An ‘on-demand’ Data Communication Architecture for Supplying Multiple Applications from a Single Data Source: *An Industrial Application Case Study*

Ifeanyi Azolibe*, Euan W. McGookin*

**Aerospace Sciences Research Division, School of Engineering,*
University of Glasgow
University Avenue, Glasgow G12 8QQ, U.K.
i.Azolibe.1@research.gla.ac.uk; Ifeanyi.Azolibe@yahoo.com; Euan.McGookin@glasgow.ac.uk

Keywords: *test automation, data reuse architecture, aircraft fuel control system, MRO*

Abstract

A key aspect of automation is the manipulation of feedback sensor data for the automated control of particular process actuators. Often in practice this data can be reused for other applications, such as the live update of a graphical user interface, a fault detection application or a business intelligence process performance engine in real-time. In order for this data to be reused effectively, appropriate data communication architecture must be utilised to provide such functionality. This architecture must accommodate the dependencies of the system and sustain the required data transmission speed to ensure stability and data integrity. Such an architecture is presented in this paper, which shows how the data needs of multiple applications are satisfied from a single source of data. It shows how the flexibility of this architecture enables the integration of additional data sources as the data dependencies grow. This research is based on the development of a fully integrated automation system for the test of fuel controls used on civil transport aircraft engines.

1 Introduction

The automation of technical processes has become an essential business consideration for many industries. Whether the aim is modernization, waste reduction, improved efficiency or increased throughput, automation presents opportunities to grow a business’s potential [1].

The reusability of process data for technical or non-technical applications is one such opportunity. For example to determine the health of systems through analysis of process data which contains insights about the operating conditions of different components. This could be the basis for a preventative data-driven maintenance strategy that involves fixing faults before they become failures that could cost substantial amounts of money [2].

It is essential to feedback well-conditioned data from the sensors when the process is being automated [3][4][5]. In order to measure and collect all that data, appropriate data acquisition systems (DAQs) are usually employed. Therefore, the specifications of the DAQs must meet the requirements for automating the process. Although this is extremely important, it is crucial that the data from existing DAQs is suitable to be reused for automation. In such cases modifications can be made to the data to ensure it is in an appropriate format or the automation systems can be designed to accommodate mismatches with the existing systems e.g. update rates.

The context of this research is the automation of the test of fuel control systems used on civil aircraft engines, where traditional DAQ (or t-DAQ as it is called) systems are used in the measurement of data for test procedures [6]. In this context the purpose of such systems is to acquire data from specific sensors, condition such data and make it available for retrieval by Remote Terminal Units (RTUs) through a digital communication interface. This data can be simultaneously displayed on a monitor and reused by multiple applications that serve an aspect of automation of the test [7].

In order to reuse test data for other applications besides the automation of the main activity—process control, each t-DAQ and its sensor network should be evaluated on their capability to meet the requirements of two types of data consumer applications: time critical and non-time critical. Then the management of the process of extraction and reuse of data should be implemented in a software centric environment.

The non-triviality of reusing data from t-DAQs for applications of automation arises from the limitations of the interfaces through which their data can be extracted; the rate at which the internal temporary memory is updated; the rate(s) of data extraction, and the format of data extracted. As a result, an architecture for a data network that reduces the effect of these constraints and ensures that such data is accessible and reusable by other applications: is critical to automate the control of processes that used to be controlled manually.

A time critical application in our context is one that needs to know about its dependent process at a minimum frequency of

50 Hz. On the other hand, a non-time critical application is an application that requires information about its dependent process at a maximum of 20 Hz.

A software centric architecture is presented in this paper that enables the reuse of data from one source and is scalable for the integration of additional data sources. The concept is based on having a live table that contains the process data collected from different DAQs. The data is merged with additional information like an instrument name tag, data type tag & a description. These additions make the data identifiable, extractable and reusable, based on an 'on-demand' data communication architecture [8]. The context of our discussion is based on the automation of the test of fuel control systems used on civil transport aircraft engines [9][8]. The focus of this paper involves the automated control of test conditions: the core application that reuses process data from different t-DAQs.

The requirements of applications that will reuse process data from different t-DAQ systems is reviewed Section 2. Then an outline of the automated process control application structure is presented Section 3, which leads to a review of the limitations of a raw numeric data, unidirectional broadcast, data reuse architecture alone; to reuse data from multiple t-DAQs: with results. In order to reduce these limitations, a "featurized" data subscription communication architecture is proposed in Section 4. Its implementation and evolution into an 'on-demand' data subscription communication architecture then treated in Section 5 with results and analysis of its impact. Finally a conclusion is given in Section 6.

2 Requirements of data dependent applications

Each application that reuses test data is designed to fulfill a tangible business benefit. Take a Fault Detection Isolation and Recovery (FDIR) application for example; it reuses process data from different aspects of the test systems network to detect faulty operation of systems; learning *signature patterns* and their eventual failure modes [2]. This function could then be used to automatically manage moderate faults or activate an alarm before a failure occurs. As a result it can help reduce the frequency of unplanned downtime given that it can establish the "ball park" of likely root causes. Thus leading to an increase in the availability of test systems for business operations.

1) The FDIR application: This application is required to function online for active prevention rather than as a corrective advisory tool. It uses test data from all t-DAQs in their respective test systems. Therefore, if the process data is first stored and retrieved for FDIR, this causes a delay which is unsuitable for our application. The goal is to have the FDIR application "advising" the Process Control Automation (PCA) application on strategies to manage faults and prevent failures from occurring [2][10][11].

2) The Process Control Automation (PCA) application: It sets different processes at specified test conditions, using process data from a t-DAQ. The business benefit of this application is one of improved test efficiency, through a reduced total test time which translates into extra capacity for more tests and reduced variable overhead costs, e.g. electricity consumption per test.

3) The Augmented Reality (AR) application: This uses real-time process data in a format that gives a virtualized representation (AR), of the product-under-test (PUT). For our application, this is a 'live schematic' of the PUT with real-time colour variations that depict whether the different subsystems of the PUT are functional and operating within designed/specified ranges.

The AR application reduces the mental burden that used to be placed on an operator to generate mental models of the health of the PUT using numeric process data displayed on a monitor. Such that subsystems with the PUT that are functioning okay have a dynamic colour of green (normal); about-to-fail show amber (warning) or have exceeded the safe limit red (imminent failure). The AR application is not time critical.

4) The Business Analytics (BA) application: This implements BA concepts to extract operational insights from available test process data. It serves managers' needs to know process quality, operating capacity, anticipate unplanned maintenance and systems health in real-time. The D-board application is not time critical.

5) The Data Entry & Recording Automation (DERA) application: Conventionally, operators read the screens of multiple DAQ systems displaying numeric values of the states of different test process media. They then type these into a processing application for onward logging into a business database. However, this application automates the entire process of data entry and recording by extracting test data from the t-DAQs and communicating these to the processing application which *parses* the data on to the business database for storage [12].

The DERA application has a requirement that all recorded data should be extracted simultaneously, when set test conditions are in *steady state* and the PUT has achieved the test objective [12]. Table I shows a summary of the requirements of all the applications that would reuse test data from the different t-DAQs as part of the automation of the test process.

Applications	Data rate requirements	Data size	Data variety
FDIR	2 Hz	52	6
PCA	< 50 Hz	3	2
AR	1 Hz	20	3
BA	0.33 Hz	32	4
DERA	1 Hz	20	3

Table I: Requirements of data reuse client applications

This table shows crucial functional aspects such as the minimum frequency at which an application needs its data; the volume of test data each needs to execute its function correctly and the variety/types of data i.e. whether numeric, textual or logical. It can be clearly seen that the frequencies of data speed required range from less than 1 Hz, up to 50 Hz for the PCA application which has the least data volume and variety.

It is obvious that physically connecting wires to the analog input terminals of any t-DAQ, in order to connect multiple DAQ devices for each application is not feasible. The risk of loose connections and electrical loading issues are high and will result in inaccurate measurements of test processes; excluding other non-functional costs. Thus software centric data management is seen as the practical way to extract the data from the t-DAQs through their data communication interfaces, for reuse by the PCA, the DERA, FDIR, AR and BA applications.

It is obvious that in order to supply data to all of these applications, such data management architecture must be capable to transmit data at the rate required by the application with the highest data frequency requirement i.e. the PCA application, at ≥50 Hz. So the first design that has been tested is engineered to give a quick solution. It was based on a unidirectional data broadcast communication architecture. This design concept is shown in Fig. 1.

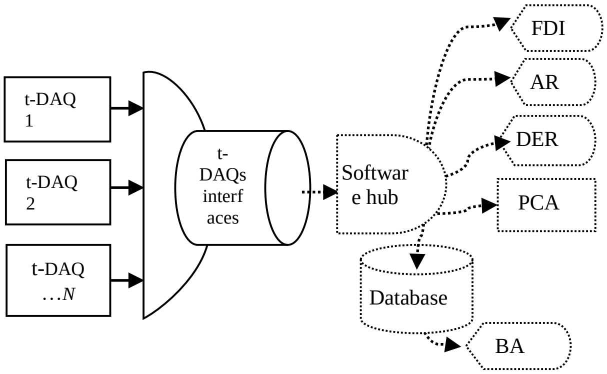

Fig. 1: Architecture of unidirectional broadcast communication of test data to automation applications

It should be noted that the number of process data which has to be retrieved from the t-DAQs for the PCA, FDIR, AR and DERA applications is 20; then 32 sets of data for the BA and FDIR applications are digital signals that are not integrated with an extant DAQ system.

The challenge with reusing data from the t-DAQs is, it must be supplied to the time critical PCA applications. Therefore an attempt has to be made to prove that this data communication architecture can support the automation of the test process and enable other applications that reuse the test data to meet their goals.

For completeness, the network of sensors used during the test of fuel control systems includes pressure transmitters, temperature sensors, fuel flow meters, turbine tachometers, Linear Variable Differential Transformers (LVDT), resolvers and relays.

3 The limitations of a unidirectional data broadcast architecture

To discuss the limitations of this data management architecture appropriately, the structure of the PCA applications that deliver the primary benefits of automation of the test process is presented first.

A PCA controller have the structure shown in Fig. 2, which highlights how critical the data from a t-DAQ is for each PCA to correctly set its process to the commanded set condition, using process feedback data.

This t-DAQ system is a Daytronic System10 product. It has a processor that is capable of updating its ‘holding’ memory with data scanned from analog input channels at a frequency of 3 kHz. This is the memory from which data is supplied to RTUs using an RS232 digital communication protocol with BAUD settings of 5-7-2-0 [7]. This implies the number of actively installed channels determines how frequently memory is updated i.e. more channels less frequently and fewer channels more frequently.

The number of installed active channels is 100 analog inputs. Of this number, 20 channels are required for the test procedure being discussed. And the data from 3 out of the 20 channels are used by the PCA application.

Given the BAUD setting of the t-DAQ system’s serial communication link, data can only be extracted from its holding memory by a dependent RTU application at a rate of 33.33 Hz with overheads [7]. Although the BAUD setting could be changed to increase the bit rate of transmission, there are other business systems that are configured to work with this setting, so it cannot be modified. These mean that dependent applications must accommodate this inherent delay and could only be served through a software interface. Reason being only one application can access the data at a time to avoid data jamming which undermines data integrity.

Theoretically, the inherent time delay incurred by using the t-DAQ system means that the PCA application with a data frequency requirement of at least 50 Hz cannot be supported. The evidence of this is shown in Fig. 3. Fig. 3 shows the response of a PCA process, PCA process #3. Besides the process response being marginally stable, it overshot the set point command by up to 400%. This affirms that the frequency of data update was slower than the PCA could accommodate. Therefore the benefits of automation cannot be realized with such performance.

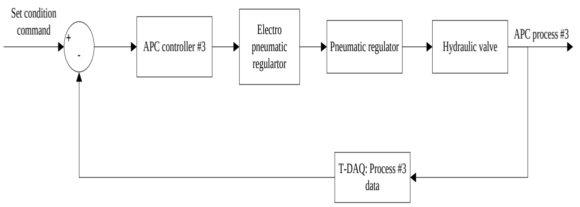

Fig. 2: Control architecture of a PCA application

An option to mitigate this limitation of the t-DAQ is to buy an additional serial card and configure it for a higher bit rate. The other option is to introduce a modern DAQ device integrated with three new sensors for the processes of the APC. This was the selected option, given that replacement of the t-DAQs will be expensive.

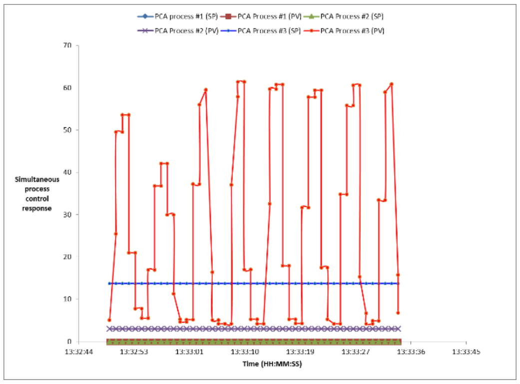

Fig. 3: The impact of data management architecture using a broadcast data communication model on a PCA application

A review of how data from these two sources could be managed led to the redesign of the data reuse architecture. It is based on a "featurized" live table of test process data that is accessible by each application of automation via a dedicated data channel as a form of 'subscription'.

4 On-demand data subscription architecture

Practical architectures of data reuse on a systems level found in the literature show little work is being done in this area [13][14]. Reviewed publications have involved micro implementations that focused on specific technical outcomes like efficient retrieval methods to extract data from memory or the reuse of data to reduce the cycle time of a technical mechanism. For instance, one case study focused on data reuse architecture for MPEG video coding [15][16].

The majority of published works in the literature focus on data reuse algorithms or mechanisms for technical applications, which are narrow in scope and focus on one dimensional data reuse objectives.

The challenge here is the management of the dataset; make it easily retrievable and efficiently accessible to every application that needs it. None of these can be achieved with just the raw process data alone when based on a broadcast communication architecture. More so, this dataset must be live and have no copies.

It has been established that additional information about each test data is essential to make it accessible, retrievable and reusable by a dependent application.

In order to achieve this, metadata is generated from the channel configuration of each t-DAQ system for each test data. These metadata are the data name, the data's channel number, a numeric tag, its value (the actual data) and the data's type (e.g. short or long integer). Metadata in a cluster are padded to their data and these clusters of "featurized" test data compose a live table (an array of sorts), upon which the data reuse architecture is based. Each application then uses a dedicated data lane to pull their data from the live table: when it is needed.

Although the disciplines of software engineering and computer programming propose a conceptual framework: of a single, live data source. This is called the singleton design pattern and it does not apply holistically to the context of this application [17]. The singleton design enforces the existence of a single copy of data in active memory in the lifetime of an application. But it does not deal with how to manage the constraints of the data's extraction from different physical sources, reusing it for time critical and non-time critical applications. Some of these functions affect the performance of time critical systems in the physical domain.

Based on our proposed design shown in Fig. 4, a substantial number of applications can reuse data from a single source. However new data sources had to be used to reduce the impact of data reuse on the PCA application. Therefore the proposed architecture is adapted for the new data source for the PCA applications, by duplicating the data subscription communication architecture to serve it. This duplicate inherits some metadata of the first architecture and was daisy-chained to it using an application which required all test data for its purposes: the FDIR application.

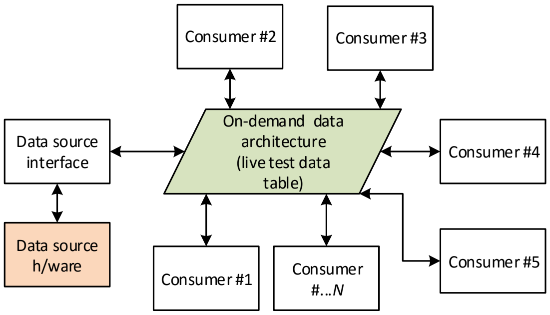

Fig. 4: Data management architecture of "featurized" test data based on 'on-demand' data subscription architecture

Fig. 5 shows the daisy-chained architecture that resulted. Its realization means that although the three PCA process data are duplicated, each dataset is visible to all consumer applications. Thereby enabling all of the dependent applications to fulfill their requirements.

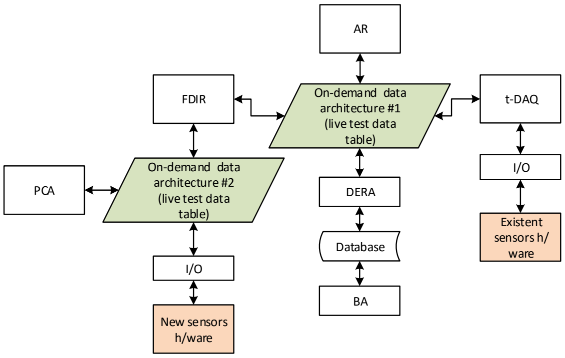

Fig. 5: Adapted data reuse architecture that enables the integration of new data sources

The PCA alone is the time-critical application going by its data requirements. By design it utilizes data from the duplicate architecture, whilst all of the other applications reuse data managed by the live table (#1) connected to a t-DAQ (Fig. 4).

All the other applications are local to the test process operation except for the BA application. Therefore data communication models such as the server client model used for remote application communications and based on a protocol like TCP/IP are deemed an unnecessary complexity for this application. For example there is the commercial product, LabVIEW™ Current Value Table which is similar to our proposed concept but implements a server client communication model [19].

For our application designated communication lanes are established between each consumer application and its live data table (Fig. 4). Then each consumer application 'pulls' the data it needs when it needs it. Thereby decoupling the dependency of each consumer application. This design does not address the fact that the number of consumer applications could grow in the future. Given that such a scenario will require new data lanes to be programmed for every new consumer application. Thus the data lanes are redirected to proxy global copies of each live table. These copies are updated at a rate of 1 kHz from the live tables. As such a new consumer application only requires the name of this global copy to retrieve test data, then search for its own dataset and execute its function to meet its requirement(s).

Nevertheless, a fundamental limitation of our proposed architecture is that the speed of retrieval of data elements by applications that will reuse them is directly proportional to the size of data it contains and the volume of data that each application requires [4] [6]. In a broader consideration, this is a challenge for all applications concerned with multidimensional data reuse.

$$\text{Data retrieval interval (s)} = (1 \div \text{retrieval frequency of each data element}) \times \text{data volume}$$

5 Results

One PCA controller performance will be used as a benchmark to evaluate the impact of using appropriate data reuse architecture. Guided by the result of Fig. 3, the goal is to increase the frequency at which data is available to the PCA application with a poor performance. Before a new data source was implemented the data collected from the t-DAQ system was reduced to three for the PCAs alone. The results in Fig. 6 show an improved performance of process control when the volume of data is reduced. It also supports our observation that the performance of the PCA is affected by inherent limitations of data extraction from the t-DAQ.

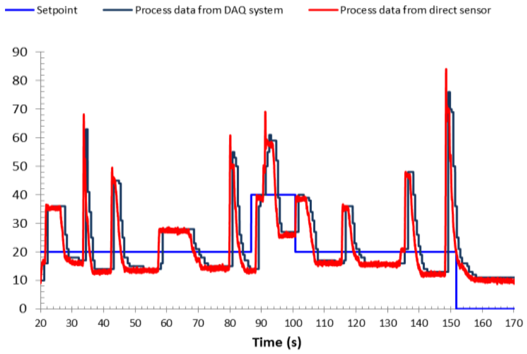

Fig. 6: Effect of a reduction in volume of data on response of most affected process

Upon installation of the additional sensors and implementation of the daisy-chained architecture, Fig. 7 shows a better performance of the PCA application. In particular, the minimization of the oscillatory response of the most affected PCA process. There is an overshoot of about 20%, a steady state error of 8% and an impulse response to a decrement in set point (which can then be dealt with by further tuning of the PCA controller).

The additional sensors have been integrated into a National Instruments analogue input module PXIe-6361. It provided data acquisition rates of up to 666 kHz per PCA application [20]. The results demonstrate that the sharing of data between applications that need it at a frequency of say 2 Hz and those that need it at a frequency of >50 Hz, is a design challenge that can be solved in different ways. This is one of the drivers

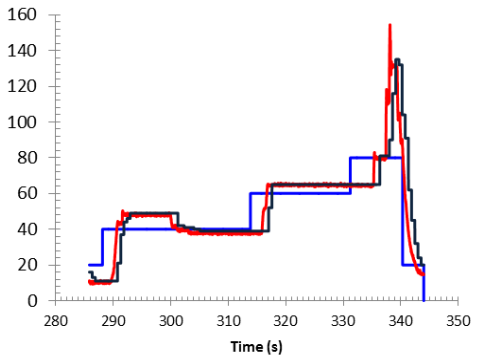

Fig. 7: The impact of daisy-chained architecture that integrates new sources of data

behind modern data acquisition systems being software oriented. In our case, one data source is for time critical applications and the other for non-time critical applications. And this could differ for other data reuse applications in the field.

Furthermore, both data sources are observed to give similar representations of the state of the process under control (Fig. 8). Except for the 1.8 seconds lag in data extracted from the t-DAQ.

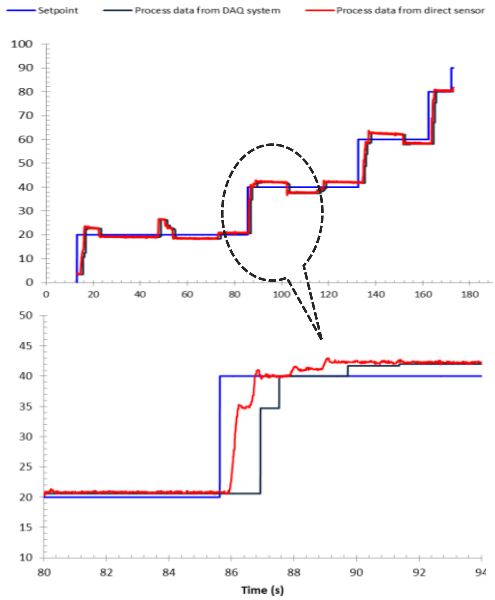

Fig. 8: Comparison of process response representation from multiple data sources

Though these data may drift apart owing to the natural degradation of sensor hardware, hardware redundancy concepts could be introduced in the FDIR application to compensate for this automatically. It does so through the reconfiguration of certain PCA parameters.

6 Conclusion

Although a single data source could serve multiple consumer applications, it would be impractical if these applications are a mix of time critical and non-time critical applications. This conclusion is based on our research experience. Especially if an extant DAQ system is only accessible via a single interface.
Using a software oriented architecture, multiple applications could reuse the same test data, but at the expense of achieving a pivotal objective i.e. of automation of process control. This resulted in the redesign of the data reuse architecture, to accommodate additional data sources for time critical applications alongside non-time critical applications.

This work illustrates that stretching the utility of legacy systems could hurt the benefit of applications which can deliver business benefit from modern technological concepts such as automation. Therefore to use one data source to serve multiple applications, a software centric design is recommended, using a capable data acquisition system for time critical applications. Nevertheless the time requirements of all data dependent applications be understood first.

Acknowledgement

This project began as a UK Knowledge Transfer Partnership funded by the then Technology Strategy Board (TSB) U.K; January 2011 – 2013. The authors are grateful to Woodward Inc., Prestwick, UK, a subsidiary of Woodward Inc. for sponsoring and hosting the project. Also the contributions of the Test Engineering team of Woodward Inc., USA are duly acknowledged.

References

[1] Frost & Sullivan (2009). *Global Commercial Engine MRO Market Assessment (CF34, CFM56 & V2500), M3B8-22* [Online]. Frost & Sullivan. April 2009. Available from: http://www.frost.com. (Accessed: 23/04/2011)
[2] Isermann, R., 1997. Supervision, Fault-Detection and Fault-Diagnosis Methods - An Introduction. *Control Engineering Practice*, 5(5), pp.639–652.
[3] Thompson, S., Dutton, K.; Barraclough, B. (1996) *The Art of CONTROL ENGINEERING* Pearson Education. Published by Addison Wesley, 20 Sept. 2010
[4] Bolton, W. (2011). Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering. 5th Ed., Essex, Pearson.
[5] Franklin, G.F., Powell, D.J. and Emami-Naeini A. (1991). Feedback Control of Dynamic Systems. 2nd Edition. U.S.A., Addison-Wesley.
[6] Azolibe, I., (2012). *Towards Autonomous Test of Aircraft Engine Fuel Control Systems*. Woodward Inc. 2012. Internal Report.
[7] Daytronic (2006*), System10 User Manual*
[8] IBM (2005). *IMS on demand service-oriented architecture tools and solutions.* Available from: http://www.ibm.com/developerworks/data/library/techarticle/dm-0507klein/. (Accessed: 20/08/2014)
[9] Woodward Inc. (2011). *The V2500 FMU test.* Internal Report.
[10] Huynh, T.K. et al., 2010. *Value of condition monitoring information for maintenance decision-making.* Reliability and Maintainability Symposium (RAMS), 2010 Proceedings – Annual

[11] Black, S.E. et al., 2004. *Diagnostic/prognostic Modeling and reconfigurable control*. IEEE AUTOTESCON 2004, Sept 2004.

[12] Woodward Inc. (2011). *Requirements & Specifications for automation of the test of fuel controls—PHASE I*. Revision A. Internal Report.

[13] Wagh, M.; Vasudevan, A.(2008) White paper: *Merits of Data Reuse Hints*. [Online] Intel 2008. Available from: http://www.intel.co.uk/content/dam/www/public/us/en/documents/white-papers/data-reuse-hints-white-paper.pdf (Accessed: 13/04/2012)

[14] Tuan J., Chang T., Jen C. *On the Data Reuse and Memory Bandwidth Analysis for Full-Search Block-Matching VLSI Architecture*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 12, No. 1, January 2002

[15] Lai Y., Chen L. *A high data-reuse architecture with double-slice processing for full-search block-matching algorithm.* IEEE International Symposium on Circuits And Systems (ISCAS) 2003. Pages II-716 – II-719 Vol. 2, 28 May 2003

[16] Moshnyaga, V.G.; Masunaga, K.; Kajiwara, N. *A data reusing architecture for MPEG video coding*. IEEE International Symposium on Circuits And Systems (ISCAS) 2004. Pages III-797-800 Vol. , 26 May 2004

[17] Townsend M., Microsoft Corporation (2002), *Exploring the Singleton Design Pattern* [Online]. Microsoft Developer Network. Available from: http://msdn.microsoft.com/en-us/library/ee817670.aspx (Accessed: 10/01/2012)

[18] Sybase (2010), *White paper: Data Governance, Data Architecture and Metadata Essentials* [Online]. Sybase—An SAP Company. Available from: http://www.sybase.co.uk/files/White_Papers/Sybase_Data_architecture_and_data_governance_WP.pdf (Accessed: 12/06/2014)

[19] NI (2014), *Current Value Table (CVT) Reference Library* [Online]. National Instruments. Available from: http://www.ni.com/example/30326/en/ (Accessed: 14/05/ 2014)

[20] National Instruments (2012). *NI X Series Multifunction Data Acquisition*. [Online]. NI. Available from: http://sine.ni.com/ds/app/doc/p/id/ds-151/lang/en (Accessed: 22/11/2013)