NUCLOTRON AND NICA CONTROL SYSTEM DEVELOPMENT STATUS

Evgeny V. Gorbachev, Vasiliy Andreev, Alexander Kirichenko, Dmitrii Vladimirovich Monakhov, Sergey Romanov, Tatyana Vladimirovna Rukoyatkina, Georgy Sergeevich Sedykh, Valery Volkov, JINR, Dubna, Russia

Abstract

The Nuclotron is a 6 GeV/n superconducting proton synchrotron operating at JINR, Dubna since 1993. It will be the core of the future accelerating complex NICA which is under construction now. NICA will provide collider experiments with heavy ions at nucleon-nucleon centre-of-mass energies of 4-11 GeV. The TANGO based control system of the accelerating complex is under development now. This paper describes its structure, main features and present status.

INTRODUCTION

NICA complex will consists of heavy-ion and polarized particles sources, RFQ injector, heavy- and light-ion linear accelerators, superconducting booster synchrotron, Nuclotron and two superconducting collider rings [1].

The control system of the NICA complex aims at accomplishing few main tasks:

- Management of large amount of equipment which is distributed on the accelerator complex area.
- Realization of different regimes of the accelerator complex working cycle – colliding or fixed target experiments, various ion types and energy.
- Strict synchronization of accelerators in the chain.
- Comprehensive beam diagnostics during the entire cycle.
- Providing protection and safety measures.

We can specify few important features that the NICA control system has to provide:

- Centralized administration and monitoring of the control system components.
- Reliable operation, quick recovery after possible failures.
- Access to equipment has to be restricted for certain personnel with rights limitation according to user roles.
- Ease of support, modification and scaling during long accelerator complex life-time which will operate until 2045.
- Rapid development and easy deployment of the control system, taking limited time and man-power into account.
- Possibility to integrate third-party control systems as some components of the accelerator complex will be designed and constructed by external organizations.

CONTROL SYSTEM LAYOUT

The control system is distributed network of computers which communicate by means of transport protocol over TCP/IP. The NICA control system uses the TANGO controls [2] as the middleware providing such communication protocol.

TANGO is the modern distributed control system framework based on CORBA. The fundamental unit of TANGO is a device, which is an abstraction hiding real equipment or program component behind the standard interface. TANGO provides high level client application interface which has necessary programming classes to implement client-server communications - synchronously or asynchronously execute commands, read or write attributes, or use events to acquire the data from the TANGO devices. TANGO incorporates a number of tools to build efficient control system environment including centralized administration and monitoring, access control, logging system, data archiving and code generation for rapid development of the TANGO device servers using C++, Java and Python.

Three layers of the NICA control system components can be distinguished (Figure 1):

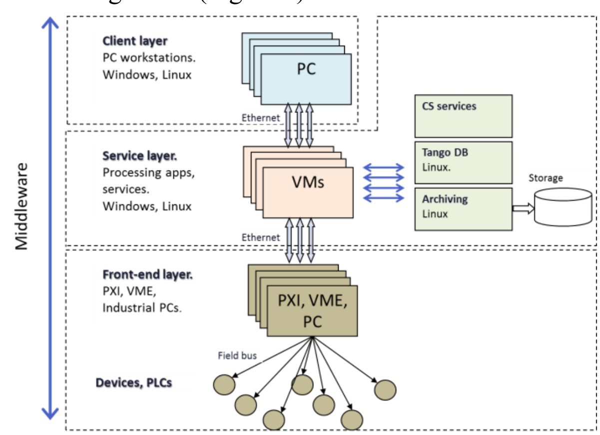

Figure 1: NICA control system layout.

1) Front-end layer consists of industrial computers, intellectual controllers, crates that directly control equipment and acquire data from sensors. Front-end computers run low-level TANGO programs that realize data acquisition, equipment handling and hide protocol and connection details from higher layer components.
2) Service layer consists of high level TANGO devices representing entire subsystems. They collect data from front-end TANGO devices, process it and realize algorithms to control some large subsystems. Those high-level programs provide standard TANGO interface for entire subsystems allowing client software to execute commands, read and write attributes without knowledge of subsystems structure. Besides, the service layer provides a set of services that are necessary for efficient control

ISBN 978-3-95450-148-9

system functioning – administration, control system hardware and software management, monitoring, data archiving, development services.

3) Client layer represents the accelerator complex state to operator, visualizes acquired data and allows operator to perform control tasks. We would like to provide global interface to allow operator to access the entire accelerator complex control with possibility to navigate to its components.

Front-end Layer

Primary requirements for the front-end layer hardware are easy and rapid development, good reliability and performance, easy maintenance during the accelerator complex operation. The development and support of custom hardware for the large accelerator complex is a very resource-consuming task. It is decided to use commercial hardware for most of the control systems tasks. The base of the NICA controls front-end layer is National Instruments PXI, which is modular eurocard packaging platform with PCI and PCI express busses with additional synchronization and trigger lines.

PXI platform offers a wide range of modular instruments of many types including digital and analog I/O, industrial interfaces, digitizers and scopes, signal generators and many others. National Instruments provides Windows and Linux drivers for most of the hardware and supports Labview as well as text based languages.

While most of the control system tasks can be implemented by modular instruments, some unique signal generation, processing and synchronization tasks can be realized using programmable logic devices. National Instruments provides few FPGA solutions to implement custom hardware. One of them is FlexRIO product family, which consists of PXI and PXI express modules with Xilinx FPGA, large amount of on-board memory and standard or custom I/O modules that are attached to the front panel of FlexRIO and have access to pins of FPGA. Developer can implement custom FPGA firmware as well as custom input and output signal processing, conditioning and conversion.

Another solution is CompactRIO controller, which is FPGA and processor based platform with standard or custom I/O modules. It is running real-time OS and controlled over Ethernet.

The TANGO device servers for a range of National Instruments hardware performing common acquisition tasks were developed. They include drivers for digitizers and scopes, analog and digital input and output, timers and counters, digital multimeters, temperature sensors acquisition modules.

Software for FPGA based boards consists of two parts: FPGA firmware and FPGA interface program performing data exchange between controller and FPGA. LabVIEW provides rapid firmware development using graphical language and VHDL. Interface programs can be developed as the TANGO devices using FPGA Interface C API. Those TANGO devices are running on PXI or CompactRIO controllers providing an easy way to integrate FPGA based solutions into the TANGO environment.

Developed drivers allow quick deployment of almost any National Instruments acquisition board. Each driver can perform a single input or output task so it is necessary to run few TANGO devices to realize all features of an acquisition board. The acquisition properties such as sample frequency, samples number, trigger source, and others are configured as the TANGO devices properties.

Client Layer

TANGO provides the set of graphic user interface toolkits for rapid development of graphical client applications in all supported programming languages – C++, Java and Python. There are also LabVIEW TANGO bindings that allow developing full featured TANGO client applications using rich Labview visualization features. Most of the existing desktop client applications for the NICA control system are developed in LabVIEW with the TANGO bindings.

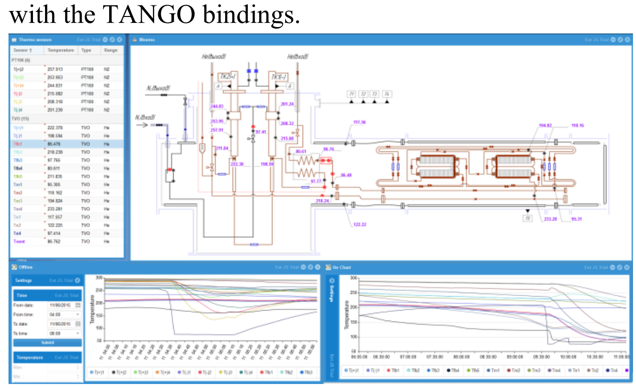

Figure 2: Thermometry web application for the NICA superconducting magnets test bench.

Few auxiliary TANGO devices were developed to simplify creation of web client applications by providing access to TANGO from web browser:

- The TANGO device server RestDS realizes REST protocol to access TANGO devices attributes and commands. It contains embedded http server providing access to TANGO devices attributes and commands at URL combined from host, port, device and attribute names. The reply is JSON string consisting of attribute name, value, timestamp and quality. POST method can be used to execute command with input arguments.
- The TANGO device server WebSocketDS also contains embedded http server and is intended for accessing attributes via WebSocket protocol. The attributes names and the device name are specified in properties, and the data update rate is configured via polling period.
- The TANGO device server TangoWebAuth performs authentication of users to restrict access of web applications to TANGO devices. The access rights are configured in a special database.

ISBN 978-3-95450-148-9

The developed auxiliary TANGO servers allow creation of web application in JavaScript without any intermediate application servers. Thermometry web application for the NICA superconducting magnets test bench is shown in Figure 2.

Service Layer

High level applications of the service layer do not interface with any hardware, so they can run in any place of the control system and can be virtualized. Virtualization is an effective way to deploy control system applications providing easier management of virtual machines (VM), better tasks isolation, fine tuning of CPU time or disk space allocation and more efficient usage of hosts resources. Virtualization also makes possible to obtain high availability features – any virtual machine can be quickly restored or migrated from failed physical server to run on a different host.

Proxmox VE is used as a virtualization solution. It is a free server platform for creation and management of virtual infrastructure including virtual machines, local and shared storage and high-availability clusters. It supports both Kernel based Virtual Machines (KVM) and Linux container-based virtualization which provides isolated Linux containers with no performance impact.

Shared storage is the vital part of virtualization to provide high availability of virtual machines. There are few requirements for the shared storage:

- Good performance as we need to run many images on it.
- Redundancy to allow using VM images even if some disks or servers failed.
- Scalability – storage must be expandable without losing performance to provide the data storage for future control system tasks and data acquisition.

The NICA control system uses distributed object storage cluster Ceph [3] as shared storage solution. Ceph storage cluster consists of nodes which communicate with each other via network and distribute and replicate data dynamically. Ceph's RADOS block device (RBD) is an ideal way to store VM images that are stripped and replicated across the entire storage cluster. Ceph RBD is thin provisioned so unused image space can be reclaimed.

It is possible to run Ceph services directly on Proxmox VE nodes, so the same hosts are used to provide both distributed storage and virtualization cluster. The cluster presently consists of four nodes each with dual eight core E5-2600 Intel Xeon processors, 32 GB of RAM, five reliable 15k RPM SAS2 disks and enterprise class solid-state drive (SSD) for operating system and Ceph journal devices. The network equipment includes two 1 Gbit network switches to separate public and Ceph cluster network. One of them will be upgraded to 10 Gbit network switch to provide network capacity suitable for Ceph cluster network.

The high availability of control system is achieved by hardware solutions which include using of uninterruptable power supplies (UPS), redundant power supplies for servers, enterprise class long-lifetime drives and data redundancy for virtual machines and databases.

All virtual machines run on Proxmox VE cluster with images stored on CEPH storage with replication factor of three. Hard drives failures are handled by Ceph and not affecting the operation of the control system at all. All virtual machines are periodically backed up to external Network File System (NFS) storage located on redundant array if disks (RAID). The uninterruptable power supplies are constantly monitored by using Network UPS Tools (NUT) [4] to ensure proper VMs shutdown on low charge.

The TANGO database is essential for the NICA control system operation. This is a MySQL database that runs in Linux container on fast local SSD for performance reasons. The high-availability of the TANGO database is achieved by means of MySQL replication. Semi-synchronous MySQL replication with two master MySQL servers with fully symmetrical configuration is deployed. Each master MySQL server is a slave of the other MySQL server but only one of them is getting external connections to avoid possible conflicts. A proxy solution for the TCP–based applications HAProxy [5] is used to periodically check health of MySQL servers and route external MySQL traffic to primary or backup MySQL backend server. HAProxy runs on high-available VM to ensure its permanent availability. Periodic dump of the TANGO database from MySQL slave server is used as additional measure to provide hourly TANGO database backups for two weeks of operation.

The storage space scalability is provided by Ceph which retains or improves its characteristics when adding more disks and more nodes.

Storage space and throughput, CPU cores number and amount of RAM can be increased by adding more nodes to the control system cluster. As a result, more data and VM images can be stored and more virtual machines can be run. The only drawback is a short burst of network traffic during data rebalancing to new storage daemons.

CONTROL SYSTEM ADMINISTRATION

The TANGO device servers are running on many distributed computers, hence the ability to control all the control system components remotely is necessary. TANGO provides a special service Starter which is running on all control system computers. Its main task is to start all necessary TANGO device servers in proper order at system startup and keep them running.

The TANGO manager Astor is used to visualize the state of entire control system and provide tools to start or stop the TANGO device servers remotely, access their logs and settings in the TANGO database. Operator can see if computers and TANGO device servers are running or not. However, a computer can have performance problems or the TANGO device can be in fault state which cannot be seen using Astor.

An additional TANGO monitoring system which consists of two parts was developed. First part is a

ISBN 978-3-95450-148-9

TANGO device to monitor computer resources. It runs on each computer of the control system and periodically collects information about the disk, processor and memory usage. Another part is a TANGO device intended for monitoring states of every TANGO device of certain subsystem. The monitoring client application (Figure 3) represents states and statuses of the all TANGO devices and computers of the control system. Alerts and notifications are sent to developers in case of problems.

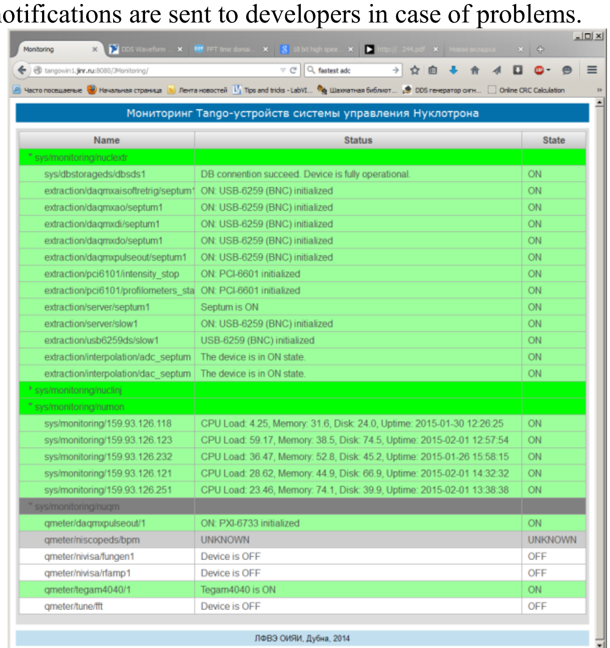

Figure 3: TANGO monitoring.

Apart from the Tango monitoring, monitoring solution Zabbix [6] is used to monitor network traffic, hypervisors resources, Proxmox VE cluster, UPS devices, TANGO database performance, Ceph cluster performance and health, local and shared storage space.

Another important aspect of control system operation is access control. We want to restrict access to control system software for certain computers and users and give them corresponding rights to perform specific tasks. Network access limitation, including private sub networks for subsystems and proper firewalls configuration is one of the possible measures. Another possibility is Tango AccessControl service that restricts connections to TANGO devices to certain users connecting from certain computers. The access checks are performed on the client side before opening connection to the TANGO device.

Additional server-side authorization service was developed to provide more secure access control and more flexible rights restrictions. It uses additional authorization TANGO device which accepts username/password pair from client application and authenticate the user using Linux Pluggable Authentication Modules (PAM) and information from MySQL database. If authentication is passed, a temporary session is opened for the specific user, IP address and process ID. TANGO device which needs to check the user's rights makes request to authorization server from command execution method with TANGO device name, command name, client IP and process ID. After that, authorization server checks the IP and process id pair against opened sessions and checks user rights from the database.

It is possible to precisely tune the access rights using MySQL regular expressions. Operator-expert rights separation can also be implemented.

CONTROL SYSTEM MANAGEMENT

The NICA complex consists of few large facilities, such as accelerators and transfer lines. Each of them has a number of subsystems. Every subsystem consists of large number of equipment, TANGO drivers, cables, computers, network equipment which are developed and maintained by certain people.

The NicaControls [7] database was developed to store this information and describe relations between its components. Using these relations one can find documentation for hardware or software, links to source code and executable files, developer name, rack location, computer IP address and other useful information.

Information in the database is added during the control system development. The NicaControls database is also used by other control system components. The TANGO monitoring system periodically queries NicaControls database to include or exclude the TANGO devices from monitoring. The server-side access control also gets access rights information from it.

CONCLUSIONS

Design of the NICA control system infrastructure was presented. It provides rapid development of hardware and software using the TANGO Controls and National Instruments equipment, reliable and scalable operation by using virtualization and Ceph storage cluster, comprehensive administration and monitoring, efficient management of the control system equipment and software by using the NicaControls database.

Several Nuclotron subsystems and prototypes for the future NICA accelerators were developed using the described approach.

REFERENCES

[1] G.Trubnikov, N.Agapov, O.Brovko at al., NICA project at JINR, Proceedings of IPAC2013, Shanghai, China, May, 2013

[2] TANGO Controls: http://www.tango-controls.org

[3] Ceph: http://www.ceph.com

[4] Network UPS Tools: http://www.networkupstools.org

[5] HAProxy: http://www.haproxy.org

[6] Zabbix: http://www.zabbix.com

[7] Gorbachev E.V., Sedykh G.S. The equipment database for the control system of the NICA accelerator complex, Proceedings of ICALEPCS2013. – San Francisco, CA, USA, 2013, Pp. 1111-1113

ISBN 978-3-95450-148-9