

CoCoA-5 - Feature #1174

package of a function --> of an identifier!

05 Apr 2018 09:26 - Elisa Palezzato

Status:	Closed	Start date:	05 Apr 2018
Priority:	High	Due date:	
Assignee:	Anna Maria Bigatti	% Done:	100%
Category:	CoCoA-5 function: new	Estimated time:	0.00 hour
Target version:	CoCoA-5.4.0	Spent time:	2.75 hours
Description			
How to have a function that says in which package is in the input? Why does not describe have this feature?!			
2021-01 PackageOf("<name>"); describe cannot know the package, because it receives the evaluated object, and not its name/identifier.			
Related issues:			
Related to CoCoA-5 - Feature #346: Describe should indicate package name		Closed	29 Apr 2013
Related to CoCoA-5 - Bug #1200: Package exported function: wrong package name		Closed	27 Jul 2018
Related to CoCoA-5 - Feature #531: Package protected variables should know wh...		Closed	09 Apr 2014

History

#1 - 05 Apr 2018 13:43 - John Abbott

- Related to Feature #346: Describe should indicate package name added

#2 - 01 Aug 2018 09:37 - Anna Maria Bigatti

- Category set to CoCoA-5 function: new
- Assignee set to Anna Maria Bigatti
- Target version set to CoCoA-5.2.4

Waiting for a proper implementation in C++, we can do something like starting.

#3 - 01 Aug 2018 09:37 - Anna Maria Bigatti

- Related to Bug #1200: Package exported function: wrong package name added

#4 - 06 Aug 2018 16:17 - Anna Maria Bigatti

- Target version changed from CoCoA-5.2.4 to CoCoA-5.3.0

#5 - 01 Oct 2019 14:22 - John Abbott

- Priority changed from Normal to High
- Target version changed from CoCoA-5.3.0 to CoCoA-5.4.0

#6 - 04 Mar 2020 21:31 - John Abbott

As Anna said in comment 2: the function starting is able to get the information. So we should look there to see how to do it.

#7 - 11 Dec 2020 11:48 - John Abbott

Anna has a prototype in coclib.cpkg5. Currently commented out -- how long has it been there??

If it were built-in then it could be used by **describe**.

#8 - 11 Dec 2020 11:53 - Anna Maria Bigatti

- Status changed from New to In Progress

- % Done changed from 0 to 50

#9 - 15 Jan 2021 08:11 - Anna Maria Bigatti

- % Done changed from 50 to 60

should the name be PackageOf?

I have investigated: belonging to a package (variable) is a property of the variable (**VariableSlot**), not of the value of the variable.

Now I have written a function in Interpreter.C taking a string (the variable name).

There was a function called **findOwner** (now **FindOwnerPkg**) in Interpreter.C in anonymous name space. I used it.

Can I make PackageOf take a variable? will see

#10 - 15 Jan 2021 08:17 - Anna Maria Bigatti

- Related to Feature #531: Package protected variables should know which package protected them added

#11 - 15 Jan 2021 08:29 - Anna Maria Bigatti

Example:

```
/**/ anna := starting;
/**/ PackageOf("starting");
$coclib
/**/ PackageOf("anna");
<not in a package>
```

So, to have it working with the variable, not the string with the variable name, the argument should not be evaluated, as it happens for **IsDefined** (only case like that)

#12 - 15 Jan 2021 09:07 - John Abbott

Well done! I suggest we accept the version taking a string for the time being (unless you can easily make a fn like IsDefined).

When exactly is it useful to find the package? I think I use it only when debugging (or perhaps if the documentation seems unclear, and I want to see what the code actually does).

Other things which do not evaluate a variable are: describe, (un)protect.

#13 - 15 Jan 2021 12:24 - Anna Maria Bigatti

- *Subject changed from package of a function to package of a function --> of an identifier!*

#14 - 22 Jan 2021 08:31 - Anna Maria Bigatti

- *Description updated*

- *Status changed from In Progress to Feedback*

- *% Done changed from 60 to 90*

I'm now convinced that the only exception to the evaluation of the argument of a function should be `IsDefined`. There is no point in making an exception for such a small function as `PackageOf`.

So, in conclusion, the only great advantage of `PackageOf` vs `starting` is the good name ;-)
It is also faster, but it's pretty irrelevant for its typical use (i.e. not within loops).

#15 - 29 Jan 2021 15:43 - Anna Maria Bigatti

- *Status changed from Feedback to Closed*

- *% Done changed from 90 to 100*