

RSBag - Bug #867

Include IDL in channel format field

02/08/2012 07:39 PM - I. Lütkebohle

Status:	Resolved	Start date:	02/08/2012
Priority:	High	Due date:	
Assignee:	J. Moringen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	0.6		

Description

One of the most important aspects of the TIDE format is that the channel description includes a copy of the IDL of the logged data. This is essential to ensure future readability of log files.

At the moment, rsbag does not record this data. I understand this is due to an issue with acquiring it at runtime, but was also under the impression that a solution for that is now available with RST.

I would be very happy if you could look into a solution for this as soon as possible.

Related issues:

Related to RSBag - Feature # 735: Newer tool versions should be able to read ...	Resolved	11/25/2011	
Related to RSBag - Feature # 542: bag-info should display channel meta-data	Resolved	09/03/2011	
Related to RSBag - Feature # 875: Newer tool versions should be able to write...	In Progress	02/14/2012	
Related to RSBag - Bug # 923: Computation of channel format is a horrible hack	New	03/04/2012	
Blocks Robotics Service Bus - Tasks # 920: Release version 0.6	Closed	03/02/2012	03/05/2012

Associated revisions

Revision 6986a814 - 03/04/2012 10:07 PM - J. Moringen

Added transform-format in src/transform/{protocol,rsb-event}.lisp
refs #867

- src/transform/rsb-event.lisp (transform-format rsb-event): new method; return Protocol Buffer IDL descriptions for the event and, if possible, the payload; this is quite some hack
- src/transform/protocol.lisp (transform-format): new generic function; return an object describing the serialization format implemented by a transform
- src/transform/package.lisp (package rsbag.transform): added exported symbol transform-format

Revision 2000165a - 03/04/2012 10:07 PM - J. Moringen

Changed channel-format-for in src/rsb/{protocol,channel-strategies}.lisp
refs #867

- src/rsb/protocol.lisp (channel-format-for): added transform parameter
(channel-format-for t eql nil t t): added eql nil specializer
(channel-format-for t t t t): new method; default behavior for non-nil transform is to ask transform for format
- src/rsb/channel-strategies.lisp
(make-channel-for channel-connection event scope-and-type): pass transform to `channel-format-for`

Revision 15ef3daa - 03/04/2012 10:07 PM - J. Moringen

Handle IDL-related commandline options in bag-record/main.lisp

fixes #867

- bag-record/main.lisp (update-synopsis): add IDL options via
`make-idl-options`
- (main): call `process-idl-options`

Revision 036901ad - 03/05/2012 01:27 AM - J. Moringen

Moved IDL options to common.rst from logger.rst

refs #867

- manual/common.rst: added IDL options
- manual/logger.rst: moved IDL options to manual/common.rst

Revision d48b5106 - 03/05/2012 01:39 AM - J. Moringen

Added IDL-commandline options in bag-{cat,record}.rst

refs #867

- bag-cat.rst: added reference to IDL-related options in RSB
- bag-record.rst: likewise

Revision ca4d0568 - 03/05/2012 01:51 AM - J. Moringen

Printing of format in bag-info/main.lisp

refs #867

- bag-info/main.lisp (header): updated copyright
- (update-synopsis): added print-format commandline option
- (main): print format, if requested

History

#1 - 02/08/2012 11:06 PM - J. Moringen

- *Priority changed from Normal to High*

#2 - 02/23/2012 03:41 PM - J. Moringen

To achieve the requested behavior in the upcoming release, our current plan is to

1. Extend the TIDELog specification such that multiple IDLs can be stored in the CHAN.format field by making it a list of blobs instead of single blob.

We would submit this modification for inclusion in future versions of the TIDELog specification.

2. Implement filesystem-based lookup and reading of IDL in bag-record.

@Ingo: Is 1) OK for you or do you have different plans?

#3 - 02/23/2012 03:41 PM - J. Moringen

- *Target version set to 0.6*

#4 - 02/23/2012 04:12 PM - I. Lütkebohle

The expectation in the definition of the "format" field is that can you simply drop it, as-is, into your chosen serialization library, and have it parse the corresponding messages. The reason for this is that it makes the interaction with the serialization library unidirectional.

If you instead make it into a list on the TIDE level, this will result in bidirectional interaction with the serialization library, because it will need to go back to the TIDE log to look up further type information. This requires extensibility on the side of the serialization library that may be missing.

Therefore, I wouldn't recommend it. My suggestion would be to have an internal structure within the format field that allows including multiple message definitions. This structure may be specific to your serialization format.

#5 - 02/24/2012 01:39 PM - S. Wrede

Thanks for the clarification. We also thought already about inventing a micro-format to represent the respective information but wondered if that would be in the spirit of the log format. That said, we we will try to come up with an initial solution to this as part of the next RSB release `rsb:version:"0.6"`.

#6 - 02/24/2012 04:56 PM - J. Moringen

- Assignee set to J. Moringen

#7 - 03/02/2012 09:49 PM - J. Moringen

- Status changed from New to In Progress

- % Done changed from 0 to 20

#8 - 03/04/2012 05:01 PM - J. Moringen

- % Done changed from 20 to 70

We will address this issue only very basically for the 0.6 release:

- Users will have to load IDLs into bag-record manually
(via `--idl-path` and `--load-idl` as with the Common Lisp logger)
- We will store two textual Protocol Buffer definitions in the format field of each channel:
 1. `rsb.protocol.Notification` and its transitive dependency closure
 2. Payload type and its transitive dependency closure, if possibleThis can fail (possibly generating a warning, but continuing the recording) if
 - The payload type is not a protocol buffer type
 - The corresponding IDL has not been loaded
 - The corresponding IDL version does not match the actual payload**this problem will not be detected**
- The two textual definitions are separated by a `:` character
- The stored IDL information will not be used by any tools, for now

#9 - 03/04/2012 06:03 PM - I. Lütkebohle

Regarding detection of a mismatch between IDL and what's being used: Is there a checksum field in the message which could be checked against? ROS and LCM have md5 sums of the idl for this.

#10 - 03/04/2012 08:46 PM - J. Moringen

Currently, RSB events contain a "wire-schema" string which designates a payload serialization. However, this method is insufficient for two things:

1. Distinguishing serialization mechanisms (e.g. protocol buffer serialization vs. something else)
2. Verifying that a given payload has actually been serialized by means of the specified serialization mechanism

#11 - 03/04/2012 10:10 PM - J. Moringen

- Status changed from In Progress to Resolved

- % Done changed from 70 to 100

Applied in changeset r345.

#12 - 07/20/2012 12:57 PM - J. Moringen

- *Subject changed from Include IDL in Channel Format Field to Include IDL in channel format field*