Journal of Achievements in Materials and Manufacturing Engineering
VOLUME 21 ISSUE 2 April 2007

Development of real-time data filtering for SCADA system

L. Wiliem [a,*], D.Hargreaves [a], R.F. Stapelberg [b], P.K.D.V. Yarlagadda [a]
[a] School of Engineering Systems, Queensland University of Technology, 2 George Street, Brisbane, Qld 4001, Australia
[b] Education and Training, CIEAM (Cooperative Research Centre for Integrated Engineering Asset management), Brisbane, QLD 4001, Australia
* Corresponding author: E-mail address: l.wiliem@qut.edu.au

Received 13.10.2006; accepted in revised form 15.11.2006

Industrial management and organisation

ABSTRACT

Purpose: Purpose to develop a suitable algorithm to filter data from the SCADA system.
Design/methodology/approach: A real-time filtering method for SCADA system is developed by capturing the occurrence of data change in SCADA data, which is followed by recording several data before this data change occurs. Then, the algorithm is modeled and developed and in the final step an experiment to verify the algorithm is conducted. Finally, the result from the experiment is analysed to check the effectiveness of the algorithm.
Findings: As a result, SCADA data analysis will be easier to conduct since only essential information is left. In fact, in comparison to the the entire data collection, only around 8-22 % of data is changed.
Research limitations/implications: By utilizing this algorithm, data analysis will be easier to conduct since only the essential information as a starting point of analyses is left. However, this paper only describes the reasons and steps of data filtering algorithm development, how the algorithm works and the result after it is implemented to analyse data from the SCADA system. Further analyses to the data filtering results haven't been done yet. The next step will be to analyse the results in order to establish the root cause of why the data is changing.
Originality/value: It has been noted in many research papers that the SCADA system is able to increase the efficiency of the monitoring itself. However, the SCADA system creates a huge amount of data which is difficult to analyse. This paper proposes a real-time data filtering for the SCADA system. The philosophy that is applied in this algorithm is only to "catch" the occurrence of data change in SCADA data, which is followed by recording several data before this data changes. As a result, SCADA data analysis will be easier to be conducted since only essential information is left. In fact compared to the entire data collection, only around 8-22 % of data is changed. Therefore, this method is highly suitable for the SCADA system.
Keywords: Productivity and performance management; Real-time; SCADA; LabVIEW; Data filterin

1. Introduction

Most organizations use computer technology in their business activities [1, 2]. Computer technology has penetrated every facet of business activities, from simple calculators to complete Enterprise Resource Planning (ERP) Systems [3]. Efficient, fast and effective data acquisition is the hallmark of computer technology [4].

However, plaguing this computer technology are many problems relating to data management. The speed of collection of data far exceeds the speed with which organisations analyse it and draw useful information [5, 6]. This problem is exacerbated with on-line data acquisition systems [7], specifically data obtained from Supervisory Control and Data Acquisition (SCADA) systems [8].

Condition monitoring of machines in engineering play a very important role in their use, as it provides information on the operational characteristics, health and maintenance requirements of the machine [9, 10]. Unscheduled breakdowns of mechanical equipment in process plants can therefore be reduced [11, 12]. Condition monitoring is continually undergoing rapid development with new techniques in the areas of tribology, thermography, acoustics and vibration, in addition to basic monitoring of operational characteristics such as temperature, pressure, flow, etc [13].

Advances in computer technology have had a significant impact on this process [14-16]. Advancement in sensor technologies, automated controls and also data telemetry has made possible new and innovative methods in condition monitoring. Rapid growth in networking systems especially through the Internet have overcome the barriers of distance, allowing, real-time data transfer to occur easily from different locations [17]. With real time data transfer, current equipment condition can be measured immediately and any operational abnormality easily identified.

A specialized system for real-time condition monitoring, namely Supervisory Control and Data Acquisition (SCADA), has superceded Distributed Control Systems (DCS) in industrial process plant [18]. This system is basically the state of the art of real-time monitoring systems. As its name suggests, the duties of this system are supervisory control and data acquisition.

However, without adequate techniques, analysing data obtained from a SCADA system is complicated and difficult since the data is influenced by many factors. There must be a method or technique to filter the data, so that only essential data is left while the other is filtered.

The purpose of this study is to develop a suitable algorithm to filter the data from the SCADA system. This paper is organized as follows. Section 2 gives an overview of algorithm development. Experiment overview is presented in section 3. Section 4 explains experiment result discussion. The conclusion is discussed in Section 5.

2. Algorithm development

In general, when conducting measurements, the conditions or states of the measurement objects are most probably changed if the measured parameters are changed. Therefore, only events where the parameters have changed are essential to "catch" while the rest of data will be used for further analysis.

Basically, there are three steps in the process in the algorithm as illustrated in Figure 1.

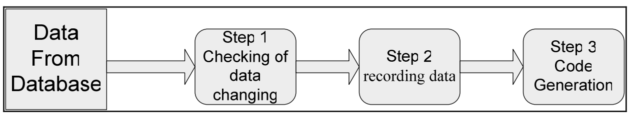

Fig. 1. Steps of Algorithm

Explanations for each of these steps are:
1. Step 1: checking changes in data from each parameter.
2. Step 2: recording the last data from each of the parameters if there is at least one changing parameter.
3. Step 3: generating a code to notify which of these parameters is changing.

Code that is used in this algorithm is a simple code where each parameter is represented by one word. If there is no change in the parameter then it will be represented by the letter "O", while other letters such as "A", "B", "C", etc. will be used if the parameter is changed. Furthermore, an index when at least one of these parameters is changed is then recorded for further analysis .For example the experiment in this paper is using data from motor measurement. There are 10 different parameters measured for this motor. Therefore, the code will be:
AOOOOOOOOO which means the first parameter is changed
OBCOOFOOOO which means the second, third and sixth parameters are changed, etc.

3. Experiment overview

Once the algorithm has been devised, the next step in the process is to test the algorithm. In this experiment the algorithm is tested to filter data from a motor. This motor is owned by Water Company in Australia and it is used to drive the centrifugal water pump. There are 10 different measurements to monitor the condition of the motor, which are:
1. Winding Blue Temp
2. Drive Bearing Temp
3. Drive Bearing Vibration
4. Heat Exchanger Air In Temp
5. Heat Exchanger Air Out Temp
6. Housing Vibration
7. Non Drive Bearing Temp
8. Non Drive Bearing Vibration
9. Winding Red Temp
10. Winding Yellow Temp

In this experiment, the data is manipulated in order to mimic the real condition of the SCADA system. Matlab 7.0.4 are used to do this initial manipulation. After the data is ready, a simulation of data recording process for the SCADA system is modelled (this model is written at the programming language platform LabVIEW 7.1) in which the filtering algorithm is applied. This stage of the experiment is illustrated in Figure 2 below.

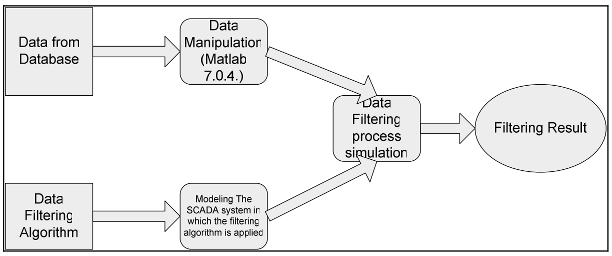

Fig. 2. The Stage of Experiment

In this simulation, ten different parameters are filtered at the same time. If there is a change in at least one of these parameters, the last ten data from each of them are then recorded (the number of last data that are recorded can be set flexibly). Furthermore, the code to notify which of these parameters are changing is then generated.

4. Experiment result discussion

Three different files are obtained from the experiments. One file is storing the last ten data records from each parameter before the data change; these include time, date and code. One file is storing only the time, date and code, and the other file is storing the count of data change from each parameter. Several Figures below show each of these files.

From Figure 3, it is clear that the algorithm is working properly and this information can be used for further data analyses. Figure 4 shows that the events of data change in every parameter happen only around 8-22 % of the whole data and these events are very useful as a starting point for further analyses.

Data Filtering Algorithm Experiment
Queensland University of Technology (Brisbane, Australia)
PhD Student: Leonard Wiliem
Date: 10/05/2006Time: 11:17 AM

No.	Date	Time	Note	Index
1	14/09/03	02:06:50	abcdefghij	1
2	14/09/03	00:00:20	oooooogooo	2
3	14/09/03	00:00:30	oooooogooo	3
4	14/09/03	00:00:40	ooodoooooo	4
5	14/09/03	00:00:50	oooooogooj	5
6	14/09/03	00:01:00	ooooeooooo	6
7	14/09/03	00:01:20	ooooeogooj	8
8	14/09/03	00:01:30	ooodoooooo	9
9	14/09/03	00:01:40	oooooogooj	10
10	14/09/03	00:01:50	ooodoogoio	11
11	14/09/03	00:02:00	ooodoooooo	12
12	14/09/03	00:02:10	oooooogoio	13
13	14/09/03	00:02:20	oooooooooj	14
14	14/09/03	00:02:30	oooooogooo	15
15	14/09/03	00:02:40	abodoooooo	16
16	14/09/03	00:02:50	ooodeooooo	17
17	14/09/03	00:03:00	ooooooooij	18
18	14/09/03	00:03:10	abooeoooio	19
19	14/09/03	00:03:20	oooooogooo	20
20	14/09/03	00:03:30	ooooeooooo	21
21	14/09/03	00:03:40	ooodoooooj	22
22	14/09/03	00:04:00	oooooooooj	24
23	14/09/03	00:04:10	ooodoooooo	25
24	14/09/03	00:04:20	ooooeooooj	26
25	14/09/03	00:04:30	ooodeooooj	27
26	14/09/03	00:04:40	oooooogooo	28
27	14/09/03	00:04:50	ooodeooooo	29
28	14/09/03	00:05:10	ooodeooooo	30
29	14/09/03	00:05:10	oooooogooo	31

Fig. 3. Record of code, index, date and time of when data change occur

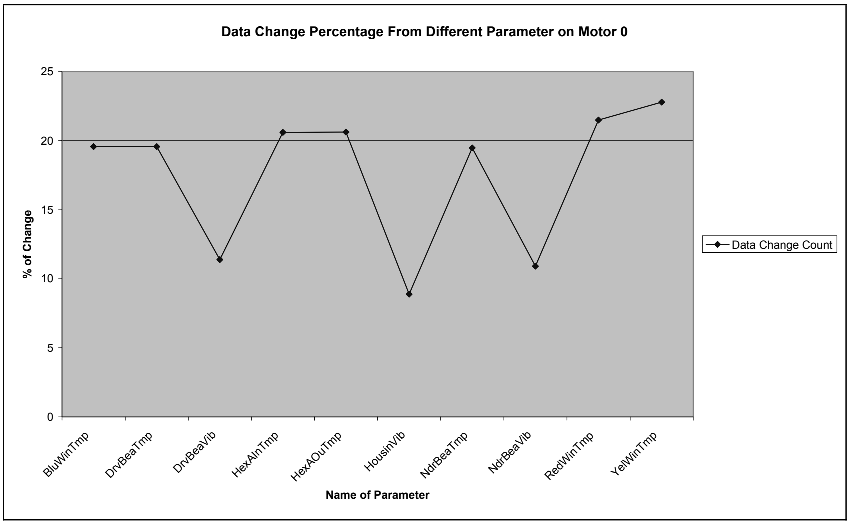

Fig. 4. Data change percentage from different parameter

5. Conclusions

In conclusion, the proposed algorithm is able to notify when the event of data change occurs and "capture" this essential information by recording a number of last data bits before this event occurs.

By utilizing this algorithm, data analysis will be easier to conduct since the information has been filtered. Only the essential information as a starting point of analyses is left. Therefore, this algorithm is really suitable to apply in SCADA system.

References

[1] C.-J. Tseng, H. Abdalla, A human-computer system for collaborative design (HCSCD), Journal of Materials Processing Technology Proceedings of the International Conference on Advances in Materials and Processing Technologies 2 (2004) 1964-1971.

[2] F. Nyhuis, Controlling and monitoring functions as supplement to MRPXX-systems, Journal of Materials Processing Technology 107 (2000) 439-449.

[3] J.K.C. Ng, W. H. Ip, Web-ERP: the new generation of enterprise resources planning, Journal of Materials Processing Technology IMCC2000 138 (2003) 590-593.

[4] M.C. Kang, J.S. Kim, J.H. Kim, A monitoring technique using a multi-sensor in high speed machining, Journal of Materials Processing Technology 113 (2001) 331-336.

[5] A.G. Taylor, Database development for dummies, Foster City, CA, IDG Books Worldwide, 2001.

[6] A. Matus-Castillejos, R. Jentzsch, A time series data management framework, presented at Information Technology: Coding and Computing, 2005, ITCC 2005.

[7] F. Cus, M. Milfelner, J. Balic, An intelligent system for monitoring and optimization of ball-end milling process, Journal of Materials Processing Technology Achievements in Mechanical & Materials Engineering 175 (2006) 90-97.

[8] L.Wiliem, P.K.D.V. Yarlagadda, D.Hargraves, L. Ma, R. F. Stapelberg, A Real-Time Lossless Compressing Algorithm for Real-Time Condition Monitoring System, COMADEM 2006, Lulea University of Technology, Sweden, 2006.

[9] T. McGrail, Condition monitoring-a user perspective, HV Measurements, Condition Monitoring and Associated Database Handling Strategies (Ref. No. 1998/448), IEE Colloquium on, 1998.

[10] B.K.N. Rao, Handbook of condition monitoring, 1st ed. Oxford, UK: Elsevier Advanced Technology, 1996.

[11] Pruftechnic Ltd, Can Condition Monitoring Improve Your Plant Performance, Worldpump, 2004, 40-47.

[12] A. Davies, Handbook of condition monitoring techniques and methodology, 1st ed. London: Chapman & Hall, 1998.

[13] C.P. Ephraim, Condition Monitoring In Automated Manufacture, IEE Colloquium, 6/1-6/4, 1997.

[14] R. Itschner, C. Pommerell, and M. Rutishauer, GLASS, Remote monitoring of embedded systems in power engineering, IEEE Internet Computing 2 (1998) 46-48.

[15] A. Laugier, N. Allahwerdi, J. Baudin, P. Gaffney, W. Grimson, T. Groth, and L. Schilders, Remote instrument telemaintenance, Computer Methods and Programs in Biomedicine 50 (1996), 187-194.

[16] F.B. Zhou, M.D. Duta, M.P. Henry, S. Baker, and C. Burton, Remote condition monitoring for railway point machine, Railroad Conference, 2002 ASME/IEEE Joint, 2002.

[17] L.Wiliem, P.K.D.V. Yarlagadda, S. Zhou, Development of Internet Based Real-Time Water Condition Monitoring System, COMADEM 2006, Lulea University of Technology, Sweden, 2006.

[18] S.N. Laboratories, History of SCADA, 2006, http://www.sandia.gov/scada/history.htm.