

How to Install and Use MMALPHA Version 2.1

Tanguy Risset and Patrice Quinton
Revision of April 25, 2010

April 2010

1 How to install MMALPHA

This section contains MMAlpha installation instructions for Unix users, Windows NT or XP users, MacOSX and Linux users.¹

The MMALPHA software can be downloaded at url <http://www.irisa.fr/cosi/ALPHA/>. In case of problems, send a mail to patrice.quinton@irisa.fr .

1.1 Installing MMALPHA on Unix-like systems

By Unix-like, we mean Unix, Mac OS X, or Linux. The following procedure has been checked for Unix and Mac OS X, not for Linux.

1. This first action must be done only if MMALPHA has not yet been installed on the server or the computer you are using. If MMALPHA is already installed, go to step 2.

The current distribution, as obtained from the MMALPHA web site, is a file named `mmalphaV2-1-0.zip`. Double-click on this file to expand this directory. You may also run the following command

```
unzip mmalphaV2-1-0.zip
```

from a shell window.

2. Create a MMALPHA environment variable containing the path of the directory where MMALPHA has been installed. If you use a `csh` shell, you have to type:

```
setenv MMALPHA path-of-mmalpha
```

and for a `bash` shell:

¹The source of this document is in the `doc/Install` directory. It appears also as an appendix of the AlphaStart document.

```
export MMALPHA=path-of-mmalpha
```

These commands may alternatively be added to your `.cshrc` or to your `.bashrc` file (or to your `.login` file) in order to be executed automatically when starting a shell.

3. You have then to set the `PATH` environment variable to contain the directory where the binary files of MMALPHA are. On Mac OS X, this directory is `path-of-mmalpha/bin.darwin`. For a `csh` shell:

```
setenv PATH ${MMALPHA}/bin.darwin:${PATH}
```

and for a `bash` shell:

```
export PATH=${MMALPHA}/bin.darwin:${PATH}
```

Again, you may add this command to your `.cshrc` or `.bashrc` file.

4. Copy the file `$MMALPHA/config/init-for-you.m` under the name `init.m` to your Mathematica base directory² (or append it to your base `init.m` file if it already exists). For users unfamiliar with Mathematica, recall that this `init.m` file is executed whenever Mathematica's kernel is launched.
5. It's ready! Run Mathematica by typing in a shell window the command:

```
mathematica
```

and once a notebook is started, type and evaluate in it the command

```
start []
```

to launch the master notebook of MMALPHA. From this master notebook, examples and explanations are available.

6. To check that everything is OK, evaluate successively the following commands

```
test1 []  
test2 []  
test3 []  
test4 []
```

²The Mathematica base directory used to be the user home directory in Mathematica versions prior to version 5. Under Mathematica, the home directory is given by evaluating the `$HomeDirectory` Mathematica variable. Since version 5 (see documentation of Mathematica), the Mathematica base directory is given by the `$UserBaseDirectory` variable. For example, in MacOS X, it is located in `~/Library/Mathematica/Kernel` directory: this is where you should put the `init.m` file.

Each one of these command starts a set of tests and should return the value `True`. If this is not the case, something in your installation is wrong. In case of problems, see Section 1.3.

Notice that executing these commands generates a lot of error messages, and even, unexpected messages in the shell windows: as long as the final result of the test is `True`, this is not a problem.

Remarks

- On MacOS X, Mathematica should be started from the Terminal application, as otherwise, the `$MMALPHA` environment variable cannot be set (at least, I do not know how to set this variable...). Moreover, after installing Mathematica, it is needed to add an alias to start Mathematica. Usually, Mathematica is installed in directory `/Applications`, for example :

```
/Applications/Mathematica\ 5.1.app/Contents/MacOS/Mathematica
```

So, you may add an alias in your `.cshrc` file.

1.2 Installing MMALPHA on Windows NT or Windows XP

Warning: version V2 has not been tested on Windows NT, since I do not have currently access to such a configuration. PQ, Jan. 2, 2009.

1. This first action must be done only if MMALPHA has not yet been installed in the server you are using. If MMALPHA is already installed, go to step 2. The current distribution is a file `mmalphaV2-1-0.zip` file. Double-click on this file to expand this directory.
2. You have to set two environment variables: `MMALPHA` and `Path`. To do so:
 - (a) Open the system configuration panel (start -> configuration panel)
 - (b) Double click on the "System" icon.
 - (c) Choose the "Advance" panel.
 - (d) Click on "Environment variables".
 - (e) Consider the user variable panel (top part).
 - (f) Create a `MMALPHA` variable whose value is the path of the directory where MMALPHA is installed. Typically, the value of this variable should be

```
C:\...\mmalphaV2-1-0
```

To create such a variable, click on `New`, and fill the name and value. This step allows Mathematica to know where MMALPHA is located.

(g) Append

```
;%MMALPHA%\bin.cygwin32
```

to the user Path environment variable. To do so, select the Path variable, click Modify, place the cursor at the end of the string and type

```
;%MMALPHA%\bin.cygwin32
```

³This step allows the Domlib library to be launched.

3. Copy the file `$MMALPHA/init.m` to your Mathematica base directory⁴ (or append it to your base `init.m` file if it already exists). For users unfamiliar with Mathematica, recall that this `init.m` file is executed whenever Mathematica's kernel is launched.
4. It's ready! Start Mathematica. The first evaluation in the initial notebook should start MMALPHA. Normally, the Messages window of Mathematica opens and contains a few lines indicating that Mathematica was started successfully. Evaluate `start[]` in any notebook to launch the master notebook of Mathematica. From this master notebook, examples and explanations are available.
5. To check that everything is OK, evaluate successively the following commands

```
test1[]  
test2[]  
test3[]  
test4[]
```

Each one of these command starts a set of tests and should return the value `True`. If this is not the case, something in your installation is wrong. In case of problems, see Section 1.3. Notice that executing these commands generates a lot of error messages, and even, unexpected messages in the shell windows: as long as the final result of the test is `True`, this is not a problem.

1.3 In Case of Problems

Before reading this section, make sure that you have tried to evaluate the `test1[]` through `test4[]` commands.

There is a Mathematicaprogram called `diagnose.m`, located in `$MMAlpha/config/diagnose.m`, that may help locate problems. To run it, start a fresh version of MMALPHA (from the console window), then evaluate

³Or possibly, cygwin.

⁴The Mathematica base directory used to be the user home directory in Mathematica versions prior to version 5. Under Mathematica, the home directory is given by evaluating the `$HomeDirectory` Mathematica variable. Since version 5 (see documentation of Mathematica), the Mathematica base directory is given by the `$UserBaseDirectory` variable. For example, in Windows, it is probably located in `~\Library\Mathematica\Kernel` directory: this is where you should put the `init.m` file.

```
SetDirectory[ "/ your mmalpha dir/config"]
<<diagnose.m
```

This will check various potential problem, report them in the Mathematica window, and write also a file `diagnosis.txt` in your home directory.

Here are a few difficulties you may encounter, and some ways to overcome them. Additional information is given in the test documentation, in file

```
$MMALPHA/doc/Tests/testing-MMALPHA.pdf
```

1.4 The MMALPHA environment variable is not set

To check this, evaluate

```
Environment[ "MMALPHA" ]
```

in your notebook. If it does not answer the proper value, there is a problem.

First, remember that you must start MMALPHA from a shell, otherwise, Mathematica does not inherit from your environment variables (this is true on Unix-like systems, not on Windows -like).

Second, it may be that your `init.m` file was not loaded: see next section.

1.4.1 The initialization file is not correctly installed

Most of the problems come with the fact that the `init.m` file may not be installed in the correct directory. If this is the case, evaluating `test1[]` for example will just return `test1[]` unevaluated, as Mathematica was not able to load MMALPHA: indeed, loading is done through the `init.m` file.

After launching Mathematica, make sure that the Kernel is also launched by evaluating any Mathematica expression (for example, `2+2`). Before this evaluation is done, the Mathematica's Kernel should evaluate the content of the `init.m` file, which results in a message in the Messages window of Mathematica. This message ends with:

```
Alpha V2.0 Initialization
```

```
The Documentation can be found in ...
```

```
Current version in ...
```

```
Current directory is ...
```

```
If you use the notebook interface, you can open the master notebook:
```

```
In[1]:= start[];
```

If the Messages window is not opened, the `init.m` file was not called and may not be in the proper directory. To check this, evaluate the expression `$UserBaseDirectory` in Mathematica, and then check that the `init.m` file is in the `$UserBaseDirectory/Kernel`.

1.4.2 The Domlib does not start

This happen if in the Messages window, the following message appears:

```
Warning: could not install domlib
```

Also, if you run the `test1[]` command, you do have a `True` result. Several reasons may be the cause of this problem.

1. Variable `$PATH` may not contain the directory where the Domlib is. The Domlib is in the directory `$MMALPHA/bin.ostype`, where `ostype` is `darwin` for MacOS X, `linux` for Linux, and `cygwin32` for Windows. On Unix-like systems, check the value of this variable by the command `echo $PATH`. On Windows, check the value of the `PATH` environment variable.
2. The binary file for the domlib may not be in the proper directory. This means that `MMALPHA` has not been installed properly, or altered. Again, check that the directory `$MMALPHA/bin.ostype` contains `domlib`. If this is not the case, unzip the `MMALPHA` distribution to get it (if you do not find it in the distribution... just send me a mail !)
3. The `domlib` binary does not fit with your configuration. You then have to recompile it. I have not yet been able to configure `MMALPHA` so that recompiling is easy, but I try. Please, refer to the test documentation first, then to the documentation about `domlib`.

1.4.3 Other problems

You may encounter other problems, for example, some tests are corrects, some other are not. After the execution of the `test1[]` through `test4[]` commands, you find a test report file in the `$MMALPHA/tests` directory: send me this file, and I'll try to find out what is wrong in order to help you.