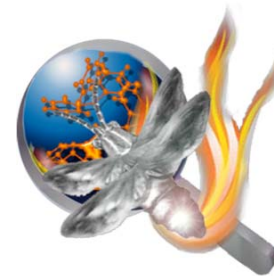


COMPLETE INSTRUCTIONS FOR
INSTALLING AND RUNNING
PC-GAMESS / FIREFLY UNDER MAC OS X
(Tested on Tiger & Leopard)



FOR MAC INTEL MACHINES ONLY!!

There is no PPC platform version available (nor will there be one in the future).

Thank you for your interest to the PC-GAMESS/Firefly package! Please follow all of the steps outlined below to successfully install PC-GAMESS/Firefly on your Intel Mac computer. Constructive feedback, bug reports and feature requests are encouraged to help improve the project. Conversely please double and triple check before reporting any “bug” to ensure that it is not “driver error”.

To view the most current version of PC-GAMESS/Firefly, you will want to visit the official project homepage at: <http://classic.chem.msu.su/gran/games/index.html>

There is also a Mac specific page posted at <http://classic.chem.msu.su/gran/games/macosex.html> although you should always also visit the main home page to find out about major project announcements, updated and general points of interest for all PC-GAMESS/Firefly users.

Lastly, there is also a Mac beta webpage located at http://classic.chem.msu.su/gran/games/macosex_beta.html that may contain newer versions that are provided on the main public release website. Please note that anything from the beta website may or may not work as expected and is intended for the adventurous in spirit and those requiring a particular new feature that is not yet available in the last public release.

STEP 1 :: Getting friendly with the UNIX roots of your Mac

Although a rather extensive set of graphical Mac native applications are provided to submit and terminate jobs as well as to test whether all dependencies are correctly intalled to run PC-GAMESS/Firefly for Mac it is still recommended that you familiarize yourself with the UNIX roots of your Mac. To make the fullest use of PC-GAMESS under Mac OS X some familiarity with the basics of UNIX is quite helpful as it will allow you to submit jobs with any options not supported with the graphical tools and to gain a better understanding of the program itself. Even though these graphical applications should greatly reduce the learning curve to install and operate PC-GAMESS / Firefly on your Mac, it is still recommended that you familiarize yourself with the

UNIX basics of your computer including running commands from a command-line and the fundamentals of UNIX file permissions and ownership. If you are not already comfortable with these concepts, a couple of links are provided below that go over these basics in a teaching friendly format. These tutorials and manuals are written for Linux, but most of the concepts also apply to any UNIX system such as Mac OS X as well (the locations of some of your files may just differ from the Linux convention such as having the home directory located at /Users rather than /home for example). If you are already comfortable with UNIX concepts and regularly use the terminal to run commands in OS X, then reviewing these websites is unnecessary.

<http://wwwhomes.doc.ic.ac.uk/~wjk/UnixIntro/index.html>

<http://www.tldp.org/LDP/intro-linux/html/index.html>

STEP 2: Getting the Dependencies Installed

Download the application that will determine what dependencies need to be installed to ensure that your computer is ready to run PC-GAMESS / Firefly for Mac. The download URL is:

<http://classic.chem.msu.su/gran/gamess/macosx/TEST-FIREFLY-DEPENDENCIES.zip>

Just double click on the downloaded .zip archive and a folder called TEST-FIREFLY-DEPENDENCIES will be created. Just double click on the Mac application “CHECK FIREFLY DEPENDENCIES RUN ME FIRST” inside this folder. If your computer already has all of the dependencies (helper applications required for normal operation) then you will see this returned:

Testing the required dependencies for PC-GAMESS / Firefly to run on your Apple Macintosh Computer:

X11 is installed on this system

Darwine is installed on this system

TextWrangler is installed on this system

End of Dependency Testing.

ALL TESTS SUCCESSFULLY PASSED.

Go ahead and run the Firefly DMG Installer (Install Code Required!!)

TO OBTAIN INSTALL CODE FOR THE Firefly DMG INSTALLER PLEASE SEND REGISTRATION E-MAIL REQUEST TO gran@classic.chem.msu.su WITH THE REQUESTED INFORMATION (SEE [Registration_email_template.txt](#) for Details)

If your computer is missing any of the dependencies, a Safari window will pop-up that provides a download link to the required programs. All dependencies for PC-GAMESS / Firefly to run on your Mac are FREE software (either open source or available for use without registration or cost to the end user). Please do not proceed to the next Step until your computer passes the dependency test and returns the text as shown above. Find below a more detailed discussion of each of the three major dependencies. There are other programs that allow you to create input files and visualize the resulting output file after a job run. These are not strictly dependencies as a number of options are available and you could operate strictly by creating input file in TextWrangler and viewing the resulting output file only in TextWrangler (although it would be quite difficult).

Dependency 1 :: Install Darwine and X11 if necessary (WINE for Mac OS X Intel)

For most users the Darwine dependency will return a fail (as you probably do not have it installed on your computer) and will automatically start a Safari web browser that loads the download website for Darwine. You will then need to download and install Wine for Mac (typically termed Darwine). Mac for Wine (Darwine) is not an officially supported release of the main Wine project. A link to the Mac OS X page for Wine is provided below for your reference and further review.

<http://wiki.winehq.org/MacOSX/Installing>

I would recommend just installing one of the stable or development precompiled binaries that are graciously provided by some volunteers (this is the download website that is automatically started when the dependency test starts).

<http://www.kronenberg.org/darwine/>

The development binary from Kronenberg, release 1.1.6 has been thoroughly tested, although others should work as well. As some advances in Wine may break previously working versions, if you get everything working I would just simply leave everything alone (in other words don't upgrade without a good reason).

I would recommend installing Darwine into the typical /Applications folder (default install). All further details will assume this standard installation location (in fact most of the automated graphical applications will only work if you install Darwine into the default location).

In principle you can install Darwine in any location you wish although I have not confirmed this path independence myself. The absolute path to the wine binary assuming a default install is then:

/Applications/Darwine/Wine.bundle/Contents/bin/wine

!!!FOR TIGER USERS ONLY!!!

X11 is a strict (sub)dependency for Darwine to install and so you MUST install X11 from your Apple provided OS X restore disk (or your Apple OS X upgrade disk if you upgraded your OS since purchase) or you will NOT be able to run Darwine correctly (and as a consequence PC-GAMESS/FIREFLY). If you have already installed X11 to support another application it will be located in the /Applications/Utilities folder on your computer. The graphical dependency test application will determine if you have X11 installed on your computer (most Leopard users WILL have it installed and most Tiger users WILL NOT have it installed).

Please note that X11 is NOT installed by default for Tiger Operating System as delivered by Apple or any computers restored or upgraded from the Apple OS install disks.

Please go to the “Other Installs” folder on the main OS restore disk or upgrade disk and run the X11 installer. This will install an OLD version of X11 that will probably be incompatible with any new Darwine build that requires a current version X11 to be installed. After X11 is installed from your OS restore disk or OS Upgrade disk, simply run “Software Update” as this will automatically bring the current version of X11 version and everything should then work fine with Darwine. Every Apple computer can support having X11 installed but, unfortunately, it is not installed by default for the workstation or server Tiger. X11 must be installed from the Apple provided OS media and then updated via software update. You CANNOT download any installer from the Internet to do this. You must have ORIGINAL Apple media to do this. If your original OS restore or upgrade media is missing, I would ask a friend to loan a disk for this purpose.

General Notes about Darwine

It will take Darwine a little bit of time to setup the wine configuration files (located in our User directory as a hidden file) when it launches for the first time. Subsequent calls to wine will take much less time to load.

All of the test job run scripts and graphical run applications rely on the absolute path to the wine binary to be /Applications/Darwine/Wine.bundle/Contents/bin/wine in order for them to run properly. If you have installed at an alternate location you will need to adjust this path in the scripts to ensure proper operation and none of the graphical job submission run applications will work for you without rebuilding them from source.

Dependency 2 :: Installing TextWrangler

You will want to have a text editor that supports UNIX, Mac and Windows text files. The TextEdit program that is found in the main /Applications directory is not suitable for the purpose of viewing text output (such as those from Firefly jobs while they are running) if the file is being written to as you view it (no auto-refresh). In addition, the TextEdit program is not the best choice to view very large text files. As such the Free program TextWrangler that is suitable for both of these purposes and has been made a dependency that must be met prior to running the DMG installer. When you run the graphical dependency test application, a browser will automatically be opened to the download website page for TextWrangler and it is reproduced below for your convenience.

<http://www.barebones.com/products/textwrangler/>

I would recommend installing TextWrangler into the typical /Applications folder (default install). All further details will assume this standard installation location (in fact most of the automated graphical applications will only work if you install TextWrangler into the default location). If this is not the case, then the output from your jobs will not automatically open in TextWrangler as they are intended to.

AGAIN, PLEASE BE SURE THAT YOUR COMPUTER PASSES THE GRAPHICAL DEPENDENCY TEST APPLICATION BEFORE PROCEEDING ANY FURTHER.

Step 3 :: Get familiar with the OS X Terminal Shell (xterm)

You will need to access the terminal shell application ("Terminal.app"). It is located in the Utilities subfolder of the main Applications folder. I would make an alias to it on your desktop or drop it into your Dock as you will be using it often. Starting the Terminal application when logged into your user account will make the default location as your user home directory (usually /Users/YOURACCOUNTNAME/). You can readily copy and paste from a text document to your command shell, or text displayed from a previous shell command. Recall that if you use a path to file string that contains a space you must put the entire string in quotation marks. For example if your path includes a space such as /Users/Joe/Folder with Spaces/PC-GAMESS/input1.inp it must be "/Users/Joe/Folder with Spaces/PC-GAMESS/input1.inp". More extensive details about using the terminal under OS X can be found by reviewing the links provided in the Step 1, as well as just performing some simple Google searches as appropriate.

Step 4 :: Obtain PC-GAMESS/FIREFLY Installation Key (Password) to run Installer

Please find below the contents of the readme.txt file for the PC GAMESS/Firefly binary distribution, as it contains important information on how to register and get installation key from the PC-GAMESS/Firefly Project Team.

First, carefully read the contents of the enclosed "License.pdf" file, which contains the PC GAMESS/Firefly Quantum Chemistry Package License Agreement (or else download it from the public PC-GAMESS/FIREFLY project website). If you do not agree with all the terms of this License Agreement, you cannot use the PC GAMESS and cannot apply for the installation key.

If you agree with the License Agreement, please read the "Registration_email_template.txt" document. Then, send the valid registration email to Dr. Alexander A. Granovsky (email: gran@classic.chem.msu.su) using "Registration_email_template.txt" as the template. Note that:

1. The subject field of your email must be "The PC GAMESS/Firefly registration".
2. Your email must be either plain text or html-formatted. **NO** attachments are allowed!
3. The only public e-mail service we accept emails from is Gmail. Otherwise, you **MUST** use your email account at your institution/organization.
4. The registration email **MUST** be written in English.
5. The header of your registration email (first 4 lines) **MUST** be exactly as in the Registration_email_template.txt.
6. You **MUST** answer questions #1 - #11, and exactly in the same order as specified.

The details of point 5 & 6 are shown below for your convenience (italicized for emphasis):

Dear Dr. Granovsky,

I agree with the terms of the PC GAMESS/Firefly Quantum Chemistry Package License Agreement. Please provide me the required installation key.

My registration information is summarized below as requested:

1. *Title (e.g., Mr., Ms., Dr., Prof., etc...)*
2. *Complete name (e.g., John C. Smith)*
3. *Position (e.g., student, graduated student, Engineer, Senior Researcher, Scientist, etc...)*
4. *Affiliation (e.g., Department of Chemistry, M.V. Lomonosov Moscow State University)*
5. *Postal address*
6. *Country (e.g., Russia, USA, Japan, etc...)*
7. *Contact email (e.g. alexey.a.popov@gmail.com)*
8. *Contact phone (including country and area codes)*
9. *The number of program copies you are planning to use (e.g., 1)*
10. *Operating System you plan to use (e.g., Leopard, Snow Leopard, etc...)*

11. List of processor(s) you are planning to use to run program (e.g., Intel Core 2 Duo, Intel Xeon 5150 Quad, etc..)

12. Provide any additional comments or questions you may have

Thank you in advance,

Your complete name (e.g., John C. Smith)

7. Registration emails which do not meet these requirements will not be considered.

8. If you have problems filling in your registration email, see the "Registration_email_sample.txt" for a sample.

If you still have any additional questions, contact Dr. Alexander A. Granovsky.

Your registration email will be answered within a couple of days. Upon receiving installation key, use it to run GAMESS/Firefly installer.

You MUST keep your installation key secure as it is the part of SOFTWARE and is subject to the same License Agreement as the PC GAMESS binaries themselves as stated in the License Agreement.

Step 5 :: Run the PC-GAMESS/FIREFLY for Mac Installer

The newest official release will be posted on the PC-GAMESS/Firefly for Mac OS X webpage:

<http://classic.chem.msu.su/gran/gamess/macosx.html>

For the latest beta version use this link instead:

http://classic.chem.msu.su/gran/gamess/macosx_beta.html

Thereafter, you should also visit the main project page for general news, updates, documentation and other things of interest such as the discussion forum (see links below).

<http://classic.chem.msu.su/gran/gamess/index.html>

Running the DMG installer is quite trivial as it mounts and install like any other Mac application. Please note that you CANNOT run this encrypted installer without the supplied license code that you will get by completing Step 4. Once complete, all of the contents will have been written to /Applications/PC-GAMESS-FIREFLY-MAC directory. IT IS NOT RECOMMENDED THAT YOU CHANGE THE INSTALLATION LOCATION OR MODIFY THE CONTENTS OF THE PC-GAMESS-FIREFLY-MAC FOLDER IN ANY WAY AS IT MAY CAUSE YOUR INSTALLATION TO STOP WORKING. You will want to copy the RUN-FIREFLY-JOBS folder and all contents to some convenient location on your computer to allow you to easily run your jobs. To view the package contents, just open your applications folders and navigate to this folder.

Step 6 :: Run the test scripts (Optional, but strongly recommended)

There are three sets of runscripts provided to test that your installation is working properly. One set for 8-core Mac Pro machines, one for 4-core Mac Pro machines (either Single Quad or 2X Dual-Core) and a set for all other Mac machines that currently possess 2-cores total. It is strongly encouraged that you run these runscripts and review the output files to ensure that everything is operating nominally for your installation.

For Mac Pro Machines only (8-Core – Dual Quad), simply open a Terminal application (in the Utilities Folder of the main Applications folder) and run the following two commands:

```
/Applications/PC-GAMESS-FIREFLY-MAC/BENCHMARKS-PARALLEL-TESTJOBS-RUNSCRIPT-MacPro-DualQuad-Only.sh
```

```
/Applications/PC-GAMESS-FIREFLY-MAC/ BENCHMARKS-THREADED-TESTJOBS-RUNSCRIPT-MacPro-DualQuad-Only.sh
```

For Mac Pro Machines only (4-Core – Single Quad or 2X Dual-Core), simply open a Terminal application (in the Utilities Folder of the main Applications folder) and run the following two commands:

```
/Applications/PC-GAMESS-FIREFLY-MAC/BENCHMARKS-PARALLEL-TESTJOBS-RUNSCRIPT-MacPro-SingleQuad-Only.sh
```

```
/Applications/PC-GAMESS-FIREFLY-MAC/BENCHMARKS-THREADED-TESTJOBS-RUNSCRIPT-MacPro-SingleQuad-Only.sh
```

For All other Mac Machines only (Dual Core), simply open a Terminal application (in the Utilities Folder of the main Applications folder) run the following two commands:

```
/Applications/PC-GAMESS-FIREFLY-MAC/BENCHMARKS-PARALLEL-TESTJOBS-RUNSCRIPT.sh
```

```
/Applications/PC-GAMESS-FIREFLY-MAC/ BENCHMARKS-THREADED-TESTJOBS-RUNSCRIPT.sh
```

****DO NOT RUN MULTIPLE BENCHMARK RUNSCRIPTS AT THE SAME TIME!**** This will overload your system and render the resulting timings to be meaningless, It will also probably crash your test jobs (and possibly your computer) in the process.

FOR TIGER USERS ONLY: to run multithreaded tests and any other multithreaded jobs correctly (such as MP3/MP4 and MCQDPT2 calculations), you **ABSOLUTELY** need to add the following line to the start of your input file:

```
$smp load=0 dianp=1 $end
```


due to some compatibility particular issues of Darwin/Tiger combination. This issue is not known to exist for Leopard users. This means that you will need to add this line to the start of all THREADED/MP4 test jobs to ensure proper runs if you are running under Tiger.

Those that are interested to run both of the benchmark scripts (whichever set is appropriate for your machine) to completion are encouraged to submit their timing results back to the PC-GAMESS/FIREFLY project for purposes of further optimization and support for the Mac platform. The output from complete runs for all tests can get somewhat large (particularly for the Mac Pro Dual Quad runscripts). You should compress the Parallel and Threaded test jobs folders separately. Simply enter into the BENCHMARKS folder and then right click on the folder that you want to compress. One of the options will be "Create Archive of PARALLEL" or "Create Archive of THREADED". When complete a compressed archive file called PARALLEL.zip or THREADED.zip will be created. It is recommended that you upload these zip files to an FTP server if you have access to one (just download and use the open source FTP Client Filezilla to access your FTP server from your Mac -> <http://filezilla-project.org/>) and sent an appropriate link back to the PC-GAMESS/FIREFLY project. If you do not have access to an FTP server, there are free services such as GETDROPBOX (<https://www.getdropbox.com/>) that allow you to upload large files and provide a public download link (when the file is located in the Public folder and you select the Copy public URL option). The e-mailing of large files may result in your message being stopped by your provider SMTP relay server or the spam and firewall of your recipient. It is always recommended that you upload your large file to location that is accessible via a public URL address and provide a download link to this location (whether sent to an FTP or via a service like getdropbox).

Step 7 :: How to run PC-GAMESS / FIREFLY jobs on your Mac:

I shall presume that you are familiar with how to create input files suitable to run on PC-GAMESS / Firefly and generally familiar with running with computational chemistry programs. If not, I would recommend pausing and reviewing the more complete installation and usage guide.

There are currently three ways to run PC-GAMESS / Firefly jobs on your Mac:

METHOD 1 (Graphical Drag and Drop Job Submission Run Applications)

There is a set of drag and drop run applications provided with your install package in the RUN-FIREFLY-JOBS directory. They can also be downloaded separately by using the URL download link below:

<http://classic.chem.msu.su/gran/gamess/macosx/RUN-FIREFLY-JOBS.zip>

A good idea, for example, it to copy the RUN-FIREFLY-JOBS folder to whatever location is most convenient for your computer and run your jobs from there. There are a total of six Mac job submission run applications, one GUI job termination application and one shell runsript contained in this download. The main seven native Mac applications located in the root of the “RUN-FIREFLY-JOBS” folder are:

Run-Firefly-Job-1-Core
Run-Firefly-Job-2-Cores
Run-Firefly-Job-4-Cores
Run-Firefly-Job-8-Cores
Run-Firefly-Job-Interactive
Run-Firefly-Job-Interactive-Batch
runpcgmac.sh
STOP-FIREFLY-JOBS

In addition, there are six specially configured Mac applications located in the “AUTO-OPEN-IN-GUI” subfolder of the main “RUN-FIREFLY-JOBS” folder and they are:

Run-Firefly-Job-1-Core-GUI
Run-Firefly-Job-Interactive-GUI
Run-Firefly-Job-2-Cores-GUI
Run-Firefly-Job-4-Cores-GUI
Run-Firefly-Job-8-Cores-GUI
Run-Firefly-Job-Interactive-Batch

A more complete discussion of each of the applications, their capabilities and limitations is detailed in this Step 7, Method 1 portion of this documentation.

Single Job Submission Run Applications

(Everything except Run-Firefly-Job-Interactive-Batch)

The name of the applications pretty well explains their usage. Simply select and drag and drop your input file onto the run application and it will start your PC-GAMESS / Firefly jobs for you. In the case of the run applications that indicate the number of processor these programs will always submit your job with the number of processors in the name of the application. Any error messages or status information will be returned in the

text window after the run application starts. If your job launches successfully, the output file will be written to the SAME directory as where the input file was located, with the .out extension added. The output file will be automatically opened with TextWrangler (one of the applications that has been installed to satisfy the dependency tests). As your job progress, the TextWrangler program will continuous display the current status of your output file. As the PC-GAMESSS / Firefly programs continues your job, you will see the updated output results in real time in TextWrangler. In addition, a PUNCH file for your job will automatically be created in the same directory as the input file with the name INPUTFILENAME.DAT (just view your input file directory to see this).

If you wish to visualize this output file while a job is in progress (for example for a long geometry optimization), simply copy and paste this output file to an alternate location and open this copied file with a program of your choice (see below for suggested programs to visualize output files).

For the fifth application called “Run-Firefly-Job-Interactive” this is a unified script that will allow you to select the number of processor that you wish to use after dragging and dropping your input file onto the run application.

Double-clicking all of these applications rather than dragging and dropping an input file onto them will prompt you to select an input file via a typical graphical interface (the final result is identical between dragging and dropping and double-clicking and then selecting a file).

Please note that all of these run application support concurrent job submission on the same script and from the same input directory (there are provisions to ensure that none of the temporary files and punch files will collide). For example, if you have four processing cores available, you could run the 1 processor script four times in succession or else when the 2 processor script twice in succession or lastly the 4 processor script one job at a time (You must wait for the previous job to terminate before starting another 4 processor job).

SPECIAL NOTES FOR [Run-Firefly-Job-Interactive-Batch Application](#)

Use of this batch mode graphical job submission run application is very simple. Just double click on the application and a graphical interface will pop-up that asks you to select the number of processing cores that you wish to use for all input files that will be selected (current options are 1, 2, 4 or 8). Next a graphical interface will pop-up that asks you to select the files that you wish to submit to this batch job application. There are no particular limits on the number of files selected but you may wish to keep it to something reasonable (perhaps 100 or less). Next assuming that you have selected a valid number of cores for your machine and you have

submitted valid input files your jobs will begin to launch sequentially (in principle very similar to how the benchmark batch commandline runscripts work). There are some notes regarding the differences between the other five single job submission run applications and this batch run application that you should be aware of.

- 1) You cannot drag and drop a set of files unto this batch job submission application. You must double click the application instead and select the files that you wish to run using this method. The other single job submission application support both drag and drop and double click input methods.
- 2) The output file will NOT automatically open with TextWrangler soon after launching as it does with the single job submission run applications. TextWrangler will instead only launch after the job has COMPLETED.
- 3) Only ONE value for the number of processing cores is supported for all input files.
- 4) This batch run application does not optimize the resources on your computer, but rather just simply sequentially launches all of the input files supplied just as if launched from a command line. You will need to ensure manually that the number of processing cores selected and memory requested take full advantage of your resources.

These limitations may or may not be removed in future version of the batch job submission run applications. As always you should report any bugs or features requests.

NOTE: This graphical batch job submission run application IS NOT suitable for running the benchmarks. Please use the benchmark runscripts as provided as some required options are not supported in the graphical batch runscript are that are needed for proper completion of these benchmark runs. Details on how to run the benchmarks as appropriate for your particular machine are contained in the complete install and usage guide.

SPECIAL NOTES ABOUT RUN APPLICATIONS IN THIS [AUTO-OPEN-IN-GUI](#) SUBFOLDER::

Run-Firefly-Job-1-Core-GUI

Run-Firefly-Job-Interactive-GUI

Run-Firefly-Job-2-Cores-GUI

Run-Firefly-Job-4-Cores-GUI

Run-Firefly-Job-8-Cores-GUI

Run-Firefly-Job-Interactive-Batch

These Firefly job submission run applications are identical to their counterparts in the main RUN-FIREFLY-JOBS folder except for the following two modifications:

1) All of these job launch applications will not close after the firefly job is started. This means that if you terminate these job launch applications, any Firefly job that was submitted by the application will also immediately terminate as well.

2) After submitting the Firefly job, each application will wait until the job is completed and then display the results in a text editor (TextWrangler is the result) and open the result of the job in a graphical visualization application (MacMolPlt is the default).

Please note that these job submission run applications will require that MacMolPlt is installed into the default location (the /Applications directory). You can download MacMolPlt if it is not already installed on your system by using the following URL below:

<http://www.scl.ameslab.gov/MacMolPlt/>

Please note that although MacMolPlt is the default graphical visualization application to open the completed job, this can be readily modified in the application internal .config file. The next section provides detailed information about how to modify the .config internal configuration files for ANY run applications.

HOW TO CUSTOMIZE YOUR GRAPHICAL JOB SUBMISSION/LAUNCH RUN APPLICATIONS

The default configuration options set for your job submission applications should (hopefully) prove quite suitable for the vast majority of Firefly for Mac users. There are, however, a number of customizations possible for all of the job submission run applications, mainly to support any non-default (non-standard) program installation paths and job launch options. This includes but is not limited to changing the number of processing cores to be used (for those applications that do not take this variable from a graphical dialog box), changing the absolute path of any dependency or helper application as well as the absolute path to the Firefly binary itself. In addition options such as custom scratch directory and file designations as well as opening the completed output file in a graphical visualization program can also be supported. If you are interested in making such changes to your Firefly for Mac job submission applications through modification of the internal configuration file, please read the notes below to get the necessary instructions. All of the important variables are loaded from an internal configuration file. This has been done to allow for the following goals or tasks to be better achieved:

1) Non-standard (non-default) installation locations are now supported for all significant variables. The default locations are entered into the internal .config file and loaded at run time and can be modified as needed. As a general rule there is no good reason to modify any of the default settings for any of the configuration variables without rather good cause and careful testing thereafter.

2) For all job launch applications you can now specify the exact scratch directories that you wish to use rather than only being able to only use the -t scratch file switch that assigns such directories automatically once the absolute root directory scratch path is provided. For most users this will not mean anything unless you are interested in running conventional jobs and have more than one hard drive to use for scratch files. Without this condition, there will not be any worthwhile improvement in file I/O operations to warrant changing the default setting. The most common reason for using the manual scratch directory assigns is most likely MP2 jobs or some other convention or semi-direct computation.

3) All configuration variables are echoed at the start of each application and the exact execution for each job is also echoed out. This verbose style is included for better diagnostic purposes if any bugs are observed. If you should observe any bugs, please include a copy of the application text dialog box output when sending in a bug report.

4) To edit any of the configuration files, just right click on the application and select "Show Package Contents". Then enter the "Contents" directory. Now enter the "Resources" directory. There will be a file with a .config file extension in this folder. Simply open this text file in a normal text editor such as TextWrangler. There is an extensive explanation of how to modify the configuration file contained with the actual .config configuration file itself. Incorrect modification will render the application non-operational. At this point there is no error checking so if you enter any incorrect configuration variables the application will just break (you will get some seemingly cryptic error message returned in the text dialog box).

5) The batch application includes a variable to open the completed output file in a graphical editor/visualization program in addition to the standard open in a text editor (this is always done for all job submission applications). This option is, however, disabled by default for the version in the main "RUN-FIREFLY-JOBS" folder for compatibility reasons (one less dependency). The simplest way to enable this option is to open the file runfireflygui-no-cpu-batch.config contained in the "Run-Firefly-Job-Interactive-Batch" application (see instructions from point 4 above) and change the 0 on the eighth line to a 1. If you have MacMolPlt installed in the default /Applications directory then everything is set to view the final output file with this program. If you have the MacMolPlt program installed in a non-default location or wish to use another program then you will also need to edit line 9 as appropriate.

6) The first line for any of the job launch applications that have a certain number of CPU hard coded is the number of CPU processing cores to be used. To make a run application that will launch each job with a different number of CPU processing cores than the 1, 2, 4, & 8 that are provided just copy one of these

applications and rename it to something logical (like Run-Firefly-Job-6-Cores if you should like to make an application to run jobs on six cores each time). The just navigate to and edit the first line of the .config file to appropriate value and you have created a job submission application to run on 6 processing cores.

The overall concept of the internal configuration file (.config) scheme was that everything should work EXACTLY as before in that it should be simple and easy to run jobs with Firefly for Mac. The documentation and variables in the internal configuration files will also allow power users to tinker with these applications if you so desire. This is certainly not necessary as the applications are designed to work perfectly out of the box, but available for those that wish to use some of the non-standard install locations and job launch options.

SPECIAL NOTES ON STOP-FIREFLY-JOBS application

The Mac application STOP-FIREFLY-JOBS must be double clicked to start (no drag and drop support for this application). Once started, the application will find all currently running Firefly for Mac jobs on your machine, provide a simple Yes-No pop-up dialog box with a short description of each Firefly job that is currently running and provide a prompt that allows you to terminate or not terminate each job found (Yes / No). In the text box that is automatically opened when the STOP-FIREFLY-JOBS application is started a more complete summary for each job is printed out and can be reviewed in greater detail is desired.

*****KNOWN LIMITATION FOR STOP-FIREFLY-JOBS APPLICATION!*****

For this application to work, you will need a 100% default installation of everything. This will mean that if you customized any of the application configuration (.config) files doing so may well break the STOP-FIREFLY-JOBS application. Due to limitations in the parsing code used for this application, if the absolute path to the input file used to run your Firefly for Mac job has any spaces in the any of the subdirectories or the actual input file name itself, this application will return nonsense data for the reported job parameters (or possibly no data at all). Is this occurs, just answer “No” to the terminate job prompt for each job and correct the file path issue.

There is currently no workaround for this issue. More generally, it is NOT a good idea to use spaces in directory names or file names when running on a UNIX system such as Mac OS X. While spaces in file names and directories is supported by OS X and most programs running on OS X, each instance of a filename or directory with a space or special character included requires a workaround to the ordinary UNIX conventions that do not support spaces. Thus the use of spaces in directory and file names needlessly increases the possibility of errors and crashes. It is recommended that any directories and filenames that are to be used in any way with the Firefly for Mac shall contain no spaces and no special characters. Good substitutes for spaces in filenames are the underscore “_” or dash “-“. As an example, my “input file 2 with spaces.inp” should become “input_file_2_with_spaces.inp” or else something like “input-file-2-with-spaces.inp”. Due to some limitations

(truncation) in the ps UNIX utility under Tiger upon which this application relies, the STOP-FIREFLY-JOBS application is only supported for Leopard or newer versions of OS X.

Error Checking and Reality Check for your Job Runs

There is some rudimentary error checking in these run applications that will prevent you from running a single job submission on more cores than are present on your machine. If your job run request exceeds the number of cores on your machine, the run application will abort. You can, however, overrun the total number of cores available if you submit multiple jobs that in AGGREGATE exceed the total number of processing cores available. Be careful not to do this as it may make your system sluggish, unstable, or even crash. If this occurs, please refer to the complete install and usage guide as termination of PC-GAMESS / Firefly is covered in greater depth. In addition, care should be taken to not request more than the total PHYSICAL memory available on your machine as memory swapping could occur as a result giving very poor performance as a result. It is encouraged to carefully monitor the progress and system usage of your job to ensure that everything is proceeding smoothly (the complete install and usage guide discusses the methods to check the real-time CPU and memory usage).

If you correctly choose your options, you can get efficiencies for many parallel jobs up to 99% for many direct SCF runs. Most single point energy type jobs should include the following two options for best parallel job run results:

```
$p2p p2p=.t. dlb=.t. $end  
$scf dirscf=.t. $end
```

Most geometry optimization type jobs should include the use delocalized coordinates (DLCs) rather than Cartesian coordinates for best results. To enable this use nzvar=1 (or other compatible nzvar setting) in \$contrl group, and an extra card \$zmat dlc=.t. auto=.t. \$end. If Firefly fails to generate DLCs it probably means you have "disjointed" pieces in your molecule and you should add nonvdw(1)=... keyword to \$zmat as appropriate to complete your zmatrix. Obviously choosing the DLC option means that your input coordinates must be in a zmatrix of one form or another. Please review the online documentation for further details as these two comments are only intended as suggestions for some of the most common optimizations.

Note: While conventional SCF jobs are fully supported with this PC-GAMESS / Firefly for Mac implementation, the file I/O limitations may limit the efficiency of such jobs.

METHOD 2 (Use the commandline runpcgmac.sh Runscript)

The runpcgmac.sh shell runscript is contained in the RUN-FIREFLY-JOBS directory (either downloaded separately or in the main distribution). The runpcgmac.sh shell runscript is an excellent way to run PC-GAMESS / Firefly jobs on your Macintosh if you are comfortable in the Terminal application and prefer

running applications from a command line. For previous GAMESS-US for Mac users, this scrip will most resemble the rungms script that you have used in the past (please note that there significant differences however and you will need to read the usage instructions to get it to work correctly). **One of the most notable differences between the runpcgmac.sh script and the rungms script is that the path to the input file and output file parameters MUST BOTH BE ABSOLUTE to get a successful job run.** For example,

The following usages **WILL NOT** work:

```
./runpcgmac.sh inputfile1.inp outputfile1.out 2
```

```
./runpcgmac.sh ./inputfile1.inp ./outputfile1.out 2
```

But the following usage **WILL** work:

```
./runpcgmac.sh /Users/tpatko/Research/Firefly/inputfile1.inp /Users/tpatko/Research/Firefly/outputfile1.out 2
```

You can graphically drag and drop your input file onto the xterm Terminal application. This will return the absolute path of the input file that you drag and dropped onto the shell terminal (quite convenient).

As is usually the case, using the command line runscrip version offers you the option to configure your runscrip extensively. Detailed instructions on how to use this command line runscrip are provided in the notes of the scrip itself. Simply open this shell scrip file with TextWrangler and read the usage notes carefully. Be careful not to modify the scrip unless you are certain of the changes that you are making as your changes may cause the scrip to not function normally. Most users will not need to make any changes to this runscrip, but rather just read the usage instructions and submit the appropriate three parameters to run the scrip. The function of the graphical run applications and command line runscrip is identical (the command line runscrip also automatically open the resulting output file in TextWrangler, for example).

METHOD 3 (Pure command line execution string)

To run a PC-GAMESSS/FIREFLY job from a command line without any scrip or run application, just use the following guideline:

```
/Applications/Darwine/Wine.bundle/Contents/bin/wine /Applications/PC-GAMESS-FIREFLY-  
MAC/pcgamess.exe -osx ***
```

*** After the `-osx` switch followed by a space, please follow all of the normal PC-GAMESS/FIREFLY command line options as detailed in the enclosed in the readme folder. You can use the provided runscrips as a quick guideline for running some types of jobs. Consult the relevant manuals and the runpcgmac.sh commandline runscrip for more complete information.

The paths selected for the input, output and scratch file locations are completely up to the user so long as you have read/write permission to that path. With this method, please note that ALL paths must be ABSOLUTE to get a successful job run. There are a couple of notes that can make it easier to use this job submission method:

- A) You can graphically drag and drop your input file onto the xterm Terminal application. This will return the absolute path of the input file that you drag and dropped onto the shell terminal (quite convenient).
- B) You can prepare your long execution string in a text editor such as TextWrangler using the benchmark runscripts as a guideline. Once ready, you can copy and paste this long execution string into your xterm terminal shell application and just run the job more conveniently.

If you wish to use this method to submit jobs, please review the benchmark runscripts in TextWrangler as a guideline as well as the standard documentation. In addition, if you wish to run submit batch job runs that require options not supported by the graphical batch job submission run application, then making an analogous batch runscript to those shown in the benchmark runscripts is your best (and only) option. Running jobs using this method offers 100% control of the exact execution string, and 0% convenience in automating the process of job submission!

SPECIAL NOTES ABOUT RUNNING PC-GAMESS/FIREFLY UNDER DARWINE

Part 1 :: Memory Limitations and OS Performance

Wine is a fully open source implementation of the 32-bit Windows runtime environment (API). The maximum memory that can be allocated per process is 190MW (1520MB) under the current implementation of Darwin. On an 8-core Mac Pro this means a maximum of 12160MB (~12GB+) total memory can be allocated for an 8-core parallel run job. For all other Mac machines with 2-cores total, this would mean that up to 3040MB (~3GB+) total memory can be allocated for a 2-core parallel job. Multiple concurrent parallel jobs are supported using Darwin, so long as the scratch files for each job are located at unique paths (be careful not to overrun your maximum number of processing cores available and total memory available on your machine). Users are encouraged to run all jobs as direct SCF calculations whenever possible (essentially sufficient memory is available), as there is some file I/O degradation running PC-GAMESS/FIREFLY under Wine that may result in slower runtime for non direct SCF jobs. In contrast running PC-GAMESS/FIREFLY under Wine using direct SCF results in very little performance loss at all (even for parallel jobs up to 8 cores). It should be noted that performance results obtained running under Leopard (10.5.5) were better than those obtained running under Tiger (10.4.11). In addition, no compatibility issues were observed running under Leopard, and some minor compatibility issues were observed running under Tiger (mainly for multithreaded jobs such as MP3/MP4, ...etc). **If Leopard OS is installed on your Mac, it is recommended to use PC-GAMESS/FIREFLY for Mac under Wine with this preferred OS.**

Part 2 :: Terminating PC-GAMESS/FIREFLY jobs run under Darwin

If you have never terminated command-line programs on your Mac before, it is slightly different than terminating a hung graphical process (which you do using your mouse). There are really two options (only one option for Tiger, actually):

COMMANDLINE TERMINATION (TIGER AND LEOPARD)

open a terminal shell application. Type “ps -A”. The terminal will return all current running processes (both those are owned by you and those that are not). Find the PID of the process(es) named “mwine”. The relevant portion of the ps -A output from a parallel 8 processor job is shown below for your reference (it will be better formatted in your shell).

```
PID COMMAND  %CPU TIME  #TH #PRTS #MREGS RPRVT RSHRD RSIZE VSIZE
1345 bash    0.0% 0:00.01 1 14 19 252K 652K 892K 18M
1344 login   0.0% 0:00.02 1 17 55 188K 264K 976K 19M
1343 mwine   99.6% 7:06.77 2 21 218 705M 1752K 241M 1265M
1342 mwine   98.5% 6:54.97 2 21 224 747M 1752K 241M 1265M
1341 mwine   98.6% 7:06.25 2 21 218 704M 1752K 241M 1265M
1333 mwine   98.2% 6:53.93 2 21 215 747M 1752K 241M 1265M
1296 mdworker 0.0% 0:00.17 3 51 32 380K 11M 1292K 37M
1247 mwine   96.1% 10:12.94 2 21 219 1037M 1752K 1029M 1265M
1246 mwine   98.6% 10:35.04 2 21 225 1038M 1752K 1029M 1265M
1245 mwine   97.5% 10:09.18 2 21 219 1036M 1752K 1029M 1265M
1244 mwine   0.0% 0:00.07 4 30 154 0 1548K 696K 39M
1243 mwine   0.0% 0:00.07 3 24 140 0 1548K 696K 37M
1240 wineserver 0.0% 0:00.70 1 20 25 144K 304K 356K 20M
1231 mwine   97.6% 10:08.22 2 21 216 1041M 1752K 1029M 1265M
927 BBEdit  0.0% 0:34.83 8 179 501 14M 24M 24M 432M
879 mdworker 0.0% 0:00.93 4 74 161 784K 7412K 2044K 74M
```

In this case since this is a single node 8 processor parallel job, terminating any of the mwine PID will also automatically terminate all of the other mwine PID as well. For example, just typing “kill -9 1343” will immediately all mwine processes (the abnormal termination will be reported in our PC-GAMESS/FIREFLY output job).

GRAPHICAL TERMINATION (LEOPARD ONLY)

The easiest method to terminate your PC-GAMESS/Firefly job running under Leopard is just to double click on the application called STOP-FIEFLY-JOBS in the RUN-FIREFLY-JOBS subdirectory in your PC-GAMESS-FIREFLY-MAC main application directory. The application use is self-explanatory from there. Another method (if you prefer), is to go to your /Applications folder and open the /Utilites subfolder. You will find a program called “Activity Monitor”. Simply double-click and open this program. You will be able to monitor the CPU, memory and disk usage of your Mac computer with this utility. In addition, you can graphically terminate any process you wish. Be careful not to terminate an essential system service as any termination of a process is instantaneous!

Appendix 1 :: Graphical User Interface (GUI) for Mac

Mac Native Applications to prepare input files and visualize output files

The following applications are tested and known to work on Mac OS X with PC-GAMESS/FIREFLY input and output files: WebMO (Both the Free and Commercial version), MacMolPlt, Gabedit, PCModel, Orbdraw and Vibrate (from Serenasoft). For WebMO, you will want to enable the PC-GAMESS/FIREFLY interface as the administrator and then prepare an input job as though you were planning to run it directly from WebMO (although you cannot actually do this at this time). Go to the Preview tab just before submitting the actual job. This will generate the text input for your job. Copy and paste this text into a text editor like TextWrangler and modify as appropriate to optimize your job run. Importing of PC-GAMESS/FIREFLY jobs run under Wine for Mac in WebMO proceeds just as it would if the PC-GAMESS / Firefly job was run under Linux or Windows.

<http://www.webmo.net/>

WebMO has specialized install instructions for Mac users (see below).

http://www.webmo.net/support/installation_osx.html

MacMolPlt (Free) and PCModel, Orbdraw & Vibrate (Commercial) can also prepare PC-GAMESS / Firefly compatible input files, and view (visualize) the resulting job output files (such as molecular orbitals, vibrational modes, spectra, ...etc). Please note that none of these programs can currently allow you to run a job directly from the GUI (not even WebMO). You should run your jobs using one of the three options described above.

<http://www.scl.ameslab.gov/~brett/MacMolPlt/>

<http://www.serenasoft.com/>

Marvin from ChemAxon (including MarvinSketch, MarvinView and MarvinSpace) is a very convenient and intuitive way to draw chemical structures in 2D or 3D. In particular it has an extensive fragment library and support custom fragment templates. Although Marvin does not allow you to prepare or visualize Firefly input and output files, the structures can then be exported via a variety of common file format (MDL .mol, Sybl .mol2, XYZ,) which can be imported into any of the programs described here to create the final Firefly input file. ChemAxon products are provided free of charge for academic users (registration required).

<http://www.chemaxon.com/products.html>

Windows Only Applications to prepare input files and visualize output files that have been successfully reported to run using some implementation of Wine or Windows Emulation

Gabedit (FREE)

<http://sites.google.com/site/alloucheat/Home/gabedit>

Chemcraft (Commercial with Trial Version)

<http://www.chemcraftprog.com/>

Facio (FREE)

<http://www1.bbiq.jp/zfffelis/Facio.html>

Crossover Wine for Mac (Commercial Implementation)

<http://www.codeweavers.com/products/cxmac/>

or Darwin (Open Source Implementation)

<http://www.kronenberg.org/darwine/>

You will already have Darwin installed, as this is a strict dependency for running Firefly on your Mac.

If you have any difficulty running these programs under Crossover Wine or Darwin you should definitely be able to run them with VMWare Fusion or Parallels.

<http://www.vmware.com/products/fusion/>

<http://www.parallels.com/>

Other software with successful results reading and visualizing PC-GAMESS input and output files on the Mac platform would be good to know.

Appendix 2 :: Viewing Hidden & Invisible Files in Finder

The Mac Finder does not show invisible files (files that begin with “.”) and many system file folders. One of the folders that is not displayed is the temporary folder (/tmp). To view this and folder folders not normally visible under the normal Apple Finder you can use a program TinkerTool that allows you to toggle between

viewing hidden (invisible) and system files and folders and returning back to the default hidden status. Just simply click on the Show hidden and system files checkbox under the Finder tab and click on relaunch Finder button. To disable viewing hidden and invisible files just uncheck the same checkbox and relaunch Finder again. You can download TinkerTool at:

<http://www.bresink.de/osx/TinkerTool.html>

** Please note that modifying any of the systems files (delete, rename, move, ...etc) can severely damage your operating system. Please be extra careful when show invisible and hidden files and folder is enabled.

Appendix 3 :: pcgameess.exe binary for early Mac Intel Machines

The supplied pcgameess.exe binary in your application will be suitable for the vast majority of Mac Intel users running under Tiger or Leopard. Some early Mac Intel machines using the Core Duo based architecture rather than the newer Core 2 based architecture should rename the pcgameess-EARLY-INTEL-MAC-COREDUO-ONLY.exe binary to pcgameess.exe as this version is better optimized for these early Mac machines (just rename the existing pcgameess.exe binary to something like pcgameess-P4.exe rather than deleting it). This would also be the version to use for any Mac machine running the low power Atom processor, although there are currently no Mac machines using this architecture but this might well change in the future. Please note that the active binary must always be named “pcgameess.exe” exactly for any of the graphical job submission applications to work properly.

Mac specific installation notes and usage guide

Written by

Thomas Patko, Dept. of Chemistry & Biochemistry, California State University, Long Beach, USA

Edited by

Alex Granovsky, Dept. of Chemistry, Moscow State University (MSU), Russia

Nick Greeves, Dept. of Chemistry, University of Liverpool, UK

Firefly icon and logo original design by Lyndsey Vernon, Dept. of Chemistry, University of Liverpool

The PC GAMESS/Firefly Project Team

Last Revised February 16, 2009

Corrections to this Mac specific installation guide are encouraged, as are additions that would be mutually beneficial to future Mac users of the PC-GAMESS/FIREFLY QC program.