

# Robotics Service Bus - Enhancement #1780

## Improve width computation of column-based styles

02/20/2014 09:07 AM - J. Moringen

<b>Status:</b>	Resolved	<b>Start date:</b>	02/20/2014
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	J. Moringen	<b>% Done:</b>	100%
<b>Category:</b>	Common Lisp Tools	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	rsb-0.11		
<b>Description</b>			
<b>Related issues:</b>			
Blocks Robotics Service Bus - Bug # 1778: timeline/scope style does not show ...		<b>Resolved</b>	<b>02/18/2014</b>

### Associated revisions

#### Revision 86ae603a - 10/24/2014 03:21 AM - J. Moringen

Simulated annealing column width optimizer in formatting/dynamic-width.lisp

refs #1780

Column width computations are now performed by a simulated annealing optimizer that takes an available width and width and priority specifications for individual columns as input and outputs a width for each columns.

- formatting/dynamic-width.lisp (make-widths): new function; expand width specifications into lists of possible widths
- (random-but-not): new function; random number from a range with one exception
- (random-elt-between): new function; random element from a range of ordered elements
- (score-table): new constant; precomputed first 1000 score values
- (%optimize-widths): new function; the actual optimizer
- (optimize-widths): new function; convenience function for handling cases with fewer than two columns
- formatting/package.lisp (package rsb.formatting): added exported symbol optimize-widths
- test/formatting/dynamic-width.lisp: new file; contains unit tests for `optimize-widths`
- cl-rsb-formatting.asd (system cl-rsb-formatting-test) added file

#### Revision 6a1e4899 - 10/24/2014 03:21 AM - J. Moringen

Infrastructure for using width optimizer in formatting/text-style-mixins.lisp

refs #1780

The new generic functions

- `style-dynamic-width-columns`
- `style-compute-column-widths`
- `style-assign-column-widths`

form a protocol for dynamic computation of column widths. The mixin classes

```
`width-specification-mixin'  
`widths-caching-mixin'
```

make implementing the protocol convenient.

- formatting/types.lisp (width-specification?): new function; helper for type `width-specification`  
(width-specification): new type; for column width specifications
- formatting/protocol.lisp (style-dynamic-width-columns): new generic function; return columns for which width should be dynamically adjusted  
(style-compute-column-widths): new generic function; compute widths for dynamic width columns  
(style-assign-column-widths): new generic function; assign computed widths to dynamic width columns  
(style-dynamic-width-columns t): new method; default behavior  
(style-compute-column-widths t sequence integer): likewise  
(style-assign-column-widths t sequence sequence): likewise
- formatting/text-style-mixins.lisp (width-specification-mixin):  
(self column-widths :before t width-specification-mixin): new method; check width specification  
(shared-initialize :before width-specification-mixin t): new method; check initargs  
(shared-initialize :after width-specification-mixin t): new method; assign width specification  
(width-mixin): removed width initform and allowed width 0 to interoperate with `width-specification-mixin`  
(widths-caching-mixin): new class; caches computed widths  
(style-compute-column-widths widths-caching-mixin sequence integer): new method; compute widths or use cached values
- formatting/package.lisp (package rsb.formatting): added exported symbols width-specification, style-dynamic-width-columns, style-compute-column-widths, style-assign-column-widths, width-specification-mixin, column-widths and column-priority
- test/formatting/text-style-mixins.lisp  
(test suite width-specification-mixin-root): new test suite; for class `width-specification-mixin`  
(test case width-specification-mixin-root::construction): new test case; test constructing `width-specification-mixin` instances

#### Revision 02340751 - 10/24/2014 03:21 AM - J. Moringen

Use width optimizer instead of switching tricks in formatting/\*.lisp

fixes #1780, fixes #1778

Use the simulated annealing column width optimizer in all column-based styles, eliminating the need for switching between hand-crafted layouts for a handful of width thresholds.

This change allows merging the "compact" and "full" variants of now, origin and id columns into one respective column with two discrete widths.

- formatting/protocol.lisp (style-separator-width): new generic; return width of a style's separator  
(style-separator-width t): new method; default behavior
- formatting/dynamic-width.lisp (number-of-columns): removed; no longer needed  
(define-dynamic-width-style): likewise
- formatting/text-style-mixins.lisp (%separator-width): new function; utility used by methods on `separator-width`  
(when-column-fits): removed; no longer needed  
(format-header columns-mixin t): do not truncate columns via `when-column-fits`  
(format-event columns-mixin t): likewise
- formatting/quantity-column.lisp (quantity-column): added superclass `width-specification-mixin`
- formatting/columns.lisp (define-simple-column): accept priority; generate widths specification and priority; use `width-specification-mixin` as superclass  
(define-simple-column :now): allow widths 15 and 32 instead of only 32; switch to appropriate display style based on width  
(define-simple-column :now/compact): removed  
(define-simple-column :origin): allow widths 8 and 36 instead of only 8; switch to appropriate display style based on width  
(define-simple-column :origin-full): removed  
(define-simple-column :id): like :origin  
(define-simple-column :id-full): removed  
(define-simple-column :scope): allow a range of widths  
(define-simple-column :wire-schema): likewise  
(define-simple-column :data): likewise  
(define-simple-column :call): likewise  
(define-simple-column :result): likewise  
(column-constant): added superclass `width-specification-mixin`  
(define-meta-data-column): likewise  
**(basic-columns)**: width adjustments; added some priorities
- formatting/timeline.lisp (timeline): added superclass `width-specification-mixin`; added widths default initarg
- formatting/event-style-compact.lisp (event-style-compact-line): new class; helper class for `event-style-compact`  
(default-compact-sub-styles): new function; helper for computing default sub-styles  
(defmethod find-style-class eql :compact): new method; find style class  
(event-style-compact): renamed basic-compact-style -> event-style-compact; compact event formatting style width dynamic column width computation  
(sub-style-for event-style-compact): changed specializer basic-compact-style -> event-style-compact  
(format-header event-style-compact t): likewise  
(format-event :before t event-style-compact t): new method; compute column widths and assign to sub-styles

- (define-compact-style): removed; no longer needed
- (define-compact-style compact/80): likewise
- (define-compact-style compact/128): likewise
- (define-compact-style compact/180): likewise
- (define-dynamic-width-style compact): likewise
- formatting/event-style-monitor.lisp (basic-monitor-style): added superclass widths-caching-mixin
- (style-dynamic-width-columns basic-monitor-style): new method; return dynamic width columns for sub-styles
- (format-event :before eql :trigger basic-monitor-style t): new method; compute and apply column widths for sub-styles
- (define-dynamic-width-monitor-style): removed; no longer needed
- (define-monitor-style timeline): changed define-{dynamic-width ->}monitor-style
- (define-monitor-style scope): likewise
- (define-monitor-style origin): likewise
- (define-monitor-style type): likewise
- (define-monitor-style size): likewise
- formatting/event-style-statistics.lisp (define-statistics-style): removed; no longer needed
- (define-statistics-style statistics/80): likewise
- (define-statistics-style statistics/128): likewise
- (define-statistics-style statistics/180): likewise
- (define-statistics-style statistics/220): likewise
- (define-dynamic-width-style statistics): likewise
- (find-style-class eql :statistics): new method; find new statistics class
- (style-statistics): new class; replaces previous width-specialized statistics classes
- (format-event :before eql :trigger style-statistics t): new method; compute widths of columns
- formatting/event-style-timeline.lisp (define-timeline-style scope): added widths specification and priority
- (define-timeline-style origin): likewise
- formatting/package.lisp (package rsb.formatting): removed exported symbols style-compact/80, style-compact/128, style-compact/180, style-statistics/80, style-statistics/128, style-statistics/180, and style-statistics/220; added exported symbols style-separator, style-separator-width
- test/formatting/style-compact.lisp (test suite style-compact/80-root): removed; replace by style-compact test suite
- (test case style-compact/80-root::smoke): likewise
- (test suite style-compact/128-root): likewise
- (test case style-compact/128-root::smoke): likewise
- (test suite style-compact-root): new test suite; for style-compact
- (test case style-compact-root::smoke): new test case; test compact style for 80 and 128 columns

**Revision 87beaa6a - 10/24/2014 04:11 AM - J. Moringen**

Mention width optimization for column-based styles in news.rst

refs #1780

- news.rst (0.11): mention the fact that column-based styles now compute column widths dynamically instead of via predefined layouts

#### **Revision 71de361e - 10/26/2014 06:11 PM - J. Moringen**

Adjusted RPC-related column width specifications in formatting/columns.lisp

refs #1780

The minimum width 57 of RPC request- and reply-related columns made them disappear entirely for terminal widths below 82. This behavior was worse than necessary and also not in line with the formatting of non-RPC events.

- formatting/columns.lisp (define-simple-column id): reduced priority 1.5 -> 1 for symmetry with call-id column; this should be fine since full event IDs should be of interest only very rarely  
(define-simple-column call): reduced minimum width to 26 for symmetry with non-RPC columns  
(define-simple-column call-id): provide width options 8 and 36 with priority 1 like the id column; switch between compact and full ID formatting based on selected width option  
(define-simple-column result): reduced minimum width to 26 for symmetry with non-RPC columns

#### **History**

---

##### **#1 - 03/25/2014 05:30 PM - J. Moringen**

- Target version changed from rsb-0.11 to rsb-0.12

##### **#2 - 10/24/2014 03:40 AM - J. Moringen**

- Status changed from New to Resolved

- % Done changed from 0 to 100

Applied in changeset commit:rsb-tools-cl|023407512034b4b05ef06b3403263526c2b8aa42.

##### **#3 - 10/24/2014 04:15 AM - J. Moringen**

- Target version changed from rsb-0.12 to rsb-0.11