

Phil Weinstein / Accomplishments -- March 2015 -- Edit 4-02-2015**General Development Accomplishments**-----
[I.A] New/Enhanced Software / BOR-ABQ #3Graphical Teacup and Animation for Links / "Output Canvas" Output Device

Support for Output Canvas Flow Lines for RiverWare 6.7 was substantially completed in the prior month (February 2015). In March, the following issue was addressed and the code changes were again reviewed, and merged with the 6.7 development git branch.

Object Icons on the Output Canvas were introduced in last month's Flow Lines development. These are associated with actual "live instances" of simulation objects (as are teacups). To benefit from all of the relevant ancillary features of teacups, e.g. associated dynamic text items uniformly referring to slots on the simulation object, object icons had been implemented as a *special mode* of the teacup graphical item class. Dynamic switching between a teacup and object icon appearance had been tested, but we decided against this presentation. Instead, object icons are presented as a completely different sort of output canvas object, organized in their own object icon groups. Left over from that development had been some incorrect semantics of settings, operations, and tooltips, still referring to *teacups* when in an object icon / object icon group context. Additionally, a separate "Delete All Object Icons" operation was added.

[I.A] New/Enhanced Software / BOR-ABQ #6Display RPL Predefined Function Help Content in RiverWare 6.7

RiverWare 6.7 displays RPL Predefined Function Online Help Content -- as HTML in a QWebView -- in the following three dialogs:

1. **Predefined Function Editor** dialog.
2. **RPL Palette** ... on the "Predefined Functions" tab when Descriptions are shown.
3. **RPL Set Editor** ... when "Show Predefined Groups" and "Show Descriptions" are checked.

Prior to this enhancement, panels and dialogs which normally show a user-provided description for a RPL Function, in the case of RPL *Predefined* Functions, showed either a blank panel or the message, "See Online Help for documentation of predefined functions". These panels now show an HTML version of online documentation for the particular RPL Predefined Function. Also, a Description panel was added to the RPL Palette's "Predefined Function Tab", similar to, and controlled in parallel with the visibility of the "Description" panel on the "User-Defined Functions" tab.

The Function HTML help content -- for all, approximately 200 RPL Predefined Functions -- is bound to the RiverWare executable as a Qt Resource. (A compressed copy of the full HTML page -- without images -- is part of the RiverWare executable). Referenced images are loaded at run time by RiverWare's integrated QWebKit-implemented HTML browser.

A significant part of this effort involved developing a *process* and *supporting tools* for exporting the content from the FrameMaker source document and processing that content for inclusion into the RiverWare build. This includes:

1. Development of a Python 3 script to transform display-oriented HTML exported from FrameMaker into structured HTML having content-specific meta-data.
2. Documentation of the overall process of porting help content from FrameMaker into the RiverWare build.

These two documents were prepared:

Feature Document: RPL Predefined Function Help content in RiverWare 6.7 / March 2015

R:\doc\RPL\EmbedRplDoc\2015\RplFuncHelpInRiverWare-March2015.docx

R:\doc\RPL\EmbedRplDoc\2015\RplFuncHelpInRiverWare-March2015.pdf

Processing RPL Predefined Function Help for use in the RiverWare 6.7 Build (Internal).

R:\doc\onlineHelp\process\GenRplFuncContent.html

R:\doc\onlineHelp\process\GenRplFuncContent-March2015.pdf

[I.A] New/Enhanced Software / TVA
Min/Max/Bound Semantics revisions

We are in the process of revising the presentation of slot minimum and maximum values as either "bounds" or "optimization limits for table verification" to reflect the actual, limited uses of those values in RiverWare.

Maintenance Accomplishments / March 2015

[II] RiverWare Software Maintenance / Software Updates / Bug Fixes

The following bugs were fixed:

- Bug 5607: RPL predefined function toCelsius is spelled incorrectly
- Bug 5609: Open Account dialog not updated when creating supplies on the workspace.
- Bug 5611: Tool tip on Input accounting slots is incorrect.
- Bug 5623: Running CRSS and hitting an error gives many new assertion messages.

Bug 5614 -- Annual Aggregation time series slots are not correctly summing monthly flows in units of AF/month -- is in the process of being addressed.

Most of the aspects of this bug are related to the special Time Aggregation Series Slot which users can create on Data Objects.

One high-level problem is that rate unit types (notably, FLOW) were being summed in the "normal" way, which is not correct for rate (e.g. flow) values for different timesteps within a single time series. Two options which were considered for the SUM function of Time Aggregation Series Slots were: (1) "Integrating" such values with respect to time (e.g. showing the sum of those flows as a volume), or (2) Showing the sum as the average value, but with a rate unit having a "per time" factor matching the time aggregation size (e.g. showing an annual aggregation of "per month" values with the analogous "per year" unit). We have implemented the latter approach.

Another high-level problem is that rate values needed to be summed (for both the *sum* and *average* functions) using a "weighted" sum and average. This is relevant specifically for summing rate series having an irregular timestep (i.e. per month or per year).

Ancillary to these Time Aggregation Series Slot issues, we have also implemented these related enhancements:

- Support for aggregating to Days (e.g. of hourly data). Previously, only aggregating to Months and Years was supported.
- The SCT's time aggregation capacity was expanded to support annual aggregation of daily series. That scale of aggregation (365 or 366 detail rows for each time aggregation) presented both extreme graphical and program performance problems for the SCT's time aggregation "summary" row cells. Those are normally "painted" in separate vertical slices -- one slice for each timestep -- primarily to depict each timestep's flag state as a background color. This time aggregation summary cell feature is now turned off for time aggregations of 100 or more timesteps.

In the course of testing the Time Aggregation Series Slot fixes, a broader problem was discovered within our unit conversion algorithms. **Rate values within an annual series are not correctly converted to per-month units.** The result was effectively assuming an average month-length of 31 days -- i.e. the length of the month (December) preceding the Date_Time designating the end of an annual timestep. We've determined that our variant of the "convertWithinType" C++ method which is given a Date_Time to resolve irregular time units also needs an explicit DeltaTime (symbolic time interval) to correctly compute the required value. We are addressing this problem before completing the testing of the "5614" issues specifically involving the Time Aggregation Series Slot.

To address this latter problem, an analysis of the *internal structure* and *use* of our unit conversion utility methods was done. This analysis included observations about how to improve the unit conversion API and implementation for both better run-time performance and application-code readability. A formal document for this analysis will be provided in April.

--- (end) ---