

## Robotics Service Bus - Enhancement #458

### Remove ServerRequestId in Request Event?

07/30/2011 04:41 PM - J. Moringen

Status:	Resolved	Start date:	07/30/2011
Priority:	Low	Due date:	
Assignee:	J. Moringen	% Done:	100%
Category:	Specification	Estimated time:	0.00 hour
Target version:	0.4		

**Description**

See [[RequestReply]] for an explanation of the current scheme.

I think, we could implement this more efficiently as follows:

1. The request event does not have a ServerRequestId meta-data item
2. The requesting RemoteServer uses the event id (let's call it REQUEST\_EVENT\_ID here) to identify the method call
3. The providing Server adds a meta-data item in-reply-to: REQUEST\_EVENT\_ID to its reply event
4. The requesting RemoteServer associates replies to requests using the in-reply-to meta-data items

We decided to do this [[Meetings2011-08-02|here]], so:

- Implement in Java □ (nothing to do since the functionality is not implemented)
- Implement in C++ □
- Implement in Python □
- Implement in Common Lisp □
- Update [[RequestReply|Specification]] □

#### Associated revisions

##### Revision ac5a72f4 - 08/03/2011 09:47 AM - J. Moringen

Do not send additional id in method calls in rsb/patterns/\_\_init\_\_.py

refs #458

- rsb/patterns/\_\_init\_\_.py: do not add user-item ServerRequestId and send the request before storing the call in RemoteMethod.\_\_call\_\_; use user-item rsb:reply in RemoteMethod.\_handleReply; use rsb:reply and rsb:error? in LocalMethod.\_handleRequest

##### Revision 071f9463 - 08/03/2011 10:06 AM - J. Moringen

Do not send additional id in method calls in src/rsb/patterns/\*Server\*

refs #458

- src/rsb/patterns/RemoteServer.{h,cpp}: added typedefs for mutex and lock types in WaitingEventHandler; changed mutex to recursive\_mutex in WaitingEventHandler; use rsb:reply in WaitingEventHandler::handle; do not send ServerRequestId user-item in RemoteServer::callMethod; use rsb:error? user-item; send event before storing the method call
- src/rsb/patterns/Server.cpp: use event id instead of user-item ServerRequestId to identify method calls in RequestHandler::handle; changed user-item isException -> rsb:error?

##### Revision 2b264b3e - 08/03/2011 10:17 AM - J. Moringen

Fixed incorrect variable reference in rsb/patterns/\_\_init\_\_.py

refs #458

- rsb/patterns/\_\_init\_\_.py: fixed incorrect variable reference in

**Revision 2567a474 - 08/03/2011 10:29 AM - J. Moringen**

Removed leftover debugging stuff in src/rsb/patterns/RemoteServer.cpp

refs #458

- src/rsb/patterns/RemoteServer.cpp: removed leftover debug output in WaitingEventHandler::handle

**Revision 0118b547 - 08/03/2011 11:35 AM - J. Moringen**

Adapted client/server implementation to changed spec in src/patterns/

refs #458

- src/patterns/variables.lisp: new file
  - (**local-call**): new special variable; used to detect direct method calls within a single thread
- src/patterns/remote-server.lisp
  - (setf %method-listener :after t remote-method): exploit '\*' if possible
  - (call t remote-method event): do generate a UUID for the call, but let the remote server use the event id in its reply; bind
  - local-call**: store the call after sending the event; change user-item key isException -> rsb:error?
- src/patterns/local-server.lisp (call :around t local-method event): removed; method calls do not use the checked user-item anymore
- (call t local-method event): use event id to identify request instead of dedicated user-item ServerRequestId; exploit
- local-call**, if possible
- cl-rsb.asd (system cl-rsb): added file src/patterns/variables.lisp; added dependencies of src/patterns/{local,remote}-server.lisp on src/patterns/variables.lisp

**Revision 67e2c7ab - 08/03/2011 12:05 PM - J. Moringen**

Fixed certain keywords in src/transport/spread/conversion.lisp

refs #458

- src/transport/spread/conversion.lisp (**keyword-readtable**): new variable; contains a readtable for converting between keywords and strings
- (keyword->bytes): use **keyword-readtable**; use symbol names containing ':' unmodified instead of printing them with the special readtable
- (bytes->keywords): use **keyword-readtable**; don't choke on strings containing ':'

---

## History

### #1 - 08/02/2011 08:02 PM - J. Moringen

- Description updated
- Status changed from Feedback to In Progress

### #2 - 08/03/2011 04:28 AM - J. Moringen

- Description updated

- % Done changed from 0 to 10

**#3 - 08/03/2011 10:04 PM - J. Moringen**

- Description updated

- % Done changed from 10 to 80

**#4 - 08/05/2011 03:46 AM - J. Moringen**

- Description updated

- Status changed from In Progress to Resolved

- % Done changed from 80 to 100