CoCoA-5 - Design #1453

Use symbolic links for release files

30 Apr 2020 15:05 - John Abbott

Status:	New	**Start date:**	30 Apr 2020
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Release	**Estimated time:**	0.00 hour
Target version:	CoCoA-5.4.2	**Spent time:**	0.70 hour

Description

It would be better for release files to be in fixed place, whose path name does not change with version number.
Currently the path contains a component like **CoCoA-5.3** which must be changed when we make version 5.4, etc.

I have tried to do this for CoCoALib, and it seems to work.

Work out how to do this for CoCoA-5 too.

Related issues:

Related to CoCoA-5 - Design #990: CoCoA-5 distribution: tidying	**Closed**	**05 Dec 2016**
Related to CoCoA-5 - Support #322: Installation instructions (on website)	**In Progress**	**26 Feb 2013**

History

#1 - 30 Apr 2020 15:06 - John Abbott

- Related to Design #990: CoCoA-5 distribution: tidying added

#2 - 30 Apr 2020 15:08 - John Abbott

Here is an excerpt from comment 13 of issue #990:

Here is a quick summary of what I think the installation "script" for CoCoALib does:

- create new dir with name like /usr/local/include/CoCoA-0.99700 and put the header files in there
- create a symbolic link so that /usr/local/include/CoCoA actually "points to" the directory created above
- put the library in /usr/local/lib/libcocoa-0.99700.a
- create a symbolic link so that /usr/local/lib/libcocoa.a "points to" the latest library

An advantage of this approach is that if installation goes wrong (but not horribly wrong, like when I wiped out half of /usr), then a "skilled" use can simply move the symbolic links to point back to the previous installation. Note that the previous installation is not removed.
I think doing something similar could be reasonable for CoCoA-5 too.

- all files for the distribution are put into cocoa-5.3.0
- then we make a symbolic link so that cocoa-5 (or even just cocoa???) points to the latest version
- any older installations are not removed (but perhaps an informative message about them could be printed?)

Does this imply that the user must run a script to "install" CoCoA-5: the script would make the appropriate symbolic links.

It may be difficult to do this on Microsoft?

#3 - 30 Apr 2020 16:00 - Anna Maria Bigatti

John Abbott wrote:

> I may be difficult to do this on Microsoft?

This is the point: I have no clue how to make a script in M$

#4 - 30 Apr 2020 17:09 - John Abbott

We can do it anyway for Linux and MacOS. Hopefully someone will tell us how to do it for Microsoft.

A first idea for a possible design is to have a structure like the following (similar to what I have tried to do for CoCoALib):

- the root dir is **/usr/local/bin/CoCoA-5/** or maybe just CoCoA instead of CoCoA-5???
- inside root dir each version unpacks into **ROOTDIR/CoCoA-5.3.2/** say (what to do if this subdir already exists? error? flag to force?)
- make a symbolic link from **ROOTDIR/CurrentVersion** to the just unpacked dir
- this means that all files relevant to CoCoA-5 are in **ROOTDIR/CurrentVersion/**

If an experienced user wants to switch back to an earlier version then it suffices to change just the symlink.

A similar structure can also be built in a personal file-space (so there would be no need to be a sudoer).

#5 - 30 Apr 2020 17:28 - Anna Maria Bigatti

Just to remember: we already have pretty good scripts for configuration on Mac and on Linux.

#6 - 12 May 2020 14:30 - John Abbott

- Related to Support #322: Installation instructions (on website) added

#7 - 12 May 2020 14:32 - John Abbott

On MacOS a user can install software easily from a "dpkg" file.
It would be nice to have a single-file-install capability for Linux.

One possibility is a "shar" file. Not sure how well that fits with a large TARGZ file.
How standard is gzip?

UPDATE: Wikipedia suggests that "shar" files are too risky... but anyway a helpful installation script has to be executable... mmm?

#8 - 03 Feb 2022 19:39 - John Abbott

- Target version changed from CoCoA-5.4.0 to CoCoA-5.4.2