

## CoCoA-5 - Feature #328

### Swap

05 Mar 2013 21:39 - John Abbott

<b>Status:</b>	Closed	<b>Start date:</b>	05 Mar 2013
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	John Abbott	<b>% Done:</b>	100%
<b>Category:</b>	CoCoA-5 function: new	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	CoCoA-5.0.9	<b>Spent time:</b>	1.25 hour
<b>Description</b>			
I think C5 should have a <b>swap</b> procedure.			
Presumably it could simply be			
<pre>Define swap(ref A, ref B)   tmp := A;   A := B;   B := tmp; EndDefine;</pre>			
Perhaps an internal impl could avoid actually making a copies of the values.			

### History

#### #1 - 24 May 2013 14:49 - John Abbott

- Target version changed from CoCoA-5.0.3 to CoCoA-5.0.9

#### #2 - 24 Feb 2014 18:05 - John Abbott

- Status changed from New to In Progress

- Assignee set to John Abbott

Added first basic impl to coclib.cpkg5.

It should be possible to swap most types of value in C++ without actually copying them! I'll do this soon (maybe?)

Added manual entry too.

#### #3 - 25 Feb 2014 10:58 - John Abbott

- % Done changed from 0 to 20

**Giovanni** any good hints on how I could implement an efficient swap function in C++ (rather than doing 3 assignments in the interpreter)?

#### #4 - 25 Feb 2014 14:31 - Giovanni Lagorio

If I remember correctly, the swap above should not make any useless copy (that is, objects should be shared until they are modified). I don't remember the details of the implementation, in C++ you could probably just swap a couple of pointers, but it shouldn't be noticeably faster than the 3 assignments above (unless you call swap a gazillion times in a row ;-)).

**#5 - 26 Feb 2014 11:18 - John Abbott**

- *Status changed from In Progress to Feedback*

- *% Done changed from 20 to 80*

I tried swapping two long lists (10000000 entries), and it was pretty quick; so I deduce that the values are not copied -- thanks for the tip-off, Giovanni.

**#6 - 25 Mar 2014 15:11 - John Abbott**

- *Status changed from Feedback to Closed*

- *% Done changed from 80 to 100*

No trouble after 1 month in feedback.

Minor revision to the documentation.

Closing.