

MTConnect® Standard

Part 2 – Components and Data Items  
Version 1.2.0 – Draft Final A

Prepared for: MTConnect Institute

Prepared by: John Turner

Prepared on: July 6, 2011

MTConnect® Specification and Materials

AMT - The Association For Manufacturing Technology (“AMT”) owns the copyright in this MTConnect® Specification or Material. AMT grants to you a non-exclusive, non- transferable, revocable, non-sublicensable, fully-paid-up copyright license to reproduce, copy and redistribute this MTConnect® Specification or Material, provided that you may only copy or redistribute the MTConnect® Specification or Material in the form in which you received it, without modifications, and with all copyright notices and other notices and disclaimers contained in the MTConnect® Specification or Material.

If you intend to adopt or implement an MTConnect® Specification or Material in a product, whether hardware, software or firmware, which complies with an MTConnect® Specification, you SHALL agree to the MTConnect® Specification Implementer License Agreement (“Implementer License”) or to the MTConnect® Intellectual Property Policy and Agreement (“IP Policy”). The Implementer License and IP Policy each sets forth the license terms and other terms of use for MTConnect® Implementers to adopt or implement the MTConnect® Specifications, including certain license rights covering necessary patent claims for that purpose. These materials can be found at www.MTConnect.org, or by contacting Paul Warndorf at <mailto:pwarndorf@mtconnect.hyperoffice.com>.

MTConnect® Institute and AMT have no responsibility to identify patents, patent claims or patent applications which may relate to or be required to implement a Specification, or to determine the legal validity or scope of any such patent claims brought to their attention. Each MTConnect® Implementer is responsible for securing its own licenses or rights to any patent or other intellectual property rights that may be necessary for such use, and neither AMT nor MTConnect® Institute have any obligation to secure any such rights.

This Material and all MTConnect® Specifications and Materials are provided “as is” and MTConnect® Institute and AMT, and each of their respective members, officers, affiliates, sponsors and agents, make no representation or warranty of any kind relating to these materials or to any implementation of the MTConnect® Specifications or Materials in any product, including, without limitation, any expressed or implied warranty of noninfringement, merchantability, or fitness for particular purpose, or of the accuracy, reliability, or completeness of information contained herein. In no event shall MTConnect® Institute or AMT be liable to any user or implementer of MTConnect® Specifications or Materials for the cost of procuring substitute goods or services, lost profits, loss of use, loss of data or any incidental, consequential, indirect, special or punitive damages or other direct damages, whether under contract, tort, warranty or otherwise, arising in any way out of access, use or inability to use the MTConnect® Specification or other MTConnect® Materials, whether or not they had advance notice of the possibility of such damage.

Table of Contents

1 Overview 1

1.1 MTConnect® Document Structure 2

2 Purpose of This Document 3

2.1 Terminology 3

2.2 Terminology and Conventions 5

3 Devices and Components 6

3.1 Devices 8

3.1.1 Device 8

3.1.1.1 Device Structure 9

3.1.1.2 Device Attributes 9

3.1.1.3 Device Elements 10

3.2 Component 11

3.3 Component Schema 11

3.3.1 Component Attributes 12

3.3.2 Component Elements 12

3.3.2.1 Component Description 13

3.3.2.2 Component Components 14

3.3.2.3 Component DataItems 14

3.3.2.4 Component Configuration 14

3.4 Types of Components and Sub-Components 15

3.4.1 Axes 15

3.4.2 Controller 16

3.4.2.1 Path 16

3.4.3 ~~Power~~  Deprecated in Rel. 1.1 16

3.4.4 Door 16

3.4.5 Actuator 16

3.4.6 Sensor 17

3.4.7 Sensor Configuration 19

3.4.7.1 SensorConfiguration Elements 20

3.4.7.2 Sensor Channel Attributes 20

3.4.7.3 Sensor Channel Elements 20

3.4.8 Sensor Types 21

3.4.8.1 Thermostat (DEPRECATED in Rel. 1.2. See TEMPERATURE) 21

3.4.8.2 Vibration (DEPRECATED in Rel. 1.2. See DISPLACEMENT, FREQUENCY, etc.) 21

3.4.8.3 Acceleration 21

3.4.8.4 Angular Acceleration 21

3.4.8.5 Angular Velocity 21

3.4.8.6 Amperage 21

3.4.8.7 Angle 21

3.4.8.8 Concentration 21

3.4.8.9 Conductivity 21

3.4.8.10 Direction 21

3.4.8.11 Displacement 21

3.4.8.1 Electrical Energy 21

3.4.8.2 Flow 21

3.4.8.3 Frequency 21

3.4.8.4 Fill Level 22

3.4.8.5 Linear Force 22

3.4.8.6 Load 22

3.4.8.7 Mass 22

3.4.8.8 PH 22

3.4.8.9 Pressure 22

3.4.8.10 Position 22

3.4.8.11 Power Factor 22

3.4.8.12 Resistance 22

3.4.8.13 Rotary Velocity 22

3.4.8.14 Sound Level 22

3.4.8.15 Strain 22

3.4.8.16 Temperature 22

3.4.8.17 Time 22

3.4.8.18 Tilt 22

3.4.8.19 Torque 23

3.4.8.20 Volt Ampere (VA) 23

3.4.8.21 Volt Ampere Reactive (var) 23

3.4.8.22 Velocity 23

3.4.8.23 Viscosity 23

3.4.8.24 Voltage 23

3.4.8.25 Wattage 23

3.4.9 Systems 23

3.4.9.1 Hydraulic 23

3.4.9.2 Pneumatic 23

3.4.9.3 Coolant 23

3.4.9.4 Lubrication 23

3.4.9.5 Electric 24

4 Data Items 25

4.1 DataItem Schema 25

4.1.1 Data Item Attributes 27

4.1.2 Data Item Elements 28

4.1.2.1 Constraints Elements 28

4.1.3 Data Item attribute: category 29

4.1.4 Data Item attribute: coordinateSystem 30

4.1.5 Data Item attribute: units 30

4.1.6 Data Item attribute: statistic 31

4.1.7 Data Item attribute: representation 32

4.1.8 Data Item Attribute: nativeUnits 32

4.2 Data Item Types for SAMPLE Category 34

4.3 Data Item Types for EVENT Category 37

4.4 Data Item Types for CONDITION Category 39

5 Component and Data Item Relationships 41

5.1 Device 41

5.1.1 Device DataItems 41

5.1.2 Components of Device 41

5.2 Common Components and Related Data Items 41

5.2.1 Axes 42

~~5.2.1.1~~ ~~Axes DataItems~~ 42

5.2.1.2 Sub-components of Axes 42

5.2.2 Linear (Subcomponent of Axes) 42

5.2.2.1 Linear Axes DataItems (Samples and Events) 42

5.2.2.2 Linear Axes Conditions 42

5.2.3 Rotary (Subcomponent of Axes) 43

5.2.3.1 Rotary Axes DataItems (Samples and Events) 43

5.2.3.2 Rotary Axes Conditions 43

5.2.4 Controller 43

5.2.4.1 Sub-components of Controller 43

5.2.4.1 Controller DataItems (Samples and Events) 43

5.2.4.2 Controller Conditions 44

5.2.5 Path (Subcomponent of Controller) 44

5.2.5.1 Path DataItems (Samples and Events) 44

5.2.5.2 Path Conditions 44

5.2.6 ~~Power~~ DEPRECATED in Rel 1.1 44

5.2.7 Sensors 44

5.2.7.1 Sensor Conditions 45

5.2.8 ~~Thermostat~~ Deprecated in REL 1.2 Replaced with DataItem Temperature 45

~~5.2.8.1~~ ~~DataItem types~~ 45

~~5.2.8.2~~ ~~Condition types~~ 45

~~5.2.9~~ ~~Vibration~~ Deprecated in REL 1.2 Replaced with Data Items to measure Vibration (Displacement, Frequency, etc). 45

~~5.2.9.1~~ ~~DataItem types~~ 45

~~5.2.9.2~~ ~~Condition types~~ 45

~~5.2.10~~ ~~Pressure~~ Deprecated in REL 1.2 Replace with DataItem Pressure 45

~~5.2.10.1~~ ~~DataItem types~~ 45

~~5.2.10.2~~ ~~Condition types~~ 46

5.2.11 Door 46

5.2.11.1 Door DataItems (Samples and Events) 46

5.2.11.2 Door Conditions 46

5.2.12 Actuator 46

5.2.12.1 Acutator DataItems (Samples and Events) 46

5.2.12.2 Actuator Conditions 46

5.2.13 ~~Spindle~~ – DEPRECATED in Rel. 1.1 46

5.2.14 Systems 46

5.2.14.1 Sub-components of Systems 46

5.2.15 Hydraulic (Subcomponent of Systems) 47

5.2.15.1 Hydraulic Conditions 47

5.2.16 Coolant (Subcomponent of Systems) 47

5.2.16.1 Coolant DataItems (Samples and Events) 47

5.2.16.2 Coolant Conditions 47

5.2.17 Lubrication (Subcomponent of Systems) 47

5.2.17.1 Lubrication DataItems (Samples and Events) 47

5.2.17.2 Lubrication Conditions 47

5.2.18 Electric (Subcomponent of Systems) 48

5.2.18.1 Electrical DataItems (Samples and Events) 48

5.2.18.2 Electric Conditions 48

6 Annotated XML Examples 49

6.1 Simplest Device 49

6.2 More Complex Example of probe 50

Appendices 53

A. Bibliography 53

B. Machine Tool Modeling 55

B.1. Vertical Three Axis Mill 56

B.2. Two Axis Lathe 59

B.3. HyperQuadrex 61

B.4. Sensors 66

Table of Figures

Figure 1: Example Devices Structure 6

Figure 2: Device Schema Diagram 9

Figure 3: Component Schema 11

Figure 4: Component Schema 13

Figure 5: Axes Example With Three Linear Axes and one Rotary Axis 15

Figure 6: Example Sensors Structure 17

Figure 7: Example DataItem Structure 25

Figure 8: DataItem Schema Diagram 26

Figure 9: Constraints Schema 29

Figure 10: Right Hand Rule Coordinate Planes 55

Figure 11: Rotational Right Hand Rule 55

Figure 12: Three Axis Mill 56

Figure 13: Two Axis Lathe 59

Figure 14: HyperQuadrex Lathe 61

Figure 15: Spindle Sensing System 66

# Overview

MTConnect® is a standard based on an open protocol for data integration. MTConnect® is not intended to replace the functionality of existing products, but it strives to enhance the data acquisition capabilities of devices and applications and move toward a plug-and-play environment to reduce the cost of integration.

MTConnect® is built upon the most prevalent standards in the manufacturing and software industries, maximizing the number of tools available for its implementation and providing the highest level of interoperability with other standards and tools in these industries.

To facilitate this level of interoperability, a number of objectives are being met. Foremost is the ability to transfer data via a standard protocol which includes:

* + A device identity (i.e. model number, serial number, calibration data, etc.).
  + The identity of all the independent components of the device.
  + Possibly a device’s design characteristics (i.e. axis length, maximum speeds, device thresholds, etc.).
  + Most importantly, data captured in real or near-real-time (i.e. current speed, position data, temperature data, program block, etc.) by a device that can be utilized by other devices or applications (e.g. utilized by maintenance diagnostic systems, management production information systems, CAM products, etc.).

The types of data that may need to be addressed in MTConnect® could include:

* + Physical and actual device design data
  + Measurement or calibration data
  + Near-real-time data from the device

To accommodate the vast amount of different types of devices and information that may come into play, MTConnect® will provide a common high-level vocabulary and structure.

The first version of MTConnect® focused on a limited set of the characteristics mentioned above that were selected based on the fact that they could have an immediate effect on the efficiency of operations. Subsequent versions of the standard have and will continue to add additional functionality to more completely define the manufacturing environment.

## MTConnect® Document Structure

The MTConnect® specification is subdivided using the following scheme:

Part 1: Overview and Protocol – Version 1.2.0

Part 2: Components and Data Items – Version 1.2.0

Part 3: Streams, Events, Samples, and Condition – Version 1.2.0

Part 4: Assets – Version 1.2

All information applicable to all basic machines and devices will be included in the basic standard defined within these four parts. Additional parts will be added to provide information and extensions to the standard focused on specific devices, components, or technologies. All information specific to the topic of each additional part **SHALL** be included within that document.

Extensions to the standard will be made according to this scheme and new sections will be added as new areas are addressed. Documents will be named as follows:

MTC\_Part\_<Number>\_<Description>.doc.

All documents will be developed in Microsoft® Word format and released in Adobe® PDF format.

For example, this document is MTC\_Part\_2\_Components.doc.

# Purpose of This Document

The four base MTConnect® documents are intended to:

* define the MTConnect® standard;
* specify the requirements for compliance with the MTConnect® standard;
* provide engineers with sufficient information to implement Agents for their devices;
* provide developers with the necessary guidelines to use the standard to develop applications.

Part 2 of the MTConnect® standard focuses on structure and description of the information that is available from the device. The actual device state is not provided in this section, but is covered in Part 3 covering Streams, Samples, Events, and Condition. The descriptive data is similar to the schema of the data, it describes the components available in a device and what data items are provided by each component.

This part also covers instructions on how a piece of equipment should be modeled, the structure of the component hierarchy, the names for each component (if restricted), and allowable data items for each of the components. Some components, like Linear axis, use the naming conventions as laid out in this document. This allows for a consistent meaning across devices.

## Terminology

**Adapter** An optional software component that connects the Agent to the Device.

**Agent** A process that implements the MTConnect® HTTP protocol, XML generation, and MTConnect protocol.

**Alarm** An alarm indicates an event that requires attention and indicates a deviation from normal operation.

**Application** A process or set of processes that access the MTConnect® Agent to perform some task.

**Attribute** A part of an element that provides additional information about that element. For example, the name element of the Device is given as <Device **name=“mill-1”**>...</Device>

**CDATA** The text in a simple content element. For example, *This is some text*, in <mt:Condition ...>This is some text</mt:Condition>.

**Component** A part of a device that can have sub-components and data items. A component is a basic building block of a device.

**Controlled Vocabulary** The value of an element or attribute is limited to a restricted set of possibilities. Examples of controlled vocabularies are country codes: US, JP, CA, FR, DE, etc…

**Current** A snapshot request to the Agent to retrieve the current values of all the data items specified in the path parameter. If no path parameter is given, then the values for all components are provided.

**Data Item** A data item provides the descriptive information regarding something that can be collected by the Agent.

**Device** A piece of equipment capable of performing an operation. A device may be composed of a set of components that provide data to the application. The device is a separate entity with at least one component or data item providing information about the device.

**Discovery** Discovery is a service that allows the application to locate Agents for devices in the manufacturing environment. The discovery service is also referred to as the *Name Service.*

**Element** An XML element is the central building block of any XML Document. For example, in MTConnect® the Device element is specified as <**Device** >...</**Device**>

**Event** An event represents a change in state that occurs at a point in time. Note: An event does not occur at predefined frequencies.

**HTTP** Hyper-Text Transport Protocol. The protocol used by all web browsers and web applications.

**Instance** When used in software engineering, the word *instance* is used to define a single physical example of that type. In object-oriented models, there is the class that describes the thing and the instance that is an example of that thing.

**LDAP** Lightweight Directory Access Protocol, better known as Active Directory in Microsoft Windows. This protocol provides resource location and contact information in a hierarchal structure.

**MIME** Multipurpose Internet Mail Extensions. A format used for encoding multipart mail and http content with separate sections separated by a fixed boundary.

**Probe** A request to determine the configuration and reporting capabilities of the device.

**REST** REpresentational State Transfer. A software architecture where the client and server move through a series of state transitions based solely on the request from the client and the response from the server.

**Results** A general term for the Samples, Events, and Condition contained in a ComponentStream as a response from a sample or current request.

**Sample** A sample is a data point from within a continuous series of data points. An example of a Sample is the position of an axis.

**Socket** When used concerning interprocess communication, it refers to a connection between two end-points (usually processes). Socket communication most often uses TCP/IP as the underlying protocol.

**Stream** A collection of Events, Samples, and Condition organized by devices and components.

**Service** An application that provides necessary functionality.

**Tag** Used to reference an instance of an XML element.

**TCP/IP** TCP/IP is the most prevalent stream-based protocol for interprocess communication. It is based on the IP stack (Internet Protocol) and provides the flow-control and reliable transmission layer on top of the IP routing infrastructure.

**URI** Universal Resource Identifier. This is the official name for a web address as seen in the address bar of a browser.

**UUID** Universally unique identifier.

**XPath** XPath is a language for addressing parts of an XML Document. See the XPath specification for more information. <http://www.w3.org/TR/xpath>

**XML** Extensible Markup Language. <http://www.w3.org/XML/>

**XML Schema** The definition of the XML structure and vocabularies used in the XML Document.

**XML Document** An instance of an XML Schema which has a single root element and conforms to the XML specification and schema.

**XML NMTOKEN** The data type for XML identifiers. It SHALL start with a letter, an underscore “\_” or a colon “:” and then it **SHALL** be followed by a letter, a number, or one of the following “.”, ”-“, ”\_”, “:”. An NMTOKEN cannot have any spaces or special characters.

## Terminology and Conventions

Please refer to Part 1 “Overview and Protocol” Section 2 for XML Terminology and Documentation conventions.

# Devices and Components

MTConnect organizes information and data from a data source (typically a machine) into a structured architecture that clearly defines the relationship between each piece of information (data). This structured architecture also allows an application to interpret the data received and easily correlate that data to its original definition, value, and context.

The MTConnect data structure contains three primary containers: Devices, Device, and Components. Sub-Components are Components that when grouped together perform the function of a Component. Sub-Components provide an additional layer of granularity to further clarify the relationship between data items. All attributes of a Component can be applied to a Sub-Component. Additionally, Sub-Components may have Sub-Components, as required, to provide a complete description of all data items.

DataItem is a piece of information that can be collected from a Device or Component. The DataItem specifies the type of data being collected and an array of optional attributes that further defines that data. The value of the data is provided in the Streams response.

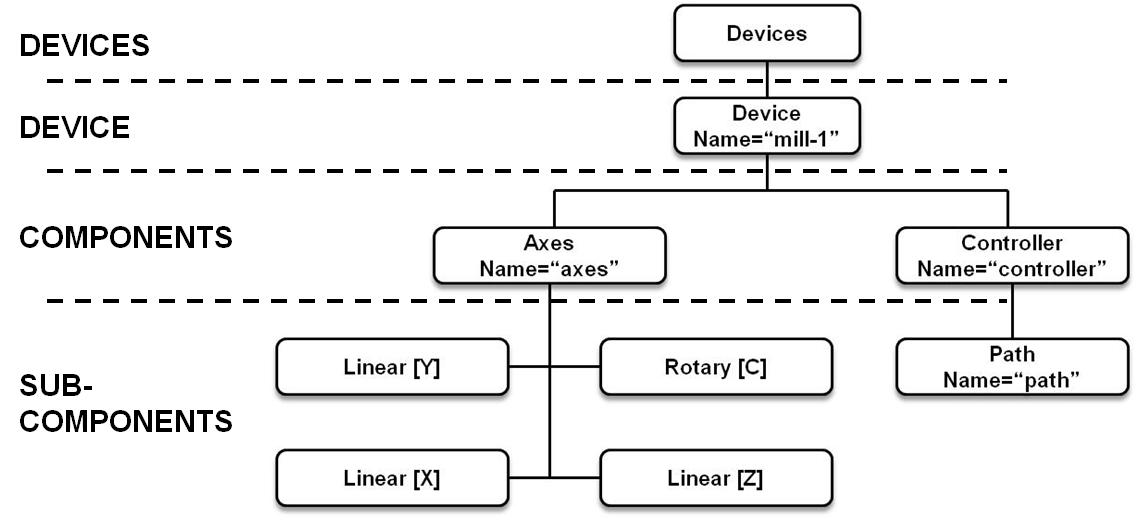


Figure : Example Devices Structure

The first, or highest, level container in the MTConnect data structure is the Devices container. The Devices container is comprised of one or more Device elements. Devices provides a mechanism for grouping data from multiple Device elements that are logically associated to each other. Devices has no attributes and is only used to group Device elements together.

A Device typically represents a single piece of equipment or a machine. However, it can also represent any logical grouping of Components that function together to perform a function. Every Device in MTConnect® **SHALL** have an Availability data item; availability represents the devices ability to provide information about itself.

In the MTConnect® schema, a Device is a unique type of Component (defined below). Although the Device is a Component, it shall be uniquely identified throughout the MTConnect® Standard as a Device to clearly define the difference between the logical collection of components that function together (Device) and each physical Component within that Device.

The next container in the MTConnect Data Structure is the Components container. The Components container groups all subcomponents together. Based on the example above, the Linear Axes and Rotary Axis are grouped together into the Axis component.

Some Components may not be integral to a parent device or another component. These components may function independently or produce data that is not relevant to a parent device. An example would be a temperature sensor installed in a plant to monitor the ambient air temperature. In this case, the Component **MAY** be modeled in the MTConnect schema as a Device. When modeled as a Device, the component **SHALL** provide all of the data and capabilities defined for a Device. It is also possible for these components to be defined as a Component of a parent device and simultaneously as an independent Device; communicating data associated with the parent Device incorporated into that device’s data stream and independently communicating additional data in a separate data stream using its own uuid.

Figure 1 above illustrates a Device called “mill-1”. The mill has the following components: Axes and Controller. The Axes component also has sub-components; these are the three Linear axes and one Rotary axis representing the spindle. The Controller component controls the axes and runs the program using a single Path component.

The following document structure defines Mill-1 described above:

MTConnectDevices  
 Devices  
 Device  
 Components  
 Axes

Rotary [C]

Constraints SPINDLE

DataItems

DataItem [Cvel]   
 Linear [X]  
 DataItems  
 DataItem [Xpos]  
 Linear [Y]  
 DataItems  
 DataItem [Ypos]  
 Linear [Z]  
 DataItems  
 DataItem [Zpos]  
 Controller

Path  
 DataItems  
 DataItem [mode]  
 DataItem [execution]

These containers make it easier to address individual parts of the XML document. For example, if one wanted to retrieve only the DataItems for the Controller you can express this using the following XPath: //Controller/DataItems/\*. If you were interested in retrieving only the subcomponets of the Axes component, you would write the following XPath: //Axes/Components/\*.

All Devices, Components, and DataItems require an id attribute. The id attribute **SHALL** adhere to the w3c standard ID-type and SHALL be unique within the entire XML document. The id attributes **SHALL** start with a :, \_, or letter (A-Z, a-z) and then may be followed with numbers, letters, -, or a period (.). For more information see: http://www.w3.org/TR/REC-xml/#NT-Name.

## Devices

The Devices element is the top level container for every Device. It may contain multiple Device elements. Devices is a similar container to Components except it may only contain elements of type Device.

| **Elements** | **Description** | **Occurrence** |
| --- | --- | --- |
| Device | The root of each device. The Device is contained within the top level Devices container. There can be multiple Device elements. | 1..INF |

### Device

A Device is an element that holds all the components associated with a piece of equipment or a logicial grouping of components that perform a function. The Device **SHALL** have an Availability data item that indicates if this device is available to provide information.

#### Device Structure

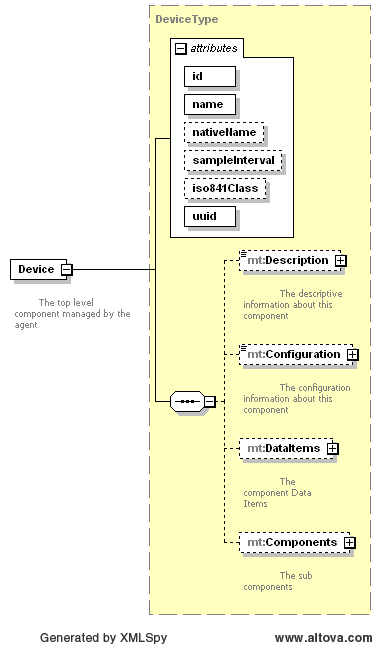


Figure : Device Schema Diagram

#### Device Attributes

| **Attribute** | **Description** | **Occurrence** |
| --- | --- | --- |
| iso841Class | **DEPRECATED in Release 1.1.0** |  |
| uuid | A unique identifier that will only refer to this device. For example, this may be the manufacturer code and the serial number. The uuid should be alphanumeric and not exceeding 255 characters. An NMTOKEN XML type. | 0..1\* |
| name | The name of the device. This name should be unique within the machine to allow for easier data integration. An NMTOKEN XML type. | 1 |
| nativeName | The name the device manufacturer assigned to the device. If the native name is not provided, it **SHALL** be the name. | 0..1 |
| id | The unique identifier for this device in the document. An id SHALL be unique across all the id attributes in the document. An XML ID-type. | 1 |
| sampleRate | **DEPRECATED IN REL. 1.2 (REPLACED BY sampleInterval** |  |
| sampleInterval | The interval in milliseconds between the completion of the reading of one sample of data from a device until the beginning of the next sampling of that data. This is the number of milliseconds between data captures. If the sample interval is smaller than one millisecond, the number can be represented as a floating point number. For example, an interval of 100 microseconds would be 0.1. | 0..1\*\* |

Notes: \* The uuid **SHALL** be provided for the Device. It is optional for all other Components. \*\* The sampleInterval is used to aid an application in interpolating values. This is the desired sample interval and may vary depending on the capabilities of the device.

#### Device Elements

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Description | An element that can contain any descriptive content. This can contain configuration information and manufacturer specific details. | 0..1 |
| Configuration | An element that can contain descriptive content defining the configuration information for a Device. | 0..1 |
| Components | Sub-components of this Device. | 0..INF\* |
| DataItems | The data items this Device provides. The data items are descriptions of the data events for reporting. | 0..INF\* |

Notes: \*At least one of Components or DataItems **SHALL** be provided.

## Component

Components are physical or logical sub-elements of a device. Component contains information and data defining the sub-element’s operational state, the environment in which it is functioning, and its health or status. This information and data is defined as DataItems and will be discussed in Section 4 of this document. The Agent delivers DataItems associated with each component to an application. The actual values for those data items are delivered in Streams and will be discussed in Part 3 of the standard on *Streams, Samples, and Events*.

*Figure 1 Example of Devices Structure* further defines the relationship between *Device*, *Components*, and *Sub-Components*. Sub-Components are Components that function together to perform a like task – they are elements of a parent Component. Example: Linear Axis X, Linear Axis Y, Linear Axis Z, and Rotary Axis C are each a Component. They are also Sub-Components of the parent Component Axes.

## Component Schema

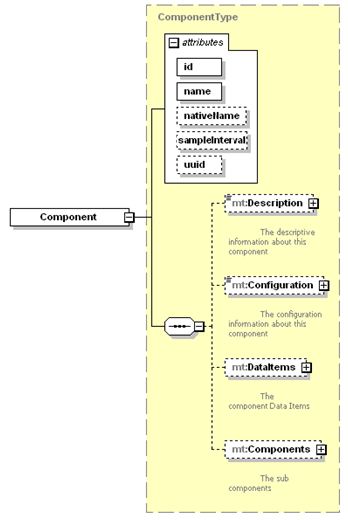


Figure : Component Schema

### Component Attributes

Every component has the following composition:

| **Attribute** | **Description** | **Occurrence** |
| --- | --- | --- |
| uuid | A unique identifier that will only refer to this component. For example, this can be the manufacturer code and the serial number. The uuid should be alphanumeric and not exceeding 255 characters. An NMTOKEN XML type. | 0..1\* |
| name | The name of the component. This name should be unique within the machine to allow for easier data integration. An NMTOKEN XML type. | 1 |
| nativeName | The name the device manufacturer assigned to the component. If the native name is not provided it **SHALL** be the name. | 0..1 |
| id | The unique identifier for this component in the document. An id SHALL be unique across all the id attributes in the document. An XML ID-type. | 1 |
| sampleRate | **DEPRECATED IN REL. 1.2 (REPLACED BY sampleInterval)** |  |
| sampleInterval | The interval in milliseconds between the completion of the reading of one sample of data from a component until the beginning of the next sampling of that data. This is the number of milliseconds between data captures. If the sample interval is smaller than one millisecond, the number can be represented as a floating point number. For example, an interval of 100 microseconds would be 0.1. | 0..1\*\* |

Notes: \* While the uuid **SHALL** be provided for the Device, it is optional for all Components. \*\* The sampleInterval is used to aid the application in interpolating values. This is the desired sample Interval and may vary depending on the capabilities of the component.

### Component Elements

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Description | An element that can contain any descriptive content. This can contain information about the component and manufacturer specific details. | 0..1 |
| Components | Sub-components of this component. | 0..INF\* |
| Configuration | An element that can contain descriptive content defining the configuration information for a Component. | 0..1 |
| DataItems | The data items this component provides. The data items are descriptions of the data events for reporting. | 0..INF\* |

Notes: \*At least one of Components or DataItems **SHALL** be provided.

#### Component Description

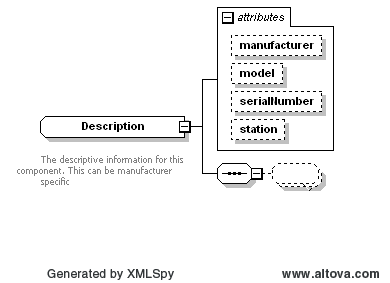


Figure : Component Schema

| **Attribute** | **Description** | **Occurrence** |
| --- | --- | --- |
| manufacturer | The name of the manufacturer of the component | 0..1 |
| model | The model description of the component | 0..1 |
| serialNumber | The component’s serial number | 0..1 |
| station | The station where the component is located when a component is part of a manufacturing unit or cell with multiple stations that share the same physical controller. | 0..1 |

The CDATA of the Description is any additional descriptive information the implementer chooses to include regarding the component. An example of a description is as follows:

<Description manufacturer="Example Co" serialNumber="A124FFF" station="2">

Example Co Simulated Vertical 3 Axis Machining center.

</Description>

The information can be provided for any component, for example an electrical power sensor, can be defined as follows:

<Description manufacturer="Example Co"   
 serialNumber="EXCO-TT-099PP-XXXX">

Advanced Pulse watt-hour transducer with pulse output.

</Description>

#### Component Components

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Component | One or more sub- components. This can also include the subtypes of Component like Axes, Linear, Path, etc... | 1..INF |

#### Component DataItems

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| DataItem | Only elements of types DataItem can be specified | 1..INF |

#### Component Configuration

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Configuration | An element that can contain descriptive content defining the configuration information for a Component. Not all Components support Configuration. When Configuration is supported, details on the schema for Configuration will be included in the applicable sections of the MTConnect standard. | 1..INF |

Configuration data is structured in the MTConnect schema as shown below:

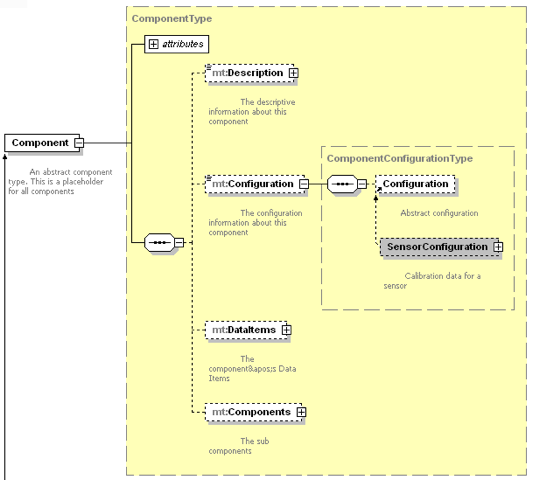


Figure 5: Component Configuration Schema

## Types of Components and Sub-Components

A component is an abstract type that allows for extensibility. As the specification progresses, more component types will be added to support new devices and parts of new devices. Some examples of components are Axes, Controller, and Path. Any of these components can have data items and sub-components. Appendix B contains reference models for common equipment to guide developers in implementing MTConnect on their devices.

### Axes

Axes is the root of all device components that have linear or rotational motion; currently there are only Linear and Rotary axes supported and when axes are defined the Axes component **SHALL** contain at least one Linear or Rotary axis. The Linear axes **SHALL** be named X, Y, Z with numbers appended for additional axes in the same plane, for example X2, Y2, and Z2 are the secondary axes to X, Y, and Z. Rotary axes **SHALL** be named A, B, and C and rotate around the X, Y, and Z axes respectively. As with the Linear axes, a number **SHALL** be appended for additional axes in the same plane.

The Axes represent the physical data for the axis components. When data is defined specifically for the physical axes, positions **SHALL** be given in MACHINE coordinates. The WORK coordinates are represented in the Path component of the Controller.

DEPRECATION WARNING: In version 1.1 of the MTConnect® standard, the Spindle component was no longer supported. The Spindle will now be represented by a rotary axis that has a RotaryMode of SPINDLE. The S(n) axis nomenclature **SHOULD** be removed and replaced with A, B, or C to clearly identify which primary plane the rotary axis is rotating around. All data items **SHOULD** now be named accordingly.

*Note:* The convention for multiple linear and rotary axes having the same designation is to index the axes letter with a number. For this standard, the secondary axis number starts at 2 (i.e. X, X2, X3, … or C, C2, C3, C4, …). This is in compliance with the ISO-841-2001. Please refer to that specification for more details.



Figure : Axes Example With Three Linear Axes and one Rotary Axis

**Linear** A linear axis represents the movement of a physical device, or a portion of a device, in a straight line. Movement may be in either a positive or negative direction.

**Rotary** An axis whose function is to provide rotary motion may function as a continuous rotation (i.e. spindle mode), continuous-path contour cutting in a rotary motion (i.e. contouring), or repositioning (i.e. indexing) different faces of the part. As such, a rotary axis **SHALL** operate in one of the three following modes: SPINDLE, INDEX, or CONTOUR.

### Controller

The Controller component represents an intelligent device. Examples include a CNC (Computer Numerical Control) or PAC (Programmable Automation Control) which may be referred to as a *Motion Control* or *General Purpose Motion Control*. The Control provides information regarding the execution of a control program and the execution state of the device. There are no required sub-components of the Controller.

Note: Version 1.1.0 and later implementations **SHOULD** use a Path sub-component to represent an individual tool path and execution state (see Path). When the machine is capable of executing more than one simultaneous program, the implementation **SHALL** use the Path components.

#### Path

For more complex devices and controllers, each path will be represented by a Path sub-component. A Path represents the motion of a control point as it moves through space as controlled by a set of control instructions (i.e. vector move). The Path will encapsulate the position, feedrate, and rotation of the control point as presented by the controller. The control point is the positioning of a tool at a point in space.

If the controller is capable of running more than one task simultaneously, a Path component **SHALL** be given for each task under the Controller component.

### ~~Power~~ Deprecated in Rel. 1.1

**NOTE:** Power as an indication of availability will be changed to the data item AVAILABILITY and electrical current and power consumption will be represented by the Electric system, see *3.4.9.5* *Electric* below.

### Door

This component represents a door closure that can be opened or closed. It **SHALL** have a data item DoorState to indicate if it is opened or closed. A device may contain multiple door components.

### Actuator

An actuator is a device for moving or controlling a mechanism or system. It takes energy, usually transported by air, electric current, or liquid, and converts it into some kind of motion. An Actuator may be a Component of a Device or it may be a Sub-Component of a parent Component.

### Sensor

Sensor is an abstract type component that provides measurement data related to a device or component. A sensor may or may not be integral to a parent device or component – it can function as a device. A sensor can be external to the parent device and can be moved from one device to another. Sensors **MAY** have their own uuid so they can be tracked throughout their lifetime. Sensors that are not integral to a parent component or device **SHALL** have an uuid.

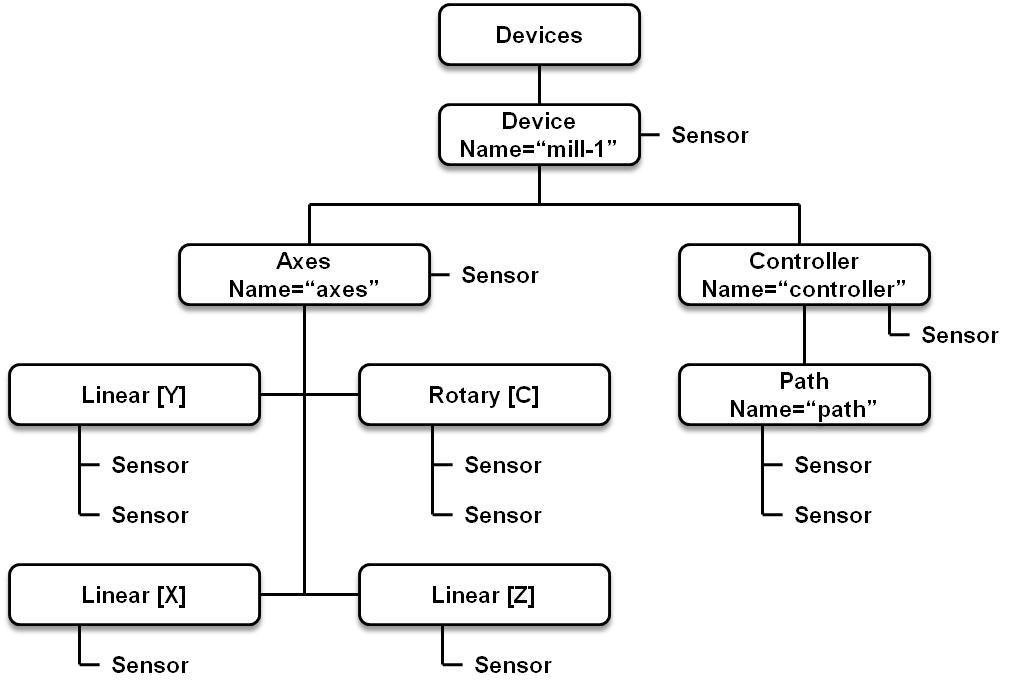


Figure : Example Sensors Structure

Sensors are typically comprised of two major components – the sensing element (provides a signal or measured value) and the sensor interface (signal processing, conversion, and communications). In MTConnect, the sensor interface is modeled as a component called Sensor. The sensing element or measured value is modeled as a DataItem. Example: A pressure transducer could be modeled as a Sensor component with a name = *Pressure Transducer B* and its measured value could be modeled as a DataItem type PRESSURE.

Sensors and sensor dataItems **SHOULD** be modeled into the MTConnect schema in such a manner that the sensor dataItem is coupled with the component or sub-component to which it is directly associated. Example: If a temperature sensor is monitoring the temperature of the machine’s X axis, it should be modeled as a dataItem TEMPERATURE for the component X AXIS.

The Sensor may provide the signal processing for multiple sensing elements. In this case, the Sensor **MAY** be modeled as an independent component or sub-component within a device. The sensor may provide additional configuration data and diagnostic information specific to the sensor interface itself or the individual sensing elements. Configuration data is provided in the Component Element Configuration(see 3.3.2.4 and sensors specific details in 3.4.7).

The following document structure can be used to define a sensor element when it is directly associated with a component or sub-component. For this example, rotary axis C (spindle) has a temperature sensor that reports the average temperature of the spindle. The temperature sensing element is located on channel 1 of the sensor interface.

<Components>

<Rotary id="c" name="C">

<DataItems>

<DataItem type="TEMPERATURE" category="SAMPLE" name="Ctemp" id="ct"

units="CELSIUS" statistic="AVERAGE">

<Source componentId="s1">channel:1</Source>

</DataItem>

</DataItems>

</Rotary>

</Components>

Additionally, the sensor interface for this sensor element can be modeled as below. In this case, calibration data for the sensor interface is provided. The sensor element above (represented as the temperature of the spindle) is defines to be connected to Channel 1 of the sensor interface and is named “A/D:1”. Additionally, the sensor interface is defined to have two internal diagnostics reporting the actual input voltage to the sensor interface (SAMPLE) and the health of that input voltage (CONDITION).

<Sensor id="sensor" name="sensor">

<Configuration>

<SensorConfiguration>

<FirmwareVersion>2.02</FirmwareVersion>

<CalibrationDate>2010-05-16</CalibrationDate>

<NextCalibrationDate>2010-05-16</NextCalibrationDate>

<CalibrationInitials>WS</CalibrationInitials>

<Channels>

<Channel number="1" name="A/D:1">

<Description>A/D With Thermister</Description>

</Channel>

</Channels>

</SensorConfiguration>

</Configuration>

<DataItems>

<DataItem category="CONDITION" id="senvc" type="VOLTAGE" />

<DataItem category="SAMPLE" id="senv" type="VOLTAGE" units="VOLT"

subType="DIRECT" />

</DataItems>

</Sensor>

### Sensor Configuration

Configuration information is provided for Sensors. The sensor configuration data provides information required maintenance and support of the sensor devices. It also provides configuration data for each sensing element, as required.

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Configuration | An element that can contain descriptive content defining the configuration information for a Sensors. For Sensors, the valid configuration is SensorConfiguration. SensorConfiguration provides data items from a subset of items commonly found in transducer electronic data sheet for sensors and actuators called TEDS. TEDS formats are defined in IEEE 1451.0 and 1451.4 transducer interface standards (ref 15 and 16, respectively). MTConnect does not support all of the data represented in the TEDS data, nor does it duplicate the function of the TEDS data sheets. | 0..1 |

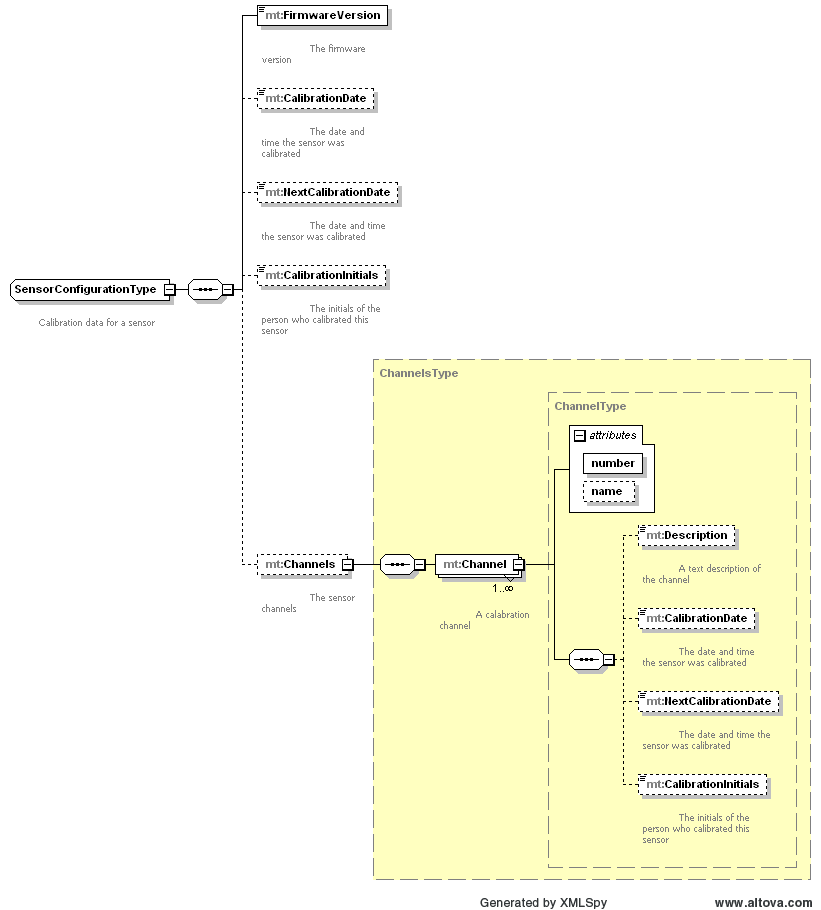


Figure 7: **Example Sensors Structure**

#### SensorConfiguration Elements

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| FirmwareVersion | Version number for the Sensor as specified by the manufacturer | 1 |
| CalibrationDate | Date upon which the Sensor was last calibrated. Dates **SHALL** be represented in the W3C ISO 8601 format | 0..1 |
| NextCalibrationDate | Date upon which the Sensor is next scheduled to be calibrated. Dates **SHALL** be represented in the W3C ISO 8601 format | 0..1 |
| CalibrationInitials | The initials of the person verifying the validity of the calibration data | 0..1 |
| Channels | A sensor can be comprised of multiple sensing elements. Each sensing element represents a Channel for the Sensor. | 0..1 |

#### Sensor Channel Attributes

Channel represents each sensing element connected to a sensor interface. Each sensor channel has the following composition:

| **Attribute** | **Description** | **Occurrence** |
| --- | --- | --- |
| Number | A unique identifier that will only refer to this component. For example, this can be the manufacturer code and the serial number. The uuid should be alphanumeric and not exceeding 255 characters. An NMTOKEN XML type. | 1 |
| Name | The name of the component. This name should be unique within the machine to allow for easier data integration. An NMTOKEN XML type. | 0..1 |

#### Sensor Channel Elements

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Description | An element that can contain any descriptive content. This can contain information about the sensor and manufacturer specific details. | 0..1 |
| CalibrationDate | Date upon which the Sensor was last calibrated. Dates **SHALL** be represented in the W3C ISO 8601 format | 0..1 |
| NextCalibrationDate | Date upon which the Sensor is next scheduled to be calibrated. Dates **SHALL** be represented in the W3C ISO 8601 format | 0..1 |
| CalibrationInitials | The initials of the person verifying the validity of the calibration data | 0..1 |

### Sensor Types

Types of measurements provided by Sensors include:

#### Thermostat (DEPRECATED in Rel. 1.2. See TEMPERATURE)

#### Vibration (DEPRECATED in Rel. 1.2. See DISPLACEMENT, FREQUENCY, etc.)

#### Acceleration

The measurement of the linear acceleration of an object.

#### Angular Acceleration

The measurement of the acceleration of a rotating object

#### Angular Velocity

The measurement of the velocity of a rotating object

#### Amperage

The measurement of electrical current flow.

#### Angle

The measurement of angular position of an object.

#### Concentration

The measurement of how much of a substance is mixed with another substance.

#### Conductivity

The measurement of the ability of a material to conduct electricity.

#### Direction

The direction of movement of an object. Normally, this will be reported as clockwise and counter clockwise for rotary motions and positive or negative for linear motions.

#### Displacement

The measurement of the distance of movement of an object.

#### Electrical Energy

The measurement of electrical energy consumed by a component over a period of time.

#### Flow

The measurement of the rate at which a volume of a fluid moves within a system.

#### Frequency

The measurement of the number of occurrences of a repeating event per unit time.

#### Fill Level

The measurement of the amount of a substance remaining compared to the planned maximum of that substance.

#### Linear Force

The magnitude of push or pull introduced by an actuator or exerted on an object.

#### Load

The measurement of the percentage of the standard rating of a device.

#### Mass

The measurement of the mass of an object(s) or an amount of material.

#### PH

The measurement of the acidity or alkalinity.

#### Pressure

The measurement of the force per unit area exerted by a gas or liquid.

#### Position

The measurement of an object’s position relative to a coordinate system.

#### Power Factor

The measurement of the ratio of real power flowing to a load to the apparent power in an AC circuit.

#### Resistance

The measurement of the degree to which an object opposes an electric current through it

#### Rotary Velocity

The measurement of the rotational speed of a rotating object.

#### Sound Level

The measurement of sound level.

#### Strain

The measurement of the amount of deformation per unit length of an object.

#### Temperature

The measurement of the temperature of an object.

#### Time

The measurement of time: may be reported as accumulated time associated with a component or clock time.

#### Tilt

The measurement of the angular displacement of an object.

#### Torque

The measurement of torque applied to or by an object.

#### Volt Ampere (VA)

The measure of the [apparent power](http://en.wikipedia.org/wiki/Apparent_power) in an [electrical circuit](http://en.wikipedia.org/wiki/Electrical_circuit), equal to the product of [root-mean-square](http://en.wikipedia.org/wiki/Root-mean-square) (RMS) [voltage](http://en.wikipedia.org/wiki/Voltage) and RMS [current](http://en.wikipedia.org/wiki/Electrical_current).

#### Volt Ampere Reactive (var)

The measurement of [reactive power](http://en.wikipedia.org/wiki/Reactive_power) in an AC electric power system

#### Velocity

The measurement of the linear velocity of an object.

#### Viscosity

The measurement of a fluid's resistance to flow.

#### Voltage

The measurement of electrical potential between two points

#### Wattage

The measurement of power consumed or dissipated by an electrical circuit or device

### Systems

A component similar to axes that groups sub-components that comprise complex parts that are not easily deconstructed. The systems will be used to represent general information about the health and viability of all of its parts and sub-parts.

#### Hydraulic

A hydraulic system comprises all the parts involved in moving and distributing pressurized liquid for the purpose of delivering a source of power to specific types of actuators.

#### Pneumatic

A pneumatic system comprises all the parts involved in moving and distributing pressurized gas regardless of purpose or activity.

#### Coolant

The coolant system comprises all the parts involved in distribution and management of coolants.

#### Lubrication

The lubrication system comprises all the parts involved in distribution and management of the lubricants.

#### Electric

The electric system represents the main power supply or generator for the device. The electric system will provide all the data with regard to current, voltage, and frequency that applies overall to the *Device*. Data regarding electric power that is specific to a component or sub-component will be reported as a DataItem of that component.

# Data Items

A DataItem describes a piece of information that can be collected from a component or device. Many DataItems **MAY** report both a numeric value (a numeric quantity reported as either a Sample or Event category) and a health status (reported as a Condition category).

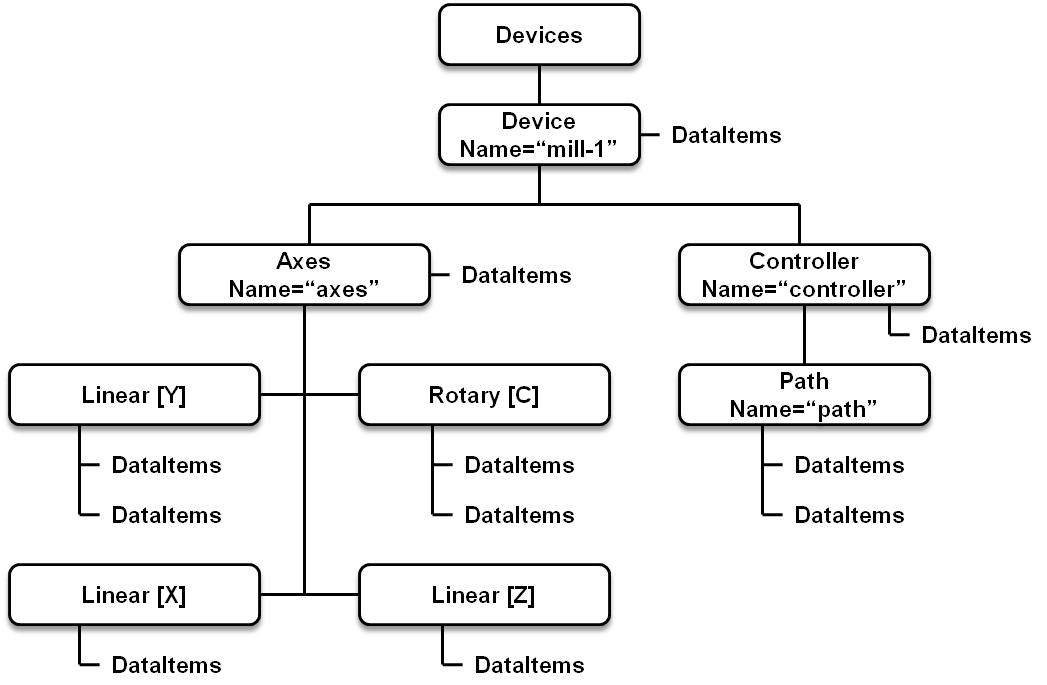


Figure : Example DataItem Structure

## DataItem Schema

The data item **SHALL** specify the type of data being collected, the id of the data item, and the category of the item. Since many data item types provide both a value (reported as either a Sample or Event category) and a health status (reported as a Condition category), each data item **SHALL** report a category for each data type to aid the application in determining the specific meaning of the data. The data item **MAY** specify a Source sub-element to provide the native name for the data feed.

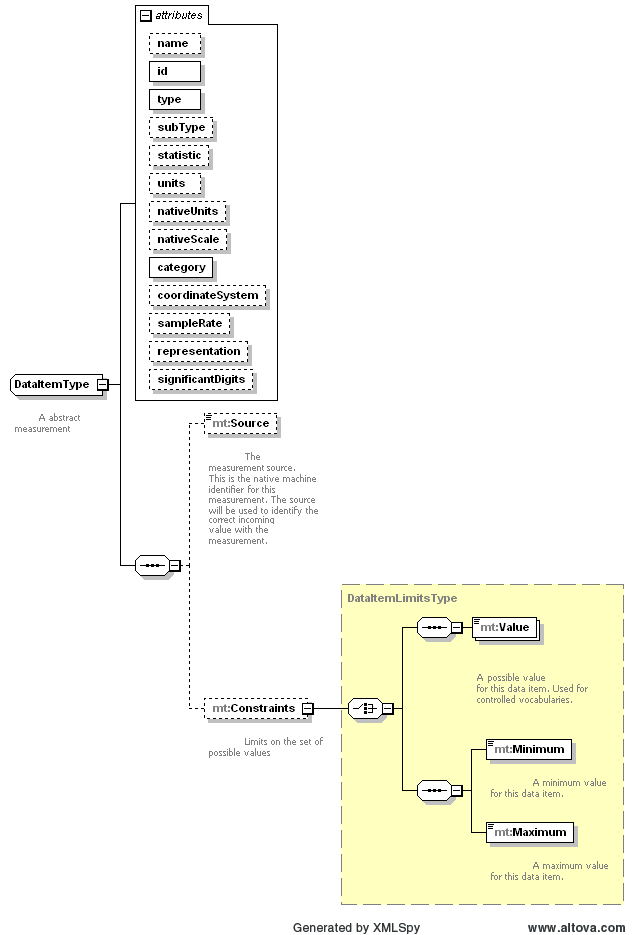


Figure : DataItem Schema Diagram

A DataItem **MAY** also specify the subtype to further qualify the type of data being requested. Subtypes are required for certain data items. For example, the POSITION has two subtypes: ACTUAL and COMMANDED. These are two separate data items that can be reported independently. See section 4.2.1, 4.2.2, and 4.2.3 for a complete list of type/subtype relations.

The units **SHALL** be specified for any data item with category Sample. The nativeUnits **MAY** be specified if they apply to the type of data and if they differ from the units. The Agent is responsible for converting the nativeUnits to the units before sending them to the applications. In addition, nativeUnits **MAY** be scaled using the nativeScale attribute; for example, if the device measures velocity in 100 ft/min, MTConnect® would represent it with the following attributes: nativeUnits=“FEET/MINUTE” and nativeScale=“100”.

For information on the transformation between DataItem name as returned in a Probe Request and the corresponding data returned in a Stream element, see Part 3, Section 3.5.

### Data Item Attributes

| **Attribute** | **Description** | | **Occurrence** |
| --- | --- | --- | --- |
| id | The unique identifier for this data item. The id attribute SHALL be unique across the entire document including the ids for components. An XML ID-type. | | 1 |
| name | The name of the data item. A name is provided as an additional human readable identifier for this data item in addition to the id. It is not required and will be implementation dependent. The identity of this data item is the type and sub-type. An NMTOKEN XML type. | | 0..1 |
| type | The type of data being measured. Examples of types are POSITION, VELOCITY, ANGLE, BLOCK, ROTARY\_VELOCITY, etc. | | 1 |
| subType | A sub-categorization of the data item type. For example, the subtypes of POSITION are ACTUAL and COMMANDED. Not all types have subtypes and this can be left off. | | 0..1 |
| category | This is how the meaning of the data item will be determined. The available options are SAMPLE, EVENT, or CONDITION. | | 1 |
| statistic | Data calculated specific to a DataItem. Examples of statistics are AVERAGE, MINIMUM, MAXIMUM, ROOT\_MEAN\_SQUARE, RANGE, MEDIAN, MODE, AND STANDARD\_DEVIATION. | | 0..1 |
| representation | Data consisting of multiple data points or samples or a file presented as a single DataItem. Each representation will have a unique format defined for each representation Examples of representations are VALUE, TIME\_SERIES, MP3, WAV, etc. Initially, the representation for TIME\_SERIES and VALUE are defined. If a representation is not specified, it **SHALL** be determined to be VALUE. | | 0..1 |
| nativeUnits | The native units used by the component. These units will be converted before they are delivered to the application. | | 0..1 |
| units | Units SHALL be present for all samples. If the data represented by a data item is a numeric value, except for line number and count, the units SHALL be specified. | | 0..1 |
| nativeScale | The multiplier for the native units. The received data **MAY** be divided by this value before conversion. If provided, the value **SHALL** be numeric. | | 0..1 |
| significantDigits | The number of significant digits in the reported value. This is used by applications to determine accuracy of values. This **SHOULD** be specified for all numeric values. | | 0..1 |
| sampleRate | The rate at which successive samples of a dataItem are recorded. Sample rate is expressed in terms of samples per second. If the sample rate is smaller than one, the number can be represented as a floating point number. For example, a rate 1 per 10 seconds would be 0.1 | | 0..1\*\* |
| coordinateSystem | | The coordinate system being used. The available values for coordinateSystem is WORK and MACHINE | 0..1 |

### Data Item Elements

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Source | Source is an optional element that contains the long name of the data item if it is too complex for the name attribute. For example, if we want to name the data item Xact, but the axis position is delivered from the device as Axis.channel.0.position, Source is used to provide the mapping. If the source is not specified, it will be assumed to be the same as the name. | 0..1 |
| Constraints | The set of possible values this data item can be assigned. This provides a way to specify the capabilities for this component by limiting the choices. For example, for ROTARY\_MODE the axis can be limited to SPINDLE for an axis that can only spin. | 0..1 |

#### Constraints Elements



Figure : Constraints Schema

| **Element** | **Description** | **Occurrence** |
| --- | --- | --- |
| Value | A constraint on the possible values for this data item. If there is only one value listed here, the value data item will be constant. In the case of a constant data item, the value is not required to be supplied in the streams document. | 0..INF |
| Maximum | The maximum value for this data item. This will be the bounded upper range. This will only be relevant when the data item has a numeric type. | 0..1 |
| Minimum | The minimum value for this data item. This will be the bounded lower range. This will only be relevant when the data item has a numeric type. | 0..1 |

### Data Item attribute: category

MTConnect® provides three different categories of data items, SAMPLE, EVENT, and CONDITION. The category will indicate where the results will be reported in the XML Document as a response to a sample or current request. See Part 3 Section 3 on *Streams, Samples, and Events* for more information.

**SAMPLE** A Sample is the reading of the value of a continuously variable or analog data item. A continuous value can be sampled at any point-in-time and will always produce a result. An example of a continuous data item is the Linear X axis position. Sample data items that are continuous are always scalar floating point or integers that can have an infinite number of possible values. This is different from state or discrete data items that have a limited number of possible values. Samples **SHALL** have units.

**EVENT** An Event comprises discrete information from the device. There are two types of events: those representing state, with two or more discrete values, and those representing messages that contain plain text data. An example of a state event is a DoorStatus that can be either OPEN, UNLATCHED, or CLOSED. An example of a message is a PROGRAM that can be any valid string of characters. Events do not have intermediate values that vary over time, as do Samples. Events can be thought of as streaming information that if taken at any point in time represents the current state of the device.

**CONDITION** A data item that communicates the device’s health and ability to function. A condition can be one of Unavailable, Normal, Warning, or Fault and there can be multiple active conditions at one time; whereas a sample or event can only have a single value at one point in time.

### Data Item attribute: coordinateSystem

A data item can specify an optional coordinate system that is being used. If not specified, the Axes coordinates **SHALL** be MACHINE and the Path coordinates **SHALL** be WORK. The possible values of coordinates are:

**MACHINE** An unchangeable coordinate system that has machine zero as its origin.

**WORK** The coordinate system that represents the working area for a particular workpiece whose origin is shifted within the MACHINE coordinate system. If the WORK coordinates are not currently defined in the device, the MACHINE coordinates will be used.

### Data Item attribute: units

| **Units** | **Description** |
| --- | --- |
| AMPERE | Amps |
| CELSIUS | Degrees Celsius |
| COUNT | A counted event |
| DECIBEL | Sound Level |
| DEGREE | Angle in degrees |
| DEGREE/SECOND | Angular degrees per second |
| DEGREE/SECOND^2 | Angular acceleration in degrees per second squared |
| HERTZ | Frequency measured in cycles per second |
| JOULE | A measurement of energy. |
| KILOGRAM | Kilograms |
| LITER | Liters |
| LITER/SECOND | Liters per second |
| MICRO\_RADIAN | Measurement of Tilt |
| MILLIMETER | Millimeters |
| MILLIMETER/SECOND | Millimeters per second |
| MILLIMETER/SECOND^2 | Acceleration in millimeters per second squared |
| MILLIMETER\_3D | A point in space identified by X, Y, and Z positions and represented by a space delimited set of numbers each expressed in millimeters. |
| NEWTON | Force in Newtons |
| NEWTON\_METER | Torque, a unit for force times distance. |
| OHM | Measure of Electrical Resistance |
| PASCAL | Pressure in Newtons per square meter |
| PASCAL\_SECOND | Measurement of Viscosity |
| PERCENT | Percentage |
| PH | A measure of the acidity or alkalinity of a solution |
| REVOLUTION/MINUTE | Revolutions per minute |
| SECOND | A measurement of time. |
| SIEMENS/METER | A measurement of Electrical Conductivity |
| VOLT | Volts |
| VOLT\_AMPERE | Volt-Ampere (VA) |
| VOLT\_AMPERE\_REACTIVE | Volt-Ampere Reactive (var) |
| WATT | Watts |
| WATT\_SECOND | Measurement of electrical energy, equal to one Joule |

### Data Item attribute: statistic

The statistic attribute provides additional values that may be reported for any DataItem. These values are calculated values generated by the component or device providing additional data regarding a DataItem sampled over a specified period of time. All statistic data is reported in the standard units of the DataItem.

Statistic values are periodically reset. When statistic values are reported as a *Streams* value, the statistic value **SHALL** include an attribute Duration. Duration defines the time elapsed since the statistic calculation was last reset.

| **Statistic** | **Description** |
| --- | --- |
| AVERAGE | Mathematical Average value calculated for the DataItem during the calculation period |
| KURTOSIS | A measure of the “peakedness” of a probability distribution; i.e., the shape of the distribution curve |
| MAXIMUM | Maximum or peak value recorded for the DataItem during the calculation period |
| MEDIAN | The middle number of a series of numbers |
| MINIMUM | Minimum value recorded for the DataItem during the calculation period |
| MODE | The number in a series of numbers that occurs most often |
| RANGE | Difference between the Maximum value and Minimum value of a DataItem during the calculation period. Also represents Peak-to-Peak measurement in an waveform. |
| ROOT\_MEAN\_SQUARE | Mathematical Root Mean Value (RMS) value calculated for the DataItem during the calculation period |
| STANDARD\_DEVIATION | Statistical Standard Deviation value calculated for the DataItem during the calculation period |

### Data Item attribute: representation

The representation attribute defines the format for data consisting of multiple data points or samples or a file presented as a single DataItem. Each representation will have a unique format defined for each representation. At this time, the only representations defined are TIME\_SERIES and VALUE.

Details on the structure and format of each representation is provided in Part 3, Section 3.8.3.

| **Representation** | **Description** |
| --- | --- |
| VALUE | The measured value of a Sample. If no representation is specified for a data item, the representation **SHALL** be determined to be VALUE. |
| TIME\_SERIES | A series of sampled data. The data is collected for a specified number of samples and each sample is collected with a fixed period |

### Data Item Attribute: nativeUnits

The nativeUnits attribute adds additional values to the units values. This is the list currently supported by MTConnect® and the MTConnect® schema.

| **Native Units** | **Description** |
| --- | --- |
| CENTIPOISE | A measure of Viscosity |
| DEGREE/MINUTE | Rotational velocity in degrees per minute |
| FAHRENHEIT | Temperature in Fahrenheit |
| FOOT | Feet |
| FOOT/MINUTE | Feet per minute |
| FOOT/SECOND | Feet per second |
| FOOT/SECOND^2 | Acceleration in feet per second squared |
| FOOT\_3D | A point in space identified by X, Y, and Z positions and represented by a space delimited set of numbers each expressed in feet. |
| GALLON/MINUTE | Gallons per minute. |
| INCH | Inches |
| INCH/MINUTE | Inches per minute |
| INCH/SECOND | Inches per second |
| INCH/SECOND^2 | Acceleration in inches per second squared |
| INCH\_3D | A point in space identified by X, Y, and Z positions and represented by a space delimited set of numbers each expressed in inches. |
| INCH\_POUND | A measure of torque in inch pounds. |
| KILOWATT | A measurement in kilowatt. |
| KILOWATT\_HOUR | Kilowatt hours which is 3.6 mega joules. |
| LITER | Measurement of volume of a fluid |
| LITER/MINUTE | Measurement of rate of flow of a fluid |
| MILLIMETER/MINUTE | Velocity in millimeters per minute |
| POUND | US pounds |
| POUND/INCH^2 | Pressure in pounds per square inch (PSI). |
| RADIAN | Angle in radians |
| RADIAN/SECOND | Velocity in radians per second |
| RADIAN/SECOND^2 | Rotational acceleration in radian per second squared |
| RADIAN/MINUTE | Velocity in radians per second. |
| REVOLUTION/SECOND | Rotational velocity in revolution per second |
| OTHER | Unsupported units |

## Data Item Types for SAMPLE Category

The types are given in **bold** and the subtypes are indented and in plain text.

| **Data Item type/subtype** | **Description** | **Units** |
| --- | --- | --- |
| **ACCELERATION** | Rate of change of velocity | MILLIMETER/SECOND^2 |
| **ACCUMULATED\_TIME** | The measurement of accumulated time associated with a component | SECOND |
| **ANGULAR\_ACCELERATION** | Rate of change of angular velocity. | DEGREE/SECOND^2 |
| **ANGULAR\_VELOCITY** | Rate of change of angular position. | DEGREE/SECOND |
| **AMPERAGE** | The measurement of AC Current or a DC current | AMPERE |
| * + ALTERNATING | The measurement of alternating current. If not specified further in Statistic, defaults to RMS Current | AMPERE |
| * + DIRECT | The measurement of DC current | AMPERE |
| **ANGLE** | The angular position of a component relative to the parent. | DEGREE |
| * + ACTUAL | The angular position as read from the physical component. | DEGREE |
| * + COMMANDED | The angular position computed by the controller. | DEGREE |
| **AXIS\_FEEDRATE** | The feedrate of a linear axis. | MILLIMETER/SECOND |
| * + ACTUAL | The actual federate of a linear axis. | MILLIMETER/SECOND |
| * + COMMANDED | The feedrate as specified in the program. | MILLIMETER/SECOND |
| * + OVERRIDE | The operator’s overridden value. Percent of commanded. | PERCENT |
| **CLOCK\_TIME** | The reading of a timing device at a specific point in time. Clock time **SHALL** be reported in W3C ISO 8601 format. | YYYY-MM-DDThh:mm:ss.ffff |
| **CONCENTRATION** | Percentage of one component within a mixture of components | PERCENT |
| **CONDUCTIVITY** | The ability of a material to conduct electricity | SIEMENS/METER |
| **DISPLACEMENT** | The displacement as the change in position of an object | MILLIMETER |
| **ELECTRICAL\_ENERGY** | The measurement of electrical energy consumption by a component | WATT\_SECOND |
| **FILL\_LEVEL** | The measurement of the amount of a substance remaining compared to the planned maximum amount of that substance | PERCENT |
| **FLOW** | The rate of flow of a fluid | LITER/SECOND |
| **FREQUENCY** | The measurement of the number of occurrences of a repeating event per unit time | HERTZ |
| **~~GLOBAL\_POSITION~~** | Deprecated in Rel. 1.1 |  |
| **~~LEVEL~~** | Deprecated in Rel. 1.2 See Fill\_Level |  |
| **LINEAR\_FORCE** | The measure of the push or pull introduced by an actuator or exerted on an object | NEWTON |
| **LOAD** | The measurement of the percentage of the standard rating of a device | PERCENT |
| **MASS** | The measurement of the mass of an object(s) or an amount of material | KILOGRAM |
| **PATH\_FEEDRATE** | The feedrate of the tool path. | MILLIMETER/SECOND |
| * + ACTUAL | The three-dimensional feedrate derived from all components. | MILLIMETER/SECOND |
| * + COMMANDED | The feedrate as specified in the program | MILLIMETER/SECOND |
| * + OVERRIDE | The operator’s overridden value. Percent of commanded. | PERCENT |
| **PATH\_POSITION** | The current program control point or program coordinate in WORK coordinates. The coordinate system will revert to MACHINE coordinates if WORK coordinates are not available. | MILLIMETER\_3D |
| * + ACTUAL | The position of the component as read from the device. | MILLIMETER\_3D |
| * + COMMANDED | The position computed by the controller. | MILLIMETER\_3D |
| * + TARGET | The target position for the movement. | MILLIMETER\_3D |
| * + PROBE | The position provided by a probe | MILLIMETER\_3D |
| **PH** | The measure of the acidity or alkalinity. | PH |
| **POSITION** | The position of the component. Defaults to MACHINE coordinates. | MILLIMETER |
| * + ACTUAL | The position of the component as read from the device. | MILLIMETER |
| * + COMMANDED | The position as given by the Controller. | MILLIMETER |
| * + TARGET | The target position for the movement. | MILLIMETER |
| **POWER\_FACTOR** | The measurement of the ratio of real power flowing to a load to the apparent power in that AC circuit. | PERCENT |
| **PRESSURE** | The force per unit area exerted by a gas or liquid | PASCAL |
| **RESISTANCE** | The measurement of the degree to which an object opposes an electric current through it | OHM |
| **ROTARY\_VELOCITY** | The rotational speed of a rotary axis. | REVOLUTION/MINUTE |
| * + ACTUAL | The rotational speed the rotary axis is spinning at. ROTARY\_MODE SHALL be SPINDLE. | REVOLUTION/MINUTE |
| * + COMMANDED | The rotational speed as specified in the program. | REVOLUTION/MINUTE |
| * + OVERRIDE | The operator’s overridden value. Percent of commanded. | PERCENT |
| **SOUND\_LEVEL** | Measurement of a sound level or sound pressure level relative to atmospheric pressure | DECIBEL |
| * + NO\_SCALE | No weighting factor on the frequency scale | DECIBEL |
| * + A\_SCALE | A Scale weighting factor. This is the default weighting factor if no factor is specified | DECIBEL |
| * + B\_SCALE | B Scale weighting factor | DECIBEL |
| * + C\_SCALE | C Scale weighting factor | DECIBEL |
| * + D\_SCALE | D Scale weighting factor | DECIBEL |
| **SPINDLE\_SPEED** | **DEPRECATED in REL 1.2.** Replaced by ROTARY\_VELOCITY |  |
| * + ACTUAL | The rotational speed of a rotary axis. ROTARY\_MODE SHALL be SPINDLE. | REVOLUTION/MINUTE |
| * + COMMANDED | The rotational speed the as specified in the program. | REVOLUTION/MINUTE |
| * + OVERRIDE | The operator’s overridden value. Percent of commanded. | PERCENT |
| **STRAIN** | Strain is the amount of deformation per unit length of an object when a load is applied. | PERCENT |
| **TEMPERATURE** | The measurement of temperature | CELSIUS |
| **TILT** | A measurement of angular displacement | MICRO\_RADIAN |
| **TORQUE** | The turning force exerted on an object or by an object | NEWTON\_METER |
| **VOLT\_AMPERE** | The measure of the [apparent power](http://en.wikipedia.org/wiki/Apparent_power) in an [electrical circuit](http://en.wikipedia.org/wiki/Electrical_circuit), equal to the product of [root-mean-square](http://en.wikipedia.org/wiki/Root-mean-square) (RMS) [voltage](http://en.wikipedia.org/wiki/Voltage) and RMS [current](http://en.wikipedia.org/wiki/Electrical_current)’ (commonly referred to as VA) | VOLT\_AMPERE |
| **VOLT\_AMPERE\_REACTIVE** | The measurement of [reactive power](http://en.wikipedia.org/wiki/Reactive_power) in an AC electrical circuit (commonly referred to as var) | VOLT\_AMPERE\_REACTIVE |
| **VELOCITY** | The rate of change of position. | MILLIMETER/SECOND |
| **VISCOSITY** | A measurement of a fluid’s resistance to flow | PASCAL\_SECOND |
| **VOLTAGE** | The measurement of electrical potential between two points | VOLT |
| * + ALTERNATING | The measurement of alternating voltage. If not specified further in Statistic, defaults to RMS voltage | VOLT |
| * + DIRECT | The measurement of DC voltage | VOLT |
| **WATTAGE** | The measurement of power consumed or dissipated by an electrical circuit or device | WATT |

## Data Item Types for EVENT Category

Note: The Event does not have any units since these values are not scalars.

| **Data Item type/subtype** | **Description** |
| --- | --- |
| **ACTUATOR\_STATE** | The state of the actuator. ACTIVE or INACTIVE. |
| **ALARM** | DEPRECATED: Replaced with CONDITION category. Rel. 1.1. |
| **ACTIVE\_AXES** | The set of axes associated with a path that the controller is controlling. If this data item is not provided, it will be assumed the controller is controlling all axes. |
| **AVAILABILITY** | Represents the components ability to communicate its availability. This **SHALL** be  provided for the device and **MAY** be provided for all other components |
| **AXIS\_COUPLING** | Describes the way the axes will be associated to each other. This is used in conjunction with COUPLED\_AXES to indicate the way they are interacting. The possible values are: TANDEM, SYNCHRONOUS, MASTER, and SLAVE. The coupling **SHALL** be viewed from the perspective of the axis, therefore a MASTER coupling indicates that this axis is the  master of the COUPLED\_AXES. |
| **BLOCK** | The block of code being executed. The block contains the entire expression of the step in the program. |
| **CODE** | DEPRECATED. Rel 1.1.0 |
| **CONTROLLER\_MODE** | The current controller’s mode. AUTOMATIC, MANUAL, MANUAL\_DATA\_INPUT, FEED\_HOLD, or SEMI\_AUTOMATIC. |
| **COUPLED\_AXES** | Refers to the set of associated axes. The value will be a space delimited set of axes names. |
| **DIRECTION** | The direction of motion. CLOCKWISE or COUNTER\_CLOCKWISE |
| ROTARY | The rotational direction of a rotary device. CLOCKWISE or COUNTER\_CLOCKWISE |
| LINEAR | The direction of motion of a linear device. POSTIVE or NEGATIVE |
| **DOOR\_STATE** | The opened or closed state of the door. OPEN, UNLATCHED, or CLOSED. |
| **EMERGENCY\_STOP** | The current state of the emergency stop actuator. ARMED (the circuit is complete and the device is operating) or TRIGGERED (the circuit is open and the device SHALL cease operation). |
| **EXECUTION** | The execution status of the Controller. READY, ACTIVE, INTERRUPTED, or STOPPED |
| **LINE** | The current line of code being executed |
| MAXIMUM | The maximum line number of the code being executed. |
| MINIMUM | The minimum line number of the code being executed. |
| **MESSAGE** | An uninterpreted textual notification. |
| **PART\_COUNT** | The current count of parts produced as represented by the controller. SHALL be an integer value. |
| ALL | The count of all the parts produced. If the subtype is not given, this is the default. |
| GOOD | Indicates the count of correct parts made. |
| BAD | Indicates the count of incorrect parts produced. |
| **PART\_ID** | An identifier of the current part in the device |
| **PATH\_MODE** | The operational mode for this Path. SYNCHRONOUS, MIRROR, or INDEPENDENT. Default value is INDEPENDENT if not specified. |
| **POWER\_STATE** | The ON or OFF status of the component. **DEPRECATION WARNING**: **MAY** be  deprecated in the future. |
| LINE | The state of the high voltage line. |
| CONTROL | The state of the low power line. |
| **POWER\_STATUS** | DEPRECATED. Rel. 1.1. |
| **PROGRAM** | The name of the program being executed |
| **ROTARY\_MODE** | The mode for the Rotary axis. SPINDLE, INDEX, or CONTOUR. |
| **~~TOOL\_ID~~** | **DEPRECATED in Rel. 1.2.** See Tool\_ASSET\_ID. ~~The identifier of the tool currently in use for a given Path~~ |
| **TOOL\_ASSET\_ID** | The identifier of the tool currently in use for a given Path |
| **WORKHOLDING\_ID** | The identifier for the workholding currently in use for a given Path |

## Data Item Types for CONDITION Category

These are items that indicate the devices' health and ability to operate. They are reported differently than Samples or Events: they **SHALL** be reported as Normal, Warning, and Fault. Unlike the other two categories, a Component or Device **MAY** have values for a Condition type DataItem that has multiple concurrently active values at any point in time. Additionally, these items **MAY** be further defined to provide differentiation for different Condition states; example an AMPERAGE *Condition* may differentiate between HIGH amperage and LOW amperage. These differences are further defined as *qualifier* in Part 3, Section 3.11

| **Data Item type/ qualifier** | **Description** |
| --- | --- |
| **ACCELERATION** | Rate of Change of Velocity |
| **ACCUMULATED\_TIME** | The measurement of accumulated time associated with a component |
| **ACTUATOR** | An actuator related condition. |
| **AMPERAGE** | A high or low condition for the electrical current. |
| **ANGLE** | The angular position of a component relative to the parent. |
| **ANGULAR-ACCELERATION** | Rate of change of angular velocity. |
| **ANGULAR\_VELOCITY** | Rate of change of angular position |
| **COMMUNICATIONS** | A communications failure indicator. |
| **CONCENTRATION** | Percentage of one component within a mixture of components |
| **CONDUCTIVITY** | The ability of a material to conduct electricity |
| **DATA\_RANGE** | Information provided is outside of expected value range |
| **DIRECTION** | The direction of motion of a component |
| **DISPLACEMENT** | The change in position of an object |
| **ELECTRICAL\_ENERGY** | The measurement of electrical energy consumption by a component |  |
| **FILL\_LEVEL** | Represents the amount of a substance remaining compared to the planned  maximum amount of that substance |
| **FLOW** | The rate of flow of a fluid |
| **FREQUENCY** | The number of occurrences of a repeating event per unit time |
| **HARDWARE** | The hardware subsystem of the component operation condition. |
| **~~LEVEL~~** | **DEPRECATED in Rel 1.2. See FILL\_LEVEL** |
| **LINEAR\_FORCE** | The measure of the push or pull introduced by an actuator or exerted by an object |
| **LOAD** | The measure of the percentage of the standard rating of a device |
| **LOGIC\_PROGRAM** | An error occurred in the logic program or PLC (programmable logic controller). |
| **MASS** | The measurement of the mass of an object(s) or an amount of material |
| **MOTION\_PROGRAM** | An error occurred in the motion program. |
| **PATH\_FEEDRATE** | The federate of the tool path |
| **PATH\_POSITION** | The current control point of the path |
| **PH** | The measure of acidity or alkalinity |
| **POSITION** | The position of a component. |
| **POWER\_FACTOR** | The ratio of real power flowing to a load to the apparent power in that AC circuit. |
| **PRESSURE** | The measurement of the force per unit area exerted by a gas or liquid. |
| **RESISTANCE** | The measurement of the degree to which an object opposes an electric current through it |
| **ROTARY\_VELOCITY** | The rotational speed of a rotary axis |
| **SOUND\_LEVEL** | The measurement of sound pressure level |
| **SPINDLE\_SPEED** | **DEPRECATED in Rel 1.2. See ROTARY\_VELOCITY** |
| **STRAIN** | Indicates the amount of deformation per unit length of an object when a load is applied |
| **SYSTEM** | A condition representing something that is not the operator, program, or  hardware. This is often used for operating system issues. |
| **TEMPERATURE** | Indicates the temperature of a component. |
| **TILT** | The measure of angular displacement |
| **TORQUE** | The measured of the turning force exerted on an object or by an object |
| **VOLT\_AMPERAGE** | The measure of the apparent power in an electrical circuit (commonly referred to as VA) |
| **VOLT\_AMPERAGE\_REACTIVE** | The measure of reactive power in an AC electrical power circuit (commonly  referred to as var). |
| **VELOCITY** | Indicated the velocity of a component. |
| **VISCOSITY** | The measure of a fluid’s resistance to flow |
| **VOLTAGE** | The measurement of electrical potential between two points |
| **WATTAGE** | The measurement of power consumed or dissipated by an electrical circuit or  device |

# Component and Data Item Relationships

This section will discuss the association between Component, DataItems, and DataItem Categories (Events, Condition, and Samples). For each component, there are a limited set of allowable sub-components and a limited set of data items. For example, an Axes component may not have a Device or a Controller as a child, and it may not have Block as a DataItem type, since it is incapable of running a program.

Many DataItems can be applied to a wide variety of components. In the sections below, only those data items that are specific to each component will be defined. By inference, all other data items may be applied to these components as required.

## Device

The Device is the only top level element in the component tree. Since an MTConnect® Agent can manage multiple devices, the schema provides a top level container Devices to hold the Device elements.

A device **SHALL** always contain an Availability data item that represents this device is functioning and able to communicate.

### Device DataItems

* EMERGENCY\_STOP - The emergency stop state of the machine or device.
* AVAILABILITY - **Required**

### Components of Device

* Axes
* Controller
* Systems
* Door
* Actuator
* Sensor

## Common Components and Related Data Items

A common set of DataItems have been defined to provide the flexibility to provide a wide variety of information about a machine or process. Any DataItem can be used with any Device or Component providing that the standard naming conventions are implemented. Any component **MAY** also include an arbitrary set of sensors as sub-components or data items. A sensor may be an external device that will collect data and report it to the Agent.

Additionally, *Conditions* are defined as a specific category of DataItem that indicates the health of a component or device. Any Condition can be used with any Device or Component providing that the standard naming conventions are implemented.

Only DataItems unique to each component are detailed below. It can be assumed that all other DataItems can be applied to any of the components.

### Axes

The Axes component is a container for the actual axes of which there are currently two types: Linear and Rotary.

#### ~~Axes DataItems~~

* ~~GLOBAL\_POSITION~~ – DEPRECATED in Rel 1.1
* ~~PATH\_FEEDRATE~~ – Moved to Path
* ~~ACCELERATION~~ – Moved to Path
* ~~VELOCITY~~ – Moved to Path

#### Sub-components of Axes

* Linear
* Rotary
* ~~Spindle~~ – DEPRECATED in Rel 1.1

### Linear (Subcomponent of Axes)

A linear axis represents travel along a straight line. The name of the linear axis **SHOULD** follow the conventions of the industry.

#### Linear Axes DataItems (Samples and Events)

* AXIS\_FEEDRATE
* DIRECTION
* POSITION
* ~~SLAVE\_OF\_AXIS~~ (DEPRECATED in Rel. 1.1)
* LINEAR\_FORCE
* VELOCITY

#### Linear Axes Conditions

* POSITION
* LINEAR\_FORCE
* VELOCITY

### Rotary (Subcomponent of Axes)

A rotary axis revolves around a line or vector.

#### Rotary Axes DataItems (Samples and Events)

* ANGLE
* ANGULAR\_ACCELERATION
* ANGULAR\_VELOCITY
* DIRECTION
* ROTARY\_MODE
* ROTARY\_VELOCITY
* ~~SLAVE\_OF\_AXIS~~ (SHOULD DEPRECATED in Rel 1.1)
* ~~SPINDLE\_SPEED~~

DEPRECATED in Rel 1.2. Replaced by ROTARY\_VELOCITY

* TORQUE

#### Rotary Axes Conditions

* ANGLE
* ANGULAR\_ACCELERATION
* ANGULAR\_VELOCITY
* ROTARY\_VELOCITY
* TORQUE

### Controller

The controller component is the component that controls a device, executes a program, and sends instructions to the other components of the machine. It is the brains of the machine and can be asked for its current execution state and program name.

#### Sub-components of Controller

* Path

#### Controller DataItems (Samples and Events)

* ~~CODE~~ **DEPRECATED in Rel 1.1**
* CONTROLLER\_MODE
* EXECUTION
* EMERGENCY\_STOP
* MESSAGE
* PART\_COUNT
* PART\_ID
* PROGRAM
* ~~TOOL\_ID~~ **Deprecated in Rel 1.2.**
* TOOL\_ASSET\_ID
* WORKHOLDING\_ID

#### Controller Conditions

* COMMUNICATIONS
* HARDWARE
* LOGIC\_PROGRAM
* MOTION\_PROGRAM
* SYSTEM

### Path (Subcomponent of Controller)

A Path represents the motion of a control point as it moves through space as controlled by a set of control instructions (i.e. vector move). When Path is not defined, path dataItems **May** be reported for the Controller.

#### Path DataItems (Samples and Events)

* ACTIVE\_AXES
* AXES\_COUPLING
* BLOCK
* ~~CODE~~ **DEPRECATED**
* COUPLED\_AXES
* CONTROLLER\_MODE
* EMERGENCY\_STOP
* EXECUTION
* LINE
* PART\_COUNT
* PART\_ID
* PATH\_FEEDRATE
* PATH\_POSITION
* PROGRAM
* ~~TOOL\_ID~~ **Deprecated in Rel 1.2.**
* TOOL\_ASSET\_ID
* VELOCITY
* WORKHOLDING\_ID

#### Path Conditions

* MOTION\_PROGRAM

### ~~Power~~ DEPRECATED in Rel 1.1

### Sensors

Sensors are components that may or may not be integral to a parent component or device. When a sensor is not integral to a parent device or component – it can function as a device. Sensor data **SHALL** be associated with its most relevant component and **SHALL** be represented as a DataItem for that component.

#### Sensor Conditions

* COMMUNICATION
* HARDWARE

### ~~Thermostat~~ Deprecated in REL 1.2 Replaced with DataItem Temperature

~~A sensor capable of measuring the temperature of a component. The temperature is always given in Celsius.~~

#### ~~DataItem types~~

* ~~TEMPERATURE~~

#### ~~Condition types~~

* ~~COMMUNICATION~~
* ~~HARDWARE~~
* ~~TEMPERATURE~~

### ~~Vibration~~ Deprecated in REL 1.2 Replaced with Data Items to measure Vibration (Displacement, Frequency, etc).

~~A sensor capable of measuring the vibration of a component.~~

#### ~~DataItem types~~

* ~~ACCELERATION~~
* ~~DISPLACEMENT~~
* ~~FREQUENCY~~
* ~~VELOCITY~~

#### ~~Condition types~~

* ~~ACCELERATION~~
* ~~COMMUNICATION~~
* ~~DISPLACEMENT~~
* ~~HARDWARE~~
* ~~VIBRATION~~

### ~~Pressure~~ Deprecated in REL 1.2 Replace with DataItem Pressure

~~A sensor capable of measuring the pressure.~~

#### ~~DataItem types~~

* ~~PRESSURE~~

#### ~~Condition types~~

* ~~COMMUNICATION~~
* ~~HARDWARE~~
* ~~PRESSURE~~

### Door

A opening that can be closed.

#### Door DataItems (Samples and Events)

* DOOR\_STATE

#### Door Conditions

* DOOR\_STATE
* COMMUNICATIONS
* HARDWARE

### Actuator

A mechanical device for moving or controlling a mechanism or system.

#### Acutator DataItems (Samples and Events)

* ACTUATOR\_STATE

#### Actuator Conditions

* COMMUNICATIONS
* HARDWARE

### ~~Spindle~~ – DEPRECATED in Rel. 1.1

~~The spindle is a rotational axis that revolves at high speed and has its speed expressed in REVOLUTION/MINUTE~~

### Systems

The systems component is a place holder for all the system types.

#### Sub-components of Systems

* Hydraulic
* Pneumatic
* Coolant
* Lubrication

### Hydraulic (Subcomponent of Systems)

A component representing the hydraulics and hydraulic distribution system of a device.

#### Hydraulic Conditions

* COMMUNICATIONS
* HARDWARE

### Coolant (Subcomponent of Systems)

A component representing the coolant and coolant distribution system of a device.

#### Coolant DataItems (Samples and Events)

* CONCENTRATION
* CONDUCTIVITY
* PH
* VISCOSITY

#### Coolant Conditions

* COMMUNICATIONS
* HARDWARE
* CONCENTRATION
* CONDUCTIVITY
* PH
* VISCOSITY

### Lubrication (Subcomponent of Systems)

A component representing the lubricant and lubrication distribution system of a device.

#### Lubrication DataItems (Samples and Events)

* PH
* VISCOSITY

#### Lubrication Conditions

* COMMUNICATIONS
* HARDWARE
* PH
* VISCOSITY

### Electric (Subcomponent of Systems)

A component representing the electrical supply for a device.

#### Electrical DataItems (Samples and Events)

* AMPERAGE
* ELECTRICAL\_ENERGY
* FREQUENCY
* POWER\_FACTOR
* VOLTAGE
* VOLT\_AMPERE
* VOLT\_AMPERE\_REACTIVE
* WATTAGE

#### Electric Conditions

* AMPERAGE
* FREQUENCY
* VOLTAGE
* WATTAGE

# Annotated XML Examples

## Simplest Device

For the simplest possible device we are modeling a saw that has only an Availability (the minimal set of data items). To retrieve this information we send the following request to the Agent:

[http://10.1.23.10/ LinuxCNC/probe](http://10.1.23.10/%20LinuxCNC/probe)

The Agent responds as follows:

1. <?xml version="1.0" encoding="UTF-8"?>
2. <MTConnectDevices xmlns:m="urn:mtconnect.com:MTConnectDevices:0.9" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:mtconnect.com:MTConnectDevices:0.9" xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:0.9 /schemas/MTConnectDevices.xsd">
3. <Header sender="10.1.23.10" bufferSize="100000"

creationTime="2008-07-07T23:07:50-07:00" version="0.9"

instanceId="1214527986"/>

Line 3 provides the instanceId as a unique number for this request. For this example, the Agent does not persist the Samples, Events, and Condition. Therefore, this number will change every time. The bufferSize indicates that this Agent is capable of storing 100,000 Samples, Events, and Conditions.

1. <Devices>
2. <Device iso841Class="6" uuid="linux-01" name="LinuxCNC"

sampleInterval="100.0" id="d">

1. <Description manufacturer="NIST" serialNumber="01"/>

The above device description includes the unique id and a sample interval of ten times per second. Since there are no telemetry data being collected, sampling at once per second is typically adequate.

1. <DataItems>
2. <DataItem type="AVAILABILITY" name="avail" category="EVENT"

id="a"/>

1. </DataItems>

As was stated previously, the device is only required to have one AVAILABILITY data item which **SHALL** report the device’s represented availability to communicate. The DataItem on line 8 has an **id** of “a”. This will allow events responding to this data item to be easily associated.

1. </Components>
2. </Device>
3. </Devices>
4. </MTConnectDevices>

Lines 11 through 14 terminate each element type and closes the document.

## More Complex Example of probe

The sample was generated with the following request:

* http://10.1.23.5/LinuxCNC/probe

The following is an example of a 3 axis mill simulation. The mill has three linear axes and one rotary axis (spindle):

1. <?xml version="1.0" encoding="UTF-8"?>
2. <MTConnectDevices xmlns:m="urn:mtconnect.com:MTConnectDevices:0.9" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:mtconnect.com:MTConnectDevices:0.9" xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:0.9 /schemas/MTConnectDevices.xsd">
3. <Header sender="10.1.23.5" bufferSize="100000" creationTime=

"2008-07-07T23:07:50-07:00" version="0.9"

instanceId="1214527986"/>

1. <Devices>
2. <Device iso841Class="6" uuid="linux-01" name="LinuxCNC"

sampleRate="100.0" id="d1">

Here we provide the top level container Devices and the information on the Device.

1. <Description manufacturer="NIST" serialNumber="01"/>
2. <DataItems>
3. <DataItem type="AVAILABILITY" name="avail" category="EVENT"

id="a"/>

1. </DataItems>
2. <Components>
3. <Axes name="Axes" id="3">

On line 11 we introduce the collection of Axes. The Axes component is a special component that acts as an abstract component as well as a collection. The Axes component contains various data items that have a global context; they are not associated with any one data item, but they go across all axes.

1. <Components>
2. <Rotary name="C" id="c1">
3. <DataItems>
4. <DataItem type="ROTARY\_VELOCITY" name="Cspeed" category="SAMPLE"

id="c2" nativeUnits="REVOLUTION/MINUTE" subType="ACTUAL"

units="REVOLUTION/MINUTE">

1. <Source>Sspeed</Source>
2. </DataItem>
3. <DataItem type=”ROTARY\_MODE” name”Cmode” category=”EVENT”

id=”c3”>

1. <Values><Value>SPINDLE</Value><Values>
2. </DataItem>
3. </DataItems>
4. </Rotary>

The spindle component declared on line 13 is the C axis and has spindle specific data items.

1. <Linear name="X" id="x1">
2. <DataItems>
3. <DataItem type="POSITION" name="Xact" category="SAMPLE" id="x2"

nativeUnits="MILLIMETER" subType="ACTUAL" units="MILLIMETER"/>

1. <DataItem type="POSITION" name="Xcom" category="SAMPLE" id="x3"

nativeUnits="MILLIMETER" subType="COMMANDED"

units="MILLIMETER"/>

1. </DataItems>
2. </Linear>
3. <Linear name="Y" id="y1">
4. <DataItems>
5. <DataItem type="POSITION" name="Yact" category="SAMPLE" id="y2"

nativeUnits="MILLIMETER" subType="ACTUAL"

units="MILLIMETER"/>

1. <DataItem type="POSITION" name="Ycom" category="SAMPLE" id="y3"

nativeUnits="MILLIMETER" subType="COMMANDED"

units="MILLIMETER"/>

1. </DataItems>
2. </Linear>
3. <Linear name="Z" id="z1">
4. <DataItems>
5. <DataItem type="POSITION" name="Zact" category="SAMPLE" id="z2"

nativeUnits="MILLIMETER" subType="ACTUAL" units="MILLIMETER"/>

1. <DataItem type="POSITION" name="Zcom" category="SAMPLE" id="z3"

nativeUnits="MILLIMETER" subType="COMMANDED"

units="MILLIMETER"/>

1. </DataItems>
2. </Linear>
3. </Components>
4. </Axes>

Lines 23, 29, and 35 define the three linear axes X, Y, and Z respectively. In this example device, the Agent is only collecting the actual and commanded positions.

The Controller is capable of providing the program name, block, and the current line being executed:

1. <Controller name="Controller" id="8">
2. <Components>
3. <Path id=”p1” name=”path”>
4. <DataItems>
5. <DataItem type="LINE" name="line" category="EVENT" id="p1"/>
6. <DataItem type="CONTROLLER\_MODE" name="mode" category="EVENT"

id="p2"/>

1. <DataItem type="PROGRAM" name="program" category="EVENT"

id="p3"/>

1. <DataItem type="EXECUTION" name="execution" category="EVENT"

id="p4"/>

1. <DataItem type="PATH\_FEEDRATE" name="feedrate" category="SAMPLE"

id="p4" units=”MILLIMETER/SECOND”

nativeUnits=”MILLIMETER/SECOND” />

1. <DataItem type="PATH\_POSITION" name="position" category="SAMPLE"

id="p4" units=”MILLIMETER\_3D” nativeUnits=”INCH\_3D”/>

1. </DataItems>
2. </Path>
3. </Components>
4. </Controller>
5. </Components>
6. </Device>
7. </Devices>
8. </MTConnectDevices>

Appendices

1. Bibliography
2. Engineering Industries Association. *EIA Standard - EIA-274-D*, Interchangeable Variable, Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically Controlled Machines. Washington, D.C. 1979.
3. ISO TC 184/SC4/WG3 N1089. *ISO/DIS 10303-238*: Industrial automation systems and integration Product data representation and exchange Part 238: Application Protocols: Application interpreted model for computerized numerical controllers. Geneva, Switzerland, 2004.
4. International Organization for Standardization. *ISO 14649*: Industrial automation systems and integration – Physical device control – Data model for computerized numerical controllers – Part 10: General process data. Geneva, Switzerland, 2004.
5. International Organization for Standardization. *ISO 14649*: Industrial automation systems and integration – Physical device control – Data model for computerized numerical controllers – Part 11: Process data for milling. Geneva, Switzerland, 2000.
6. International Organization for Standardization. *ISO 6983/1* – Numerical Control of machines – Program format and definition of address words – Part 1: Data format for positioning, line and contouring control systems. Geneva, Switzerland, 1982.
7. Electronic Industries Association. *ANSI/EIA-494-B-1992*, 32 Bit Binary CL (BCL) and 7 Bit ASCII CL (ACL) Exchange Input Format for Numerically Controlled Machines. Washington, D.C. 1992.
8. National Aerospace Standard. *Uniform Cutting Tests* - NAS Series: Metal Cutting Equipment Specifications. Washington, D.C. 1969.
9. International Organization for Standardization. *ISO 10303-11*: 1994, Industrial automation systems and integration Product data representation and exchange Part 11: Description methods: The EXPRESS language reference manual. Geneva, Switzerland, 1994.
10. International Organization for Standardization. *ISO 10303-21*: 1996, Industrial automation systems and integration -- Product data representation and exchange -- Part 21: Implementation methods: Clear text encoding of the exchange structure. Geneva, Switzerland, 1996.
11. H.L. Horton, F.D. Jones, and E. Oberg. *Machinery's handbook*. Industrial Press, Inc. New York, 1984.
12. International Organization for Standardization. *ISO 841-2001: Industrial automation systems and integration - Numerical control of machines - Coordinate systems and motion nomenclature.* Geneva, Switzerland, 2001.
13. ASME B5.57: *Methods for Performance Evaluation of Computer Numerically Controlled Lathes and Turning Centers,* 1998
14. ASME/ANSI B5.54: *Methods for Performance Evaluation of Computer Numerically Controlled Machining Centers. 2005.*
15. OPC Foundation. *OPC Unified Architecture Specification, Part 1: Concepts Version 1.00. July 28, 2006.*
16. 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.*
17. IEEE STD 1451.4-1994*, Standard for a Smart Transducer Interface for Sensors and Actuators – Mixed-Mode 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, SH95225, December 15, 2004.*
18. Machine Tool Modeling

The following section will provide example machine tool configurations and reference MTConnect® implementations. The following is the recommended machine modeling and implementation reference.

MTConnect utilizes the right hand rule for all coordinate systems representing physical space and orientation within a machine. The positive movement is given by extending the first three fingers on the right hand and labeling the axes in order of the digits, X, Y, and Z. The fingers will point in the positive direction. All Linear axes represent a space within a machine that is defined by coordinates according to the right hand rule.



Figure : Right Hand Rule Coordinate Planes

For Rotary axes, the right hand rule defines the direction of rotary movement by wrapping one's right-hand fingers around the axis of rotation. Clockwise rotation points the thumb toward the person, and counterclockwise rotation points the thumb away. The thumb indicates in the positive direction of the vector or axis the hand encircles. All rotational angles and movement is given according to the right hand rule for Rotary axes.

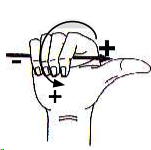


Figure : Rotational Right Hand Rule

* 1. Vertical Three Axis Mill

This is a simple machine tool with a vertical spindle and a table that can move in two dimensions. The modeling always starts with the Linear Z axis that is aligned with the primary spindle. The X axis is defined as the longest axis perpendicular to the Z axis. The spindle is now defined as a Rotary C axis that rotates around the Z axis.



Figure : Three Axis Mill

The right hand rule applies when naming the axes and defining positive motion and rotation. In this case the Rotary axis only operate as a spindle, so it will have a constant valued RotaryMode data item. This machine is only capable of executing a single program and therefore only capable of a single path. The following XML describes a simple configuration for this machine.

<?xml version="1.0" encoding="UTF-8"?>

<MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1 MTConnectDevices.xsd">

<Header bufferSize="130000" instanceId="1" creationTime="2009-11-13T02:31:40" sender="local"

version="1.1"/>

<Devices>

<Device id="d1" uuid="HM1" name="HMC\_3Axis">

<Description>3 Axis Mill</Description>

<Components>

<Axes id="a" name="base">

<Components>

<Linear id="y" name="Y">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="yp" category="SAMPLE"

name="Yact" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Linear id="x" name="X">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"

name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Linear id="z" name="Z">

<DataItems>

<DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact"

subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Rotary id="c" name="C">

<DataItems>

<DataItem type="ROTARY\_VELOCITY" id="cspd" category="SAMPLE" name="Sspeed"

subType="ACTUAL" units="REVOLUTION/MINUTE"

nativeUnits="REVOLUTION/MINUTE"/>

<DataItem type=" ROTARY\_VELOCITY " id="cso" category="SAMPLE"

name="Sovr" subType="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>

<DataItem type="ROTARY\_MODE" id="rf" category="EVENT" name="rfunc">

<Constraints>

<Value>SPINDLE</Value>

</Constraints>

</DataItem>

</DataItems>

</Rotary>

</Components>

</Axes>

<Controller id="cont" name="controller">

<Components>

<Path id="path" name="path">

<DataItems>

<DataItem type="PROGRAM" id="pgm" category="EVENT" name="program"/>

<DataItem type="BLOCK" id="blk" category="EVENT" name="block"/>

<DataItem type="LINE" id="ln" category="EVENT" name="line"/>

<DataItem type="PATH\_FEEDRATE" id="pf" category="SAMPLE" name="Fact"

units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL"/>

<DataItem type="PATH\_FEEDRATE" id="pfo" category="SAMPLE" name="Fovr"

units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>

<DataItem type="PATH\_POSITION" id="pp" category="SAMPLE" name="Ppos"

units="MILLIMETER\_3D" nativeUnits="FOOT\_3D" coordinateSystem="WORK"/>

<DataItem type="EXECUTION" id="exec" category="EVENT" name="execution"/>

<DataItem type="CONTROLLER\_MODE" id="cm" category="EVENT" name="mode"/>

</DataItems>

</Path>

</Components>

</Controller>

</Components>

</Device>

</Devices>

</MTConnectDevices>

* 1. Two Axis Lathe

The next machine is a simple two axis horizontal lathe with a Z and an X axis where the Linear Z axis aligned with the primary spindle Rotary C. The material is now held in the C axis and the tool is fixed.



Figure : Two Axis Lathe

<?xml version="1.0" encoding="UTF-8"?>

<MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1 MTConnectDevices.xsd">

<Header bufferSize="130000" instanceId="1" creationTime="2009-11-13T02:31:40" sender="local"

version="1.1"/>

<Devices>

<Device id="d1" uuid="HM1" name="HMC\_3Axis">

<Description>3 Axis Mill</Description>

<Components>

<Axes id="a" name="base">

<Components>

<Linear id="x" name="X">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"

name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Linear id="z" name="Z">

<DataItems>

<DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact"

subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Rotary id="c" name="C">

<DataItems>

<DataItem type=" ROTARY\_VELOCITY " id="cspd" category="SAMPLE" name="Sspeed"

subType="ACTUAL" units="REVOLUTION/MINUTE"

nativeUnits="REVOLUTION/MINUTE"/>

<DataItem type=" ROTARY\_VELOCITY " id="cso" category="SAMPLE" name="Sovr"

subType="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>

<DataItem type="ROTARY\_MODE" id="rf" category="EVENT" name="rfunc">

<Constraints>

<Value>SPINDLE</Value>

<Value>INDEX</Value>

</Constraints>

</DataItem>

</DataItems>

</Rotary>

</Components>

</Axes>

<Controller id="cont" name="controller">

<Components>

<Path id="path" name="path">

<DataItems>

<DataItem type="PROGRAM" id="pgm" category="EVENT" name="program"/>

<DataItem type="BLOCK" id="blk" category="EVENT" name="block"/>

<DataItem type="LINE" id="ln" category="EVENT" name="line"/>

<DataItem type="PATH\_FEEDRATE" id="pf" category="SAMPLE" name="Fact"

units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL"/>

<DataItem type="PATH\_FEEDRATE" id="pfo" category="SAMPLE" name="Fovr"

units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>

<DataItem type="PATH\_POSITION" id="pp" category="SAMPLE" name="Ppos"

units="MILLIMETER\_3D" nativeUnits="FOOT\_3D" coordinateSystem="WORK"/>

<DataItem type="EXECUTION" id="exec" category="EVENT" name="execution"/>

<DataItem type="CONTROLLER\_MODE" id="cm" category="EVENT" name="mode"/>

</DataItems>

</Path>

</Components>

</Controller>

</Components>

</Device>

</Devices>

</MTConnectDevices>

* 1. HyperQuadrex

Figure : HyperQuadrex Lathe

<?xml version="1.0" encoding="UTF-8"?>

<MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1 ../MTConnectDevices.xsd">

<Header bufferSize="130000" instanceId="1" creationTime="2009-11-13T02:31:40" sender="local"

version="1.1"/>

<Devices>

<Device id="d1" uuid="HM1" name="HyperQuadrex">

<Description>Mazak - HyperQuadrex</Description>

<Components>

<Axes id="a" name="base">

<Components>

<Linear id="x" name="X" nativeName="X1">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"

name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE">

<Source>X1pos</Source>

</DataItem>

<DataItem type="FORCE" id="xl" category="SAMPLE" name="Xforce" units="NEWTON">

<Source>X1force</Source>

</DataItem>

</DataItems>

</Linear>

<Linear id="y" name="Y" nativeName="Y1">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="yp" category="SAMPLE"

name="Yact" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE">

<Source>Y1pos</Source>

</DataItem>

<DataItem type="FORCE" id="yl" category="SAMPLE" name="Yforce" units="NEWTON">

<Source>Y1force</Source>

</DataItem>

</DataItems>

</Linear>

<Linear id="z" name="Z" nativeName="Z1">

<DataItems>

<DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact"

subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE">

<Source>Z1pos</Source>

</DataItem>

<DataItem type="FORCE" id="zl" category="SAMPLE" name="Zforce" units="NEWTON">

<Source>Z1force</Source>

</DataItem>

</DataItems>

</Linear>

<Linear id="x2" name="X2" >

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="x2p" category="SAMPLE"

name="X2act" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE"/>

<DataItem type="FORCE" id="x2l" category="SAMPLE" name="X2force" units="NEWTON">

<Source>X2force</Source>

</DataItem>

</DataItems>

</Linear>

<Linear id="y2" name="Y2">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="y2p" category="SAMPLE"

name="Y2act" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE"/>

<DataItem type="FORCE" id="y2l" category="SAMPLE" name="Y2force"

units="NEWTON"/>

</DataItems>

</Linear>

<Linear id="z2" name="Z2">

<DataItems>

<DataItem type="POSITION" id="z2p" category="SAMPLE" name="Z2act"

subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE">

<Source>Z2pos</Source>

</DataItem>

<DataItem type="FORCE" id="z2l" category="SAMPLE" name="Z2force"

units="NEWTON"/>

</DataItems>

</Linear>

<Linear id="z3" name="Z3" nativeName="W">

<DataItems>

<DataItem type="POSITION" id="z3p" category="SAMPLE" name="Z3act"

subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"

coordinateSystem="MACHINE">

<Source>Wpos</Source>

</DataItem>

<DataItem type="FORCE" id="z3l" category="SAMPLE" name="Z3force" units="NEWTON">

<Source>Wforce</Source>

</DataItem>

</DataItems>

</Linear>

<Rotary id="c" name="C " nativeName="C1">

<DataItems>

<DataItem type="FORCE" id="Cl" category="SAMPLE" name="Cforce" units="NEWTON"/>

<DataItem type=" ROTARY\_VELOCITY " id="cspd" category="SAMPLE" name="Sspeed"

subType="ACTUAL" units="REVOLUTION/MINUTE"

nativeUnits="REVOLUTION/MINUTE"/>

<DataItem type=" ROTARY\_VELOCITY " id="cso" category="SAMPLE" name="Sovr"

subType="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>

<DataItem type="DIRECTION" id="cdir" category="EVENT" name="Sdir"/>

<DataItem type="ANGLE" id="cpos" category="SAMPLE" name="Cpos"

subType="ACTUAL" units="DEGREE" nativeUnits="DEGREE" nativeScale="-1.0"/>

<DataItem type="ROTARY\_MODE" id="rf" category="EVENT" name="rfunc">

<Constraints>

<Value>SPINDLE</Value>

<Value>INDEX</Value>

</Constraints>

</DataItem>

</DataItems>

</Rotary>

<Rotary id="c2" name="C2">

<DataItems>

<DataItem type="FORCE" id="C2l" category="SAMPLE" name="C2force"

units="NEWTON"/>

<DataItem type=" ROTARY\_VELOCITY " id="c2spd" category="SAMPLE" name="Sspeed"

subType="ACTUAL" units="REVOLUTION/MINUTE"

nativeUnits="REVOLUTION/MINUTE"/>

<DataItem type=" ROTARY\_VELOCITY " id="c2so" category="SAMPLE" name="Sovr"

subType="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>

<DataItem type="DIRECTION" id="c2dir" category="EVENT" name="S2dir"/>

<DataItem type="ROTARY\_MODE" id="rf2" category="EVENT" name="rfunc">

<Constraints>

<Value>SPINDLE</Value>

</Constraints>

</DataItem>

</DataItems>

</Rotary>

<Rotary id="b" name="B" nativeName="S1">

<DataItems>

<DataItem type="FORCE" id="bl" category="SAMPLE" name="Bforce" units="NEWTON"/>

<DataItem type=" ROTARY\_VELOCITY " id="bspd" category="SAMPLE" name="Sspeed"

subType="ACTUAL" units="REVOLUTION/MINUTE"

nativeUnits="REVOLUTION/MINUTE"/>

<DataItem type=" ROTARY\_VELOCITY " id="bso" category="SAMPLE" name="Sovr"

subType="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>

<DataItem type="DIRECTION" id="bdir" category="EVENT" name="S3dir"/>

<DataItem type="ROTARY\_MODE" id="brf" category="EVENT" name="rfunc">

<Constraints>

<Value>SPINDLE</Value>

</Constraints>

</DataItem>

</DataItems>

</Rotary>

<Rotary id="b2" name="B2" nativeName="S2">

<DataItems>

<DataItem type="FORCE" id="b2l" category="SAMPLE" name="B2force"

units="NEWTON"/>

<DataItem type=" ROTARY\_VELOCITY " id="b2spd" category="SAMPLE" name="Sspeed"

subType="ACTUAL" units="REVOLUTION/MINUTE"

nativeUnits="REVOLUTION/MINUTE"/>

<DataItem type=" ROTARY\_VELOCITY " id="b2so" category="SAMPLE" name="Sovr"

subType="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>

<DataItem type="DIRECTION" id="b2dir" category="EVENT" name="S3dir"/>

<DataItem type="ROTARY\_MODE" id="b2rf" category="EVENT" name="rfunc">

<Constraints>

<Value>SPINDLE</Value>

</Constraints>

</DataItem>

</DataItems>

</Rotary>

</Components>

</Axes>

<Controller id="cont" name="controller">

<Components>

<Path id="path1" name="path1">

<DataItems>

<DataItem type="ACTIVE\_AXES" category="EVENT" name="axes" id="act\_axes1"/>

<DataItem type="PROGRAM" id="pgm1" category="EVENT" name="program"/>

<DataItem type="BLOCK" id="blk1" category="EVENT" name="block"/>

<DataItem type="LINE" id="ln1" category="EVENT" name="line"/>

<DataItem type="PATH\_FEEDRATE" id="pf1" category="SAMPLE" name="Fact"

units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL"

coordinateSystem="WORK"/>

<DataItem type="PATH\_FEEDRATE" id="pfo1" category="SAMPLE" name="Fovr"

units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>

<DataItem type="PATH\_POSITION" id="pp1" category="SAMPLE" name="Ppos"

units="MILLIMETER\_3D" nativeUnits="MILLIMETER\_3D"

coordinateSystem="WORK"/>

<DataItem type="TOOL\_ASSET\_ID" id="tid1" category="EVENT" name="Tid"/>

<DataItem type="PART\_ID" id="pid1" category="EVENT" name="Pid"/>

<DataItem type="EXECUTION" id="exec1" category="EVENT" name="execution"/>

<DataItem type="CONTROLLER\_MODE" id="cm1" category="EVENT" name="mode"/>

</DataItems>

</Path>

<Path id="path2" name="path2">

<DataItems>

<DataItem type="ACTIVE\_AXES" category="EVENT" name="axes" id="act\_axes2"/>

<DataItem type="PROGRAM" id="pgm2" category="EVENT" name="program"/>

<DataItem type="BLOCK" id="blk2" category="EVENT" name="block"/>

<DataItem type="LINE" id="ln2" category="EVENT" name="line"/>

<DataItem type="PATH\_FEEDRATE" id="pf2" category="SAMPLE" name="Fact"

units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL"

coordinateSystem="WORK"/>

<DataItem type="PATH\_FEEDRATE" id="pfo2" category="SAMPLE" name="Fovr"

units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>

<DataItem type="PATH\_POSITION" id="pp2" category="SAMPLE" name="Ppos"

units=" MILLIMETER\_3D" nativeUnits=" MILLIMETER\_3D"

coordinateSystem="WORK"/>

<DataItem type="TOOL\_ASSET\_ID" id="tid2" category="EVENT" name="Tid"/>

<DataItem type="PART\_ID" id="pid2" category="EVENT" name="Pid"/>

<DataItem type="EXECUTION" id="exec2" category="EVENT" name="execution"/>

<DataItem type="CONTROLLER\_MODE" id="cm2" category="EVENT" name="mode"/>

</DataItems>

</Path>

</Components>

</Controller>

<Door id="d" name="door">

<DataItems>

<DataItem id="ds" category="EVENT" name="door" type="DOOR\_STATE"/>

</DataItems>

</Door>

</Components>

</Device>

</Devices>

</MTConnectDevices>

* 1. Sensors

Sensors are modeled with the data items associated directly with the component that is being measured. In the example below, the spindle has measurement for temperature (thermistor) and vibration (accelerometer). Additionally, the sensor unit may have its own diagnostic measurements – in this case, a temperature measurement (thermistor) to measure the health of the sensor unit.

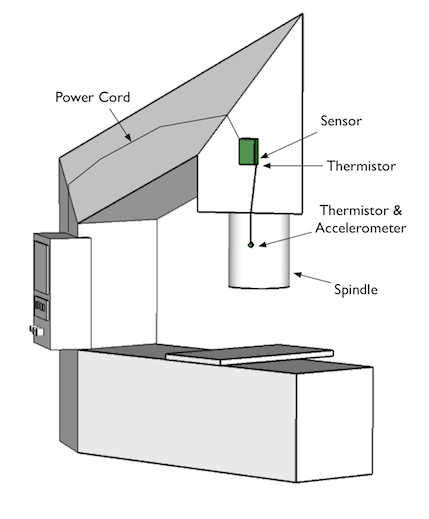


Figure : Spindle Sensing System

The basic machine is modeled below – 3 linear axes and a spindle. The spindle has two additional data items representing the sensors for temperature and acceleration.

<?xml version="1.0" encoding="UTF-8"?>

<MTConnectDevices xmlns="urn:mtconnect.org:MTConnectDevices:1.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mtconnect.org:MTConnectDevices:1.2 ../MTConnectDevices\_1.2.xsd">

<Header bufferSize="130000" instanceId="1" creationTime="2009-11-13T02:31:40" sender="local"

version="1.2"/>

<Devices>

<Device id="d1" uuid="HM1" name="HMC\_3Axis">

<Description>3 Axis Mill</Description>

<DataItems>

<DataItem type="AVAILABILITY" category="EVENT" id="avail" />

</DataItems>

<Components>

<Axes id="a" name="base">

<Components>

<Linear id="y" name="Y">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="yp" category="SAMPLE" name="Yact"

units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Linear id="x" name="X">

<DataItems>

<DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE" name="Xact"

units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Linear id="z" name="Z">

<DataItems>

<DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact" subType="ACTUAL"

units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>

</DataItems>

</Linear>

<Rotary id="c" name="C">

<DataItems>

<DataItem type="ROTARY\_VELOCITY" id="cspd" category="SAMPLE" name="Sspeed"

subType="ACTUAL" units="REVOLUTION/MINUTE"

nativeUnits="REVOLUTION/MINUTE"/>

<DataItem type="ROTARY\_VELOCITY" id="cso" category="SAMPLE" name="Sovr"

subType="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>

<DataItem type="ROTARY\_MODE" id="rf" category="EVENT" name="rfunc">

<Constraints>

<Value>SPINDLE</Value>

</Constraints>

</DataItem>

<DataItem type="TEMPERATURE" category="SAMPLE" name="Ctemp" id="ct"

units="CELSIUS" statistic="AVERAGE">

<Source componentId="s1">channel:1</Source>

<DataItem type="ACCLERATION" category="SAMPLE" name="Sacc" id="sa"

units="MILLIMETERS/SECOND^2" statistic="MAXIMUM">

<Source componentId="s2">channel:2</Source>

</DataItem>

</DataItems>

</Rotary>

</Components>

</Axes>

Additionally, the sensor unit is modeled with its configuration information and a data item sample (Voltage) and a data item condition (Voltage).

<Components>

<Sensor id="sensor" name="sensor">

<Configuration>

<SensorConfiguration>

<FirmwareVersion>2.02</FirmwareVersion>

<CalibrationDate>2010-05-16</CalibrationDate>

<NextCalibrationDate>2010-05-16</NextCalibrationDate>

<CalibrationInitials>WS</CalibrationInitials>

<Channels>

<Channel number="1" name="A/D:1">

<Description>A/D With Thermister</Description>

<Channel number="2" name="A/D:2">

<Description>A/D With Accelerometer</Description>

</Channel>

</Channels>

</SensorConfiguration>

</Configuration>

<DataItems>

<DataItem category="CONDITION" id="senvc" type="VOLTAGE" />

<DataItem category="SAMPLE" id="senv" type="VOLTAGE" units="VOLT"

subType="DIRECT" />

</DataItems>

</Sensor>

</Components>

</Device>

</Devices>

</MTConnectDevices>