

Robotics Service Bus - Enhancement #1026

Add transport class [Common Lisp]

06/25/2012 02:48 PM - J. Moringen

Status:	Resolved	Start date:	06/25/2012
Priority:	Normal	Due date:	
Assignee:	J. Moringen	% Done:	100%
Category:	Common Lisp	Estimated time:	0.00 hour
Target version:	rsb-0.13		
Description			
The new Transport class would represent properties of transport as a whole (e.g. Spread, socket, inprocess), <ul style="list-style-type: none">- Instances should aggregate the respective factory objects (in-push, in-pull, out, etc.)- Some Connector properties such as schema list could be moved into the new Transport class			
Related issues:			
Related to Robotics Service Bus - Tasks # 1030: Expose connector "remoteness"...		Resolved	06/25/2012
Related to Robotics Service Bus - Enhancement # 2199: Spread connection pooli...		Resolved	03/11/2015

Associated revisions

Revision 6ec17f13 - 04/29/2015 03:45 AM - J. Moringen

Adapted access to transport information in common/help.lisp, src/commands/info.lisp

refs #1026

- common/help.lisp (print-all-uri-synopsis):
- src/commands/info.lisp (info):
 - (shared-initialize):
 - (command-execute):

Revision ea3ce916 - 02/10/2016 01:30 PM - J. Moringen

Added transport service in src/transport/transport.lisp

refs #1026

This change introduces first-class transport objects which

- 1) define the supported schemas and the wire-type of the transport
- 2) are services managing multiple providers - one for each direction - which construct connector objects

Transport objects in turn are managed by a transport service.

- src/transport/protocol.lisp (header): updated copyright
 - (transport-wire-type): new generic function; return wire-type of the transport
 - (define-transport-accessor): new local macro; define accessor for transports
 - (make-connector): prepared for changed connector instantiation; adapted documentation string

- (make-connectors): adapted documentation string
- src/transport/transport.lisp: new file; transport class and necessary service-provider extensions
- src/transport/connector-class.lisp
 - (define-connector-class-accessor transport-wire-type): new accessor; delegate from connector to class
- src/transport/package.lisp (header): updated copyright
 - (package rsb.transport): added exported symbols transport-wire-type, register-{transport,connector}, transport; adapted documentation string
- test/transport/package.lisp (define-basic-connector-test-cases): will use service-provider protocol instead of `find-connector-class`
- cl-rsb.asd (system cl-rsb): added file src/transport/transport.lisp

Revision 5ae10eb7 - 02/10/2016 02:36 PM - J. Moringen

Store transport in connector class in src/transport/*.lisp

refs #1026

In order to move properties into the respective transport object, connector instances and connector classes have to

- 1) know the transport object they belong to
- 2) delegate property reads to the transport

- src/transport/protocol.lisp (connector-transport): new generic function; return transport to which the connector belongs
 - (define-connector-class-accessor connector-transport): new accessors
- src/transport/connector-class.lisp (connector-class::transport): new slot; stores transport to which the connector belongs
 - (shared-initialize :before connector-class t): accept transport keyword parameter; coerce into transport object and store in transport slot
 - (connector-transport connector-class): new method; return slot value if present, delegate to superclasses otherwise
 - (define-connector-class-accessor connector-transport): new accessor; delegate from connector to class
- src/transport/package.lisp (package rsb.transport): added exported symbol connector-transport
- test/event-processing/in-route-configurator.lisp (header): updated copyright
 - (register-transport :mock): mock transport
 - (mock-connector): added :transport option
- test/transport/connector-class.lisp
 - (test connector-class-root::construction/valid): register mock transport; use in :transport option in mock transport class

Revision 9a179a9c - 02/10/2016 02:37 PM - J. Moringen

Use new transport infrastructure in src/transport/inprocess/*.lisp

refs #1026

- src/transport/inprocess/transport.lisp: new file; contains first-class transport object for inprocess transport
- src/transport/inprocess/connectors.lisp (header): updated copyright
 - (**by-scope**): removed; moved into transport object
 - (**cached-machine-instance**): likewise
 - (**cached-process-id**): likewise
 - (update-cached-machine-instance-and-process-id): moved to src/transport/inprocess/transport.lisp
 - (connector): added :transport option
 - (connector::transport): new slot; stores transport object to which the connector belongs
 - (shared-initialize :around connector t): new method; added %transport, :host and :port initargs to method call
 - (notify in-connector scope eql :attached): use transport object stored in connector instead of **by-scope**
 - (notify in-connector scope eql :detached): likewise
 - (register-connector :inprocess :in-pull): new; register in-pull connector
 - (register-connector :inprocess :in-push): similar for in-push
 - (register-connector :inprocess :out): similar for out
 - (handle out-connector event): use transport object stored in connector instead of **by-scope**
- rsb-transport-inprocess.asd (header): updated copyright
 - (system rsb-transport-inprocess): added file src/transport/inprocess/transport.lisp

Revision 15674aa6 - 02/10/2016 02:37 PM - J. Moringen

Use new transport infrastructure in src/transport/socket/*.lisp

refs #1026

- src/transport/socket/protocol.lisp (header): updated copyright
 - (transport-ensure-bus): new generic function; return a specified bus, creating it if necessary
 - (transport-ensure-bus): new generic function; return, creating it if necessary, a `bus-client` or `bus-server` instance for given parameters
- src/transport/socket/util.lisp (header): updated copyright
 - (with-locked-bus): new macro; moved here from src/transport/socket/bus.lisp
- src/transport/socket/transport.lisp: new file; contains first-class transport for socket transport implementation
- src/transport/socket/bus.lisp (header): updated copyright
 - (with-locked-bus): removed; moved to src/transport/socket/util.lisp
- src/transport/socket/bus-client.lisp (header): updated copyright
 - (**bus-clients**): removed; no longer needed
 - (**bus-clients-lock**): likewise

- (ensure-bus-client): removed; moved to
src/transport/socket/transport.lisp
- src/transport/socket/bus-server.lisp (header): updated copyright
(**bus-servers**): removed; no longer needed
(**bus-servers-lock**): likewise
(ensure-bus-server): removed; moved to
src/transport/socket/transport.lisp
- src/transport/socket/connector.lisp (header): updated copyright
(**default-host**): removed; moved to
src/transport/socket/transport.lisp
(**default-port**): likewise
(connector): added :transport option
(notify connector scope eql :attached): adapted to use
`transport-ensure-bus' instead of `%get-bus'
(%get-bus): removed; replaced by `transport-ensure-bus' generic
function
- src/transport/socket/in-pull-connector.lisp (header): updated
copyright
(register-connector :socket :in-pull): new; register connector
- src/transport/socket/in-push-connector.lisp (header): updated
copyright
(register-connector :socket :in-push): new; register connector
- src/transport/socket/out-connector.lisp (header): updated copyright
(register-connector :socket :out): new; register connector
- test/transport/socket/bus.lisp (header): updated copyright
(test transport-socket-bus-root::client/smoke): adapted to changes
(test transport-socket-bus-root::server/smoke): likewise
(test transport-socket-bus-root::server/automatic-port): likewise
- test/transport/socket/package.lisp (header): updated copyright
(package rsb.transport.socket.test): removed imported-from
rsb.transport.socket symbols **bus-{client,server}**,
ensure-bus-{client,server}; added imported-from rsb.transport.socket
symbols transport-ensure-bus-{client,server}
(test suite transport-socket-root): adapted to changes
- rsb-transport-socket.asd (header): updated copyright
(system rsb-transport-socket): added file
src/transport/socket/transport.lisp

Revision c6c7eba4 - 02/10/2016 02:37 PM - J. Moringen

Use new transport infrastructure in src/transport/spread/*.lisp

refs #2199 refs #1026

- src/transport/spread/transport.lisp: new file; contains first-class
transport object for the transport implementation
- src/transport/spread/connector.lisp (header): updated copyright
(connector): added :transport option
- src/transport/spread/in-pull-connector.lisp (header): updated
copyright
(register-connector :spread :in-pull): new; register connector
- src/transport/spread/in-push-connector.lisp (header): updated

- copyright
- (register-connector :spread :in-push): new; register connector
- src/transport/spread/out-connector.lisp (header): updated copyright
- (register-connector :spread :out): new; register connector
- rsb-transport-spread.asd (header): updated copyright
- (system rsb-transport-spread): added file
- src/transport/spread/transport.lisp

Revision e1990e4b - 02/10/2016 02:37 PM - J. Moringen

Store wire-type in transport instead of connector in src/transport/*.lisp

refs #1026

- src/transport/protocol.lisp (connector-wire-type): removed; replaced by `transport-wire-type'
- (define-connector-class-accessor connector-wire-type): removed; no longer needed
- (make-connector): retrieve wire-type from transport instead of connector-class
- src/transport/connector-class.lisp (connector-class::wire-type): removed slot; now stored in transport
- (shared-initialize :before connector-class t): do not accept wire-type keyword parameter
- (connector-wire-type): removed; replaced by delegation to transport
- (define-connector-class-accessor transport-wire-type): new method; delegate from connector class to transport
- src/transport/package.lisp (package rsb.transport): removed exported symbol connector-wire-type
- src/transport/inprocess/connectors.lisp (connector): removed :wire-type option
- src/transport/socket/connector.lisp (connector): likewise
- src/transport/spread/connector.lisp (connector): likewise
- test/event-processing/in-route-configurator.lisp (mock-connector): likewise
- test/transport/connector-class.lisp
- (test connector-class-root::construction/valid): removed :write-type option from mock connector class; changed {connector -> transport}-wire-type
- test/transport/package.lisp (check-connector-class): similar
- (check-connector): similar

Revision 04127dd9 - 02/10/2016 02:37 PM - J. Moringen

Store schemas in transport instead of connector in src/transport/**/*.*.lisp

refs #1026

- src/transport/protocol.lisp (transport-schemas): new generic function; return schemas for transport

- (define-transport-accessor transport-schemas): new methods; lookup transport when given symbol
- (connector-schemas): removed; no longer needed
- (make-connector): do not pass :schema initarg to `make-instance'; connector classes no longer accept the initarg
- src/transport/transport.lisp (transport::schemas): new slot; stores schemas supported by the transport
 - (transport): added :schemas default initarg
 - (shared-initialize :after transport t): new method; set schemas slot
 - (describe-object transport t): print schemas
 - (make-provider t transport): accept schema keyword parameter
 - (find-provider eql transport symbol): new method; make schemas into aliases for the transport
 - (find-provider eql transport cons): new method; interpret cons as schema and direction
 - (make-provider eql transport cons): similar
- src/transport/connector-class.lisp (connector-class::schemas): removed; no longer needed
 - (connector-schemas connector-class): likewise
 - (define-connector-class-accessor transport-schemas): new methods; delegate from connector to transport
 - (define-connector-class-accessor connector-schemas): removed; no longer needed
- src/transport/package.lisp (package rsb.transport): added exported symbol transport-schemas; removed exported symbol connector-schemas
- src/transport/inprocess/transport.lisp (register-transport inprocess): added :schemas initarg
- src/transport/inprocess/connectors.lisp (connector): removed :schemas option; removed :schema default initarg
- src/transport/socket/transport.lisp (register-transport socket): added :schemas initarg
- src/transport/socket/connector.lisp (connector): removed :schemas option
- src/transport/spread/transport.lisp (register-transport spread): added :schema initarg
- src/transport/spread/connector.lisp (connector): removed :schemas option; removed schema default initarg
- src/clon.lisp (header): updated copyright
 - (make-options-for-connector-class): changed {connector -> transport}-schemas
- test/event-processing/in-route-configurator.lisp
 - (register-transport :mock): added :schemas initarg
- test/transport/connector-class.lisp
 - (test connector-class-root::construction/valid): likewise; adapted to changed protocol
- test/transport/package.lisp (check-connector-class): likewise

Revision 3f689767 - 02/10/2016 02:37 PM - J. Moringen

Adaptation to changed transport protocol in src/clon.lisp

refs #1026

- src/clon.lisp (header): updated copyright
 - (make-options-for-connector-class): adapted to changed transport and connector protocols
 - (make-options-for-connector-classes): introspect transports using the transport service

Revision 8f0260cb - 02/10/2016 02:37 PM - J. Moringen

Removed connector class family in src/transport/*.lisp

fixes #1026

First-class transport objects and the transport service have taken over all responsibilities. Remove the class-family-based parts.

- src/transport/protocol.lisp
 - (dynamic-classes:define-findable-class-family transport): removed; no longer needed
 - (make-connector): adapted to changed connector instantiation; adapted documentation string
 - (make-connectors): adapted documentation string
- src/transport/package.lisp (package rsb.transport): removed exported symbols {no-such,find}-transport and transport-classes
- src/transport/inprocess/connectors.lisp
 - (find-transport-class eql :inprocess-in-pull): removed; no longer needed
 - (find-transport-class eql :inprocess-in-push): likewise
 - (find-transport-class eql :inprocess-out): likewise
- src/transport/socket/in-pull-connector.lisp
 - (find-transport-class eql :socket-in-pull): likewise
- src/transport/socket/in-push-connector.lisp
 - (find-transport-class eql :socket-in-push): likewise
- src/transport/socket/out-connector.lisp
 - (find-transport-class eql :socket-out): likewise
- src/transport/spread/in-pull-connector.lisp
 - (find-transport-class eql :spread-in-pull): likewise
- src/transport/spread/in-push-connector.lisp
 - (find-transport-class eql :spread-in-push): likewise
- src/transport/spread/out-connector.lisp
 - (find-transport-class eql :spread-out): likewise
- test/transport/package.lisp (define-basic-connector-test-cases): use service-provider protocol instead of `find-connector-class`

Revision e13fb0f4 - 02/10/2016 03:04 PM - J. Moringen

Adaptations to transport changes in src/transport/yarp[tcp]/*.lisp

refs #1026

- src/transport/yarptcp/connector.lisp (header): updated copyright
 - (register-transport :tcp+yarp): new form; register YARP TCP carrier

(connector): removed :wire-type and :schemas options; added :transport option

- src/transport/yarptcp/in-pull-connector.lisp (header): updated copyright
(find-transport-class eql :tcp+yarp-in-pull): removed; no longer needed
(register-connector :tcp+yarp :in-pull): new form; register YARP TCP in-pull connector
- src/transport/yarptcp/in-push-connector.lisp (header): updated copyright
(find-transport-class eql :tcp+yarp-in-push): removed; no longer needed
(register-connector :tcp+yarp :in-push): new form; register YARP TCP in-push connector
- src/transport/yarptcp/in-push-connector.lisp (header): updated copyright
(find-transport-class eql :tcp+yarp-out): removed; no longer needed
(register-connector :tcp+yarp :out): new form; register YARP TCP out connector
- src/transport/yarp/connectors.lisp (header): updated copyright
(register-transport :yarp): new form; register YARP transport with nameserver lookup
(connector): removed :wire-type and :schemas options; added :transport option
(define-connector): do not generate `find-transport-class' method; generate `register-connector' form
(%make-peer): use `make-connector instead of `make-instance'

Revision 3d97b411 - 02/10/2016 03:26 PM - J. Moringen

Adaptations to changed transport protocol in src/transport/[tcp]ros/**/* .lisp

refs #1026

- src/transport/connector-mixins.lisp (header): updated copyright
(message-definition-mixin): removed :wire-type option
- src/transport/ros/connector.lisp (header): updated copyright
(register-transport :ros): new; register the transport
(connector): removed :schemas option; added :transport option; removed default initargs
- src/transport/ros/in-connector.lisp
(find-transport-class eql :ros-in-push): removed; no longer necessary
(register-connector :ros :in-push): new; register the connector
- src/transport/ros/out-connector.lisp (header): updated copyright
(find-transport-class eql :ros-out): removed; no longer necessary
(register-connector :ros :out): new; register the connector
- src/transport/tcpros/connector.lisp (header): updated copyright
(register-transport :tcp+ros): new; register the transport
(connector): removed :schemas option; added :transport option; removed default initargs

- src/transport/tcpros/in-pull-connector.lisp (header): updated copyright
(find-transport-class eql :tcp+ros-in-pull): removed; no longer needed
(register-connector :tcpros :in-pull): new; register the connector
- src/transport/tcpros/in-push-connector.lisp (header): updated copyright
(find-transport-class eql :tcp+ros-in-push): removed; no longer needed
(register-connector :tcpros :in-push): new; register the connector
- src/transport/tcpros/out-connector.lisp (header): updated copyright
(find-transport-class eql :tcp+ros-out): removed; no longer needed
(register-connector :tcpros :out): new; register the connector

Revision 4493873f - 02/10/2016 03:37 PM - J. Moringen

Adapted transport info access in src/{common/help,commands/info}.lisp

refs #1026

- common/help.lisp (print-uri-synopsis): adapted to transport protocol changes
(print-all-uri-synopsis): likewise
- src/commands/info.lisp (info::transports?): renamed slot connectors?
-> transports?
(shared-initialize shared-initialize :after info t): adapted to renamed slot
(command-execute info): likewise; adapted to transport protocol changes
- info/main.lisp (header): updated copyright
(update-synopsis): changed commandline option connectors -> transport
(main): adapted to commandline option and command class changes
- test/commands/info.lisp (header): updated copyright
(test info-root::construction): changed initarg connectors? -> transports?
- CMakeLists.txt: adapted to renamed commandline option

Revision 56c126fe - 02/10/2016 04:17 PM - J. Moringen

Changed transports commandline option in tool-info.rst

refs #1026

- tool-info.rst (Description): adapted to renamed commandline option connectors -> transports

History

#1 - 04/17/2013 04:09 PM - J. Moringen

- Target version changed from rsb-0.9 to rsb-0.10

#2 - 12/10/2013 11:47 PM - J. Moringen

- Target version changed from rsb-0.10 to rsb-0.11

#3 - 03/07/2014 03:21 AM - J. Moringen

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

#4 - 04/23/2015 11:24 AM - J. Moringen

- Status changed from New to In Progress

- % Done changed from 0 to 20

#5 - 04/23/2015 11:25 AM - J. Moringen

- Related to Enhancement #2199: Spread connection pooling for informers [Common Lisp] added

#6 - 04/27/2015 11:13 AM - J. Wienke

- Target version changed from rsb-0.12 to rsb-0.13

#7 - 02/10/2016 03:00 PM - J. Moringen

- Status changed from In Progress to Resolved

- % Done changed from 20 to 100

Applied in changeset commit:rsb-cl|8f0260cb2d69f1f3316af8f720a28f5f3f40cf4b.

#8 - 09/18/2017 07:39 PM - J. Moringen

- Subject changed from Add Transport class to Add transport class [Common Lisp]