DIAGNOSTIC AND REMOTE MAINTENANCE OF DISTRIBUTED CONTROL SYSTEMS USING FREEWARE SOLUTIONS

R. Bartolome, F. Havart, T. Riesco, CERN, Geneva, Switzerland

Abstract

The Monitoring and Access group (ST/MA) at CERN is facing in the next few years the installation and implementation of a number of new systems: the Safety Alarm Monitor, the Radiation Monitor, the Technical Infrastructure Monitor and the Safety Information Panels. All of these have a common question to resolve, how the system will monitor its own performance? This problem is in the process of being resolved using freeware solutions and a plug-and-play installation kit. This paper shows the work that has been done in the terms of problem definitions, compilation of user requirements, tests of different implementations and the results of the first installation.

INTRODUCTION

The ST/MA group has analyzed its needs in terms of system monitoring for the current installations and for future projects to be installed in the Technical Control Room (TCR). As a result, it launched the System Performance Information and Management (SPI-ME) project with the objective of providing a common monitoring platform.

THE PLATFORMS

The computer infrastructure in the TCR is based on four different types of platform:

- HPUX platforms: They are the nucleus of the TCR data diffusion system and they are composed by 12 servers.
- Windows based platforms: based on NT and 2000. At the end of 2003 the TCR will have 31 installations, dedicated to the control of ventilation equipment and electrical distribution.
- Linux based platforms: Their number is expected to increase when the HPUX platforms disappear. The TCR has 2 installations.
- PLC platforms: the base for today's data acquisition system, their number will certainly grow. The TCR uses 35 equipments at present.

The percentage of the disk space used, the correct execution of key processes and the status of the network connection are the parameters to be monitor by the HPUX, Windows and Linux platforms. For the PLC platform, it is the status of the PLC (operational or not) that will be monitored.

THE PROJECTS

After an analysis of the monitoring needs for the projects being developed at present in the ST/MA group at CERN, we have concluded that they have similar requirements. Their main common need is that they have to monitor platforms 24 hours a day, 365 days per year, as they are all related to the Technical Control Room and/or the Safety Control Room.

CSAM – CERN Safety Alarm Monitoring [5]

The future acquisition and diffusion system for safety alarms and safety data needs to monitor 44 Windows-based servers and a double number of PLCs, installed at all of the CERN sites. The monitoring tools must permit to know the status of the hosts and to remotely start corrective actions when/where needed.

RAMSES – Radiation Monitoring [6]

The project will start its design phase soon. The modularity and portability of the solution proposed by the SPI-ME project should help in finding a common solution.

TIM – Technical Infrastructure Monitoring [7]

This project is actually in the design phase. The technical infrastructure monitoring system contains approximately 35 PLCs and 12 servers. Its requirements are very close to CSAM; this will help to use a common solution.

SIP – Safety Information Panels

The future LHC safety information service must also be monitored, as it will allow personnel entering the LHC tunnel to be informed about safety conditions. The monitoring needs are also very close to those from CSAM.

THE USER REQUIREMENTS

The project team has compiled a list of user requirements [1] among the TCR operators and the team members of the projects described above. As an example, here are some of the requirements and their implication in the solution:

- *"The tool shall monitor the operational status of PLCs":* The market tools do not have this possibility, so it was decided to develop this part at CERN. In this case the tool has been chosen to be an open source.
- *"The tool shall allow the users to browse remote diagnostics files":* To be compliant with this requirement, it was decided to choose a tool with remote control capabilities.
- *"If possible, only one common monitoring solution should be developed for HPUX, Windows, Linux and PLC platforms":* The tool should be multiplatform.

- *"The configuration should be kept in a central database":* It is mandatory to configure the tool from a central repository, in order to easily update it. As they should be installed on different platforms, it was decided to choose a tool with the capability to be configured using flat ASCII files.

THE MONITORING TOOL

The monitoring tool that was finally chosen is Nagios [2], freeware software running on a Linux platform. This solution has already been used by the ST/MA group at CERN. In addition to the good results showed by the tool, it presents some other advantages:

- From the developer point of view, Nagios is an open source product, which means that the developer is able to add, delete or modify the product behaviour. This contributes to better adjusting the tool to the user requirements. In our case, this was mandatory, as some of the requirements need to be developed.
- From the system operator point of view, the use of Nagios permits to implement its comments very easily. The behaviour of the tool is better adjusted and so too is the operation.
- From the project management point of view, the product is licensed under the GNU General Public License, so there is no price-per-license to be added to the project cost.

Nagios has a web interface to view its data and diagrams on the web, so a web server needs to be installed. The project team chose the Apache web server [3], freeware developed by The Apache Software Foundation. This software is both easy to install and use, and it has shown its robustness in previous installations in the TCR. It runs on a Linux platform.

THE REMOTE CONTROL TOOL

For the remote control tool, the project team has chosen Virtual Network Computing (VNC) [4]. VNC is also freeware, developed at the AT&T laboratories.

It consists of two types of components: a display server, which generates the view, and a viewer, which draws the view on the remote screen. There are two features of VNC that make it easy to implement and to use in a heterogeneous environment:

- The server and the viewer may be on different machines and on different architectures
- No state is stored at the viewer. Breaking the viewer's connection to the server and then reconnecting will not result in any loss of data.

The architecture of this product is the simplest that can be defined. The viewer is displaying what the server is storing in its remote frame buffer. The secure connection between the server and the viewer is implemented using the Secure Shell over the TCP/IP protocol. It runs on a Linux platforms.

THE PILOT INSTALLATION

In order to test the behaviour of the complete system, the project team decided to set up a pilot installation in the TCR. The objectives of this pilot installation were:

- To test the easiness of installation.
- To test the automatic configuration of the tools from a central database.
- To test the robustness and maintainability of the global solution.
- To test the alarms activation and their management, and the interfaces with the TCR alarm system.
- To test the tools' add-in capabilities.

The architecture of the pilot installations is as follows:

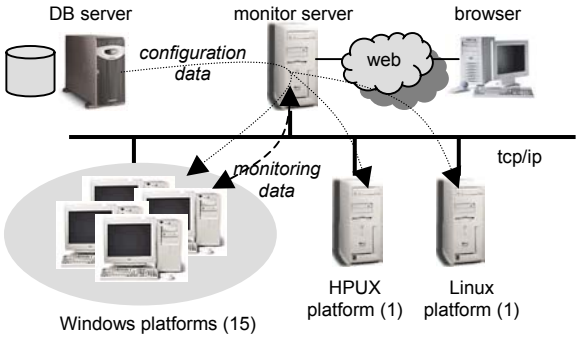

Figure 1. Architecture of the pilot installation.

The three tools Nagios, Apache and VNC were installed in a Linux server running the CERN certified Linux version 7.2.1.

The servers to be monitored were 15 Windows NT computers running commercial Wizcon® SCADA applications, one HPUX server and one Linux server.

The configuration of the complete installation was stored in an Oracle database.

The access to the monitor data was done through the web via a standard web browser.

CONCLUSIONS

The experience acquired with the pilot installation shows that the solution is robust and its performance is adequate for the project. The pilot installation has been running since August/2003 without failures, and no major server maintenance has been necessary to date.

The installation of the three tools was straight forward and well documented.

The automatic configuration of the tools needs some development in the Oracle DB, the creation of tables and SQL scripts to correctly format the data. This shall be improved in the final installation. Future work includes a solution based on a web interface.

The possibility to easily add new functionality to the tool is very useful. In our case, from the first definition of the URD new requirements have been declared by the users, mainly concerning the development of external interfaces for the central server. This has been treated

without major issues due to the open source facilities provided by the tools.

REFERENCES

[1] Bartolome R. "System Performance Information and Measurement project – User requirements document," CERN ST/MA (2003-001)

[2] http://www.nagios.org
[3] http://www.apache.org
[4] http://www.realvnc.com
[5] http://st-proj-csam.web.cern.ch/st-proj-csam
[6] http://proj-ramses.web.cern.ch/proj-ramses
[7] http://stmoin.home.cern.ch/stmoin/projects/TIM/tim.html