

Computer infrastructure for the Variable Young Stellar Objects Survey

Josh Walawender^a, Bo Reipurth^b, and Martin Paegert^c

^aUniversity of Hawaii at Hilo, Physics and Astronomy Dept., 200 W. Kawili St., Hilo, HI, 96720, USA;

^bUniversity of Hawaii at Manoa, Institute for Astronomy, 640 N. Aohoku Pl., Hilo, HI, 96720, USA;

^cVanderbilt University, Institute for Physics and Astronomy, 6301 Stevenson Center, Nashville, TN 37235, USA

ABSTRACT

An increasing number of remote or robotically controlled telescopes are using commercial "off the shelf" hardware and software. We describe a system which has been implemented in the Variable Young Stellar Objects Survey (VYSOS) project which uses simple, commercially available software and hardware to enable the quick restoration of observatory operations in the event of a computer failure.

Keywords: Young Stellar Objects, Pre-main Sequence Stars, Robotic Telescopes, Remote Telescopes, Redundant Computer Systems, Fault Tolerance, KernelPro, Virtual Machines

1. INTRODUCTION

The goal of the Variable Young Stellar Objects Survey (VYSOS) project is to make repeated photometric measurements of tens of thousands of stars which are members of nearby star forming regions to search for variable phenomena related to star formation.

Young stars undergo major changes during their evolution towards the main sequence. Additionally, the presence of a circumstellar accretion disk and its magnetic coupling to the central star leads to constant perturbations. It is therefore not a surprise that virtually all young stars are variable, although amplitudes and timescales differ from one type to another, and depend as well on the evolutionary stage. It is becoming increasingly understood that the rather static view of pre-main sequence evolution that has prevailed for many years is misleading, and that young stars cannot be understood without taking their continuous changes into account. Indeed, time-dependent phenomena may hold the key to an understanding of the way young stars grow and their circumstellar environments evolve, eventually leading to the formation of planetary systems.

While a number of general survey telescopes have come into operation in recent years, like the PanSTARRS project, they generally do not have a cadence of observations that match well with the nature of variability of young stars, which require observations on a nightly basis or more frequently. Also, many of the existing or upcoming survey telescopes are large (e.g., LSST), and hence saturate brighter stars. Among young stars, the brighter stars are the most important to monitor, because they are within reach of high-resolution spectroscopy, and thus the mechanism of variability can be analyzed in detail.

The VYSOS project will monitor and characterize the irregular variability of young stars resulting from star-disk interactions, discover and study new major FUor and EXor disk accretion events, discover and analyze eclipsing pre-main sequence binaries and thus derive fundamental parameters for young stars, measure the periodic signal from star spots rotating around young stars and thus understand the angular momentum evolution towards the main sequence. So far, millions of data points have been obtained in the year and a half that the survey has been underway.

Further author information: (Send correspondence to J.W.)

J.W.: E-mail: JoshuaMW@Hawaii.edu, Telephone: 1 619 928 4320



Figure 1. The 6 foot diameter VYSOS-5 dome (lower center) and the 12.5 foot diameter VYSOS-20 dome (right) at the Mauna Loa Observatory located at 11,135 feet above sea level on the North slope of Mauna Loa on the Big Island of Hawaii. The domes of observatories on Mauna Kea can be seen on the peak in the background.

To perform the photometric monitoring described above, the VYSOS project has built a pair of robotically controlled observatories using mostly "off the shelf" components for both the telescope hardware and the computer and software infrastructure which controls their operations.

In subsequent sections, we give a brief description of the VYSOS project hardware (Section 2), describe our remote port software solution (Section 3), describe our use of virtual machines (Section 4), and discuss how the combination of these two software tools allows us to have a very robust and fault tolerant observatory control infrastructure (Section 5).

2. VYSOS HARDWARE AND SOFTWARE

The VYSOS project consists of a pair of robotic telescopes located at the Mauna Loa Observatory (MLO) run by the National Oceanic and Atmospheric Administration. Established in 1957, MLO is dedicated to atmospheric measurements and is best known as the original site for "Keeling Curve" data which shows the increasing concentration of carbon dioxide in the atmosphere. Located on the north slope of Mauna Loa on the Big Island of Hawai'i at an elevation of 11,135 feet above sea level, MLO is often above the tropical inversion layer commonly present on the island and is thus also an excellent site for astronomical observations, with a similar number of clear nights as nearby Mauna Kea Observatory (which is 2,667 feet higher).

The VYSOS project consists of two telescopes. The first telescope, VYSOS-5, is a 5.3 inch (135 mm) aperture f/5.4 refractor built by Stellarvue Telescopes. Using a three element field corrector, it illuminates a very wide field of view with high quality star images. When coupled with a 16 megapixel Apogee Alta U16M CCD camera it provides a 2.9 degree square field of view (8.4 degrees²). In a standard survey visit of three 100 second exposures, it reaches an 5σ limiting magnitude in the r' filter of 17.7.

The second telescope, VYSOS-20, is a 20 inch f/8.2 Ritchey-Chretien telescope built by Ritchey Chretien Optical Systems (RCOS). Using a field corrector and an identical 16 megapixel Apogee Alta U16M CCD camera, it provides a 30 arcminute square field of view. In a standard survey visit of three 120 second exposures, it reaches a 5σ limiting magnitude in the r' filter of 20.5.

The observatory control software used in the VYSOS project is a combination of Astronomers Control Panel (ACP) and ACP Scheduler available from DC3 Dreams Software (<http://www.dc3.com/>). These two programs



Figure 2. The 5.3 inch f/5.4 VYSOS-5 telescope built by Stellarmv Telescopes (left) in its dome at Mauna Loa Observatory. The 20 inch f/8.2 VYSOS-20 telescope built by RCOS (right).

handle sequencing of observatory operations and control the individual hardware components either through the MaxIm DL (<http://www.cyanogen.com/>) software which controls our CCD cameras or through ASCOM (<http://www.ascom-standards.org/>) compatible hardware drivers. All of this software runs under the Microsoft Windows operating system (Windows XP Pro SP3 in our case).

In order to create fault tolerance and the ability to recover from computer failures remotely (without a trip to the observatory site), we have created a system which uses two additional software solutions (described in the following sections) to enable remote diagnostics and recovery.

3. REMOTE PORT ACCESS

In order to keep heat generating computers out of the domes of both telescopes, we have built a small weatherproof computer enclosure between the two domes. Both control-computers, along with computers to control the low light camera system, the all sky camera system, and to host the disk array for the data, are all kept in the enclosure and are thermally isolated from the telescopes. VYSOS-20 is particularly sensitive to thermal currents because the dome slit is quite narrow in relation to the telescope aperture and it suffers from dome seeing if heat is generated in the dome.

All of our critical components (CCDs, filter wheels, dome controllers, weather sensors, etc.) use one of two standard communication connections: USB or Serial (DB9). The disadvantage to directly plugging devices in to the control computer using long cable runs is that USB cables are limited in the specification to 16 feet, which means that to place the control computer in another building, we would need to use USB extenders. In addition, serial connections are susceptible to noise pickup on long cable runs unless isolated from other lines such as AC power.

Rather than install a series of USB and serial extenders, we chose to connect all of our components to a small, low power computer (the host computer) and then use two port sharing applications available from KernelPro (<http://www.kernelpro.com/>) to make the USB and serial devices available to the control-computers over our local area network (LAN).

The first application is called "USB over Ethernet" and makes the USB ports on the host machine available to a client elsewhere on the LAN (Fig. 3). The second application is "Advanced Virtual COM Port" (Fig. 4) and shares the serial ports in a similar fashion. The combined cost of the two software packages depends on how many ports are shared, but is around a few hundred dollars.



Figure 3. A screenshot of the client interface for the "USB over Ethernet" software. Available devices are shown with green symbols. In this case four available devices are shown: an Apogee Alta CCD camera, an Apogee filter wheel, our cloud sensor (which shows up as FT323R USB UART), and an SBIG guide CCD (which shows up as USBF-CAM Engine).

The host computers we chose to place in the domes with the telescopes are custom built to have four serial connections and four USB connections. They use a small compact flash card as the primary hard drive and use a low power VIA EPIA N700-05LE Nano ITX Mainboard with a 500 MHz processor. Because it does not have a traditional spinning disk hard drive and because the processor has low enough power consumption not to need a cooling fan, these computers have no moving parts and are very reliable. These computers would be underpowered if they were used to manage all of the observatory control operations and data handling, but their only role is to share the USB and Serial connections to other, more powerful machines on the local area network.

There are several advantages of using this port sharing system. The first is that additional diagnostics are available remotely. If a device is not responding on the control computer, one can immediately determine if there is a physical connection problem by logging in to the host computer and looking at the KernelPro software to determine if the connection to the device is active. If the connection is active on the host computer and the control computer, then the problem is likely to be a software or configuration problem instead of a physical problem with the connection.

Another important advantage is that if the control computer suffers from a fatal crash of some sort, a remote user can log in to another computer, connect it to the hardware using the KernelPro software, and take control of the system and place it in a safe state. While many remote or robotic observatory systems are designed to autonomously place themselves in a safe state if communication with the main control computer is lost, this provides an additional method to do this if those automatic systems also fail. This was particularly important for VYSOS-5 which at one point suffered from a particular crash in the dome control system which, when it occurred, would freeze the control computer so that it would not respond properly to unsafe conditions (such as weather), but would also freeze the dome controller and its "dead man timer" which was intended to close the dome automatically if the control computer failed. In this failure mode (now fixed), the dome would remain open until a human intervened. The ability to take control of the system from an alternate control computer

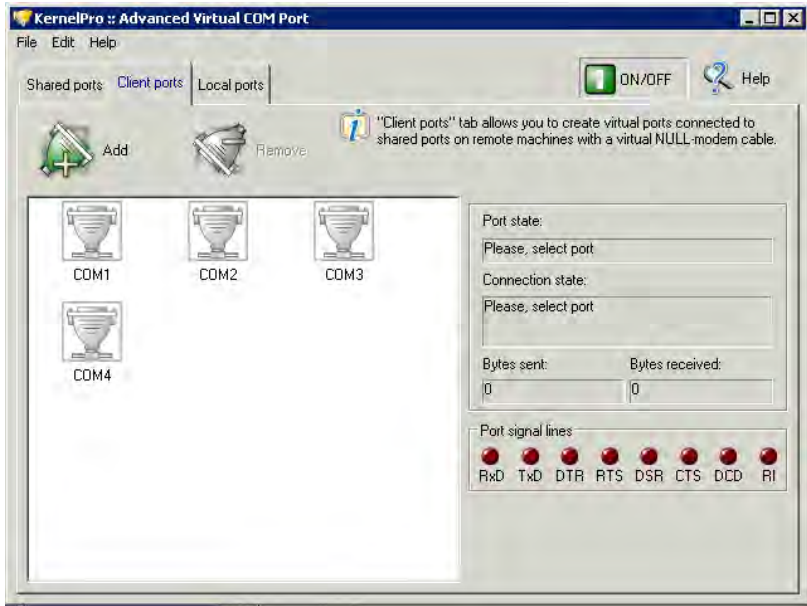


Figure 4. A screenshot of the "Advanced Virtual COM Port" software indicating that four serial ports (COM1 to COM4) are connected to this client machine. The symbols are grey because none of the ports is currently sending or receiving information.

was valuable for putting the system in a safe mode.

4. VIRTUAL MACHINES

While the remote port solution described in Section 3 allows a spare computer to take over control of the observatory system in case of a computer failure, in practice, this is limited to basic control to shut down the system. The reason is that a spare computer is unlikely to have complete and up to date configurations for the observatory control system. In the case of VYSOS, it would require two spare computers to be maintained, one for each telescope.

Rather than keeping physical machines online and up to date, we chose to use virtual machines for all of our telescope control tasks. Because virtual machines exist as files (or folders) on the host machine, they can be automatically copied and backed up on a networked storage device. If backed up regularly, then all recent setting and configurations for the observatory are automatically maintained in event of a failure. The system can be brought back on line by copying the backup virtual machine to the appropriate physical machine and booting it up.

In the case of VYSOS, we use three Mac Mini computers running Mac OS X 10.6 as the host machines. All three use VMWare Fusion as their virtualization software. Two of them run Win XP Pro virtual machines which control the VYSOS-5 and VYSOS-20 observatory systems. The third is available to step in as a back up and runs our all sky camera and acts as a data server to store copies of the raw data from each telescope.

The backups of the virtual machines are performed using `crontab` and `rsync` to copy the relevant file to a networked storage device on a daily basis. If there is a failure of the VM (i.e. an unrecoverable software fault) or a failure of the host machine (i.e. a hard drive crash), then the backup of the VM is copied to an alternate host and booted. This alternate host can be a spare physical machine or even a second VM running on the other host computer (Fig. 5).

Another advantage of using VMs is that if the failure is caused by a software freeze in the control system or operating system of the VM, the host machine is still accessible. For example, if the VM dies, the host machine can be logged in to remotely, the situation diagnosed, and the virtualization software restarted. If the crashed

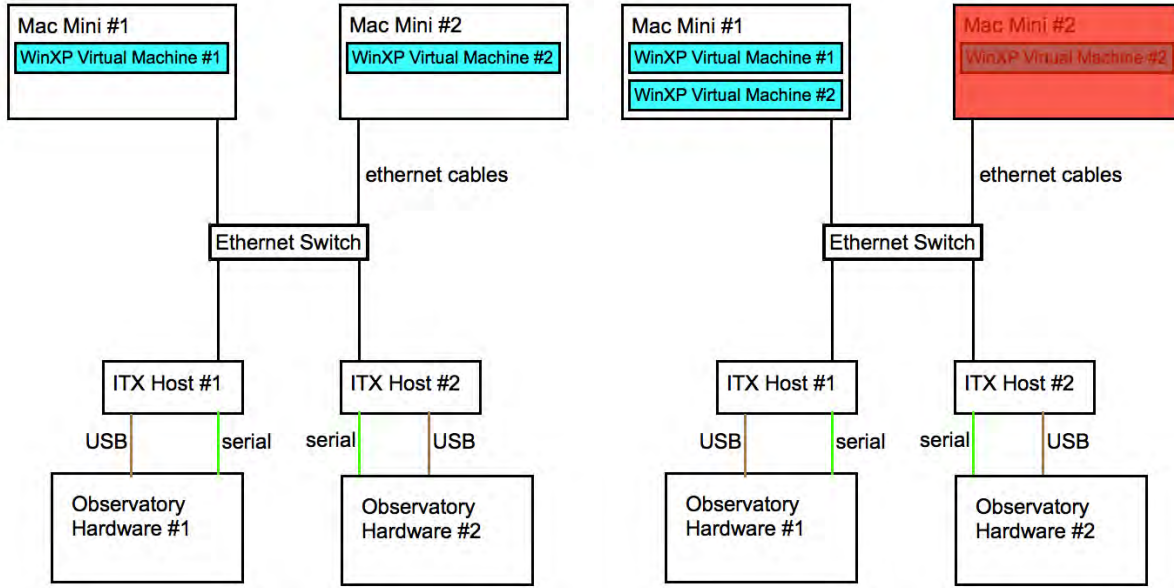


Figure 5. A simplified diagram of the infrastructure described in sections 3 and 4. The left hand diagram shows the relationship between the control-computers (Mac Mini #1 and #2) and the device host computers (ITX Host #1 and #2) during normal operations. The right hand diagram shows the situation after a crash of either Mac Mini #2 or its virtual machine (marked in red). The other control computer (Mac Mini #1) has a copy of the virtual machine to control observatory #2, has booted it, and taken control of the observatory systems.

machine were physical, it would be more difficult to diagnose what exactly had happened and it would require a hard reboot of the machine to recover.

5. SUMMARY

As of this writing the VYSOS telescopes are still in early science mode, so no long term effects on reliability have been measured. Although VYSOS-5 has run for over 1.5 years, the KernelPro port sharing software was installed and used for only about one year of operations, so we can compare operations before and after that change.

The only problem we encountered when using the KernelPro port sharing system was that one of our components (the Paramount ME mount which carries VYSOS-5) was noticeably slower in responding when connected using a serial port and shared over the LAN. This is likely due to network delays and how the driver software reads/writes the port. To mitigate the problem, we simply used the USB connection option for the Paramount ME instead of Serial. When using the USB connection, the port sharing over the LAN appears to have a negligible effect on the response time of the system.

Aside from when using the Paramount connected over serial, we have seen no significant degradation in performance when using this software. Image downloads, mount control, focus control, and dome control all appear to be generally as fast as they were when directly connected to the control computer through a USB extender.

In addition, we have found that connecting hardware in this fashion is more reliable than using USB extenders and USB to Serial adapters which was an early configuration that was used. To connect 3 serial devices and 3 USB devices using extenders required 8 components: two USB extender boxes each with their own power supply, a USB hub, and three USB to serial adapters. In contrast, the same connections require only two components under the new system: the host computer and its power supply. We were forced to replace several components in the extender system over the roughly six months it was in use due to failures, but have had only one failure of the port sharing system in roughly one year of operations.

We have only recently made the change to using virtual machines as our control-computers. So far, no adverse affects have been noted due to that change. We have tested the recovery process by shutting down the primary control machine and booting a copy of the virtual machine on a spare computer and using it to take control of the system.

While not applicable to all projects, the system described in this paper allows small remote or robotic observatory projects, especially those like VYSOS which try to avoid custom hardware and software, to build a fault tolerant system that allows recovery from computer failures without needing a trip to the remote site.

ACKNOWLEDGMENTS

The authors acknowledge support from the National Aeronautics and Space Administration through the NASA Astrobiology Institute under Cooperative Agreement No. NNA08DA77A issued through the Office of Space Science.