

Robotics Service Bus - Enhancement #890

Logger should accept multiple URIs

02/17/2012 12:33 PM - J. Moringen

Status:	Resolved	Start date:	02/17/2012
Priority:	Normal	Due date:	
Assignee:	J. Moringen	% Done:	100%
Category:	Common Lisp Tools	Estimated time:	0.00 hour
Target version:	rsb-0.11		
Description			
For example:			
<pre>logger --style 'timeline' 'tcp+yarp://129.70.143.10:10110' 'spread://azurit:4803'</pre>			

Associated revisions

Revision a3fedbec - 02/25/2014 02:00 PM - J. Moringen

Handle error policies properly in participant creation functions and macros

refs #890

- src/protocol.lisp (make-listener): added error-policy keyword parameter; adapted documentation string
(make-reader): likewise
(make-informer): likewise
- src/participant.lisp (make-participant): accept error-policy keyword parameter; when supplied, install error-policy in the error hook of the created participant; adapted documentation string
(define-participant-creation-uri-methods): added error-policy keyword parameter
- src/reader.lisp (make-reader scope): accept error-policy keyword parameter; pass to `make-participant`
- src/listener.lisp (make-listener scope): likewise
- src/informer.lisp (make-informer scope): similar
- src/macros.lisp (header): updated copyright
(define-with-participant-macro): accept error-policy keyword parameter
- src/patterns/request-reply/protocol.lisp (make-remote-server): accept error-policy keyword parameter; adapt documentation string accordingly
(make-local-server): similar
- src/patterns/request-reply/server.lisp (define-lazy-creation-method): pass error-policy of the server to its methods
- src/patterns/request-reply/local-server.lisp (make-local-server): accept error-policy keyword parameter; install as error hook of created server
- src/patterns/request-reply/remote-server.lisp (make-remote-server): similar
- test/listener.lisp (header): updated copyright
(define-basic-participant-test-cases listener): test error-policy keyword parameter

- test/reader.lisp (header): updated copyright
(define-basic-participant-test-cases reader): test error-policy keyword parameter
- test/informer.lisp (define-basic-participant-test-cases informer):
test error-policy keyword parameter
- test/macros.lisp (test macros-root::with-listener/smoke): renamed
with-listener{- -> /}smoke
(test macros-root::with-reader/smoke): similar
(test macros-root::with-informer/smoke): similar
(test macros-root::with-listener/error-policy): new
test; test handling of error-policy keyword parameter in
'with-listener' macro
(test macros-root::with-reader/error-policy): similar
(test macros-root::with-informer/error-policy): similar
- test/package.lisp (package rsb.test): added exported symbol
mock-transform/error
(mock-transform/error): new function; transform function which signals
an intentional error
- test/patterns/request-reply/macros.lisp
(test suite with-methods-root): removed
(test macros-root::with-methods/smoke): renamed
with-methods-root::smoke -> macros-root::with-methods/smoke
(test macros-root::with-methods/macroexpand): similar
(test macros-root::with-local-server/error-policy): new test; test
handling of error-policy keyword parameter in 'with-local-server'
macro
(test macros-root::with-remote-server/error-policy): similar

Revision 317c99d1 - 02/25/2014 02:31 PM - J. Moringen

Handle error policies properly in participant creation functions and macros

refs #890

- src/protocol.lisp (make-listener): added error-policy keyword
parameter; adapted documentation string
(make-reader): likewise
(make-informer): likewise
- src/participant.lisp (make-participant): accept error-policy keyword
parameter; when supplied, install error-policy in the error hook of
the created participant; adapted documentation string
(define-participant-creation-uri-methods): added error-policy keyword
parameter
- src/reader.lisp (make-reader scope): accept error-policy keyword
parameter; pass to 'make-participant'
- src/listener.lisp (make-listener scope): likewise
- src/informer.lisp (make-informer scope): similar
- src/macros.lisp (header): updated copyright
(define-with-participant-macro): accept error-policy keyword parameter
- src/patterns/request-reply/protocol.lisp (make-remote-server): accept
error-policy keyword parameter; adapt documentation string accordingly
(make-local-server): similar

- src/patterns/request-reply/server.lisp (define-lazy-creation-method):
pass error-policy of the server to its methods
- src/patterns/request-reply/local-server.lisp (make-local-server):
accept error-policy keyword parameter; install as error hook of
created server
- src/patterns/request-reply/remote-server.lisp (make-remote-server):
similar
- test/listener.lisp (header): updated copyright
(define-basic-participant-test-cases listener): test error-policy
keyword parameter
- test/reader.lisp (header): updated copyright
(define-basic-participant-test-cases reader): test error-policy
keyword parameter
- test/informer.lisp (define-basic-participant-test-cases informer):
test error-policy keyword parameter
- test/macros.lisp (test macros-root::with-listener/smoke): renamed
with-listener{- -> /}smoke
(test macros-root::with-reader/smoke): similar
(test macros-root::with-informer/smoke): similar
(test macros-root::with-listener/error-policy): new
test; test handling of error-policy keyword parameter in
'with-listener' macro
(test macros-root::with-reader/error-policy): similar
(test macros-root::with-informer/error-policy): similar
- test/package.lisp (package rsb.test): added exported symbol
mock-transform/error
(mock-transform/error): new function; transform function which signals
an intentional error
- test/patterns/request-reply/macros.lisp
(test suite with-methods-root): removed
(test macros-root::with-methods/smoke): renamed
with-methods-root::smoke -> macros-root::with-methods/smoke
(test macros-root::with-methods/macroexpand): similar
(test macros-root::with-local-server/error-policy): new test; test
handling of error-policy keyword parameter in 'with-local-server'
macro
(test macros-root::with-remote-server/error-policy): similar

Revision 93c356b2 - 02/25/2014 05:33 PM - J. Moringen

Support multiple URIs in logger in logger/main.lisp

fixes #890

The logger program now creates multiple listeners, all feeding into a thread-safe queue. The main loop extracts and processes events sequentially.

- logger/main.lisp (header): updated copyright
(update-synopsis): adapted URI part of synopsis; added
max-queued-events cmdline option
(queue-overflow-error): new error condition; signaled when a

fixed-capacity event queue overflows

(make-queue-push-handler): new function; helper function for constructing handlers that push event onto a queue with appropriate restarts

(make-queue-pushing-listener): new function; helper function for constructing listeners that push events onto a queue

(process-events): new function; helper function for popping events from a queue and formatting them with appropriate restarts

(main): extract multiple URIs from commandline arguments; turned listener creation into a local function; process events via a queue; detach listeners when unwinding

- logger/help.lisp (make-help-string): adapted to explain multiple URIs (make-examples-string): likewise
- logger/package.lisp (header): updated copyright (package rsb.tools.logger): added used package more-conditions
- cl-rsb-tools-logger.asd (system cl-rst-tools-logger): added dependency on lparallel
- CMakeLists.txt: added test cases with multiple URIs and queue capacity for logger binary

Revision 11b4385b - 02/25/2014 06:01 PM - J. Moringen

Describe multiple-URLs feature in logger.rst

refs #890

- logger.rst (Synopsis): [URI] -> URI* (Description): mention the possibility of supplying multiple URLs; added description of max-queued-events commandline option

Revision b12b8c0b - 02/25/2014 06:07 PM - J. Moringen

Mention multiple-URLs feature in news.rst

refs #890

- news.rst (RSB 0.11): mention that the logger now accepts multiple URIs

History

#1 - 05/21/2013 02:35 PM - J. Moringen

- Status changed from New to In Progress
- % Done changed from 0 to 50

#2 - 01/27/2014 03:09 AM - J. Moringen

- Target version set to rsb-0.11
- % Done changed from 50 to 70

#3 - 02/25/2014 05:50 PM - J. Moringen

- Status changed from In Progress to Resolved
- % Done changed from 70 to 100

Applied in changeset rsb-tools-cl|commit:93c356b2a8b8c7bc65e0dfb9bb7f10ea22ae54ca.