Automating a Telescope's Control

Marcellin Nshimiyimana - August 10, 2010

In early 1600s, the use of a telescope was more mechanical [(1)] and astronomers had to pass nights outside, analyzing stars and studying celestial objects' movement. This work was very tiresome and less productive because, for example, if one needed to track a comet, they had to take notes about the comet's previous position, magnitude, brightness, and speed ...so that they would know where to find the comet the following day. However, nobody could stop doing this based on the importance that they gained from their observations. This importance includes predicting orbits and speed of usual and unusual objects in order to help in preventing some of cosmos-disaster, especially with Earth, human satellite, and spacecrafts. Therefore, they had to find ways to make their work more convenient by trying to incorporate electronic circuits to control telescope movement. This was achieved in the 1980s, when the first telescopes were made with fully computer controlled capabilities [(2)]. And in 1992 using Bernhard Schmidt's invention [(3)], Meade Instrument Corporation introduced their innovative LX200, which is the type of telescope that we use in **The Cordell-Lorenz Observatory. During the nine** weeks that I spent in the observatory, my supervisor and I needed to take full advantage of this wonderful instrument so as to improve observations and data intake in general.

My work started at the end of May and it focused on the computer-control system of the telescope. I worked full time during the night. However, my work was confusing and highly depended on weather in the first weeks because I had to learn the basics of

astronomy and build my understanding of how telescope's work, which would help me quickly understand Maxim DL and ASCOM [(a)], the two software programs that we were using to control the telescope, and then make some improvement of their use or write my own driver [(b)] if it was necessary. Besides this, I had to learn Visual Basic, a programming language in which most of the telescope drivers were written. Despite all the updates that were done on the ASCOM software, its installation document still had the source code for the drivers that came with the installation CD, so it became easy for me to understand how drivers are written and work[(c)].

After becoming familiar with the telescope control and seeing what the supervisor often did, I started the first project, which was to give me a feel of programming in a new language and so as to prepare me for the main telescope control project. However, I did not achieve my expectations because the first project did not succeed and the second had little difficulties, even if it worked.

The first project consisted of writing a program that would make copies of raw images at the same time as the image is taken. Therefore, I had to find a way to access the CCD Camera [(d)] interface at runtime, which would provide me with the raw image directory. However, this access caused interference that caused the Maxim DL to loose connection with the camera because of constructing a CCD Camera object that is already constructed and their no other ways to access the CCD interface without constructing a

new object in my program. Therefore, I changed the image directory access to manual access. That is, the user should manually set where to find raw images to copy and where to put copies and after the program would check for newer images to copy in the directory every two minutes. Since this automatic check stopped the "Operation System paste" option from working, we uninstalled it and I stopped changing it because of time.

In the second project, which focused in computer-telescope link, I changed the driver to the one that had all the methods/capabilities to fully control the telescope with the computer if the telescope was left power on. In fact, the former driver that we were using was an empty shell. That is, it contained telescope methods with no codes to execute when they were called [(e)], so when the user triggered a method, the telescope would run the built in code for that method and if not possible, the driver would throw an error message.

The lack of fully overriding the telescope methods prevented it from working efficiently. For example, if the observer has left the telescope with power off, in the R.A.-Dec position without unlocking clutches[(f)], when the telescope is turned on again, it could not find its position on its own because the driver had no way to save previous position, the coordinates systems in which the telescope is set, remains the same at all times and the driver is also not able to translate them to Altitude-Azimuth Coordinates, which depend on time and location of the observation site so as to update the position where

telescope is pointing. Therefore, I started writing the driver that contained most of the missing codes such as accessing telescope-GPS readings in order to retrieve the time and location to use in determining Alt-Az coordinates, translating R.A.-Dec coordinates back and forth to Alt-Az coordinates [(g)], move (slew) the telescope to a known position, using Alt-Az coordinates system, which will make the telescope point straight up before turning it off. To achieve this, I had to refer to the driver codes that were provided by the manufacturer of the software and fortunately, I found one of the drivers (ASCOM Meade Driver) that contained not only the inner codes that I was about to write but also others which were very helpful, such as to use the telescope in Alt-Az coordinates system. Although that driver was written in the old Microsoft Visual Basic language, I was able to understand the code, which made me stop writing my own, which required of me some of the programming skills that I did not have at that moment, and start figuring out how we could use ASCOM Meade Driver, the one fully written and installed to control the telescope.

By following the help settings, I wrote the steps that I needed to use the telescope with the new driver. It had two parking positions which could be used to put the telescope in a preferred position that the driver had to save before turning the telescope off and it was able to translate its position in RA.-Dec to and from Alt-Az coordinates system, which assured me that the driver use would allow the telescope to recognize its position on start up and it did of course. However, instead of using GPS readings, it required the

user to manually enter the site location, site elevation, and the telescope time was set to the computer time, which did not succeed because the computer that we were using had a program to synchronize the time, so the telescope time was at least two second-later than the computer time. Moreover, this was a big issue since we needed to use the time depended coordinates system because it could increase the position error of the telescope, but we had no other choice since the method that dealt with time could not be modified in any way. Despite those positioning errors, the parking was able to know its position at start up, so every time we were done observing, we would park the telescope using the computer before powering it off and at the following use, we only had to turn it on and then connect the software. In general, we have completely removed any mechanical control of the telescope and we should be able to totally control the telescope without going outside if we leave its power on.

Besides the computer control oriented project, I learned so many things in the astronomical field, such as star position in the sky and how this position is found, the difference between galaxies, nebulae, Milk Way…[(h)], and how unusual objects in the cosmos are discovered. Before I had no understanding of what a constellation is and how stars were named but after passing so many nights outside with experts in Astronomy, I was able to understand, for example that constellation or nebulae are named based on their shape and appearance and other objects like comets are named based the time of discovery and the Discover's name. Moreover, I did a couple of

publications of comets and minor planets, which are actually information about magnitude, position, brightness, speed, and movement direction of these objects, that are determined after processing a bunch of images and that information is sent to the Minor Planet Center, where it would contribute to improvement in orbital analysis prediction of these celestial moving objects.

In summary, when I got this internship, I needed to find a link between physics and computer science, which is my intended major, so I started working with the computer control of the telescope. However, since I had little understanding of astronomy, I was required to learn stars and other sky objects' behavior, which allowed me to understand how the stars can be the most convenient guides for traveling in unknown places. Moreover, I learned the importance of knowing many different programming languages. Because I had the drivers written in Visual Basic and an empty structured driver code written in C# [(j)], I had to refer to the fully written ones in order to complete the empty one. That is, I needed the basic understanding of these two languages in order to accomplish my work. Although I did not finish the driver, I learned the two languages, which followed Java, the first language that I knew.

Further information:

Automating a Telescope's Control

Marcellin Nshimiyimana - August 10, 2010

(a) **Maxim DL and ASCOM**: are the two software that we used to control the telescope; maxim DL contained the CCD Camera and Telescope interfaces and ASCOM was a platform that contained all telescope driver, it also has the complete list of Telescope and CCD Camera's properties and methods used when ones is writing any computer program that works with any of these tools.

(b) A **driver** is piece of software program that is used to control a hardware connected to a computer such as printer, disk drive, or keyboard…, it acts like a translator between the device and programs that use the device.

(c) How **drivers are written and work**: most programs access devices by using generic commands. The driver accepts generic commands from a program and then translates them into specialized commands for the device. For example, the telescope had its built in commands (methods and properties), so the driver must take computer commands and translate them into commands that the telescope must understand, so the driver has to own all methods for any telescope command.

(d) **CCD Camera** stands for charge-coupled device Camera, which is a great camera for digital imaging.

(e) **No codes to execute:** this driver had the method headers to call the built in commands of the telescope and it only had the error-throwing code that would be executed when the telescope could not respond to triggered command

(f) **Clutches:** the telescope has the **Right Ascension (R.A.) Lock** that Controls the manual horizontal rotation of the telescope and the **Declination (Dec.) Lock** that Controls the manual vertical movement of the telescope.

(g) R.A.-Dec. and Alt-Az coordinates system: any particular star has the same **Right Ascension (**RA) and **Declination** (Dec) for all observers on Earth, and that position remains the same, night after night. **Altitude** (Alt) and **Azimuth** (Az), on the other hand, are local coordinates for a star, which depends on observer's location and they change over just a few minutes as the star appears to rise, move across the sky, and set.

(h) **Galaxies, Milky Way, Nebulae: Galaxy** is collection of stars, gas, and dust bound together by gravity, **Milky Way** is the spiral galaxy that contains our solar system, and **Nebula** is a diffuse mass of interstellar dust or gas or both, visible as luminous patches or areas of darkness depending on the way the mass absorbs or reflects incident radiation.

(i) **Empty structured driver code:** the program that had all needed methods, functions, and properties headers with no code lines in their bodies.

Resources:

(1) **The Telescope, in Science.** The Galileo Project.

Retrieved from http://galileo.rice.edu/sci/instruments/telescope.html

(2) Buying a computer controlled Telescope? In Library. Company Seven. Retrieved from http://www.company7.com/library/cmpadvc.html

(3) Bernhard Schmidt and His Coma-Free Reflector, by Mayall, N. U. Publications of the Astronomical Society of the Pacific, Vol. 58, No. 344, p.282. Retrieved from http://adsabs.harvard.edu/full/1946PASP...58..282M