are not reported here either because their run times did not change or were so short that (given the precision of the run time measurements) no meaningful quantitative result could be obtained.

Model	Change in Run time
CRSS Shortage DEIS	-8%
Truckee	-17%
URGWOM Planning	-16%
Tarrant Water District	-16%

The task to replace the RogueWare C++ library was wrapped up in June, and a finalized document describing the work can be found in:
/projects/riverware/doc/simlib/RogueWaveReplacement.Summary.pdf

Qt4 Porting

The RiverWare 5.2 source code was analyzed with respect to the work required to complete the port of RiverWare code from Qt3 to Qt4. (Although RiverWare is now built with Qt4, it makes use of the Qt3 Compatibility Library to integrate the remaining Qt3 components). The major tasks required to complete this port were defined, and various enumerations of Qt3 components were identified. Instances and subclasses of Q3Table and Q3ListView widgets were enumerated as a representative measure of the more difficult application-level modules needing attention in this port. SEE:
http://cadswes2.colorado.edu/~philw/2010/Qt4Port/PortAssessment5p2.html

Major sections:
(1) Major High Level Tasks in the Qt4 Port
(1.1) Converting all "Qt Designer" built UI files from Qt3 to Qt4.
(1.2) Replacing all uses of Qt3 classes (widgets, etc.) with Qt4 classes.
(1.3) Removing all uses of Qt3-compatibility methods in Qt4 classes.
(2) Some RiverWare 5.2 Enumerations
(2.1) RiverWare Qt3 and Qt4 Designer UI Files
(2.2) Dialogs in Qt3 Designer UI Files
(2.3) Dialogs in Qt4 Designer UI Files
(2.4) Dialogs defined in C++ code (not UI files).
(2.5) RiverWare uses of Q3Table
(2.6) RiverWare uses of Q3ListView
(2.7) RiverWare uses of QTableView / QTableWidget (Qt4 Tables)
(2.8) RiverWare uses of QTreeView / QTreeWidget (Qt4 Lists)

After the Qt4 Port Completion Analysis, some work was done to port Qt3 components in RiverWare to Qt4. This included:

- Automated substitution of "safe" widget type replacements in most of RiverWare. (The original Qt4 porting tools used in the preparation of RiverWare 5.0 were overly conservative). This included substitution of the following Qt3 widgets to their Qt4 counterparts: Q3Frame, Q3PopupMenu, Q3BoxLayout, Q3VBoxLayout, Q3HBoxLayout and Q3GridLayout.
- Translation of Qt3-style includes with Qt4-style includes in most RiverWare modules.
- Porting of ten Q3MainWindows (dialog widgets) to Qt4 QMainWindows.
- Proper integration of several Qt4 UI files into the standard make process.
- Writing of a "RiverWare / Qt3 Compatibility Widget Porting Guide" for three types of Qt3 widgets, with quite a bit of detail especially for Q3ListViews. SEE:
  http://cadswes2.colorado.edu/~philw/2010/Qt4Port/guide

The Qt3 UI files in the following RiverWare libraries were migrated to Qt4 UI files. UI files define the widget layouts for dialogs, and are edited with the Qt "Designer" program.

- QtAccounting (7 modules)
- QtUtils (1 module)
- QtRun (8 modules)
- QtSCT (4 modules)
- Q3GUI Plotting Dialogs (19 modules)
- Q3GUI, Remaining Dialogs (15 modules)
- QtRpl (8 modules)
- QtDmi (4 modules)

The remaining nineteen Q3MainWindows in RiverWare were ported to Qt4 QMainWindows. This porting required some code changes for the different treatment of toolbars and dockable child windows in Qt4. See: http://cadswes2.colorado.edu/~philw/2010/Qt4Port/Q3MainWindow/

RiverWare Q3ListViews were then ported to Qt4 QTreeWidgets in the QtAccounting Library (11 dialogs).

Graphical lists and "treeviews" are the most numerous substantial components in RiverWare still needing to be ported from Qt3 to Qt4. Although Qt4's Item/Model architecture for lists and trees provides the most flexible and powerful solution, the added complexity of that architecture is not warranted for all instances of lists and treeviews in RiverWare.

An analysis of the alternative approaches to porting Qt3 GUI lists and treeviews was done in June, and applied to the remaining Q3ListViews in these three areas:

* QtAccounting library dialogs
* The Configure Slots dialog
* The Snapshot Manager dialog

Some general utilities to allow the simpler item-based widgets (Qt4 QTreeWidget) to satisfy the requirements for certain lists in RiverWare were developed, including low-level support for item-specific context menus and multiple-column value sorting.

An outline for use of Qt4's Item/Model architecture for porting lists and trees was documented and was incorporated into this RiverWare Qt4 Port Completion progress report:
http://cadswes2.colorado.edu/~philw/2010/Qt4Port/PortAssessment5p3.html

RplListView was also separately ported from the Qt 3 support layer to Qt 4 proper. RPL set and group editors display a treeview of display RPL groups, blocks, and functions. Some desired new functionality could not be implemented with the Qt3 support layer, so this code was rewritten to use Qt 4 proper. The old behavior was replicated and some new functionality was added as follows:

- RPL statements can now be optionally shown in the treeview. Not all the behavior that is supported for other types of treeview items are supported for statements, but much of it is.
- Drag and drop within a RPL set (which could be across dialogs) moves the RPL object being dragged, but if an object is dropped in a dialog which is associated with a set that is not its own, then the object is copied (i.e., not automatically removed from its original location).
- The Edit->Append operation is more widely available than it formerly was.

Code Maintenance

Old Expression Slots were removed. RiverWare no longer supports the old expression slots, which were based on the constraint language. RPL based expression slots now provide expression slot functionality. The constraint language based expression slots were deprecated in 5.1 and are no longer supported at all. When a model containing old expression slots is now loaded, a warning is issued and those slots are otherwise ignored.

Code for the old (constraint language-based) optimization system was removed. The old optimization controller has not been available for several releases, but the code supporting the old optimization system was not removed because RiverWare still provided limited support for old expression slots. With old expression slots now gone, most of the code supporting the old software system and constraint language could be removed. Approximately 15,000 lines of code in 87 files were removed. File and class names were also changed to reflect the new, simpler software architecture. Also as part of this work, several unused parameters were removed from the controller-specific parameter dialog for the optimization controller.

Code for the Open slot dialog code was reworked. In the last several years, partly as a result of the transition to Qt 4, the set of C++ classes which provide the user interface to slots has changed significantly. The resulting software architecture was difficult to maintain, leading to bugs and inefficient new development. Therefore, this code was significantly revamped with the following changes:

- Obsolete code was removed from SlotQtDlg.
- SlotQtDlg code to support RPL expressions was factored out so that it could be used by the scalar slot dialog as well.
- The scalar slot dialog (ScalarSlotDlg) was extended to support ScalarSlots computed by RPL expression.
- Removed RplExprSlotDlg which had originally displayed SeriesSlots and ScalarSlots expression slots and more recently was only used for ScalarSlots with expressions.
- Removed the abstract base class SlotQtDlgBase. This was the parent class to SlotQtDlg and RplExprSlotDlg and was necessary only because we chose to have two SeriesSlot-like dialog implementations, which is no longer the case.

Five classes in the Utils directory that related to dates were no longer being used. These were removed from the current RiverWare code base. Also a compilation warning in Reservoir.cpp indicated a coding error. The incorrectly defined pointer was fixed.

The C++ class RplListView was removed from the source code because it is not longer used. It has been replaced by RplObjView, RplObjModel, and RplObjMimeData. In addition, some unused code was removed from QtMultiRunEditDlg. Prior to RiverWare 5.2 this code had supported the drag/drop paradigm for adding rules to the iterative MRM tab, but this functionality is no longer supported.

Download, Installation, and Release Processes

InstallShield by Acresso is the software used to create RiverWare installation files. InstallShield 12 was the version used to generate releases over the last fiscal year. The following modifications for InstallShield 12 were made this year:

- For the 5.2 RiverWare release, new release files were included into the InstallShield 12 project file. The release script, rw-linkbin, was also updated to include the new release files.

- Setup for the RIVERWARE_HOME_<ver#> environment variable was removed from the Environment Variables under System Configuration because this variable is no longer required by RiverWare.

InstallShield 2010, a newer version of the software, was evaluated and set up over the course of the year. This version will be used for the next RiverWare release. The work related to InstallShield 2010 was as follows:

- The "Remove" radio button was deleted from the RiverWare Setup Maintenance Program dialog during uninstallation since the button had no effect on the uninstalling of RiverWare.
- Time was spent figuring out the new license model that was implemented in the IS 2010 release. The IS license is now a single-node license, meaning we cannot install IS on as many machines as we had before. Rehosting of the license is allowed only four times per year.
- Cleaned up all the old release disk image directories and files carried over from the previous project file, reorganized the components and the data and files associated with each component, and reorganized the path variables used for finding the project files and release files.
- Consolidated strings in the project files by removing duplicated and unused strings.
- Added and edited a billboard image that will be displayed during the install process.
- Performed initial testing of the install file generated by InstallShield 2010 and fixed problems with the data and files associated with the components "main" and "license" not building into the release disk image, two icons and a few release files that did not get removed cleanly after the uninstall, and sysadm privileges being required for installation even when it is set to NO in the project file.
- Updated all the Windows download and install instruction files, making them even shorter and more concise.
- Added RiverWare support information to the dialog that cancels the installation and the dialog that exits the installation because of an error.
- Created a snapshot project file based on the final version of the release project file and made changes to the snapshot project file for working with the development builds area.
- Worked with InstallShield support on the issue of forcing users to enter the user name and the organization entries in the Customer Information dialog before the Next button is enabled for continuing to the next step. The proposed solution will cause confusion for our users because the Next button will not be automatically enabled until user clicks on any control event to refresh the Next button. Since this issue is bound by MSI limitations when the project type is a Basic MSI project, we will not force users to enter the Customer Information in the dialog.
- Created a new dialog to display the instructions for setting the Acroread environment variable. This will be available in the final install complete dialog and only will be displayed when user chooses to click on it. In the current install file the popup dialog is always displayed and is an extra dialog for user to click to close.
- Created different project files for releasing 32-bit and 64-bit versions of Windows releases. Restructured the Windows release directory tree in all three build areas (builds, prerel, and release) so each has two separated directories, one for holding the Windows 32-bit release files and one for holding the Windows 64-bit release files.
- Created a file association for RiverWare model files in the install program and added a new RiverWare model file icon to the project files. After the new RiverWare release 6.0 is installed on

a system, a file with the .mdl extension will be displayed with the new RiverWare model file icon on the desktop and the extension will be associated with the latest installed RiverWare.

- Work was started to add a radio button group in the CustomerInformation dialog, which will enable end users to specify whether to install RiverWare for "all users" (per machine) or "Only for me" (per user). This setup configuration will have different effects on Windows 7, Vista, and Windows XP.
- In the Solaris/Windows release configuration files and the IS project files, from both builds and prerel areas, removed cplex.tar, goals.tar, and cplex110.dll and added cplex122.dll. Updated holding directories for the release files and updated the rw-linkbin tool with the changes.

The procedure document for generating a Windows release version of RiverWare using all versions of InstallShield was updated. Instructions were added for setting up the Samba share "isadmin" for accessing all the IS project files on animas. New steps were added for building and releasing Windows Remote executables and new regression test information was included.

Licensing

Ongoing through the year was the work of issuing Commercial Licenses, including creating and shipping of Solaris and Windows versions of licensee, sponsor, evaluation, temporary, and viewer licenses. Specific tasks included tracking expiring licenses and contacting users to get information; generating license files; updating the license data records; providing instructions for RiverWare download, install, and floating license manager service configuration setup; and problem-solving for users. There was also ongoing maintenance of licenses for the CADSWES internal development environment.

A final report on FLEXID dongle testing from last fiscal year was generated. Setup instructions were written for FLEXID dongle license users. Procedures for the required driver download/installation and how to generate a dongle license file were developed. Information on the purchase of dongle hardware keys from Flexera Software was documented.

The Flexnet licensing 11.8 utility programs lmgrd and lmtools were downloaded and tested. The two programs were upgraded into the RiverWare releases. These two utility programs were backward compatible with previous RiverWare releases and did not require code changes.

A new option was provided to users for specifying the Windows license file path location: setting an environment variable CADSWES_LICENSE_FILE. The variable can be set as a system variable or a user variable. When the variable is set, user does not get the error "Error in saving configuration settings: You are not a license administrator."

The following updates to licensing instruction files were made over the course of the fiscal year:

- Based on the fixes from bug 4750, all license installation instructions files were updated to include information on how RiverWare stores the Flexlm license path in the registry, and instruct users how to start RiverWare as an administrator initially to set the license path in the registry on the machine.
- Dongle installation instructions were added to the online license guide document.
- Created the license renewal setup instruction file for the 3-Server Redundancy Floating license type. Tested the procedure on two different sets of 3-machine setups.

- Updated the Solaris Floating license server manager setup instructions for how to access a Solaris license server manager from a Windows machine as well as a Solaris machine. Also added information about setting the file descriptors in a ksh shell before starting the license server manager.

- Updated the Windows Floating license server manager setup instructions by adding instructions for how to reset the license path registry key on each RiverWare machine when the license server manager is moved to run on a different machine.

- Updated all of the Windows installation instructions to instruct the user how to set up the environment for RiverWare to find the license file (or the license server machine for a floating license) based on the level of system administrative privileges of the user.

- Finished the instruction guide for the Windows version of how to set the TCP port number in a floating license file.

- Updated the license generation instruction file.

The following bullets summarize research that was conducted into alternatives for replacing the FLEXnet Publisher software that is used for RiverWare's license management:

- No suitable freeware licensing software could be found.

- RLM by Reprise Software, annual fee starts at $3,000.

- EasyLicenser by Agilis Software, annual fee starts at $5,000.

- SoftwareShield by Yummy Interactive, pricing is available online, but it does not support Solaris.

- LM-X by X-Formation. A Denmark company.

- Quick License Manager from Interactive Studios, pricing is available online, but it does not support Solaris.

Based on the results, an evaluation copy of Reprise RLM software was obtained, and “proof-of-concept” incorporation of the software into RiverWare was tested. With successful testing, a license for RLM was purchased. RLM Version8 for all three platforms was downloaded and installed on animas. Private and public key files were generated and were identical among the three platforms. The three versions were configured and built (Solaris on animas, Windows 32-bit on scuba, and Windows-64 bit on spurwink). The RLM directory tree was structured for the development environment, release environment, and license generation environment, and the required release files and license generation tools were set up on the three platforms. Code for the incorporation of RLM into RiverWare was formalized, include error handling code.

With the next major release of RiverWare, all new licenses going forward on Windows will be generated with the RLM software, although older licenses generated with FLEXlm will continue to work with RiverWare. A new line called “License Manager” was added to the license file format to indicate which license manager (FLEXlm or RLM) was used to issue the license file. Also the version number of the license file was changed from 99.0 to 100.0 going forward to differentiate a FLEXlm license file from a RLM license file.

Regression Tests

Regression tests are a suite of RiverWare models that were automatically run every night to look for changes in results or performance from new code that was checked in to the development area. Results of the tests were evaluated on a daily basis to identify unexpected changes in model results. The model comparisons performed in the nightly regression tests can show expected differences (for example, because a new method category may have been added). When this occurred, the regression tests had to be updated to reflect the current state of the development area so model comparisons did not fail going forward. In addition, every week, the daily history of each regression test was analyzed to determine if the run time or model size had significantly changed because of new development.

In addition to the regular regression test work, the following regression test items were addressed over the course of the year:

- The regression test scripts were modified so that the optimization test tvaOptRPL would succeed on windows. Previously, the diagnostics comparison failed because of some slightly different optimal values being posted. Now the diagnostics are filtered and rounded to the appropriate significant figure.

- The concurrentMRM regression test failed on Solaris. This test was developed on Windows and required some changes to its Perl script to run correctly on Solaris.

- Some of the daily regression tests were observed to have slowed down by a few seconds. The code base and regression tests were analyzed to determine if the slowdown was due to changes made to RiverWare. It was concluded that the change was in fact due to some external factor, possibly a new process on the host machine.

- Three new tests were added that use the optimization controller: oneResOpt, unitPowerOpt, and an hourly optimization model.

- Different optimization parameter files were implemented leading to alternative optima on windows optimization regression tests. Optimization parameters were updated into the optimization models.

Bug Fixes

The following table summarizes RiverWare bugs that were closed in FY 2010:

Number	Description
994	Enhancement: toggle display for AggSeries sim col/all col.
1862	Dispatch Info dialog should show pre-sim disp of reaches
1942	Table Series Slots do not refresh
2249	No menu or accelerators for Slots/Methods in DispInfo diag
2250	Slot flags menu & accelerators don't scroll to next val
2403	RPL function evaluation need to detect infinite recursion
2445	TableSlot doesn't issue callback when column data is imported.
2479	Obsolete slots in thermal object
2575	Old Power Lambda Coefficient Tables are not dealt with correctly

3114	RiverWare doesn't ANY output slots in MRM if any are incorrectly specified
3387	Managing plot labels in different dialogs
3387	Various Plot Dialog issues
3441	Graphs take a long time to redraw after size modification
3463	Import DMI units and set_units keywords do not recognize HOUR
3464	Use of plotting tool results in computer freeze
3507	DMI doesn't retain ordering of user parameters
3510	Font size is not user configurable in RiverWare dialogs
3538	RiverWare license failure when running batch mode while not logged on to XP workstation
3635	RiverWare crashes when running after opening and closing the online help this is on Windows with the new Adobe Acrobat
3638	Link plots doesn't link some configuration/editing functionality
3715	Periodic slot deleteLastColumn() has bad code
3726	riverware --model <4.6 model file> fails
3728	Can't select PeriodicSlots for plotting
3745	Its difficult to select DMIs in the new MRM dialog
3755	Allow the Workspace to be raised above the Locator View
3762	Batch mode is case sensitive even on Windows, but doesn't really need to be
3767	Editing and MRM configuration take a LONG time to bring up the Edit dialog. And there is no hourglass
3802	Should not be able to create links to/from SnapShot Obj Slots
3925	Open TableSlot needs “Insert Column” for DataObj TableSlots
3945	Open ListSlot needs standard open slot features; Crash fixes
3987	MRM dialogs accept text input one character at a time
4049	Excel writing from MRM has memory issues with large output
4059	MRM GUI initial offset field on input tab requires <enter>
4080	Plot X and Y Grid Settings not saved.
4101	Open Obj Dlg: Chg AggObj Linking Struct invalidates Accts.
4136	diagnostics for DMI input
4210	Problem with regular intervals on periodic slots
4259	New slots are placed out of order when in ascending view.
4277	Post-simulation accounting controller is not executing a simulation run first
4284	Direct from RiverWare Excel generation doesn't work with large MRM runs
4292	Core dump in MRM run control
4294	Edit Acct Dlg: Change “Configure.” to “Open Account.”
4299	Subbasin Disabled state is not retained after file save.
4318	DMI cannot be imported into model
4323	Database DMI name maps should allow empty alternative names
4349	Individual data values missing using DMI
4367	Multiple GUI fields appear to not be editable
4395	No confirmation when deleting an aggregate element
4414	Mismatch in linked slot step sizes is reported better now
4415	Automatic time series adjustment needed for SubSlots linked to MultiSlots
4455	Problem ith synching two linked MultiSlots
4465	GPAT graphs not automatically formatted for some types
4488	Model Run Analysis, hot keys Ctrl + D, E.
4518	Configure>Precision does nothing for Scalar Slots
4544	Open Slot and Edit Acct Dlg Row Headers cut off on Windows.
4549	Qt4 Display Group Editor - Legend graphics and other problems
4562	Automatic time series adjustment needed for SubSlots linked to MultiSlots
4616	Editing aggregate element name - Shouldn't have to hit enter
4640	Cannot remove a dataset in the new Database DMI editor
4653	remove dataset feature does not work
4659	Incorrect error dialog from output configuration dialog

4676	Boolean optimization parameters not read by CPLEX
4697	TableSeriesSlot not setting column labels correctly
4699	TableSeriesSlot start date support
4706	Optimization parameter files can not be overwritten
4712	Open Object Dialog - Slot Type sorting is non-deterministic.
4744	Output Config doesn't use the individual Snapshot selections.
4746	Problem with writing optimization log files
4753	SeriesSlot “insert before” w/ multiple selection
4755	This bug has allowed users to create inflow/outflow type supplies between accounts within the same reservoir; only transfer type of supplies should be allowed in this situation.
4758	Bugs with enable/disable of RPL list items, RPL statements
4759	Model Run Analysis: Rules Details not re-opening correctly
4760	Problem with units/labeling in statistical table slot
4773	Run is slow with Edit Account dialogs open
4779	direct Riverware-HDB DMI model_run_id selection fails
4780	configuration of direct Riverware-HDB DMI dataset fails
4782	Duplicate priority date via “Priority Right” method
4783	Auto-scroll workspace when dragging over edge.
4787	A new instream flow account was not solving until the model was saved and reloaded.
4788	Acct summary not showing correct flag colors
4789	The units on the Storage slot added to a passthrough account were defaulting to standard units instead of the unit preferences identified in the Account System Configuration dialog.
4795	SeriesSlot Notes column gives incorrect summary info
4796	Several problems with new expression slot dialog
4797	Several Integer Indexed Slots problems
4799	ABORT message dialog window unselectable after multiple aborts
4800	The RPL predefined function IsInput was failing erroneously
4801	Problems with RPL set analysis dialog
4802	Find Inputs function does not find inputs in AggSeries cols
4806	RPL predefined function RunStartDate not working properly
4807	ScalarSlots with expression don't display DateTime values correctly
4810	Output Device Configuration dialog: Clear Workspace doesn't close config dialog
4811	Empty slots in snapshot if not saving model output
4812	Crash associated with interrupted MRM runs
4813	Update problem with single run controller in MRM runs
4814	On a Reach configured to use Time Lag routing, the Local Inflow was not being set at pre-simulation timesteps when it was assumed to be zero.
4815	Flashing dialog for RPL sets embedded in a model
4816	The RPL set analysis dialog didn't support nested statements
4817	Problem with drag and drop in Release executable
4818	Problems with shortcuts to RPL expression editing actions
4819	Tabular Series Slot Out Device doesn't work in batch mode
4820	The RPL debugger “Disable debugging” action was sometimes incorrectly enabled
4821	WS link tool tips should be shown in status bar
4822	Config AggSeries Timeseries Multi-line Col Labels
4823	Problems with selection/clicking in RPL tree views
4824	The unregulated spill near the crest was incorrect. This led to cases where the pool was above the crest but there was no unregulated spill.
4826	The RuleSet manager wasn't managing auto-generated post-optimization rulesets
4827	Output Manager Plot Page - Using red X doesn't close, Crashes
4828	Unit converter sometimes hides significant digit.
4829	annual RDF output doesn't include first year
4829	OpenObjDlg: Keyboard shortcuts work only after opening menu
4830	Editing slots, TAB moves to next, but keeps 1st cell selected

4831	Creating a computational subbasin is awkward
4833	Inline editing of slot values hides significant digits.
4834	The return flow slot on diversion account was not being allocated a data series on model file load if it was not the outflow slot when the model file was saved.
4835	Periodic Slot Export Copy - Row header text has invalid time
4836	Accounting diagnostic user method messages not printing correctly
4837	Ghost Dots in Qwt Plots w/long series using Ellipse Symbols.
4840	Link Dsp Groups: Can't revert 2-Slot Set definition to 1-Slot
4842	Core dump in solaris when writing CPLEX problems
4842	Core dump logging optimization run
4843	Fixed: ListSlot Dialog no longer has OK/Apply/Reset/Cancel Buttons
4844	Runtime error in optimization with TVA's hourly model
4846	Usability / Confusion Problem with Open Scalar Slot Dialog
4847	Accounting period consistency in MRM
4848	Saving RBS set in model file when it uses a global function set
4849	Problem with expression slot sync with run behavior
4850	Crash when creating HDB model run ID from HDB database DMI dialog
4851	Fractional scales over 1.0 were sometimes displayed without the fraction part
4853	RPL list equality not working
4854	Can't add an inline comment to the left-hand side of a RPL assign statement
4858	Enabling RPL debugging slows down runs inordinately
4858	RPL Debugger not always raising when it should
4859	Exp Slot Dlg: Paste Note ctx op is enabled, but doesn't work
4864	Double-click on slot value changes it to an input
4865	Accounting Method Rpl Set dialog problems
4866	False warning about convolved stage flow tailwater table
4867	Loading model gives error about unexpected OLAM group
4868	RPL function evaluation need to detect infinite recursion
4870	File export of a statistical table containing dates would not show dates properly in the exported file.
4872	Excel worksheet label length in Database DMI
4873	crash after deleting slots from dataset
4875	SCT Export Copy - Add option to include column headers
4876	After renaming Database DMI, DMI menu crashes
4877	Database DMI allows restricted slots to be set during RBS
4878	RplBlock input Database DMI fails with “setting user input” error
4880	Simulation view icons off canvas
4881	RPL debugger not pausing before some errors
4882	Problem with loss of SubSlot values saved in model file
4883	Failure to load RPL set with disabled rule
4884	RPL debugger doesn't always pause before error
4885	Incorrect multiplication in GainLoss
4886	Can't edit RPL function constraints
4888	Optimization unit power problem
4888	Unit power internal error starting in version 5.2+ but not 5.1.6
4895	Windows access to solaris hides optimization policy
4896	Minor issues with substitution approximation in optimization
4897	Some RPL functions need beginning of run behavior
4898	Crash after dismissing DMI Parameter Dialog from Window Title Bar
4899	Can't change a non-NaN scalar/table slot date value to NaN
4900	Minor GUI problem in RPL set editor
4902	Diversion account solve at run init
4903	RplUnits file not always being found
4904	Plotting NaNs in series
4904	In a slot with mixed real values and NaNs, NaNs are plotted as arbitrary constant values.

4906/4929	Adding element to an Agg doesn't display correctly
4908	Crash adding a statement to a rule which has a breakpoint
4909	Accounting period does not span run range and leads to problems
4910	Problems with RPL function execution constraints
4911	Spline plots don't go through data points
4913	Problems with RPL function execution constraints
4914	Subbasins can't contain aggregate objects without elements
4916	Problem involving the RPL Set editor's selection description panel
4917	Crash renaming AggObj having an Element / RWCString repl
4919	Problem with RPL debugger and RPL function caching
4920	Problem with RPL unit display
4921	New Excel format files (.xlsx) cannot be generated from the Excel Output Device in the Output Manager.
4922	SCT, adding divider is awkward.
4923	The Excel Database DMI could not locate dates in an Excel worksheet if the dates were entered as formulas.
4926	Rule executing over and over again
4927	Qt 4.6 Rules model run anal dlg cell drawing color probs
4928	Discrepancies between output manager and MRM RDF files
4930	Import Paste operation not reading numbers with commas
4931	Problem with RPL's use of the resource database (riverwareDb file)
4932	Core dump adding a function/rule to a group
4933	“Step” button should say “Init” when not in run.
4934	Output Conf dlg not adding Table Slots correctly
4935	Incorrect Plotting of Multislot Subslots / Supplies
4936	Needed to remove diagnostic output / Exch Payback Source
4937	An optimization run-time warning was missing context information
4938	Partial Re-enabling Priority Rights results in non-unique P-dates
4938	Need enforcement of unique water rights accounting priority dates
4939	Open Obj Dlg: Slot display info discarded when dlg closed.
4940	Problem deleting some RPL sets on workspace clear
4941	Inappropriately presenting users with a confirmation dialog
4942	Problem with Optimization's handling of ineffective constraints
4943	Problem in SolveWaterRights accounting RPL predefined function.
4946	Qt debug assertion failure when loading a model in debug.
4947	Problem with mapping reservoir elevation to storage and area
4948	A slope power reservoir's flashboards were failing at the incorrect timestep.
4949	Core dump in several RPL predefined functions
4950	The max capacity computation on a slope power reservoir was not converging.
4951	Update issue in RPL editors “Show” checkboxes
4952	Init. Rules breakpoints were being lost on model load
4953	Typing numbers in slots is behaving incorrectly.
4954	Core dump closing RPL set (as opposed to the set's dialog).
4958	Adding a curve to a plot does not use the color shown in the config dialog
4959	GetMaxOutflowGivenInflow was using the wrong Tailwater elevation in certain circumstances.
4960	Problem with RPL debugger
4961	Control point routing coefficients are not adding columns when upstream reservoirs are added
4962	Hypothetical simulation was not converging
4965	Water Rights Accounting performance problem