Contents

Introduction

There are a variety of ways to be “open” in the technology world today. For amateur astronomers who would like to depart from the realm of prepackaged, commercial software, the term open is synonymous with “free”, “community-supported” and “open source”. The most important component of the three is “community-supported”, which implies both that the application is in widespread use in the astronomical community, and that users can readily find support from other users.

My journey in astronomy has been paralleled by my journey through the world of technology. My first telescope, a 3” Tasco reflector on a rickety mount, was replaced by a 13.1” Dobsonian of the same design as the Coulter Odyssey telescope popular at the time (in fact my scope used Coulter optics). At the same time my parents had finally given in and bought my first computer, an Apple //e with an amazing 64k of RAM and 126K of storage on a single density, single sided floppy disk.

This was the mid-eighties of course, so astronomical computing was in its infancy. I did pick up a BASIC program named Astro-Aid, which contained routines to calculate the AltAz coordinates of objects based on the time, Right Ascension, and Declination of the object, which combined with a level and protractor on my scope got me closer to my objects of interest.

But of course, “push-to” wasn’t quite where I wanted to go – I wanted my telescope to find the objects and track them so I could see more each night, especially when I was out under dark skies! And take amazing pictures! That objective would lead to more than one redesign of my dobsonian to add drive motors with a simple board from Mel Bartels and a DOS based laptop to run it. Eventually I got my hands on a DS114 scope with commercial goto through the Autostar system and now the endeavor became making it work, instead of creating it from scratch.

Over time the technology has reached the point where it is reliable and capable of automating a telescope to a significant degree, but as capability has improved, complexity has started to increase as well. This book aims to bring that complexity down to a level where an average non-programmer can configure and operate his own robotic telescope.

Did you know you are buying a subscription to this book rather than a one-time item? So feel free to head over to my web site, enter the account information emailed to you when you registered the book, and download the latest copy! The site address is: https://www. openastronomy.ca. Enjoy the book!

Note from the Author:

Did you get this book for free? Yeah, I know, stuff happens. I would like you to consider supporting me in the continued development of this book as well as my supporting web site through either purchasing this book (and getting a subscription to new and revised content!) or at least toss me a donation through the site at www.gordtulloch.com. I would appreciate it very much!

Regards,
Gord

Chapter 1 - Assembling a Tool Set

One of the first tasks before us is to assemble a tool set which we can use to build our various systems to use for astronomy. The software and hardware detailed in this book are the most current "best of breed" for open astronomy software at the time of this writing. They offer all the capabilities of commercial software at low or no cost to the user. Check back often since the tool set will evolve over time!

The toolset we'll be focusing on in this book is as follows:

- Hardware – Raspberry Pi running Astroberry
- Planetarium Software – KSTARS
- Telescope, Observatory operation – KSTARS / EKOS
- DSO Image Acquisition, Guiding – KSTARS / EKOS / PhD2
- Planetary Image Acquisition – EKOS / FireCapture / AstroDMX Capture
- DSO Image Processing – SIRIL / GIMP
- Planetary Image Processing – WinJuPos, PIPP, Autostakkart, Registax
- Platform for processing – Linux, Windows 11

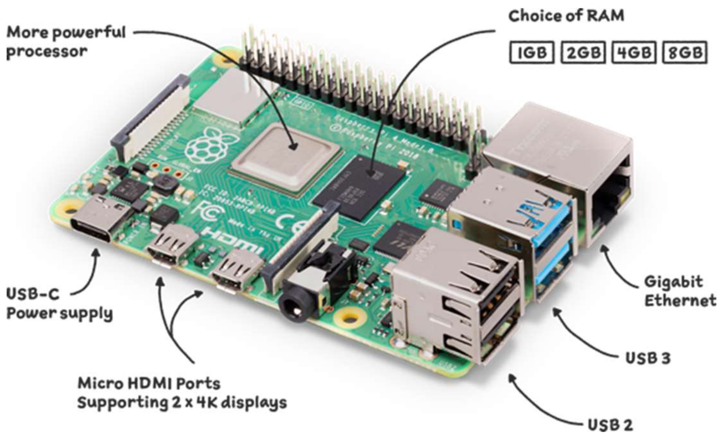
While this tool set is intended to keep the scope of this book manageable it's not fixed in stone – for example you can run most of this software in Windows, Mac, or Linux. In some cases (for example, INDI) a particular operating platform (in this case some Unix derivative such as Linux or MacOS) is required but most software discussed is cross platform.

I should note that some open source hardware, particularly Raspberry Pi is a little difficult to get hold of right now at a reasonable cost and timeframe. I will look at some alternatives in Appendix B.

Also where other open source software is available I'll add a section at the end to mention it and eventually expand future sections of the book to include it.

Hardware – Raspberry Pi 4

The Raspberry Pi (https://www.raspberrypi.org) is the product of a UK based foundation that aims to put inexpensive computing in the hands of people all over the world. The Raspberry Pi (henceforth abbreviated RPi) comes in quite a few flavours, from the integrated keyboard model reminiscent of old 8 bit computers in the dawn of the computer age like the RPi 400 to tiny boards like the Pico and Zero models. While at any given time there's quite a few models available we'll select the new RPi 4 Model B which offers a fast quad core processor, lots of available RAM (up to 8 GB) and lots of ports for connecting peripherals.



The core board needs to be accessorized to function with a case and power supply. A HDMI monitor and USB mouse / keyboard are for initial setup and configuration although once you've loaded the operating system to the device you can connect to it wirelessly from your computer or tablet via a built-in wireless hub. The board also includes a gigabit Ethernet port for direct connection to your network, which I tend to prefer over wireless unless out in the field.

RPi boards use a microSD card of up to 128gb for storage which is reasonable until you start saving large amounts of data such as for deep sky imaging and especially planetary imaging. I generally get a USB3 hard drive of 250gb and set it up as bootable so I have lots of space and faster performance. I've been out in the field and experienced failures to boot using microSD cards so a USB3 hard drive also improves reliability. Get one that's small and Velcro it to the top of the case.

There are many many suppliers of Raspberry Pi kits besides the manufacturer, just be sure to get a case, heat sinks for the chips on the board, a fan (RPi4s can get hot!) and a power cable as well as a micro HDMI adapter cable if you want to connect a monitor (and you will!) A 32gb microSD card is fine although if you choose not to use a SSD hard disk you will be a bit cramped on image storage. Here's who I generally get mine from:

https://www.canakit.com/raspberry-pi-4-starter-kit.html

Operating Systems – AstroBerry Server

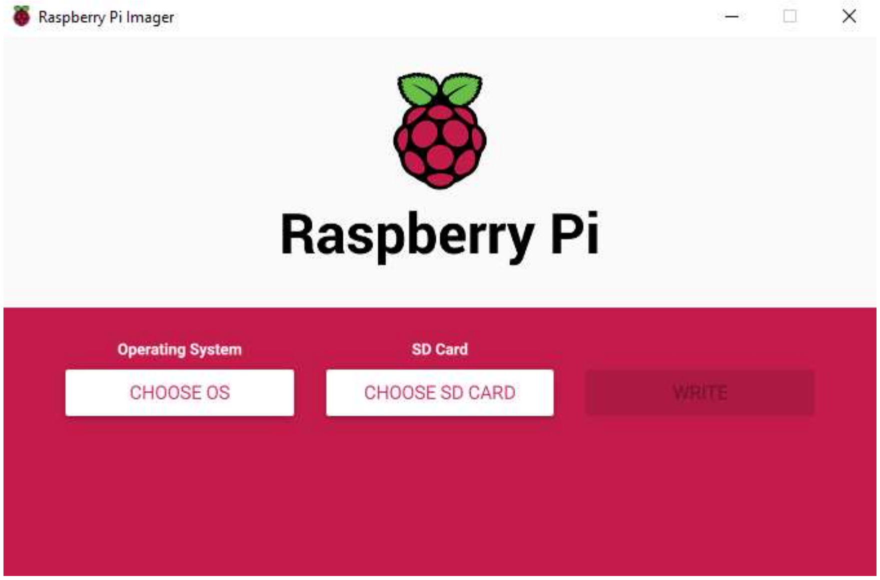
Once you have your RPi you're going to need to load an operating system onto the microSD card or (hopefully) USB3 hard drive so you can operate the device. The process for doing so is pretty simple and tends to be a quick way to resolve problems and do upgrades as well, since the built-in operating system update process can occasionally cause problems. As we will see you will need a small microSD card in any case.

I should note that I am not religiously devoted to purely free, open-source software. Where a company has made a significant value add to the open-source toolset I'll happily fork over a few dollars, particularly if the author is releasing software as open source as well.

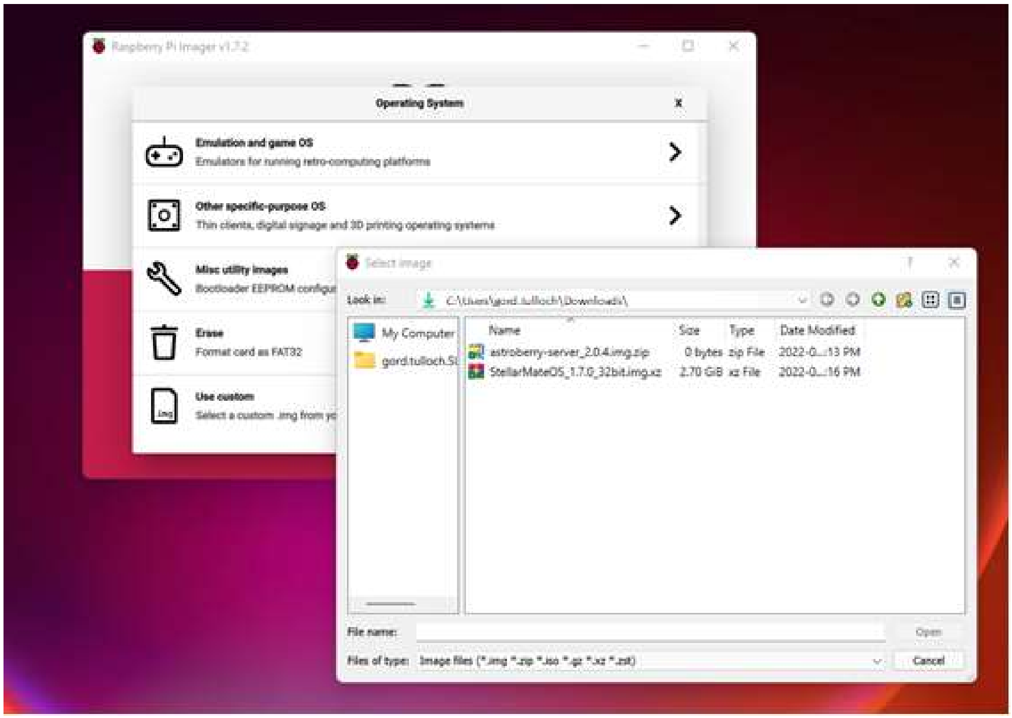
A good example of this is Jasem Mutlak of Ikarus Technologies and his Stellarmate product. This is a Ubuntu derived Linux distro for Raspberry Pi that includes all of the major open source tools (including Matlak's KStars / EKOS software) but adds a nice mobile device interface for IOS/Android as well as a cloud based management interface (EKOS Live). The product is $49US for just the Stellarmate OS or $229USD for a Raspberry Pi 4B based Stellarmate Plus appliance (EKOS Live is a $4.99USD per month subscription). If you're not interested in the mobile and cloud capabilities then Astroberry Server (https://www.astroberry.io) has the same basic tool set in a fully open source, free download. Stellarmate produces a great manual for Stellarmate and KStars so I won't cover much about the product in this book.

Here's step by step instructions on loading the operating system software:

1. Download a disk image from the vendor site (astroberry.io). The image is a compressed image of the disk with the extension img.zip (Astroberry).
2. Download the Raspberry Pi Imager from https://www.raspberrypi.com/software/ for your computer operating system of choice. This program will write the disk images onto your microSD card or USB3 hard drive.



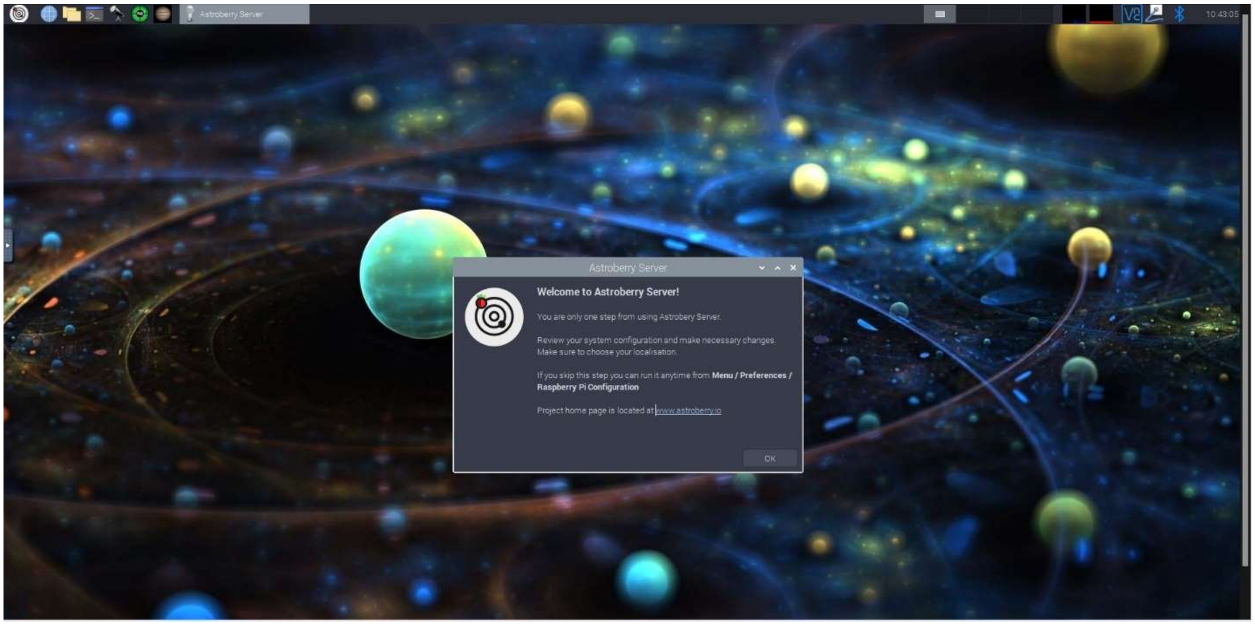
3. IF USING A USB3 hard drive:
    a. You need to configure your RPi4 to boot from the hard drive. This is simply a matter of writing a special disk image from the Raspberry Pi Imager to a microSD card (hopefully you got one with your kit?) Insert your microSD card into the adapter that came with your kit and insert into a USB slot on your computer. Run the Raspberry Pi Imager software and under Operating System scroll down to Misc. Utility Images and click to open the next menu. Select Misc. utility images, select Bootloader then select USB Boot. This will open a disk image that will change the RPi4 settings while it boots.
    b. Back on the main page pick your microSD card under Storage. Be careful to choose the option with the right amount of storage (e.g. 32gb) and not a hard drive as the storage will be overwritten! Click on Write to download and write the image to the microSD card.
    c. When complete insert the microSD card into the RPi4 and power it up by inserting the USB-C power cable into the power port. If a screen is connected it will flash green, letting you know you have successfully updated the boot settings on your RPi4.
4. In the Raspberry Pi Imager software you can choose to load your Astroberry Server image to your microSD card or USB3 hard drive by clicking on Operating System and scroll down to Use Custom and click it. You can then select either Astroberry from your Downloads folder.


Raspberry Pi Imager v1.7.2
Operating System
Emulation and game OS
Emulators for running retro-computing platforms
Other specific-purpose OS
Thin clients, digital signage and 3D printing operating systems
Misc utility images
Bootloader EEPROM configu
Erase
Format card as FAT32
Use custom
Select a custom .img from yo
Select image
Look in:
C:\Users\gord.tulloch\Downloads\
My Computer
gord.tulloch.S
File name:
Files of type:
Image files (*.img *.zip *.iso *.gz *.xz *.zst)
Open
Cancel

Name	Size	Type	Date Modified
astroberry-server_2.0.4.img.zip	0 bytes	zip File	2022-0...13 PM
StellarMateOS_1.7.0_32bit.img.xz	2.70 GiB	xz File	2022-0...16 PM

5. Select the microSD card or USB3 hard drive under CHOOSE SD CARD and click Write. Once the image is written insert the microSD card into the RPi4 or plug the USB3 hard drive into one of the blue USB3 ports on the RPi4.
6. Power up the RPi4 by inserting the USB-C power cable into the power port and you should now see the opening screen of your operating system!

Getting Started With AstroBerry Server



Astroberry Server

Welcome to Astroberry Server!

You are only one step from using Astrobery Server.

Review your system configuration and make necessary changes.
Make sure to choose your localisation.

If you skip this step you can run it anytime from **Menu / Preferences / Raspberry Pi Configuration**

Project home page is located at www.astroberry.io

Starting up with an AstroBerry image on your MicroSD card or SSD will boot the AstroBerry operating system and present you with an operating system screen similar to Windows as shown above. The OS runs the Raspberry Pi configuration utility that allows you to change the configuration of your Raspberry Pi – for now, you can just click on ok to close the utility (it can always be run from the menu at top left under Preferences/Raspberry Pi Configuration.

If you are prompted for a password at any time, the default is “astroberry”.

Network Connections

Astroberrys come preconfigured to use the built-in network adapter on the RPi to create a local wireless access point so users can connect to the device using a laptop, tablet, or phone. This is very useful out in the field obviously since it removes the necessity to connect via some form of cable you end up tripping over in the dark.

On your device, bring up the network browser and you will see a new Wireless network named Stellarmate or Astroberry depending on your OS. If you select that network you will be prompted for a password (astroberry on Astroberry) and your device will connect to the wireless network on the RPi.

Unfortunately this means you won’t have Internet access since your RPi isn’t connected to the Internet – this can be a hassle obviously. While there’s no absolute reason why you need Internet while using Astroberry, often you run into tasks that would be made more convenient by having access. There’s a couple of ways to work around this:

- Rather than connecting to the RPi wireless, have the RPi connect to a different wireless network that allows Internet access. In your back yard this might mean your house wireless network or alternatively some public wireless service you subscribe to.
- If you have tethering activated on your cell phone, you could connect the RPi to your phone for mobile internet access.

- When I'm in my back yard I can use PowerLine ethernet, which is a way to use your home network from anywhere where you have access to electrical power. Since I run an extension cord to my telescope in the back yard, a little box that plugs onto the extension cord allows me to connect to my network over high speed ethernet, which is always much more reliable than wireless. A set of adapters like those in figure X are $60 on Amazon and support speeds up to a gigabit.

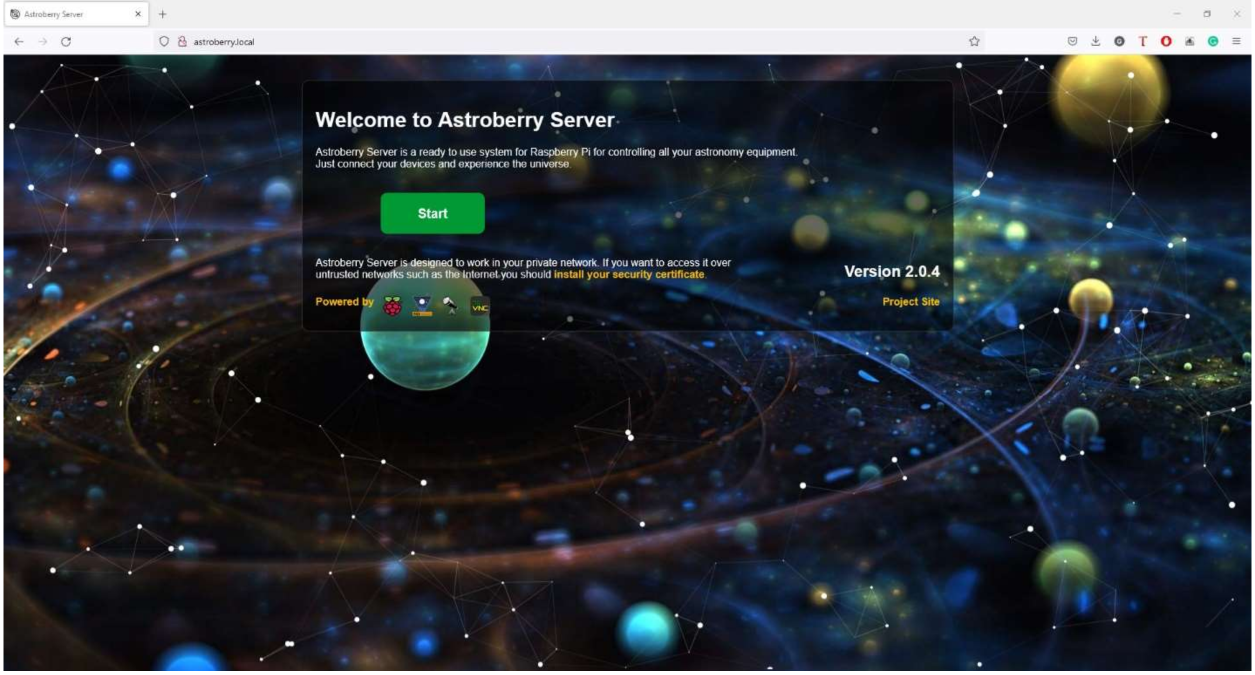
Once you are connected to the RPi over the network, you have lots of options for controlling your devices.

The address of your RPi for both Stellarmate and Astroberry will be broadcast as the local address stellarmate.local or astroberry.local (respectively) to access from a remote workstation or tablet.

Connecting via Remote Desktop

Remote desktop uses a tool named VNC (Virtual Network Computing) to connect to your desktop, show you what would normally be displayed on a locally attached screen, and pass back your mouse and keyboard actions to the RPi so you can control it. AstroBerry is preconfigured with a version of VNC named NoVNC that doesn't require a VNC client on your laptop, tablet or phone, or you can use a standalone VNC client on these devices. To connect to your device, open a browser and enter http://astroberry.local on the address bar. If all goes well, you'll connect to the web interface for the OS.

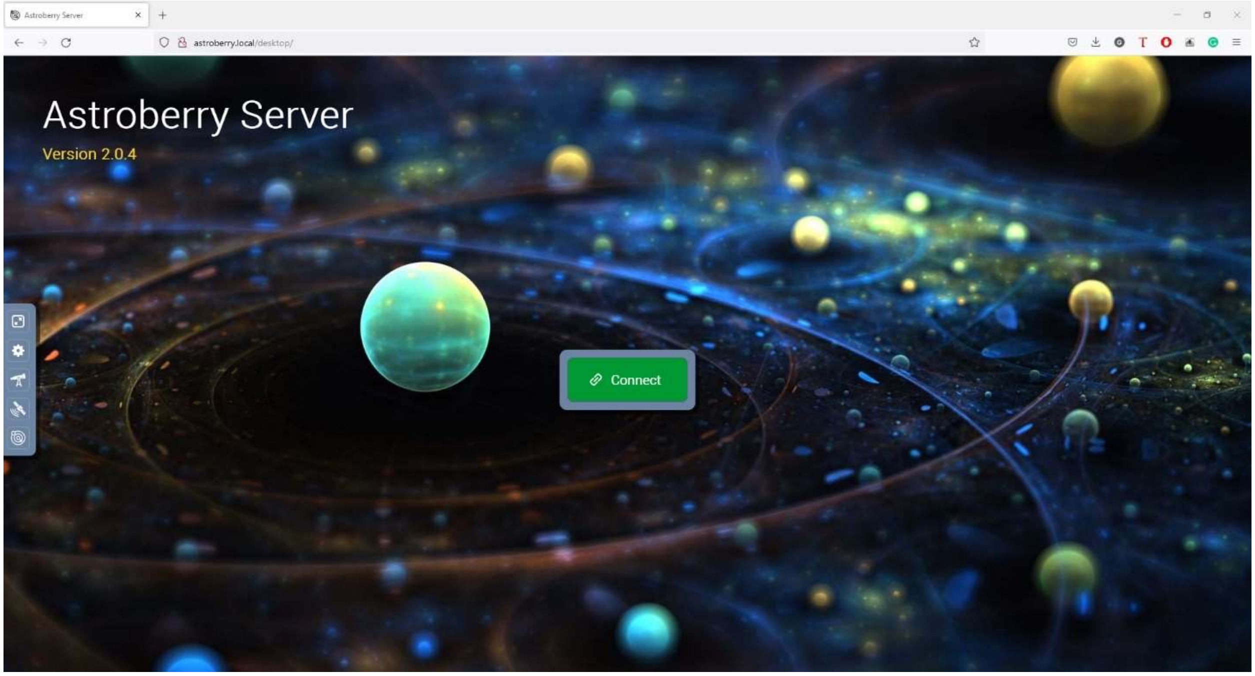
On an Astroberry OS the web interface simply takes you to a NoVNC login as shown below. Clicking on Start from the initial splash page presents a Connect button from NoVNC. You will be prompted to enter a password (default is astroberry) and you will be connected to the desktop.



Welcome to Astroberry Server

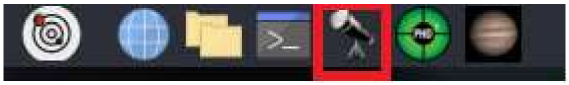
Astroberry Server is a ready to use system for Raspberry Pi for controlling all your astronomy equipment. Just connect your devices and experience the universe.

Astroberry Server is designed to work in your private network. If you want to access it over untrusted networks such as the Internet you should install your security certificate.



Configuring KStars and EKOS

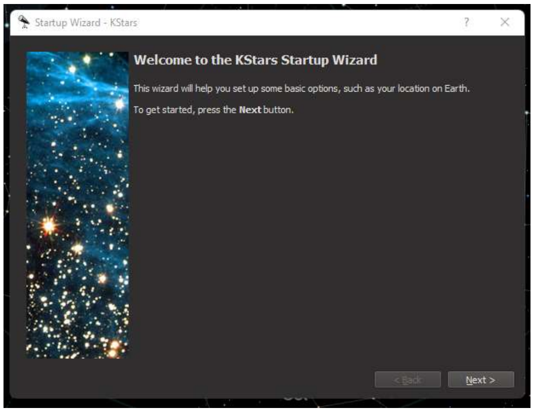
On AstroBerry KStars is not run by default but an icon is provided on the top toolbar (here highlighted in red):



The default location in KStars is however (unless you have a GPS module attached, which we'll cover later in this book) will be Greenwich England and the time will be Coordinated Universal Time (UT), so we do have some configuration that needs to be done before we can start using it. Kstars needs to know where you are and what time it is to display the correct information.

You'll notice that KStars is more of a star atlas program that is not photorealistic like planetarium programs such as Stellarium. These are different tools for different purposes – Kstars (and it's integrated imaging tool, EKOS) is primarily aimed at planning observations, finding objects, telescope control, and imaging operations. Since the RPi is more limited in processing and graphics capabilities than your desktop computer, it's a good idea to use the photorealistic tools on your desktop and keep the tools on your RPi focused on the task at hand. As I mentioned, Stellarium (www.stellarium.org) is a great tool available across multiple platforms.

KStars (https://edu.kde.org/kstars/) is free, open source software that's been around for many years as part of the KDE desktop project on Linux. It is now available cross-platform on Windows, MacOS (BSD), and Linux with a light version available for Android. It was originally written by Jason Harris but is currently being maintained by Jasem Mutlaq who has extended it with INDI/EKOS which provides device control for telescopes, focusers, filter wheels etc. Obviously as an open source project there have been many contributors over the life of the project, adding new features and fine tuning existing functions.



Welcome to the KStars Startup Wizard

This wizard will help you set up some basic options, such as your location on Earth.

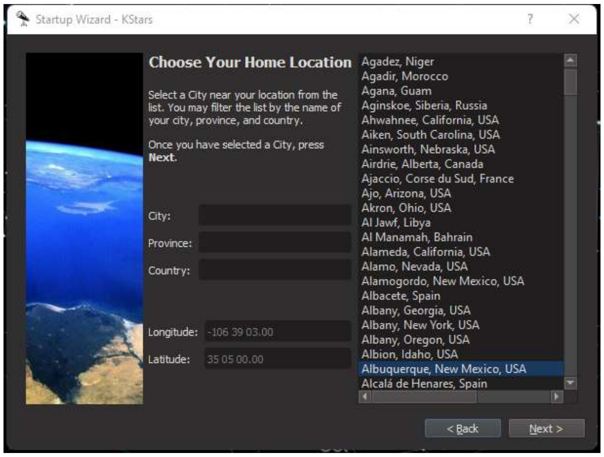
To get started, press the **Next** button.

KStars, in addition to star charting, also has lots of functions that will be covered in the next chapter but for now we'll configure the software for your area so the chart you see is correct.

First of all we need to set up our location – click on the Settings menu and then Startup Wizard. This will run a wizard that will prompt you for initial setup options and some additional data sources you can download.

Click Next on the splash screen.

The first screen prompts you to choose your home location. A scrolling list of preset cities is available at right - click on a city then enter a letter to reposition the list, or use the scroll bar on the right hand side to find the city closest to you. If your city is not listed you can choose a large city that is close to you. However, since you'll be using KStars/EKOS to position your telescope you probably want to be pretty exact on specifying your exact observing location. We'll fix this in the next chapter using EKOS and INDI.



Startup Wizard - KStars

Choose Your Home Location

Select a City near your location from the list. You may filter the list by the name of your city, province, and country.

Once you have selected a City, press **Next**.

City:

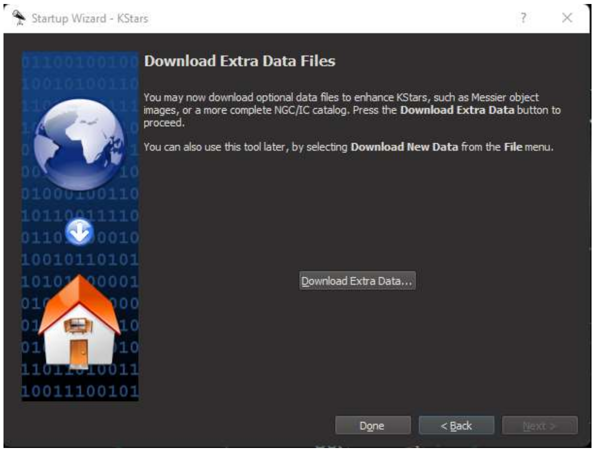
Province:

Country:

Longitude: -106 39 03.00

Latitude: 35 05 00.00

Agadez, Niger
Agadir, Morocco
Agana, Guam
Aginskoe, Siberia, Russia
Ahwahnee, California, USA
Aiken, South Carolina, USA
Ainsworth, Nebraska, USA
Airdrie, Alberta, Canada
Ajaccio, Corse du Sud, France
Ajo, Arizona, USA
Akron, Ohio, USA
Al Jawf, Libya
Al Manamah, Bahrain
Alameda, California, USA
Alamo, Nevada, USA
Alamogordo, New Mexico, USA
Albacete, Spain
Albany, Georgia, USA
Albany, New York, USA
Albany, Oregon, USA
Albion, Idaho, USA
Albuquerque, New Mexico, USA
Alcalá de Henares, Spain



Startup Wizard - KStars

Download Extra Data Files

You may now download optional data files to enhance KStars, such as Messier object images, or a more complete NGC/IC catalog. Press the **Download Extra Data** button to proceed.

You can also use this tool later, by selecting **Download New Data** from the **File** menu.

Finally you'll be prompted to Download Extra data files. These are entirely optional and can be downloaded at a later date so we'll sky this step for now and click on Done. KStars is now set up and ready to go!