

Trigger-based Intelligent Controller for Distributed SCADA Systems

Majdi Khalel, Ahmad Aljaafreh *

Tafila Technical University, Tafila, Jordan, 66110

Abstract

In this paper, we take advantage of the power of database-triggers in industrial field to build a trigger-based Intelligent Controller for Distributed SCADA Systems to provide fast response and intelligent system. In addition to utilizing HTTP protocol as a communications infrastructure to exchange data between distributed stations and main servers. The system provides monitoring solution compatible with Personal computers, Phones, PDA's. This paper discusses a new Web Distributed SCADA approach that matches contemporary needs and requirements using open and proven database triggers technology. It completely eliminates many of the existing concerns and costs associated with IP technology. This paper proposes a new solution for world-wide remote industrial control and monitoring. This solution is cost-less, efficient, and highly fast. This new solution provides a smart controller using a neuro-fuzzy algorithm. This solution provides interoperability with industrial related applications such as supervisory control and data acquisition (SCADA) systems. This system uses the power of the database triggers and the flexibility of HTTP protocol as a communications infrastructure. It also uses normal web-hosting service that supports MySQL database and server-based scripting language such as PHP. A database trigger is generated by an interpreter based on conditions set by an administrator. Multiple mirror servers are used as redundant servers and one core server.

Keywords: *Distributed SCADA, Web SCADA, Intelligent Web Controller.*

1. Introduction

There are broad collections of Distrusted SCADA solutions provided by industrial companies for remotely controlled equipments and centralized monitoring systems. Most of these technologies are based on IP reservation, where each station has to reserve a public IP and communicate based on TCP or UDP protocol. In addition, few of them provide centralized intelligent controlling service and this due to the complexity of such systems and the overhead of installing and design of such systems. Many of these systems need complex equipments and highly trained stuff to install. Thus, system troubleshooting and maintenance should be done by a specialist. It is also expensive to link many station with each other for distributed application. Many customers refuse this service at all; even it is one of the key success services in any industrial facility. Such customers always think about the security risks. Because the whole system is connected through Internet, even a VPN is used; it is still possible to hack such systems.

As we all know Internet now takes its place in almost everything technical in our life, so using Internet will give us a lot of advantages. However, we have to be ware about a lot of risks that will shows up in the horizon.

WEB SCADA is the most popular solution that is used by many industrial facilities [4]. WEB SCADA does not need very complex equipments and highly trained stuff to be installed. It is also easy to be troubleshot and maintained. However, it needs complex electronic equipments to provide intelligent web controller.

In SCADA systems, interoperability is a crucial consideration to think about. To provide interoperability a standard common protocol for communications is needed like the X10 protocol. This protocol was originally used for domotics in the USA, but such protocols cannot be applied in industrial world-wide remote controlling purpose, due to technical limitations of this protocol, which use power line to transmit data. In this paper, we propose a solution based on three main open-source technologies: MySQL database system, PHP programming language, and HTTP protocol as the communications infrastructure as in Ref. [1]. Using database-triggers concept gives the ability to build real-time system. Database trigger is the best solution since web-based applications is described as action-based systems. For example, when you browse a website you will not have a new content until you press a link; send a request, the new content produced by a server-side scripting language, Like PHP or a database trigger.

* Corresponding author. Tel.: +962777411126

Fax: +9876543210; E-mail: a.aljaafreh@ttu.edu.jo

2. Previously Proposed Solutions

In this section most of the distributed SCADA solutions offered and used by most of the industrial facilities are presented in this section. Although the solutions discussed in this section have already been presented in the literature, we discuss them here for purposes of completeness. Readers familiar with these solutions may wish to skip to Section 3.

2.1. Distributed SCADA Linked By IP

In this solution a dedicated server is used in which each station in the distributed system has a public IP address. Communications infrastructure selection such as, GPRS, ADSL, VSAT, depends on numerous factors like environmental limitations and governmental restrictions.

In this solution a lot of network equipments are used. Thus, probability of failure is considerable. This solution uses a lot of industrial special equipments like special interfacing devices, communications tools to handle the public IP, and it also has a running cost, which is the cost of public IP.

There are some transport protocol limitations which depends on the used protocol. Most of the developers use UDP protocol which provides fast solution to exchange data, but it is also considered as one of the connection-less protocols which does not support feedback about the communication state.

This solution has some security risks as each station has a public IP as in Fig.1. It will be seen to the whole Internet networks, which means it will be easy to be attacked. For example, using DoS (Denial of service) attack will kill this station and cut it off out of the system or at least such kind of attacks make the station busy all the time.

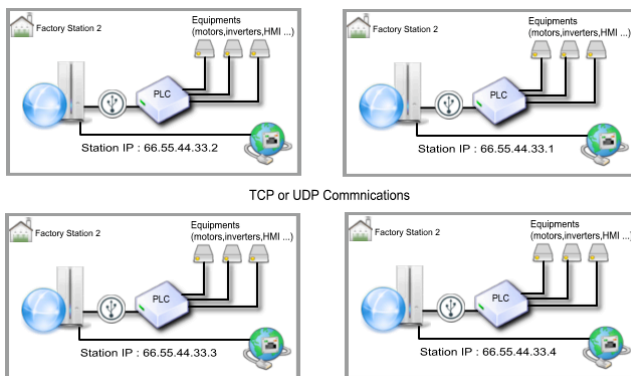


Fig. 1. Distributed SCADA Linked By IP

2.2. Distributed SCADA Linked By RF

In this solution the communications infrastructure used is the radio frequency such as, Microwave, AM, FM, VSAT... etc.

Such solution provides fast communications infrastructure but it is complex to build and install as shown in Fig.2. In addition the cost will be very high. For example, using Microwave technology is a good idea but you have to take in mind the cost overhead which will be between \$40,000 ~ \$60,000 for a distributed SCADA system consisting of three stations. This cost will be higher if more microwave towers and equipments

are needed. Distance between stations also increases the cost dramatically. This solution has some risks like, signal jamming.

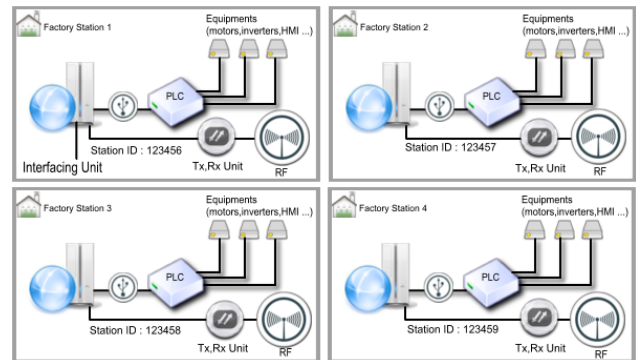


Fig. 2. Distributed SCADA Linked By RF

2.3. Distributed SCADA Web-Based

This solution is the most common solution used by industrial facilities. Web SCADA is easy to install, very effective, cost-less, provide interoperability and it is easy to be secured. Most of the Industrial OEM (Original Equipment Manufacturer) now provides there equipments with many protocols that can be communicate through TCP, UDP, FTP and HTTP. HTTP provides a communication channel to send everything like, control Signals, and multimedia streaming. Web SCADA was used as an alarming system and nothing father that before it is used in industrial control as shown in Fig.3. Evolution of the Internet and Internet based devices makes Web-SCADA a major player in the world of industrial control.

Web SCADA enables administrators to monitor the state in there facility anytime and anywhere. It also enables them to have a full statistics daily or every specific time and have it on their phones or PDAs [3]. The general structure of Web-Based Distributed SCADA system is very simple and needs no complex equipments to use.

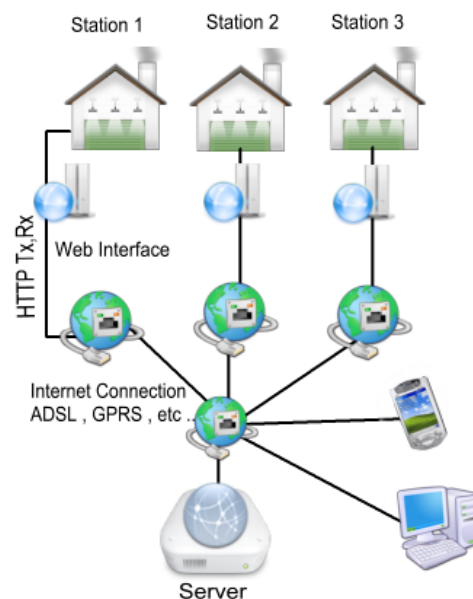


Fig. 3. Distributed SCADA Web-Based

3. Database System

In this paper, we use MySQL as a database-trigger system. We did a deep research on MySQL and Oracle as a database system. We chose MySQL for many considerations as in table 1. Table 1 has a comparison between the most two common database-trigger systems [2]. MySQL fits our needs, since it handles the logging and triggers efficiently which is the most important part in this solution. MySQL is free, simple, efficient, open source, common in web developing field, and easy to interface with web scripting languages like PHP and Ruby.

Table 1. Compare MySQL vs. Oracle

	MySQL	Oracle
Cost	Free	High
Strengths	Middle OLTPs	Large OLTPs
Server Load	Low	High
Popularity	Extremely	Extremely
Application Domains	Web Applications	Large DB Applications.

3.1. Database Trigger

All database systems provide a procedural code that executed automatically when certain event happens. This procedural code is used to guarantee the integrity of the information on the database. For example, when a new record which in our case represents a new data from any station added to the table then another value will be modified according to the trigger defined by the database designer as shown in Fig.4.

Database trigger provides us with many benefits. Below are the most important ones [2]:

1. It provides a faster application development. Because the database stores triggers, the trigger actions don't have to be coded into each database application.
2. It does not require complex maintenance procedures. Only the corresponding trigger program needs to be changed instead of changing the whole application, when the system policy is changed.
3. It improves the performance in client/server environment. All rules run in the server before the result returns.

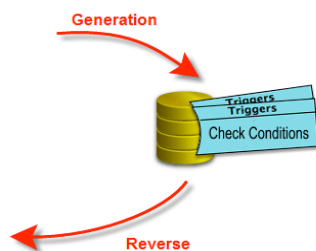


Fig. 4. Trigger Life Cycle .

4. Proposed Solution

Our solution consists of two modes of operation: The normal mode which is the station mode, and the administration mode.

4.1.1. Normal Mode

In this mode the station uses the protocol as a library or OCX control tool to communicate with the server to send station parameters. In addition to polling the server if there is any new data or control command available to the station. All what the SCADA designer have to do is to include the protocol agent in your SCADA application as a simple OCX tool. By configuring settings like server URL, station ID and authentication details the station will be able to communicate with the server and exchange data between all stations using the core server according to its permission. So the protocol OCX will look like add-ons to your SCADA system – most of SCADA designers provide the ability to use OCX tools or DLL libraries in your system. This protocol can be used in Linux OS environment using interface package like Wine. Fig. 5 represents a SCADA system designed by Microsoft Visual Basic 6. Most of the SCADA equipments provider now provide there SCADA equipments like HMI touch Screen with the ability of using OCX tools. OCX tools are considered as a powerful feature in SCADA soft wares.

```
Private Sub Form_Load()
    Agent.ServerURL = "www.control-server.com"
    Agent.StationID = "Jo-Aqaba"
    Agent.Auth_USR = "jo-aqaba-station"
    Agent.Auth_PWD = "11222dddfee3"
    Agent.EncryptionFile = "Encryption.txt"
    ↑
End Sub
```

Fig. 5. Trigger Life Cycle .

4.1.2. Administration Mode

In this mode the administrator has a full access to all stations. In this level the administrator sets the essential conditions for the whole system. For example, if the production of station 2 is increased over 10% then the server sends a signal to station 1 to send more raw materials to station 2. In this stage the administrator uses a simple conditions designer to add rules and will be provided with a simple GUI which will help him to write the system conditions easily.

4.2. Core Interpreter

Core interpreter uses PHP to generate the right trigger. It applies the theory of Neural Networks and in some cases Fuzzy Logic to modify the parameters in complex systems. This will be done by adaptive learning from the modifications. Then the appropriate database triggers are generated and then send it to the Database server [2].

In the case of multiple servers, one server will be active and do the entire job and the other servers will be redundant servers. In case of losing connection with a server the station will smartly notice long time-out so it sends a control message to the next server to be the main operator. Once all the stations engage with the new server, then the system will run again. This process of changing the server in charge takes about 2~3 seconds. Intelligent Transmission controller which is included in every station agent will calculate the transmission performance with all servers. It picks the fastest one and then sends an engage signal. This engaging signal will not be activated unless all the stations agree.

4.3. Web Interface

This solution uses normal web hosting thus it will be easy to login to any station through this server. This process will be like browsing a website and modifying any parameter in any station depending on permissions. Using security measures like HTTPS will be preferred. In case no networking security measures are available this solution applies crucial security measures. It detects a login process from different IPs which done in short time. Which is considered as a hacking process so the system will stop it and send an alarm message to the administrator? This security measures include also a complex encryption system.

Web Interface gives the ability to use your phone or PDA to monitor, control and configure the system in any station, this due to high performance alarming system which uses email alarm technique.

Conclusion

This paper proposes a new approach to control and monitor industrial facilities through the internet. This solution is compared with other solutions that are used commercially. The proposed solution outperforms most of these solutions. It eliminates many of the existing concerns and costs associated with IP technology that are adapted with most industrial facilities.

References

- [1] Shiflett C. HTTP Developer's Handbook. Sams Publishing March 21, 2003.
- [2] Oracle® SQL Developer Supplementary Information for MySQL Oracle.: April 2008
- [3] Zecevic, G; "Web based interface to SCADA system," *Power System Technology, 1998. Proceedings. POWERCON '98. 1998 International Conference on* , vol.2, no., pp.1218-1221 vol.2, 18-21 Aug 1998.
- [4] Duo Li; Serizawa, Y.; Mai Kiuchi; , "Concept design for a Web-based supervisory control and data-acquisition (SCADA) system," *Transmission and Distribution Conference and Exhibition 2002: Asia Pacific. IEEE/PES* , vol.1, no., pp. 32- 36 vol.1, 6-10 Oct. 2002.