Journal of Machine Engineering, Vol. 13, No. 1, 2013

agent based design,
process monitoring,
data interoperability

Jeff MORGAN[1]
Eoin O'DRISCOLL[1]
Garret E. O'DONNELL[1]

DATA INTEROPERABILITY FOR RECONFIGURABLE MANUFACTURING PROCESS MONITORING SYSTEMS

The ability for manufacturing organisation to be responsive to changing market conditions is imperative for sustainable market competitiveness, and enabling self preservation. Responsive behaviour requires the incorporation of not only robust sustainable manufacturing process chains, but the functional capacity to facilitate reconfiguration and adaption. Manufacturing process chains require process monitoring and condition based monitoring systems to achieve high accuracy manufacturing systems, sustainable production capability, and resource efficiency. The focus of this research is aimed at developing a novel data interoperability architecture to enable the creation of reconfigurable process monitoring systems. To achieve reconfigurable process monitoring capability an information communication paradigm known as agent based design is reviewed, and an emerging agent based standard known as MTConnect is also reviewed. Through this research a novel data interoperability system is developed and defined to enable dynamic data acquisition for reconfigurable process monitoring. A case study is carried out to demonstrate the validity of the architecture to achieve multi data stream unification. This case study demonstrates the ability to actively condition monitor a HURCO VM2 three axis CNC machine dynamically, through the measurement of machine energy requirements and vibration, which are examples of process variables associated with condition based machine monitoring.

1. INTRODUCTION

Since 1976 machining process monitoring has been an interesting topic for engineers trying to understand machine actions and subsequent reactions [1]. Since then manufacturing processes and interconnecting process chains have become more complex through the exponential rise in the advancement of industrial technology, and the need for Reconfigurable Manufacturing Systems (RMS) that provide manufacturing responsiveness to changing market conditions [2]. New challenges become present with the introduction of these changes, as the performance of these complex manufacturing structures is dependent on their ability to be adaptive within internal and external environments [3]. Subsequently this diversity and requirement for flexible application is prevalent in process

[1] Mechanical and Manufacturing Engineering Department, Trinity College Dublin, Dublin 2, Ireland

monitoring, as the ability to measure reactive elements may remain the same, but the ability to interpret these values effectively for a changing process, may not. Process monitoring systems must become adaptable to their new process, as manufacturing systems and process become more flexible and adapt to their new environments. Process monitoring systems must adopt similar practises to the machines that are to be monitored, meaning process monitoring systems must become reconfigurable to allow for adjustment to new process components and operations. Reconfiguration requires features to allow for current system changes and must also allow for the incorporation of new elements from new components that have been included in the process, ultimately ensuring interoperability within the system. Process monitoring reconfiguration not only facilitates RMS but all standard manufacturing systems, as functional reconfiguration enables the adaption of process monitoring systems to different manufacturing systems, while enabling the flexibility required in RMS to meet a changing process. This work aims at outlining a novel approach to achieving data interoperability for reconfigurable process monitoring systems. An introduction to process monitoring and process monitoring systems is provided, followed by a review of a developing paradigm known as agent based design that is aimed at handling complex system interactions. An engineering standard known as MTConnect is also reviewed as it utilises agent based ideologies in the aim of achieving unified manufacturing process monitoring connectivity. The resultant research concludes with the introduction of a novel architecture for data interoperability for an agent based process monitoring system, and a case study that provides a real life representation of the developed architecture in a machining environment.

1.1. PROCESS MONITORING

Predominantly process monitoring incorporates indirect measurement, which utilises auxiliary measurements from other process variables to correlate reactive effects to scaled measurement parameter deviation. A recent CIRP keynote paper Teti et al [1] provided a comprehensive study on machine process monitoring, illustrating core fundamentals and state-of-the-art developments. Through this work process monitoring is able to be characterised into the following steps; (a) Measurement, physical hardware, e.g. sensors, for measuring the physical process parameter, (b) Acquisition, interconnecting hardware and software elements for providing high speed data acquisition from the sensor to a computational device, (c) Filtering, mathematical manipulation of data for specific process feature extraction, (d) Analysis, methods, techniques and algorithms for variable correlation of required process attributes, (e) Decision Support, subsequent methods, techniques and algorithms appertaining to identifying the required corresponding process action from analysed results, (f) Closed-loop control, hardware and software elements associated with facilitating corrective action from decision support functions. Each step along this sequential path has received varied academic research over the past decade, and will continue to receive further r esearch as it has been stated that the future enhancement of machining systems and their operation performance will vitally depend upon the development and implementation of innovative sensor monitoring systems [1]. New data manipulation algorithms and paradigms have been a topic of extensive research to show

how data can be filtered and used to identify Key Process Indicators (KPI's) for decision making support [4],[5],[6]. Additionally ongoing and recent technology advancements have and will continue to improve the availability of process data, and capacity to be fused together [7-10]. This continuous industrial push for machine optimisation is highlighting the need for the continuous research into the various process monitoring steps, but is also identifying the need for more unified process monitoring solutions. These unified solutions do not just consist of one previously defined process monitoring step, but all the combined steps to create adaptable sustainable process monitoring systems. The realisation of such systems requires the integration of hardware and software, incorporation of multiple hardware/software item collaboration, each requiring their own local operational structure and global interaction communication methods and goal seeking ideologies. Throughout the past 20 years a conceptual paradigm has been emerging to handling this complexity of providing sustainable operation and implementation for distributed multi functional systems. This paradigm consists of interactive entities known as agents that form networks to complete localised goals and network wide goals.

1.2. AGENT BASED DESIGN

Agent based design is a paradigm of information communication technology [11]. Agents aim to address autonomy and complexity through adaptive capabilities allowing agents to be resilient to changes and disruptions, exhibit intelligence and are distributed in nature. Agents are individual problem solvers with some capacity of sensing and acting upon their environment, for deciding their own course of action, as well as communicating with other agents. Depending on the problem and available resources/technology agents can apply various faculties of problem solving, including searching, reasoning, planning, and learning [11]. Agent based software and the concept of multi agent systems has been in ongoing development since the 1980's [12]. One of the most important concepts around agent based design is the capability to interact with other agents in order to satisfy design and configured objectives. These interactions allow agents to separate themselves from passive programming entities, allowing agents to be seen as semi-autonomous entities. A passive programming entity has a state and can perform operations, while an agent has a state and can perform operations, but this state and operation can be influenced by interactions via communication to multiple agents. A Multi Agent System (MAS) is formed by a network of computational agents that interact and typically communicate with one another. The decisions and actions of various agents in MAS's are not necessarily influenced by interaction. However a MAS can occasionally solve problems that are beyond the limits of the competence of the individual agents. This interaction between agents requires coordination between agents with a regulated flow of information between the agent and the surrounding environment. This requires a defined direct or indirect communication medium, or language. MAS design and development has been well documented over the past four years. In 2009 Leitao [13] carried out a state-of-the-art survey reviewing agent based design, reviewing current research developments and industrial implications. In this work Leitão identified the lack of industrial adoption of agent based architecture. In 2011 Leitao and Vrba [14] continued this survey collating all agent based industrial applications in the manufacturing domain, in the logistics domain, and in other domains that included dynamic rescheduling, intelligent decision support, and energy demand forecasting. Additionally multiple European Union (EU) research and development projects were reviewed; ArchitecturE for Serice-Oriented Process (AESOP) monitoring control,InteGration of pRocess and quAlity Control using multiagEnt technology (GRACE), Instantly Deployable Evolvable Assembly Systems (IDEAS), COSt-driven adaptive factory based on MOdular Self contained factory units (COSMOS), Soft Collaborative Intelligent Systems (COLLIS.EUS), Cooperative Objects Network of Excellence (CONET). In 2012 Barbati et al [15] provided a state-of-the-art survey of agent based models for use in optimisation problems. The result of the review identified the use of agents for solving complex scheduling problems and supply chain planning problems. As can be seen from the previous research summaries agent based systems are being designed and continuously developed within an array of different manufacturing sectors for complex problem solving. The recent development and developments associated with the EU project AESOP outlines the forward direction for large scale machine control and process monitoring systems with unified architectures, and cross plant layer service oriented collaborations [16]. Additional to research endeavours a more standardised entity has evolved, associated with the standardisation of machine monitoring through use of agent based technology, known as MTConnect [17].

1.3. MTCONNECT

MTConnect is an open, extensible, and royalty free standard, that outlines a universal factory floor communication protocol for the shop floor environment, that enables users to link data from shop floor machines to software applications used to run their businesses [17]. The MTConnect protocol is based on standard internet technologies, such as; HyperText Transfer Protocol (HTTP), and eXtensible Mark-up Language (XML). The standard sets out a structure of five fundamental components which interact with one another (Fig. 1); (1) Device: represents a piece of equipment, commonly a machine tool or a data source, (2) Adapter: an optional piece of software/hardware that provides a link or conversion from the data source and proprietary data definition in the device to the MTConnect data definition standards, (3) Agent: a piece of software that collects, arranges, and stores data from the device or adaptor, while also receiving requests for the data from external applications, and processes the requests to further transmit the required data, (4) Network: the physical connection between a data source and the external data application, (5) Application: the actual requestor and consumer of MTConnect data.

The operational functionality of an MTConnect system can be summarised with the following definition; MTConnect organizes information and data from a data source, typically a machine, into an information model that defines the relationship between each piece of data and the source of that data [18]. Furthermore, this information model allows an application to interpret the data received from a data source and correlate that data to the original definition, value, and context. MTConnect was the solution proposed by the committee known as the Shop Floor Connectivity Working Group, to solve the problem

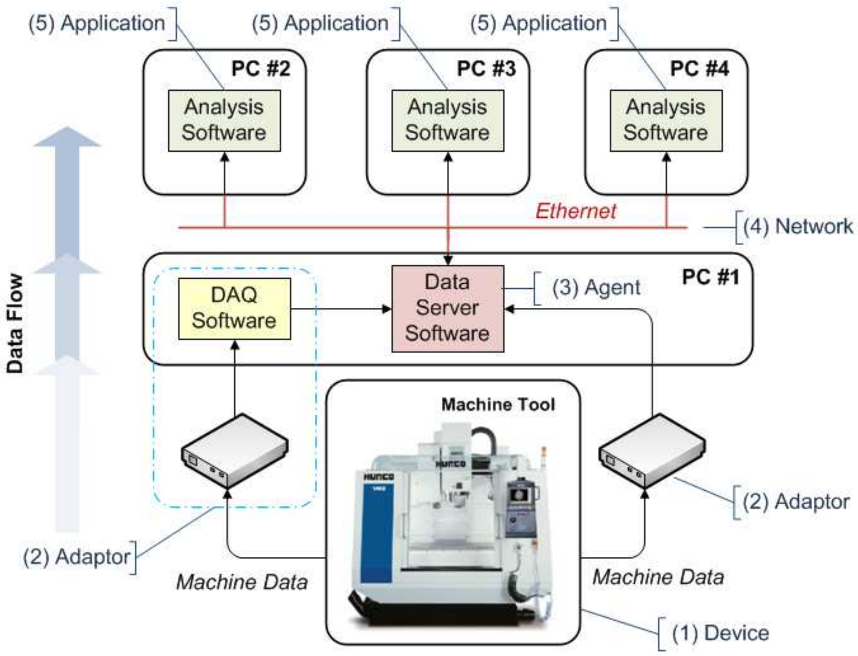

Fig. 1. MTConnect component interaction topology

of how to connect to existing machine tools and new machine tools. The distinctive difference of MTConnect compared to the numerous communication solutions available, such as; OMAC, Cam-X, OPC, is that MTConnect is the first standard to define a dictionary for manufacturing data, meaning that data from multiple machines will have common definitions, e.g. name, units, values, context. MTConnect can be seen to provide an array of different functional problem solving abilities for addressing machine tool process monitoring requirements, such as; production dashboard or monitoring, alerts, equipment availability and usage, machine downtime analysis, overall equipment effectiveness, overall equipment effectiveness, production reporting/tracking, maintenance tracking/planning. Examples of these applications can be reviewed in a report generated by the National Center for Defense Manufacturing and Machining (NCDMM) and The Connecticut Center for Advanced Technology, Inc. (CCAT) contracted with the MTConnect Institute, which was aimed at evaluating the MTConnect standard and software applications that use MTConnect data to address sustainable aerospace manufacturing [19].

The MTConnect standard has clear influences from agent based models, these influences can be seen in three points. Firstly MTConnect simply uses the word agent to represent one of the standards fundamental components. Secondly MTConnect has identified the benefits of agent based design through the adoption of the functional capability to achieve entity interaction to satisfy design/configured objectives. The data communication structure present requires several components to continuously communicate with one another in a hierarchical way to achieve local system goals and global system goals. Thirdly agent based design incorporates the separation of application functionality, and incorporates servicing functional capabilities with entity sub-systems interoperability. This distribution of function removes repetition within the MAS, allowing multiple entities to achieve their goals through interaction. This feature can be seen at the base of the MTConnect hierarchical communication structure, as the adaptor components allow agents to acquire data from multiple sources and format this data into a way the agents can understand. This feature facilitates an open source configuration for the agent to monitor multiple data sources, allowing it to be highly adaptable to an array of different environments through plug-and-play functionality.

1.4. AGENT BASED DESIGN AND MTCONNECT CORRELATION

MTConnect has clear influences from agent based design, but does not comply to some of the core fundamentals, including; resilient to changes and disruptions, exhibiting intelligence, problem solving, etc. Instead MTConnect has recognised some key traits of agent based design that would be of most benefit to process monitoring systems, which include; segmentation of system functions into autonomous entities, and element interactive communication. MTConnect then builds on this architecture to produce a connectivity, communication and structural standard for data interoperability for machine tools. Primarily the MTConnect standard outlines the method and medium for system interactions, which incorporates HTTP and XML, however for the sub-system elements, such as the adaptors, which provide a conceptual plug-and-play interoperability solution, is not defined. The method for achieving this is left up to the developer of the monitoring system. The benefits of this can be seen in the openness of the system to be developed in multiple ways, as long as the result is further converted for MTConnect standard compliance. However as stated previously, the reconfiguration required in process monitoring systems requires features to allow for current system changes and also allow for the incorporation of new elements from new components that have been included in the process, ultimately ensuring interoperability within the system. The structure developed to provide data interoperability at sub-system elements, e.g. adaptors, must allow for parameter reconfiguration of current elements, but more importantly must be open to connect to newly connected elements. This dynamic behavioural requirement outlines the need for developers to produce data interoperability agents that are open to multiple sub-element communication connections, to ensure multiple data sources can be incorporated into the process monitoring system dynamically.

2. RECONFIGURABLE PROCESS MONITORING DATA INTEROPERABILITY ARCHITECTURE DEVELOPMENT

The unique ability of process monitoring Data Interoperability Agents (DIA) is their ability to unite multiple process data streams, and ultimately facilitating data correlation. These streams can come in different data formats, such as numerical data, boolean data, text-based data, etc. All of which potentially represent an endless number of machine operations and reactive variables; vibration, energy consumption, rotational speed, robotic motion, etc. Each variable for monitoring may require a sensor, communication medium, and acquisition software application to produce the data required for analysis. This process is represented by the first four process monitoring steps characterised previously; measurement, acquisition, filtering, and analysis. An example of which would be a tri-axial accelerometer for vibration measurement, which would require a data acquisition unit to actively acquire the measurement data from the sensor, in addition the unit is required to communicate this data to a computational unit, for example across a Universal Serial Bus (USB) connection to a computer, where the data can be filtered and analysed by dedicated software. Due to the complexity and diversity of machine tools and automation processes a variety of different components may be required to monitor all the required process data for analysis. Within the MTConnect standard the entities known as adaptors are used to convert acquired data from multiple sources into a format that could be understood by and communicated to the DIA. A communication medium is required for a DIA to dynamically connect to and acquire data from multiple adaptors. The programming software utilised for development of this architecture was National Instruments LabVIEW, which is a system design software built specifically for tasks performed by engineers and scientists.

2.1. DATA INTEROPERABILITY COMMUNICATION MEDIUM CONSIDERATIONS

Several methods were evaluated for use within this work to achieve dynamic data interoperability for DIA's, including;

- Application Programming Interface (API), specifies how clients should interact with software components, API's are typically distributed as a software library, with each item consisting of a reusable functional building blocks that can be incorporated into end-user applications [20].
- File I/O, a simplistic method of transferring data between different software applications, through the utilisation of inter-application files, for example; text files, binary files, spreadsheet files, etc [21].
- TCP/IP, a set of communication protocols which utilise TCP/UDP ports for data transfer, with one object typically acting as a server, while the other acts as a client. A benefit of TCP/IP is that TCP/IP can be used across a variety of operating systems, since TCP/IP is based off the internet protocol suite and not a specific operating system [21].
- ActiveX, is a framework that defines reusable software controls that can be used across a variety of programming languages enabling cross application interaction, the framework is based on the Window Component Object Model (COM) and Object Linking and Embedding (OLE) technology, [21].
- National Instruments – Publish Subscribe Protocol (NI-PSP), a proprietary technology that provides fast reliable data transmission with Uniform Resource Locators (URL), through use of NI's Shared Variable Engine (SVE) [22].

The previously defined methods all provide a different way of data communication between software entities, however core requirements for the sub-level communication architecture are to dynamically communicate with other entities, while providing high data transmission rates. Dynamic communication is not possible with API's as another programming layer of operation and interpretation would be required to standardise the communication between applications to ensure dynamic plug-and-play operation. File I/O will allow for dynamic communication locally and across a network, but high data acquisition rates cannot be achieved due to the handshaking required via bi-directional communication, and the associated computational overhead required from file reading and writing. While TCP/IP, ActiveX and NI-PSP all provide for dynamic data interoperability.

An additional function required would be the ability to communicate across a network. This function will allow adaptors to be positioned across multiple computers that are networked together, while still allowing an agent to actively pole data. TCP/IP is based from the internet protocol suite and is naturally designed for communication across networks, while similarly NI-PSP facilitates optimised network data communication. AcitiveX does not facilitate cross network communication and would require another programming layer of operation and interpretation, which leaves TCP/IP and NI-PSP as the leading options. Other additional functional requirements of the sub-level communication architecture include; the transmission of different applications data types, the processing and buffering of incoming and outgoing data, and core plug-and-play architecture. All of these features are high level operational requirements that would utilise the communication methods discussed. However NI-PSP is a proprietary network associated with the NI Shared Variable Engine (NI-SVE), this framework facilitates all the additional functional requirements stated. Through use of the NI-SVE data communication, interaction, and operation is streamlined through a dedicated server allowing for network wide connectivity and data transmission (Fig. 2).

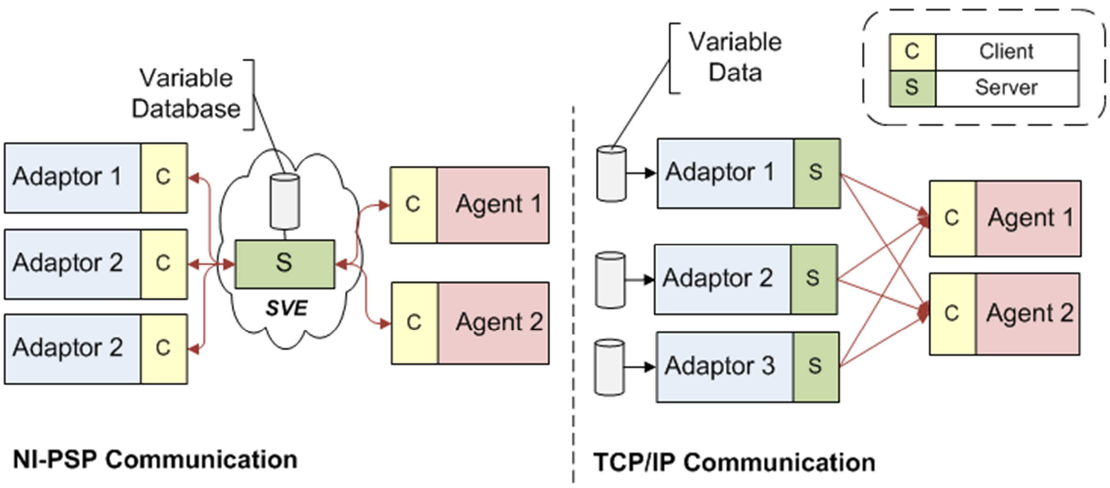

Fig. 2. NI-PSP and TCP/IP data interoperability agent communication topology

TCP/IP is a well established communication medium used worldwide for fast data transmission and open platform communication. However a framework would need to be built around the TCP/IP protocol to enable the service requirements of the dynamic data interoperability. This adds a lot of computational overhead associated with every data acquisition adaptor, and in true agent based design the NI-SVE segments this system function into an autonomous entity allowing other applications to utilise functions via NI-PSP, providing an efficient and robust solution for DIA's sub-level data communication.

2.2. DATA INTEROPERABILITY COMMUNICATION STRUCTURE

The NI-SVE utilise network published variables, these variables are defined within sinking software applications which are then published to the NI-SVE where the data is hosted for publishing to other sourcing software applications. An example of this would be an adaptor sinking process data to the NI-SVE and a DIA sourcing this data from the NI-SVE. The NI-SVE requires the programmer to define the type of variable that is to be published in the sinking application; numerical, boolean, wave-point, variant, etc. Once defined the connected sourcing application must interpret the data into the correct format. The architecture developed in this work outlines the use of 4 types of shared variable that will enable the DIA to dynamical control, monitoring and acquire high speed process data from the available adaptors. Due to the dynamic nature of the DIA the architecture within this work utilises the variant data type for communication. The variant data type stores both a value and meta-data specifying how data should be interpreted, providing a way for two applications to translate services between one another. The variant data type is defined in the Microsoft COM specification.

The 4 shared variables used for data interoperability are;

1. Controls: specifies controls that the DIA can utilise to actively control the operation of the adaptor. Controls include; Start: to acquire data, Stop: to stop acquiring data, Reset: to reset the adaptor, Shutdown: to terminate the application.
2. Statuses: specifies the optional status of the adaptor, allowing the operation of the adaptor to be monitored for successful operation. Statuses include; Alive: is the adaptor operational, Stream: is the adaptor actively acquiring data, Data-Error: has an error occurred, Error: error text associated with the occurrence of an error.
3. Data: the actively streamed data being produced by the adaptor, representing the process data being monitored by the adaptor. The data is sunk in packets of one dimensional array data types, with the first entry indicating the data ID relating to the data stream the data is related to, this data is then buffered by the NI-SVE with a First-In-First-Out (FIFO) structure. Additionally the NI-SVE handles the routing of data to sourcing adaptors, by replicating and publishing data independently to subscribed sourcing DIA's.
4. Information: identifies all core information associated with the adaptor for the DIA to successfully acquire all available data variables for monitoring. Information such as; the adaptors name, the quantity of variables for monitoring, the name of each variable, the data type of the associated variables, etc.

A DIA can utilise an adaptor by first successfully connecting to each of the adaptors shared variables. The connection interface in Fig. 3 outlines an example of the user interface connection panel for a DIA. The user must specify the address of each shared variable, this address is in a central location, and can be obtained by simply querying the NI-SVE, which provides a list of all published and active shared variables. These links are tested for successful operational connectivity through use of the available controls, and stored in a local database where other functions within the DIA can utilise the links to monitor, control and acquire data from the adaptor. These shared variable links are the only connection the DIA has to the adaptors, allowing the DIA to dynamically connect to multiple adaptors constructed to the specified architecture defined in this work.

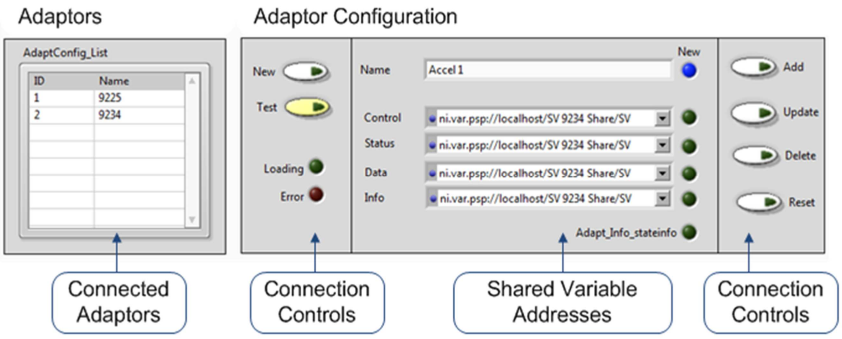

Fig. 3. Data interoperability agent shared variable configuration interface

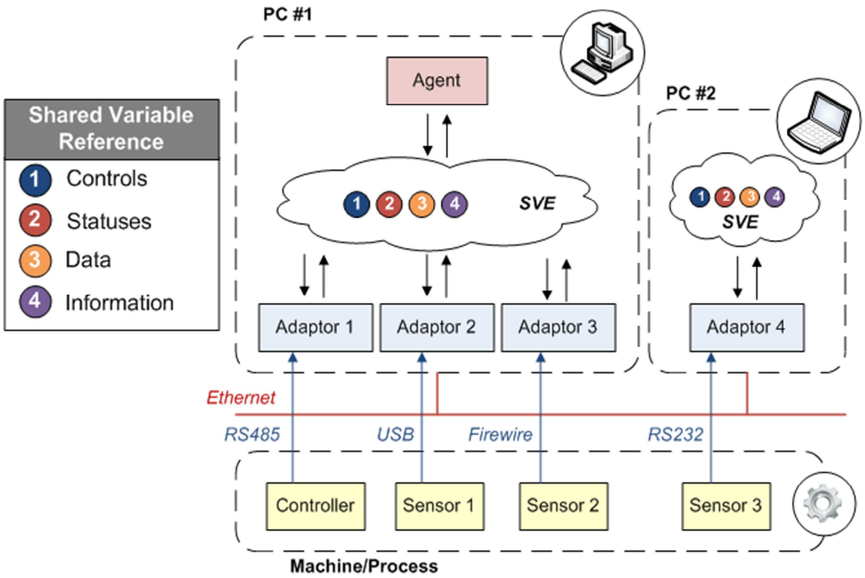

Fig. 4. Data interoperability agent shared variable configuration interface

A network topology of the DIA communication architecture can be seen in Fig. 4. In this example a singular DIA is connected to four adaptors, three are communicating locally on the computer hosting the DIA through the NI-SVE, while the forth adaptor is being hosted by a separate computer and is communicating across a network connection. The connectivity of the fourth adaptor does not differ in any way from the others as the two PC's hosting the NI-SVE are networked together through an Ethernet connection, allowing both NI-SVE's on both systems to be directly linked. This unique functionality allows the DIA to utilise user friendly plug-and-play operation. Each adaptor represents a different machine/process variable to be monitored, all of which have different communication mediums. This outlines the unique ability of DIA's to actively acquire and monitor a potentially limitless array of process data, as the adaptors provide for unique data acquisition while incorporate a conversion medium that facilitates data fusion via the DIA. The reconfigurable ability of the presented architecture will provide for the variance caused by manufacturing process changes, or the adaption requirements involved with incorporating new process machinery. This is achieved through the establishment of stability, robustness, sustainability, and reliability, within the defined data communication medium, and data interoperability communication structure. As the manufacturing process changes the only change required of the monitoring system is the introduction of new adaptors or the modification of configuration options in the process monitoring system.

3. CASE STUDY: MACHINE TOOL MONITORING VIA DYNAMIC DATA INTEROPERABILITY AGENT

In order to validate the developed data interoperability architecture presented within this work a case study was carried out to monitor a CNC machine using a variety of different measurement devices, which all communicate through different mediums, such as wireless communication, USB and Ethernet wired communication. The aim of the study was to demonstrate how multiple data sources could be integrated together using a data interoperability software agent. This agent would connect to, control, monitor and acquire process data, through utilisation of the NI-SVE in connection with the previously defined data interoperability architecture, to facilitate reconfigurable process monitoring. The CNC machine being monitored for the case study is a HURCO VM2 three axis CNC Milling machine, as seen in Fig.5. The parameters for measurement are machine power in the form of current (I) and voltage (V), and vibration at the spindle and the work-piece table in the form of tri-axial acceleration (g). These variables correspond to condition based machine monitoring variables, which are reactive elements within the process.

Spindle Drive	11 kW
Spindle Speed Range	400 – 8000 rpm
AC Voltage Supply	400 V (3 phase)
Frequency	50 Hz
Machinable Area	1168 mm x 457 mm
Tool Changer Capacity	16
Table Load Capacity	550 kg
Machine Weight	4080 kg

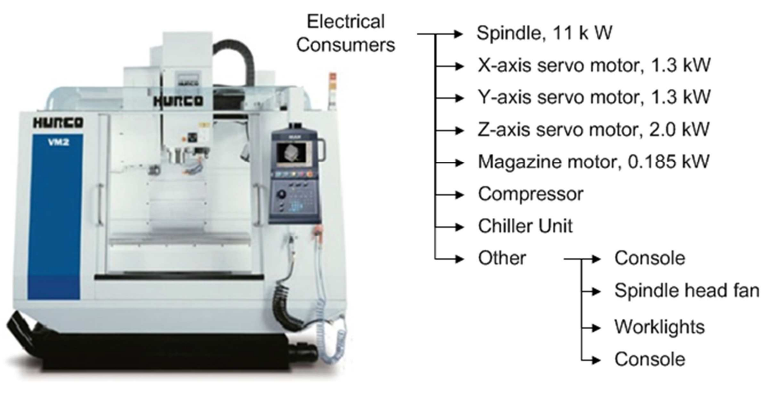

Fig. 5. Hurco 3-Axis CNC machine tool overview

3.1. MACHINE TOOL MONITORING - EQUIPMENT CONFIGURATION

A system connectivity overview of the interconnecting system components within the case study can be seen in Fig. 6. A Triaxial Integrated Electronics Piezo Electric (IEPE) accelerometer was placed on CNC spindle housing adhered with a magnetic mount, additionally a Triaxial IEPE accelerometer was placed on the work-piece table, also adhered with a magnetic mount. Each sensor was connected to a NI 9234 analogue signal acquisition module, these devices connected to individual NI cDAQ data acquisition devices for data collection. One NI cDAQ connected to PC number 1 via a Ethernet cable, while the other communicated to PC number 2 via a WiFi connection. The CNC machine utilised a 3 phase power supply system. A voltage divider circuit is utilised to scale the voltage from 400 Vrms to 200Vrms, to achieve measurement via a NI 9225 analogue input module, and three LEM current transformers are utilised to provide current measurement at a 1 x 0.0066 scale, to achieve measurement via a NI 9239 analogue input module. Both the NI 9225 and NI 9239 modules connect to a multi slot NI cDAQ 9178 data acquisition device, which communicates via a USB connection to PC number 1. Each measurement module has their

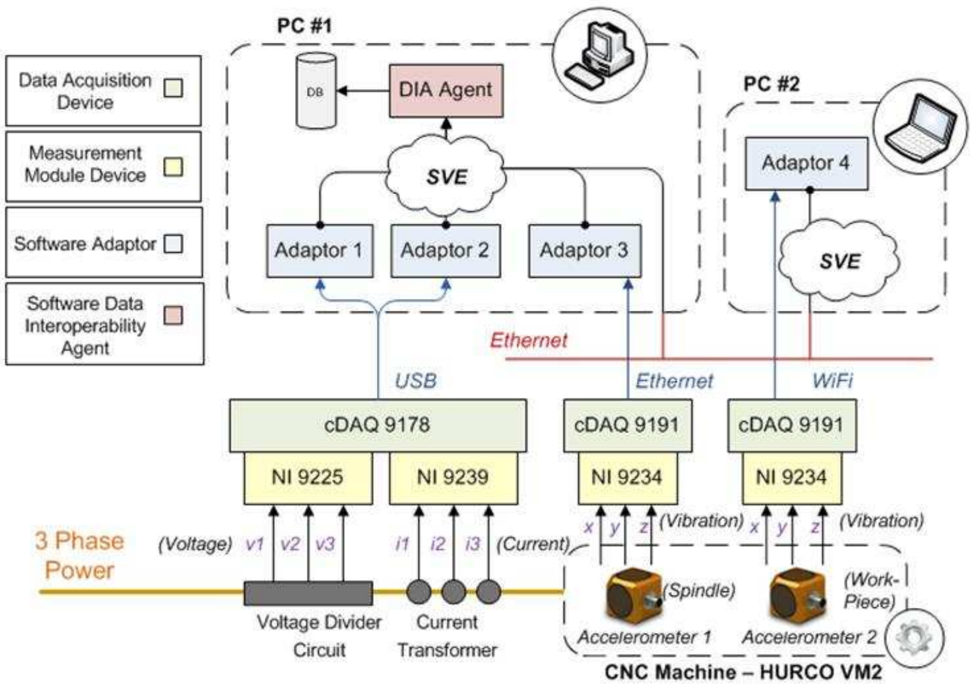

Fig. 6. Case study data acquisition connectivity overview

own software adaptor which provides configuration controls for acquiring the required data associated with their connected measurement module. Additionally there are operational controls, statuses and data display outlets present with the adaptors. These adaptors are designed for data communication via the NI-SVE and are designed to facilitate connectivity and interoperability via the DIA. There are 12 data channels being acquired by the DIA via the associated adaptors at different data acquisition rates. The NI 9225 and NI 9239 modules are both set to acquire samples at 5kHz across each channel, with each module acquiring three data channels for each phase of the CNC power supply. Both NI 9234 modules are acquiring data at 8kHz across each channel, with each module acquiring three data channels for each axis of the tri-axial accelerometers. The DIA is configured to acquire each data channel from the associated adaptor and processes data once data is received. The data produced by the adaptors is time stamped through referencing an internal PC CPU timer for accurate measurement. The DIA aligns the data from the different sources by synchronising the time stamps from the different data points acquired. Once data alignment is achieved, data is then stored in a limited data buffer for real-time utilisation and also stored in a database for post process monitoring analysis.

3.2. MACHINE TOOL MONITORING - DATA ACQUISITION AND ANALYSIS

The CNC machine was set to increase the spindle rotation speed at an increment of 1000RPM from 0RPM to 4000RPM over a period of 40 seconds. Over the 40 second period 3.12 Mega samples of data was produced in total, which is 24.96MB at 8 bytes per sample. The different data streams being acquired, formatted and communicated by the adaptors, are being aligned for real time parallel processing by the DIA. An example of the data acquired and aligned by the DIA is shown in Fig. 7. The electrical current being drawn by the machine tool is measured and aligned with the vibration data acquired from the spindle and work-piece table. The data alignment allows for variable correlation to be achieved between data channels. A base current is present when the spindle is stationary due to the CNC machine electrical consumers associated with machine operations, e.g. compressor, chiller unit, work lights, spindle head fan, etc. The amplitude of the current increases rapidly during spindle speed changes, then decreases and stabilises at a higher current amplitude than the previous base current. The large spike in current is due to the large power requirement of the CNC machine to accelerate the spindle, as maintaining a constant speed is not as power intensive. A reactive effect of increasing the spindle rotation speed is the increase in vibration present in the spindle housing due to the increase in rotary motion.

In order to further interpret the process monitoring data, analysis can be achieved through current-process monitoring and post-process monitoring. MTConnect utilises additional software applications which request data from the DIA to perform unique current-process data analysis. The DIA stores the multi channel process data to a file, to enable post process data analysis. The associated timestamps for each data channel is also recorded which allows for direct correlation of process parameters. An example of post-process data analysis can be seen in Fig. 8. The amplitude of the phase 1 electrical current and x-axis spindle vibration is extracted, then subjected to a moving average filter to create a trend line, and scaled to size to allow referencing between data channels. Additionally the original spindle x-axis vibration data is subject to a Butterworth low-pass filter set at a cut off frequency of 25Hz, and a moving average filter to provide a more intuitive representation of the vibration response. The results display a direct machine reaction to the

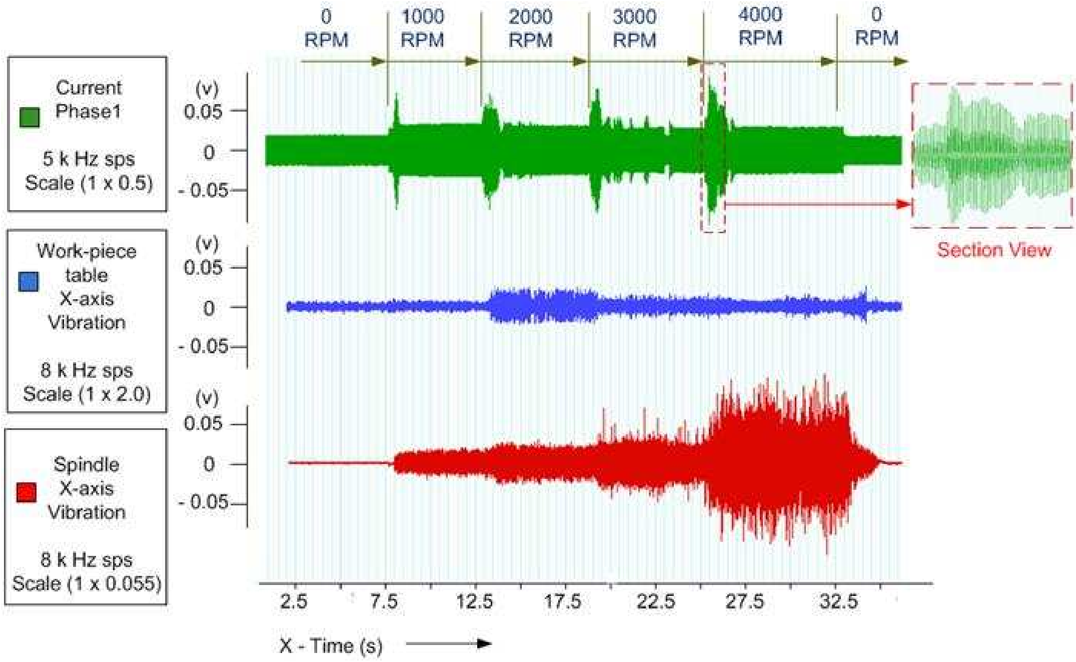

Fig. 7. Machine data integration, spindle speed acceleration increase; spindle vibration, work-piece vibration, and machine current vs time

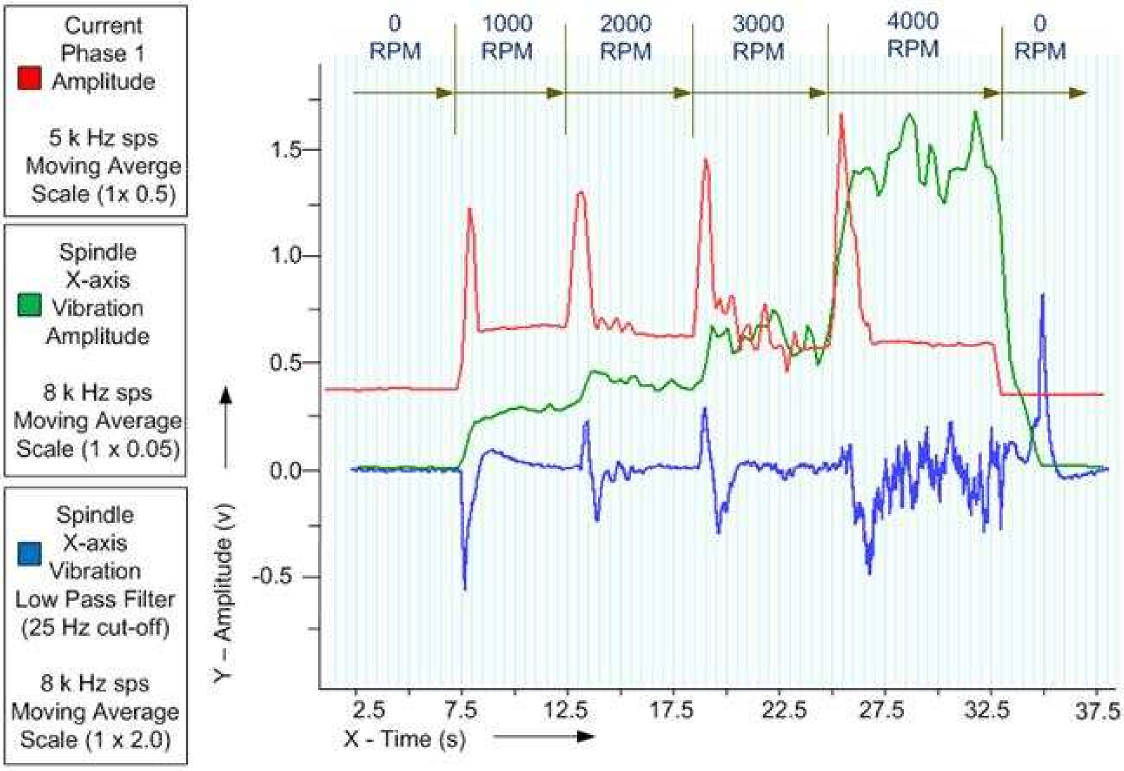

Fig. 8. Machine data integration under spindle speed acceleration increase; spindle vibration, work-piece vibration, and machine current vs time

increase in spindle speed, with a spike in electrical current amplitude and a stepped increase in vibration amplitude response, due to the incremental increase in spindle rotation speed. A clear representation of machine operation through data filtering and manipulation, allows for a clear identification of process variable interaction to be achieved. Achieving more defined data trends allows for a better understanding of variable correlation, while also providing for more stable robust current-process analysis, for autonomous supervisory control, as the post-process analysis can be transposed and adapted for current-process monitoring applications easily.

4. CONCLUSION

The realisation of a reconfigurable nature within manufacturing systems has been defined by the requirements of a dynamic market place. Specific tools are required to enable engineers to effectively create, implement, operate and sustain production quantity and quality, to facilitate this reconfiguration and dynamic behaviour. The establishment of MTConnect aims at providing a facility for reconfigurable manufacturing, which was achieved through the identification of key agent based design concepts to provide unique benefits and functional capabilities for process monitoring systems. The aim of the research presented herein was to introduce the concept of reconfigurable process monitoring systems, and outline the development of a novel highly effective data interoperability architecture that facilitates dynamic data integration for reconfigurable process monitoring systems. The utilisation of the NI-SVE enables the streamlining of process monitoring data, and enables plug-and-play functionality within the monitoring system. The data interoperability architecture defined allows for the reconfiguration and incorporation of different process monitoring variables for unified parallel data processing. The validity of the architecture was demonstrated in the case study where multiple process variables associated with condition based monitoring of a CNC machine. The ability to dynamically acquire different data sources enables a process monitoring system to adapt to the present environment as the process being monitored can change but the systematic approach to acquire the required process data does not. The flexibility within the system to adapt to changing environments further enables the analysis applications present to provide data correlation, support supervisory control, quantitative analysis, or decision support capabilities. Future work from the research present herein would be the development of dynamic data analysis applications to further source process data from DIA's, through an efficient, manageable, sustainable structure. The realisation of such entities would be the next step in providing a fully functional reconfigurable process monitoring system, all of which would require a solid base platform of data interoperability like the one defined in this work.

REFERENCES

[1] TETI T., JEMIELNIAK K.., O'DONNELL G., DORNFELD D., 2010, *Advanced monitoring of machining operations*, CIRP Annals - Manufacturing Technology, 59/2, 717-739.

[2] KOREN Y., HEISEL U., JOVANE F., MORIWAKI T., PRITSCHOW G., ULSOY G., van BRUSSEL H., 1999, *Reconfigurable Manufacturing Systems*, CIRP Annals - Manufacturing Technology, 48/2, 527-540.
[3] MONOSTORI L., Cs CSAJI B., KADAR B., PFEIFFER B., ILIE-ZUDOR E., KEMENY Zs., SZATHMARI M., 2010, *Towards adaptive and digital manufacturing*, Annual Reviews in Control, 34/1, 118-128.
[4] TIKKALA V.M.., LIISA S., JOUNELA J., 2012, *Monitoring of caliper sensor fouling in a board machine using self-organising maps*, Expert Systems with Applications, 39/12, 11228-11233.
[5] ABELE E., SIELAFF T., SCHIFFLER A., 2012, *Method for chatter detection with standard PLC systems*, Production Engineering, 6/6, 611-619.
[6] VIJAYARAGHAVAN A., DORNFELD D., 2010, *Automated energy monitoring of machine tools*, CIRP Annals - Manufacturing Technology, 59/1, 21-24.
[7] ECKSTEIN M., MANKOVA I., 2012, *Monitoring of drilling process for highly stressed aeroengine components*, CIRP Conference on High Performance Cutting, 5/1, 604 - 609.
[8] GRAUER M., KARADGI S., METZ D., 2010, *Proactive control of manufacturing processes using historical data*, Knowledge-based and intelligent information and engineering systems, 14/2, 399-408.
[9] BRECHER C., RUDOLF T., 2009, *Adaptive logging module for monitoring applications using control internal digital drive signals*, Production Engineering, 3/3, 305-312.
[10] GUNTER P., ALTINTAS Y., JOVANE F., KOREN Y., MITSUISHI M., TAKATA S., van BRUSSEL H., WECK M., YAMAZAKI K., 2001, *Open controller architecture – past, present and future*, CIRP Annals - Manufacturing Technology, 50/2, 463-470.
[11] MONOSTORI L., VANCZA J., KUMARA S.R.T., 2006, *Agent-based systems for manufacturing*, CIRP Annals - Manufacturing Technology, 55/2, 697-720.
[12] WOOLDRIDGEY M.J., CIANCARINI P., *Agent-oriented software engineering: The state of the art*, 2001, Springer-Verlag Berlin Heidelberg, 1/1, 1-28.
[13] LEITAO P., 2009, *Agent-based distributed manufacturing control: A state-of-the-art survey*, Engineering Applications of Artificial Intelligence, 22/7, 979-991.
[14] LEIT P., VRBA P., *Recent developments and future trends of industrial agents*, 2001, International conference on Industrial applications of holonic and multi-agent systems for manufacturing, 5/1, 15-28.
[15] BARBATI M., BRUNO G., GENOVESE A., 2012, *Applications of agent-based models for optimization problems: A literature review*, Expert Systems with Applications, 39/5, 6020-6028.
[16] DELSING J., ELIASSON J., KYUSAKOV R., COLOMBO A.W., JAMMES F., NESSAETHER J., KARNOUSKOS S., DIEDRICH C., 2011, *A migration approach towards a SOA-based next generation process control and monitoring*, Annual Conference on IEEE Industrial Electronics Society, 37/1, 1-6.
[17] MTConnect®, 2011, *Getting started with MTConnect - Connectivity Guide*.
[18] MTConnect®, 2012, *MTConnect® Standard Part 2 - Overview and Protocol*.
[19] MTConnect®, , 2011, *Project report sustanable aerospace manufacturing initiative*.
[20] Reddy, M., 2011, *API Design for C++2011*, 30 Corporate Drive, Suite 400, Burlington MA 01803, USA, Elsevier.
[21] National-Instruments, 2011, *Connecting LabVIEW to 3rd Party Software Packages*.
[22] National-Instruments, 2012, *Using the LabVIEW shared variable*.