CoCoALib - Design #1801

BuildInfo

25 Mar 2024 18:47 - John Abbott

Status:	In Progress		Start date:	25 Mar 2024
Priority:	Normal		Due date:	
Assignee:			% Done:	10%
Category:	Tidying		Estimated time:	0.00 hour
Target version:	CoCoALib-0.99880		Spent time:	0.85 hour

Description

Three things:

- why do some preprocessor flags have prefix **COCOA_** while others have **CoCoA_** ?
- for whom are the messages printed by BuildInfo intended?
- would it be more sensible/coherent for BuildInfo just to print out the name of the macro followed by its value?

Related issues:

Related to CoCoALib - Bug #1661: Microsoft: cannot compile with signal handling	**Closed**	**09 Feb 2022**

History

#1 - 25 Mar 2024 18:49 - John Abbott

We need to consider the behaviour both in CoCoALib and in CoCoA-5.

Anna suggests that it is more helpful to maintainers/developers if the name of the preprocessor macro is given explicitly in the message; she suggests that the messages are not really intended for the "casual user" (other than to report a bug).

#2 - 25 Mar 2024 18:51 - Anna Maria Bigatti

- % Done changed from 0 to 10

The function for CoCoA-5 is VersionInfo() defined in BuiltingFunctions.C

#3 - 25 Mar 2024 18:52 - Anna Maria Bigatti

- Related to Bug #1661: Microsoft: cannot compile with signal handling added

#4 - 13 Apr 2024 22:18 - John Abbott

- Status changed from New to In Progress

Here is a list of CPP symbols beginning with **COCOA_**

- COCOA_ULONG2LONG
- COCOA_ULONGLONG2LONGLONG
- COCOA5_VERSION, COCOA5_VER_MAJ, COCOA5_VER_MIN
- COCOA_VERSION
- COCOA_CXX, COCOA_CXXFLAGS, COCOA_PLATFORM

I notice that all of these come essentially via the configuration script (and have a dual role also as shell variables)