CoCoALib - Bug #1600

Detect updated versions of external libs

14 Jun 2021 16:19 - John Abbott

Status:	In Progress	**Start date:**	14 Jun 2021
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	10%
Category:	Safety	**Estimated time:**	0.00 hour
Target version:	CoCoALib-0.99880	**Spent time:**	0.55 hour

Description

Florian Walsh reported a compilation failure after updating BOOST on his (linux) system.
My understanding is that he used the package manager to update BOOST, but did not then re-run configure for CoCoALib.
This triggered a compilation failure.

Find a way to detect updated extlibs, and tell the user to reconfigure. (or somehow warn the user).

Related issues:

Related to CoCoALib - Bug #1601: Compilation ambiguity	**Closed**	**16 Jun 2021**
Related to CoCoALib - Support #1700: boost_1_80_0	**Closed**	**14 Oct 2022**
Related to CoCoA-5 - Design #1696: Which BOOST libs are actually needed?	**Closed**	**18 Aug 2022**

History

#1 - 14 Jun 2021 16:24 - John Abbott

Some ideas and comments (for checks to make whenever trying to compile):

- first idea: check whether some extlib header file is more recent than autoconf.mk; **defect:** how to find the path to the header file
- next idea: have a script (for each extlib?) which determines the version; *e.g.* compile a file in /tmp and see what it prints out; **advantage:** we should have the necessary compilation flags to hand

Suggest for KISS: make these checks only from the main Makefile (at least for the first attempt at resolving)

#2 - 16 Jun 2021 16:28 - John Abbott

- Related to Bug #1601: Compilation ambiguity added

#3 - 23 Jun 2021 15:17 - John Abbott

- Target version changed from CoCoALib-0.99800 to CoCoALib-0.99850

#4 - 12 Nov 2022 13:54 - John Abbott

- Related to Support #1700: boost_1_80_0 added

#5 - 12 Nov 2022 13:55 - John Abbott

Is this a duplicate of issue #1700?

#6 - 23 Nov 2022 17:39 - John Abbott

- Status changed from New to In Progress

- % Done changed from 0 to 10

Here is a copy of #1700#note-5 (which I closed because it duplicates this issue).

I wonder if there is a simple away to warn users of potential problems?

A similar problem could arise with any of the external libs, and the solution is no doubt to rebuild dependencies -- and this should normally be done by re-running configure (which may include some other "sanity checks" for external libs).

Here is an idea:
for each ext lib configure computes a check-sum for some header file in that ext lib, and saves the result in autoconf.mk. Then make can verify that the check-sums are correct -- if not, a warning with a delay, can be produced before proceeding with compilation.
[the delay gives the user a chance to hit Ctrl-C]

#7 - 30 Nov 2022 17:17 - John Abbott

- Related to Design #1696: Which BOOST libs are actually needed? added

#8 - 16 Feb 2024 10:02 - John Abbott

- Target version changed from CoCoALib-0.99850 to CoCoALib-0.99880