Proceedings of ASME 2012 International Mechanical Engineering Congress & Exposition
IMECE 2012
November 9-15, 2012 in Houston, Texas, USA

IMECE2012-88726

A Sensor Model for Enhancement of Manufacturing Equipment Data Interoperability

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

Abstract
Sensors can provide real-time production information to optimize manufacturing processes in a factory. Recently, more attention has been paid to the application of sensors in smart manufacturing systems. Sensor data exchange, sharing, and interoperability are challenges for manufacturing equipment monitoring in smart manufacturing. Standardized sensor data formats and communication protocols can help to solve these problems. MTConnect is an open, free, extensible protocol for the data exchange between monitoring applications and shop floor devices which include machine tools, sensors, and actuators. This paper introduces a sensor model for MTConnect to enhance manufacturing equipment data interoperability. The sensor model defines a Sensor and SensorChannel, as well as an interface to access the Sensor and its SensorChannels, which include sensing element, calibration, signal conditioning, and analog-to-digital conversion (ADC) information. The sensor model has been implemented in a virtual milling machine with a built-in sensor. Two case studies of MTConnect Probe and Sample requests for sensor information are provided to verify the sensor model.

Keyworks:
Agent; Agent Client; Device; MTConnect; Sensor; Sensor Channel; Sensor Model; XML Schema

INTRODUCTION
Real-time production information is needed to optimize manufacturing activities in a factory [1]. Sensors can provide real-time production data for smart manufacturing systems via wired and wireless means. Therefore, sensors play a critical role in real-time monitoring and data collection in a shop floor to optimize manufacturing system operations. Sensor data exchange, sharing, and interoperability are major challenges for manufacturing equipment monitoring in smart manufacturing. Standardized sensor data formats and communication protocols can help to solve these problems. 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 [2]. MTConnect allows the manufacturing industry to facilitate retrieval of data from any device or data source, such as machine tools, sensors, and controllers. Figure 1 shows the architecture of MTConnect. In this architecture, the Agent Client (Application) communicates with the MTConnect Agent using MTConnect request protocols to access shop floor devices. MTConnect Agent may use different adaptors to access various devices through device proprietary interfaces.

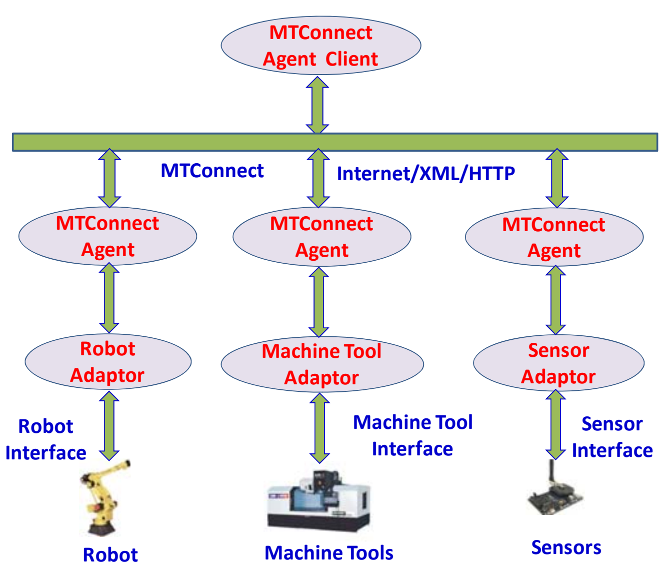

Figure 1. Architecture of MTConnect.

MTConnect has the following benefits:

- It is a free and open protocol which enables devices from different vendors to retrieve and share information in a common format.
- It improves device interoperability at the network (Ethernet/Internet) level in manufacturing systems.
- It significantly reduces startup time, overall costs, and long term maintenance of software application interfaces.

The National Institute of Standards and Technology (NIST) staff have been working with the MTConnect standard working group on standardized sensor data access and exchange. Sensor was introduced and described in the MTConnect® Standard Part 2 – Components and Data Items Version 1.2 [3]. The motivation of this paper mainly focuses on sensors for MTConnect, hereby called MTConnect sensors, to enhance manufacturing equipment data interoperability. This paper analyzes and examines the existing problems of MTConnect sensors, and then proposes a sensor model for improvement in MTConnect Part 2. The sensor model we provide is a sensor data model for MTConnect to achieve sensor data interoperability. It does not address data transmission delay between sensor(s) and the MTConnect Client(s), which are highly dependent on network or Internet speed and traffic, and the MTConnect communication protocol defined in MTConnect Part I. The sensor model is verified in a virtual milling machine with a built-in sensor via two case studies of MTConnect – a Probe request and a Sample request of sensor information.

MTCONNECT® STANDARD PART 2 – COMPONENTS AND DATA ITEMS VERSION 1.2

The MTConnect Part 2 defines Components and Data Items of Devices, which focuses on the data model and description of the information that is available from the device [3]. MTConnect organizes information and data for devices into a hierarchical architecture that clearly defines the relationship between each piece of information (data). This structure describes the components available in a device and what data items are provided by each component. A device is a piece of equipment capable of performing an operation. Figure 2 shows an eXtensible Markup Language (XML) schema of Device. The Device has a number of attributes, such as id (identification), name, nativeName, sampleInterval, and uuid (universal unique identification). It also has a few elements, such as Description, Configuration, DataItems, and Components. The device may be composed of a set of components.

MTConnect defines the metadata of a device as a hierarchical structure, which contains: Device, Components, and Sub-Components as shown in Figure 3. Components can be comprised of groups of Sub-Components. All attributes of a Component can be applied to a Sub-Component. Additionally, Any Sub-Component may have its own Sub-Components to provide a complete description of all data items. DataItem is a piece of information that can be collected from a Device or Component.

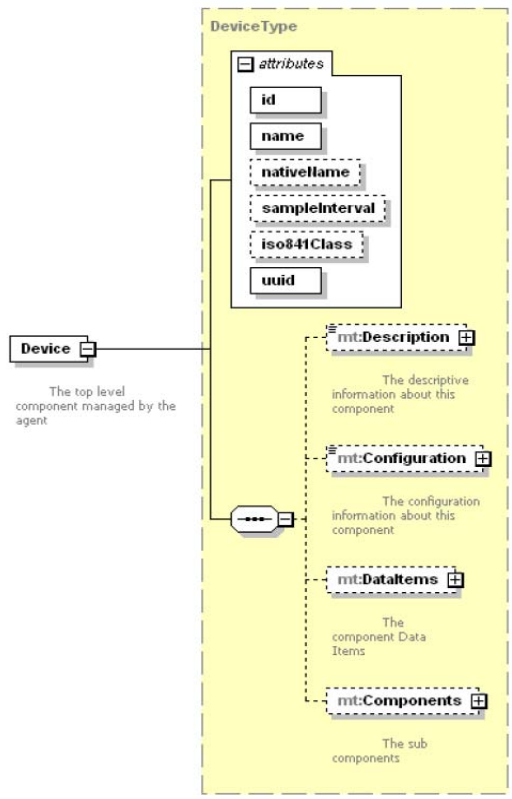

Figure 2. XML schema of Device.

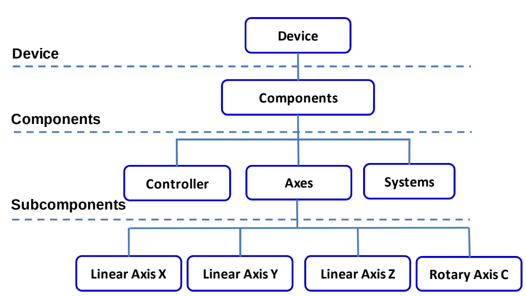

Figure 3. Example of Device Metadata as a Hierarchical Structure.

In MTConnect Part 2 V1.2, Sensor component was introduced and described. A **sensor** is typically comprised of two major components – a **sensing element** and a **sensor interface** [3].

- A **sensing element** provides a measured signal or discrete value. It is modeled as a **DataItem.**
- A **sensor interface** has capabilities, such as signal processing, conversion, and communications, it is modeled as a Component called **Sensor**.
- Each **sensor interface** may have multiple **sensing elements**, which represent the data for a variety of measured values.
- A **Sensor** may measure values associated with any Component, Sub-Component, or Device. When **Sensor** represents the sensor interface for multiple sensing element(s), each **sensing element** is represented by a **Channel**.
- A **Channel** represents one sensing element and can have its own attributes and Configuration data.

The existing problems of MTConnect Sensor can be summarized as follows:

- Sensor Interface and Sensor are easily confused. Normally, a sensor interface means a physical connection and related communication protocol, such as RS232 or RS485. A sensor interface does not have any sensing elements, while a sensor is a module consisting of both sensing elements and an interface. Therefore, a sensor interface can not be modeled as a Sensor.
- Sensing Element (Channel) and DataItem are rather confusing. Normally, a sensing element is an element used to sense a physical phenomenon, such as a thermister. Sensing Element is a sub component, which has its own attributes and configuration data. While DataItem is only a piece of information that can be collected from a Device, Component, or Sub-Component, it is not a Sub-Component. Therefore, a sensing element cannot be modeled as a DataItem.
- Configuration means the arrangement of the parts or elements. Calibration is the act of checking or adjusting (by comparison with a standard or a reference) the accuracy of a measuring instrument. Calibration information should not be part of configuration as it currently is in Part 2. We separate Calibration from Configuration. Only SensorChannel has calibration information.

A SENSOR MODEL FOR MTCONNECT

We propose a sensor model for MTConnect based on our analysis in order to solve the problems described above and to clarify and improve the MTConnect Sensor concept. This sensor model defines Sensor, an MTConnect Component or Device, which shall consist of one or multiple SensorChannels and an interface to access the Sensor and each SensorChannel. Each SensorChannel is a component of Sensor.

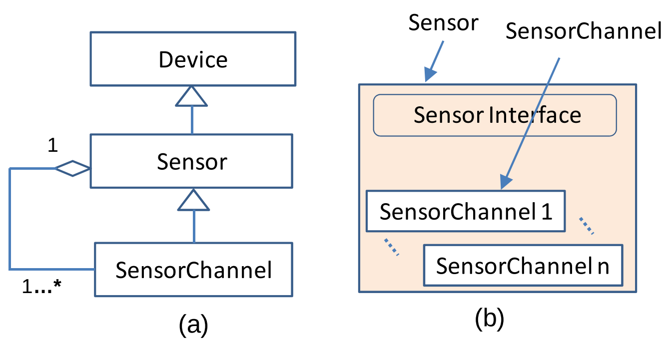

Figure 4. Sensor and SensorChannel.

- **Sensor** shown in Figure 4 (a) is an MTConnect Device or a component or sub-component. Therefore, Sensor inherits all attributes and elements from the MTConnect Device.
- **SensorChannel** shown in Figure 4 (a) is a component of Sensor. Therefore, SensorChannel inherits all attributes and elements from the Sensor and MTConnect Device. **SensorChannel** also has one sensing element, which contains calibration information and handles analog-to-digital conversion (ADC) and signal conditioning.
- A **Sensor** is a module shown in Figure 4 (b), which must contain one or multiple SensorChannels as well as an interface to access Sensor and each SensorChannel. It may output values that are synthesized from information from multiple sensor channels if desired.

A Sensor is used to monitor another device or component. A Sensor could be an independent Device if it is used to measure or monitor other Devices or Components as an external device, such as two Sensors shown in Figure 5, which are associated with Axis X and Spindle measurements, respectively. A Sensor could be a Component if the Sensor is built into the Device or Component. Two Sensors shown in Figure 5 are connected to Axis Y and Axis Z, respectively.

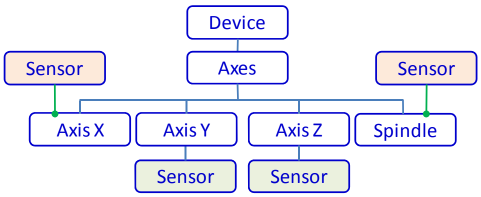

Figure 5. Example of Hierarchical Structure of Devices.

Figure 6 shows the XML schema of an MTConnect Device. Device has a number of attributes, such as id, name, nativeName, sampleInterval, and uuid. It also has a few elements, such as Description, Configuration, DataItems, and Components. The Description consists of manufacturer, model, serial number, and station. Configuration consists of firmwareVersion and numberOfComponents. Each Device has a number of DataItems and Components. We separated calibration from configuration, because only SensorChannel needs to be calibrated.

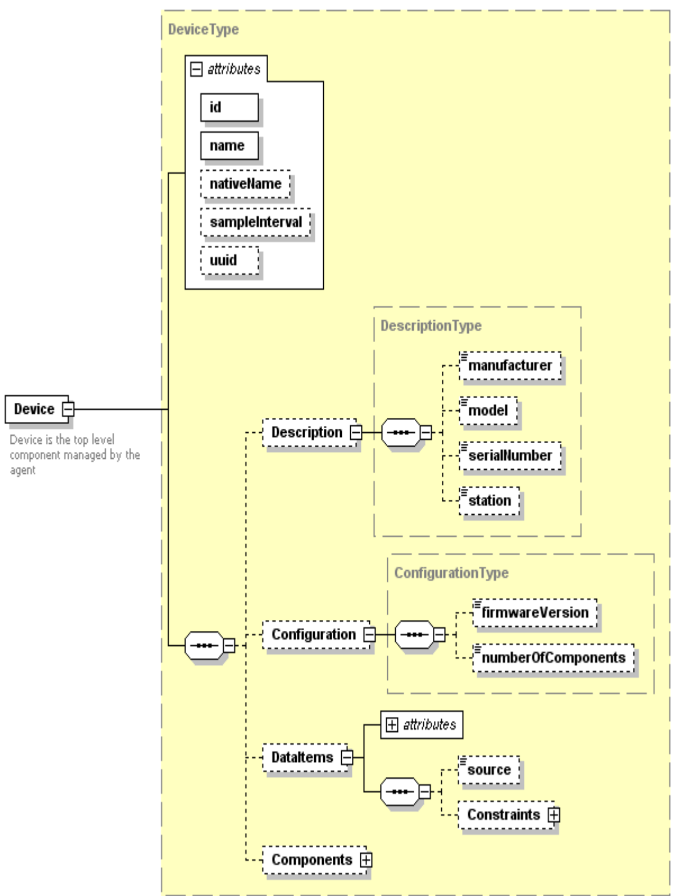

Figure 6. XML Schema of MTConnect Device.

Figure 7 Shows the XML schema of our proposed MTConnect Sensor. Sensor is a type of MTConnect Device. Therefore, Sensor has all attributes of a Device. It also has Description, Configuration, DataItems, and Components. Description consists of manufacturer, model, serial number, and station. Configuration consists of firmwareVersion and numberOfComponents. DataItems are output values that are synthesized from multiple sensor sources. Components consist of a number of SensorChannels.

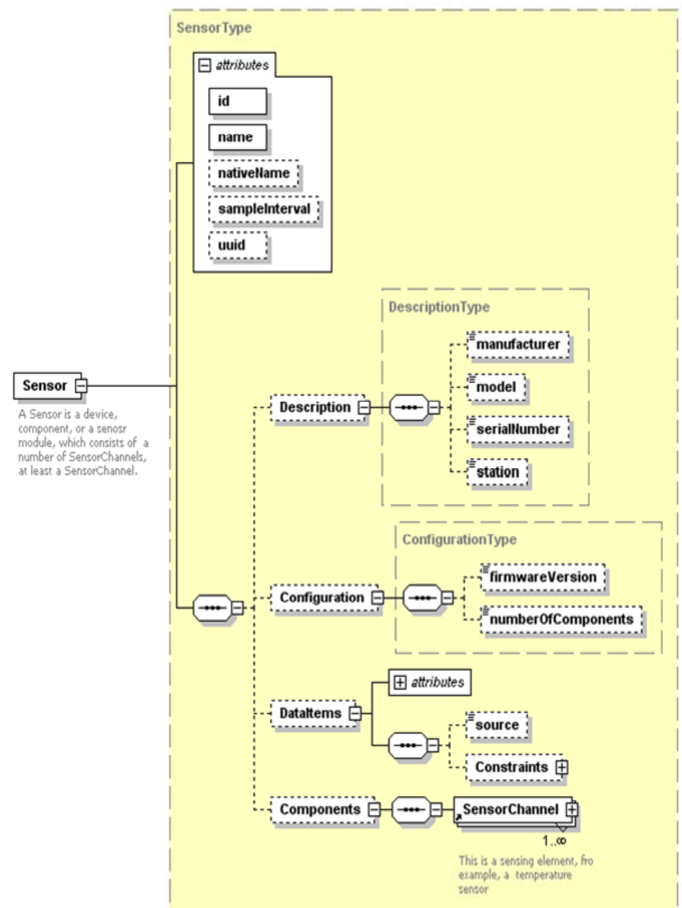

Figure 7. XML Schema of Sensor.

SensorChannel can be defined as a component of Sensor. It has one sensing element, contains calibration information, and handles analog-to-digital conversion (ADC) and signal conditioning. Figure 8 shows the XML schema of SensorChannel. SensorChannel has all attributes of Sensor. It also has Description, Configuration, DataItems, and additional Calibration. SensorChannel can not have any Component. SensorChannel has Calibration information, such as calibrationDate, calibrationInitial, and nextCalibrationDate. DataItems are measurement values.

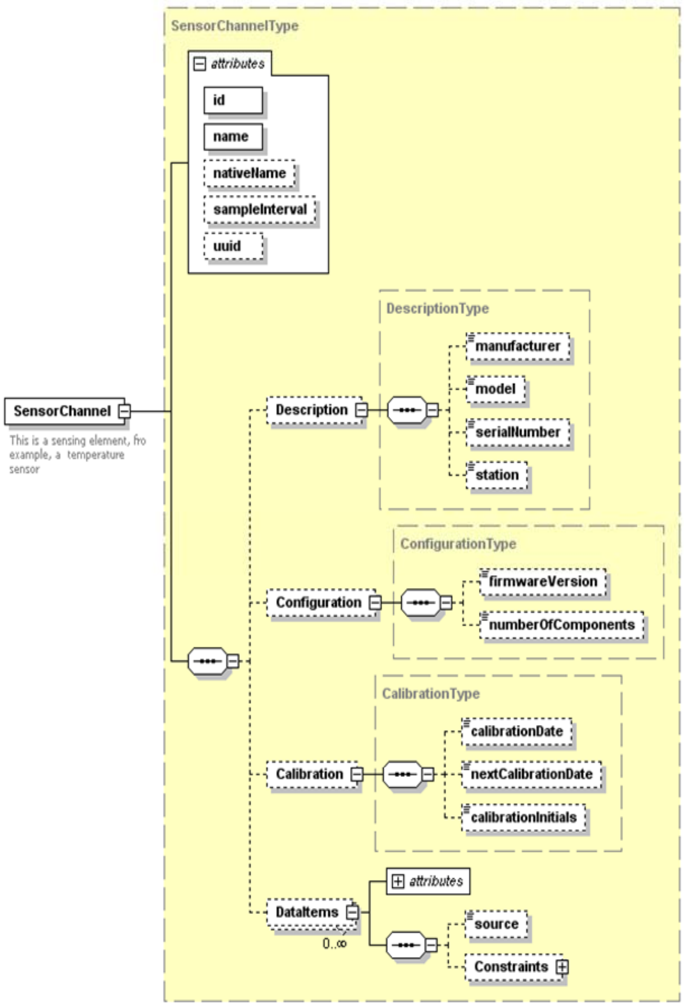

Figure 8. XML Schema of SensorChannel.

AN IMPLEMENTATION OF THE SENSOR MODEL FOR MTCONNECT

MTConnect Part 1 provides an overview and defines a communication protocol as shown in Figure 9. MTConnect specifies the use of Hypertext Transfer Protocol (HTTP) to transport all messages. The response data must be sent back in valid eXtensible Markup Language (XML) format. Each MTConnect Agent must represent at least one device, and may represent more than one device if desired. Currently, an MTConnect Agent supports three major types of requests [4]:

- Probe request – to retrieve components and data items of one or more devices. The XML response data include physical, design, and calibration data of the device. The device hierarchical data structure is defined in Part 2 [4].
- Current request – to retrieve the most recent values or the state of the device at a point in time. The XML stream data is a current value of DataItems.
- Sample request – to retrieve samples from a DataItems in a device.

The XML stream data formats of Current and Sample requests are defined in Part 3 [5].

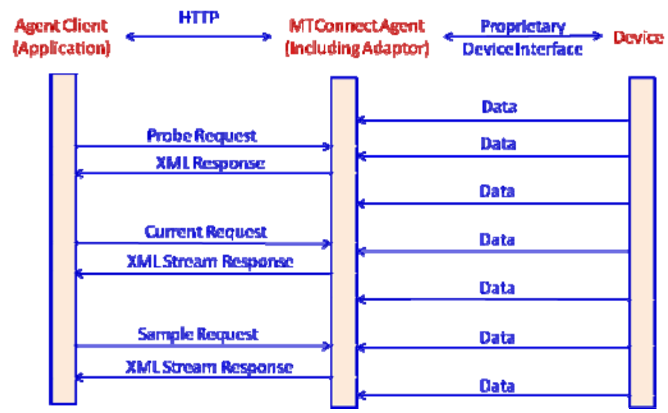

Figure 9. MTConnect Communication Protocol.

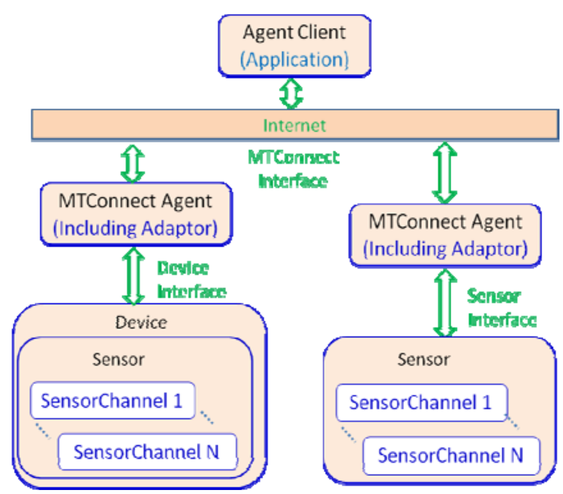

Figure 10. Accessing MTConnect Sensors.

Figure 10 shows how to access MTConnect Sensors, which can be classified into two types: a Sensor built into a device and an independent Sensor as a Device. The Sensor built into a Device is a component of Device. It shall be accessed through the device interface. The independent Sensor is an MTConnect Device, and it can be accessed through its sensor interface. A case study to access an independent Sensor is provided and discussed [6]. This paper discusses a sensor built into a device. Agent Client (Application) sends a request to the MTConnect Agent using the MTConnect protocol. The Agent may use an adaptor to access an independent Sensor's data through the sensor interface or a Sensor built into a Device through the device interface.

Figure 11 shows an implementation of the proposed MTConnect Sensor specification. This implementation prototype consists of an Agent Client (an application running on a laptop), an MTConnect Agent (implemented in a laptop), and a virtual milling machine with a built-in sensor.

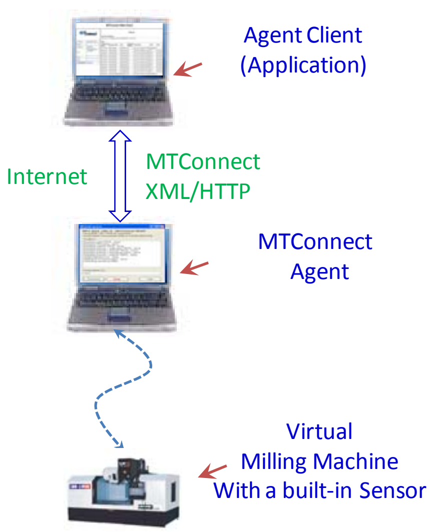

Figure 11. An MTConnect Reference Implementation.

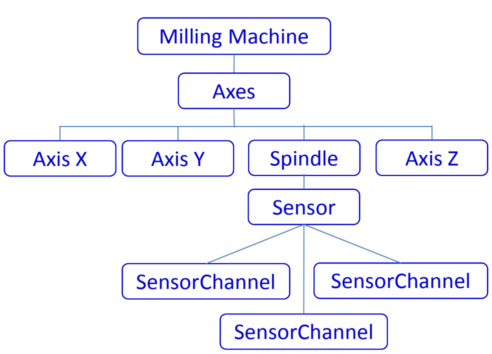

Figure 12. A Virtual Milling Machine with a Built-in Sensor.

Figure 12 shows a virtual milling machine with a built-in sensor. This virtual milling machine consists of three axes (X, Y, and Z) and a rotary axis (spindle). A displacement sensor is built into the spindle. It is a component of the spindle. This displacement sensor consists of three SensorChannels, which are used to monitor the spindle displacement in X, Y, and Z axes. The sensor data of this virtual milling machine is created in an XML file. When an Agent receives a Probe request from an Agent Client, it reads the XML file and then sends the response back to the Agent Client.

CASE STUDY

An MTConnect Probe request calls for the retrieval of the components and data items of the device. Figure 13 shows a screenshot of a Probe request. In this case, there are two devices: a virtual milling machine and a built-in Sensor. The milling machine shown in Figure 12 has three linear X, Y, and Z axes, and one rotary axis (spindle). The spindle has a Sensor component called SpindleDisplacement, which in turns has three SensorChannels: SpindleXDisplacement, SpindleYDisplacement, and SpindleZDisplacement. Figure 14 shows the response to a Probe request in XML.

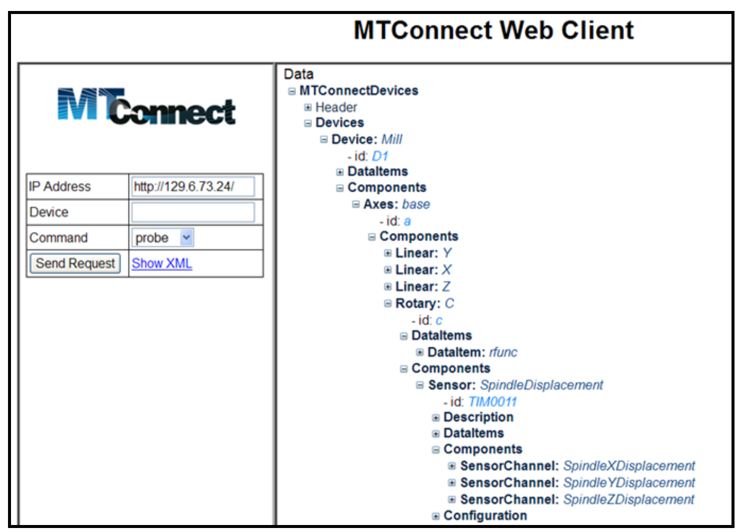

Figure 13. Screenshot of a Probe Request.

```
- <MTConnectDevices xsi:schemaLocation="urn:mtconnect.org:MTConnectDevices:1.1 http://www.mtconnect.org/schemas/MTConnectDevices_1.1.xsd">
    <Header version="1.1" sender="68.33.124.166:50000" creationTime="2011-08-16T08:56:42-04:00" bufferSize="5000" instanceId="1"/>
  - <Devices>
    - <Device id="D1" name="Mill" uuid="DDDDDDDDDDDDDDDDDDDDDDDDDDDD">
      + <DataItems></DataItems>
      - <Components>
        - <Axes id="a" name="base">
          - <Components>
            + <Linear id="y" name="Y"></Linear>
            + <Linear id="x" name="X"></Linear>
            + <Linear id="z" name="Z"></Linear>
            - <Rotary id="c" name="C">
              + <DataItems></DataItems>
              - <Components>
                - <Sensor name="SpindleDisplacement" id="TIM0011" uuid="ssssssssssssssssssss">
                  + <Description></Description>
                  + <DataItems></DataItems>
                  - <Components>
                    + <SensorChannel name="SpindleXDisplacement" id="TIM0011Ch1" uuid="xxxxxxxxxxxxxxxxxxxx"></SensorChannel>
                    + <SensorChannel name="SpindleYDisplacement" id="TIM0011Ch2" uuid="yyyyyyyyyyyyyyyyyyyy"></SensorChannel>
                    + <SensorChannel name="SpindleZDisplacement" id="TIM0011Ch3" uuid="zzzzzzzzzzzzzzzzzzzz"></SensorChannel>
                    </Components>
                  + <Configuration></Configuration>
                  </Sensor>
                </Components>
              </Rotary>
            </Components>
          </Axes>
        </Components>
      </Device>
    </Devices>
</MTConnectDevices>
```


Figure 14. Screenshot of XML Data of a Probe Request.

MTConnect Web Client

IP Address	http://129.6.73.24/
Device	Mill/
Command	sample
path=	
from=	1000
count=	6
Send Request	Show XML

Results

Command Message

http://129.6.73.24/Mill/sample?from=1000&count=6

Data

Component Name	Data Item Name	Type	Sequence Number	Time Stamp	Value	Units
Y	Yact	Position	1004	2011-08-17T13:17:57-04:00	-28.1907786235784	MILLIMETER
X	Xact	Position	1003	2011-08-17T13:17:57-04:00	-10.2606042997668	MILLIMETER
Z	Zact	Position	1005	2011-08-17T13:17:57-04:00	0.17	MILLIMETER
SpindleXDisplacement	XDisplacement	Displacement	1000	2011-08-17T13:17:57-04:00	0.0235	MILLIMETER
SpindleYDisplacement	YDisplacement	Displacement	1001	2011-08-17T13:17:57-04:00	0.0238	MILLIMETER
SpindleZDisplacement	ZDisplacement	Displacement	1002	2011-08-17T13:17:57-04:00	0.0269	MILLIMETER

Figure 15. Screenshot of Data of a Sample Request.

Figure 15 shows the screenshot of a Sample request with a sequence number of 1000. To make the display less cluttered, values were entered in the "from" and "count" text fields. The client program used the entered values 1000 and 6 and formatted them for use in an MTConnect command. These parameters tell the Agent to only send data items from the milling machine device that has sequence numbers between 1000 and 1005, inclusive.

SUMMARY

This paper describes a sensor model for MTConnect Sensor in order to improve manufacturing equipment data interoperability. The XML schema of the sensor model is presented. An implementation prototype of the sensor model for MTConnect has been developed and a case study of a virtual milling machine with a built-in sensor is provided to illustrate that the model works. The sensor model was also shown to work in a different case study in reference 6. This sensor model has been presented to the MTConnect Sensor Working Group for new version discussion. The contribution of this work is to clarify the Sensor concept via the sensor model.

Future work is to integrate MTConnect Sensor with IEEE 1451 wireless sensor networks and to pursue the harmonization of industry sensor standards.

** Commercial equipment and software, many of which are either registered or trademarked, are identified in order to adequately specify certain procedures. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

REFERENCES

1. Factory Equipment Network Testing Framework, http://www.nist.gov/el/isd/cs/facteqnetwtestfrwk.cfm
2. MTConnect, http://www.mtconnect.org/institute/mtconnect-overview/faqs.aspx
3. MTConnect Standard Version 1.2, Part 2: Components and Data Items. http://mtconnect.org/media/7562/mtc_part_2_components_1.2.0_draft_final_d.pdf
4. MTConnect Standard Version 1.2, Part 1: Protocol and Overview. http://mtconnect.org/media/7559/mtc_part_1_overview_1_2_0_final_draft_c.pdf
5. MTConnect Standard Version 1.2, Part 3: Streams, Samples, and Events. http://mtconnect.org/media/7565/mtc_part_3_streams_rel_1_2_draft_final_c.pdf
6. Kang Lee, Eugene Song, Peter Gu, "Integration of MTConnect with standard-based sensor networks for Manufacturing Equipment Monitoring", ASME 2012 International Manufacturing Science and Engineering Conference, MSEC 2012, June 4-8, 2012 in Notre Dame, Indiana, USA.