

## CoCoA-5 - Feature #1364

### Emacs UI: start cocoa5 with system command enabled

15 Nov 2019 15:30 - John Abbott

<b>Status:</b>	Closed	<b>Start date:</b>	15 Nov 2019
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	John Abbott	<b>% Done:</b>	100%
<b>Category:</b>	EmacsUI	<b>Estimated time:</b>	1.66 hour
<b>Target version:</b>	CoCoA-5.3.0	<b>Spent time:</b>	1.65 hour
<b>Description</b>			
Ange wants to use SystemCommand inside some of her code, but when she starts CoCoA-5 from inside emacs SystemCommand is deactivated.			
Add a way of starting CoCoA-5 with SystemCommand activated.			

#### History

##### #1 - 15 Nov 2019 15:40 - John Abbott

Ange's situation is that she has the source code in file.cocoa5 and wants to send it to CoCoA-5 with SystemCommand enabled. She is used to using "source file to (C-c C-f).

Here are two possible ways to implement:

- **(A)** new emacs commands *e.g.* **cocoa5-with-system-cmd**
- **(B)** a global emacs setting (flag?) which says whether SystemCommand should be enabled (default is not)

A problem with **(A)** is that various actions can cause a new CoCoA-5 session to be started, and each such action would need to know whether to start with or without enabling SystemCommand. Approach **(B)** would make this easier.

I expect Ange (and perhaps others who need SystemCommand) would like emacs to remember for all future sessions to enable SystemCommand; so if we use a global setting/flag this should be stored in .emacs (name may depend on platform?)

A GUI emacs has a drop down menu under "Options", and then "Customize emacs"... we could use that mechanism, but also need something for non-GUI emacs. Perhaps **cocoa5-enable-system-cmd**, and maybe **cocoa5-disable-system-cmd**?

##### #2 - 18 Nov 2019 10:51 - Anna Maria Bigatti

John Abbott wrote:

Here are two possible ways to implement:

- **(A)** new emacs commands *e.g.* **cocoa5-with-system-cmd**
- **(B)** a global emacs setting (flag?) which says whether SystemCommand should be enabled (default is not)

A problem with **(A)** is that various actions can cause a new CoCoA-5 session to be started, and each such action would need to know whether to start with or without enabling SystemCommand. Approach **(B)** would make this easier.

I expect Ange (and perhaps others who need SystemCommand) would like emacs to remember for all future sessions to enable SystemCommand; so if we use a global setting/flag this should be stored in .emacs (name may depend on platform?)

A GUI emacs has a drop down menu under "Options", and then "Customize emacs"... we could use that mechanism, but also need something

for non-GUI emacs. Perhaps `cocoa5-enable-system-cmd`, and maybe `cocoa5-disable-system-cmd`?

I vote for **B**. We have similar options: look in the CoCoA menu for **Preferences**. So it's very natural to do it (and copy it) the same way.

**#3 - 27 Nov 2019 22:42 - John Abbott**

- *Status changed from New to In Progress*
- *% Done changed from 0 to 40*

Prototype sent to Anna via email, so she can test it.

**#4 - 03 Dec 2019 14:36 - John Abbott**

- *Status changed from In Progress to Feedback*
- *Assignee set to John Abbott*
- *% Done changed from 40 to 90*
- *Estimated time set to 1.66 h*

I have checked in a cleaned-up version of the code... Anna said it worked fine for her.

Now we can test for it for a couple of months before making the "vietnamese" release.

**#5 - 13 Feb 2020 16:37 - John Abbott**

- *Status changed from Feedback to Closed*
- *% Done changed from 90 to 100*

The option is available under the CoCoA-5 drop down menu. It remains active only the current emacs session: if you close emacs, the flag returns to the default "off" setting.

Ange seems happy enough with it. Closing.