

AN EXTENSIBLE FRAMEWORK FOR OBSERVATORY SYSTEM BASED ON DOCKER CLOUD

Jun Han¹, Chenzhou Cui¹, Dongwei Fan¹, Yunfei Xu¹, Changhua Li¹, Shanshan Li¹, and Linying Mi¹

RESUMEN

Los observatorios remotos juegan cada vez un papel más importante en investigación científica, educación en Astronomía y ciencia ciudadana. Con muchos años de desarrollo, los observatorios remotos han experimentado un gran progreso tanto a nivel de hardware como de software. Para una alta tasa de funcionalidad de un observatorio, más y más instalaciones han comenzado a funcionar remotamente para proporcionar servicios en campos diversos. Pero también han surgido algunos desafíos, como por ejemplo, administrar varios telescopios, acomodar las peticiones de los diversos observadores, actualizar el programa controlador, y lograr la coordinación y cooperación entre diferentes telescopios. En cuanto a los problemas, proponemos un marco extensible para el sistema de observación basado en la nube (Docker). No solo ayudaría a resolver los desafíos de los múltiples telescopios a nivel hardware, sino también a implementar software variado de una manera relativamente fácil.

ABSTRACT

Remote observatory is playing more and more important role in scientific research, astronomy education and citizen science. With many years' development, remote observatory whether hardware or software has made great progress. It supports single telescope well and has been very mature. For high utility rate of an observatory, more and more observatories has began to run multiple telescopes to provide more observation services. But it also takes some challenges, for example how to manage telescopes, how to manage lots of observers, how to update driver or application quickly and how to realize coordination and cooperation between different telescopes. After taking into full consideration of the problems, we propose an extensible framework for observatory system based on Docker cloud. It not only could solve that challenges from multi telescopes, but also could make software application to support more hardware platform easily.

Key Words: instrumentation: miscellaneous — methods: miscellaneous — telescopes

1. INTRODUCTION

Astronomy is an observational discipline and observation environment is very important for an observatory. With quick development of city, on the one hand it promotes the progress of remote observatory construction for science research or education, but on the other hand it also brings more light pollution. Observatories are usually built far from cities to obtain good observation environment, but it also let observers to give more time to carry out observation.

Remote observation capability is very necessary for an observatory, and especially the time domain astronomy research is growing fast and has been one of the research focuses in recent years. As the development of telescope technology, the equipments in amateur observatories are not worse than the professional observatories, and there are no real equip-

ment difference between amateur and professional, especially in medium or small observatory. The rise of citizen science also promotes observatory development, and the citizen or amateur is becoming an important research group. Data-driven Astronomy Education and Public Outreach(DAEPO) proposed by China-VO has been one group of International Astronomical Union(IAU)(Cui & Li 2018). Astronomical observatories are entering into big data and citizen science era.

During the last decade, the technology of robotic autonomous observatories has made great progress, and it is more feasible and efficient. It has become a standard tool or way in astronomy observation. As the random time and position for a burst, a network with a number of robotic telescopes is of special importance, and have made great progress, such as BOOTES Network(Castro-Tirado et al. 2012) and MASTER-Net(Liapunov et al. 2012). At present, more and more observatories are tending to build more telescopes for saving more costs and making

¹National Astronomical Observatories, Chinese Academy of Sciences, 20A Datun Road, Chaoyang District, Beijing, China(hanjun@nao.cas.cn).



Fig. 1. Gemini Astronomical Manor was built in year 2017, and the left and right figure are the panoramagram and top view respectively. It is located in Lijing, China, and has more than 50 telescopes now. The telescope aperture is from dozens of centimeters to meter, and several larger telescopes are also in plan. Some auxiliary equipments, like all-sky camera, weather station, nephometer, UPS and so on, are also equipped. It focuses on citizen science, education and astronomical research.

the most of station space. Some public observatories also are playing more and more important role in science discovery and research. Xingming Observatory, located in Xinjiang China, is a typical representative for amateur public observatory. It was built in year 2007, and also the first amateur robotic observatory in China. Based on the released figures, they organize a citizen science project named Popular Supernova Project(PSP)² to search supernova by comparison of different days figures in the same region(Li et al. 2016). Until now, 17 supernova and nova candidates have been reported, and 12 of them were confirmed by optical spectrum. In year 2017, Gemini Astronomical Manor(Figure 1) was built. It is located in Lijing, China and has more 50 telescopes now. The telescope aperture is from dozens of centimeters to meter, and several larger telescopes are also in plan. It focuses on citizen science, astronomical research and education.

Although observatory network has made great progress, it also has more possibilities to be improved, such as observer management, driver or application update quickly, and the coordination and cooperation between different telescopes or even different telescope arrays. In this article, we propose an extensible framework for observatory system based on Docker cloud. It could not only solve that challenges from multi telescopes, but also could make software application to support more hardware platform conveniently. In section 2, we will describe Docker, image registry, Kubernetes and the stability test result. In section 3, we will describe the extensible observatory framework.

²PSP web link <http://psp.china-vo.org>.

2. DOCKER CLOUD

An observatory with multiple telescopes has been common design method for making most of station space. Observatory is a complex system, and the mutual coordination between each other constructs an observation control system. Every telescope could be treated as a computing node to provide service for observers, and all nodes work together to compose a whole system. Here we adopt Docker cloud technology to integrate all telescopes. Observatory normal operations, like telescopes cooperation, driver or application update, system seamless migration and so on, could be realized through cloud computing function module.

2.1. Docker

Docker Container technology³ was launched in 2013 as an open source, and is very popular in Linux world. It not only supports Linux operation system, but also supports Windows, Unix well. In addition, it also supports different architectures, such as x86, amd64, arm and so on. Containers are more portable and efficient relative to Virtual Machines and could launch an application within very short time. Docker container adopts shared kernel to virtualize operating system instead of hardware, so it is more lightweight to deploy application conveniently(Figure 2). Image is the basic unit and packages required dependencies, such as codes, runtime, library, configure, environment and so on. It also could access system bus to communicate with external device in privileged mode. Through this mode,

³Docker Container web link <https://www.docker.com>

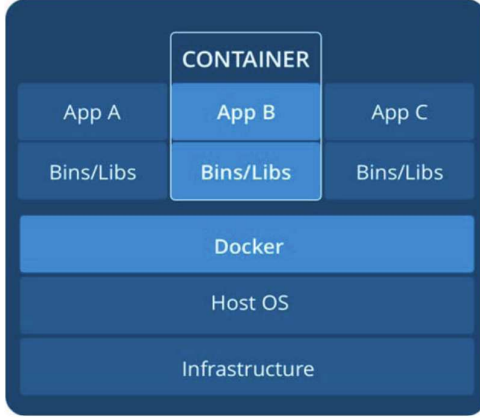


Fig. 2. Docker container technology framework. It not only packages required dependencies, such as codes, runtime, library, configure, environment and so on, but also adopts shared kernel to keep lightweight so as to develop conveniently. Your App and its dependencies, like Bins/Libs, constitutes the container, and it could interact with system hardware in privileged mode.

Docker could be used to package driver or application to communicate with telescope and provide service for observers or other software application. It has become an micro-service application mainstream and supports loose coupling.

2.2. Image Registry

Image Registry is used to store and distribute Docker images. In our design, we adopt Harbor⁴ as the registry. It not only integrates image storage and distribution of the open source Docker Distribution, but also makes some extensions by adding the functionalities usually required by users such as security, identity and management. What's more, it not only could scan images for vulnerabilities and signs images as trusted, but also is easy to increase new nodes to improve performance and supports synchronous backup to keep image data security. When you build an image, you could upload it to the registry and setup necessary permissions. If you have the pull permission, you could use it. It also provides multiple installation methods, like docker-compose, Kubernetes and so on.

2.3. Kubernetes

Kubernetes is launched in July 2015 and builds upon 15 years cloud computing operation experience from Google. As the official website said, it is an open-source system for automating deployment, scaling, and management of containerized ap-

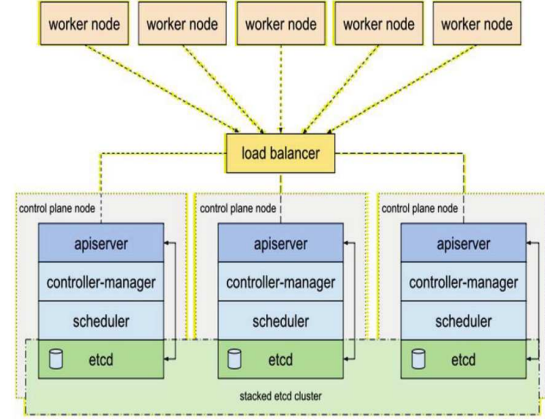


Fig. 3. The Kubernetes' high availability system framework. The worker nodes mainly run proxy and kubelet module to communicate with the master through load balancer method. The master nodes include etcd, scheduler, controller-manager, apiserver core modules to realize resource management, scheduling, monitoring and other functions to the worker nodes. Only one module will be the leader to provide service through election. Another one will be the new leader to provide service once it fails, so as to keep high availability.

plication⁵. Figure 3 shows a high availability system framework. It includes apiserver, controller-manager, scheduler, etcd, kubelet and proxy modules. They work together to be a whole cloud environment. The master and worker node communicate with each other through the load balancer load. As a result, it is easy to add or remove the nodes using this framework without affecting other node. The apiserver module is a component of its control plane for exposing cloud platform API. The controller-manager module is a separate process to notice and response node status, maintain correct pod number and manage service account and endpoint object. The scheduler module runs on the master node and is used to watch newly created pods with taking into account for the scheduling decisions. The etcd module is a kind of key value database for storing all cluster data, including status, resources and so on. The kubelet module is the node agent that runs on each node to ensure the containers are running and healthy. The proxy module is used to forward TCP/UDP stream to keep communication among nodes and maintain network rules on nodes. Except above, container runtime is also indispensable. It is responsible for running containers and supports multi container runtimes, like Docker, containerd, cri-o, rktlet. Through this framework, ev-

⁴Harbor web link <https://goharbor.io>

⁵Kubernetes web link <https://kubernetes.io>

TABLE 1
SINGLE MASTER CONFIGURATION
IN LOW LOAD CONDITION

configuration	Worker number
4C 8G	≤ 5
4C 16G	≤ 20
8C 32G	≤ 100
16C 64G	≤ 200
Multi-master	More

ery worker node could run some containers, like telescope driver, application et al., and the master could manage them through its function modules.

Although the high availability system framework provides high stability for cluster and could support several thousand work nodes, it is not very necessary for multiple telescopes observatory as there are no so many telescopes in general. The table 1 shows the single master CPU core number and memory size in low load condition, and it also could keep high stability. When the number of work node exceeds 200, one single master is no longer reliable, and multi-master should be considered and even replace mechanical hard disk with solid hard disk to enhance data storage and query efficiency. In most time, the control computer of telescope used as a worker node usually keep in low load, and a minicomputer or an ARM development board, like Raspberry Pi, is enough.

3. THE EXTENSIBLE OBSERVATORY FRAMEWORK

An observatory not only includes telescopes to perform a set of observation, but also includes some auxiliary equipments to monitor itself so as to adjust itself to some extent. The network has brought a revolution in industry application ever since it was developed. The internet takes a very important part in industry communication and control, and has become an inevitable development trend. In system design, both hardware and software should be considered, and network communication is adopted for these application. In the following, we will describe the design details.

3.1. The Hardware Integration System

The power management is one basic insurance for stable operation of observatory. We propose a hardware integration system module as a bridge between equipment and control software. It supports common observation equipments, adopts universal protocol as the interface, and could be extended.

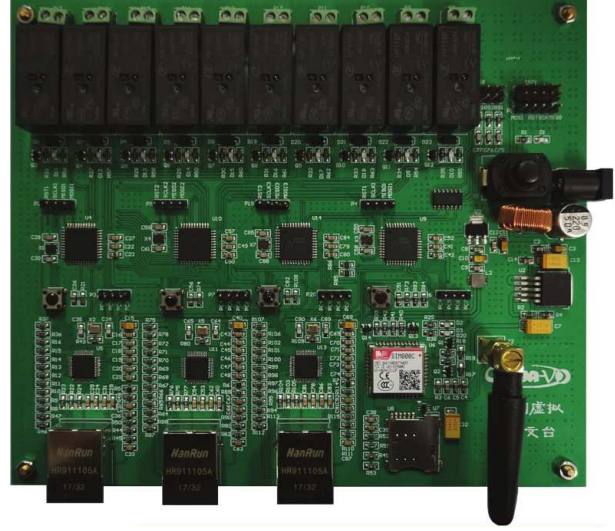


Fig. 4. The finished product schematic diagram of the Hardware Integration System. It is used as a bridge between equipment and control software to manage the observatory equipments power. It supports three kinds of approaches (Local, Network, Phone) for users to access this system to control the relays at the top of the figure. There are three RJ45 interfaces for network control at the bottom of the figure, and one antenna used to receive control commands from phone. At top right corner, some connectors are provided used to connect buttons for local control.

TABLE 2
RELEVANT PARAMETERS OF
THE PRINTED CIRCUIT BOARD

Parameter	Value
Input Voltage	5VDC
Rated Current	0.45A
Maximum Current	3.1A
Operating Temperature	$-30^{\circ}\text{C} - 85^{\circ}\text{C}$
Relative Humidity	5% – 85%

Three kinds of approaches (Local, Network, Phone) are provided for users to access the hardware integration system to control the relays, and ATmega 16, W5500, SIM800C are selected as the main control chips to process control commands. The finished PCB(Printed Circuit Board) product is showed in Figure 4, and the relevant parameters are listed in table 2. It has good stability and self-adaptability with long time test at low and high temperature. See (Han et al. 2018) for more details.

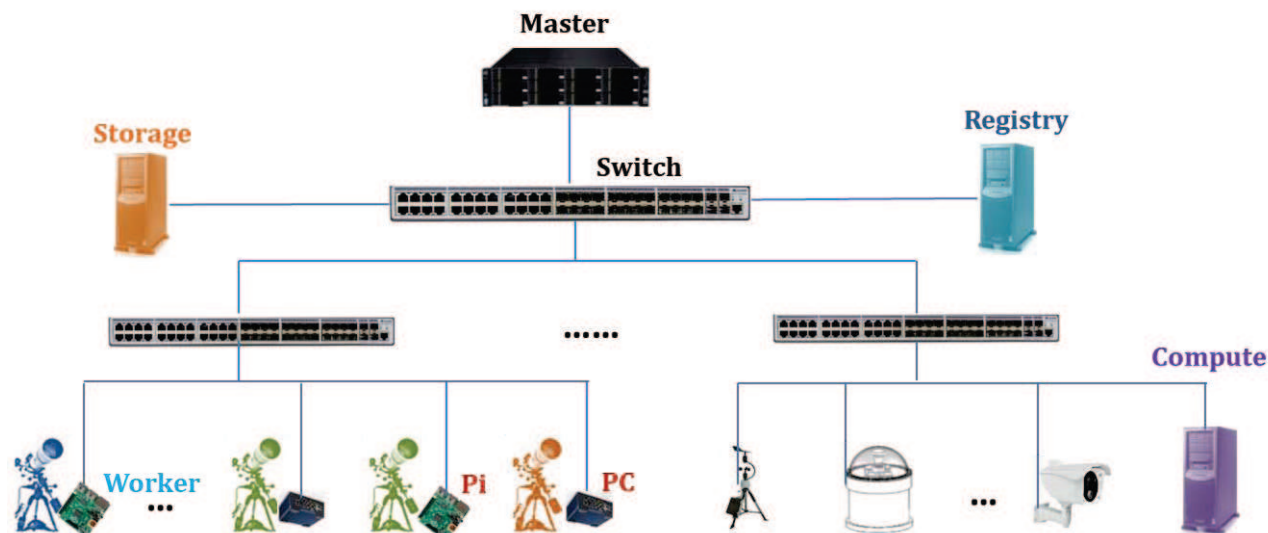


Fig. 5. The extensible observatory system framework. It consists of seven core units, master computer, registry, storage, switch, telescope, compute unit and auxiliary equipment. All the resources are integrated together as a whole system through Kubernetes. It inherits the characteristics of cloud computing, turns the problem into a computational problem and could realize application seamless migration without affecting others. This figure is a simple framework for conveniently describing main function, and could be extended or modified according to the observatory plan.

3.2. The Software System Framework

Hardware provides the basis for efficiency, but only software can fully exploit the benefits of that infrastructure. An observatory has several telescopes, management and maintenance are the problems that observatory needs to face. For this reason, we propose an extensible software framework based on Docker cloud to manage and maintain telescope observation control system. Every telescope is treated as worker node. In this condition, it inherits all characteristics of cloud, like authority management, micro service, high availability service and so on. Figure 5 shows the extensible framework structure of observatory software system. It is a simple framework for conveniently describing main function, and could be extended or modified according to the observatory plan.

In this system framework, it consists of seven core units, master computer, registry, storage, switch, telescope, compute unit and auxiliary equipment. All the applications are packaged through Docker and deployed through Kubernetes's micro service. They communicate with each other through TCP/IP protocol, and some telescope control softwares, like INDI library⁶, have supported network well. INDI could run in Linux, Windows and Unix operating system, and provide Docker image support. We have tested it in the machine with ARM or x86 architec-

ture through micro service of Kubernetes in Linux environment, and it works well.

Master computer acts as a observatory operation controller to inform specified worker nodes or telescope units to start, close or update their containers according to observation strategy. It could manage all the computer resource in this framework, including storage and compute unit, as they also run as worker nodes. The switch is not limited to this structure and could extend according to the requirement. Its main function is to keep all the resource in the same network and to be managed conveniently. Registry is used to store all the Docker images and provides image service for all the worker nodes. When the worker nodes need to start new containers, they could pull specified images from registry. The control computers of telescopes also run as worker nodes. They could pull, run, stop and update specified application containers to carry out observation according to instructions from master. The control computers could be any hardware architecture, which supports Docker. The Docker agent could pull the right image according to its manifest information. The auxiliary equipments include weather station, all-sky camera, video camera and so on. They are used to monitor observation environment, and the monitoring data could be processed and analyzed by compute unit to generate operation strategy, and then inform the master to adjust the worker nodes.

⁶INDI web link <https://www.indilib.org>

The compute units also could be extended to any network node for other purpose, such as data process, receiving astronomical event alert, user login control. Storage is used to save observation data from the telescopes.

For adding or removing a telescope, it is also very convenient, and what you should do is registering or un-registering it to the master. It also could replace an old equipment or update softwares seamlessly without affecting application through cloud migration strategy. The telescopes also could be grouped as different arrays to execute different operations. In addition, Kubernetes integrates many role control strategies to manage application or users, and supports many monitoring plug-ins to show cloud resource status. All in all, it inherits the characteristics of cloud computing, and turns the problem into a computational problem.

4. CONCLUSION

In this article, we described the extensible system framework for observatory. It is a combination between cloud computing and telescope management. It could solve many practical management problems, such as multi telescopes management and maintenance, application seamless update and migration, resource monitor, user management and so on. We have tested INDI as the control software to carry out observation test through Kubernetes, and it works well under this framework. However, there are still pending problems, how to process pod failure to keep application in high availability and how to design a

reasonable communication encoding between different application. In the next step, we will focus on them.

Acknowledgements: This work is supported by National Natural Science Foundation of China(NSFC)(11573019, 11803055), the Joint Research Fund in Astronomy(U1531246, U1731125, U1731243, U1931132) under cooperative agreement between the NSFC and Chinese Academy of Sciences(CAS), the 13th Five-year Informatization Plan of Chinese Academy of Sciences (No.XXH13503-03-107). Data resources are supported by China National Astronomical Data Center(NADC) and Chinese Virtual Observatory(China-VO). This work is supported by Astronomical Big Data Joint Research Center, co-founded by National Astronomical Observatories, Chinese Academy of Sciences and Alibaba Cloud. Thanks Zhou Fang for providing the Gemini Astronomical Manor's panoramagram and top view.

REFERENCES

- Castro-Tirado, A. J., Jelínek, M., Gorosabel, J., et al. 2012, *ASInC*, 7, 313
- Cui, C. & Li, S. 2018, arXiv e-prints, arXiv:1801.05098
- Han, J., Fan, D., Cui, C., et al. 2018, arXiv e-prints, arXiv:1801.03667
- Han, J., Wang, C., Fan, D., et al. 2018, *A&C*, 25, 89
- Li, S., Fan, D., Gao, X., & Cui, C. 2016, 10th International Technology, Education and Development Conference, INTED2016 Proceedings, Valencia, Spain, 7561-7569
- Liapunov, V. M., Kornilov, V., Gorbovskoy, E., et al. 2012, *ASInC*, 7, 275