



MTConnect[®] Standard

Part 2 – Components and Data Items

Version 1.2.0 – Final

Prepared for: MTConnect Institute
Prepared by: John Turner
Prepared on: February 17, 2012

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 MUST 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 non-infringement, 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.....	7
3.2	Device.....	7
3.2.1.1	Device Structure.....	8
3.2.1.2	Device Attributes.....	9
3.2.1.3	Device Elements.....	9
3.3	Components.....	10
3.4	Component Types.....	10
3.4.1	Component Schema.....	12
3.4.2	Component Attributes.....	12
3.4.3	Component Elements.....	13
3.4.3.1	Component Description.....	14
3.4.3.2	Component Components.....	15
3.4.3.3	Component DataItems.....	15
3.4.3.4	Component Configuration.....	15
3.5	DataItem.....	16
3.5.1	DataItem Schema.....	17
3.5.2	DataItem Attributes.....	18
3.5.3	Data Item Elements.....	19
3.5.3.1	Source Attributes.....	19
3.5.3.2	Constraints Elements.....	20
3.5.4	Data Item attribute: category.....	20
3.5.5	Data Item attribute: coordinateSystem.....	21
3.5.6	Data Item attribute: units.....	22
3.5.7	Data Item attribute: statistic.....	23
3.5.8	Data Item attribute: representation.....	24
3.5.9	Data Item Attribute: nativeUnits.....	24
3.5.10	Data Item Types for SAMPLE Category.....	26
3.5.11	Data Item Types for EVENT Category.....	29
3.5.12	Data Item Types for CONDITION Category.....	31
3.5.13	Schema Structure for DataItems.....	33
3.6	Component Types and Subcomponents.....	33
3.6.1	Axes.....	34
3.6.2	Controller.....	35
3.6.2.1	Path.....	35
3.6.3	Power DEPRECATED in Rel. 1.1.....	35
3.6.4	Door.....	35
3.6.5	Actuator.....	36
3.6.6	Sensor.....	36
3.6.6.1	Sensor data.....	36
3.6.6.2	Sensor Interface.....	37
3.6.6.3	Sensor as a Device.....	40
3.6.7	Sensor Configuration.....	41
3.6.7.1	SensorConfiguration Elements.....	42

3.6.7.2	Sensor Channel Attributes	42
3.6.7.3	Sensor Channel Elements	43
3.6.8	<i>Sensor Types</i>	44
3.6.8.1	Thermostat (DEPRECATED in Rel. 1.2. See TEMPERATURE).....	44
3.6.8.2	Vibration (DEPRECATED in Rel. 1.2. See DISPLACEMENT, FREQUENCY, etc.).....	44
3.6.8.3	Acceleration.....	44
3.6.8.4	Angular Acceleration.....	44
3.6.8.5	Angular Velocity	44
3.6.8.6	Amperage	44
3.6.8.7	Angle	44
3.6.8.8	Concentration	44
3.6.8.9	Conductivity	44
3.6.8.10	Direction	44
3.6.8.11	Displacement	44
3.6.8.1	Electrical Energy	44
3.6.8.2	Flow.....	44
3.6.8.3	Frequency	44
3.6.8.4	Fill Level	45
3.6.8.5	Linear Force.....	45
3.6.8.6	Load.....	45
3.6.8.7	Mass.....	45
3.6.8.8	PH.....	45
3.6.8.9	Pressure	45
3.6.8.10	Position	45
3.6.8.11	Power Factor.....	45
3.6.8.12	Resistance	45
3.6.8.13	Rotary Velocity.....	45
3.6.8.14	Sound Level.....	45
3.6.8.15	Strain.....	45
3.6.8.16	Temperature.....	45
3.6.8.17	Time.....	45
3.6.8.18	Tilt	45
3.6.8.19	Torque.....	46
3.6.8.20	Volt Ampere (VA).....	46
3.6.8.21	Volt Ampere Reactive (var).....	46
3.6.8.22	Velocity	46
3.6.8.23	Viscosity	46
3.6.8.24	Voltage.....	46
3.6.8.25	Wattage.....	46
3.6.9	<i>Systems</i>	46
3.6.9.1	Hydraulic	46
3.6.9.2	Pneumatic	46
3.6.9.3	Coolant	46
3.6.9.4	Lubrication	46
3.6.9.5	Electric.....	47
4	Component and Data Item Relationships.....	48
4.1	Device.....	48
4.1.1	<i>Device DataItems</i>	48
4.1.2	<i>Components of Device</i>	48
4.2	Common Components and Related Data Items	48
4.2.1	<i>Axes</i>	49
4.2.1.1	<i>Axes DataItems</i>	49
4.2.1.2	<i>Subcomponents of Axes</i>	49
4.2.2	<i>Linear (Subcomponent of Axes)</i>	49
4.2.2.1	<i>Linear Axes' DataItems (Sample and Event)</i>	49
4.2.2.2	<i>Linear Axes' Condition</i>	49

4.2.3	<i>Rotary (Subcomponent of Axes)</i>	50
4.2.3.1	Rotary Axes' DataItems (Sample and Event).....	50
4.2.3.2	Rotary Axes' Condition.....	50
4.2.4	<i>Controller</i>	50
4.2.4.1	<i>Subcomponents of Controller</i>	50
4.2.4.2	Controller DataItems (Sample and Event).....	50
4.2.4.3	Controller Condition.....	51
4.2.5	<i>Path (Subcomponent of Controller)</i>	51
4.2.5.1	Path DataItems (Sample and Event).....	51
4.2.5.2	Path Condition.....	51
4.2.6	Power DEPRECATED in Rel 1.1.....	51
4.2.7	<i>Sensors</i>	52
4.2.7.1	Sensor Condition.....	52
4.2.8	Thermostat DEPRECATED in REL 1.2. Replaced with a DataItem called Temperature.....	52
4.2.8.1	DataItem types	52
4.2.8.2	Condition types	52
4.2.9	Vibration DEPRECATED in REL 1.2. Replaced with DataItems to measure vibration (Displacement, Frequency, etc).....	52
4.2.9.1	DataItem types	52
4.2.9.2	Condition types	52
4.2.10	Pressure DEPRECATED in REL 1.2. Replace with DataItem Pressure.....	53
4.2.10.1	DataItem types	53
4.2.10.2	Condition types	53
4.2.11	<i>Door</i>	53
4.2.11.1	Door DataItems (Sample and Event).....	53
4.2.11.2	Door Condition.....	53
4.2.12	<i>Actuator</i>	53
4.2.12.1	Actuator DataItems (Sample and Event).....	53
4.2.12.2	Actuator Condition.....	53
4.2.13	Spindle DEPRECATED in Rel. 1.1.....	53
4.2.14	<i>Systems</i>	53
4.2.14.1	<i>Subcomponents of Systems</i>	53
4.2.15	<i>Hydraulic (Subcomponent of Systems)</i>	54
4.2.15.1	Hydraulic Condition.....	54
4.2.16	<i>Pneumatic (Subcomponent of Systems)</i>	54
4.2.16.1	Pneumatic Condition.....	54
4.2.17	<i>Coolant (Subcomponent of Systems)</i>	54
4.2.17.1	Coolant DataItems (Sample and Event).....	54
4.2.17.2	Coolant Condition.....	54
4.2.18	<i>Lubrication (Subcomponent of Systems)</i>	54
4.2.18.1	Lubrication DataItems (Sample and Event).....	54
4.2.18.2	Lubrication Condition.....	54
4.2.19	<i>Electric (Subcomponent of Systems)</i>	55
4.2.19.1	Electrical DataItems (Sample and Event).....	55
4.2.19.2	Electric Condition.....	55
5	Annotated XML Examples	56
5.1	Simplest Device.....	56
5.2	More Complex Example of Probe.....	57
	Appendices	60
A.	Bibliography	60
B.	Machine Tool Modeling	62
B.1.	Vertical Three Axis Mill.....	63
B.2.	Two Axis Lathe.....	66

B.3. HyperQuadrex	68
B.4. Sensors.....	74

Table of Figures

Figure 1: Example Container Structure	6
Figure 2: Device Schema Diagram	8
Figure 3: Component/Subcomponent Diagram	11
Figure 3: Component Schema.....	12
Figure 4: Component Schema.....	14
Figure 8: Example DataItem Structure.....	16
Figure 9: DataItem Schema Diagram	17
Figure 10: Constraints Schema	20
Figure 5: Axes Example With Three Linear Axes and one Rotary Axis.....	34
Figure 6: Sensor Data Associations.....	37
Figure 7: Sensor Associations	38
Figure 11: Right Hand Rule Coordinate Planes.....	62
Figure 12: Rotational Right Hand Rule	62
Figure 13: Three Axis Mill	63
Figure 14: Two Axis Lathe	66
Figure 15: HyperQuadrex Lathe	68
Figure 16: Spindle Sensing System	74

1 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 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.

41 **1.1 MTConnect[®] Document Structure**

42 The MTConnect[®] specification is subdivided using the following scheme:

43 Part 1: Overview and Protocol

44

45 Part 2: Components and Data Items

46

47 Part 3: Streams, Events, Samples, and Condition

48

49 Part 4: Assets

50

51 These four documents are considered the basis of the MTConnect Standard. Information
52 applicable to basic machine and device types will be included in these documents. Additional
53 parts to the standard will be added to provide information and extensions to the standard focused
54 on specific devices, components, or technologies considered requiring separate emphasis. All
55 information specific to the topic of each additional part **MUST** be included within that document
56 even when it is subject matter of one of the base parts of the standard.

57

58 Documents will be named (file name convention) as follows:

59 MTC_Part_<Number>_<Description>.doc.

60 For example, the file name for Part 2 of the standard is MTC_Part_2_Components.doc.

61 All documents will be developed in Microsoft[®] Word format and released in Adobe[®] PDF
62 format.

63

64 2 Purpose of This Document

65 The four base MTConnect[®] documents are intended to:

- 66
- 67 • define the MTConnect[®] standard;
- 68
- 69 • specify the requirements for compliance with the MTConnect[®] standard;
- 70
- 71 • provide engineers with sufficient information to implement *Agents* for their devices;
- 72
- 73 • provide developers with the necessary guidelines to use the standard to develop applications.

74 Part 1 of the MTConnect Standard provides an overview of the MTConnect Architecture and the
75 Protocol; including communications, fault tolerance, connectivity, and error handling
76 requirements.

77 Part 2 of the MTConnect[®] standard focuses on the data model and description of the information
78 that is available from the device. The descriptive data defines how a piece of equipment should
79 be modeled, the structure of the component hierarchy, the names for each component (if
80 restricted), and allowable data items for each of the components.

81 Part 3 of the MTConnect standard focuses on the data returned from a `current` or `sample`
82 request (for more information on these requests, see Part 1). This section covers the data
83 representing the state of the machine.

84 Part 4 of the MTConnect[®] standard provides a semantic model for entities that are used in the
85 manufacturing process, but are not considered to be a device nor a component. These entities are
86 defined as MTConnect[®] Assets. These assets may be removed from a device without detriment
87 to the function of the device, and can be associated with other devices during their lifecycle. The
88 data associated with these assets will be retrieved from multiple sources that are responsible for
89 providing their knowledge of the asset. The first type of asset to be addressed is Tooling.

90 2.1 Terminology

91	Adapter	An optional software component that connects the Agent to the Device.
92	Agent	A process that implements the MTConnect [®] HTTP protocol, XML generation, 93 and MTConnect protocol.
94	Alarm	An alarm indicates an event that requires attention and indicates a deviation 95 from normal operation. Alarms are reported in MTConnect as <code>Condition</code> .
96	Application	A process or set of processes that access the MTConnect [®] <i>Agent</i> to perform 97 some task.
98	Attribute	A part of an XML element that provides additional information about that 99 XML element. For example, the name XML element of the <code>Device</code> is given 100 as <code><Device name="mill-1">...</Device></code>

101	CDATA	The text in a simple content element. For example, <i>This is some text</i> ,
102		in <code><Message ...>This is some text</Message></code> .
103	Component	A part of a device that can have sub-components and data items. A
104		component is a basic building block of a device.
105	Controlled Vocabulary	The value of an element or attribute is limited to a restricted set of
106		possibilities. Examples of controlled vocabularies are country codes: US, JP,
107		CA, FR, DE, etc...
108	Current	A snapshot request to the <i>Agent</i> to retrieve the current values of all the data
109		items specified in the path parameter. If no path parameter is given, then the
110		values for all components are provided.
111	Data Item	A data item provides the descriptive information regarding something that can
112		be collected by the <i>Agent</i> .
113	Device	A piece of equipment capable of performing an operation. A device may be
114		composed of a set of components that provide data to the application. The
115		device is a separate entity with at least one component or data item providing
116		information about the device.
117	Discovery	Discovery is a service that allows the application to locate <i>Agents</i> for devices
118		in the manufacturing environment. The discovery service is also referred to as
119		the <i>Name Service</i> .
120	Event	An event represents a change in state that occurs at a point in time. Note: An
121		event does not occur at predefined frequencies.
122	HTTP	Hyper-Text Transport Protocol. The protocol used by all web browsers and
123		web applications.
124	Instance	When used in software engineering, the word <i>instance</i> is used to define a
125		single physical example of that type. In object-oriented models, there is the
126		class that describes the thing and the instance that is an example of that thing.
127	LDAP	Lightweight Directory Access Protocol, better known as Active Directory in
128		Microsoft Windows. This protocol provides resource location and contact
129		information in a hierarchal structure.
130	MIME	Multipurpose Internet Mail Extensions. A format used for encoding multipart
131		mail and http content with separate sections separated by a fixed boundary.
132	Probe	A request to determine the configuration and reporting capabilities of the
133		device.
134	REST	REpresentational State Transfer. A software architecture where the client and
135		server move through a series of state transitions based solely on the request
136		from the client and the response from the server.

137	Results	A general term for the <code>Samples</code> , <code>Events</code> , and <code>Condition</code> contained in a <code>ComponentStream</code> as a response from a sample or current request.
138		
139	Sample	A sample is a data point from within a continuous series of data points. An example of a <code>Sample</code> is the position of an axis.
140		
141	Socket	When used concerning inter-process communication, it refers to a connection between two end-points (usually processes). Socket communication most often uses TCP/IP as the underlying protocol.
142		
143		
144	Stream	A collection of <code>Events</code> , <code>Samples</code> , and <code>Condition</code> organized by devices and components.
145		
146	Service	An application that provides necessary functionality.
147	Tag	Used to reference an instance of an XML element.
148	TCP/IP	TCP/IP is the most prevalent stream-based protocol for inter-process 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.
149		
150		
151		
152	URI	Universal Resource Identifier. This is the official name for a web address as seen in the address bar of a browser.
153		
154	UUID	Universally unique identifier.
155	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
156		
157	XML	Extensible Markup Language. http://www.w3.org/XML/
158	XML Schema	The definition of the XML structure and vocabularies used in the XML Document.
159		
160	XML Document	An instance of an XML Schema which has a single root XML element and conforms to the XML specification and schema.
161		
162	XML Element	An element is the central building block of any XML Document. For example, in MTConnect [®] the Device XML element is specified as <code><Device>...</Device></code>
163		
164		
165	XML NMTOKEN	The data type for XML identifiers. It MUST start with a letter, an underscore “_” or a colon “:” and then it MUST be followed by a letter, a number, or one of the following “.”, “-”, “_”, “:”. An NMTOKEN cannot have any spaces or special characters.
166		
167		
168		

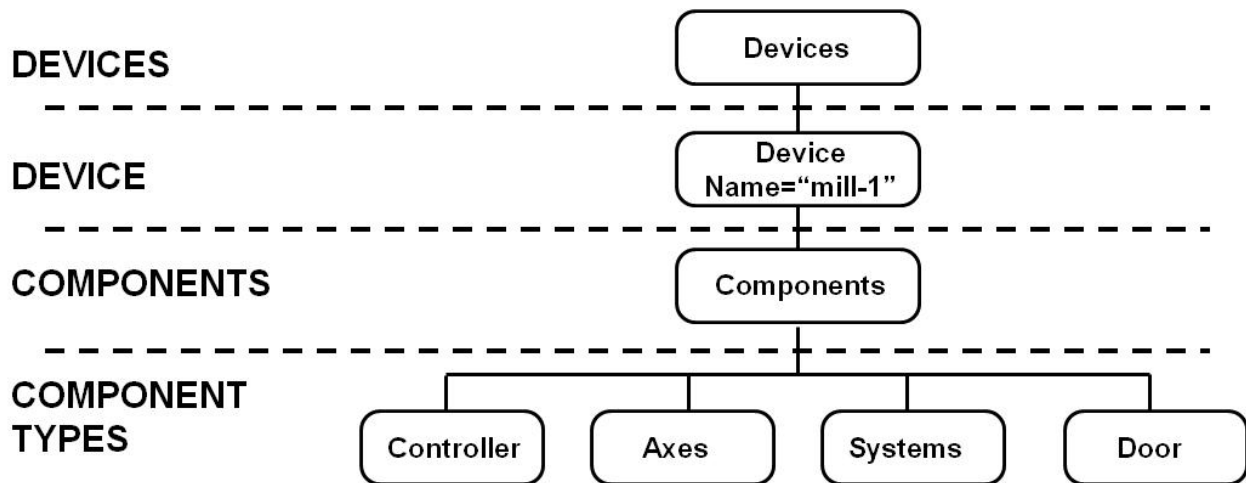
169 2.2 Terminology and Conventions

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

172 3 Devices and Components

173 MTConnect organizes information and data from a data source (typically a machine) into an
 174 information model that defines the relationship between each piece of data and the source of that
 175 data. This information model allows an application to interpret the data received from a data
 176 source and correlate that data to its original definition, value, and context.

177 The basic MTConnect information model contains three primary containers: `Devices`,
 178 `Device`, and `Components`. These containers are the building blocks used to organize
 179 information about a piece of equipment. They also define how the various parts of a piece of
 180 equipment relate to each other.



181
 182 **Figure 1: Example Container Structure**
 183

184 The first, or highest, level container in the MTConnect data structure is the `Devices` container.
 185 The `Devices` container is comprised of one or more `Device` XML Element(s). The
 186 `Devices` container provides a mechanism for grouping data from multiple `Device` elements
 187 that are providing their data through a common MTConnect *Agent*. `Devices` has no
 188 attributes and is only used to group data from `Device` elements together.

189 The next level container is `Device`. A `Device` typically represents a single piece of
 190 equipment or a machine. However, it can also represent any logical grouping of components
 191 that operate together to perform a function. Every `Device` in MTConnect[®] **MUST** have an
 192 `Availability` data item. `Availability` represents the device's ability to provide
 193 information about itself. The `Device` container is comprised of one or more `Components`
 194 XML Elements.

195 The third container in the MTConnect Data Structure is the `Components` container(s).
 196 `Components` provides a mechanism for grouping sub-elements of a `Device` into logical
 197 groups that are associated with each other. `Components` has no attributes and is only used to
 198 group `Component` elements together. The `Components` container is comprised of one or
 199 more `Component` XML Elements.

200 3.1 Devices

201 The Devices XML Element is the top level container for every Device. It may contain
202 multiple Device elements. Devices may only contain elements of type Device .

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

203

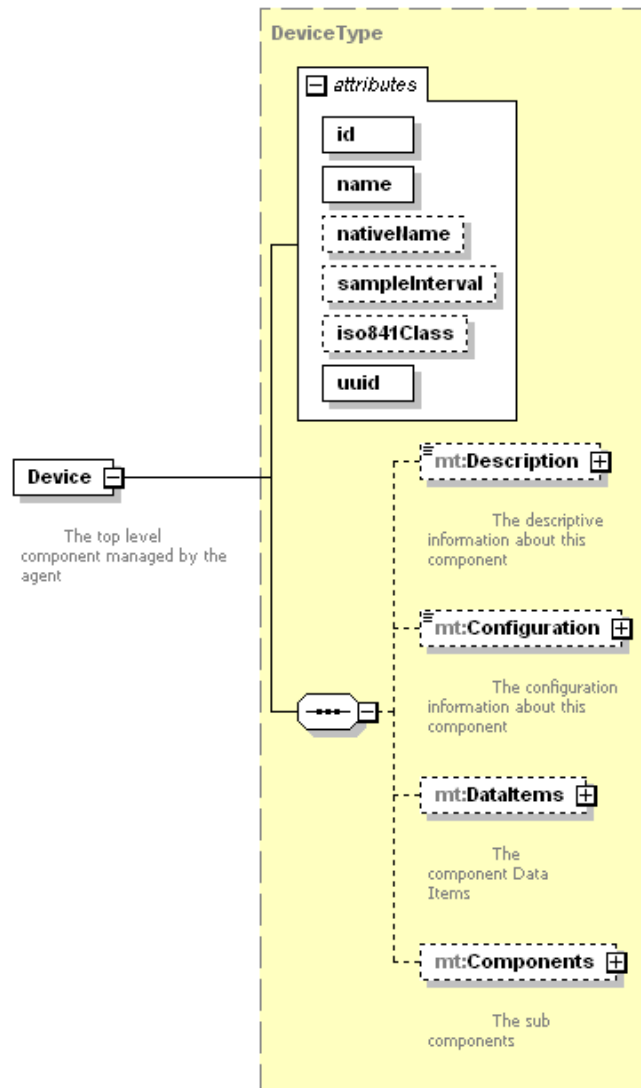
204 3.2 Device

205 A Device is a XML Element that holds all the Components associated with a piece of
206 equipment. This can be a logical grouping of Component XML Elements that perform a
207 particular function. The Device **MUST** have an Availability data item that indicates if
208 this device is available to provide information.

209 In the MTConnect[®] schema, a Device is actually a unique type of Component (defined
210 below). A Device supports all of the functions and capabilities defined for a Component .
211 However, it **MUST** be uniquely identified throughout the MTConnect[®] Standard and schema as a
212 Device to clearly define the difference between a logical collection of components that
213 function together as a Device and the identification of each Component that forms the
214 structure within a Device.

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

224 3.2.1.1 Device Structure



225

Generated by XMLSpy

www.altova.com

226

Figure 2: Device Schema Diagram

227

228 **3.2.1.2 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's 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 this Device. If the native name is not provided, it MUST be the name.	0..1
id	The unique identifier for this Device in the document. An id MUST 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**

229
 230 Notes: * The uuid **MUST** be provided for the Device. It is optional for all other
 231 Component types.
 232 ** The sampleInterval is used to aid an application in interpolating values. This is
 233 the desired sample interval and may vary depending on the capabilities of the device.

234 **3.2.1.3 Device Elements**

Element	Description	Occurrence
Description	An XML element that can contain any descriptive content. This can contain configuration information and manufacturer specific details.	0..1
Configuration	An XML element that can contain descriptive content defining the configuration information for a Device.	0..1
Components	A container for lower level Component XML Elements associated with this Device.	0..INF*
DataItems	The data items (defined below) provided by this Device. The data items define the measured values to be reported by this Device.	0..INF*

235
 236 Notes: *At least one of Components or DataItems **MUST** be provided.

237 3.3 Components

238 Components is a container that provides structure for sub-elements of a device.

239 Components contains one or more Component XML Elements.

Elements	Description	Occurrence
Component	Types of Component XML Elements. There can be multiple Component XML Elements.	1..INF

240

241 3.4 Component Types

242 A Component XML Element defines physical or logical sub-element of a device.

243 Component is an abstract type and will never appear in the MTConnect XML document.

244 Component elements are represented as XML Element sub-types such as Axes,
245 Controller, Door, etc.

246 Component elements contain information and data defining the element's operational state, the
247 environment in which it is functioning, and its health or status. This information and the
248 measured values associated with a component are defined as DataItems and will be discussed
249 in *Section 3.5* of this document.

250 Component can be further sub-divided into smaller Components XML Elements to provide
251 additional detail on the structure and configuration of a Component. These sub-elements have
252 all the characteristics and capabilities of the parent component.

253 While these sub-elements are by definition Components, they **SHALL** be called
254 *subcomponents* within the MTConnect Standard to provide clarity on the relationship between
255 the parent component and its associated sub-elements (*subcomponents*). Additionally,
256 *subcomponents* may be further subdivided into additional Components, as required, to provide
257 a complete description of a device and its measured values (DataItems).

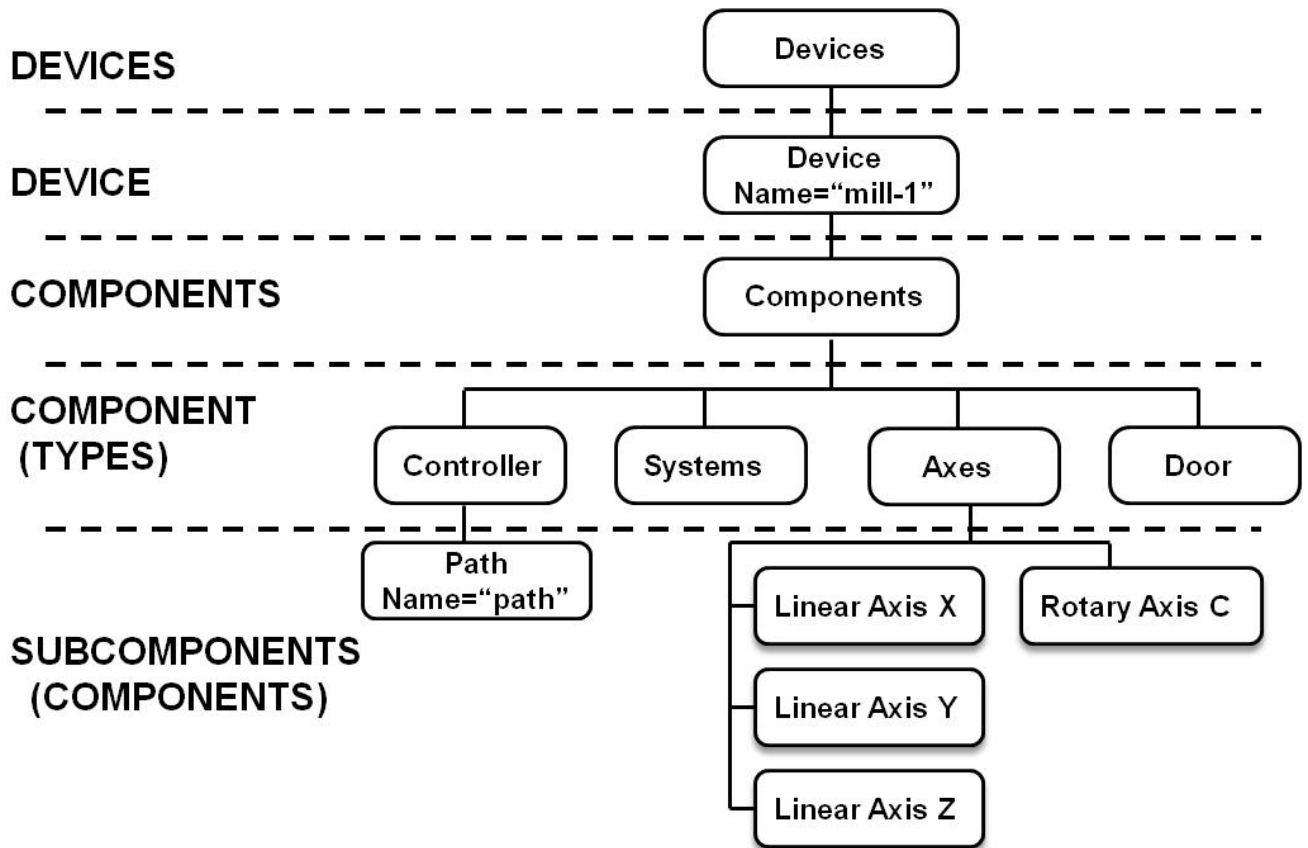
258 Components and related *subcomponents* are represented in the XML schema as follows:

```

259     <Devices>
260       <Device>
261         <Components>
262           <Axes( Component Type Subcomponent )>
263           <Components>
264             <Linear (Component Type Subcomponent) >
265               < Components>
266                 <Etc. >
267
268
```

269 Figure 3 below describes the relationship between Component and *subcomponents*.

270



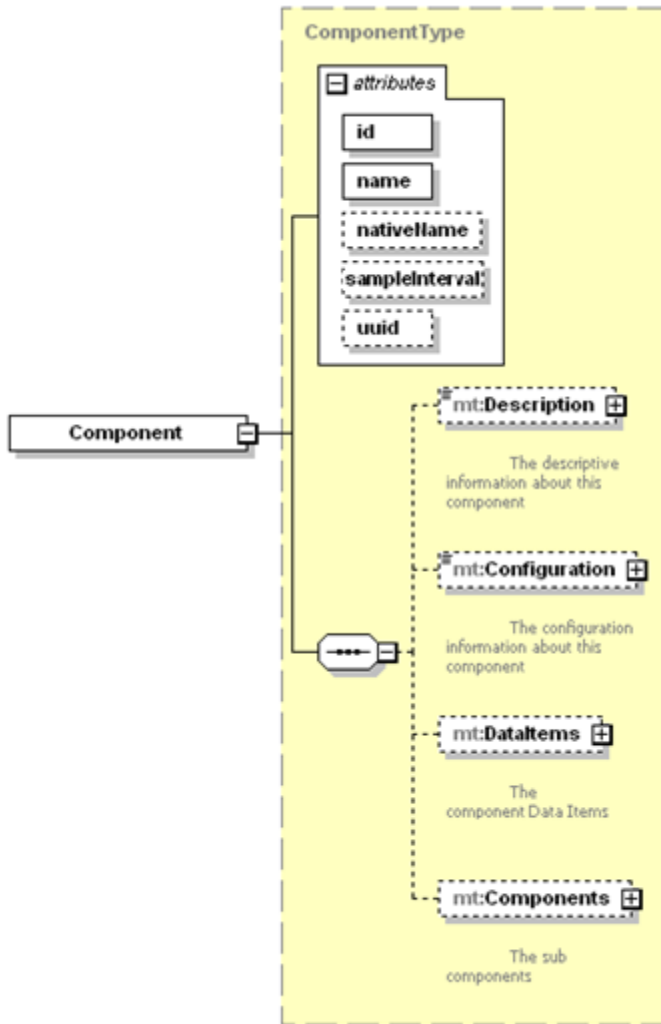
271

272

273

Figure 3: Component/Subcomponent Diagram

274 **3.4.1 Component Schema**



275

276

Figure 4: Component Schema

277 **3.4.2 Component Attributes**

278 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’s 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 MUST be the name.	0..1

Attribute	Description	Occurrence
id	The unique identifier for this Component in the document. An id MUST 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**

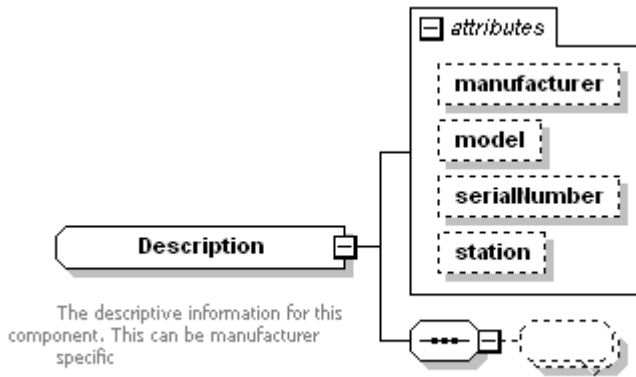
279
 280 Notes: * The uuid **MUST** be provided for the Device. It is optional for all other
 281 Component types.
 282 ** The sampleInterval is used to aid the application in interpolating values. This is
 283 the desired sample Interval and may vary depending on the capabilities of the
 284 component.

285 3.4.3 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	A container for lower level Component XML Elements associated with 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 define the measured values to be reported by this Component.	0..INF*

286
 287 Notes: *At least one of Components or DataItems **MUST** be provided.

288 **3.4.3.1 Component Description**



289

290

Figure 5: Component Schema

291

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

292

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

```
295 <Description manufacturer="Example Co" serialNumber="A124FFF"  

    296     station="2"> Example Co Simulated Vertical 3 Axis Machining center.  

    297 </Description>
```

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

```
300 <Description manufacturer="Example Co"  

    301     serialNumber="EXCO-TT-099PP-XXXX"> Advanced Pulse watt-hour transducer  

    302     with pulse output  

    303 </Description>
```

304

305 **3.4.3.2 Component Components**

Element	Description	Occurrence
Components	One or more subcomponents. This can also include the subtypes of Component like Axes, Linear, Path, etc...	1..INF

306

307 **3.4.3.3 Component DataItems**

Element	Description	Occurrence
DataItems	Only XML elements of types DataItem can be specified	1..INF

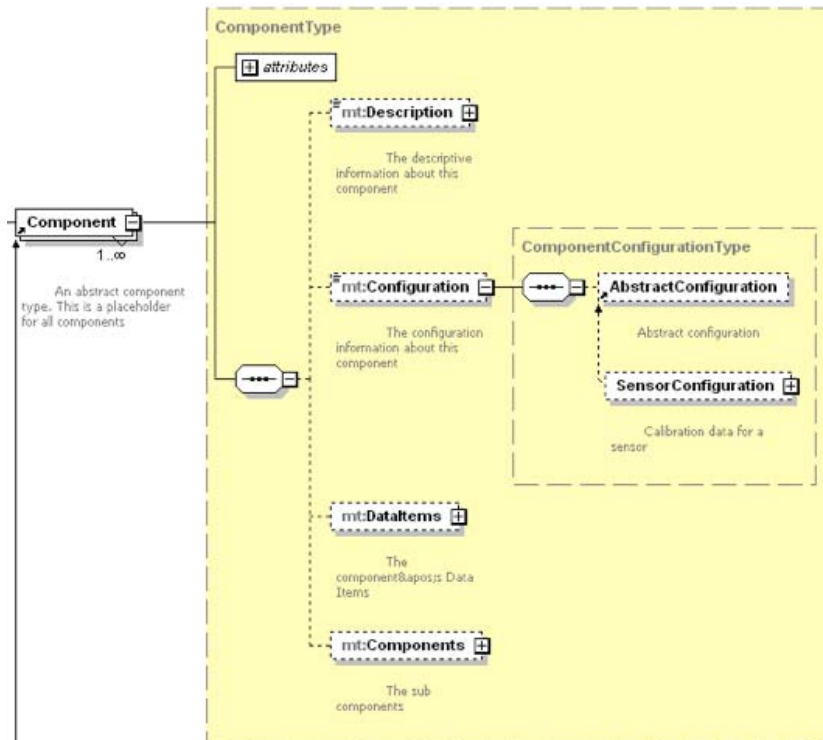
308

309 **3.4.3.4 Component Configuration**

Element	Description	Occurrence
Configuration	An XML element that can contain descriptive content defining the configuration information for a Component. Not all Component <i>types</i> 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

310

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



312

313

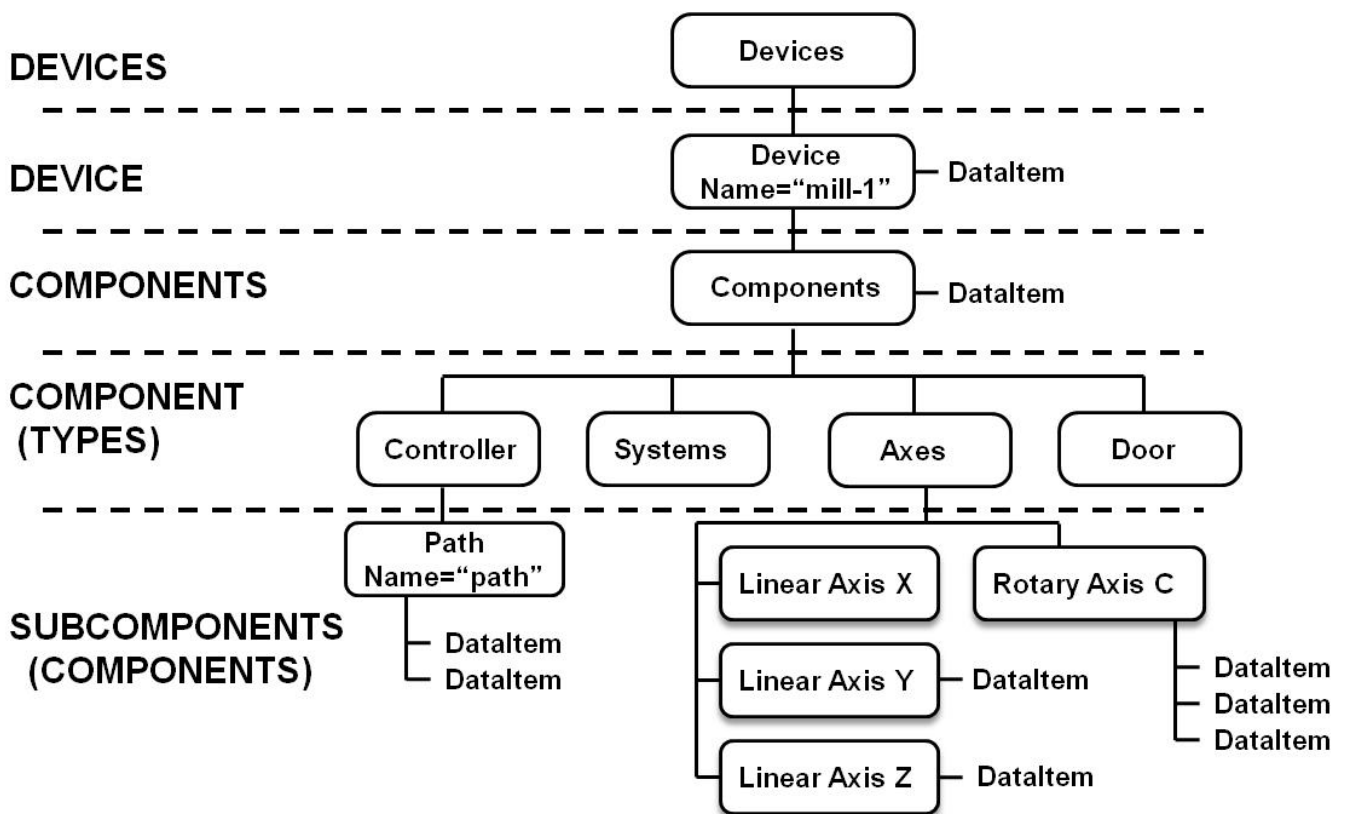
Figure 5: Component Configuration Schema

314 **3.5 DataItem**

315 DataItem is a piece of information that can be collected from a Device , Component, or
 316 *subcomponent*. A DataItem **MAY** report both a numeric value (a numeric quantity reported as
 317 either a Sample or Event category) and a health status (reported as a Condition category).
 318 A DataItem specifies the type of data being collected and an array of optional attributes that
 319 further defines that data. The value of the data is provided in the Streams response.

320 The Agent transmits data items associated with each Component to an application. The actual
 321 values for those data items are delivered in Streams and will be discussed in detail in the
 322 MTCConnect Standard *Part 3: Streams, Samples, and Events*.

323



324

325

326

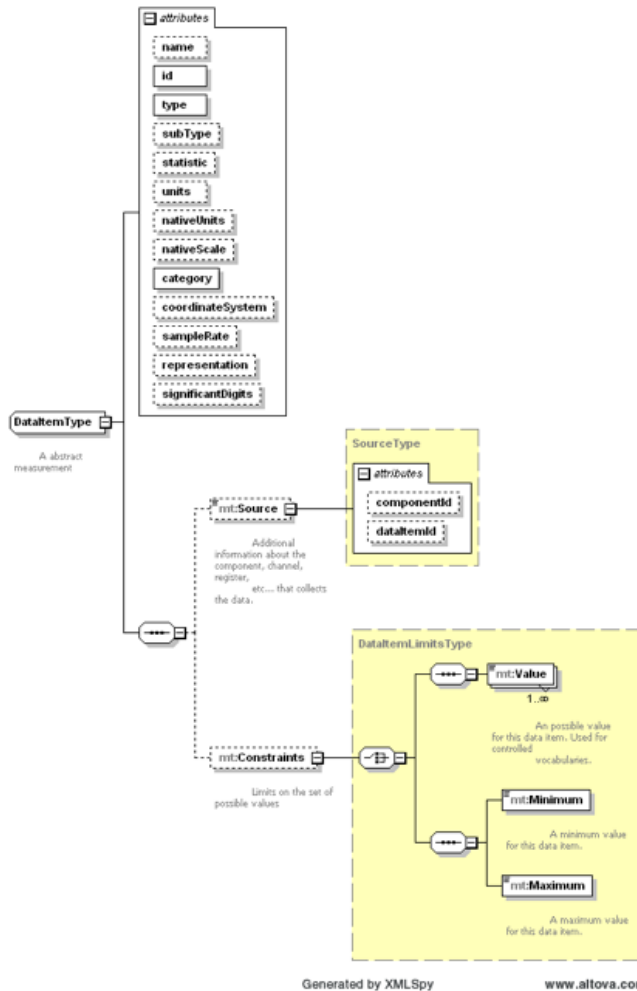
327

328

Figure 6: Example DataItem Structure

329 **3.5.1 DataItem Schema**

330 A DataItem **MUST** specify the type of data being collected, the id of the DataItem, and the
 331 category of the item. Since many data item types provide both a value (reported as either a
 332 Sample or Event category) and a health status (reported as a Condition category), each
 333 DataItem **MUST** report a category for each data type to aid the application in determining the
 334 specific meaning of the data. The DataItem **MAY** specify a Source sub-element to identify
 335 where the physical connection to the data source originates; ex. data relative to a servo motor
 336 may actually originate from a measurement made in the controller.
 337



338
 339 **Figure 7: DataItem Schema Diagram**

340
 341 A DataItem **MAY** also specify a *subtype* to further qualify the type of data being provided.
 342 *Subtypes* are required for certain data items. For example, the Position has two *subtypes*:
 343 ACTUAL and COMMANDED. These are two separate DataItem(s) that can be reported
 344 independently. See the sections below addressing Sample, Event, and Condition for a
 345 complete list of *type/subtype* relations.

346 For information on the transformation between DataItem name as returned in a Probe
 347 request and the corresponding data returned in a Stream element, see the MTConnect Standard
 348 *Part 3, Section 3.5*.

349 3.5.2 DataItem Attributes

350

Attribute	Description	Occurrence
id	The unique identifier for this DataItem. The id attribute MUST be unique across the entire document including the ids for components. An XML ID-type.	1
name	The name of the DataItem. A name is provided as an additional human readable identifier for this DataItem in addition to the id. It is not required and will be implementation dependent. 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 statistic 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 representation are VALUE, TIME_SERIES, MP3, WAV, etc. Initially, the representation for TIME_SERIES and VALUE are defined. If a representation is not specified, it MUST 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 MUST be present for all samples. If the data represented by a DataItem is a numeric value, except for line number and count, the units MUST 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 MUST be numeric.	0..1

Attribute	Description	Occurrence
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. SampleRate 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

351
352

353 3.5.3 Data Item Elements

Element	Description	Occurrence
Source	Source is an optional XML element that identifies the Component, subcomponent, or DataItem where the physical connection to the data source originates. The CDATA of the Source element MAY also contain 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 for X axis actual position “Xact”, but the X axis position is delivered from the device as Channel.0.position, Source is used to provide the necessary 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 that can be assigned to this DataItem. Constraints provide a way to specify the capabilities for this Component by limiting the choices for the value that is reported in the Streams response. For example, for ROTARY_MODE the axis can be limited to SPINDLE for an axis that can only spin.	0..1

354

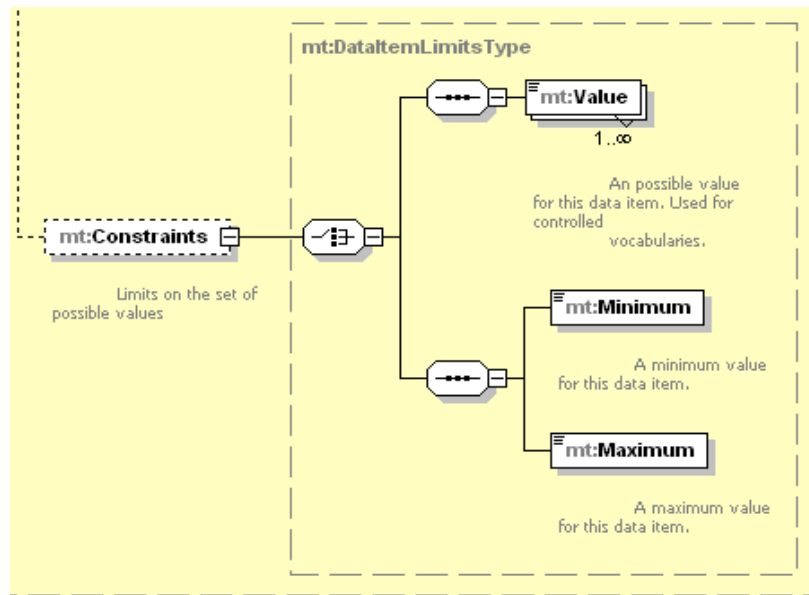
355 3.5.3.1 Source Attributes

356 Source identifies the physical device or data source where the data represented by the
357 DataItem is generated:

Attribute	Description	Occurrence
componentID	A unique identifier that references a specific Component from which the data represented by the DataItem originates. This MUST be the unique identifier defined for the component in its id attribute and MUST occur elsewhere in the XML document. It is an XML xs:IDREF type.	0..1

Attribute	Description	Occurrence
dataItemID	A unique identifier that references a specific DataItem from which the data represented by this DataItem is generated. This MUST be the unique identifier defined for the DataItem id attribute and MUST occur elsewhere in the XML document. It is an XML xs:IDREF type.	0..1

358 **3.5.3.2 Constraints Elements**



Generated by XMLSpy www.altova.com

359 **Figure 8: 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 DataItem value will be constant. In the case of a constant DataItem value, the value is not required to be supplied in the streams document.	0..INF
Maximum	The maximum value for this DataItem. This will be the bounded upper range. This will only be relevant when the DataItem has a numeric type.	0..1
Minimum	The minimum value for this DataItem. This will be the bounded lower range. This will only be relevant when the DataItem has a numeric type.	0..1

361

362 **3.5.4 Data Item attribute: category**

363 MTConnect[®] provides three different categories of DataItem - SAMPLE, EVENT, and
 364 CONDITION. The category will indicate where the results will be reported in the XML

365 Document as a response to a `Sample` or `Current` request. See *Part 3 Section 3 on Streams,*
 366 *Samples, and Events* for more information.

367 **SAMPLE** A `Sample` is the reading of the value of a continuously variable or analog
 368 `DataItem`. A continuous value can be sampled at any point-in-time and will
 369 always produce a result. An example of a continuous `DataItem` is the
 370 Linear X axis position.
 371

372 A `DataItem` of the category `Sample` that are continuous are always scalar
 373 floating point or integers that can have an infinite number of possible values.
 374 This is different from state or discrete type `DataItem` that has a limited
 375 number of possible values. A `DataItem` of category `Sample` **MUST** have
 376 units.

377 **EVENT** A `DataItem` of the category `Event` comprises discrete information from
 378 the device. There are two types of `Event`: those representing state, with two
 379 or more discrete values; and those representing messages that contain plain
 380 text data. An example of a state type `Event` is a `DoorStatus` that can be
 381 either `OPEN`, `UNLATCHED`, or `CLOSED`. An example of a message type
 382 `Event` is a `PROGRAM` that can be any valid string of characters. A
 383 `DataItem` of category `Event` does not have intermediate values that vary
 384 over time, as do `Samples`. An `Event` can be thought of as streaming
 385 information that if taken at any point in time represents the current state of the
 386 device.

387 **CONDITION** A `DataItem` that communicates the device's health and ability to function.
 388 A `DataItem` of category `Condition` can be one of `UNAVAILABLE`,
 389 `NORMAL`, `WARNING`, or `FAULT`. A `DataItem` of category
 390 `Condition` **MAY** report multiple active conditions at one time; whereas a
 391 `DataItem` of category `Sample` or `Event` can only have a single value at
 392 any one point in time.

393 **3.5.5 Data Item attribute: coordinateSystem**

394 A `DataItem` can specify an optional coordinate system that is being used. If not specified, the
 395 `Axes` coordinates **MUST** be `MACHINE` and the `Path` coordinates **MUST** be `WORK`. The
 396 possible values of coordinates are:

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

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

402

403 3.5.6 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

Units	Description
VOLT_ AMPERE	Volt-Ampere (VA)
VOLT_ AMPERE_ REACTIVE	Volt-Ampere Reactive (var)
WATT	Watts
WATT_ SECOND	Measurement of electrical energy, equal to one Joule

404

405 Units **MUST** be specified for any DataItem with category Sample. The nativeUnits
 406 **MAY** also be specified if they apply to the type of data and if they differ from the units. The
 407 *Agent* is responsible for converting the nativeUnits to the units before sending them to
 408 the applications. In addition, nativeUnits **MAY** be scaled using the nativeScale
 409 attribute; for example, if the device measures velocity in 100 ft/min, MTConnect® would
 410 represent it with the following attributes: nativeUnits="FEET/MINUTE" and
 411 nativeScale="100".

412 3.5.7 Data Item attribute: statistic

413 The statistic attribute indicates that the data has been processed using a statistical operation
 414 like average, mean, or root square. statistic may be reported for any Sample type
 415 DataItem. These values are calculated values generated by the Component or Device
 416 providing additional data regarding a DataItem sampled over a specified period of time. All
 417 statistic data is reported in the standard units of the DataItem.

418 The value of statistic is periodically reset. When statistic values are reported as a
 419 Streams value, the value of the statistic **MUST** include an attribute Duration.
 420 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 and Minimum value of a DataItem during the calculation period. Also represents Peak-to-Peak measurement in a waveform.

Statistic	Description
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

421 3.5.8 Data Item attribute: representation

422 The representation attribute defines the format for data consisting of multiple data points
 423 or a file presented as a single DataItem. Each representation will have a unique format
 424 defined for each representation. At this time, the only representations defined are
 425 TIME_SERIES and VALUE.

426 Details on the structure and format of each representation is provided in *Part 3, Section*
 427 *3.8.3* of the MTConnect Standard.

Representation	Description
VALUE	The measured value of a sample. If no representation is specified for a DataItem, the representation MUST 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

428 3.5.9 Data Item Attribute: nativeUnits

429 The nativeUnits attribute adds additional values to the units values. This is the list of
 430 nativeUnits 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.

Native Units	Description
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 minute.
REVOLUTION/SECOND	Rotational velocity in revolution per second
OTHER	Unsupported units

431
432

433 **3.5.10 Data Item Types for SAMPLE Category**

434

435 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 <i>statistic</i> , 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 MUST 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

Data Item type/subtype	Description	Units
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 the Controller.	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.	MILLIMETER
COMMANDED	The position as given by the Controller.	MILLIMETER

Data Item type/subtype	Description	Units
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 MUST 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 MUST 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

Data Item type/subtype	Description	Units
VOLT_ AMPERE	The measure of the apparent power in an electrical circuit, equal to the product of root-mean-square (RMS) voltage and RMS current' (commonly referred to as VA)	VOLT_ AMPERE
VOLT_ AMPERE_ REACTIVE	The measurement of 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 <i>statistic</i> , 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

436

437

438 3.5.11 Data Item Types for EVENT Category

439

440 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. <i>Rel 1.1.</i>
ACTIVE_ AXES	The set of axes associated with a Path that the Controller is controlling. If this DataItem is not provided, it will be assumed the Controller is controlling all axes.
AVAILABILITY	Represents the ability of a Component to communicate. This MUST be provided for a Device and MAY be provided for any other Component. AVAILABLE or UNAVAILABLE .
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 MUST 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. Block contains the entire expression for a line of program code.
CODE	DEPRECATED. <i>Rel 1.1.0</i>

Data Item type/subtype	Description
CONTROLLER_MODE	The current mode of the Controller. AUTOMATIC, MANUAL, MANUAL_DATA_INPUT, 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 using the right hand rule convention as defined in <i>Appendix B</i> . 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 MUST cease operation).
EXECUTION	The execution status of the Controller. READY, ACTIVE, INTERRUPTED, FEED_HOLD, 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.
PALLET_ID	The identifier for the pallet currently in use for a given Path
PART_COUNT	The current count of parts produced as represented by the Controller. MUST 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. <i>Rel. 1.1.</i>
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 an individual tool asset.
TOOL_NUMBER	The identifier of a tool provided by the device controller.
WORKHOLDING_ID	The identifier for the workholding currently in use for a given Path

441 3.5.12 Data Item Types for CONDITION Category

442
 443 Condition is a DataItem that indicates the device's health and ability to operate. They are
 444 reported differently than Samples or Events: they **MUST** be reported as NORMAL,
 445 WARNING, FAULT, or UNAVAILABLE. Unlike the other two categories, a Component or
 446 Device **MAY** have a Condition type DataItem that has multiple concurrently active
 447 values at any point in time. Additionally, these items **MAY** be further defined to provide
 448 differentiation for different condition states; example an AMPERAGE *Condition* may
 449 differentiate between HIGH amperage and LOW amperage. These differences are further
 450 defined as *qualifier* in Part 3, Section 3.11 of the MTConnect Standard.

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.
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 ingredient within a mixture of ingredients
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 's 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

Data Item type/ qualifier	Description
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

451

452

453

454 3.5.13 Schema Structure for DataItems

455
456 The following document structure defines a typical machine with rotary and linear axes and a
457 controller.

```

458
459     MTConnectDevices
460         Devices
461             Device
462                 Components
463                     Axes
464                         Rotary [C]
465                             DataItems
466                                 DataItem [Cvel]
467                                     Constraints SPINDLE
468                         Linear [X]
469                             DataItems
470                                 DataItem [Xpos]
471                         Linear [Y]
472                             DataItems
473                                 DataItem [Ypos]
474                         Linear [Z]
475                             DataItems
476                                 DataItem [Zpos]
477                 Controller
478                     Path
479                         DataItems
480                             DataItem [mode]
481                             DataItem [execution]

```

482 The above example shows how the various containers make it easier to address individual parts
483 of the XML document. For example, if one wanted to retrieve only the DataItems for the
484 Controller, you can express this using the following XPath:
485 //Controller/DataItems/*. If you were interested in retrieving only the *subcomponents*
486 of the Axes component, you would write the following XPath: //Axes/Components/*.

487 3.6 Component Types and *Subcomponents*

488 Component is an abstract type that allows for extensibility. As the specification progresses,
489 more component types will be added to support new devices and parts of new devices. Some
490 examples of component types are Axes, Controller, and Systems. Any of these
491 component types can have data items and *subcomponents*. Appendix B contains reference
492 models for common equipment to guide developers in implementing MTConnect on their
493 devices.

494 The Component types presently define include:

495

496 3.6.1 Axes

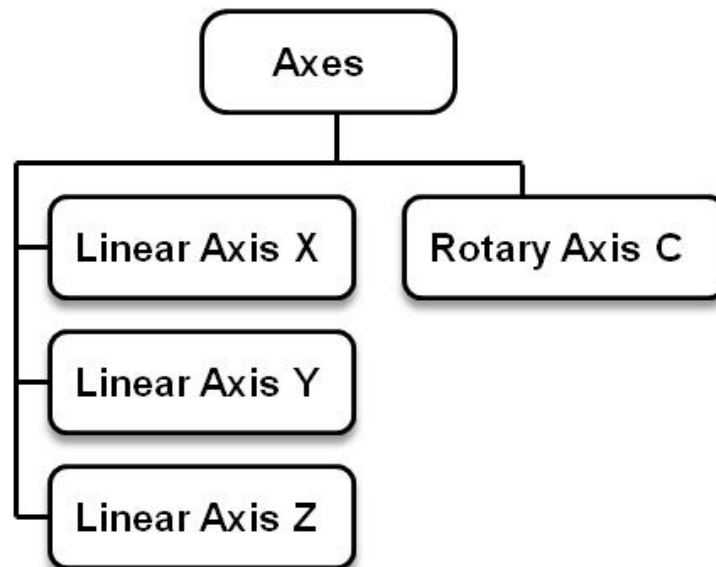
497 Axes is the root of all device components that have linear or rotational motion. Currently there
 498 are only Linear and Rotary axes supported and when axes are defined the Axes component
 499 **MUST** contain at least one Linear or Rotary axis. The Linear axes **MUST** be named X,
 500 Y, Z with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

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

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

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

516



517

518 **Figure 9: Axes Example With Three Linear Axes and one Rotary Axis**

519

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

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

528

529 **3.6.2 Controller**

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

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

539 **3.6.2.1 Path**

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

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

547 **3.6.3 Power- DEPRECATED in Rel. 1.1**

548 **NOTE:** Power as an indication of Availability will be changed to a DataItem called
 549 Availability and electrical current and power consumption will be represented by the
 550 Electric system, see 3.6.9.5 *Electric* below.

551 **3.6.4 Door**

552 This component represents a door closure that can be opened or closed. It **MUST** have a
 553 DataItem called DoorState to indicate if it is opened, closed or unlatched. A device may
 554 contain multiple door components.

555 3.6.5 Actuator

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

560 3.6.6 Sensor

561 Sensor is an abstract type component that provides measurement data related to a Device or
 562 Component. Depending on the type of data provided by the sensor, it may be modeled in the
 563 XML schema in different ways. However, it will always be modeled to associate the data
 564 contained in Sensor with the Component XML Element to which the data is most closely
 565 associated.

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

572 Sensor **MUST NOT** be modeled in the plural. Sensor will always refer to the *sensor*
 573 *interface*. Each *sensor interface* may have multiple *sensing elements*; each representing the data
 574 for a variety of measured values.

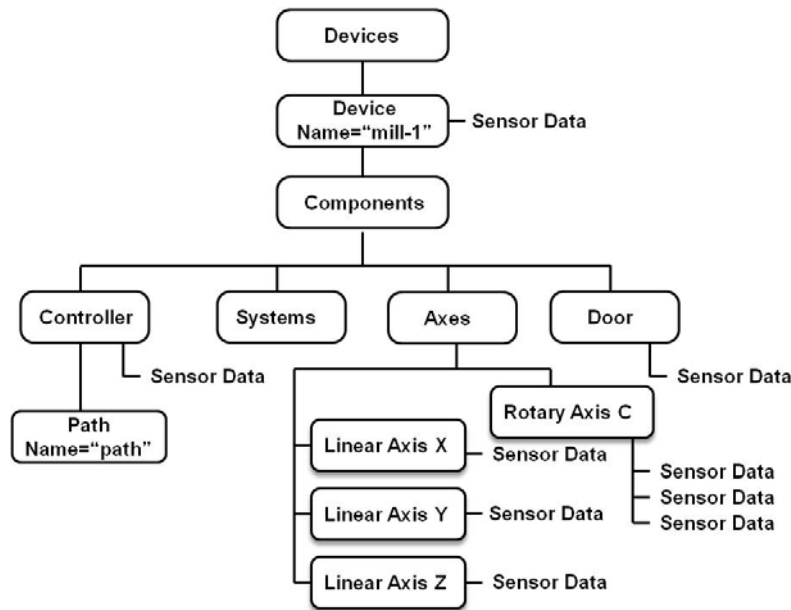
575 3.6.6.1 Sensor data

576 The most basic implementation of a *sensing element* is the providing of a measured value
 577 associated with a Component which is the Sensor data. An example would be the
 578 measured value of the Temperature of the spindle (Rotary Axis C). This would be
 579 represented as a DataItem called Temperature that is associated with the Rotary Axis C as
 580 follows:

```
581     <Components>
582     <Axes
583     <Components>
584     <Rotary id="c" name="C">
585     <DataItems>
586     <DataItem type="TEMPERATURE" id="ctemp" category="SAMPLE"
587     name="Stemp" units="DEGREE"/>
588     </DataItems>
589     </Rotary>
590     </Components>
591     </Axes>
592 </Components>
```

594

595 A sensor may measure values associated with any Component, Sub-Component, or
 596 Device. Some examples of how sensor data may be modeled are represented in Figure 6
 597 below:



598
 599 **Figure 10: Sensor Data Associations**

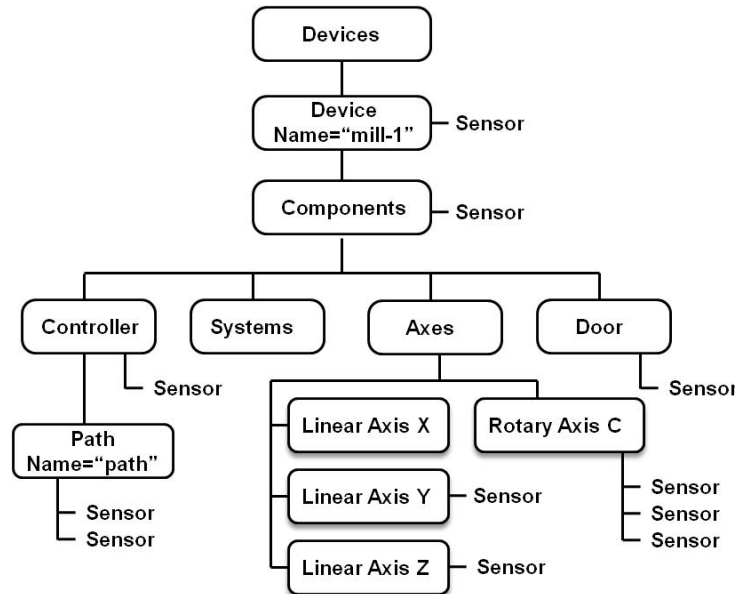
600 3.6.6.2 Sensor Interface

601 *Sensing element(s)* are most typically connected to a *sensor interface*. The *sensor interface*
 602 provides additional information concerning the *sensing element(s)*.

603
 604 Typical functions of the *sensor interface* include:

- 605
- 606 • convert low level signals from the *sensing elements* into data that can be used by other
 607 devices. (Example: Convert a non-linear millivolt signal from a temperature sensor into
 608 a scaled temperature value that can be transmitted to another device.)
- 609
- 610 • process *sensing element* data into calculated values. (Example: temperature sensor data
 611 is converted into calculated values of average temperature, maximum temperature,
 612 minimum temperature, etc.)
- 613
- 614 • provide calibration and configuration information associated with each *sensing element*
 615
- 616 • monitor the health and integrity of the *sensing elements* and the *sensor interface*.
 617 (Example: The *sensor interface* may provide diagnostics on each *sensing element* (e.g.
 618 open wire detection) and itself (e.g. measure internal temperature of the *sensor*
 619 *interface*).
- 620

621 The *sensor interface* is modeled in the XML schema as a Component called Sensor .
 622 Sensor **SHOULD** be modeled in the XML schema so that the Sensor is represented as part
 623 of the Component to which it is most closely associated.
 624
 625 Sensor may be associated with any Component, Sub-Component, or Device. Some
 626 examples of where a sensor may be modeled are represented in Figure 7 below:
 627



628

629

Figure 11: Sensor Associations

630 When a Sensor is modeled as a Component, it **MAY** have its own uuid so it can be
 631 tracked throughout its lifetime.

632

633 The following examples demonstrate how Sensor may be modeled in the XML schema
 634 differently based on how the sensor functions within the overall Device.

635

636 Example#1: If Sensor provides vibration measurement data for the spindle, it should be
 637 modeled as a Sensor for Rotary Axis C.

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

```

<Components>
  <Axes>
    <Components>
      <Rotary id="c" name="C">
        <Sensor id="spdlm" name="Spindlemonitor">
          <DataItems>
            <DataItem type="DISPLACEMENT" id="cvib" category="SAMPLE"
              name="svib" units="MILLIMETER"/>
          </DataItems>
        </Sensor>
      </Rotary>
    </Components>
  </Axes>
</Components>
  
```

653 Example#2: If Sensor provides measurement data for multiple Components within a
 654 Device and is not associated with any particular Component, it MAY be modeled in the XML
 655 schema as an independent Component of the Device.

```

656     <Device id="d1" uuid="HM1" name="HMC_3Axis">
657         <Description>3 Axis Mill</Description>
658         <Components>
659             <Sensor id="sensor" name="sensor"/>
660             <DataItems>
661                 <DataItem type="TEMPERATURE" id="sentemp" category="SAMPLE"
662                     name="Sensortemp" units="DEGREE"/>
663             </DataItems>
664         </Components>
665     </Device>
666
  
```

667
 668 While Sensor MAY be modeled in different ways in the XML schema, the measured value of
 669 the *sensing element* **MUST** always be modeled as a DataItem associated with the
 670 Component to which the measured value is most closely associated.

671
 672 Example#3: In this case, Sensor is modeled as a Component within a Device. Its
 673 measured values from the *sensing elements* are associated with other Components in the
 674 Device. The sensor also has internal diagnostics capabilities representing the condition of the
 675 sensor itself.

676
 677

678 The following represents a sensor with two *sensing elements*, one measures spindle vibration and
 679 the other measures the temperature for the X axis. The sensor also has a *sensing element*
 680 measuring the internal temperature of the *sensor interface*.

```

681
682 <Device id="d1" uuid="HM1" name="HMC_3Axis">
683   <Description>3 Axis Mill</Description>
684   <Components>
685     <Sensor id="sens1" name="Sensorunit">
686       <DataItems>
687         <DataItem type="TEMPERATURE" id="sentemp" category="SAMPLE"
688           name="Sensortemp" units="DEGREE"/>
689       </DataItems>
690     </Sensor>
691     <Axes>
692       <Components>
693         <Rotary id="c" name="C">
694           <DataItems>
695             <DataItem type="DISPLACEMENT" id="cvib" category="SAMPLE"
696               name="Svib" units="MILLIMETER"/>
697           </DataItems>
698         </Rotary>
699         <Linear id="x" name="X">
700           <DataItems>
701             <DataItem type="TEMPERATURE" id="xt"
702               category="SAMPLE" name="Xtemp" units="DEGREE"/>
703           </DataItems>
704         </Linear>
705       </Components>
706     </Axes>
707   </Components>
708 </Device>
709

```

710 3.6.6.3 Sensor as a Device

711 A sensor may function as an independent device. In this case, it is not associated with a parent
 712 Device or Component.

713 Examples of a sensor functioning as a Device would be a sensor used to monitor the ambient
 714 temperature of a building or an air quality monitoring system. Another example would be a
 715 vibration monitoring system that is moved from one machine to another. In these cases, the
 716 sensor functions as an intelligent device performing a specific function.

717

718 A sensor functioning as a Device would be modeled in the XML schema as follows:

```

719
720 <Device id="s1" uuid="HM1" name="AMBIENT_MONITOR">
721   <Description>Ambient Temperature Monitor</Description>
722   <DataItems>
723     <DataItem type="TEMPERATURE" id="ambtemp" category="SAMPLE"
724       name="Ambienttemp" units="DEGREE"/>
725   </DataItems>
726 </Device>

```

727 A sensor that is modeled as a device **MUST** have an `uuid` so that it can be uniquely tracked.

728 **3.6.7 Sensor Configuration**

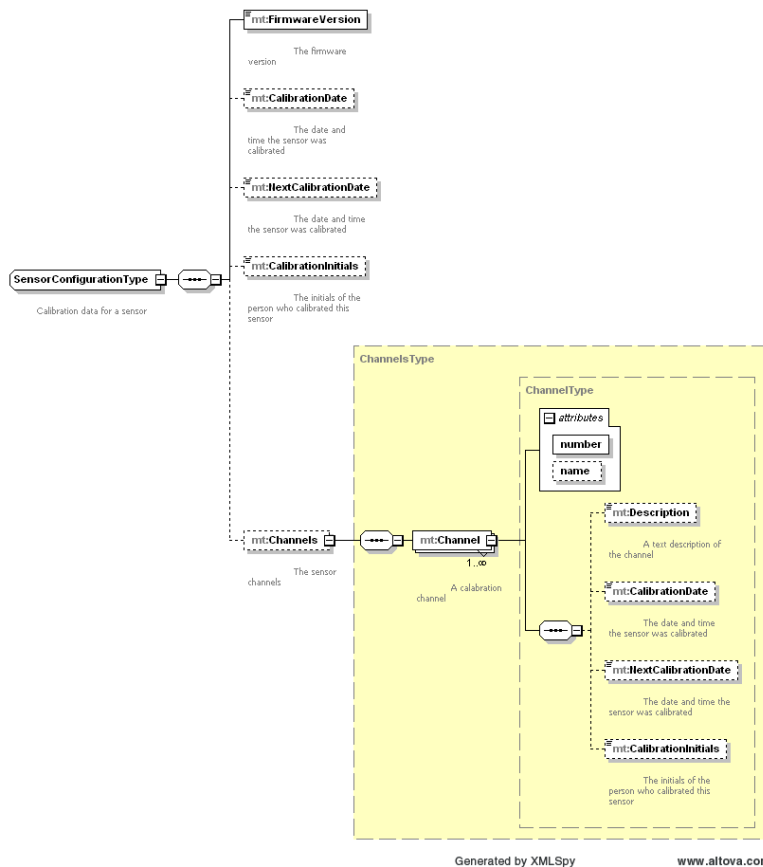
729 When a sensor is modeled in the XML schema as a Component or a Device, it may provide
 730 additional configuration information for the *sensor elements* and the *sensor interface* itself.

731
 732 The Sensor configuration data provides information required for maintenance and support of
 733 the sensor.

734
 735 Sensor configuration data is *only* available when the sensor is modeled as a Component or a
 736 Device. For details on the modeling of Configuration data in the XML schema, see *Part*
 737 *2, Section 3.4.7.1 Component Configuration*. Details specific to
 738 SensorConfigurationType are provided below.

739 When Sensor represents the *sensor interface* for multiple *sensing element(s)*, each *sensing*
 740 *element* is represented by a Channel . Each Channel represents one *sensing element* and can
 741 have its own attributes and Configuration data.

742
 743



744
 745
 746
 747
 748

Figure 7: Configuration Data for Sensors

Element	Description	Occurrence
Configuration (SensorConfigurationType)	An element that can contain descriptive content defining the configuration information for Sensor. For Sensor, the valid configuration is SensorConfiguration. SensorConfiguration provides data from a subset of items commonly found in a 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). MTCConnect 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

749

750 **3.6.7.1 SensorConfiguration Elements**

751

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 MUST be represented in the W3C ISO 8601 format	0..1
NextCalibrationDate	Date upon which the sensor is next scheduled to be calibrated. Dates MUST 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	When Sensor represents multiple <i>sensing elements</i> , each <i>sensing element</i> is represented by a Channel for the Sensor.	0..1

752

753 **3.6.7.2 Sensor Channel Attributes**

754 Channel represents each *sensing element* connected to a *sensor interface*. Each Sensor
755 Channel has the following composition:

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

756

757

758 **3.6.7.3 Sensor Channel Elements**

759

Element	Description	Occurrence
Description	An XML element that can contain any descriptive content. This can contain information about the <i>sensor element</i> and manufacturer specific details.	0..1
CalibrationDate	Date upon which the <i>sensor element</i> was last calibrated. Dates MUST be represented in the W3C ISO 8601 format	0..1
NextCalibrationDate	Date upon which the <i>sensor element</i> is next scheduled to be calibrated. Dates MUST 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

760

761 The following is an example of the configuration data for Sensor that is modeled as a
762 Component. It has Configuration data for the *sensor interface*, one Channel named
763 A/D:1, and two DataItems – Voltage (as a Sample) and Voltage (as a Condition or
764 alarm).

765

```

766     <Sensor id="sensor" name="sensor">
767       <Configuration>
768         <SensorConfiguration>
769           <FirmwareVersion>2.02</FirmwareVersion>
770           <CalibrationDate>2010-05-16</CalibrationDate>
771           <NextCalibrationDate>2010-05-16</NextCalibrationDate>
772           <CalibrationInitials>WS</CalibrationInitials>
773           <Channels>
774             <Channel number="1" name="A/D:1">
775               <Description>A/D With Thermister</Description>
776             </Channel>
777           </Channels>
778         </SensorConfiguration>
779       </Configuration>
780       <DataItems>
781         <DataItem category="CONDITION" id="senvc" type="VOLTAGE" />
782         <DataItem category="SAMPLE" id="senv" type="VOLTAGE" units="VOLT"
783           subType="DIRECT" />
784       </DataItems>
785     </Sensor>

```

786

787 **3.6.8 Sensor Types**

788 Types of measurements provided by `SENSOR` include:

789 **3.6.8.1 Thermostat (DEPRECATED in Rel. 1.2. See TEMPERATURE)**

790 **3.6.8.2 Vibration (DEPRECATED in Rel. 1.2. See DISPLACEMENT,**
791 **FREQUENCY, etc.)**

792 **3.6.8.3 Acceleration**

793 The measurement of the linear acceleration of an object.

794 **3.6.8.4 Angular Acceleration**

795 The measurement of the acceleration of a rotating object

796 **3.6.8.5 Angular Velocity**

797 The measurement of the velocity of a rotating object

798 **3.6.8.6 Amperage**

799 The measurement of electrical current flow.

800 **3.6.8.7 Angle**

801 The measurement of angular position of an object.

802 **3.6.8.8 Concentration**

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

804 **3.6.8.9 Conductivity**

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

806 **3.6.8.10 Direction**

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

809 **3.6.8.11 Displacement**

810 The measurement of the distance of movement of an object.

811 **3.6.8.1 Electrical Energy**

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

813 **3.6.8.2 Flow**

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

815 **3.6.8.3 Frequency**

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

- 817 **3.6.8.4 Fill Level**
818 The measurement of the amount of a substance remaining compared to the planned maximum of
819 that substance.
- 820 **3.6.8.5 Linear Force**
821 The magnitude of push or pull introduced by an actuator or exerted on an object.
- 822 **3.6.8.6 Load**
823 The measurement of the percentage of the standard rating of a device.
- 824 **3.6.8.7 Mass**
825 The measurement of the mass of an object(s) or an amount of material.
- 826 **3.6.8.8 PH**
827 The measurement of the acidity or alkalinity.
- 828 **3.6.8.9 Pressure**
829 The measurement of the force per unit area exerted by a gas or liquid.
- 830 **3.6.8.10 Position**
831 The measurement of an object's position relative to a coordinate system.
- 832 **3.6.8.11 Power Factor**
833 The measurement of the ratio of real power flowing to a load to the apparent power in an AC
834 circuit.
- 835 **3.6.8.12 Resistance**
836 The measurement of the degree to which an object opposes an electric current through it
- 837 **3.6.8.13 Rotary Velocity**
838 The measurement of the rotational speed of a rotating object.
- 839 **3.6.8.14 Sound Level**
840 The measurement of sound level.
- 841 **3.6.8.15 Strain**
842 The measurement of the amount of deformation per unit length of an object.
- 843 **3.6.8.16 Temperature**
844 The measurement of the temperature of an object.
- 845 **3.6.8.17 Time**
846 The measurement of time: may be reported as accumulated time associated with a component or
847 clock time.
- 848 **3.6.8.18 Tilt**
849 The measurement of the angular displacement of an object.

850 **3.6.8.19 Torque**

851 The measurement of torque applied to or by an object.

852 **3.6.8.20 Volt Ampere (VA)**853 The measure of the apparent power in an electrical circuit, equal to the product of root-mean-
854 square (RMS) voltage and RMS current.855 **3.6.8.21 Volt Ampere Reactive (var)**

856 The measurement of reactive power in an AC electric power system

857 **3.6.8.22 Velocity**

858 The measurement of the linear velocity of an object.

859 **3.6.8.23 Viscosity**

860 The measurement of a fluid's resistance to flow.

861 **3.6.8.24 Voltage**

862 The measurement of electrical potential between two points

863 **3.6.8.25 Wattage**864 The measurement of power consumed or dissipated by an electrical circuit or device
865866 **3.6.9 Systems**867 A component similar to *Axes* that groups *subcomponents* that comprise complex *Components*
868 that are not easily deconstructed. *Systems* will be used to represent general information about
869 the health and viability of all of its parts and sub-parts.870 **3.6.9.1 Hydraulic**871 A hydraulic system comprises all the parts involved in moving and distributing pressurized liquid
872 for the purpose of delivering a source of power to specific types of actuators.873 **3.6.9.2 Pneumatic**874 A pneumatic system comprises all the parts involved in moving and distributing pressurized gas
875 regardless of purpose or activity.876 **3.6.9.3 Coolant**

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

878 **3.6.9.4 Lubrication**879 The lubrication system comprises all the parts involved in distribution and management of the
880 lubricants.

881

882 **3.6.9.5 Electric**

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

887 **4 Component and Data Item Relationships**

888 This section will discuss the association between Component, DataItem, and DataItem
 889 categories (Events, Condition, and Samples). For each Component, there are a limited
 890 set of allowable *subcomponents* and a limited set of DataItems. For example, an Axes
 891 component may not have a Device or a Controller as a child, and it may not have Block
 892 as a DataItem type, since it is incapable of running a program.

893 Many types of DataItem can be applied to a wide variety of Component (s). In the
 894 sections below, only those types of DataItem that are specific to each Component will be
 895 defined. By inference, all other types of DataItem may be applied to these Component (s)
 896 as required.

897 **4.1 Device**

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

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

903 **4.1.1 Device DataItems**

- 904 • EMERGENCY_STOP - The emergency stop state of the machine or device.
- 905 • AVAILABILITY - **Required**

906 **4.1.2 Components of Device**

- 907 • Axes
- 908 • Controller
- 909 • Systems
- 910 • Door
- 911 • Actuator
- 912 • Sensor

913 **4.2 Common Components and Related Data Items**

914 A common set of DataItems have been defined to provide a wide variety of information about
 915 a machine or process. Any DataItem can be used with any Device or Component
 916 providing that the standard naming conventions are implemented. Any Component **MAY** also
 917 include an arbitrary set of sensors that may be modeled as either a *subcomponent* or a
 918 DataItem. A Sensor may be an external device that will collect data and report it to the
 919 Agent.

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

923 Only the types of `DataItem` unique to each Component are detailed below. It can be
 924 assumed that all other type of `DataItem` can be applied to any of the Components.

925 **4.2.1 Axes**

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

928 **4.2.1.1 Axes DataItems**

- 929 • ~~GLOBAL_POSITION~~ - DEPRECATED in Rel 1.1
- 930 • ~~PATH_FEEDRATE~~ - Moved to Path
- 931 • ~~ACCELERATION~~ - Moved to Path
- 932 • ~~VELOCITY~~ - Moved to Path

933 **4.2.1.2 Subcomponents of Axes**

- 934 • Linear
 - 935 • Rotary
 - 936 • ~~Spindle~~ - DEPRECATED in Rel 1.1
- 937

938 **4.2.2 Linear (Subcomponent of Axes)**

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

941 **4.2.2.1 Linear Axes' DataItems (Sample and Event)**

- 942 • AXIS_FEEDRATE
- 943 • DIRECTION
- 944 • POSITION
- 945 • ~~SLAVE_OF_AXIS~~ (DEPRECATED in Rel. 1.1)
- 946 • LINEAR_FORCE
- 947 • VELOCITY

948 **4.2.2.2 Linear Axes' Condition**

- 949 • POSITION
 - 950 • LINEAR_FORCE
 - 951 • VELOCITY
- 952

953 **4.2.3 Rotary (*Subcomponent of Axes*)**

954 A rotary axis revolves around a line or vector.

955 **4.2.3.1 Rotary Axes' DataItems (Sample and Event)**

- 956 • ANGLE
- 957 • ANGULAR_ACCELERATION
- 958 • ANGULAR_VELOCITY
- 959 • DIRECTION
- 960 • ROTARY_MODE
- 961 • ROTARY_VELOCITY
- 962 • ~~SLAVE_OF_AXIS~~ **DEPRECATED in Rel 1.1**
- 963 • ~~SPINDLE_SPEED~~ **DEPRECATED in Rel 1.2. Replaced by**
- 964 **ROTARY_VELOCITY**
- 965 • TORQUE

966 **4.2.3.2 Rotary Axes' Condition**

- 967 • ANGLE
- 968 • ANGULAR_ACCELERATION
- 969 • ANGULAR_VELOCITY
- 970 • ROTARY_VELOCITY
- 971 • TORQUE

972 **4.2.4 Controller**

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

976 **4.2.4.1 Subcomponents of Controller**

- 977 • Path

978 **4.2.4.2 Controller DataItems (Sample and Event)**

- 979 • ~~CODE~~ **DEPRECATED in Rel 1.1**
- 980 • CONTROLLER_MODE
- 981 • EXECUTION
- 982 • EMERGENCY_STOP
- 983 • MESSAGE
- 984 • PALLET_ID
- 985 • PART_COUNT
- 986 • PART_ID
- 987 • PROGRAM
- 988 • ~~TOOL_ID~~ **DEPRECATED in Rel 1.2.**
- 989 • TOOL_ASSET_ID
- 990 • WORKHOLDING_ID
- 991

992 **4.2.4.3 Controller Condition**

- 993 • COMMUNICATIONS
- 994 • HARDWARE
- 995 • LOGIC_PROGRAM
- 996 • MOTION_PROGRAM
- 997 • SYSTEM

998 **4.2.5 Path (Subcomponent of Controller)**

999 A Path represents the motion of a control point as it moves through space as controlled by a set
 1000 of control instructions (i.e. vector move). When Path is not defined, DataItems relative to
 1001 the Path **MAY** be reported for the Controller.

1002 **4.2.5.1 Path DataItems (Sample and Event)**

- 1003 • ACTIVE_AXES
- 1004 • AXIS_COUPLING
- 1005 • BLOCK
- 1006 ~~• CODE DEPRECATED~~
- 1007 • COUPLED_AXES
- 1008 • CONTROLLER_MODE
- 1009 • EMERGENCY_STOP
- 1010 • EXECUTION
- 1011 • LINE
- 1012 • PALLET_ID
- 1013 • PART_COUNT
- 1014 • PART_ID
- 1015 • PATH_FEEDRATE
- 1016 • PATH_POSITION
- 1017 • PROGRAM
- 1018 • ~~TOOL_ID~~ DEPRECATED in Rel 1.2.
- 1019 • TOOL_ASSET_ID
- 1020 • VELOCITY
- 1021 • WORKHOLDING_ID

1022 **4.2.5.2 Path Condition**

- 1023 • MOTION_PROGRAM

1024 **4.2.6 ~~Power~~ DEPRECATED in Rel 1.1**

1025

1026 **4.2.7 Sensors**

1027 Sensor is a component that may or may not be integral to a parent component or device.
 1028 When Sensor is not integral to a parent device or component – it can function as a device.
 1029 Sensor data **MUST** be associated with its most relevant Component and **MUST** be represented
 1030 as a DataItem for that Component.

1031 **4.2.7.1 Sensor Condition**

- 1032 • COMMUNICATION
- 1033 • HARDWARE
- 1034

1035 **4.2.8 ~~Thermostat~~ DEPRECATED in REL 1.2. Replaced with a DataItem called** 1036 **Temperature**

1037 ~~A sensor capable of measuring the temperature of a component. The temperature is always given~~
 1038 ~~in Celsius.~~

1039 **4.2.8.1 DataItem types**

- 1040 • ~~TEMPERATURE~~

1041 **4.2.8.2 Condition types**

- 1042 • ~~COMMUNICATION~~
- 1043 • ~~HARDWARE~~
- 1044 • ~~TEMPERATURE~~

1045 **4.2.9 ~~Vibration~~ DEPRECATED in REL 1.2. Replaced with DataItems to measure** 1046 **vibration (Displacement, Frequency, etc).**

1047 ~~A sensor capable of measuring the vibration of a component.~~

1048 **4.2.9.1 DataItem types**

- 1049 • ~~ACCELERATION~~
- 1050 • ~~DISPLACEMENT~~
- 1051 • ~~FREQUENCY~~
- 1052 • ~~VELOCITY~~

1053 **4.2.9.2 Condition types**

- 1054 • ~~ACCELERATION~~
- 1055 • ~~COMMUNICATION~~
- 1056 • ~~DISPLACEMENT~~
- 1057 • ~~HARDWARE~~
- 1058 • ~~VIBRATION~~
- 1059 _____

1060 **4.2.10 Pressure** DEPRECATED in *REL 1.2*. Replace with **DataItem Pressure**

1061 A sensor capable of measuring the pressure.

1062 **4.2.10.1 DataItem types**

1063 • ~~PRESSURE~~

1064 **4.2.10.2 Condition types**

1065 • ~~COMMUNICATION~~

1066 • ~~HARDWARE~~

1067 • ~~PRESSURE~~

1068

1069 **4.2.11 Door**

1070 A opening that can be closed.

1071 **4.2.11.1 Door DataItems (Sample and Event)**

1072 • DOOR_STATE

1073 **4.2.11.2 Door Condition**

1074 • DOOR_STATE

1075 • COMMUNICATIONS

1076 • HARDWARE

1077 **4.2.12 Actuator**

1078 A mechanical device for moving or controlling a mechanism or system.

1079 **4.2.12.1 Acutator DataItems (Sample and Event)**

1080 • ACTUATOR_STATE

1081 **4.2.12.2 Actuator Condition**

1082 • COMMUNICATIONS

1083 • HARDWARE

1084 **4.2.13 ~~Spindle~~ – DEPRECATED** in *Rel. 1.1*

1085 ~~The spindle is a rotational axis that revolves at high speed and has its speed expressed in~~

1086 ~~REVOLUTION/MINUTE~~

1087 **4.2.14 Systems**

1088 The Systems component is a place holder for all the system types.

1089 **4.2.14.1 Subcomponents of Systems**

1090 • Hydraulic

1091 • Pneumatic

1092 • Coolant

1093 • Lubrication

1094 • Electric

1095 **4.2.15 Hydraulic (*Subcomponent of Systems*)**

1096 A component representing the hydraulics and hydraulic distribution system of a device.

1097 **4.2.15.1 Hydraulic Condition**

- 1098 • COMMUNICATIONS
- 1099 • HARDWARE

1100 **4.2.16 Pneumatic (*Subcomponent of Systems*)**

1101 A component representing the pneumatics and compressed air distribution system of a device.

1102 **4.2.16.1 Pneumatic Condition**

- 1103 • COMMUNICATIONS
- 1104 • HARDWARE

1105 **4.2.17 Coolant (*Subcomponent of Systems*)**

1106 A Component representing the coolant and coolant distribution system of a device.

1107 **4.2.17.1 Coolant DataItems (Sample and Event)**

- 1108 • CONCENTRATION
- 1109 • CONDUCTIVITY
- 1110 • PH
- 1111 • VISCOSITY

1112 **4.2.17.2 Coolant Condition**

- 1113 • COMMUNICATIONS
- 1114 • HARDWARE
- 1115 • CONCENTRATION
- 1116 • CONDUCTIVITY
- 1117 • PH
- 1118 • VISCOSITY

1119 **4.2.18 Lubrication (*Subcomponent of Systems*)**

1120 A Component representing the lubricant and lubrication distribution system of a device.

1121 **4.2.18.1 Lubrication DataItems (Sample and Event)**

- 1122 • PH
- 1123 • VISCOSITY

1124 **4.2.18.2 Lubrication Condition**

- 1125 • COMMUNICATIONS
- 1126 • HARDWARE
- 1127 • PH
- 1128 • VISCOSITY
- 1129

- 1130 **4.2.19 Electric (*Subcomponent of Systems*)**
- 1131 A Component representing the electrical supply for a device.
- 1132 **4.2.19.1 Electrical DataItems (Sample and Event)**
- 1133 • AMPERAGE
- 1134 • ELECTRICAL_ENERGY
- 1135 • FREQUENCY
- 1136 • POWER_FACTOR
- 1137 • POWER_STATE
- 1138 • VOLTAGE
- 1139 • VOLT_AMPERE
- 1140 • VOLT_AMPERE_REACTIVE
- 1141 • WATTAGE
- 1142 **4.2.19.2 Electric Condition**
- 1143 • AMPERAGE
- 1144 • FREQUENCY
- 1145 • VOLTAGE
- 1146 • WATTAGE

1147 5 Annotated XML Examples

1148 5.1 Simplest Device

1149 For the simplest possible device, we are modeling a saw that has only an `Availability` (the
1150 minimal set of `DataItem`). To retrieve this information, we send the following request to the
1151 *Agent*:

```
1152 http://10.1.23.10/ LinuxCNC/probe
```

1153 The *Agent* responds as follows:

```
1154 1. <?xml version="1.0" encoding="UTF-8"?>
1155 2. <MTConnectDevices xmlns:m="urn:mtconnect.com:MTConnectDevices:0.9"
1156    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
1157    xmlns="urn:mtconnect.com:MTConnectDevices:0.9"
1158    xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:0.9
1159    /schemas/MTConnectDevices.xsd">
1160 3.   <Header sender="10.1.23.10" bufferSize="100000"
1161      creationTime="2008-07-07T23:07:50-07:00" version="0.9"
1162      instanceId="1214527986"/>
```

1163
1164 Line 3 provides the `instanceId` as a unique number for this request. For this example, the
1165 *Agent* does not persist the `Samples`, `Events`, and `Condition`. Therefore, this number will
1166 change every time that it is recorded. The `bufferSize` indicates that this *Agent* is capable of
1167 storing 100,000 `DataItem` of category `Sample`, `Event`, and `Condition`.

```
1168 4.   <Devices>
1169 5.     <Device iso841Class="6" uuid="linux-01" name="LinuxCNC"
1170       sampleInterval="100.0" id="d">
1171 6.     <Description manufacturer="NIST" serialNumber="01"/>
```

1172 The above device description includes the unique `id` and a sample interval of ten times per
1173 second. Since there are no telemetry data being collected, sampling at once per second is
1174 typically adequate.

```
1175 7.     </Components>
1176 8.     <DataItems>
1177 9.       <DataItem type="AVAILABILITY" name="avail" category="EVENT"
1178         id="a"/>
1179 10.    </DataItems>
```

1180 As was stated previously, the device is only required to have one `DataItem` and it is of the
1181 type `AVAILABILITY` which **MUST** report the device's represent ability to communicate. The
1182 `DataItem` on line 9 has an `id` of "a". This will allow events responding to this `DataItem` to
1183 be easily associated.

1184

```

1185 11.      </Components>
1186 12.      </Device>
1187 13.    </Devices>
1188 14.  </MTConnectDevices>
1189

```

1190 Lines 11 through 14 terminate each element type and close the document.

1191 **5.2 More Complex Example of Probe**

1192 The Sample was generated with the following request:

```
1193 http://10.1.23.5/LinuxCNC/probe
```

1194 The following is an example of a 3 axis mill simulation. The mill has three linear axes and one
1195 rotary axis (spindle):

```

1196 1.  <?xml version="1.0" encoding="UTF-8"?>
1197 2.  <MTConnectDevices xmlns:m="urn:mtconnect.com:MTConnectDevices:0.9"
1198    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
1199    xmlns="urn:mtconnect.com:MTConnectDevices:0.9"
1200    xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:0.9
1201    /schemas/MTConnectDevices.xsd">
1202 3.    <Header sender="10.1.23.5" bufferSize="100000" creationTime=
1203      "2008-07-07T23:07:50-07:00" version="0.9"
1204      instanceId="1214527986"/>
1205 4.    <Devices>
1206 5.      <Device iso841Class="6" uuid="linux-01" name="LinuxCNC"
1207        sampleRate="100.0" id="d1">
1208

```

1209 Here we provide the top level container `Devices` and the information on the `Device`.

```

1210 6.      <Description manufacturer="NIST" serialNumber="01"/>
1211 7.      <DataItems>
1212 8.        <DataItem type="AVAILABILITY" name="avail" category="EVENT"
1213          id="a"/>
1214 9.      </DataItems>
1215 10.    <Components>
1216 11.    <Axes name="Axes" id="3">
1217

```

1218 On line 11 we introduce the collection of `Axes`. The `Axes` component is a special component
1219 that acts as an abstract component as well as a collection. The `Axes` component contains
1220 various `DataItems` that have a global context; they are not associated with any one axis but
1221 they go across all axes.

1222


```

1223 12.      <Components>
1224 13.          <Rotary name="C" id="c1">
1225 14.              <DataItems>
1226 15.                  <DataItem type="ROTARY_VELOCITY" name="Cspeed" category="SAMPLE"
1227                     id="c2" nativeUnits="REVOLUTION/MINUTE" subType="ACTUAL"
1228                     units="REVOLUTION/MINUTE">
1229 16.                      <Source>Sspeed</Source>
1230 17.              </DataItem>
1231 18.                  <DataItem type="ROTARY_MODE" name="Cmode" category="EVENT"
1232                     id="c3">
1233 19.                      <Constraints>
1234 20.                          <Value>SPINDLE</Value>
1235 21.                      </Constraints>
1236 22.                  </DataItem>
1237 23.              </DataItems>
1238 24.          </Rotary>
1239

```

1240 The spindle component (Rotary Axis C) declared on line 13 is the C axis and has spindle specific
1241 DataItems.

```

1242 25.      <Linear name="X" id="x1">
1243 26.          <DataItems>
1244 27.              <DataItem type="POSITION" name="Xact" category="SAMPLE" id="x2"
1245                     nativeUnits="MILLIMETER" subType="ACTUAL" units="MILLIMETER"/>
1246 28.              <DataItem type="POSITION" name="Xcom" category="SAMPLE" id="x3"
1247                     nativeUnits="MILLIMETER" subType="COMMANDED"
1248                     units="MILLIMETER"/>
1249 29.          </DataItems>
1250 30.      </Linear>
1251 31.      <Linear name="Y" id="y1">
1252 32.          <DataItems>
1253 33.              <DataItem type="POSITION" name="Yact" category="SAMPLE" id="y2"
1254                     nativeUnits="MILLIMETER" subType="ACTUAL"
1255                     units="MILLIMETER"/>
1256 34.              <DataItem type="POSITION" name="Ycom" category="SAMPLE" id="y3"
1257                     nativeUnits="MILLIMETER" subType="COMMANDED"
1258                     units="MILLIMETER"/>
1259 35.          </DataItems>
1260 36.      </Linear>
1261 37.      <Linear name="Z" id="z1">
1262 38.          <DataItems>
1263 39.              <DataItem type="POSITION" name="Zact" category="SAMPLE" id="z2"
1264                     nativeUnits="MILLIMETER" subType="ACTUAL" units="MILLIMETER"/>
1265 40.
1266

```

```

1267         <DataItem type="POSITION" name="Zcom"
1268             category="SAMPLE" id="z3"
1269             nativeUnits="MILLIMETER" subType="COMMANDED"
1270             units="MILLIMETER" />
1271 41.     </DataItems>
1272 42.     </Linear>
1273 43.     </Components>
1274 44.     </Axes>
1275

```

1276 Lines 25, 31, and 37 define the three linear axes X, Y, and Z respectively. In this example
 1277 device, the *Agent* is only collecting the actual and commanded positions.

1278 The Controller is capable of providing the program name, block, and the current line being
 1279 executed:

```

1280 45.     <Controller name="Controller" id="cn8">
1281 46.     <Components>
1282 47.     <Path id="pth1" name="path">
1283 48.     <DataItems>
1284 49.     <DataItem type="LINE" name="line" category="EVENT" id="p1"/>
1285 50.     <DataItem type="CONTROLLER_MODE" name="mode" category="EVENT"
1286             id="p2"/>
1287 51.     <DataItem type="PROGRAM" name="program" category="EVENT"
1288             id="p3"/>
1289 52.     <DataItem type="EXECUTION" name="execution" category="EVENT"
1290             id="p4"/>
1291 53.     <DataItem type="PATH_FEEDRATE" name="feedrate"
1292 54.     category="SAMPLE" id="p5" units="MILLIMETER/SECOND"
1293             nativeUnits="MILLIMETER/SECOND" />
1294 55.     <DataItem type="PATH_POSITION" name="position"
1295 56.     category="SAMPLE" id="p6" units="MILLIMETER_3D"
1296 57.     nativeUnits="INCH_3D" />
1297 58.     </DataItems>
1298 59.     </Path>
1299 60.     </Components>
1300 61.     </Controller>
1301 62.     </Components>
1302         </Device>
1303     </Devices>
1304 63. </MTConnectDevices>
1305

```

1306

Appendices

1307 A. Bibliography

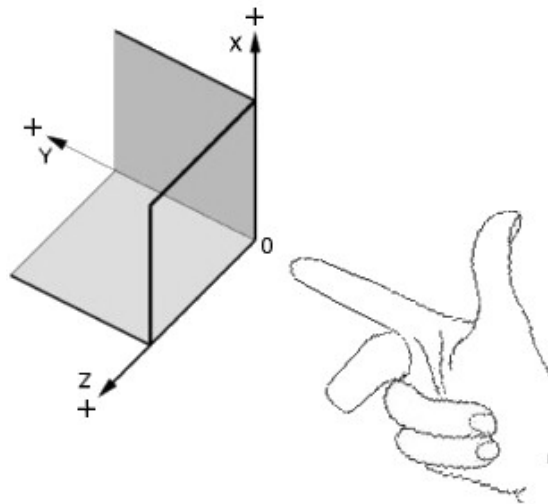
- 1308 1. Engineering Industries Association. *EIA Standard - EIA-274-D*, Interchangeable Variable,
1309 Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically
1310 Controlled Machines. Washington, D.C. 1979.
- 1311 2. ISO TC 184/SC4/WG3 N1089. *ISO/DIS 10303-238*: Industrial automation systems and
1312 integration Product data representation and exchange Part 238: Application Protocols:
1313 Application interpreted model for computerized numerical controllers. Geneva,
1314 Switzerland, 2004.
- 1315 3. International Organization for Standardization. *ISO 14649*: Industrial automation systems
1316 and integration – Physical device control – Data model for computerized numerical
1317 controllers – Part 10: General process data. Geneva, Switzerland, 2004.
- 1318 4. International Organization for Standardization. *ISO 14649*: Industrial automation systems
1319 and integration – Physical device control – Data model for computerized numerical
1320 controllers – Part 11: Process data for milling. Geneva, Switzerland, 2000.
- 1321 5. International Organization for Standardization. *ISO 6983/1* – Numerical Control of
1322 machines – Program format and definition of address words – Part 1: Data format for
1323 positioning, line and contouring control systems. Geneva, Switzerland, 1982.
- 1324 6. Electronic Industries Association. *ANSI/EIA-494-B-1992*, 32 Bit Binary CL (BCL) and 7
1325 Bit ASCII CL (ACL) Exchange Input Format for Numerically Controlled Machines.
1326 Washington, D.C. 1992.
- 1327 7. National Aerospace Standard. *Uniform Cutting Tests - NAS Series: Metal Cutting*
1328 *Equipment Specifications*. Washington, D.C. 1969.
- 1329 8. International Organization for Standardization. *ISO 10303-II*: 1994, Industrial
1330 automation systems and integration Product data representation and exchange Part 11:
1331 Description methods: The EXPRESS language reference manual. Geneva, Switzerland,
1332 1994.
- 1333 9. International Organization for Standardization. *ISO 10303-21*: 1996, Industrial
1334 automation systems and integration -- Product data representation and exchange -- Part
1335 21: Implementation methods: Clear text encoding of the exchange structure. Geneva,
1336 Switzerland, 1996.
- 1337 10. H.L. Horton, F.D. Jones, and E. Oberg. *Machinery's handbook*. Industrial Press, Inc. New
1338 York, 1984.
- 1339 11. International Organization for Standardization. *ISO 841-2001: Industrial automation*
1340 *systems and integration - Numerical control of machines - Coordinate systems and*
1341 *motion nomenclature*. Geneva, Switzerland, 2001.

- 1342 12. ASME B5.57: *Methods for Performance Evaluation of Computer Numerically Controlled*
1343 *Lathes and Turning Centers*, 1998
- 1344 13. ASME/ANSI B5.54: *Methods for Performance Evaluation of Computer Numerically*
1345 *Controlled Machining Centers*. 2005.
- 1346 14. OPC Foundation. *OPC Unified Architecture Specification, Part 1: Concepts Version 1.00.*
1347 *July 28, 2006.*
- 1348 15. IEEE STD 1451.0-2007, *Standard for a Smart Transducer Interface for Sensors and*
1349 *Actuators – Common Functions, Communication Protocols, and Transducer Electronic*
1350 *Data Sheet (TEDS) Formats*, IEEE Instrumentation and Measurement Society, TC-9, The
1351 *Institute of Electrical and Electronics Engineers, Inc., New York, N.Y. 10016, SH99684,*
1352 *October 5, 2007.*
- 1353 16. IEEE STD 1451.4-1994, *Standard for a Smart Transducer Interface for Sensors and*
1354 *Actuators – Mixed-Mode Communication Protocols and Transducer Electronic Data*
1355 *Sheet (TEDS) Formats*, IEEE Instrumentation and Measurement Society, TC-9, The
1356 *Institute of Electrical and Electronics Engineers, Inc., New York, N.Y. 10016, SH95225,*
1357 *December 15, 2004.*
- 1358

1359 B. Machine Tool Modeling

1360 The following section will provide example machine tool configurations and reference
 1361 MTConnect[®] implementations. The following is the recommended machine modeling and
 1362 implementation reference.

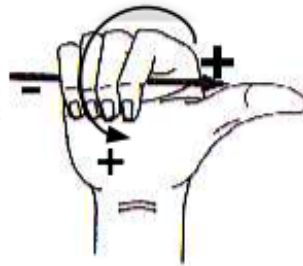
1363 MTConnect utilizes the right hand rule for all coordinate systems representing physical space
 1364 and orientation within a machine. The positive movement is given by extending the first three
 1365 fingers on the right hand and labeling the axes in order of the digits, X, Y, and Z. The fingers
 1366 will point in the positive direction. All linear axes represent a space within a machine that is
 1367 defined by coordinates according to the right hand rule.



1368

1369 **Figure 12: Right Hand Rule Coordinate Planes**

1370 For Rotary axes, the right hand rule defines the direction of rotary movement by wrapping one's
 1371 right-hand fingers around the axis of rotation. Clockwise rotation points the thumb toward the
 1372 person, and counterclockwise rotation points the thumb away. The thumb indicates in the
 1373 positive direction of the vector or axis the hand encircles. All rotational angles and movements
 1374 are given according to the right hand rule for Rotary axes.

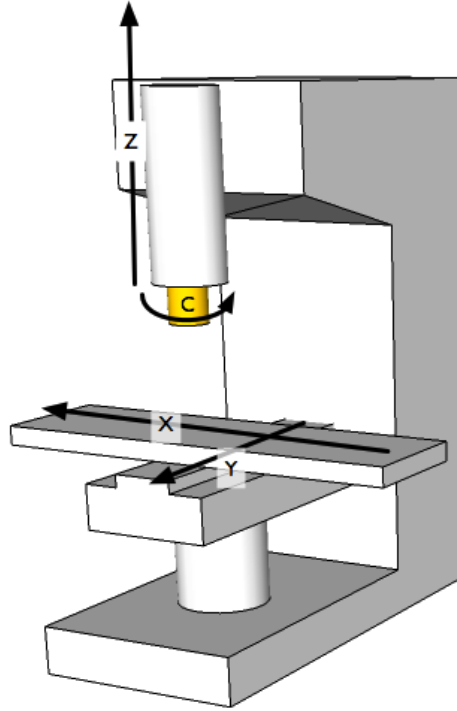


1375

1376 **Figure 13: Rotational Right Hand Rule**

1377 B.1. Vertical Three Axis Mill

1378 This is a simple machine tool with a vertical spindle and a table that can move in two
 1379 dimensions. The modeling always starts with the Linear Z axis that is aligned with the
 1380 primary spindle. The X axis is defined as the longest axis perpendicular to the Z axis. The
 1381 spindle is now defined as a Rotary C axis that rotates around the Z axis.



1382

1383

Figure 14: Three Axis Mill

1384 The right hand rule applies when naming the axes and defining positive motion and rotation. In
 1385 this case the Rotary axis only operate as a spindle, so it will have a constant valued
 1386 DataItem called RotaryMode. This machine is only capable of executing a single program
 1387 and therefore only capable of a single path. The following XML describes a simple
 1388 configuration for this machine.

```

1389 <?xml version="1.0" encoding="UTF-8"?>
1390 <MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1"
1391   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
1392   xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1
1393   MTConnectDevices.xsd">
1394   <Header bufferSize="130000" instanceId="1" creationTime="2009-11-
1395     13T02:31:40" sender="local" version="1.1"/>
1396   <Devices>
1397     <Device id="d1" uuid="HM1" name="HMC_3Axis">
1398       <Description>3 Axis Mill</Description>
1399       <Components>
1400         <Axes id="a" name="base">
1401           <Components>
  
```

```

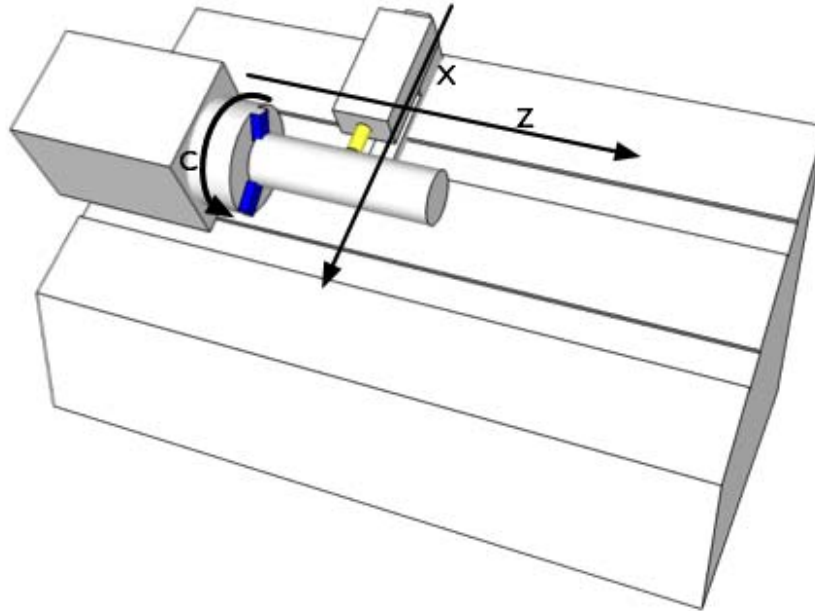
1402     <Linear id="y" name="Y">
1403         <DataItems>
1404             <DataItem type="POSITION" subType="ACTUAL" id="yp"
1405                 category="SAMPLE" name="Yact" units="MILLIMETER"
1406                 nativeUnits="MILLIMETER" coordinateSystem="MACHINE" />
1407         </DataItems>
1408     </Linear>
1409     <Linear id="x" name="X">
1410         <DataItems>
1411             <DataItem type="POSITION" subType="ACTUAL" id="xp"
1412                 category="SAMPLE" name="Xact" units="MILLIMETER"
1413                 nativeUnits="MILLIMETER" coordinateSystem="MACHINE" />
1414         </DataItems>
1415     </Linear>
1416     <Linear id="z" name="Z">
1417         <DataItems>
1418             <DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact"
1419                 subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"
1420                 coordinateSystem="MACHINE" />
1421         </DataItems>
1422     </Linear>
1423     <Rotary id="c" name="C">
1424         <DataItems>
1425             <DataItem type="ROTARY_VELOCITY" id="cspd" category="SAMPLE"
1426                 name="Sspeed" subType="ACTUAL" units="REVOLUTION/MINUTE"
1427                 nativeUnits="REVOLUTION/MINUTE" />
1428             <DataItem type=" ROTARY_VELOCITY " id="cso" category="SAMPLE"
1429                 name="Sovr" subType="OVERRIDE" units="PERCENT"
1430                 nativeUnits="PERCENT" />
1431             <DataItem type="ROTARY_MODE" id="rf" category="EVENT"
1432                 name="rfunc" />
1433             <Constraints>
1434                 <Value>SPINDLE</Value>
1435             </Constraints>
1436         </DataItem>
1437     </DataItems>
1438 </Rotary>
1439 </Components>
1440 </Axes>
1441 <Controller id="cont" name="controller">
1442     <Components>
1443         <Path id="path" name="path">
1444             <DataItems>
1445                 <DataItem type="PROGRAM" id="pgm" category="EVENT"
1446                     name="program" />
1447                 <DataItem type="BLOCK" id="blk" category="EVENT" name="block" />
1448                 <DataItem type="LINE" id="ln" category="EVENT" name="line" />
1449                 <DataItem type="PATH_FEEDRATE" id="pf" category="SAMPLE"
1450                     name="Fact" units="MILLIMETER/SECOND"
1451                     nativeUnits="FOOT/MINUTE" subType="ACTUAL" />
1452                 <DataItem type="PATH_FEEDRATE" id="pfo" category="SAMPLE"
1453                     name="Fovr" units="PERCENT" nativeUnits="PERCENT"
1454                     subType="OVERRIDE" />
1455                 <DataItem type="PATH_POSITION" id="pp" category="SAMPLE"
1456                     name="Ppos" units="MILLIMETER_3D" nativeUnits="FOOT_3D"
1457                     coordinateSystem="WORK" />
1458                 <DataItem type="EXECUTION" id="exec" category="EVENT"

```

```
1459         name="execution"/>
1460         <DataItem type="CONTROLLER_MODE" id="cm" category="EVENT"
1461         name="mode"/>
1462     </DataItems>
1463 </Path>
1464 </Components>
1465 </Controller>
1466 </Components>
1467 </Device>
1468 </Devices>
1469 </MTConnectDevices>
1470
```


1471 B.2. Two Axis Lathe

1472 The next machine is a simple two axis horizontal lathe with a Z and an X axis where the Linear Z
 1473 axis is aligned with the primary spindle Rotary C. The material is now held in the C axis and
 1474 the tool is fixed.



1475
 1476 **Figure 15: Two Axis Lathe**
 1477

```

1478 <?xml version="1.0" encoding="UTF-8"?>
1479 <MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1"
1480     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
1481     xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1
1482     MTConnectDevices.xsd">
1483 <Header bufferSize="130000" instanceId="1"
1484     creationTime="2009-11-13T02:31:40" sender="local" version="1.1"/>
1485 <Devices>
1486 <Device id="d1" uuid="HM1" name="HMC_3Axis">
1487 <Description>3 Axis Mill</Description>
1488 <Components>
1489 <Axes id="a" name="base">
1490 <Components>
1491 <Linear id="x" name="X">
1492 <DataItems>
1493 <DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"
1494     name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER"
1495     coordinateSystem="MACHINE"/>
1496 </DataItems>
1497 </Linear>
1498 <Linear id="z" name="Z">
1499 <DataItems>
1500
```

```

1501         <DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact"
1502             subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"
1503             coordinateSystem="MACHINE" />
1504     </DataItems>
1505 </Linear>
1506 <Rotary id="c" name="C">
1507     <DataItems>
1508         <DataItem type="ROTARY_VELOCITY" id="cspd" category="SAMPLE"
1509             name="Sspeed" subType="ACTUAL" units="REVOLUTION/MINUTE"
1510             nativeUnits="REVOLUTION/MINUTE" />
1511         <DataItem type="ROTARY_VELOCITY" id="cso" category="SAMPLE"
1512             name="Sovr" subType="OVERRIDE" units="PERCENT"
1513             nativeUnits="PERCENT" />
1514         <DataItem type="ROTARY_MODE" id="rf" category="EVENT" name="rfunc">
1515             <Constraints>
1516                 <Value>SPINDLE</Value>
1517                 <Value>INDEX</Value>
1518             </Constraints>
1519         </DataItem>
1520     </DataItems>
1521 </Rotary>
1522 </Components>
1523 </Axes>
1524 <Controller id="cont" name="controller">
1525     <Components>
1526         <Path id="path" name="path">
1527             <DataItems>
1528                 <DataItem type="PROGRAM" id="pgm" category="EVENT" name="program" />
1529                 <DataItem type="BLOCK" id="blk" category="EVENT" name="block" />
1530                 <DataItem type="LINE" id="ln" category="EVENT" name="line" />
1531                 <DataItem type="PATH_FEEDRATE" id="pf" category="SAMPLE" name="Fact"
1532                     units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE"
1533                     subType="ACTUAL" />
1534                 <DataItem type="PATH_FEEDRATE" id="pfo" category="SAMPLE"
1535                     name="Fovr" units="PERCENT" nativeUnits="PERCENT"
1536                     subType="OVERRIDE" />
1537                 <DataItem type="PATH_POSITION" id="pp" category="SAMPLE" name="Ppos"
1538                     units="MILLIMETER_3D" nativeUnits="FOOT_3D"
1539                     coordinateSystem="WORK" />
1540                 <DataItem type="EXECUTION" id="exec" category="EVENT"
1541                     name="execution" />
1542                 <DataItem type="CONTROLLER_MODE" id="cm" category="EVENT"
1543                     name="mode" />
1544             </DataItems>
1545         </Path>
1546     </Components>
1547 </Controller>
1548 </Components>
1549 </Device>
1550 </Devices>
1551 </MTConnectDevices>

```

1552 B.3. HyperQuadrex

Mazak - HyperQuadrex

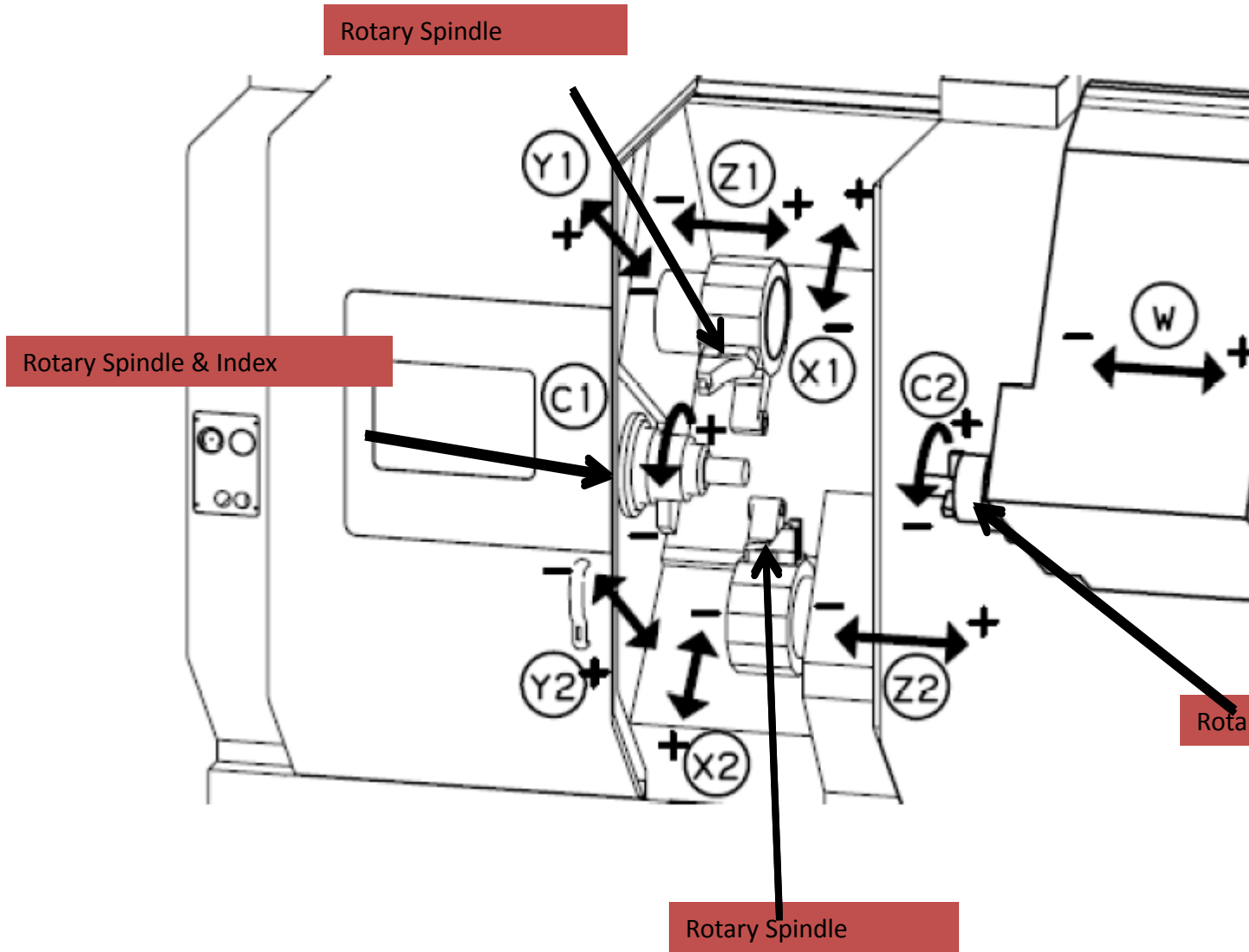


Figure 16: HyperQuadrex Lathe

1553
1554
1555

```

1556 <?xml version="1.0" encoding="UTF-8"?>
1557 <MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1"
1558     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
1559     xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:
1560     1.1 ../MTConnectDevices.xsd">
1561   <Header bufferSize="130000" instanceId="1" creationTime="
1562     2009-11-13T02:31:40" sender="local" version="1.1"/>
1563   <Devices>

```

```

1564 <Device id="d1" uuid="HM1" name="HyperQuadrex">
1565   <Description>Mazak - HyperQuadrex</Description>
1566   <Components>
1567     <Axes id="a" name="base">
1568       <Components>
1569         <Linear id="x" name="X" nativeName="X1">
1570           <DataItems>
1571             <DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"
1572               name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER"
1573               coordinateSystem="MACHINE">
1574               <Source>Xlpos</Source>
1575             </DataItem>
1576             <DataItem type="LOAD" id="x1" category="SAMPLE" name="Xload"
1577               units="PERCENT">
1578               <Source>Xlload</Source>
1579             </DataItem>
1580           </DataItems>
1581         </Linear>
1582         <Linear id="y" name="Y" nativeName="Y1">
1583           <DataItems>
1584             <DataItem type="POSITION" subType="ACTUAL" id="yp" category="SAMPLE"
1585               name="Yact" units="MILLIMETER" nativeUnits="MILLIMETER"
1586               coordinateSystem="MACHINE">
1587               <Source>Ylpos</Source>
1588             </DataItem>
1589             <DataItem type="LOAD" id="y1" category="SAMPLE" name="Yload"
1590               units="PERCENT">
1591               <Source>Ylload</Source>
1592             </DataItem>
1593           </DataItems>
1594         </Linear>
1595         <Linear id="z" name="Z" nativeName="Z1">
1596           <DataItems>
1597             <DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact"
1598               subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"
1599               coordinateSystem="MACHINE">
1600               <Source>Zlpos</Source>
1601             </DataItem>
1602             <DataItem type="LOAD" id="z1" category="SAMPLE" name="Zload"
1603               units="PERCENT">
1604               <Source>Zlload</Source>
1605             </DataItem>
1606           </DataItems>
1607         </Linear>
1608         <Linear id="x2" name="X2" >
1609           <DataItems>
1610             <DataItem type="POSITION" subType="ACTUAL" id="x2p"
1611               category="SAMPLE" name="X2act" units="MILLIMETER"
1612               nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
1613             <DataItem type="LOAD" id="x21" category="SAMPLE" name="X2load"
1614               units="PERCENT">
1615               <Source>X2load</Source>
1616             </DataItem>
1617           </DataItems>
1618         </Linear>
1619         <Linear id="y2" name="Y2">
1620           <DataItems>

```

```
1621     <DataItem type="POSITION" subType="ACTUAL" id="y2p"  
1622         category="SAMPLE" name="Y2act" units="MILLIMETER"  
1623         nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>  
1624     <DataItem type="LOAD" id="y2l" category="SAMPLE" name="Y2load"  
1625         units="PERCENT"/>  
1626     </DataItems>  
1627 </Linear>  
1628 <Linear id="z2" name="Z2">  
1629     <DataItems>  
1630  
1631
```

```

1632     <DataItem type="POSITION" id="z2p" category="SAMPLE" name="Z2act"
1633         subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"
1634         coordinateSystem="MACHINE">
1635         <Source>Z2pos</Source>
1636     </DataItem>
1637     <DataItem type="LOAD" id="z2l" category="SAMPLE" name="Z2load"
1638         units="PERCENT"/>
1639 </DataItems>
1640 </Linear>
1641 <Linear id="z3" name="Z3" nativeName="W">
1642     <DataItems>
1643         <DataItem type="POSITION" id="z3p" category="SAMPLE" name="Z3act"
1644             subType="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER"
1645             coordinateSystem="MACHINE">
1646             <Source>Wpos</Source>
1647         </DataItem>
1648         <DataItem type="LOAD" id="z3l" category="SAMPLE" name="Z3load"
1649             units="PERCENT">
1650             <Source>Wload</Source>
1651         </DataItem>
1652     </DataItems>
1653 </Linear>
1654 <Rotary id="c" name="C " nativeName="C1">
1655     <DataItems>
1656         <DataItem type="LOAD" id="C1" category="SAMPLE" name="Cload"
1657             units="PERCENT"/>
1658         <DataItem type=" ROTARY_VELOCITY " id="cspd" category="SAMPLE"
1659             name="Sspeed" subType="ACTUAL" units="REVOLUTION/MINUTE"
1660             nativeUnits="REVOLUTION/MINUTE"/>
1661         <DataItem type=" ROTARY_VELOCITY " id="cso" category="SAMPLE"
1662             name="Sovr" subType="OVERRIDE" units="PERCENT"
1663             nativeUnits="PERCENT"/>
1664         <DataItem type="DIRECTION" id="cdir" category="EVENT" name="Sdir"/>
1665         <DataItem type="ANGLE" id="cpos" category="SAMPLE" name="Cpos"
1666             subType="ACTUAL" units="DEGREE" nativeUnits="DEGREE"
1667             nativeScale="-1.0"/>
1668         <DataItem type="ROTARY_MODE" id="rf" category="EVENT" name="rfunc">
1669             <Constraints>
1670                 <Value>SPINDLE</Value>
1671                 <Value>INDEX</Value>
1672             </Constraints>
1673         </DataItem>
1674     </DataItems>
1675 </Rotary>
1676 <Rotary id="c2" name="C2">
1677     <DataItems>
1678         <DataItem type="LOAD" id="C2l" category="SAMPLE" name="C2load"
1679             units="PERCENT"/>
1680         <DataItem type=" ROTARY_VELOCITY " id="c2spd" category="SAMPLE"
1681             name="Sspeed" subType="ACTUAL" units="REVOLUTION/MINUTE"
1682             nativeUnits="REVOLUTION/MINUTE"/>
1683         <DataItem type=" ROTARY_VELOCITY " id="c2so" category="SAMPLE"
1684             name="Sovr" subType="OVERRIDE" units="PERCENT"
1685             nativeUnits="PERCENT"/>
1686         <DataItem type="DIRECTION" id="c2dir" category="EVENT"
1687             name="S2dir"/>
1688

```

```

1689     <DataItem type="ROTARY_MODE" id="rf2" category="EVENT" name="rfunc">
1690         <Constraints>
1691             <Value>SPINDLE</Value>
1692         </Constraints>
1693     </DataItem>
1694 </DataItems>
1695 </Rotary>
1696 <Rotary id="b" name="B" nativeName="S1">
1697     <DataItems>
1698         <DataItem type="LOAD" id="bl" category="SAMPLE" name="Blood"
1699             units="PERCENT"/>
1700         <DataItem type=" ROTARY_VELOCITY " id="bspd" category="SAMPLE"
1701             name="Sspeed" subType="ACTUAL" units="REVOLUTION/MINUTE"
1702             nativeUnits="REVOLUTION/MINUTE"/>
1703         <DataItem type=" ROTARY_VELOCITY " id="bso" category="SAMPLE"
1704             name="Sovr" subType="OVERRIDE" units="PERCENT"
1705             nativeUnits="PERCENT"/>
1706         <DataItem type="DIRECTION" id="bdir" category="EVENT" name="S3dir"/>
1707         <DataItem type="ROTARY_MODE" id="brf" category="EVENT" name="rfunc">
1708             <Constraints>
1709                 <Value>SPINDLE</Value>
1710             </Constraints>
1711         </DataItem>
1712     </DataItems>
1713 </Rotary>
1714 <Rotary id="b2" name="B2" nativeName="S2">
1715     <DataItems>
1716         <DataItem type="LOAD" id="b2l" category="SAMPLE" name="B2load"
1717             units="PERCENT"/>
1718         <DataItem type=" ROTARY_VELOCITY " id="b2spd" category="SAMPLE"
1719             name="Sspeed" subType="ACTUAL" units="REVOLUTION/MINUTE"
1720             nativeUnits="REVOLUTION/MINUTE"/>
1721         <DataItem type=" ROTARY_VELOCITY " id="b2so" category="SAMPLE"
1722             name="Sovr" subType="OVERRIDE" units="PERCENT"
1723             nativeUnits="PERCENT"/>
1724         <DataItem type="DIRECTION" id="b2dir" category="EVENT"
1725             name="S3dir"/>
1726         <DataItem type="ROTARY_MODE" id="b2rf" category="EVENT"
1727             name="rfunc">
1728             <Constraints>
1729                 <Value>SPINDLE</Value>
1730             </Constraints>
1731         </DataItem>
1732     </DataItems>
1733 </Rotary>
1734 </Components>
1735 </Axes>
1736 <Controller id="cont" name="controller">
1737     <Components>
1738         <Path id="path1" name="path1">
1739             <DataItems>
1740                 <DataItem type="ACTIVE_AXES" category="EVENT" name="axes"
1741                     id="act_axes1"/>
1742                 <DataItem type="PROGRAM" id="pgm1" category="EVENT" name="program"/>
1743                 <DataItem type="BLOCK" id="blk1" category="EVENT" name="block"/>
1744                 <DataItem type="LINE" id="ln1" category="EVENT" name="line"/>
1745

```

```

1746     <DataItem type="PATH_FEEDRATE" id="pf1" category="SAMPLE"
1747         name="Fact" units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE"
1748         subType="ACTUAL" coordinateSystem="WORK" />
1749     <DataItem type="PATH_FEEDRATE" id="pfo1" category="SAMPLE"
1750         name="Fovr" units="PERCENT" nativeUnits="PERCENT"
1751         subType="OVERRIDE" />
1752     <DataItem type="PATH_POSITION" id="pp1" category="SAMPLE"
1753         name="Ppos" units="MILLIMETER_3D" nativeUnits="MILLIMETER_3D"
1754         coordinateSystem="WORK" />
1755     <DataItem type="TOOL_ASSET_ID" id="tid1" category="EVENT"
1756         name="Tid" />
1757     <DataItem type="PART_ID" id="pid1" category="EVENT" name="Pid" />
1758     <DataItem type="EXECUTION" id="exec1" category="EVENT"
1759         name="execution" />
1760     <DataItem type="CONTROLLER_MODE" id="cm1" category="EVENT"
1761         name="mode" />
1762     </DataItems>
1763 </Path>
1764 <Path id="path2" name="path2">
1765     <DataItems>
1766         <DataItem type="ACTIVE_AXES" category="EVENT" name="axes"
1767             id="act_axes2" />
1768         <DataItem type="PROGRAM" id="pgm2" category="EVENT" name="program" />
1769         <DataItem type="BLOCK" id="blk2" category="EVENT" name="block" />
1770         <DataItem type="LINE" id="ln2" category="EVENT" name="line" />
1771         <DataItem type="PATH_FEEDRATE" id="pf2" category="SAMPLE"
1772             name="Fact" units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE"
1773             subType="ACTUAL" coordinateSystem="WORK" />
1774         <DataItem type="PATH_FEEDRATE" id="pfo2" category="SAMPLE"
1775             name="Fovr" units="PERCENT" nativeUnits="PERCENT"
1776             subType="OVERRIDE" />
1777         <DataItem type="PATH_POSITION" id="pp2" category="SAMPLE"
1778             name="Ppos" units="MILLIMETER_3D" nativeUnits="MILLIMETER_3D"
1779             coordinateSystem="WORK" />
1780         <DataItem type="TOOL_ASSET_ID" id="tid2" category="EVENT"
1781             name="Tid" />
1782         <DataItem type="PART_ID" id="pid2" category="EVENT" name="Pid" />
1783         <DataItem type="EXECUTION" id="exec2" category="EVENT"
1784             name="execution" />
1785         <DataItem type="CONTROLLER_MODE" id="cm2" category="EVENT"
1786             name="mode" />
1787     </DataItems>
1788 </Path>
1789 </Components>
1790 </Controller>
1791 <Door id="d" name="door">
1792     <DataItems>
1793         <DataItem id="ds" category="EVENT" name="door" type="DOOR_STATE" />
1794     </DataItems>
1795 </Door>
1796 </Components>
1797 </Device>
1798 </Devices>
1799 </MTConnectDevices>
1800

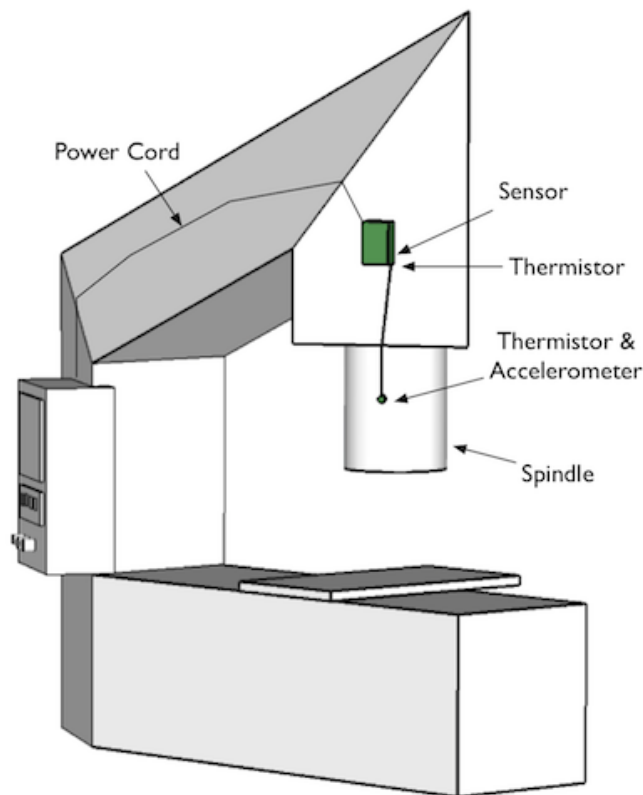
```


1801 B.4. Sensors

1802

1803 Sensors are modeled with the `DataItem` types associated directly with the `Component` that
 1804 is being measured. In the example below, the spindle has measurement for temperature
 1805 (thermistor) and vibration (accelerometer). Additionally, the sensor unit may have its own
 1806 diagnostic measurements – in this case, a temperature measurement (thermistor) to measure the
 1807 health of the sensor unit.

1808



1809

1810

Figure 17: Spindle Sensing System

1811

1812 The basic machine is modeled below – 3 linear axes and a spindle. The spindle has two
 1813 additional `DataItems` representing the sensors for temperature and acceleration.

```

1814 <?xml version="1.0" encoding="UTF-8"?>
1815 <MTConnectDevices xmlns="urn:mtconnect.org:MTConnectDevices:1.2"
1816   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
1817   xsi:schemaLocation="urn:mtconnect.org:MTConnectDevices:
1818     1.2 ../MTConnectDevices_1.2.xsd">
1819   <Header bufferSize="130000" instanceId="1" creationTime="
1820     2009-11-13T02:31:40" sender="local" version="1.2"/>

```

```

1821 <Devices>
1822   <Device id="d1" uuid="HM1" name="HMC_3Axis">
1823     <Description>3 Axis Mill</Description>
1824     <DataItems>
1825     <DataItem type="AVAILABILITY" category="EVENT" id="avail" />
1826     </DataItems>
1827     <Components>
1828       <Axes id="a" name="base">
1829         <Components>
1830           <Linear id="y" name="Y">
1831             <DataItems>
1832               <DataItem type="POSITION" subType="ACTUAL" id="yp"
1833                 category="SAMPLE" name="Yact" units="MILLIMETER"
1834                 nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
1835             </DataItems>
1836           </Linear>
1837           <Linear id="x" name="X">
1838             <DataItems>
1839               <DataItem type="POSITION" subType="ACTUAL" id="xp"
1840                 category="SAMPLE" name="Xact" units="MILLIMETER"
1841                 nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
1842             </DataItems>
1843           </Linear>
1844           <Linear id="z" name="Z">
1845             <DataItems>
1846               <DataItem type="POSITION" id="zp" category="SAMPLE"
1847                 name="Zact" subType="ACTUAL" units="MILLIMETER"
1848                 nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
1849             </DataItems>
1850           </Linear>
1851           <Rotary id="c" name="C">
1852             <DataItems>
1853               <DataItem type="ROTARY_VELOCITY" id="cspd" category="SAMPLE"
1854                 name="Sspeed" subType="ACTUAL" units="REVOLUTION/MINUTE"
1855                 nativeUnits="REVOLUTION/MINUTE"/>
1856               <DataItem type="ROTARY_VELOCITY" id="cso" category="SAMPLE"
1857                 name="Sovr" subType="OVERRIDE" units="PERCENT"
1858                 nativeUnits="PERCENT"/>
1859               <DataItem type="ROTARY_MODE" id="rf" category="EVENT"
1860                 name="rfunc">
1861                 <Constraints>
1862                   <Value>SPINDLE</Value>
1863                 </Constraints>
1864               </DataItem>
1865               <DataItem type="TEMPERATURE" category="SAMPLE" name="Ctemp"
1866                 id="ct" units="CELSIUS" statistic="AVERAGE">
1867                 <Source componentId="s1">channel:1</Source>
1868               <DataItem type="ACCLERATION" category="SAMPLE" name="Sacc"
1869                 id="sa" units="MILLIMETERS/SECOND^2" statistic="MAXIMUM">
1870                 <Source componentId="s2">channel:2</Source>
1871               </DataItem>
1872             </DataItems>
1873           </Rotary>
1874         </Components>
1875       </Axes>
1876     </Device>

```

1877 Additionally, the sensor unit is modeled with its configuration information and a DataItem of
 1878 category Sample (Voltage) and a DataItem of type Condition (Voltage).

```

1879 <Components>
1880   <Sensor id="sensor" name="sensor">
1881     <Configuration>
1882       <SensorConfiguration>
1883         <FirmwareVersion>2.02</FirmwareVersion>
1884         <CalibrationDate>2010-05-16</CalibrationDate>
1885         <NextCalibrationDate>2010-05-16</NextCalibrationDate>
1886         <CalibrationInitials>WS</CalibrationInitials>
1887         <Channels>
1888           <Channel number="1" name="A/D:1">
1889             <Description>A/D With Thermister</Description>
1890           <Channel number="2" name="A/D:2">
1891             <Description>A/D With Accelerometer</Description>
1892           </Channel>
1893         </Channels>
1894       </SensorConfiguration>
1895     </Configuration>
1896     <DataItems>
1897       <DataItem category="CONDITION" id="senvc" type="VOLTAGE" />
1898       <DataItem category="SAMPLE" id="senv" type="VOLTAGE" units="VOLT"
1899         subType="DIRECT" />
1900     </DataItems>
1901   </Sensor>
1902 </Components>
1903 </Device>
1904 </Devices>
1905 </MTConnectDevices>
1906

```