Robotics Service Bus - Tasks #489
Increase performance of less operator (bool operator< const) of Scope

08/08/2011 08:57 PM - Anonymous

Status:	Resolved		**Start date:**	08/08/2011
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	100%
Category:	C++		**Estimated time:**	0.00 hour
Target version:				

Description

When using inprocess transport in C++, Scope is used as a key type for a std::map. Every time find() is performed (e.g. in every Bus::handle call - source:trunk/cpp/core/src/rsb/transport/inprocess/Bus.cpp), the less operator is called multiple times. Right now the less operator is implemented by comparing the Scopes as string. In one of my test scenarios this consumed a significant part (~50%) of computation. I basically see two options to improve this:

1. Come up with another way of comparing Scopes
2. Render the string once when the Scope changes (which only happens on creation where the string is given anyway and on concat() AFAIR), and then let toString() just return this.

The second option would be quite straight forward and already provides a significant speed-up of inprocess transport. Any opinions on this?
(This would of course also improve performance of the == operator)

Associated revisions

Revision 96df65cf - 08/08/2011 09:50 PM - Arne Nordmann

Speed-up of Scope comparison
fixes #489

Revision 29d17584 - 08/08/2011 11:09 PM - J. Moringen

Fixed Scope::concat for warm string caches in src/rsb/Scope.cpp
refs #489
  - src/rsb/Scope.cpp: start with empty string cache in concatenation
    result
  - test/rsb/ScopeTest.cpp: test toString() method on the result of
    concatenating Scope objects with warm string caches

History

#1 - 08/08/2011 09:50 PM - Anonymous

- Status changed from New to Resolved

- % Done changed from 0 to 100

Applied in changeset r2302.