Proceedings of ASME 2012 International Manufacturing Science and Engineering Conference
MSEC 2012
June 4-8, 2012 in Notre Dame, Indiana, USA

MSEC2012-7398

Integration of MTConnect and Standard-based Sensor Networks for Manufacturing Equipment Monitoring

Kang B. Lee[1], Eugene Y. Song, Peter S. Gu
National Institute of Standards and Technology
Gaithersburg, Maryland USA
[1]Contact Author

KEYWORDS
Adaptor, Agent, Agent Client, Device, IEEE 1451, Manufacturing Equipment, MTConnect, Network Interface, Sensor, Sensor Interface, Sensor Network, Sensor Standards, Smart Transducer

ABSTRACT

MTConnect is an open and extensible protocol designed for the exchange of data between shop floor devices and software applications. MTConnect allows manufacturers to facilitate retrieval of information and data from factory devices, such as machine tools, sensors, and controllers. Currently, MTConnect users read data from sensors through proprietary sensor interfaces using adaptors. The suite of Institute of Electrical and Electronics Engineers (IEEE) 1451 standards defines a set of open, common communication interfaces for sensor networks, including both sensor interfaces and network interfaces.

This paper proposes an integration architecture of MTConnect with IEEE 1451 standard-based sensor networks. In the architecture, MTConnect plays a network interface role in the IEEE 1451 standard-based sensor networks via an MTConnect Agent. An adaptor is used to provide the mapping between the MTConnect Agent and the IEEE 1451 sensor network. A prototype system integrating MTConnect with IEEE 1451.2-based sensor network has been developed. Two case studies are provided to illustrate the integration.

NOMENCLATURE

IEEE Institute of Electrical and Electronics Engineers

HTTP Hypertext Transfer Protocol
NCAP Network Capable Application Processor
TEDS Transducer Electronic Data Sheets
TIM Transducer Interface Module
XML Extensible Markup Language

INTRODUCTION

MTConnect is a set of open, royalty-free standards intended to foster greater interoperability between controls, devices, and software applications by publishing data over networks using the Internet Protocol [1]. MTConnect allows manufacturers to facilitate the retrieval of data from machine tools, sensors, and controllers. Figure 1 shows the architecture of MTConnect. An Agent Client (application software) communicates with the MTConnect Agent using MTConnect request protocols to access shop floor devices, such as machine tools, robots, and sensors. MTConnect uses the Hypertext Transfer Protocol (HTTP) protocol as the underlying transport for all messages. The data must be sent back in a standardized Extensible Markup Language (XML) format. Each MTConnect Agent must represent at least one device or piece of equipment. It may represent more than one device, if desired. The Agent may use an adaptor to retrieve data from devices [2]. MTConnect organizes information and data for devices into a hierarchical structure that clearly defines the relationship between each piece of information (data) [3]. The MTConnect Agent collects data from various devices and delivers it to applications in response to Sample or Current requests. All the data is collected into streams and organized by devices and then by components [4].

The benefits of MTConnect are:

- Free and open protocol enables devices from different vendors to retrieve and share information in a common format.
- Improved device interoperability in the manufacturing area
- Significantly reduced startup time, overall costs, and long term maintenance of software application interfaces.

Thus MTConnect is a universal factory floor communications protocol that many experts and business owners from the manufacturing industry have agreed and accepted as necessary [5].

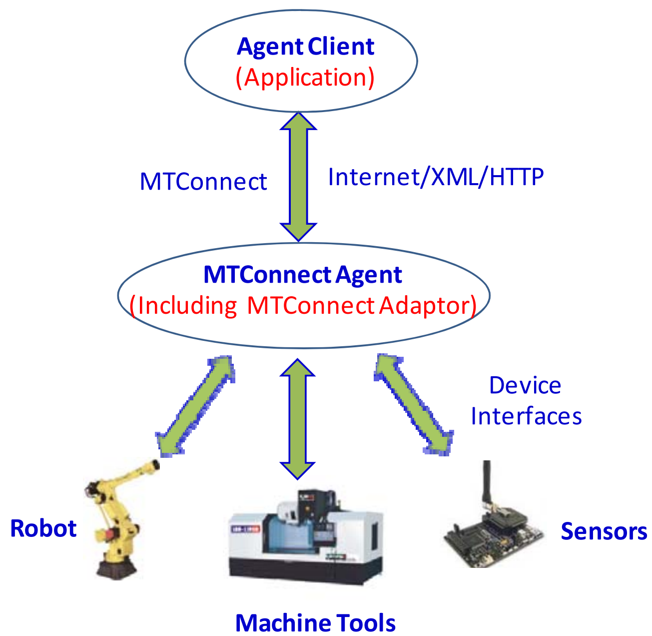


Figure 1 - Architecture of MTConnect

The Institute of Electrical and Electronics Engineers (IEEE) 1451, a family of Smart Transducer Interface Standards, defines a set of open, common, network-independent communication interfaces for connecting transducers (sensors or actuators) to microprocessors, instrumentation systems, and control/field networks [6-7]. Figure 2 shows the architecture of IEEE 1451 standard-based sensor networks. The architecture consists of a sensor application, a Network Capable Application Processor (NCAP), a network interface between the sensor application and NCAP, a Transducer Interface Module (TIM), and a transducer interface between the NCAP and TIM. The NCAP provides a gateway function for the sensor network. The TIM contains a wired and/or wireless interface, signal conditioning function, analog-to-digital and/or digital-to-analog converters, Transducer Electronic Data Sheets (TEDS), and transducer(s) grouped into transducer channels. According to the architecture, the sensor application communicates with the NCAP through the IEEE 1451.1 network interface. In turn, the NCAP communicates with the TIM via the sensor interface (IEEE 1451.X), via either a wired or wireless connection [6-7]. The main benefits of the IEEE 1451 standard include sensor self-identification, plug and play, ease of installation, upgrade, and maintenance, and reduction of the life cycle costs of sensor systems.

Presently, MTConnect users get data from sensors through proprietary sensor interfaces using adaptors. Sensors from different vendors may have different sensor interfaces; therefore, many sensor adaptors for MTConnect are needed to be developed. The development of sensor adaptors is a time-consuming and costly development. Interoperability with MTConnect depends greatly on how accurate the adaptors are implemented. The motivation of this paper is to integrate MTConnect with IEEE 1451 standard-based sensor networks to solve the problems mentioned above and to achieve sensor plug-and-play through an open, standard sensor interface. This paper mainly focuses on the integration architecture, the mapping between MTConnect and IEEE 1451 prototype system and two case studies.

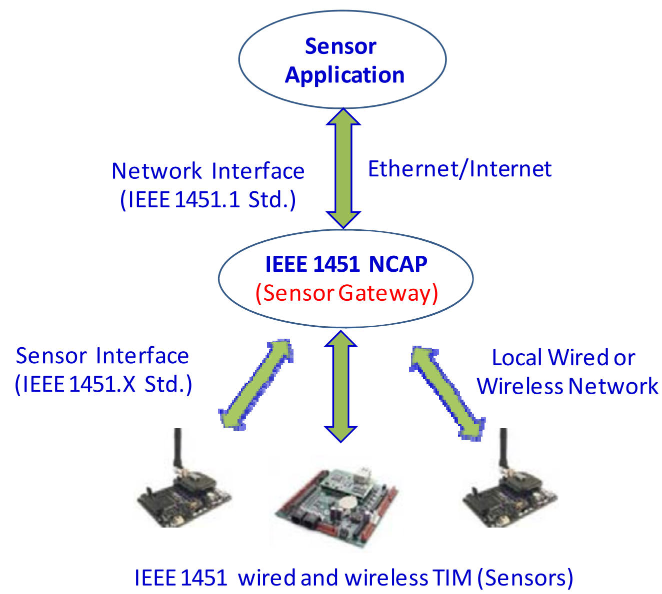


Figure 2 - Architecture of IEEE 1451 standard-based sensor networks

INTEGRATION ARCHITECTURE OF MTCONNECT WITH IEEE 1451 STANDARD-BASED SENSOR NETWORKS

MTConnect Protocol

MTConnect defines information and data of a device in a hierarchical structure. The hierarchical data structure of an MTConnect Device contains three primary containers: Devices, Device, and Components. Components may each contain within them multiple Components, as required, to provide a complete description of all data items. A Data Item is a piece of information that can be collected from a Device or Component. The value of the data is provided in the Streams response [3]. The MTConnect Agent collects data from various sources and delivers them to applications in response to Sample or Current requests. The data can be collected into streams and organized by Device, then by Component. A stream has three parts: Samples, Events, and Conditions [4]. Currently, MTConnect Agent supports three main types of requests [2]:

- **Probe request** – retrieves components and data items from a device. Response includes physical, design data, and calibration data of a device.
- **Current request** – retrieves a snapshot of the data item's most recent values or the state of the device at a point in time.
- **Sample request** – retrieves a list of past and/or current values for one or more data items.

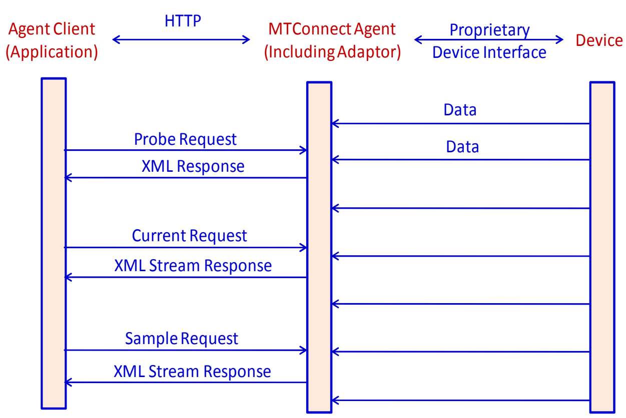


Figure 3 - MTConnect Communication Protocol

MTConnect communications among the Agent Client (Application), MTConnect Agent, and Device are shown in Figure 3 and can be described as follows:

- The Device continually sends data to the Agent through the proprietary Device interface. The Agent collects the data and saves it.
- The Agent Client sends a Probe request to the Agent. The Agent returns the component composition of the device as well as all the data items available.
- The Agent Client sends a Current request to the Agent, which returns the current state or measurement data of the device.
- The Agent Client sends a Sample request to the Agent, which returns samples of data from the device.

IEEE 1451 Communication Protocols

In the suite of IEEE 1451 standards, the IEEE 1451.1 standard defines a network interface for sensor networks [8], whereas the IEEE 1451.0 standard defines a common set of commands for communicating with sensors and actuators [9]. One of the primary goals of the IEEE 1451.0 standard is to achieve data-level interoperability when a multitude of wired and wireless sensor networks are connected together. Figure 4 shows the communication processes among the sensor application, NCAP, and TIM, which can be described as follows:

- A sensor application sends an IEEE 1451.1 request message to the NCAP through a user network, such as the Ethernet.
- The NCAP receives the IEEE 1451.1 request, converts it into an IEEE 1451.0 command message, and then sends it to the TIM through the IEEE 1451.X physical interface.
- The TIM receives the IEEE 1451.0 command message, processes it and gets an IEEE 1451.0 Reply message. Then it sends the Reply message back to the NCAP. For example, when a TIM receives a "read sensor data" command, it reads sensor data from a specific channel according to the command. Then the TIM sends the sensor reading back to the NCAP using the IEEE 1451.0 Reply message.
- After the NCAP receives and decodes the Reply message, it sends the sensor reading to the sensor application using the IEEE 1451.1 Response message.
- Finally, sensor application decodes the IEEE 1451.1 Response message to read the sensor data.

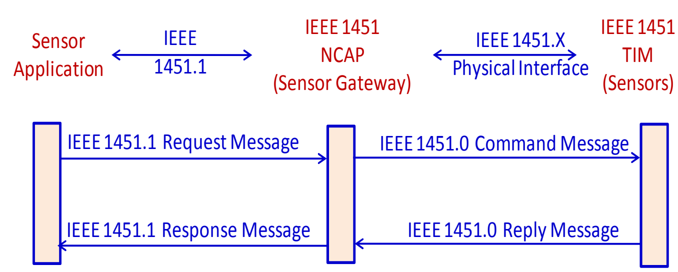


Figure 4 - IEEE 1451 Communication Protocol

Mapping Between MTConnect and IEEE 1451

In order to integrate MTConnect Sensor with IEEE 1451, a mapping between them is needed. Mapping between MTConnect and IEEE 1451 is challenging because of the differences in terminology used in the two standards. It is not a direct one-to-one mapping and care must be taken to avoid the loss of essential information.

In the MTConnect Standard Part 2 – Components and Data Items Version 1.2, the Sensor Component was introduced and described. After analyzing and examining the existing problems with sensors in MTConnect® Standard Part 2, we proposed a new sensor model in order to clarify the sensor concept and improve the current MTConnect Part 2 V1.2 specification [10]. The sensor model mainly consists of the concept of Sensor and SensorChannel. Sensor is a module, which contains at least one or more SensorChannels and an interface for outside communication to the Sensor and SensorChannel. Sensor is also an MTConnect Device or a Component. SensorChannel is a required Component of Sensor. A SensorChannel represents one sensing element that contains calibration information and handles signal conditioning and analog-to-digital conversion (ADC). MTConnect uses an HTTP protocol via the Internet, whereas IEEE 1451.1 works as a network interface. Therefore, Sensors and SensorChannels inherit all attributes and elements from the MTConnect Device.

Based on the proposed MTConnect Sensor specification, a mapping between MTConnect and IEEE 1451 was produced. Figure 5 shows how terms in the two standards are related. In this instance, MTConnect Sensor maps to the IEEE 1451 TIM, and the SensorChannel of the MTConnect Sensor maps into the TransducerChannel of the IEEE 1451 TIM.

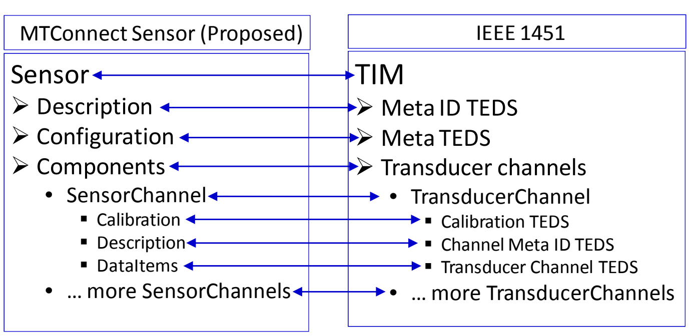


Figure 5 - Mapping between MTConnect and IEEE 1451

Integration Architecture

Figure 6 shows the integration architecture of MTConnect with the IEEE 1451 sensor networks. In the architecture, the MTConnect Agent resides in the NCAP. In lieu of IEEE 1451.1, the Agent acts as a network interface. An adaptor is used to access IEEE 1451.0 sensor data using MTConnect. In these scenarios, the communication processes to obtain a sensor reading can be described as follows:

- The NCAP reads sensor metadata from an available TIM and creates a device architecture XML file which can be read by MTConnect Probe requests.
- The NCAP periodically sends IEEE 1451.0 commands to read sensor data from a specified Sensor Channel on the TIM.
- The TIM receives commands from the NCAP and replies with the channel's current sensor reading.
- The NCAP receives the replies and uses the Adaptor to store the data in the MTConnect format in the MTConnect Agent repository.
- The sensor application sends an MTConnect Request message to the NCAP via an Agent Client through Ethernet or the Internet using HTTP protocol.
- The MTConnect Agent in the NCAP receives the Request message and retrieves the appropriate data from its repository. It then sends an XML reply back to the sensor application.
- Finally, the sensor application receives a response in XML format, which can be stored or displayed.

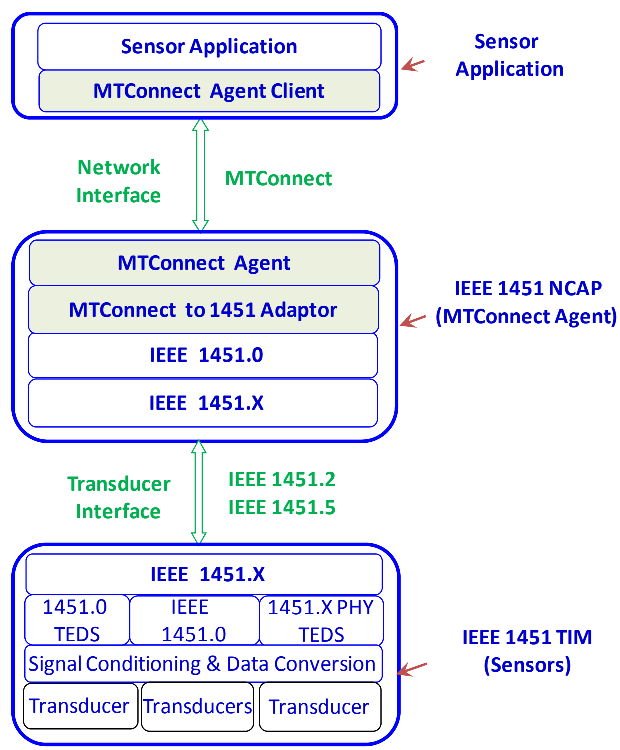


Figure 6 - Integration architecture of MTConnect with IEEE 1451 sensor networks

A PROTOTYPE SYSTEM INTEGRATING MTCONNECT WITH IEEE 1451.2 SENSOR NETWORK

Figure 7 shows a prototype system integrating MTConnect with an IEEE 1451.2-RS232 standard-based sensor network. The IEEE 1451.2 standard defines a wired serial point-to-point interface for sensors and actuators, specifically connectors and communication protocols for Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I2C), Universal Asynchronous Receiver/Transmitter (UART), and RS-232-C interfaces together with the corresponding Transducer Electronic Data Sheets (TEDS) [11-12]. This IEEE 1451.2 specification focuses on the Communication Modules that connect a TIM and NCAP using the IEEE Std. 1451.0-2007 standard. The prototype system consists of a sensor application (Agent Client), NCAP (Agent), and an IEEE 1451.2-RS232 TIM. The NCAP consists of an MTConnect Agent, an Adaptor between MTConnect and IEEE 1451.0, an IEEE 1451.0 module, and an IEEE 1451.2-RS232 communication module. The sensor application is web-based application software including an Agent Client, which can send standard MTConnect HTTP commands to the Agent and receive XML responses. The sensor application can communicate with the NCAP using the MTConnect protocol. The NCAP, implemented on a laptop computer, can communicate with the IEEE 1451.2 TIM by way of the IEEE 1451.2-RS232 serial interface. Inside the NCAP, the MTConnect Agent integrates with IEEE 1451.0 via an adaptor between MTConnect and IEEE 1451.0.

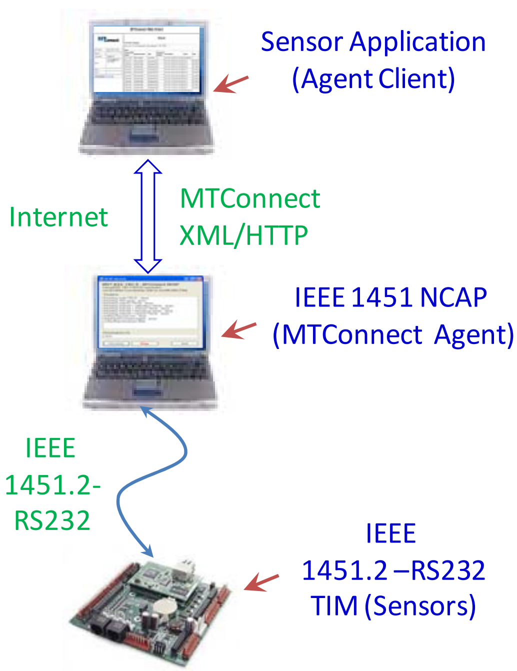


Figure 7 - Prototype integration system

IEEE 1451.2 TIM

The IEEE 1451.2-RS232 TIM consists of a thermistor-based temperature sensor, a Wolf BL2600** microprocessor board with an RCM3200 core module. Figure 8 shows the TIM hardware. A thermistor is attached to an ADC port of the RCM3200 module. The TIM is connected to the NCAP through an RS232 cable for serial communications. The TIM software is developed in Dynamic C based on the IEEE 1451.2-RS232 and IEEE 1451.0 standards. The TIM software functions include *read sensor data* (from the thermistor), *read TIM Meta TEDS*, *read Meta Identification TEDS*, *read User's Transducer Name TEDS*, *read Calibration TEDS*, *and read Transducer Channel TEDS* for the thermistor.

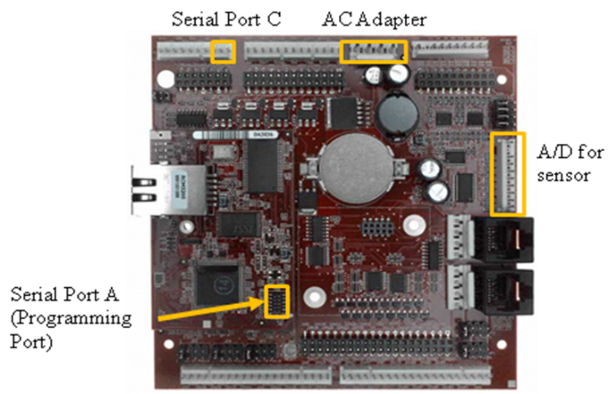


Figure 8 - Hardware setup for IEEE 1451.2 TIM

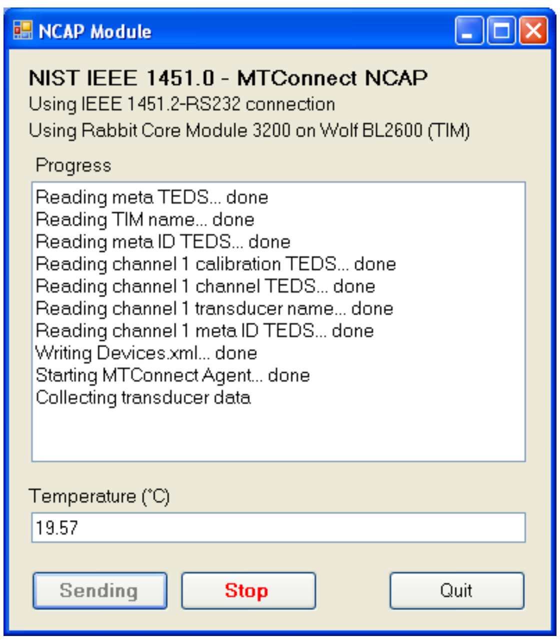


Figure 9 - NCAP Graphical User Interface

NCAP

The NCAP software running on a laptop includes MTConnect Agent, Adaptor, IEEE 1451.0, and IEEE 1451.2-RS232 communication modules. The MTConnect Agent is a slightly modified version of the open-source MTConnect Agent SDK Version 1.1, which was downloaded from the MTConnect website [1]. The Agent can receive HTTP requests from the Client and send XML responses back to Client. It also holds a repository of information about the sensor in the MTConnect standard format. The Adaptor can order the IEEE 1451.0 module to issue IEEE 1451.0 commands to read information from the TIM. It also translates this information into the MTConnect XML format. Figure 9 shows an NCAP Graphical User Interface.

Sensor Application – Agent Client

An Agent Client was developed in C# using ASP.NET. It can send standard MTConnect HTTP commands to the Agent and receive XML responses from the Agent. The MTConnect Client supports standard MTconnect requests including Probe, Current, and Sample. Each of these requests can be modified with additional parameters. Some of these can be chosen from a drop-down list, while others, such as a file path, must be typed in. Commands are sent to a specific Internet Protocol (IP) address.

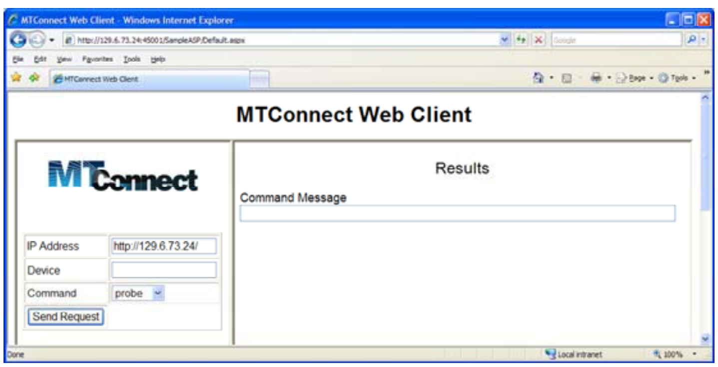


Figure 10 - Screenshot of a MTConnect Agent Client

CASE STUDIES

Case Study 1 - Probe Request

A Probe request retrieves the components and the data items of the device. The MTConnect sensor implemented is the IEEE 1451.2 TIM consisting of a thermistor for temperature measurements. The results from a Probe request are shown in Figure 11. The IEEE 1451.2 TIM has ID, description, data item, configuration, and components. The component of the TIM is a SensorChannel (thermistor). The SensorChannel has an ID. It also has DataItems and Calibration, but no sub-components. The result of Probe request is based on our proposed MTConnect Sensor XML Schema.

Case Study 2 – Sample Request

A Sample request is to retrieve a sample of sensor data from a device. This can include past data, which is stored in the MTConnect Agent repository. Figure 12 shows a Sample request returns in the form of a number of temperature readings from the IEEE 1451.2 TIM. Each temperature reading has a DataItem name, type (e.g., temperature), sequence number, timestamp, data value, and unit.

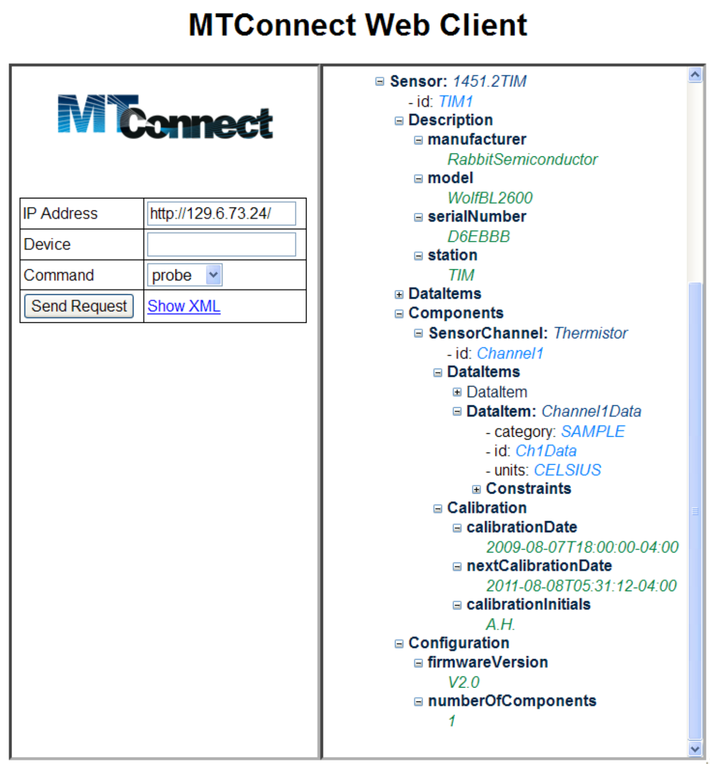
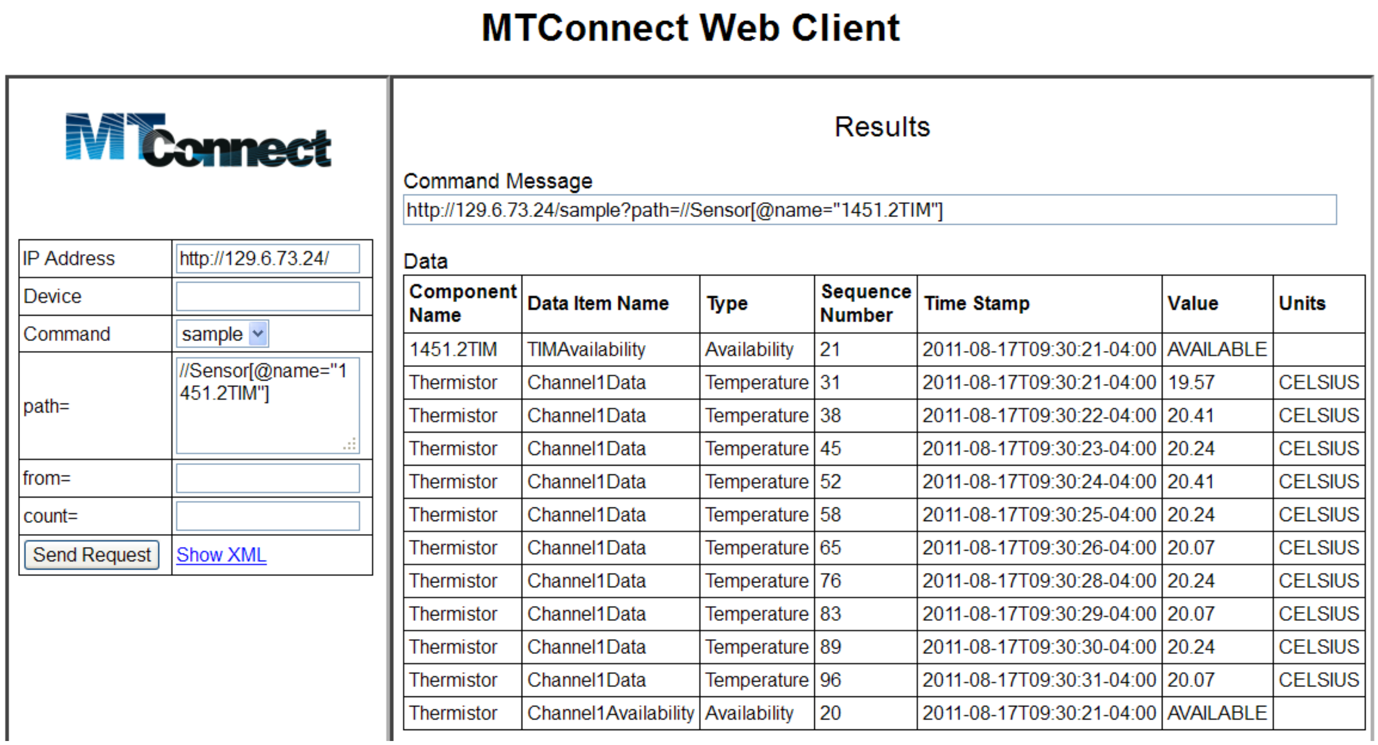


Figure 11 - Screenshot of a Probe request of a MTConnect Agent Client



Component Name	Data Item Name	Type	Sequence Number	Time Stamp	Value	Units
1451.2TIM	TIMAvailability	Availability	21	2011-08-17T09:30:21-04:00	AVAILABLE	
Thermistor	Channel1Data	Temperature	31	2011-08-17T09:30:21-04:00	19.57	CELSIUS
Thermistor	Channel1Data	Temperature	38	2011-08-17T09:30:22-04:00	20.41	CELSIUS
Thermistor	Channel1Data	Temperature	45	2011-08-17T09:30:23-04:00	20.24	CELSIUS
Thermistor	Channel1Data	Temperature	52	2011-08-17T09:30:24-04:00	20.41	CELSIUS
Thermistor	Channel1Data	Temperature	58	2011-08-17T09:30:25-04:00	20.24	CELSIUS
Thermistor	Channel1Data	Temperature	65	2011-08-17T09:30:26-04:00	20.07	CELSIUS
Thermistor	Channel1Data	Temperature	76	2011-08-17T09:30:28-04:00	20.24	CELSIUS
Thermistor	Channel1Data	Temperature	83	2011-08-17T09:30:29-04:00	20.07	CELSIUS
Thermistor	Channel1Data	Temperature	89	2011-08-17T09:30:30-04:00	20.24	CELSIUS
Thermistor	Channel1Data	Temperature	96	2011-08-17T09:30:31-04:00	20.07	CELSIUS
Thermistor	Channel1Availability	Availability	20	2011-08-17T09:30:21-04:00	AVAILABLE	

Figure 12 - Screenshot of Sample request of an Agent Client

SUMMARY

This paper describes an integration architecture of MTConnect with IEEE 1451 standard-based sensor networks. In this architecture, MTConnect Agent acts as a network interface of an IEEE 1451-based sensor network, in which an adaptor translates IEEE 1451.0 sensor information to the MTConnect format. The integration of MTConnect with an IEEE 1451.2-based sensor network has been implemented. Two case studies are provided to verify the proposed integration architecture. Future work is to integrate MTConnect with IEEE 1451 wireless sensor networks.

The work described in this paper demonstrates how to integrate MTConnect with IEEE 1451 standard-based sensor networks. The benefit and impact of this work is to help MTConnect users to adapt to an open standard-based sensor interface and realize sensor plug-and-play, as well as to avoid the time-consuming and costly development of different sensor adaptors for different sensor networks deployed.

REFERENCES

1. MTConnect, http://www.mtconnect.org/institute/mtconnect-overview/faqs.aspx
2. MTConnect Institute. MTConnect Standard Version 1.2, Part 1: Protocol and Overview. http://mtconnect.org.
3. MTConnect Institute. MTConnect Standard Version 1.2, Part 2: Components and Data Items. http://mtconnect.org.
4. MTConnect Institute. MTConnect Standard Version 1.2, Part 3: Streams, Samples, and Events. http://mtconnect.org.
5. Getting Started with MTConnect Connectivity Guide, http://www.mtconnect.org/media/7312/getting_started_with_mtconnect_-_final.pdf
6. Eugene Y. Song, Kang B. Lee, Understanding IEEE 1451-Networked smart transducer interface standard. IEEE Instrumentation & Measurement Magazine, Vol. 11, number 2, pp. 11–17 (2008).
7. Eugene Y. Song, Kang B. Lee, "IEEE 1451.5 standard-based wireless sensor networks", Book chapter, Advances in Wireless Sensors and Sensor Networks, Springer (Lecture Notes in Electrical Engineering), Springer-Verlag GmbH, Tiergartenstrasse 17, 69121 Heidelberg, Germany, 2010.
8. IEEE STD 1451.1-1999, Standard for a Smart Transducer Interface for Sensors and Actuators – Network Capable Application Processor (NCAP) Information Model, IEEE Instrumentation and Measurement Society, TC-9, The Institute of Electrical and Electronics Engineers, Inc., New York, N.Y. 10016, SH94767, June 26, 1999.
9. IEEE STD 1451.0-2007, Standard for a Smart Transducer Interface for Sensors and Actuators – Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats, IEEE Instrumentation and Measurement Society, TC-9, The Institute of Electrical and Electronics Engineers, Inc., New York, N.Y. 10016, SH99684, October 5 (2007).
10. Kang B. Lee, Eugene Y. Song, Peter Gu, A Sensor Model for Enhancement of Manufacturing Equipment Data Interoperability, ASME 2012 International Mechanical Engineering Congress & Exposition, IMECE 2012, November 9-15, 2012 in Houston, Texas, USA (Submitted).
11. IEEE Proposed 1451.2 Standard for a Smart Transducer Interface for Sensors and Actuators – Serial Point-to-Point Communication Protocol. http://grouper.ieee.org/groups/1451/2/
12. Eugene Y. Song, Kang B. Lee, "Sensor Network based on IEEE 1451.0 and p1451.2-RS232", 2008 I2MTC - International Instrumentation & Measurement Technology Conference, Victoria, Vancouver Island, British Columbia, Canada, May 12-15, 2008.