CoCoA-5 - Feature #19

How to get the version number of CoCoA-5?

04 Nov 2011 11:51 - Anna Maria Bigatti

Status:	Closed	Start date:	04 Nov 2011
Priority:	High	Due date:	
Assignee:	Anna Maria Bigatti	% Done:	100%
Category:	CoCoA-4 function to be added	Estimated time:	0.00 hour
Target version:	CoCoA-5.0.3	Spent time:	7.80 hours

Description			
How should we encode the version number of CoCoA-5? Should it be in the packages? Probably yes, together with a function to get the version number of CoCoALib.			
Related issues:			
Related to CoCoA-5 - Slug #191: Slow lex gbasis		**Closed**	**19 Jun 2012**
Related to CoCoA-5 - Feature #20: How to get the compilation flags of CoCoA-5?		**Closed**	**04 Nov 2011**

History

#1 - 04 Nov 2011 11:58 - Anna Maria Bigatti

- Project changed from CoCoA to CoCoA-5

#2 - 08 Nov 2011 15:35 - Anna Maria Bigatti

- Category set to CoCoA-4 function to be added

#3 - 08 Jun 2012 18:28 - Anna Maria Bigatti

- Status changed from New to In Progress

Something we had talked about:
cocoa-4 used to load the file **init.coc**.
I've found a *tappullo*: call this file **zzz-init.cpkg5** in **packages**/.
(should we make something cleaner? ;-)

In this init file we could write the version number (and "Use R", and the banner, ...).

- disadvantages: I think the version number should be in the executable
- advantages: we used to forget to update the version number in the C code before making (and asking to make) all the compilations (nightmare!!)

#4 - 08 Jun 2012 19:03 - John Abbott

Anna Maria Bigatti wrote:

- disadvantages: I think the version number should be in the executable
- advantages: we used to forget to update the version number in the C code before making (and asking to make) all the compilations (nightmare!!)

I agree that the version number should be in the executable.

If the version number is in its own file then recompilation should not be too costly (just recompile one tiny file, and relink).

I suggest that the version be represented as a string (at least initially).
We could perhaps make **version** return a record which initially contains just one field (being the version as a string)...

#5 - 11 Jun 2012 12:03 - Anna Maria Bigatti

John Abbott wrote:

> I agree that the version number should be in the executable.
>
> If the version number is in its own file then recompilation should not be too costly (just recompile one tiny file, and relink).

It is not really a matter of being costly... it is a matter of needing to recompile on several machines... but we can try.
I admit that, in case something goes wrong with pacakges, it would be nice to be able to get the version number.

#6 - 11 Jun 2012 13:42 - John Abbott

- % Done changed from 0 to 20

Strictly, each package should have its own version number (maybe they already do in the source text?) I'm not sure it would be necessary to have a function which gives the version number for each package -- anyway, we're hoping that packages will eventually become less important...

What should the function which gives the version be called? **version** or maybe **CoCoAVersion** or **CocoaVersion**?

#7 - 11 Jun 2012 14:35 - Anna Maria Bigatti

- Target version set to CoCoA-5.0.3

John Abbott wrote:

> Strictly, each package should have its own version number (maybe they already do in the source text?)

the information is is $package.About()

> What should the function which gives the version be called? **version** or maybe **CoCoAVersion** or **CocoaVersion**?

I suggest "version". Returning a Record with fields CoCoA5 and CoCoALib?

#8 - 11 Jun 2012 19:41 - John Abbott

Should the field be called **CoCoA5** or just simply **CoCoA**?

JAA prefers **CoCoA**. Its value will be a string, something like "5.0.3"

#9 - 12 Jun 2012 21:50 - Anna Maria Bigatti

John Abbott wrote:

> Should the field be called **CoCoA5** or just simply **CoCoA**?
> JAA prefers **CoCoA**. Its value will be a string, something like "5.0.3"

OK. I started working on it.

#10 - 13 Jun 2012 17:16 - Anna Maria Bigatti

- % Done changed from 20 to 30

I added the functions **CoCoAVersion()** and **CoCoALibVersion()**.
Before I cvs it all in, how should the "version" files be called?
Now they are called VersionNumber.[HC]

#11 - 13 Jun 2012 19:00 - Anna Maria Bigatti

I added a mechanism to make it simple to change the date in the manual.

So that **? CoCoAManVersion** prints the version of cocoa it was written for and the date.

These data are set manually in CoCoAHelp.xml so that they are an intrinsic characteristic of that very version of the file. Maybe something can be done automatically (e.g. taking the date of the file from the system), but I'm not sure it is wise.
(not yes cvs-ed)

#12 - 14 Jun 2012 14:24 - John Abbott

@Anna: could you explain your preference for two separate functions rather than a single function?

JAA has a (mild) preference for a single function because the version of the software that one is using is a combination of the version of CoCoALib and the version of the interpreter -- it is true that for *public releases* the version of CoCoA will uniquely identify the version of CoCoALib.

Another possible name could be **VersionInfo** this could contain not only the version numbers but also other information (*e.g.* external libraries which have been included?). We could also use **VersionInfo** as the name of the files.

#13 - 18 Jun 2012 10:58 - Anna Maria Bigatti

John Abbott wrote:

> @Anna: could you explain your preference for two separate functions rather than a single function?

One "difficulty" is that creating a record for cocoa-5 has to be done "by hand", but I agree that it makes more sense (so I'm doing it)

> Another possible name could be **VersionInfo** this could contain not only the version numbers but also other information (*e.g.* external libraries which have been included?). We could also use **VersionInfo** as the name of the files.

ok. so far it only gives the versions, then we'll see.

#14 - 19 Jun 2012 17:10 - Anna Maria Bigatti

- Status changed from In Progress to Feedback

- Assignee set to Anna Maria Bigatti

- % Done changed from 30 to 90

added VersionInfo.
fixed documentation.

This could also return the other info in BuildInfo (such as compilation flags..) but these are not publicly available (whereas BuildInfo::version is)

Should we make them public and make them accessible by CoCoA-5? this could be useful to know if it compiled with debugging or optimization! ;-)

#15 - 20 Jun 2012 14:11 - Anna Maria Bigatti

How to get the (string) values out of **BuildInfo.C** ?
At the moment there is

```
    extern const char* const version;
```

Is it OK to have all data accessible via globals?

#16 - 21 Jun 2012 10:55 - Anna Maria Bigatti

see above

#17 - 21 Jun 2012 14:08 - John Abbott

I suggest removing the CoCoA prefix from the compilation date, the compiler name, and the compilation options.

I would prefer Bits instead of Digits though I know that the C++ STL calls them digits (but the user can also ask for the base of the representation system -- which is 2 in practically all instances). I wonder if a Machine prefix would make sense for these names (as the risk of making the names very long).

#18 - 21 Jun 2012 15:14 - Anna Maria Bigatti

```
/**/ indent(VersionInfo());
Record[
  CoCoALibVersion := "0.9951",
  CoCoAVersion := "5.0.3",
  CompilationDate := "Jun 21 2012 15:07:49",
  CompilationFlags := "-DCoCoA_ULONG2LONG=1 -Wall -pedantic -fPIC -m64 -mtune=core2 -march=core2 -DCoCoA_WITH_
FROBBY -DCoCoA_WITH_GSL -DCoCoA_WITH_NORMALIZ -O2",
  Compiler := "g++",
  MachineIntNumBits := 32,
  MachineLongNumBits := 64
]
```

Anything else?

#19 - 21 Jun 2012 16:14 - John Abbott

I think this latest proposal is a good start. It contains the most useful information, and the names of the fields are self-explanatory (without being excessively long). I won't be surprised if we decide later to add some more fields, but at the moment I cannot think of any which are obviously missing.

#20 - 05 Jul 2012 10:01 - Anna Maria Bigatti

This function should also return included external libraries and possibly their version

#21 - 02 Oct 2012 08:35 - Anna Maria Bigatti

Added field CompilationDefines
(used to be part of CompilationFlags)

#22 - 14 Nov 2012 11:34 - Anna Maria Bigatti

- Status changed from Feedback to Closed

- % Done changed from 90 to 100