

RSB GStreamer Integration - Bug #744

create timestamps are wrong

12/02/2011 02:33 PM - J. Wienke

Status:	Resolved	Start date:	12/02/2011
Priority:	Normal	Due date:	
Assignee:	J. Moringen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
For the 0.5 branch they are off by 3 missing zeros... Did I miss a commit while merging that back to the branch?			

Associated revisions

Revision 9bb40671 - 12/05/2011 11:58 AM - J. Moringen

Handle time carefully in src/gst_rsb_sink_base.cpp

refs #744

- src/gst_rsb_sink_base.cpp: handle integers representing microsecond time more carefully with respect to overflows; there may have been a problem on 32-bit systems

History

#1 - 12/04/2011 01:30 PM - J. Moringen

- Status changed from New to Feedback

Which element (i.e. rsbaudiosink, rsbvideosink, ...) creates the incorrect timestamps? Where can these be observed? In the GStreamer pipeline or in RSB events? Please provide additional information. If possible, include a recipe for reproducing the observed problem.

#2 - 12/05/2011 10:53 AM - J. Wienke

rsbvideosink was the problem running on a Nao with the new SDK 1.12. I just tried to reproduce the problem on my local machine and wasn't successful. One difference I see is that Nao is a 32 bit system. Could this problem be a result of an overflow while handling with the quite high microseconds numbers?

#3 - 12/05/2011 10:54 AM - J. Wienke

I could observe the wrong timestamps in the RSB logger.

#4 - 12/12/2011 10:51 PM - J. Moringen

Any new insights for this issue? Supposing the problem was fixed by r52, do we have to backport the fix to the 0.5 branch?

#5 - 12/13/2011 03:41 PM - J. Wienke

- Status changed from Feedback to Resolved

- % Done changed from 0 to 100

Looks fixed. Right now we used trunk. No idea if it is worth backporting this?

#6 - 12/13/2011 04:45 PM - J. Moringen

Probably not, if you use trunk anyway.

#7 - 01/31/2012 12:16 PM - S. Wrede

- *Target version deleted (0.5)*