Bogdan DZIADAK[1], Łukasz MAKOWSKI[1], Andrzej MICHALSKI[1,2]

Warsaw University of Technology (1), Military University of Technology (2),

Environmental monitoring system with instant communication

Abstract. *Providing clean water with particular quality is one of the main issues in environmental protection. System for this objective must measure dozens of values and provide them reliably to staff of water treatment company. In complex situation such as leak or waste polluting water reservoir the system must help emergency response teams to solve the problem as quickly as possible.*

Streszczenie. *Dostarczanie wody pitnej o zadanej jakości jest jednym z głównych problemów w ochronie środowiska. System do tego zadania musi mierzyć dziesiątki wartości i wiarygodnie udostępniać je pracownikom firmy zarządzającej wodociągami. W krytycznej sytuacji takiej jak wyciek lub zanieczyszczenie rezerwuaru wody, system musi wspomóc służby ratownicze w rozwiązaniu problemu tak szybko, jak to możliwe. (**Monitorowanie stanu środowiska z ciągłą komunikacją**)*

Keywords: measurement system, open protocols, xmpp.
Słowa kluczowe: systemy pomiarowe, otwarte protokoły, xmpp.

Introduction

In this article we discuss progress of our research in area of environmental protection with emergency alert system built upon set of wireless, distributed measurement stations. Particularly we are focused on scenario where operation area is clean water source such as lake.

Shortly comparing these two, current one is considered to have increased number of sensors, better communications, better field operation in changing environmental conditions, and improved user experience. Visualization of part of the system which performs measurements on the background of real deployment area is shown on figure 1, below. Except of Measurement Devices (MD) there is also a local On-shore Access Point (OAP) which provides link additional to direct yet expensive GSM/GPRS connection.

Dobczyckie Lake, which is our system test-bed, is an artificial reservoir in highlands valley. As it has dam and several tributary streams water quality must be controlled at several places. The lake is a clean water source for at least 0,5 mln people.

Related work and contributions

In our previous project one of successfully fulfilled goals was a creation of point-to-point data transmission between Measurement Device (MD), which was freely floating on lake, and a system server. In that research data was transmitted over GSM/GPRS channel [1]. We found some problems and drawbacks of that solution although it paved the road for our current project.

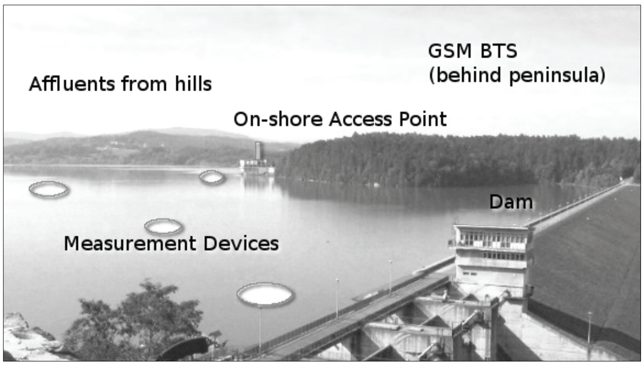

Fig.1. Main deployment area for the measurement system

For water treatment there are legal regulations such as EU Commision directives which then are implemented by member states [2]. These regulation are limiting appropriate values of measurable water parameters. Exceeding these limits initiates emergency situation procedures.

System requirements specification

First of all we consider numerous set of devices per single observation area such as lake. However, in our test-bed we are building only a couple of such nodes architecture is designed for further investments and product growth. To achieve this goal we establish a stable layer of communication that will be interface between measurement devices and higher hardware layers such as data mining server and visualization terminals.

Secondly, there are no strict limits for number of sensors and number of sensor types present in the system. Number of sensors active in particular spot is limited only by hardware of device working in that spot to which these sensors are connected.

To provide reliable, stable and sustainable solution, we have to have system that is ready for changes in legal regulations related to water quality measurements which may happen in some future. Therefore we need flexible protocol that is easy to extend and expand by features and variables that are unknown at this moment. XMPP fits these requirements.

We assume diversity of hardware especially for communication subsystems. In practice it will be choice between more expensive but widely available GSM/GPRS channel and short-range, local, wireless sensor network. In any case there must be guarantee for soft real-time data transmission at least during emergency situation.

Obviously measurements must be transmitted reliably and it involves some level of security. We cannot allow malicious attacker to rise false alarms and as well as we cannot let our system to be misguided by terrorist poisoning the water reservoir. Of course system safety is a complex matter and ultimate security is not practically available although we must follow guides for high standards of quality.

Summary of these base requirements in manner following idea of SWOT analysis is shown in table 1.

Table 1. Target requirements for system

Type	Description	Kind
Number of devices	Large (dozens to hundreds)	Challenge
Sensors per device	Large (tens)	Challenge
Link types	Many (i.e. ISM, GSM, eth)	Challenge
Reliability	High (soft real-time, 24/7)	Challenge
Environment	Hard (outdoor, 24/7)	Threat
Human factors	Possible (intrusion, theft)	Threat
Scale effect	Possible (scalability)	Opportunity

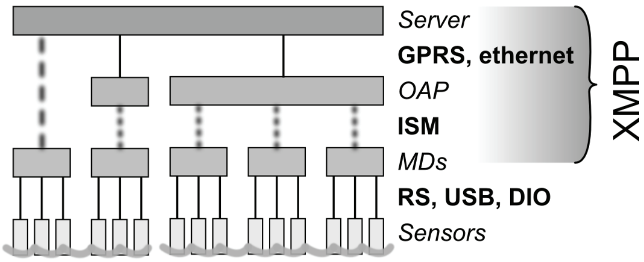

Fig.2. System architecture – general view of transmission layers

To cope with these requirements we use Extensible Messaging and Presence Protocol (XMPP) which is widely recognized as flexible and reliable choice [3]. XMPP is common creation of Jabber Foundation and IETF with strong participation of large and constantly growing community of freelancing developers, academic engineers, and huge companies [4]. General architecture of transmission layers is shown on figure 2.

Brief introduction to XMPP

For most of Instant Messaging (IM) services there is a company solely controlling servers that are providing particular service. XMPP architecture is more like e-mail servers, which means that every organization can control its own infrastructure. There are more similarities between these two solutions and one of the most noticeable is XMPP addressing, which highly resembles familiar e-mail style: name@server.domain.

XMPP is not p2p communication and for transmissions between two users a distributed network of independent servers is needed in most cases. Of course "user" is here a wide term that includes not only human beings but automated processes also. Authentication and authorization are two important transactions that usually needs to be performed before any other transmission might occur. Former one opens channel to server with Simple Authentication and Security Layer (SASL). Latter one almost completely nullifies the problem of unwanted messages which in e-mails environment have infamous name of "spam". In this procedure, which from user point of view is called "subscription", both sides add each other to a private list called "roster". Roster is stored on the server and received by client after successful authentication.

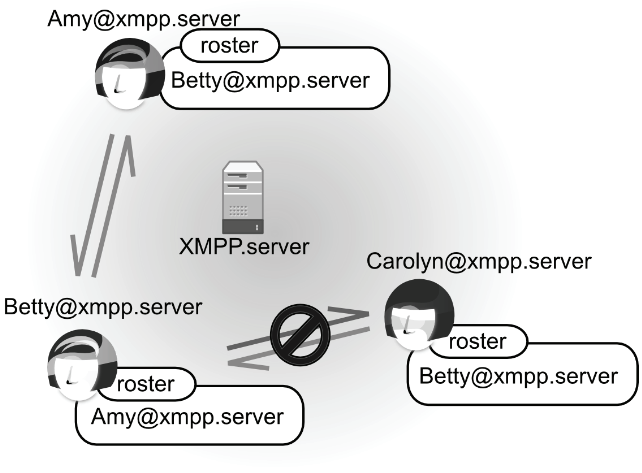

Fig.3. Sample entities and message routes

Simplified situation with only one, private server is shown on figure 3 and it is case from our project as well. In the shown example Amy can communicate with Betty as they both have each other in the roster and are connected to the same server. Betty and Carolyn cannot send to each other due to only one-sided subscription, which perhaps should be finished in the future to let them communicate.

Every message is transmitted (push) from sender through server to recipient immediately. If recipient is offline server holds the message. Availability "status" of known, authorized "buddies" (roster entries) with its longer textual description is visible to each roster owner and updated in real-time. It means that there is constant and reliable channel control on both sides. This is difference comparing to e-mails, where sender never knows neither when nor if the message will arrive.

Main drawback of XMPP, which is based on XML, is its bloated, redundant transmission of structural elements. This problem is quite successfully coped with thanks to compression that comes bundled with transmission data encryption done with standard and popular Transport Layer Security (TLS) [5]. On the other hand there are implementations of XMPP designed for low power, embedded devices with limited hardware resources [6].

Data acquisition and transmission

In this project we are in cooperation with Institute of Electron Technology, Cracow and Institute for Sustainable Technologies – National Research Institute, Radom. They are providing sensors and mechanical parts. Our part was to provide MD mainboard and all elements above it such as: communication layer, system server, up to visualization.

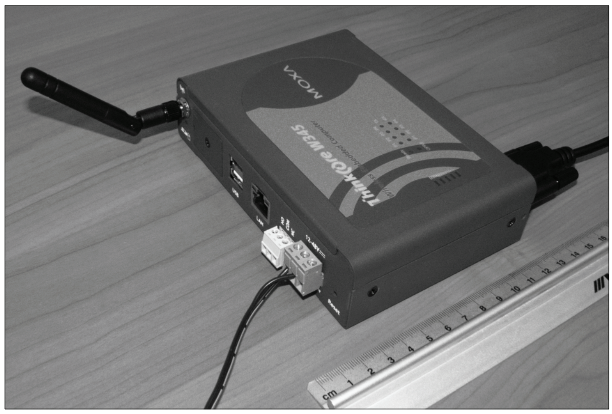

Fig.4. Main controller for MD – Moxa ThinkCore W345

For MD main controller we chose industrial computer Moxa ThinkCore W345, shown on figure 4. Machine is based on ARM7 microprocessor and provides four RS-232/485 channels, two USB ports and one relay. Unfortunately we have to deal with quite heavy constraints in memory and file-system space.

Default software environment on this machine is a Linux operating system, upon which we deploy software developed in our project. Multithreaded application architecture has four crucial modules:

1. control of transmission hardware,
2. logical management of XMPP connection,
3. control and queuing of data messages,
4. XML-RPC procedures for sensor layer API.

We are considering the fact that water quality monitoring must be performed regularly. Measurements of different quantities should be processed as data sets at specific paces although sometimes information must be obtained and sent to staff visualization computers in soft real-time. However, to continuously keep GPRS connection up on each of MD is expensive. Therefore to make solution cheaper a group of MD communicates by free of charge ISM frequencies with nerby on-shore transceiver – OAP.

OAP having data received on short range wireless channel from a set of MD, relays it to higher layers of system structure. In simpler case OAP device works just as a network router, but in case that single OAP is covering large number of MD we also consider a possibility to use XMPP in s2s manner where one of XMPP servers is available on OAP. This approach with more system layers might become useful when other water companies will join and together will participate in central monitoring of water resources in the country.

Ethernet port (IEEE 802.3), which is available on chosen Moxa platform, is to be used only as a maintenance connector in most cases. However this is the cheapest channel therefore in some cases, whenever available, the ethernet connection can become main link.

Server side

We chose recent, stable Debian GNU/Linux release for main server operating system. It creates environment for many, important network services, including visualization and data mining, although in this paper we are discussing only XMPP daemon and problems within its proximity.

Until now we used a special entity that was an automated client (so called “bot”) working as a receiver for messages from MPs. Task of this bot was to analyse incoming transmission and put information into database. This approach is perhaps the easiest one as writing such bot was not much different in comparison to writing entity that was sending measured data from MPs. However this entity application was running on the same machine that was running XMPP daemon. Another benefit from this approach is separation of applications which may increase reliability of the system as a whole. On the other hand it is not optimal solution considering performance and we must inspect this factor if we want to make the system available for more than one water treatment company.

More advanced choice, which we have under development now, is an XMPP server internal “service”. Such service should not be confused with system networking daemon. It is internal part of XMPP daemon providing specific features to all entities that subscribe it. Mostly this XMPP extension is used as a “transport” between XMPP clients and different IM networks.

To improve performance benchmarks and increase system scalability we moved from jabberd2 to ejabberd. However former one is very good example of FOSS application clearly written in C language, we will use latter one which was created with Erlang. Erlang is a programming language created by Ericsson for telecommunications and has built-in concurrency mechanisms. This improves scalability of our solution and creates opportunity to gather more water treatment companies under one monitoring system, which will be able to provide information about water resources on state level.

Visualization

Data presentation software can adjust amount and rate of data provided to system users with use of instant messaging approach. Because XMPP is only a protocol and not an application somehow limited to one platform or technology, variety of user interfaces might be created for different needs. For example a gateway between XMPP server and mobile phone is easy to build even on operator level [6]. Furthermore user interface can be built up and dynamically updated in real-time, according to messages from automated entities, to bring staff's attention to some specific parameters or situations [7]. Currently we are investigating these ideas.

Conclusion

In this paper we presented current state of our research and development of system for water quality monitoring. This part of system can be simplified to three devices: Measurement Device (MD) works as a client to XMPP service that is running on main system server. Optional, is coastal Access Point (AP) that is not only extending range and improves quality of communication but is also forming group of MD. All devices operate with TCP/IP stack and this approach is tightly related to idea known as “Internet of things”, where every device can be remotely accessed so its software can be easily updated.

We assert that system developed with use of open and free instant messaging protocol – XMPP, is reliable, scalable, and therefore: proper solution for environmental alert system with soft real-time responses required.

Work of Łukasz Makowski has been supported by the European Union in the framework of European Social Fund through the Warsaw University of Technology Development Programme.

REFERENCES

[1] Kalicki A., Michalski A., Staroszczyk Z., Dziadak B., Makowski L., "Sensor Networks for Petroleum Derivative Pollution Monitoring System", *Problemy Eksploatacji. Maintenance Problems.* 4/2006 (63)

[2] "Council Directive 98/83/EC of 3 November 1998 on the quality of water intended for human consumption ", consolidated version, 1998L0083 — EN — 07.08.2009 — 002.001 — 1

[3] Saint-Andre P., ed., "Extensible Messaging and Presence Protocol (XMPP): Core", *RFC 3920*, Oct. 2004.

[4] Saint-Andre P., "XMPP: Lessons Learned from Ten Years of XML Messaging", *IEEE Communications Magazine*, Apr. 2009

[5] Michalski A., Dziadak B., Makowski L., "Mobile Observation Point - Selected Aspects in Design and Signal Transmission", *IEEE Transactions on Instrumentation and Measurement,* Volume: 57, Issue: 8, 2008

[6] Hornsby A., Bail E., "µXMPP: Lightweight implementation for low power operating system Contiki", *International Conference on Ultra Modern Telecommunications & Workshops*, 2009.

[7] Heng-Te Chu, Wen-Shiung Chen, Yi-Hung Huang, Jeng-Yueng Chen, "A novel design of instant messaging service extended from short message service with XMPP", *Fifth IEE International Conference on 3G Mobile Communication Technologies*, 2004.

[8] Book M., Gruhn V., Mucke G., "An Instant Message-Driven User Interface Framework for Thin Client Applications", 21st *IEEE/ACM International Conference on Automated Software Engineering*, 2006. ASE '06.

***Authors**: dr inż. Bogdan Dziadak, Politechnika Warszawska, Instytut Elektrotechniki Teoretycznej i Systemów Informacyjno-Pomiarowych, ul. Koszykowa 75, 00-662 Warszawa, E-mail: bogdan.dziadak@ee.pw.edu.pl;*

dr inż. Łukasz Makowski, Politechnika Warszawska, Instytut Elektrotechniki Teoretycznej i Systemów Informacyjno-Pomiarowych, ul. Koszykowa 75, 00-662 Warszawa, E-mail: lukasz.makowski@ee.pw.edu.pl;

prof. dr hab. inż. Andrzej Michalski, Politechnika Warszawska, Instytut Elektrotechniki Teoretycznej I Systemów Informacyjno-Pomiarowych, ul. Koszykowa 75, 00-662 Warszawa, Wojskowa Akademia Techniczna, Instytut Systemów Elektronicznych, ul. Gen. Sylwestra Kaliskiego 2, 00-908 Warszawa, E-mail: anmi@iem.pw.edu.pl.