RSBag - Bug #1426
remote-controlled strategy does not respect -E command line argument

02/19/2013 06:09 PM - J. Wienke

Status:	Resolved	**Start date:**	02/19/2013
Priority:	Normal	**Due date:**	
Assignee:	J. Moringen	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	rsb-0.9		

Description

when setting -E on the command line, the length method of the remote-controlled still returns the full play length. I suspect the same also applies to -S

0.7 and master

Associated revisions

Revision 1062eaa8 - 02/19/2013 08:54 PM - J. Moringen

Minor fixes in src/rsb/replay/external-driver-mixin.lisp

refs #1426

- src/rsb/replay/external-driver-mixin.lisp
  (make-commands external-driver-mixin sequence): made primitive
  function parameters required; simplified
  (replay replay-bag-connection external-driver-mixin): use
  start-index when checking for iteration end in backward direction;
  fixed `end?' call in `step*'; produce better error message in `step*'

Revision 804184d9 - 02/19/2013 08:55 PM - J. Moringen

Relative length/index commands in src/rsb/replay/external-driver-mixin.lisp

fixes #1426

- src/rsb/replay/external-driver-mixin.lisp
  (make-commands external-driver-mixin sequence): added length keyword
  parameter; added command "relativelength" and "relativeindex"
  (replay replay-bag-connection external-driver-mixin): added
  `length*' primitive operation; pass to `make-commands'; change
  `index' primitive operation to compute either absolute or relative
  indices

Revision 9be448c9 - 02/20/2013 12:00 PM - J. Moringen

Backport: Minor fixes in src/rsb/replay/external-driver-mixin.lisp

refs #1426

- src/rsb/replay/external-driver-mixin.lisp
  (make-commands external-driver-mixin sequence): made primitive
  function parameters required; simplified
  (replay replay-bag-connection external-driver-mixin): use
  start-index when checking for iteration end in backward direction;
  fixed `end?' call in `step*'; produce better error message in `step*'

Revision a0d7a214 - 02/20/2013 12:01 PM - J. Moringen

Backport: Relative length/index commands in src/rsb/replay/external-driver-mixin.lisp

refs #1426

- src/rsb/replay/external-driver-mixin.lisp
  (make-commands external-driver-mixin sequence): added length keyword
  parameter; added command "relativelength" and "relativeindex"
  (replay replay-bag-connection external-driver-mixin): added
  `length*' primitive operation; pass to `make-commands'; change
  `index' primitive operation to compute either absolute or relative
  indices

Revision 7c557f22 - 02/20/2013 12:32 PM - J. Moringen

Added relative{length,index} in bag-play.rst

refs #1426

- bag-play.rst: added relativelength and relativeindex functions

Revision 18f6fe92 - 02/20/2013 12:33 PM - J. Moringen

Updated documentation of remote-controlled in src/rsb/replay/remote-controlled.lisp

refs #1426

- src/rsb/replay/remote-controlled.lisp (remote-controlled): added
  descriptions or relativelength and relativeindex to the
  documentation string

History

#1 - 02/19/2013 08:10 PM - J. Moringen

- Status changed from New to In Progress

The remote-controlled and interactive strategies do respect -S and -E: seek or emitandnext signal an error when trying to "leave" the restricted sequence of events. However, the functions index and length refer to the unrestricted sequence of events.

I will add new functions indexrelative and lengthrelative which respectively calculate the current index and sequence length with respect to the restricted sequence of events. Will this be sufficient?

#2 - 02/19/2013 08:22 PM - J. Wienke

Actually I do not see much sense in letting length return something other than the restriction and seek being constantly in an erroneous state outside the specified bounds. Are there any use cases? Why would I want to know a length which I cannot access?

#3 - 02/19/2013 08:37 PM - J. Moringen

> *Actually I do not see much sense in letting length return something other than the restriction*

I would rather not have length behave differently depending on wheter there happens to be a restriction of the playback range or not.

However, adding a new family of functions with this behavior is not a problem.

> *and seek being constantly in an erroneous state outside the specified bounds.*

When seek signals an error, the iteration state remains valid and seeking to an index outside the specified bounds should be impossible.

> *Are there any use cases? Why would I want to know a length which I cannot access?*

One example are the RSBag user interfaces (the current, the upcoming one and a graphical one I once hacked): these display the current index, the restriction (if any) and the whole sequence.

#4 - 02/20/2013 09:53 AM - J. Moringen

- Status changed from In Progress to Resolved

- % Done changed from 0 to 100

Applied in changeset rsbag-cl|commit:804184d93e0ec53874ea2737ce096af7ca672258.