Mobile SCADA with Thin Clients – a Web Demonstration

Lukas Tan, Dr Ken. Taylor CPEng, MIEAust

***Abstract*-- Mobile Supervisory Control and Data Acquisition Systems (SCADA) with thin clients offer an approach that is cheap to deploy and run, while allowing a great deal of flexibility in implementation and functionality. This paper describes an implementation of a Mobile SCADA demonstration system. It details the system's implementation and function, discusses how the mobile telephone network and thin clients can be used to implement complex distributed applications and offers a view to the applications for such systems.**

***Index Terms*-- SCADA, Mobile, Thin clients, Control**

I. Introduction

SYSTEMS for Supervisory Control and Data Acquisition (SCADA) can be divided into two groups. One containing complex, flexible controllers at a premium cost, and simple controllers with restricted functionality at low cost an example of which is back to base home security systems. We are endeavouring to create SCADA systems with the flexibility of existing high-end systems at a price point approaching that of existing low-end systems. We aim achieve this by utilising thin clients to act on the world, networked services and by utilising existing communications infrastructure.

The costliest part of many control systems is the communications infrastructure. Communication costs are minimised with Mobile SCADA by utilising the mobile phone network. Hardware costs are low with cheap, mass - produced mobile telephony devices costing from A$100 to A$400. Data transmission costs on the mobile phone network are much higher than for wired systems but economical for small volumes of data and these costs are also likely to drop with the advent of 3G networks. Additionally, the burden of infrastructure maintenance and upgrading is removed, and the coverage offered surpasses that of any specifically developed communications system.

A second means by which costs can be reduced without sacrificing flexibility is the choice of hardware used to implement both ends of the system. The Thin Client architecture used by the Mobile SCADA system features a web server as the central point of the network, microprocessor based thin clients at the control points, a web browser interface for humans and an XML interface to enable applications anywhere on the Internet to interact with the client devices. This scheme allows for simple low cost devices that interact with the physical world; that combined with networked based services have a functionality including integrated relational database that could otherwise only be provided by high-end controllers.

This paper deals with the implementation of a demonstration system showing the possibilities of Mobile SCADA with Thin Clients. The demonstration includes a system to be controlled, it's thin client controller, interactions between the controller and server, and the server based user interface. It is intended that users control the system observe its functionality and then copy the functionality demonstrated for their own applications.

II. The Demonstration System

The object of control selected for the demonstration system was two linked water tanks. The system is shown schematically in Figure 1 and a photograph of the actual model is shown in Figure 2.

Lukas. Tan is a student at the Faculty of Engineering and Information Technology, Australian National University, Canberra, ACT 0200, Australia (e-mail: s3164365@student.anu.edu.au).

Ken. Taylor, is with the Division of Mathematical and Information Sciences, CSIRO, GPO Box 664, Canberra ACT 2601 (e-mail: ken.taylor@csiro.au).

ICITA2002 ISBN: 1-86467-114-9

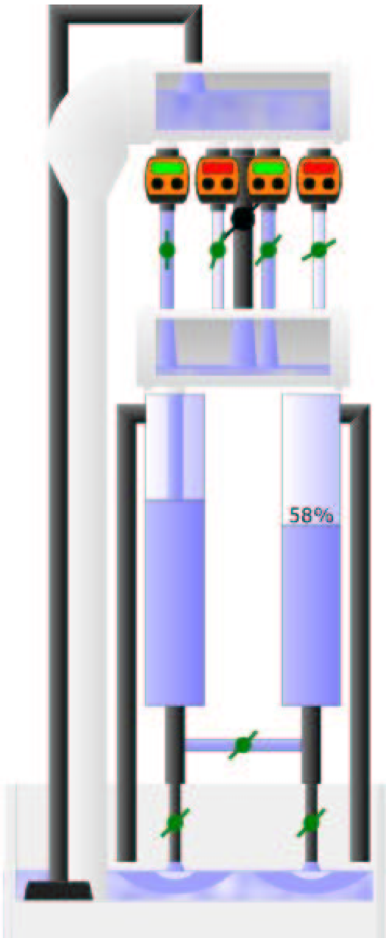

Figure 1: The Twin Tanks Model

Figure 2: Photograph of the Twin Tanks Model

The users task is to attain a fixed water level in the second of the two tanks by controlling the flow into the first. Both tanks drain through fixed size openings, and are connected at their bases through another fixed pipe.

Variable flow into the first tank is provided by means of four ball valves, each with a flow rate twice that of it's left neighbour and which can be switched open or closed. The set of combinations of these valves offers 16 rates of flow. Water level in the second tank is measured with a capacitive sensor. The valves are operated by a cheap micro controller, which also acquires the feedback data from the capacitive sensor.

While normally the mobile phone network would be used for communication the demonstration system communicates with its server via a Local Area Network This avoids the expense of transmitting data across the GPRS mobile phone network in a situation where a wired network is easily available.

The overall system architecture is shown in Figure 3. The device server stores the data acquired by the thin client, and controls the flow of requests to it. Polling is used to regularly request the latest set of data from the controller, while control requests are passed on as they occur. Data acquired by the server is stored in a SQL database, which is then accessed to provide feedback to the system users. The database is maintained so that only the most recent set of records are stored, and is also used to store all client control requests.

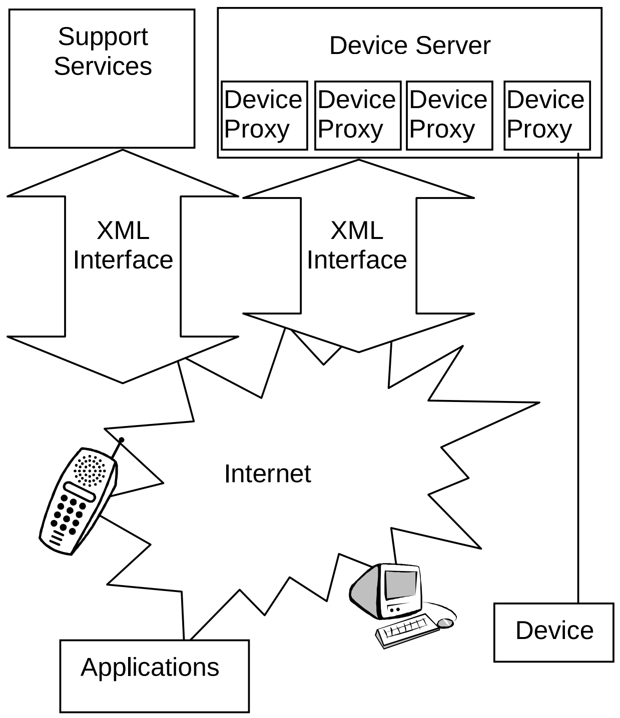

Figure 3: Overall System Architecture

The user interface to the SCADA demonstration system is a web page [1] (shown in Figure 4). This provides various forms of feedback from the system, and allows control of the valves filling the first water tank.

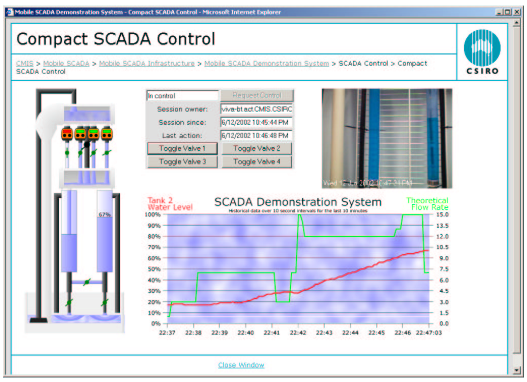

Figure 4: Browser interface to demonstration system

The control and feedback methods available are:

- An animated drawing of the tanks system (shown in Figure 1), indicating the status of the four valves and the water levels in the tanks;
- A graph showing the flow rate into the first tank and the water level in the second tank over the last ten minutes;
- A web camera which operates independently of the SCADA system to provide the user with verification of the feedback being reported by the SCADA system;
- Buttons for operating valves that are only offered to a single user at a time on a session basis.

Other users can request control and this request will be granted if the current user has been inactive for a set amount of time. A client side script keeps the web page current by polling the web server every 10 seconds until the browser is closed.

III. Implementation Details

The twin tanks system is constructed from a variety of readily available garden irrigation and domestic stormwater piping and fittings. The water supply for the four valves is from a small reservoir directly above the valves, which is filled from a large sump at the base of the system by a conventional garden pond pump. The flow from the four valves is channelled into the top of the first tank by a short section of the same pipe used to make the valve reservoir.

The flow rates through the four valves, the drains of the two tanks and the link between the bases of the two tanks is controlled by manually adjustable ball valves, which are configured to give the system properties suitable for it’s current use. A large tub acts as the sump and main water reservoir for the whole system, and contains a metal grid to which the components of the system are attached.

The four controllable valves are electronic ball valves sold for use as battery powered garden tap timers. Their timing mechanisms were removed and the valve drive mechanisms wired to a relay box, which drove the valves when given small electrical signals from the thin client controller. The water level gauge consists of a coated metal tube with a wire running down it’s length, with a modified capacitance meter kit circuit converting the capacitance measured between the wire and tube to a periodic signal fed into the controller.

While several controllers have tried for interfacing to the network based support services. The controller used for this demonstration is based on the RCM 2100 RabbitCore [2] microprocessor core module.

At the server, a small application polls the controller for the most recent data, parsing the XML it receives from the controller and inserting the data values into the database.

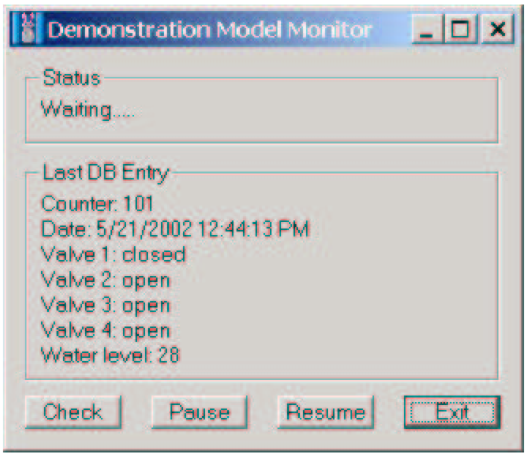

Figure 5: Controller Data Retrieval Application

A VBScript based Active Server Page (ASP) is used as the session manager, storing the details of the current session owner as global ASP application variables and using locks to ensure that only one person is in control at any one time. Client control session requests and valve toggle requests are processed by the session manager, which returns XML documents to the client summarising the current state of the session and the success of toggle requests. For every valve toggle request received, the session manager makes the corresponding toggle requests directly to the controller, as well as logging toggle requests to the database.

The web pages viewed by the client use Scalable Vector Graphics, an XML based vector graphics format, to provide the animation of the model and the historical data graph. The SVG XML is manipulated via the XML Document Object Model (DOM) provided by the SVG browser plug – in. Through the DOM, the XML for the SVG drawings is manipulated to alter the animation shown in Figure 1) and graph (shown in Figure 6) images to reflect state changes at the tanks system. The information describing the latest state of the model is retrieved by the client via a web based XML interface to the database on the server. This interface allows the specification of SQL queries, returning the query results in XML documents that are then parsed at the client, again via the XML DOM.

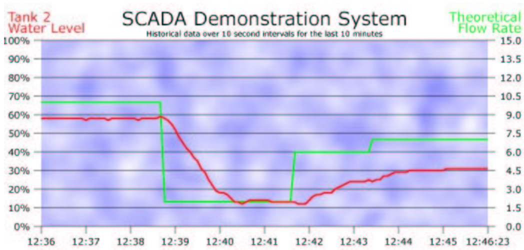

Figure 6: SVG Historical Data Graph

The client control of the system is based around a HTML form (shown in Figure 7 and Figure 8), which has buttons allowing the user to either request a control session if they are not already in one, or to toggle the four valves if they own the current control session. This form also shows details about the current control session and any submitted control actions, and is updated using values in the XML document returned by the session manager on the server. All client activities are performed with JavaScript. The feedback XML documents are processed asynchronously to avoid locking up the browser while new information is being acquired from the web server.

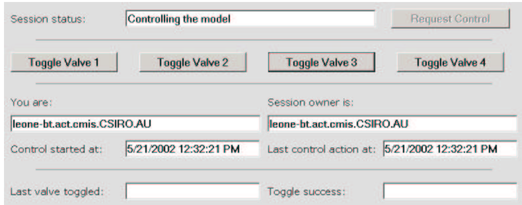

Figure 7: Control Panel for the Control Session Owner

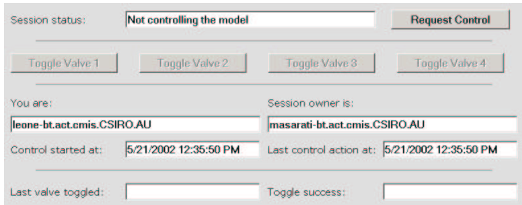

Figure 8: Control Panel when not the Control Session Owner

The overall architecture of the system drawn along controller – server – client boundaries is shown in Figure 9.

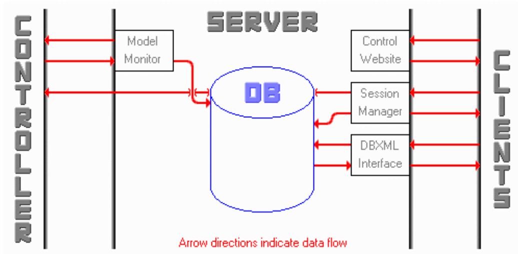

Figure 9: Demonstration System Architecture

IV. Future Work

Whilst a controller based on the RCM 2100 RabbitCore [1] microprocessor core module is used for this demonstration we have since developed a solution using an Internet Engine [3] developed by Fred Eady [4] which is cheaper, provides analog and digital I/O and is reprogrammable over the air. We intend to convert the demonstration system to use this device and to enhance the demonstrator by allowing Internet users to reprogram the controller as well as operate the model.

To further demonstrate system capabilities we intend to add control by mobile phone. Commands to the demonstrator will be issued by dialing different a phone number to toggle each of the four valves with responses received by SMS message which detail the current state of the system. Also to demonstrate the flexibility of the architecture we intend to add a higher level of control by implementing a PID loop on the controller, allowing the user to specify a target water level, which would then be automatically attained.

V. APPLICATIONS

The scope for usage of cheap and flexible SCADA systems such as this is broad and covers many fields outside those traditionally using SCADA systems.

The ability to control things remotely allows the deployment of cheap SCADA systems for applications such as remote weather stations, radiation or toxin detectors, irrigation monitoring and vehicle tracking and vending machines.

VI. CONCLUSION

The design, construction and successful operation of the Mobile SCADA system documented here provides a demonstration that utilises network based services designed to support thin client devices to deploy complex control and data acquisition applications with sophisticated interfaces at low cost. Low cost is achieved is achieved by combining simple devices that interact with the world, communications over the mobile phone network, a web browser interface and support services including a relational database that are shared across many applications and devices. Third parties can modify the demonstration system to try out ideas and the techniques demonstrated can be copied. Provision of the services shown under an application service provider model will allow developers to quickly and cheaply deploy distributed control and data acquisition solutions for their clients.

REFERENCES

[1] Tan, L; Taylor, K (2002) *"Compact SCADA Control"*, CSIRO MIS SIS: Canberra, Australia. Available online at: http://mobile.act.cmis.csiro.au/sms2/tanks/SCADA-Control-Compact.htm

[2] RCM2100 RabbitCore *"Rabbit Semiconductor product page"*, Rabbit Semiconductor: Davis, CA USA. Available online at: http://www.rabbitsemiconductor.com/products/rcm2100/index.html

[3] *"Internet Engine"*, EDTP Electronics: Merrit Island, Florida USA. Available online at: http://www.edtp.com/internet_engine.htm

[4] Eady, Fred (2001) *"An S-7600a/PIC16F877 Journey"*, Circuit Cellar Online, Vernon, USA; January 2001. Available online at: http://www.chipcenter.com/circuitcellar/january01/c0101fe1.htm