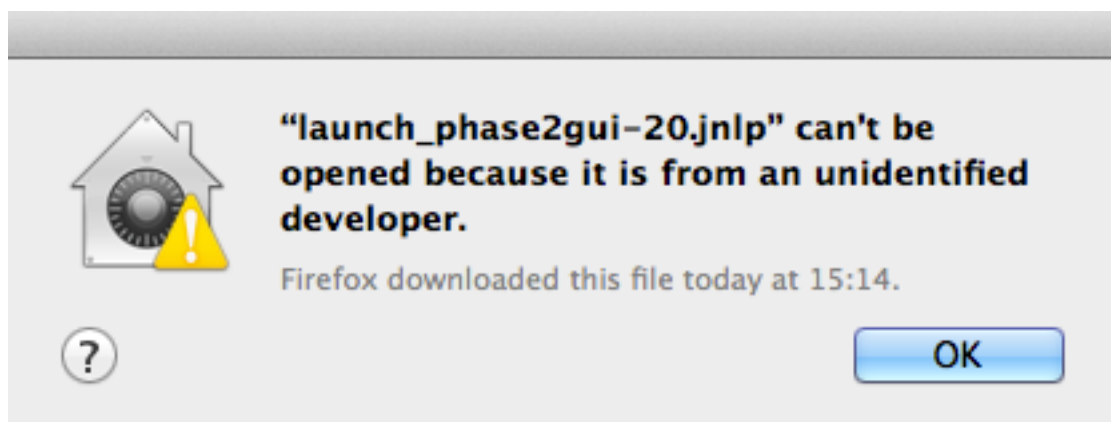


Running The Liverpool Telescope Phase 2 UI in OS-X Mavericks.

The Phase2UI will run on OS-X Mavericks. However, due to a security feature employed by Apple, Java applications have to be explicitly allowed to run in the security options of the OS (which are enforced by a system called OS-X Gatekeeper). If this selection isn't made then the process starting the application stops, sometimes without much explanation. Commonly, a dialog box similar to the following appears.



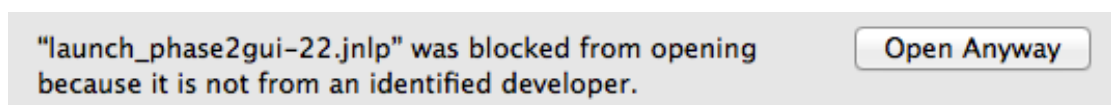
In order to run the Phase 2 UI, Java needs to be installed on the machine (see the section below entitled Installing Java on Mavericks if you haven't installed it already).

Unfortunately, it doesn't appear to be possible to run the Phase2 UI under the "identified developer" banner (see the section below entitled "Identified Developer - An Explanation" for more details if required).

Click the Apple symbol in the top left of your screen, and using the menus and tabs, select the following:

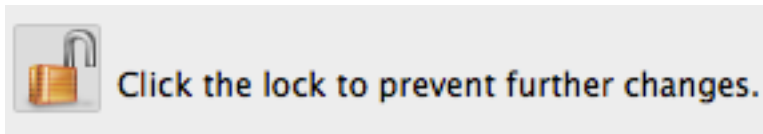
System Preferences -> Security & Privacy -> General tab

If you have already tried to open the Phase 2 UI using the web-start link on our web-site then there is a good chance that the Gatekeeper is reporting to you in this dialog that the following is the case:

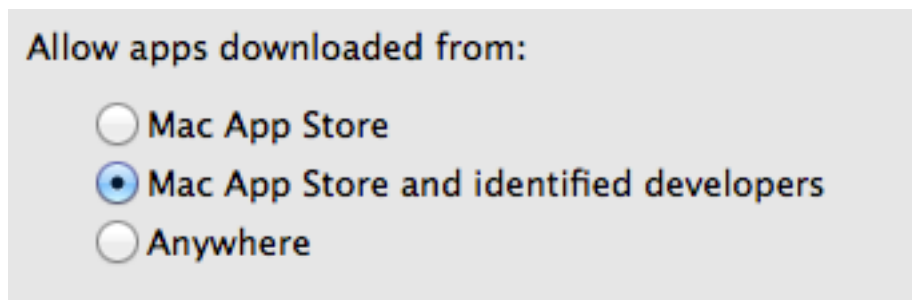


If this is the case, then you can click the “Open Anyway” button to open the application. However, be warned that if you do this, then you have to click that button every-time you try to run the application (unless you do the following).

If you haven’t run the application yet, use the lock symbol at the bottom of the dialog to make the bottom three radio button options available for editing.



This then enables the radio buttons:

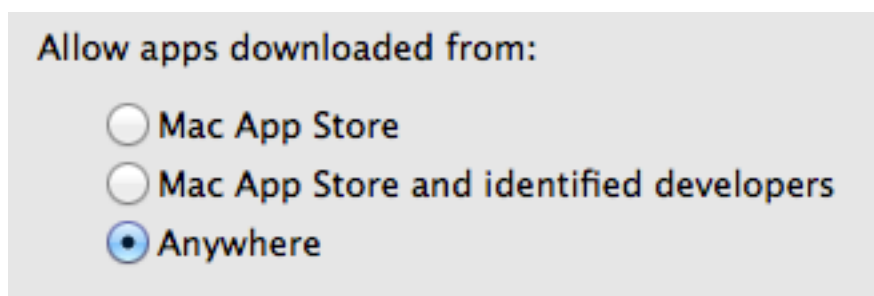


Select Anywhere (if you are happy to run applications from anywhere on your Mac, without the Gatekeeper system warning you or blocking the application).

The following dialog will appear:



Click “Allow From Anywhere”. The Anywhere radio button will then be selected.



Now, try to run the Phase2 UI again.

A different method (added later).

The above explanation consisting of changing the Gatekeeper security settings to allow apps to run from anywhere has its dangers. The Gatekeeper's settings are by default configured to stop you from running un-trusted applications. The author could argue about whether Apple's definition of trust is a particularly complete one, but having a software layer that stops unsigned code from running by default certainly has its merits.

An alternative to the above method is to use the Webstart link on the ARI Phase 2 web page just the once. This will then download a single instance of the `launch_phase2gui.jnlp` file to your computer. The Gatekeeper security model allows an application to be trusted once, after which it is considered trusted. In this scenario, a single instance of the `launch_phase2gui.jnlp` file is considered to be an instance of an application.

So, if you leave the Gatekeeper with its default settings (i.e. Mac App Store and identified developers) and then click on this button on our website:



A `launch_phase2gui.jnlp` file will be downloaded to your computer (probably into the `~/Downloads` directory). If you then double click or Open this file, the Phase2UI will fail to run, however a message similar to the following will be displayed on the Gatekeeper console (at System Preferences -> Security & Privacy -> General tab) will be displayed:

"launch_phase2gui-22.jnlp" was blocked from opening
because it is not from an identified developer.

Open Anyway

If you then click Open Anyway, the Phase2UI should run. If you always use this downloaded `launch_phase2gui.jnlp` file, which has been given permission to run by the Gatekeeper, then you can run the Phase2UI without having to set the Gatekeeper to allow all applications to run.

The danger of this, is that the `.jnlp` file you have downloaded is essentially a path descriptor for the libraries that make up the Phase2UI, and so if that path is changed (e.g. a new `.jar` file is added to the package) then trying to run the Phase2UI from this file may result in an error. However, in order to mitigate this rare occurrence, we have employed a versioning check on the server side, so we consider this to not be an issue, and this is now the method that we recommend users use.

Installing Java on Mavericks.

In order to install Java, go to:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

(Alternatively, you can of course do a web search for “Download Java for Mac OS X”).

When the downloads page appears, click on the following icon:



Click the “Accept License Agreement” (if you do) and then click on the link to:

[jdk-7u45-macosx-x64.dmg](#)

Java SE Development Kit 7u45		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.		
Product / File Description	File Size	Download
Linux ARM v6/v7 Hard Float ABI	67.67 MB	jdk-7u45-linux-arm-vfp-hflt.tar.gz
Linux ARM v6/v7 Soft Float ABI	67.68 MB	jdk-7u45-linux-arm-vfp-sflt.tar.gz
Linux x86	115.62 MB	jdk-7u45-linux-i586.rpm
Linux x86	132.9 MB	jdk-7u45-linux-i586.tar.gz
Linux x64	116.91 MB	jdk-7u45-linux-x64.rpm
Linux x64	131.7 MB	jdk-7u45-linux-x64.tar.gz
Mac OS X x64	183.84 MB	jdk-7u45-macosx-x64.dmg
Solaris x86 (SVR4 package)	139.93 MB	jdk-7u45-solaris-i586.tar.Z
Solaris x86	95.02 MB	jdk-7u45-solaris-i586.tar.gz
Solaris x64 (SVR4 package)	24.6 MB	jdk-7u45-solaris-x64.tar.Z
Solaris x64	16.23 MB	jdk-7u45-solaris-x64.tar.gz
Solaris SPARC (SVR4 package)	139.38 MB	jdk-7u45-solaris-sparc.tar.Z
Solaris SPARC	98.17 MB	jdk-7u45-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	23.91 MB	jdk-7u45-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	18.26 MB	jdk-7u45-solaris-sparcv9.tar.gz
Windows x86	123.49 MB	jdk-7u45-windows-i586.exe
Windows x64	125.31 MB	jdk-7u45-windows-x64.exe

This will download a .dmg file which when opened will mount a virtual drive on your computer. Within it there is a file called JDK Update 45.pkg, which when

run, will install Java on your machine. If a newer version of Java is available for OS-X, then make sure to use that one instead.



Identified Developer - An Explanation

Applications developed and made available to users by the LJMU Astrophysics Research Institute are code-signed. This provides a degree of trust, so that the user knows that applications downloaded and run from our web-site have been written by us, and have not been tampered with by third parties.

For a long period, this form of trust building was acceptable to Apple when it came to running many types of applications. After a set of security scares in the 2011-2013 period, Apple decided to only allow applications to run on it's OS-X platform if those applications were deployed by registered Apple Developers. This is a reasonable security measure, however code signed applications (which also operate using this kind of trust-model) are not acceptable to Apple, and software-suppliers still have to have applications signed by Apple itself. This is a reasonable policy, and becoming an Apple Developer costs in the region of \$99 a year.

Unfortunately, due to a large number of security scares involving Java in 2012 and 2013, Apple have forced Java applications to always be considered un-trustworthy (even if they are code signed). At the time of writing, the author has not come across a method of getting a Java application through the Apple code-signing process, such that it can be run without users of OS-X Mavericks machines being required to turn off the Gatekeeper system.