Technical Note

Note: Technical notes are aimed at promoting discussion. The views expressed are not necessarily those of the editors or the Australian Acoustical Society. Contributions are not formally peer-reviewed.

LOW COST REMOTE DATA ACQUISITION SYSTEM

Kristoffer K. McKee, Gareth L. Forbes, Ilyas Mazhar, Rodney Entwistle and Ian Howard
Department of Mechanical Engineering, Curtin University, Perth, Australia, and CRC for Infrastructure and Engineering Asset Management, k.mckee@curtin.edu.au

Remote data acquisition (RDAQ) is required in a number of vibration and noise measurement settings. Specialised systems exist for RDAQ, however they require significant investment to set up. This paper describes a low cost method of RDAQ which utilises a laptop computer, data acquisition system and a USB internet dongle. The system described in this paper would allow a professional vibration/noise measurement specialist to modify their existing data acquisition system for remote use with minimal cost. The system was tested both in an industrial and an academic setting for verification.

INTRODUCTION

It is common for researchers and maintenance professionals who perform diagnostics on machines to travel considerable distances to the location of a machine, acquire the needed data from the machine via a data acquisition system (DAQ), and then return to the office for data processing. As a result, a considerable amount of time, money and energy is lost in the process. Problems sometimes arise when unexpected complications and failures in the machine arise in between visits and are not captured by the DAQ, resulting in a loss of valuable information.

The ability to remotely acquire the data from onsite sensors would solve these problems and possibly create the ability to perform condition based monitoring of a machine or any other kind of continuous monitoring required. Sensors, in this case, would be left on site to continually or periodically monitor the machine. Data gathered on site could then be processed and key indicators extracted to be uploaded via internet to a server that is accessible to the user.

A GENERIC LOW COST RDAQ SYSTEM

There are two main parts to this RDAQ system: hardware and software. The hardware components can be replaced with similar parts than those presented as long as they are able to meet the task. The software components are sometimes based on the hardware components, such as those used for the data acquisition, while others are freeware and are only dependent on the type of environment used, such as command line interfaces versus graphical user interfaces.

Hardware

The remote data acquisition system (RDAQ) usually consists of 5 pieces of standard hardware, as given in Table 1. Figure 1 also shows an example of the system used for this study.

Figure 1. Set up for Remote Data Acquisition System

Software

It is assumed that the RDAQ system will be unmanned for the majority of the time. Hence, the software utilised must automate the major processes needed. Three main steps are required in the RDAQ: data acquisition, data processing and uploading the data. These three steps were programmed to occur at different times so that the maximum amount of computing resources available could be dedicated to the task.

Data acquisition

The purpose of the work presented in this paper is to outline a process to convert an existing DAQ system into a remote(R) DAQ system. In the example system used in the RDAQ measurements presented later in this paper, two DAQ systems were employed, being:

1. SignalCalc ACE Quattro by Data Physics (controlled by propriety software)
2. National Instruments DAQ (controlled by Matlab)

Data Acquisition is usually performed with a program that is associated with the DAQ hardware. In this particular case, the National Instruments' cDAQ 9178 system was controlled using a script written in MATLAB®, which possibly has become available to users in recent years. MATLAB® was chosen over National Instruments' Labview environment due to the accessibility of the program and expertise in writing

Table 1. RDAQ hardware requirements

	General hardware component	Specifications of hardware used in this study. c.f. Figure 1
1	Portable DAQ	The DAQ used was the National Instruments' cDAQ 9178 system, along with 4 B&K accelerometers to gather the data. The data was sampled at 25.6 kHz. Another DAQ also utilized was called Quattro by Data Physics. Quattro has a four channel sensor input, a trigger input, and two channels for output. Newer systems, such as the B&K Pulse System, contain a static IP address, and thus allow easier access over the internet.
2	Computer	Dell Latitude with an Intel® Core™ i5-2520M CPU, running at 2.50 GHz with 4 GB of RAM and 300 GB of hard drive space. It was running with Windows 7 Enterprise Edition, 64-bit operating system using Service Pack 1. A suitable computer would be able to analyse the collected data in real time, if necessary, or in the amount of time provided between collecting the data. Slower or less powerful processors, such as the Intel® Core i3, Celerons, Atoms, or Pentium series, may not provide the processing power needed to perform real time calculations. Experimentation with this hardware prior to deployment in the field is necessary.
3	External hard drive	500 GB external hard drive, which was deemed as being sufficient for the task.
4	USB dongle from an internet provider	Telstra Elite on the Next G network. Telstra was chosen due to its availability as an internet service provider Australia wide. A suitable internet provider should be selected based on coverage, strength of signal, and available data plans.
5	Powered USB hub (optional depending on the distance between the machine and the DAQ, and between the DAQ and the computer)	This was used to boost the signal from the DAQ to the computer since the location of the sensors was found to be too far from the computer. In some instances, the distance between the DAQ system and the computer can be at least 50 metres; hence the signal must be amplified in order to successfully transmit it. This component, which is not shown in Figure 1, may not be necessary in those instances in which the location of the DAQ and the computer are in close proximity.

scripts in the program.

For environments in which scripts are not able to be written to control the DAQ process, and only a GUI is provided, additional software is needed in order to carry out this process. This was the case when using the Quattro DAQ. In such cases, two software components were utilised, AutoHotKeys and Devcon, both of which are freeware. AutoHotKeys is a Windows scripting language that allows the user to automate the clicking and inputting of needed information. Using a text editor, scripts can be created that recognise the appearance of graphic user interfaces (GUI), locate and input strings of characters or numbers at predetermined locations on the GUI, and click on buttons found on the GUI to proceed with the tasks. Devcon is a system tool provided by Microsoft in their Windows SDK for Windows 7 package. Devcon is utilised through a command line interface, such as the dos command window, and thus can be called by AutoHotKeys via batch files. It provides the ability to turn off and on drivers for the DAQ. This is needed during those unforeseen times in which either the DAQ or the Windows operating system has been rendered inoperable and must be rebooted in order to proceed. Devcon must be run with administrator privileges.

Data processing

The extent of the data processing step is dependent on the amount of information that is desired to be uploaded onto a remote server or system. In most cases, key indicators are extracted from the collected data and are uploaded to represent features of the data. This is preferred over uploading the raw data collected because of the cost per megabyte to upload data. In this particular case, MATLAB® was utilised to process the data due to its ability to handle large amounts of data in matrix form efficiently. Due to the speed of the computer, and the efficiency of MATLAB®, data processing was performed in a matter of seconds and thus deemed to be done in 'real time'.

Uploading the data

Uploading the data may require a secure ftp (SFTP) program such as WinSCP, which is also freeware, or the use of the ftp command at the dos command window. In this particular case, a session of WinSCP was pre-configured with the information required to access a local server. In addition, a text file containing commands to open a session, load the data files, synchronise files, and close the session were created to run the ftp session.

EXAMPLE LOW COST RDAQ SETUP

The following is a description of the procedure on how the RDAQ was implemented, using SignalCalc ACE software with Quattro DAQ within a Windows 7 environment. Prior to creating and implementing the needed scripts, three software packages were needed, as given in Table 2 (all the listed software is freeware).

System Schematic

An example remote data acquisition setup is shown in Figure 1. The general system is shown schematically in Figure 2 consisting of 4 parts: a DAQ system, a computer, a wireless dongle, and a web server. The DAQ system, computer and wireless dongle are directly connected, and thus have information flowing back and forth between them. The web server could be located in a project office or this facility could be obtained from an internet service provider. The task of remotely measuring data and transferring it to the server is

Table 2. Software required in RDAQ system

	Software	Description	Download link
1	WinSCP	WinSCP is used as the SFTP or FTP program to transfer files from the computer on site to the local server. The program is freeware and provides the level of security requested by the server the data is placed on, as well as the ability to control it using a script.	http://winscp.net/eng/download.php or http://download.cnet.com
2	AutoHotKeys	AutoHotKeys is a windows scripting language, and is used as a means of clicking on parts of the screen without needing the user to be at the remote location.	http://www.autohotkey.com (tutorials provided) or http://download.cnet.com
3	Devcon	Devcon is used as a means to stop and reset a DAQ hardware in the windows environment.	http://suppport.microsoft.com/kb/311272
4*	Teamviewer	TeamViewer is a remote desktop software that needs to be placed on the laptop computer on site as well as the local computer in the office. This program can be used to login to the remote computer if it is thought to have frozen, or investigation is needed determine if the procedure is being carried out correctly on the remote computer.	www.teamviewer.com
5*	LogMeIn	Similar to TeamViewer except it can be accessed through a web browser and does not require stand-alone software.	www.logmein.com

*Optional software which was used to help to monitor the progress of the RDAQ system remotely

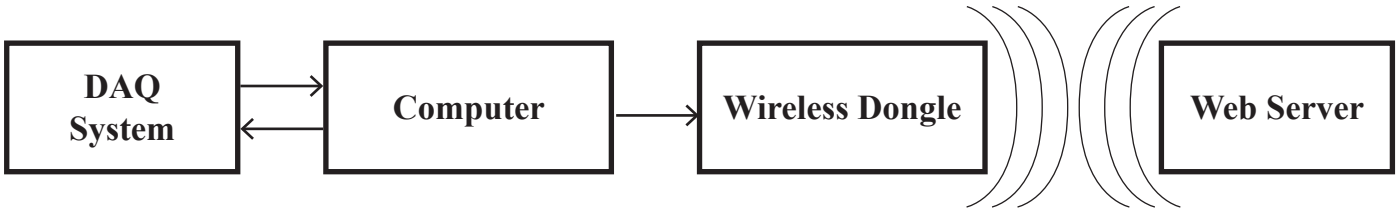

Figure 2. Schematic diagram of RDAQ System

Table 3. Main processes involved in RDAQ system

	Process	Description	Software used
1	Data acquisition	Monitors DAQ software, obtains data, controls file names and saves data to the remote computer	AutoHotKeys, Devcon, DAQ specific software
2	Data processing	Runs any processing of data before it is to be uploaded (this step is not required if the raw captured data is to be uploaded to server)	Any data processing software required e.g. MATLAB
3	Uploading data	a) Connect and monitor internet connection with USB dongle (not required if LAN connection used) b) Synchronisation of data files on remote computer and server	USB dongle software, AutoHotKeys, WinSCP

performed by utilising three main scripts to govern the flow of information; these three main scripts are shown schematically in Figure 3.

The tasks the scripts undertake are outlined in Table 3. The process can also be made to be automatic upon booting up the computer by placing the scripts into the "Startup" folder in Windows. An overall view of this 3 step process is controlled by 4 main scripts as outlined in Figure 3.

EXAMPLE IMPLEMENTATION

The example system was tested in two locations

- onsite at a remote pumping station for Gold Coast Water
- cooling water pumps for Curtin University campus wide AC system

The remote Gold Coast Water site contained two ITT Flygt CT-3231, 2 Blade, 85kW, 1475 rpm pumps which were housed in a concrete silo-like structure. These pumps were located

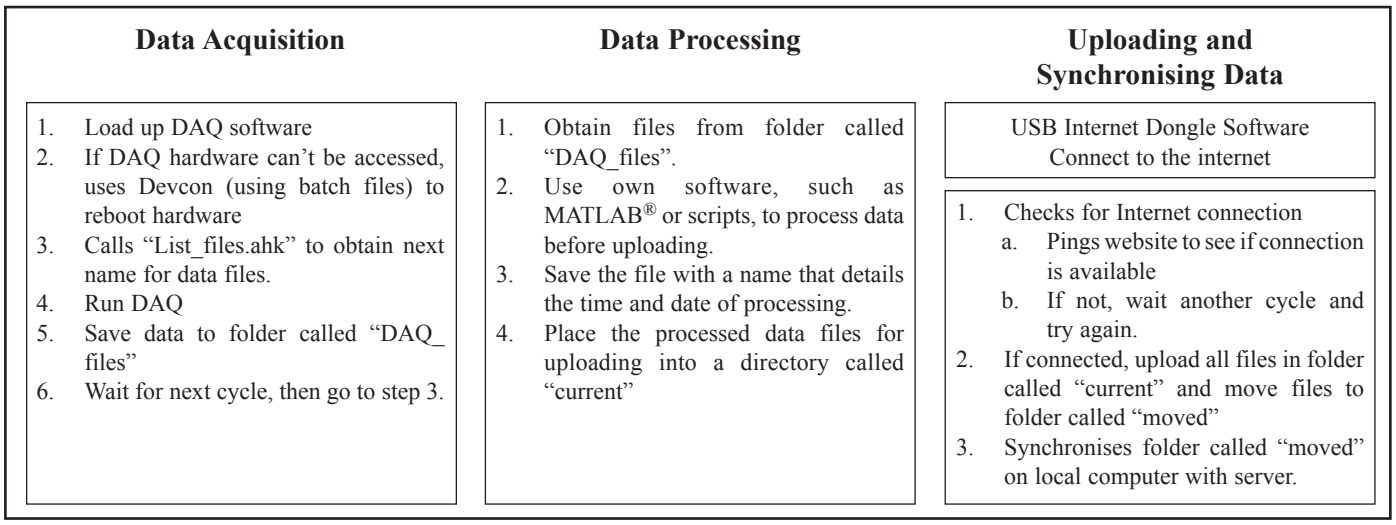

Figure 3. Overall process found in four main scripts

Table 4. Summary of RDAQ Sites

	Time interval left on-site	Time interval between uploads	Problems
Gold Coast Water	1 to 3 days	15 minutes	System freeze System reboot failed
Curtin University	20 hours	1 hour	Gaps in uploaded times

approximately 50 metres underground, while the computer, hard drive, and USB dongle were located in an aluminium cabinet above ground. The system was left at the site multiple times at different intervals, ranging from overnight to three days. At this location, the SignalCalc ACE system was used for data acquisition. The second location contained KSB E150-40 75kW, 1475rpm which was housed in a concrete building. Readings were taken from this location three times, each for an interval of approximately 20 hours. At this location, the National Instrument DAQ system was utilised. The results of these two locations are found in Table 4.

At both sites, data was transferred to the remote server with success. However, different problems arose within the interval during which the system was left onsite. At the Gold Coast Water site, the Windows Operating System would at times freeze after about a day of usage. To overcome this problem, remote access to the computer using LogMeIn and TeamViewer was used to check if this phenomenon occurred and to reboot the system if needed.

At the Curtin University site, data was acquired for the full 20 hours that it was left using the National Instruments' cDAQ system. However, data was not consistently being placed onto the server. Gaps in uploaded times occurred between 3am and 8am, which is assumed is due to the server shutting down during this time. As a result, after 8am, the files on the computer and the files on the server were synchronised, which caused a large amount of files to be placed onto the server in one cycle. Although not a problem to the system, it may cause problems when the server is inaccessible for long periods of time and the data uploaded is either large in size, or multiple files with a short interval between performing the data acquisition. This could be a problem since this might interfere or impinge on the resources that are allocated to another task while the system is attempting to compensate by synchronising data files.

CONCLUSIONS

There is a large need for a remote data acquisition system in industry as well as in the research community. However, most RDAQs that are available are very expensive, with the need of pre-existing networking structure to handle the data transfer either wirelessly or through a local area network. As a result, these systems are avoided due to cost and available infrastructure. This paper presents a cost efficient method of providing a RDAQ utilising systems a noise/vibration analyst would already have at hand. It is believed that this cost efficient method of providing an RDAQ would be beneficial to both industry and academic settings where the user constantly needs data but cannot constantly be present at the site.

ACKNOWLEDGEMENTS

This research was conducted within the CRC for Infrastructure and Engineering Asset Management, established and supported under the Australian Government's Cooperative Research Centres Programme. The authors would like to thank Gold Coast Water and Curtin University for facilitating the vibration measurements on the industrial water pumps used for the validation measurements.