Slot Attribute Specifications Redesign

in RiverWare 6.1

Author: Phil Weinstein

This document describes a new slot attribute definition architecture supporting:

- assignment of default slot attribute values to newly created slots, and
- multiple sets of display-related slot attributes ("Unit Schemes")

0.1 Document Status

6-13-2011: Ready for review; Includes development estimate.
6-14-2011: Dropped incorporation of Display Format and Precision into ScaledUnit class.

0.2 Other Relevant Documents

- RiverWare Technical Documentation ("online help") -- "Units" Section.
  The RiverWare 6.0 edition (11-22-2010) is available at this URL:
  http://cadswes.colorado.edu/PDF/RiverWare/documentation/Units.pdf
- Unit Schemes in RiverWare 6.1 (Analysis Document; June 8, 2011). A PDF copy is available here:
  http://cadswes2.colorado.edu/~philw/2011/Units/Design/UnitSchemes-2011jun08.pdf

0.3 Contents

1.0 Overview

The legacy implementation of slot attribute initialization for newly created slots on physical objects makes use of the external “riverwareDB” file to conditionally override hard-coded default values. These two general types of slot attributes are supported:

Numeric display-related slot and *slot column* attributes:

- display units
- display scale (floating point value)
- display format (integer, float, or scientific notation)
- precision (integer: fractional decimal digit count)

And other numeric slot attributes:

- minimum value (floating point value)
- maximum value (floating point value)
- convergence type (none, absolute, percent, unit percent)
- convergence limit (double)

Once these attributes are set on slots and slot columns, they can be modified through GUI-based tools within RiverWare, either on a single-slot / slot-column basis, or in groups. (Support for accounting slots is a little different. This document is describing a reimplementation of the relevant part of the physical slot attribute support which would apply also to accounting system slots).

Enhancements are being planned for RiverWare 6.1 which would address the following **requirements:**

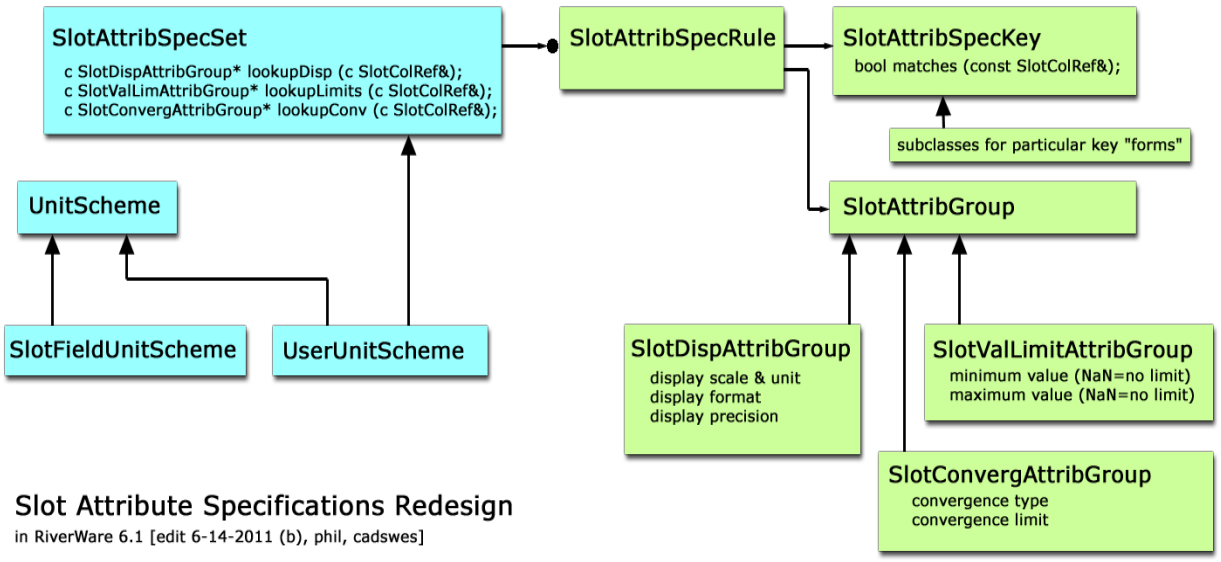
1. It is undesirable to maintain default slot attribute values in an external text file (i.e. the “riverwareDB” file). The data which had been represented in that file (i.e. mappings from “lookup keys” based on entity names and unit types to attributes and attribute values) should be supported in the RiverWare GUI, and maintained as persistent data within the RiverWare model.
2. For the numeric display-related slot attributes, it should be possible to switch between multiple sets of user-defined slot display attributes (“Unit Schemes”). It is acceptable to retain the current display configurations defined on slot and slot column instances as a *special* “Unit Scheme” among the set of multiple Unit Schemes which can be defined with the new capability.

User-Defined Unit Schemes will be defined by the user in a new dialog box as a sequence of mappings from lookup keys (e.g. object type and slot name) to display attributes (display scale and unit, display type, and display precision). While this is analogous to how slot attribute initialization values had been provided via the “riverwareDB” file, it differs from how display attributes are currently configured on slots and slot columns in that the display attributes provided to slots and slot columns from Unit Schemes are not subsequently editable via a slot’s configuration dialog. Rather, the display attributes in Unit Schemes for slots and slot columns are always edited in the context of the Unit Scheme’s sequence of mappings from lookup keys to display attributes.

(Technically, display attributes provided by Unit Schemes will be cached on slots and slot columns, at the time those attributes are needed, but that data will not be presented to, or directly modifiable by the user).

1.1 C++ Class Diagram

This diagram illustrates the relationships between the major new C++ classes described in subsequent sections.



Note: The methods depicted in the classes above are an abstraction. In fact, an additional parameter will be required for the support of “flow vs. volume” units for flow-like entities within a time series.

2.0 Slot Attribute Specification Set

A **Slot Attribute Specification Set** consists of an ordered set of mappings from **“Lookup Keys”** (similar to those used in the riverwareDB file format) to **Slot Attribute Group** of the following three types:

- **Slot Display Attribute Group:** Unit scale / Unit / Display Format / Precision
- **Slot Value Limit Attribute Group:** Minimum Value / Maximum Value
- **Slot Convergence Attribute Group:** Convergence Type / Convergence Value

A “User Unit Scheme” is an instance of a Slot Attribute Specification Set containing only Slot Display Attribute Groups.

A RiverWare model will maintain the following Slot Attribute Specification Set instances:

1. A single “Initialization Slot Attribute Specification Set” supporting all three types of Attribute Groups, but its mapping to Slot Display Attribute Groups are used for initialization of new slots *only when* no “User Unit Scheme” is current (i.e. when the Slot Field Unit Scheme is current -- see “Unit Schemes”, below).
2. Multiple “User Unit Schemes” supporting only Slot Display Attribute Groups. One of these -- or a Slot Field Unit Scheme (see below) -- will the the “current” Unit Scheme at any one time.

2.1 Slot Attribute Specification Lookup Keys

The keys used in Slot Attribute Specification Sets are modeled on the design of the "Lookup Keys" used in the riverwareDB file format. These will be extended to include:

- Support for Account Type Keys
- Support for a single Exchange Key
- Support for a "Default" Key (matching all slots, after all other mappings have been applied).
- Support for Alternate Units (which are paired with Primary Units by a time factor, e.g. Flow and Volume)

Supported keys will be of these forms:

1. <object type> <slot name> [flow / vol]*
2. <object type> <slot name> <column name> [flow / vol]*
3. <account type> <slot name> [flow / vol]*
4. <account type> <slot name> <column name> [flow / vol]*
5. <exchange> <slot name> [flow / vol]*
6. <slot name>
7. <unit type>
8. <default> *(matches all slots, after all other mappings have been applied).*

*An optional "flow vs. vol" selector will be available -- only in keys used with Slot Display Attribute Groups (see next section) -- for specifying "alt units" for entities that are inherently flows (even if their primary unit type is volume). This will allow the user to provide distinct rules for the two unit types for any "flow" slot. This option will be supported only for keys for which a prototype object or account is available (and also for exchange slots). In particular, it will not be supported for the <slot name> key (#6, above). Availability of the option will depend on whether the prototype objects' or accounts' corresponding slot has "alt units" defined -- which, in fact, in RiverWare 6.1, are currently supported only on accounts.

2.2 Slot Attribute Groups

As mentioned above, there are these three types of Slot Attribute Groups:

- **Slot Display Attribute Group:** Scaled Unit (Scale and Unit) / Display Format / Precision
- **Slot Value Limit Attribute Group:** Minimum Value / Maximum Value
- **Slot Convergence Attribute Group:** Convergence Type / Convergence Value

These groupings (aggregates) of slot attributes will simplify the handling of related attributes as a single instance. A typical (anticipated) presentation of a group's attribute's values will be as a single row of widgets.

TBD (very minor issue): The "Minimum Value" and "Maximum Value" attributes' "NaN" state will either be transparent or opaque when applying mappings to a slot or slot column lookup. That is, a "NaN" value could either imply that a "no limit" condition is applied for the slot, or that the more general (lower priority) matching rule prevails.

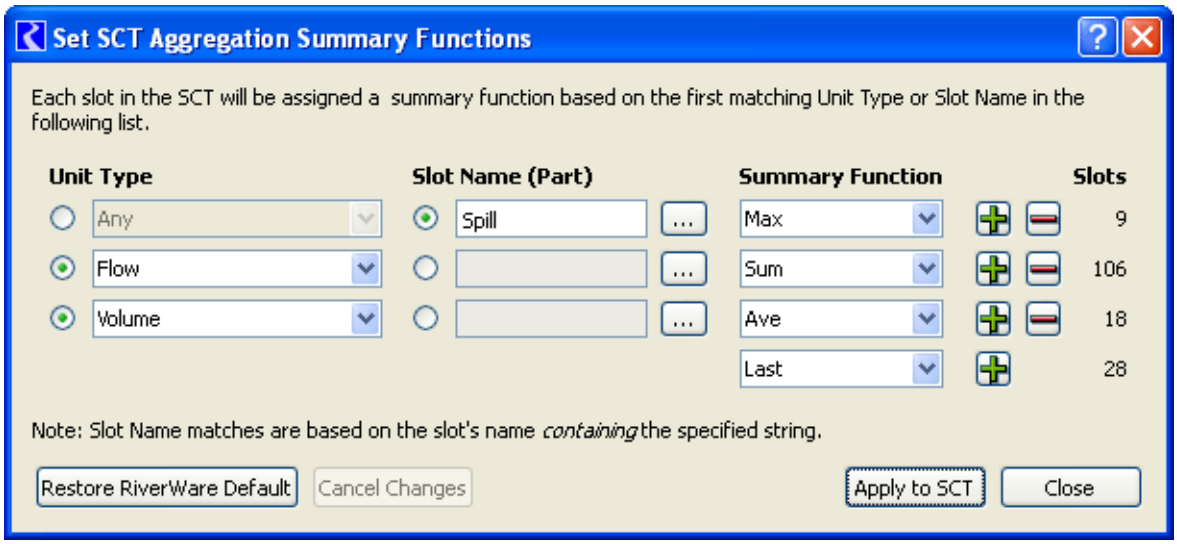
Note that the former "Max Iterations" slot property, supported by the riverwareDB file, is *no longer* set on a per-slot basis, so will not be supported in this new architecture.

2.3 Slot Attribute Specification Set Editing Support (GUI)

Any given Slot Attribute Specification Set will *conditionally* support mappings to the three different types of Slot Attribute Groups. (For example, User Unit Schemes, which are Slot Attribute Specification Sets, support only the display-related attribute group type).

Each mapping (from a Lookup Key to a Attribute Group) will be presented as a single row of widgets, and the mappings to **each of the three different types of Slot Attribute Groups** will be presented in their own conditionally-show **panel** (with up to three panels, arranged vertically).

Similar to the recently developed **"Set SCT Aggregation Summary Functions" dialog,** there will be Green Plus and Red Minus buttons on each row to allow insertion and deletion of mapping rows:



Associated with each mapping will be these types of values, displayed in one row:

- Lookup Key Type
- Lookup Key Values
- Slot Attribute Group Values

Possibly, it may be too visually complex to support in-place editing of all these values. It *may* make sense to do so only for the Slot Attribute Group Values (which will be of the same number and type in each of the three panels), and not for the Lookup Key Type and Key Values. Those could be initialized and possibly modified using a popup dialog. **(We should to look at some mockups for the various Lookup Key Type cases).**

In the panel supporting Slot Display Attribute Groups, there could be some provisions for comparing or setting RiverWare standard units in the various mappings.

3.0 Unit Schemes

A Unit Scheme resolves a Slot and Slot column reference to Slot Display Attribute Group value. This will be implemented as an abstract base class with the following concrete implementations:

- Slot Field Unit Scheme (uses the legacy slot and slot column instance-based settings)
- User Unit Scheme

3.1 Slot Field Unit Scheme

The Slot Field Unit Scheme is a “singleton”, and does not maintain any mappings. (It is not a Slot Attribute Specification Spec). This relatively trivial Unit Scheme simply retrieves Slot Display Attributes from the fields on Slots and Slot Columns.

In the GUI, the Slot Field Unit Scheme will be identified as:

- “Slot-Configured Units”

3.2 User Unit Scheme

A User Unit Scheme is a Slot Attribute Specification Spec supporting only Slot Display Attribute Groups. The user will be able to define a variable number of named User Unit Schemes, and will be able to duplicate (and modify) any existing scheme.

4.0 RiverWare Software Modifications

4.1 Modifications to Slot classes

To simplify the process of switching between the legacy slot (and slot-column) instance display attributes, those existing fields on slot classes will be “repackaged” as a Slot Display Attribute Group (SlotDispAttrGroup) instance, AND an additional “uscheme” SlotDispAttrGroup instance will be dynamically assigned on the first retrieval after the current User Unit Scheme within RiverWare is changed. This slot “uscheme” SlotDispAttrGroup will be maintained as a cached value based on a global Unit Scheme “change” serial number:

When retrieving a User Unit Scheme (“uscheme”) SlotDispAttrGroup from a slot, the slot’s _uschemeSerialNumber will be compared with a global _uschemeSerialNumber which is incremented each time a different unit scheme becomes current, or the current scheme is edited. If there is a mismatch, the slot’s “uscheme” SlotDispAttrGroup field is recomputed from the current User Unit Scheme, and its _uschemeSerialNumber is updated.

Old SeriesSlot fields:

```
ScaledUnitPtr _scaledUnit;
QString _usrFormat;
int _usrPrecision;

ScaledUnitPtr _altScaledUnit;
QString _alt_format;
int _alt_precision;
```

New SeriesSlot fields:

```
SlotDispAttrGroup _localDispAttrs;
SlotDispAttrGroup _localAltDispAttrs;

unsigned long _uschemeSerialNumber;
SlotDispAttrGroup _uschemeDispAttrs;
SlotDispAttrGroup _uschemeAltDispAttrs;
```

Old TableSlot fields:

```
cwArray<ScaledUnitPtr> *_scaledUnits;
cwArray<QString> *_usrFormat;
cwArray<int> *_usrPrecision;
```

New TableSlot fields:

```
unsigned long _uschemeSerialNumber;
QVector<SlotDispAttrGroup> _localDispAttrs;
QVector<SlotDispAttrGroup> _uschemeDispAttrs;
```

Old ScalarSlot fields:

```
ScaledUnitPtr _scaledUnit;
QString _format;
int _precision;
```

New ScalarSlot fields:

```
unsigned long _uschemeSerialNumber;
SlotDispAttrGroup _localDispAttrs;
SlotDispAttrGroup _uschemeDispAttrs;
```

4.2 Modifications to Numeric Formatting and Edit Application Functions

A relatively small number of low-level functions are used for the conversion of internal numeric slot values to displayable strings, and for the setting of internal numeric slot values from entered strings. Some consolidation occurred with the implementation of optional comma separators. We may, however, want to slightly change the low level API for these purposes:

(1) Further simplify the GUI client code. Currently, there are separate calls making use of a slot-related configuration information to: (a) retrieve a slot’s numeric value as a double-precision number in user “display” units, and (b) format that user value using the configured display format and precision. It will be clearer, and more efficient to define a low-level API method to do the string generation in a single step, especially since we’re defining an aggregate with all parameters needed for that generation.

(2) At least tactically, there is some value in modifying the low level API (even just temporarily) as a way of reliably identifying all places in code which make use of these methods -- for the purpose of designing a comprehensive set of tests.

This is relevant for the numeric slot display dialogs enumerated in the prior section -- including for import and export operations, and functions within these modules:

- Output Devices
- Excel Database DMI Datasets
- Old-Style DMIs (making use of units configured on slots).

4.3 Modifications to Slot Initialization Mechanism

Initialization of new **physical slots** (slots on newly created simulation objects -- other than when loading a RiverWare model or importing a simulation object) is now accomplished by "cloning" the corresponding prototype object (instance). The prototype objects' attributes are currently initialized from the external "riverwareDB" file. The following changes will be made:

1. The Prototype Objects' slots will no longer be initialized from the riverwareDB file.
2. After cloning a Prototype Object for the creation of a new simulation object, the settings from the Initialization Slot Attribute Specification Set, and conditionally, the current Unit Scheme (if a User Unit Scheme is current) are applied to the new object's slots.

Initialization of new **account slots** (slots on newly created accounts) will be initialized using these steps:

1. Account slot's attributes will be initialized from values on the prototype account instance (of the type of the new account).
2. The settings from the Initialization Slot Attribute Specification Set, and conditionally, the current Unit Scheme (if a User Unit Scheme is current) are applied to the new accounts's slots.

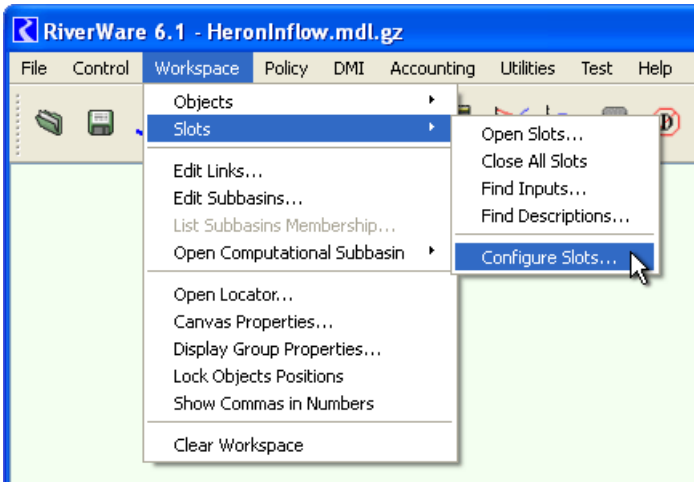
Other slots (e.g. on Exchanges) are initialized from only the Initialization Slot Attribute Specification Set, and conditionally, the current Unit Scheme (if a User Unit Scheme is current).

4.4 New GUI Modules

Two similar new dialogs will be developed. They will be added to the Workspace >> Slots menu, under the "Configure Slots" menu item:

- Slot Attribute Initialization...
- Unit Schemes...

These dialog boxes will be similarly titled. They will have a common implementation. (That is, they will be implemented by the same C++ GUI module). These dialogs will edit a single instance of a "Slot Attribute Specification Set". These push buttons will appear at the bottom of the dialog:



- OK -- *apply and close*
- Apply -- *enabled only when there are pending (unapplied) changes*
- Cancel / Close -- *label depends on whether there are pending (unapplied) changes*

The “Slot Attribute Initialization” dialog will have three panels, vertically arranged -- one for each of the different types of Slot Attribute Groups (Slot Display, Slot Value Limits, and Slot Value Convergence).

The Unit Schemes dialog will have only one panel, for specification rules involving only Slot Display Attribute Groups. This dialog will have also:

- A menu operation to create a new Unit Scheme
- A menu operation to duplicate the currently edited Unit Scheme
- A combo box to switch between the currently defined (User) Unit Schemes.

When there are any pending (un-applied) changes, these controls will be disabled, and an “Apply” button at the bottom of the dialog will be enabled.

See also the “Slot Attribute Specification Set Editing Support (GUI)” section, above.

4.5 Modifications to Existing Dialogs

Most dialogs displaying numeric slot values -- and dialogs for preparing import / export operations which make use of display units configured on slots -- will have a “Unit Scheme” combo box tied to a single, globally maintained “current Unit Scheme”. (That is, all Unit Scheme combo boxes in the RiverWare session will be synchronized).

This includes the following dialogs:

- Open Object Dialog
- Edit Account Dialog
- Open Account Dialog
- Open Slot Dialogs (of various types)
- Object Account Summary Dialog
- Exchange Balance Dialog
- SCT
- Output Manager and Output Config dialogs

TBD: There will be a global toggle to hide all the unit scheme combo boxes in dialogs (all relevant dialogs -- or only dialogs also showing units?). This toggle would automatically be turned on when loading a model having User Unit Schemes, or when adding the first scheme.

5.0 Major Deferred Issues and Design Ideas

The sections below describe the disposition of ancillary issues and ideas presented in the “Unit Schemes in RiverWare 6.1” analysis document.

5.1 Ability to “correct” data entered or imported with wrong units

The choice to continue support for slot and slot-column instance-based unit settings (where this “correction” feature is now supported) allows us to not address this function in the context of the new “Unit Schemes” architecture.

5.2 Series Timestep Size-Dependent Matching Rules

Unit Schemes will not provide “alternate” display units which are dependent on the timestep size of a series slot. For example, we will not be supporting Slot Attribute Specification Lookup Keys of the form “flow in a daily time series” (e.g. for the “Gain Loss” slot on a storage account).

5.3 Legacy Slot-Based Unit Configuration / Unit Scheme Integration

Of the two techniques mentioned -- in the Unit Schemes analysis document -- for integrating the new and old slot unit mechanisms, only the first (described above) will be implemented.

1. Unit Scheme selection controls (probably, combo boxes) could include a “Slot-Configured Units” item
2. **NOT TO BE IMPLEMENTED:** Within any given matching rule in a particular Unit Scheme, the “range value” of a matching rule could be “Slot-Configured Units” -- that is, in place of specific values for unit scale, unit, display format and precision. This setting might be independent for: (1) units, and (2) display format and precision.

6.0 Development Estimate

Task	Days	Description
Basic Slot Attribute Specification Set and "Unit Scheme" Support		
(2.1)	1.0	**Unit Scheme Manager and Stubbed Unit Scheme.** Define a module to manage multiple named instances of Unit Schemes in the RiverWare model, and a Stubbed "Slot Attribute Specification Set" class and Stubbed "User Unit Scheme" subclass. The manager will instantiate a small set of test User Unit Schemes.
(2.2)	3.0	**Basic Slot Attribute Specification Set Implementation and Supporting classes:** (1) Slot Attribute Specification Lookup Keys (with 8 matching criteria types). (2) Support for "pseudo-unit types" (for constrained implementation of "alt units"). (3) Attribute Groups: Slot Display / Slot Value Limit / Slot Convergence ... and slot attribute group lookup method.
(2.3)	3.0	**Editor Dialog for Slot Attribute Specification Set / User Unit Schemes,** using a design similar to the recently developed "Set SCT Aggregation Summary Functions" dialog.
(2.4)	1.5	Slot Attribute Specification Set **Serialization and Persistence** in RiverWare model file.
"Unit Scheme" use for Display and Edit Interpretation of numeric Slot values		
(3.1)	3.0	**Low-Level (Slot class and Aim/SlotGUIUtils) User Unit Scheme API** changes and implementation.
(3.2)	3.0	**SlotQtDlg-implemented GUI applications** (Open Slot, Edit Account, etc). (1) Globally controlled Unit Scheme selectors (QComboBox) (2) Numeric Display using the currently selected Unit Scheme (3) Edit Interpretation using the currently selected Unit Scheme
(3.3)	1.5	**SCT changes** (similar to above)
(3.4)	2.0	**Output Device changes,** testing.
(3.5)	2.5?	**DMI changes and testing**
Slot Attribute Initialization Changes		
(4.1)	1.5	**Initialization Slot Attribute Specification (Singleton)** management, GUI editor, persistence.
(4.2)	1.0	**Physical Slot initializations,** testing.
(4.3)	1.0	**Account Slot initializations,** testing.
Finishing		
(5.1)	3.0	**Testing and Review Revisions**
(5.2)	2.0	**Documentation**
	29.0	**[Days] TOTAL -- estimate revision 6-14-2011 (pw).**