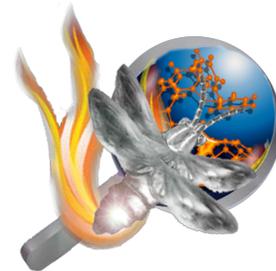


QUICK START AND USAGE GUIDE FOR INSTALLING AND RUNNING FIREFLY UNDER MAC OS X



Release 8.0.0 for Leopard, Snow Leopard, Lion & Mountain Lion

FOR MAC INTEL MACHINES ONLY!!

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

UPDATES FOR 8.0.0 RELEASE OF FIREFLY

The list of changes is detailed below for all modifications since the 7.1.G release. Read the documentation relevant for the 7.1.F and 7.1.G releases below the update guides for all common and unchanged information. These notes are ONLY relevant for the Firefly for Mac release (Linux and Windows releases may differ).

- Compatibility for Sandy Bridge & Ivy Bridge hardware. Support for Lion and Mountain Lion OS X. Mountain Lion users must install X11 following Apple prompts when a Firefly job is for the first time.
- The Snow Leopard, Lion and Mountain Lion installers are unified. Leopard users are strongly encouraged to upgrade to Snow Leopard for faster Firefly timings from the unified installer.
 - A Leopard compatible version (with slower timings) is available upon special request.
- Fixed error checking subroutine for all Drag & Drop Applications to ensure that user does not run job on more than the number of cores present. Changes to various OS X releases had broken the previous code.
- Added hyper-threading support to allow running jobs with double the number of physical cores and matching the number of logical cores present on any given machine. Please note that not all job types will benefit from hyper-threading. In addition, better or worse performance using a hyper-threaded type run will be based upon the particulars of your given hardware.
- Drag and Drop applications now support running jobs on up to 16 cores and with hyper-threading.
- Nearly complete rewrite of command line job launch script. Script name changed to firefly.sh to indicate major revision. Please review the comments in this shell script to see all new features and usage. Many thanks to Michael J. Harms of the University of Oregon for his contributions and collaboration on this front. Portions of his contributed code also appear as appropriate throughout the various drag & drop run applications.
- Benchmarks scripts now support running timing tests on 2, 4, 6, 8, 12 and 16 core machines.
- Various minor changes to enhance stability and compatibility and improve timings.
- Support for Tiger is dropped. Please upgrade your machine to at least (Snow) Leopard to use Firefly.
- All changes for 8.0.0 version as detailed in the main Firefly website are incorporated in the Mac release.

UPDATES FOR 7.1.G RELEASE OF FIREFLY

The list of changes and notes below for the 7.1.G update ONLY detail what has changed since the original 7.1.F release. For information common to the 7.1.F and 7.1.G release please read all materials after the update notes.

- The maximum memory that can be allocated per process for a Firefly job is now increased to 241 MW (1,838.7 MB of system memory). For an 8-core machine this means that the maximum aggregate memory is now raised to 14,709 MB of total system memory (provided that this is available naturally). The memory limit is identical under Leopard and Snow Leopard. Caution should be used when allocating more than 210MW of memory for any job type.
- Job timings are significantly improved for many types of job runs (by as much as 11% for parallel runs and as much as 22% for threaded runs under Snow Leopard as compared to timings from the Firefly for Mac 7.1.F release with DARWINE 1.1.18 under Leopard). Timings are overall best running the 7.1.G release under Snow Leopard, although timings are also generally slightly improved for Leopard as well.
- **RELATIVE TIMING EFFICIENCIES OF FIREFLY FOR MAC RELEASE 7.1.G (UNDER SNOW LEOPARD 10.6.2) VS. TIMINGS UNDER VISTA 64-BIT SP2 – ALL JOBS WERE 8-CORE RUNS**
 - The relative efficiencies for parallel type jobs using the benchmarks 1, 3, 5 & 6 as a guideline are on average 96.6% with a range of 89.5 to 100.0% efficiency for the individual timings.
 - The relative efficiencies for threaded type jobs using the benchmarks 2, 4 & MP4 as a guideline are on average 93.1% with a range of 89.5 to 95.9% efficiency for the individual timings.
 - The hardware information for the Dual Quad Mac Pro 2,1 used to obtain these timings is:
 - Running on Intel CPU: Brand ID 0, Family 6, Model 15, Stepping 7
 - CPU Brand String : Intel(R) Xeon(R) CPU X5365 @ 3.00GHz
 - CPU Features : CMOV, MMX, SSE, SSE2, SSE3, SSSE3, MWAIT, EM64T
 - Data cache size : L1 32 KB, L2 4096 KB, L3 0 KB
 - max # of cores/package : 4
 - max # of threads/package : 4
 - max cache sharing level : 2
 - As always the timings for your particular job may vary (slightly better or worse) depending upon your exact hardware, version of OS X, job type and the number of cores used. The timings for the benchmarks are simply meant as a general guideline of what you can reasonably expect.
- The dependency test is no longer required because:
 - X11 is included by default for both Leopard and Snow Leopard (this is a dependency).
 - DARWINE is no longer a dependency. Firefly for Mac 7.1.G release includes a custom build of WINE 1.1.33 optimized for use with Firefly of WINE is included in the main DMG installer. Note that this build is NOT recommended for use to run any Windows applications other than Firefly! Doing this may result in overloading your system memory!

- TextWrangler is still a “dependency” (or perhaps I should say STRONGLY recommended) if you wish to view your output files after a job has been launched, although theoretically you can omit this “dependency” and still run Firefly jobs successfully. BBEdit is also a perfectly suitable text editor for viewing output files and preparing input files. The absolute path to the TextWrangler binary is assumed to be /Applications/TextWrangler.app for all Run Apps.
- wxMacMolPlt is still a “dependency” (or another graphical visualization software that can readily open Firefly output files) if you wish to open our resulting output files in a GUI rather than looking at the text file only. This only is relevant if you use the job launch applications in the “AUTO-OPEN-IN-GUI” subdirectory of the main “RUN-FIREFLY-JOBS” directory or else any job launch app that has “GUI” in the app name. The path to MacMolPlt is assumed to be /Applications/MacMolPlt/wxMacMolPlt.app although you can change this in the corresponding internal configuration to something else if you wish (note that you would need to change ALL of the international configuration files in this case). Details about the differences between these two sets of job launch apps are detailed in the original 7.1.F release documentation below.
- A new set of job launch apps is now available that supports loading external basis sets and can found within the “RUN-FIREFLY-JOBS” directory as the “LOAD-EXTERNAL-BASIS” subdirectory. In addition, the set of currently posted external basis sets from the Firefly website is included in the official 7.1.G release. Note that you must use the separately provided “STOP-FIREFLY-JOBS-EXTERNAL-BASIS” app to terminate jobs using these job launch apps that load the external basis sets.
- A Mac GUI native version of DATAGAM is provided in the new “Utilities” directory. This program allows you to extract selected data from a Firefly PUNCH file. In addition the source code and compiled commandline version is included for convenience as well.
- Tiger is no longer supported as of the Firefly 7.1.G release. Tiger users should continue to use the Firefly 7.1.F release (still available for download). For 7.1.F users, the DARWINE dependency (release 1.1.21) is now hosted on the Firefly for Mac website for download (mainly because it is no longer available from the previous download location).
- It is possible to install both Firefly 7.1.F for Mac (still available for download) and the new 7.1.G Firefly for Mac on the same machine. The only issue to be aware of is that there is a single (hidden) “.wine” set of configuration files in the root of your user directory. When DARWINE 1.1.21 is run for use with Firefly 7.1.F, configuration files suitable for this DARWINE build are set. When the custom build of WINE 1.1.33 optimized for use Firefly 7.1.G is called (this is bundled in the new release), configuration files suitable for this custom WINE build are set (see notes inside the “WINE” subdirectory for details).

MAIN (COMMON) FIREFLY 7.1.F QUICK START GUIDE

Thank you for your interest to the Firefly package! Please follow the quick start guide below to get your installation up and running. Please note that this quick start guide is not intended as a replacement for the more complete install and usage guide that will give more extensive and complete information about the installation and usage of Firefly on Mac. If you should encounter any problems with the quick start guide, please refer to this more complete document. In addition, this guide assumes are reasonably good working knowledge of the current documentation and that the users reading this guide only need the specifics of how to run the Mac implementation. If this is not the case, please review the relevant posted documentation as necessary and appropriate.

To view the most current version of 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 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.

STAGE 1: Getting the “Dependencies” Installed

As mentioned in the 7.1.G updates at the top of this guide, there are no longer any “dependencies” in the strictest sense of the word (in other words the 7.1.G release of Firefly for Mac will run perfectly without installing any additional software). As such the TextWrangler and wxMacMolPlt programs are simply “strongly recommended” to be installed as they are useful, available free of charge and preconfigured as the default text editor and graphical visualization software for the job launch applications.

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 in general. The free program TextWrangler is suitable for purposes and is the default text editor for the job launch applications.

<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.

Installing wxMacMolPlt

The wxMacMolPlt program is a good choice to prepare Firefly input files and to visualize the result of the completed job output files. Other graphical input file generation software is available as well as other program to visualize the results of output files. wxMacMolPlt is configured as the default GUI program for the AUTO-OPEN-IN-GUI job launch applications, although this setting can be readily changed (please read later section for details). Precompiled binaries are available and should be installed into the default /Applications folder.

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

STAGE 2: Getting a License code to run the Mac DMG installer

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

First, carefully read the contents of the enclosed "License.pdf" file, which contains the Firefly Quantum Chemistry Package License Agreement (or else download it from the public FIREFLY project website). If you do not agree with all the terms of this License Agreement, you cannot use the 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 "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 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 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 binaries themselves as stated in the License Agreement.

STAGE 3: Run the DMG installer

The newest official release will be posted on the 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 STAGE 2.

STAGE 4: Let's run some jobs

If you are so inclined, benchmarks tests are provided to ensure that you system is running normally and to compare your timing with other Mac users (this is useful to perform and more detailed instructions on running the benchmarks are contained in the complete install and usage guide). For the impatient or excited, we will proceed in this quick start usage guide directly onto instructions for how to run jobs. I shall presume that you are familiar with how to create input files suitable to run on 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 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 subdirectory of the main /Applications/Firefly install directory.

A good idea, for example, it to copy the RUN-FIREFLY-JOBS folder to whatever location is most convenient for your computer and to run your jobs from there. There are a total of six Mac job submission run

applications, one GUI job termination application and one shell runscript 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
runfireflymac.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-2-Cores-GUI
Run-Firefly-Job-4-Cores-GUI
Run-Firefly-Job-8-Cores-GUI
Run-Firefly-Job-Interactive-Batch
Run-Firefly-Job-Interactive-GUI

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 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 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 (although 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-2-Cores-GUI

Run-Firefly-Job-4-Cores-GUI

Run-Firefly-Job-8-Cores-GUI

Run-Firefly-Job-Interactive-Batch

Run-Firefly-Job-Interactive-GUI

STOP-FIREFLY-JOBS

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 conventional 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 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). Then 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”. The STOP-FIREFLY-JOBS application is only supported for Leopard, Snow 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 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 Firefly for Mac implementation, the file I/O limitations may limit the efficiency of such jobs.

METHOD 2 (Use the commandline `runfireflymac.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 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:

```
./runfireflymac.sh input1.inp output1.out 2
```

```
./runfireflymac.sh./input1.inp ./output1.out 2
```

But the following usage **WILL** work:

```
./runfireflymac.sh /Users/tpatko/Research/Firefly/input1.inp /Users/tpatko/Research/Firefly/output1.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 runsript version offers you the option to configure your runsript extensively. Detailed instructions on how to use this command line runsript are provided in the notes of the script itself. Simply open this shell script file with TextWrangler and read the usage notes carefully. Be careful not to modify the script unless you are certain of the changes that you are making as your changes may cause the script to not function normally. Most users will not need to make any changes to this runsript, but rather just read the usage instructions and submit the appropriate three parameters to run the script. The function of the graphical run applications and command line runsript is identical (the command line runsript also automatically open the resulting output file in TextWrangler, for example).

METHOD 3 (Pure command line execution string)

To run a Firefly job from a command line without any script or run application, just use the following guideline:

```
/Applications/Firefly/WINE/bin/wine /Applications/Firefly/firefly.exe -osx ***
```

*** After the `-osx` switch followed by a space, please follow all of the normal FIREFLY command line options as detailed in the enclosed in the readme folder and the online manuals/documentations. You can use the provided runsripts as a quick guideline for running some types of jobs. Consult the relevant manuals and the `runfireflymac.sh` commandline runsript 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 FIREFLY UNDER THE SPECIALLY COMPILED WINE BUILD OPTIMIZED FOR USE WITH FIREFLY

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 241MW (1,838.7 MB of system memory) under the currently supplied custom build of the WINE 1.1.33 included with the Firefly for Mac 7.1.G release. On an 8-core Mac Pro this means a maximum of 14,709 MB of total system memory (~14GB+) total memory can be allocated for an 8-core parallel run job. For Mac machines with 2-cores total, this would be mean that up to 3,677.4MB (~3.5GB+) total memory can be allocated for a 2-core parallel job and for Mac machines with 4-cores total, this would be mean that up to 7,354.8MB (~7.15GB+) total memory can be allocated for a 4-core parallel job. Multiple concurrent parallel jobs are supported (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 FIREFLY under Wine that may result is slower runtime for non-direct (conventional and semi-direct) SCF jobs.** Running Firefly under the custom build of WINE provided using direct SCF results is very little performance loss at all (even for parallel jobs up to 8 cores often running near 100% efficiency as noted in the previous updates section). Performance results obtained running under Snow Leopard (10.6.X) were better than those obtained running under Leopard (10.5.8).

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

Mac Native Applications to prepare input files and visualize output files

The following applications are testing and known to work on Mac OS X with 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 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 FIREFLY jobs run under Wine for Mac in WebMO proceeds just as it would if the 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 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, Sybyl .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>

Gabedit (FREE -- Native Mac application running under X11, must be compiled from source)

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

<http://sites.google.com/site/allouchear/Home/gabedit/download/compilation-under-macosx>

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

Chemcraft (Commercial with Trial Version)

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

Facio (FREE)

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

Crossover Wine for Mac (Commercial Implementation)

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

or Wine/Winebottler for Mac (Open Source Implementation)

<http://winebottler.kronenberg.org/>

If you have any difficulty running these programs under Crossover Codeweavers Wine or the Wine/Winebottler for Mac open source builds 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 input and output files on the Mac platform would be good to know.

Firefly for Mac OS X Quick Start Install and Usage notes

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

Firefly Project Team

Last Revised January 6th, 2013

*PLEASE REFER TO THE COMPLETE INSTALL AND USAGE GUIDE FOR ADDITIONAL DETAILS
SPECIFIC TO RUNNING FIREFLY ON MAC THAT ARE NOT COVERED IN THE QUICK START GUIDE!*

Corrections to this guide are encouraged, as are additions that would be mutually beneficial to future Mac users of the FIREFLY QC program.