

# Premia: Overview

## version 17

C. Martini, A.Zanette

1999/15/12

## Premia 17

### Contents

<b>1</b>	<b>What Premia is</b>	<b>1</b>
<b>2</b>	<b>What Premia is NOT</b>	<b>2</b>
<b>3</b>	<b>Components</b>	<b>2</b>
<b>4</b>	<b>Directory Tree</b>	<b>4</b>
<b>5</b>	<b>Documentation Usage</b>	<b>5</b>
<b>6</b>	<b>Software Usage</b>	<b>5</b>
6.1	Functionalities . . . . .	5
6.2	Processing of premia.out . . . . .	6
6.3	Getting a Dvi or Pdf file from premia.out . . . . .	6
6.4	Integration of the Pdf output file in the documentation . . . . .	7

## 1 What Premia is

The Premia project is undertaken at [I.N.R.I.A](#) and [C.E.R.M.I.C.S.](#) . Its main goal is to provide C routines for the pricing of financial derivative products altogether with a scientific documentation. It focuses rather on the implementation of numerical analysis techniques available to compute the quantities of interest than on the financial context. It is an attempt to

[Biblio](#)

[C Source](#)

keep track of the most recent advances in the field from a numerical point of view in a well-documented manner. The ultimate aim would be to assist the R&D professional teams in their day-to-day duty. It may also be useful for academics who wish to perform tests of a new algorithm or pricing method without starting from scratch.

The project is developed by the [Premia Team](#).

Even if we did our best, the way we choose to implement a given algorithm corresponding to a published article is under our sole responsibility and does not in any case commit the authors of this article. We also choose the names of our routines in a somewhat arbitrarily manner, most notably in case of multi-authors articles, therefore we apologize for the possible omissions of some author names.

## 2 What Premia is NOT

Premia is not a set of highly optimized routines for this or that platform where basic operations are reprogrammed the best way for a purpose or another. This optimization stage with the use of a low-level library module like NAG's has not been performed at our level. Only the high-level algorithms are tackled in Premia.

Premia is not either a powerful object-oriented system designed to handle any kind of derivative products and portfolios of such derivative products. Although this is a very interesting challenge, especially in the setting of non-linear pricing, it would require a heavy formalism which is not in the spirit of Premia. We focus on the numerical analysis oddities. Other oddities in the direction of complex systems and formal tools applied to the pricing of derivative products are very interesting to study- but not here. This could be, by the way, another project undertaken at [I.N.R.I.A](#) or/and [C.E.R.M.I.C.S.](#)

## 3 Components

The Premia project is three-fold:

The first component is a C-programmed system designed to describe easily derivative products, models, pricing methods which provides basic input/output functionalities. Also it is strongly object-oriented, this software is written in C language. The basic objects are Models, which describe both the stochastic model of the underlying of the option and the market parameters, Options, which describe the derivative product itself, and Pricing

Methods which describe the algorithms of pricing.

The second component is the pricing routines themselves. Each routine is written in a separate .c file. Such a file usually bears the routine name with a prefix which says to which family of algorithm the routine belongs. There are 5 prefixes: fd\_, tr\_, mc\_, cf\_, ap\_ which stand for finite-difference, tree (like the Cox-Ross-Rubinstein algorithm), Monte-Carlo, closed formula, approximation (like the Whaley or McMillan approximation method for the American Put).

The .c file contains the piece of code of the routine itself, for which there is nothing mandatory with regards to input or output parameters-this freedom allows to include easily preexisting stuff-and which is the part of interest for people who want to plug the routines of Premia in another software, and a second part which is devoted to the integration of the routine in the above system, that is to code the PricingMethod object which corresponds to the routine.

The last component is the scientific documentation system. It is made of a lot of hyperlinked PDF files which discuss either a pricing routine (every routine has its own PDF doc file) or a more general topic like Monte Carlo methods, Tree methods... This web of Pdf files also includes a Pdf version of the whole C source code with easy jumps from the source file to the documentation file.

According to us the most valuable component is the documentation which makes use-or tries to make use- of the scientific and numerical knowledge of our institutions. We hope this will complement in a pertaining way the outstanding books devoted to theoretical option pricing.

The routine themselves come in second. We feel that on a given pricing issue some professional R&D team will certainly have much better and competitive either software or even algorithm. Nevertheless on the average Premia should be of interest to them.

Lastly the little object-oriented software is only there to provide an easy way to test things. It was mainly designed for our usage of Premia. This way Premia is more attractive than a plain library of C routines.

## 4 Directory Tree

Here is the directory structure of Premia: the root directory here is the directory **Src** :

Structure du répertoire

```
/Premia/Src
+---common
|   /---math
+---mod
|   +---bs1d
|   |   +---bs1d_doublim
|   |   +---bs1d_lim
|   |   +---bs1d_limdisc
|   |   +---bs1d_pad
|   |   /---bs1d_std
|   /---bs2d
|       /---bs2d_std2d
\---opt
    +---doublim
    +---lim
    +---limdisc
    +---pad
    +---std
    /---std2d
```

The directory **Common** contains mainly the files which define and manipulate the model, option, and pricing method objects. The subdirectory **Math** contains mathematical tools as the Normal Law, random generators, mathematical constants and so on...

The directory **Opt** contains the files which code the options. Each option corresponds to one .c file. The options are grouped in families, each family corresponding to a subdirectory, like Standard options (subdirectory **Std** ), Path-Dependant options (**Pad** ).

The directory **Mod** contains the .c files which code the underlying model, one in each subdirectory (like **Bs1D** ), and also the pricing routines, which are grouped by families, for instance **Bs1D\_Std** contains all the pricing routines of Standard options in the Black-Scholesmodel.

## 5 Documentation Usage

All you have to do is to enter the hyperlinked system of files at some point. The normal entrance point is the file

[premia17\\_doc.pdf](#)

in the directory **Src** .

Another very useful file to navigate is an hyperlinked list of all the Pdf version of the C files of the project, this is

[summary\\_doc.pdf](#)

also in the directory **Src** .

## 6 Software Usage

### 6.1 Functionalities

Premia first prompts the user for either a Pricing session or a Dynamic Test session.

A Pricing session is plainly the computation of the price of a given derivative product given a model and a pricing method.

The user first chooses the model, then the option, lastly the pricing method. For every object, the default value of the parameters are displayed and the user has the possibility either to modify or to iterate any of the parameters. At most 2 parameters may be iterated simultaneously. Premia writes the routine output, ie the price (and also, when it is meaningful, systematically the delta and maybe other routine-specific quantities like a precision in a Monte-Carlo method) in an output file `premia.out`.

A nice variant of this scenario is the possibility to choose several methods for the same pair (model,option), in order to compare various PricingMethods.

A Dynamic Test session is the simulation of the behavior of a market-maker who sells the option at the pricing time and delta-hedges until maturity where he delivers the option payoff to the counterpart, or in case of American

options until the Exercice time chosen by the buyer.

The user chooses the model, the option and the pricing method but without the possibility to iterate the parameters (since a single run takes usually a long time). Then he chooses the parameters of a Dynamic Test object, like the number of hedges he shall perform during the life of the option, the number of paths of the underlying to be simulated, the Random Generator to use, in the ... Then Premia computes the Profit&Loss for each path and usually puts in the outputfile premia.out the paths corresponding to the higher and lower P&L and also the distribution of the P&L.

## 6.2 Processing of premia.out

To go further you need Gnuplot (freeware plotting utility which exist on many platforms) which you may download at [or](#) (Unix version) at [\(Win32 version\)](#). The software (ie premia.exe) itself does a little post-processing on premia.out which in particular creates a Gnuplot command file gnupremia.plt and a Gnuplot datafile gnupremia.dat. Then you just need to run: Gnuplot gnupremia.plt. The graphs should be nicely labeled. The number of graphs you get depend on the type of session and the number of output values of the PricingMethod or DynamicTest.

In case of a Pricing session with several PricingMethods all the output common to the methods are displayed on the same graph.

## 6.3 Getting a Dvi or Pdf file from premia.out

You need at this stage latex and dvipdfm (freeware available on the CTAN sites, included in the last version of MikTeX for Win32 also available on the CTAN sites). The output given by Gnuplot is a bit frustrating since the parameters which are not iterated appear nowhere on the plots. To get an output graph with all the information Premia also creates another gnuplot command file which makes auxiliary .tex files which contain the graphs as .tex data, and also a .tex file gnupremia.tex which contain these files as inputfiles. Then after running latex on gnupremia.tex you get a nice .dvi file and if you wish dvipdfm you get a nice pdf file gnupremia.pdf with all the information. Do not worry, there is a bash shell out2pdf.bash and (for Win32) a batch file out2pdf.bat which perform the whole process.

## 6.4 Integration of the Pdf output file in the documentation

One of the interest of getting a Pdf file is the possibility to integrate this file in the web documentation system of Premia. This is achieved thanks to another bash (or batch) file, `makearchive.bash`(`makearchive.bat`), which puts the Pdf file at the right place in Premia TDS, that is in the `PricingMethod` directory. Then you can make (hyper)reference to this file in the documentation, it is also automatically added in a list of trials relevant to this `PricingMethod`. More details about this functionality may be found in the Developer's guide to input/output [Interface](#)