MTConnect® Standard
Part 2 – Components and Data Items
Version 1.1.0 – Final

Prepared for: MTConnect Institute
Prepared by: William Sobel
Prepared on: April 30, 2010

MTConnect® Specification

Table of Contents

Table of Figures

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 industry, 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® will focus on a limited set of the characteristics mentioned above that were selected based on the fact that they can have an immediate affect on the efficiency of operations.

1.1 MTConnect® Document Structure

The MTConnect® specification is subdivided using the following scheme:

Part 1: Overview and Protocol – Version 1.1.0, Final
Part 2: Components and Data Items – Version 1.1.0, Final
Part 3: Streams, Events, Samples, and Condition – Version 1.1.0, Final

Extensions to the standard will be made according to this scheme and new sections will be added as new areas are addressed. Documents will be named as follows: MTC_Part_<Number>_<Description>.doc. All documents will be developed in Microsoft® Word format and released in Adobe® PDF format. For example, this document is MTC_Part_1_Overview.doc.

2 Purpose of This Document

The three MTConnect® documents are intended to:

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

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

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

2.1 Terminology

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

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

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

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

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

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

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

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

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

Data Item	A data item provides the descriptive information regarding something that can be collected by the *Agent*.
Device	A piece of equipment capable of performing an operation. A device is composed of a set of components that provide data to the application. The device is a separate entity with at least one Controller managing its operation.
Discovery	Discovery is a service that allows the application to locate *Agents* for devices in the manufacturing environment. The discovery service is also referred to as the *Name Service*.
Element	An XML element is the central building block of any XML Document. For example, in MTConnect® the Device element is specified as <**`Device`**>`...`</**`Device`**>
Event	An event represents a change in state that occurs at a point in time. Note: An event does not occur at predefined frequencies.
HTTP	Hyper-Text Transport Protocol. The protocol used by all web browsers and web applications.
Instance	When used in software engineering, the word *instance* is used to define a single physical example of that type. In object-oriented models, there is the class that describes the thing and the instance that is an example of that thing.
LDAP	Lightweight Directory Access Protocol, better known as Active Directory in Microsoft Windows. This protocol provides resource location and contact information in a hierarchal structure.
MIME	Multipurpose Internet Mail Extensions. A format used for encoding multipart mail and http content with separate sections separated by a fixed boundary.
Probe	A request to determine the configuration and reporting capabilities of the device.
REST	REpresentational State Transfer. A software architecture where the client and server move through a series of state transitions based solely on the request from the client and the response from the server.
Results	A general term for the `Samples`, `Events`, and `Condition` contained in a `ComponentStream` as a response from a `sample` or `current` request.
Sample	A sample is a data point from within a continuous series of data points. An example of a Sample is the position of an axis.
Socket	When used concerning interprocess communication, it refers to a connection between two end-points (usually processes). Socket communication most often uses TCP/IP as the underlying protocol.
Stream	A collection of `Events` and `Samples` organized by devices and components.

Service	An application that provides necessary functionality.
Tag	Used to reference an instance of an XML element.
TCP/IP	TCP/IP is the most prevalent stream-based protocol for interprocess communication. It is based on the IP stack (Internet Protocol) and provides the flow-control and reliable transmission layer on top of the IP routing infrastructure.
URI	Universal Resource Identifier. This is the official name for a web address as seen in the address bar of a browser.
UUID	Universally unique identifier.
XPath	XPath is a language for addressing parts of an XML Document. See the XPath specification for more information. http://www.w3.org/TR/xpath
XML	Extensible Markup Language. http://www.w3.org/XML/
XML Schema	The definition of the XML structure and vocabularies used in the XML Document.
XML Document	An instance of an XML Schema which has a single root element and conforms to the XML specification and schema.
XML `NMTOKEN`	The data type for XML identifiers. It 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.

2.2 Terminology and Conventions

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

3 Devices and Components

A device can be thought of as a group of components. For example, Figure 1 illustrates a mill `Device`. The mill has the following components: every device in MTConnect® **MUST** have an `Availability` data item; availability represents the devices ability to provide information about itself. The mill also has sub-components of the `Axes` component; there are the three `Linear` axes and one `Rotary axis` representing the spindle. The `Controller` component controls the axes and runs the program using a single `Path` component.

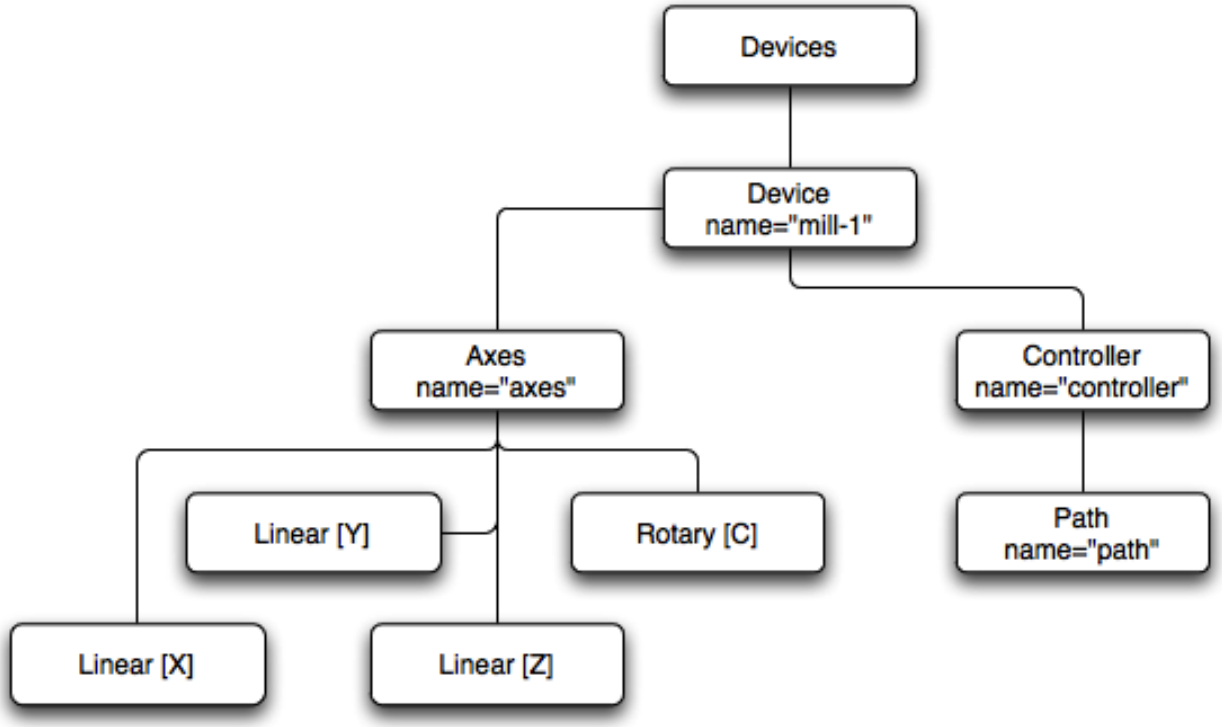


Figure 1: Example Devices Structure

Multiple devices may be represented in a top level container element called `Devices`. These container elements have no additional attributes and are only used to group sub-elements together. There are three containers used in the `MTConnectDevices` document. The first is the `Devices` container holding all `Device` elements. The next container is `Components` that groups all the subcomponents together, like the `Linear` and `Rotary` axes. The last container is `DataItems` that groups all data items for a component together.

In the following document structure:

```
   MTConnectDevices
      Devices
         Device
            Components
               Axes
                  Components
                     Rotary [C]
                   Values SPINDLE
                     Linear  [X]
```

```
                        DataItems
                         DataItem [Xpos]
                   Linear  [Y]
                      DataItems
                       DataItem [Ypos]
                   Linear  [Z]
                      DataItems
                       DataItem [Zpos]

            Controller
            Components
              Path
                   DataItems
                      DataItem [mode]
                      DataItem [execution]
```

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

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

3.1 Devices

The `Devices` element is a top level container for every `Device` returned from a `probe` request. `Devices` is a similar container to `Components` except it may only contain elements of type `Device`.

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

3.1.1 `Device`

A `Device` is a component that holds all the components associated with this piece of equipment. The `Device` **MUST** have an `Availablity` data item that indicates if this device is available to provide information.

3.1.1.1 `Device` Attributes

DEPRECATION WARNING: The ISO 841 classification is being deprecated in the next release to be replaced with a more current ontology of machine types.

Attribute	Description	Occurrence

Attribute	Description	Occurrence
iso841Class	**DEPRECATION WARNING**: The ISO 841 classification for the device.	0..1
uuid	A unique identifier that will only refer to this component. For example, this can be the manufacturer code and the serial number. The uuid should be alphanumeric and not exceeding 255 characters. An NMTOKEN XML type.	0..1*
name	The name of the component. This name should be unique within the machine to allow for easier data integration. An NMTOKEN XML type.	1
nativeName	The name the device manufacturer assigned to the component. If the native name is not provided it **MUST** be the name.	0..1
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	The rate in seconds that data is obtained from the component. This is the number of milliseconds between data captures. If the sample rate is smaller than one millisecond, the number can be represented as a floating point number. For example, for one 100 microsecond sample rate would be 0.1.	0..1**

Notes: * The uuid **MUST** be provided for the Device, it is optional for all other components.
** The sampleRate is used to aid the application in interpolating values. This is the desired sample rate and may vary depending on the capabilities of the device.

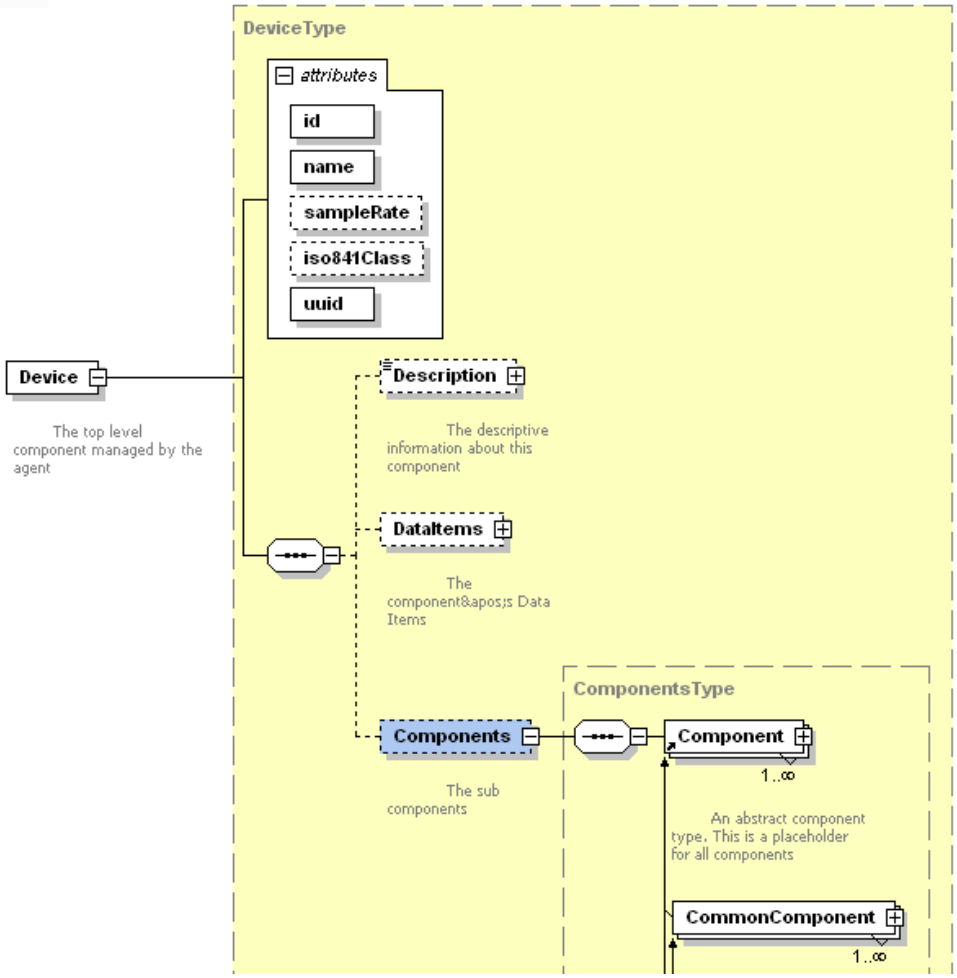
A device **MAY** be classified using one of the following identifiers from the ISO 841 specification (this will be deprecated in the next version of the specification). The following classification is taken from the appendix of the ISO 841 specification.

MTC ISO 841 Classification	Description	Figure
1	Other (Device not included in list)	
2	Parallel lathe (engine lathe)	A.2
3	Twin turret lathe with programmable tailstock	A.3
4	Vertical turning and boring lathe	A.4
5	Milling machine with horizontal spindle	A.5
6	Milling machine with vertical spindle (with W axis)	A.6
7	Boring and milling machine with horizontal spindle	A.7
8	Milling machine with vertical spindle	A.8
9	Portal-type milling machine	A.9

MTC ISO 841 Classification	Description	Figure
10	Gantry-type milling machine	A.10
11	Planer-type horizontal boring machine	A.11
12	Profile and contouring milling machine with movable table	A.12
13	Profile and contour milling machine with horizontal spindle	A.13
14	Profile and contour milling machine with tilting head	A.14
15	Profile and contour milling machine with tilting table	A.15
16	External cylindrical grinding machine	A.16
17	Tool and cutter grinding machine	A.17
18	Openside planer	A.18
19	Vertical filament winding machine	A.19
20	Horizontal filament winding machine	A.20
21	Flame cutting machine	A.21
22	Punch press	A.22
23	Drafting machine	A.23
24	Right-hand tube bender	A.24
25	Surface grinding machine with vertical grinding wheel	A.25
26	Cavity sinking EDM machine	A.26
27	Surface grinding machine	A.27
28	Coordinate measuring machine	A.28
29	Press brake	A.29
30	Wire electrical discharge machine	A.30
31	Laser cutting machine	A.31
32…	Reserved for future use.	

205

206 ### 3.1.1.2 `Device` Structure



207

208 **Figure 2: Device Schema Diagram**

209 ## 3.2 `Component`

210 The *Agent* needs to be capable of delivering data associated with each component to an
211 application. The description of these pieces of information is referred to as `DataItems` and will
212 be discussed in the section 4 of this document. The actual values for those data items are
213 delivered in `Streams` and will be discussed in Part 3 of the standard on *Streams, Samples, and*
214 *Events*.

3.3 Component Schema

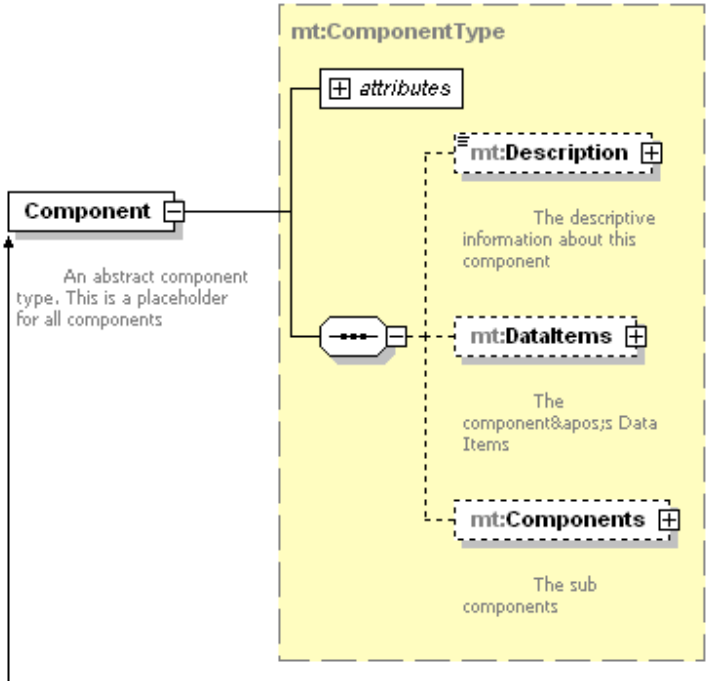


Figure 3: Component Schema

3.3.1 Common Component Attributes

Every component has the following composition:

Attribute	Description	Occurrence
uuid	A unique identifier that will only refer to this component. For example, this can be the manufacturer code and the serial number. The uuid should be alphanumeric and not exceeding 255 characters. An NMTOKEN XML type.	0..1*
name	The name of the component. This name should be unique within the machine to allow for easier data integration. An NMTOKEN XML type.	1
nativeName	The name the device manufacturer assigned to the component. If the native name is not provided it **MUST** be the name.	0..1
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	The rate in seconds that data is obtained from the component. This is the number of milliseconds between data captures. If the sample rate is smaller than one millisecond, the number can be represented as a floating point number. For example, for one 100 microsecond sample rate would be 0.1.	0..1**

Notes: * The `uuid` **MUST** be provided for the `Device`, it is optional for all other components.
** The `sampleRate` is used to aid the application in interpolating values. This is the desired sample rate and may vary depending on the capabilities of the device.

3.3.2 Component Elements

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

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

3.3.2.1 `Description`

Attribute	Description	Occurrence
`manufacturer`	The name of the manufacturer of the component	0..1
`serialNumber`	The device's serial number	0..1
`station`	The station the device is located at. When a device is part of a manufacturing unit or cell with multiple stations that share the same physical controller.	0..1

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

```
<Description manufacturer="Example Co" serialNumber="A124FFF" station="2">
  Example Co Simulated Verticle 3 Axis Machining center.
</Description>
```

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

```
<Description manufacturer="Example Co"
   serialNumber="EXCO-TT-099PP-XXXX">
  Advanced Pulse watt-hour transducer with pulse output.
</Description>
```

3.3.2.2 `Components`

Element	Description	Occurrence
`Component`	One or more components. This can also include the subtypes of Component like Axes, Linear, Thermostat, etc...	1..INF

3.3.2.3 DataItems

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

3.4 Types of Components

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

3.4.1 Axes

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

The Axes represent the physical data for the axis components and positions **MUST** be given in MACHINE coordinates. The WORK coordinates will now be represented in the Path component of the Controller.

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

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

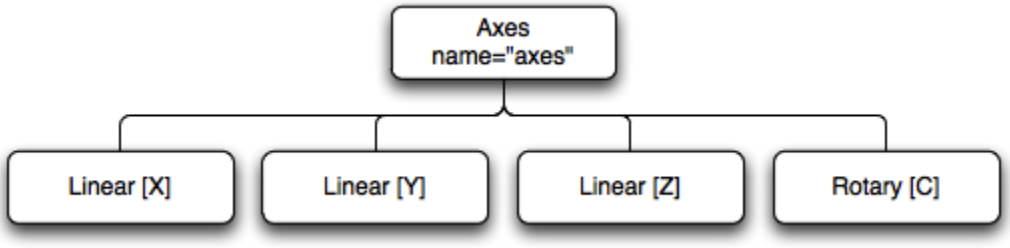


Figure 4: Axes Example With Three Linear Axes and one Rotary Axis

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

Rotary An axis whose function is to provide rotary motion either for the purpose of continuous rotation (i.e. spindle mode), for continuous-path contour cutting in a rotary direction or for repositioning (i.e. indexing) different faces of the part, for example, the purpose of metal removal. A rotary axis can operate in one of the three following modes: SPINDLE, INDEX, or CONTOUR.

3.4.2 Controller

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

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

3.4.2.1 Path

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

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

3.4.3 ~~Power~~ DEPRECATED

NOTE: Power as an indication of availability will be changed to the data item AVAILABILITY and electrical current and power consumption will be represented by the Electric system, see *3.4.7.5Electric* below.

~~The Power component is provided to report on the power status and possibly the voltage associated with its parent component. The device **MUST** contain a Power component and the Power component **MUST** contain the POWER_STATE data item. Any other data items **MAY** be added. Any other component, such as a Rotary, that can be switched on or off separately from the Device **SHOULD** have a Power component if this information is available.~~

~~Power **MUST** have a value of ON if the device is reachable and its power indicator is ON. A status of OFF means the power supply to the device has been disconnected. The one exception to this rule is if the Computer controller on the device is powered on but the rest of the device is powered off. In this case the device power status will still be considered OFF.~~

3.4.4 Door

This component represents a door closure that can be opened or closed. It **MUST** have a data item `DoorState` to indicate if it is opened or closed.

3.4.5 Actuator

An actuator is a mechanical device for moving or controlling a mechanism or system. It takes energy, usually transported by air, electric current, or liquid, and converts it into some kind of motion. (Wikipedia).

3.4.6 Sensors

Sensors are components that may or may not be integral to a parent component or device. They can be external to the device and can be moved from one device to another. They **MAY** have their own uuid so they can be tracked throughout their lifetime.

3.4.6.1 Pressure

A sensor or instrument used to measure the force exerted by a liquid or gas.

3.4.6.2 Thermostat

A sensor or instrument used to measure temperature.

3.4.6.3 Vibration

A sensor or instrument used to measure the amount and/or frequency of vibration within a system.

3.4.7 Systems

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

3.4.7.1 Hydraulic

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

3.4.7.2 Pneumatic

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

3.4.7.3 Coolant

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

3.4.7.4 Lubrication

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

3.4.7.5 Electric

The electric system represents the main power supply or generator for the device. The electric system will provide all the data with regard to current, voltage, and frequency.

4 Data Items

A `DataItem` describes a piece of information that can be collected from a component. The data item **MUST** specify the `type` of data being collected, the `id` of the data item, and the `category` of the item. There will only be one category for each `type`, but it **MUST** be included to aid the application in determining the location for the data stream. The data item **MAY** specify a `Source` sub-element to provide the native name for the data feed.

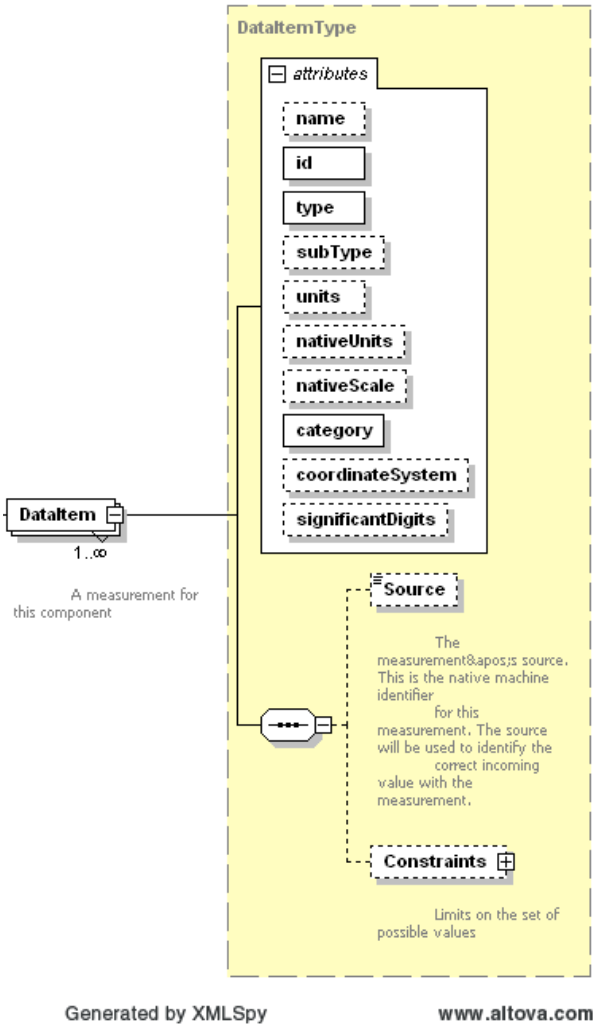


Figure 5: DataItem Schema Diagram

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

The `units` **MUST** be specified for any data item with category `Sample`. The `nativeUnits` **MAY** be specified if they apply to the type of data and if they differ from the `units`. The *Agent* is responsible for converting the `nativeUnits` to the `units` before sending them to the

applications. In addition, `nativeUnits` **MAY** be scaled using the `nativeScale` attribute; for example, if the device measures velocity in 100 ft/min, MTConnect® would represent it with the following attributes: `nativeUnits="FEET/MINUTE"` and `nativeScale="100"`.

4.1 `DataItem` Element

4.1.1 Data Item Attributes

Attribute	Description	Occurrence
`id`	The unique identifier for this data item. 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 data item. A name is provided as an additional human readable identifier for this data item in addition to the id. It is not required and will be implementation dependent. The identity of this data item is the type and sub-type. An NMTOKEN XML type.	0..1
`type`	The type of data being measured. Examples of types are `POSITION`, `VELOCITY`, `ANGLE`, `BLOCK`, `SPINDLE_SPEED`, 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 data item will be sampled. The available options are `SAMPLE, EVENT`, or `CONDITION`.	1
`nativeUnits`	The native units used by the component. These units will be converted before they are delivered to the application.	0..1
`units`	The units delivered to the application. These will always be the same for this data item type. This **MUST** be specified for all numeric values.	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
`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
`coordinateSystem`	The coordinate system being used.	0..1

4.1.2 Data Item Elements

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

4.1.2.1 Constraints Elements

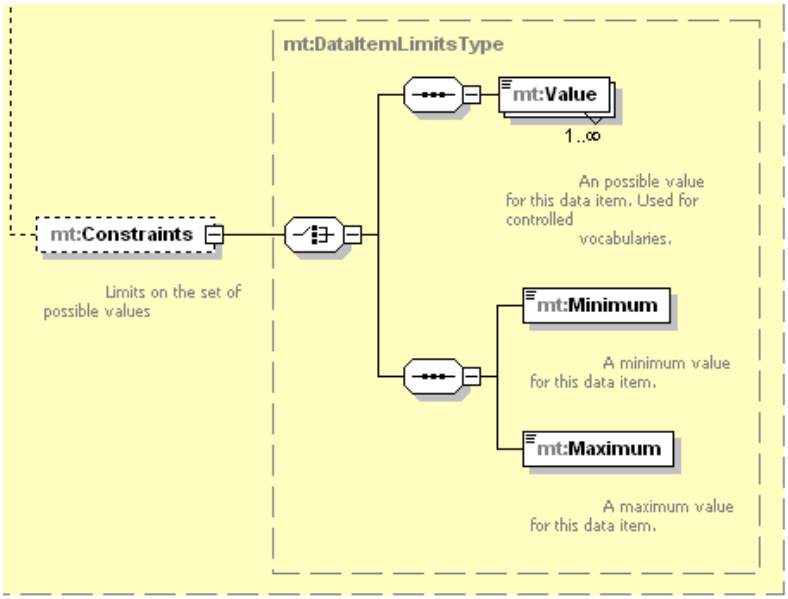


Figure 6: Constraints Schema

Element	Description	Occurrence
Value	A constraint on the possible values for this data item. If there is only one value listed here, the value data item will be constant. In the case of a constant data item, the value is not required to be supplied in the streams document.	0..INF

Element	Description	Occurrence
Maximum	The maximum value for this data item. This will be the bounded upper range. This will only be relevant when the data item has a numeric type.	0..1
Minimum	The minimum value for this data item. This will be the bounded lower range. This will only be relevant when the data item has a numeric type.	0..1

4.1.3 Data Item attribute: category

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

SAMPLE A Sample is the reading of the value of a continuously variable or analog data item. A continuous value can be sampled at any point-in-time and will always product a result. An example of a continuous data item is the Rotary C axis spindle speed.

Sample data items that are continuous are always scalar floating point or integers that can have an infinite number of possible values. This is different from state or discrete data items that have a limited number of possible values. Samples **MUST** have units.

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

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

4.1.4 Data Item attribute: coordinateSystem

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

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

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

4.1.5 Data Item attribute: units

Unit	Description
AMPERE	Amps
CELCIUS	Degrees Celsius
COUNT	A counted event
DEGREE	Angle in degrees
DEGREE/SECOND	Degrees per second
DEGREE/SECOND^2	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
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. The SI units will be used.
PASCAL	Pressure in Newtons per square meter
PERCENT	Percent
PH	pH is a measure of hydrogen ion concentration; a measure of the acidity or alkalinity of a solution.
REVOLUTION/MINUTE	Revolutions per minute
SECOND	A measurement of time.
VOLT	Volts

Unit	Description
WATT	Watts

4.1.6 Data Item attribute: nativeUnits

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

Unit	Description
DEGREE/MINUTE	Rotational velocity in degrees per minute
FAHRENHEIT	Temperature in Fahrenheit
FOOT	Feet
FOOT/MINUTE	Feet per minute
FOOT/SECOND	Feet per second
FOOT/SECOND^2	Acceleration in feet per second squared
FOOT_3D	A point in space identified by X, Y, and Z positions and represented by a space delimited set of numbers each expressed in feet.
GALLON/MINUTE	Gallons per minute.
INCH	Inches
INCH/MINUTE	Inches per minute
INCH/SECOND	Inches per second
INCH/SECOND^2	Acceleration in inches per second squared
INCH_3D	A point in space identified by X, Y, and Z positions and represented by a space delimited set of numbers each expressed in inches.
INCH_POUND	A measure of torque in inch pounds.
KILOWATT	A measurement in kilowatt.
KILOWATT_HOUR	Kilowatt hours which is 3.6 mega joules.
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

Unit	Description
RADIAN/MINUTE	Velocity in radians per second.
REVOLUTION/SECOND	Rotational velocity in revolution per second
OTHER	Unsupported units

4.2 Types and Subtypes of Data Items

What follows is the association between the various types and subtypes of data items. Each data item type **MUST** be translated into a `Sample` or `Event` with the following rules: The type name will be all in capitals with an underscore (_) between words. The element of the event or sample will be the transformation of the data item type by capitalizing the first character of each word and then removing the underscore. For example, the data item type `DOOR_STATE` is `DoorState`, `POSITION` is `Position`, and `SPINDLE_SPEED` is `SpindleSpeed`.

An example of this transformation between the `DataItem` name and the Stream element is as follows:

```
<Path name="path" id="p1">
   <DataItems>
      <DataItem type="LINE" category="EVENT" id="p2" subType="ACTUAL"
         name="line" />
      <DataItem type="CONTROLLER_MODE" category="EVENT" id="p3" name="mode"
         />
      <DataItem type="PROGRAM" category="EVENT" id="p4" name="program" />
      <DataItem type="EXECUTION" category="EVENT" id="p5" name="execution" />
      <DataItem type="BLOCK" category="EVENT" id="p6" name="block" />
   </DataItems>
</Path>
```

The transformation from the `probe` *(as defined in Part 1 of the standard)* to the `current` or `sample` will occur as follows. This also illustrates how the `subType` is also placed in the `ComponentStream` as well. The `probe` will provide the `category` meaning the sub-element of the `ComponentStream` the items will appear in. Also note how the `CONTROLLER_MODE` was changed to `ControllerMode` in the `current` request below.

```
<ComponentStream componentId="p1" component="Path" name="path">
   <Events>
      <Line dataItemId="p2" timestamp="2009-03-04T19:45:50.458305"
         subType="ACTUAL" name="line" sequence="150651130">702</Line>
      <Block dataItemId="p6" timestamp="2009-03-04T19:45:50.458305"
         name="block" sequence="150651134">x0.371524 y-0.483808</Block>

      <ControllerMode dataItemId="p3" timestamp="2009-02-26T02:02:35.716224"
         name="mode" sequence="182">AUTOMATIC</ControllerMode>
```

```
447       </Events>
448   </ComponentStream>
449
```

4.2.1 Data Item Types for SAMPLE Category

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

Data Item type/subtype	Description	Units
ACCELERATION	Rate of change of velocity	MILLIMETER/SECOND^2
ANGULAR_ACCELERATION	Rate of change of angular velocity.	DEGREE/SECOND^2
ANGULAR_VELOCITY	Rate of change of angular position.	DEGREE/SECOND
AMPERAGE	The line 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
DISPLACEMENT	The displacement as measured from zero to peak	MILLIMETER
FREQUENCY	The frequency as measure in cycles per second	HERTZ
~~**GLOBAL_POSITION (DEPRECATED)**~~	~~The position in three-dimensional space. The X, Y, and Z positions will be provided.~~	~~MILLIMETER~~
~~ACTUAL~~	~~The position of the component as read from the device.~~	~~MILLIMETER~~
~~COMMANDED~~	~~The position computed by the controller.~~	~~MILLIMETER~~
LOAD	The load on the component.	NEWTON
PATH_FEEDRATE	The feedrate of the tool path.	MILLIMETER/SECOND
ACTUAL	The three-dimensional feedrate derived from all components.	MILLIMETER/SECOND
COMMANDED	The feedrate as specified in the program	MILLIMETER/SECOND
OVERRIDE	The operator's overridden value. Percent of commanded.	PERCENT

Data Item type/subtype	Description	Units
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	Th measure of hydrogen ion concentration; a measure of the acidity or alkalinity of a solution.	PH
PRESSURE	The pressure on the component	PASCAL
POSITION	The position of the component. Defaults to MACHINE coordinates.	MILLIMETER
ACTUAL	The position of the component as read from the device.	MILLIMETER
COMMANDED	The position as given by the Controller.	MILLIMETER
TARGET	The target position for the movement.	MILLIMETER
SPINDLE_SPEED	The rotational speed of the 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 the as specified in the program.	REVOLUTION/MINUTE
OVERRIDE	The operator's overridden value. Percent of commanded.	PERCENT
TEMPERATURE	The temperature	CELSIUS
TORQUE	The torque	NEWTON_METER
VELOCITY	The rate of change of position.	MILLIMETER/SECOND
VOLTAGE	The voltage	VOLT
WATTAGE	The wattage	WATT

452

453

4.2.2 Data Item Types for EVENT Category

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

Data Item type/subtype	Description
~~ALARM~~	~~An alarm is a special data item that will report any alarm for this component. An alarm **MUST** be included as a DataItem for the Device~~ DEPRECATED: Replaced with CONDITION category.
ACTIVE_AXES	The set of axes associated with a path that the controller is controlling. If this data item is not provided, it will be assumed the controller is controlling all axes.
AVAILABILITY	Represents the components ability to communicate its availability. This **MUST** be provided for the device and **MAY** be provided for all other components
AXIS_COUPLING	Describes the way the axes will be associated to each other. This is used in conjunction with COUPLED_AXES to indicate the way the are interacting. The possible values are: TANDEM, SYNCHRONOUS, MASTER, and SLAVE. The coupling **MUST** be view from the perspective of the axis, therefore a MASTER coupling indicates that this axis is the master of the COUPLED_AXES.
BLOCK	The block of code being executed. The block contains the entire expression of the step in the program.
~~CODE~~	~~The programmatic code being executed~~. DEPRECATED
CONTROLLER_MODE	The current controller's mode. AUTOMATIC, MANUAL, or 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 rotational direction of the Axis. CLOCKWISE or COUNTER_CLOCKWISE
DOOR_STATE	The opened or closed state of the door. OPEN 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, or STOPPED
LINE	The current line of code being executed
MAXIMUM	The maximum line number of the code being executed.
MINIMUM	The minimum line number of the code being executed.
MESSAGE	An uninterpreted textual notification.
PART_COUNT	The current count of parts produced as represented by the controller. 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.

Data Item type/subtype	Description
CONTROL	The state of the low power line.
~~**POWER_STATUS**~~	~~The ON or OFF status of the component.~~ DEPRECATED
PROGRAM	The name of the program being executed
ROTARY_MODE	The mode for the Rotary axis. SPINDLE, INDEX, or CONTOUR.
TOOL_ID	The identifier of the tool currently in use for a given Path
WORKHOLDING_ID	The identifier for the workholding currently in use for a given Path

4.2.3 Data Item Types for CONDITION Category

These are items that indicate the devices' health and ability to operate. They are reported differently than Samples or Events: they **MUST** be reported as Normal, Warning, and Fault. Unlike the other two categories, a Component or Device **MAY** have values for a Condition type DataItem that has multiple concurrently active values at any point in time.

Data Item type/ qualifier	Description
ACTUATOR	A motion servo or actuator related condition.
AMPERAGE	A high or low condition for the electrical current.
COMMUNICATIONS	A communications failure indicator.
HARDWARE	The hardware subsystem of the component operation condition.
LEVEL	Represents the level of a resource.
LOAD	Indicates the load of a component is within operating limits.
LOGIC_PROGRAM	An error occurred in the logic program or PLC (programmable logic controller).
MOTION_PROGRAM	An error occurred in the motion program.
POSITION	The component's position is within operational limits.
PRESSURE	Indicates the pressure of a component is within operating limits.
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 is within operating limits.
VOLTAGE	A high or low voltage condition.
VELOCITY	A fault or warning with regard to the velocity of the component.

5 Component and Data Item Relationships

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

5.1 Overview

At the top level, a device **MUST** always contain an Availability data item that represents this device is available to do work. Any component **MAY** also include an arbitrary set of sensors as sub-components. The sensor is currently a placeholder for extensible data collection devices and is not modeled in this version of the specification. A sensor will be an external device that will collect data and report it to the *Agent*. The sensor **MUST** be correctly associated with its most relevant component. The rules governing this association will be covered in a later version of this specification.

5.2 Device

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

5.2.1 DataItem types

- EMERGENCY_STOP - The emergency stop state of the machine.
- AVAILABILITY - **Required**

5.2.2 Sub-components of Device

- Axes
- Controller
- Systems
- Door

5.3 Common Components and Data Items

A common set of DataItems have been created to provide the flexibility to define a wide variety of information about a machine or process. Any DataItem can be used with and Device or Component providing that the standard naming conventions are implemented.

5.3.1 Axes

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

5.3.1.1 DataItem types

- ~~GLOBAL_POSITION~~ - DEPRECATED
- ~~PATH_FEEDRATE~~ - Moved to Path

- ~~ACCELERATION~~ – Moved to Path
- ~~VELOCITY~~ – Moved to Path

5.3.1.2 Sub-components of Axes

- Linear
- Rotary
- ~~Spindle~~ - DEPRECATED

5.3.2 `Linear` (Subcomponent of `Axes`)

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

5.3.2.1 DataItem types

- ACCELERATION
- AXIS_FEEDRATE
- LOAD
- POSITION
- SLAVE_OF_AXIS
- VELOCITY

5.3.2.2 Condition types

- AMPERAGE
- LOAD
- POSITION
- TEMPERATURE
- VOLTAGE

5.3.3 `Rotary` (Subcomponent of `Axes`)

A rotary axis revolves around a point.

5.3.3.1 DataItem types

- ANGLE
- ANGULAR_ACCELERATION
- ANGULAR_VELOCITY
- AXIS_FEEDRATE
- DIRECTION
- LOAD
- ROTARY_MODE
- SLAVE_OF_AXIS
- SPINDLE_SPEED
- TORQUE

5.3.3.2 Condition types

- ANGLE
- AMPERAGE
- LOAD

- TEMPERATURE
- VOLTAGE
- VELOCITY

5.3.4 Controller

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

5.3.4.1 Sub-components of Controller

- Path

5.3.4.2 DataItem types

- BLOCK
- ~~CODE~~ **DEPRECATED**
- CONTROLLER_MODE
- EXECUTION
- EMERGENCY_STOP
- LINE
- MESSAGE
- PART_COUNT
- PART_ID
- PATH_FEEDRATE
- PATH_POSITION
- PROGRAM
- TOOL_ID
- WORKHOLDING_ID

5.3.4.3 Condition types

- COMMUNICATIONS
- HARDWARE
- LOGIC_PROGRAM
- MOTION_PROGRAM
- SYSTEM

5.3.5 Path (Subcomponent of Controller)

A Path represents the motion of a control point as it moves through space as controlled by a set of control instructions (i.e. vector move).

DataItem types

- ACTIVE_AXES
- AXES_COUPLING
- ACCELERATION
- BLOCK
- ~~CODE~~ **DEPRECATED**
- COUPLED_AXES

- CONTROLLER_MODE
- EXECUTION
- LINE
- MESSAGE
- PART_COUNT
- PART_ID
- PATH_FEEDRATE
- PATH_POSITION
- PROGRAM
- TOOL_ID
- VELOCITY
- WORKHOLDING_ID

5.3.5.1 Condition types

- MOTION_PROGRAM

~~5.3.6 Power~~ DEPRECATED

~~The power component represents the electrical activation of the component. The data items the power component can collect are a simple status (on/off) and three power related measurements, voltage, amperage and watts. There are no sub-components of Power. The reason for making this a separate component is the need to support legacy equipment.~~

~~For the top-level Device Power component, the PowerStatus represents the power to all components of the device except the computer controller, since the controller may be hosting the MTConnect® *Agent*, it would be impossible to report Power ON or OFF if the controller is off. Therefore, if network or physical connectivity to the device is interrupted, the Power **MUST** be considered OFF.~~

~~For all other components, the definition of OFF is the component is not connected to the power source.~~

~~5.3.6.1 DataItem types~~

- ~~POWER_STATUS DEPRECATED.~~
- ~~VOLTAGE~~
- ~~AMPERAGE~~
- ~~WATTS~~

~~5.3.6.2 Condition types~~

- ~~VOLTAGE~~
- ~~AMPERAGE~~

5.3.7 Thermostat

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

5.3.7.1 DataItem types

- TEMPERATURE

5.3.7.2 Condition types

- TEMPERATURE

5.3.8 Vibration

A sensor capable of measuring the vibration of a component.

5.3.8.1 DataItem types

- ACCELERATION
- DISPLACEMENT
- FREQUENCY
- VELOCITY

5.3.8.2 Condition types

- DISPLACEMENT
- VIBRATION

5.3.9 Pressure

A sensor capable of measuring the pressure.

5.3.9.1 DataItem types

- PRESSURE

5.3.9.2 Condition types

- PRESSURE

5.3.10 Door

A opening that can be closed.

5.3.10.1 DataItem types

- DOOR_STATE

5.3.11 Actuator

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

5.3.11.1 DataItem types

- ACCELERATION
- ANGLE
- ANGULAR_ACCELATION
- ANGULAR_VELOCITY
- LOAD
- POSITION
- PRESSURE
- VELOCITY
- TEMPERATURE
- TORQUE

5.3.11.2 Condition types

- AMPERAGE
- LOAD
- POSITION
- PRESSURE
- TEMPERATURE
- VOLTAGE

5.3.12 Systems

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

5.3.12.1 Sub-components of Systems

- Hydraulic
- Pneumatic
- Coolant
- Lubrication

5.3.13 Hydraulic (Subcomponent of Systems)

A component representing the hydraulics of a device.

5.3.13.1 DataItem types

- PRESSURE
- TEMPERATURE

5.3.13.2 Condition types

- PRESSURE
- TEMPERATURE
- LEVEL

5.3.14 Coolant (Subcomponent of Systems)

A component representing the coolant of a device.

5.3.14.1 DataItem types

- PRESSURE
- TEMPERATURE

5.3.14.2 Condition types

- LEVEL
- PH
- PRESSURE
- TEMPERATURE

5.3.15 Lubrication (Subcomponent of Systems)

A component representing the lubricant of a device.

5.3.15.1 DataItem types

- PRESSURE
- TEMPERATURE

5.3.15.2 Condition types

- PRESSURE
- TEMPERATURE
- LEVEL

5.3.16 `Electric` (Subcomponent of `Systems`)

5.3.16.1 DataItem types

- VOLTAGE
- AMPERAGE
- WATTS

5.3.16.2 Condition types

- VOLTAGE
- AMPERAGE

~~5.4 Cutting Machine Tool Components and Data Items~~

5.4.1 `Spindle` - DEPRECATED

~~The spindle is a rotational axis that revolves at high speed and has its speed expressed in REVOLUTION/MINUTE. The spindle can also have additional data items. Spindle speed has been specified as a separate data item since it receives special treatment in many applications. Velocity is used for linear axes other than spindle.~~

~~5.4.1.1 DataItem types~~

- ~~SPINDLE_SPEED~~
- ~~LOAD~~
- ~~DIRECTION~~
- ~~TORQUE~~

6 Annotated XML Examples

6.1 Simplest Device

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

http://10.1.23.10/ LinuxCNC/probe

The *Agent* responds as follows:

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

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

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

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

```
7.          <DataItems>
8.            <DataItem type="AVAILABILITY" name="avail" category="EVENT"
     id="a"/>
9.          </DataItems>
```

As was stated before, the device is only required to have one `AVAILABILITY` data item which **MUST** report the devices represented availability to communicate. The DataItem on line 13 has an id of a. This will allow events responding to this data item to be easily associated.

```
10.         </Components>
11.       </Device>
12.     </Devices>
13.   </MTConnectDevices>
```

6.2 More Complex Example of probe

The sample was generated with the following request:

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

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

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

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

```
6. <Description manufacturer="NIST" serialNumber="01"/>
7.   <DataItems>
  8.         <DataItem type="AVAILABILITY" name="avail" category="EVENT"
  id="a"/>
9.   </DataItems>
10.  <Components>
11.    <Axes name="Axes" id="3">
```

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

```
12.       <Components>
13.         <Rotary name="C" id="c1">
14.         <DataItems>
15.           <DataItem type="SPINDLE_SPEED" name="Cspeed" category="SAMPLE"
      id="c2" nativeUnits="REVOLUTION/MINUTE" subType="ACTUAL"
      units="REVOLUTION/MINUTE">
16.            <Source>Sspeed</Source>
17.           </DataItem>
18.           <DataItem type=”ROTARY_MODE” name”Cmode” category=”EVENT”
      id=”c3”>
19.            <Values><Value>SPINDLE</Value><Values>
20.           </DataItem>
21.         </DataItems>
```

```
22.      </Rotary>
```

The spindle component declared on line 16 is the S axis and has spindle-specific data items.

```
23.     <Linear name="X" id="x1">
24.       <DataItems>
25.         <DataItem type="POSITION" name="Xact" category="SAMPLE" id="x2"
      nativeUnits="MILLIMETER" subType="ACTUAL" units="MILLIMETER"/>
26.         <DataItem type="POSITION" name="Xcom" category="SAMPLE" id="x3"
      nativeUnits="MILLIMETER" subType="COMMANDED" units="MILLIMETER"/>
27.       </DataItems>
28.     </Linear>
29.     <Linear name="Y" id="y1">
30.       <DataItems>
31.         <DataItem type="POSITION" name="Yact" category="SAMPLE" id="y2"
      nativeUnits="MILLIMETER" subType="ACTUAL" units="MILLIMETER"/>
32.         <DataItem type="POSITION" name="Ycom" category="SAMPLE" id="y3"
      nativeUnits="MILLIMETER" subType="COMMANDED" units="MILLIMETER"/>
33.       </DataItems>
34.     </Linear>
35.     <Linear name="Z" id="z1">
36.      <DataItems>
37.        <DataItem type="POSITION" name="Zact" category="SAMPLE" id="z2"
      nativeUnits="MILLIMETER" subType="ACTUAL" units="MILLIMETER"/>
38.        <DataItem type="POSITION" name="Zcom" category="SAMPLE" id="z3"
      nativeUnits="MILLIMETER" subType="COMMANDED" units="MILLIMETER"/>
39.      </DataItems>
40.     </Linear>
```

Lines 24, 30, and 36 define the three linear axes X, Y, and Z respectively. In this example device the *Agent* is only collecting the actual and commanded positions.

```
41.   </Components>
42.  </Axes>
```

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

```
43.  <Controller name="Controller" id="8">
44.    <Components>
45.      <Path id=”p1” name=”path”>
46.        <DataItems>
47.          <DataItem type="LINE" name="line" category="EVENT" id="p1"/>
48.          <DataItem type="CONTROLLER_MODE" name="mode" category="EVENT"
      id="p2"/>
49.          <DataItem type="PROGRAM" name="program" category="EVENT"
      id="p3"/>
```

```
50.          <DataItem type="EXECUTION" name="execution" category="EVENT"
     id="p4"/>
51.          <DataItem type="PATH_FEEDRATE" name="feedrate" category="SAMPLE"
     id="p4" units="MILLIMETER/SECOND" nativeUnits="MILLIMETER/SECOND" />
52.          <DataItem type="PATH_POSITION" name="position" category="SAMPLE"
     id="p4" units="MILLIMETER_3D" nativeUnits="INCH_3D"/>
53.        </DataItems>
54.      </Path>
55.    </Components>
56.  </Controller>
57.  </Components>
58.  </Device>
59.  </Devices>
60.  </MTConnectDevices>
```

Appendices

A. Bibliography

1. Engineering Industries Association. *EIA Standard - EIA-274-D*, Interchangeable Variable, Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically Controlled Machines. Washington, D.C. 1979.

2. ISO TC 184/SC4/WG3 N1089. *ISO/DIS 10303-238*: Industrial automation systems and integration Product data representation and exchange Part 238: Application Protocols: Application interpreted model for computerized numerical controllers. Geneva, Switzerland, 2004.

3. International Organization for Standardization. *ISO 14649*: Industrial automation systems and integration – Physical device control – Data model for computerized numerical controllers – Part 10: General process data. Geneva, Switzerland, 2004.

4. International Organization for Standardization. *ISO 14649*: Industrial automation systems and integration – Physical device control – Data model for computerized numerical controllers – Part 11: Process data for milling. Geneva, Switzerland, 2000.

5. International Organization for Standardization. *ISO 6983/1* – Numerical Control of machines – Program format and definition of address words – Part 1: Data format for positioning, line and contouring control systems. Geneva, Switzerland, 1982.

6. Electronic Industries Association. *ANSI/EIA-494-B-1992*, 32 Bit Binary CL (BCL) and 7 Bit ASCII CL (ACL) Exchange Input Format for Numerically Controlled Machines. Washington, D.C. 1992.

7. National Aerospace Standard. *Uniform Cutting Tests* - NAS Series: Metal Cutting Equipment Specifications. Washington, D.C. 1969.

8. International Organization for Standardization. *ISO 10303-11*: 1994, Industrial automation systems and integration Product data representation and exchange Part 11: Description methods: The EXPRESS language reference manual. Geneva, Switzerland, 1994.

9. International Organization for Standardization. *ISO 10303-21*: 1996, Industrial automation systems and integration -- Product data representation and exchange -- Part 21: Implementation methods: Clear text encoding of the exchange structure. Geneva, Switzerland, 1996.

10. H.L. Horton, F.D. Jones, and E. Oberg. *Machinery's handbook*. Industrial Press, Inc. New York, 1984.

11. International Organization for Standardization. *ISO 841-2001: Industrial automation systems and integration - Numerical control of machines - Coordinate systems and motion nomenclature*. Geneva, Switzerland, 2001.

12. *ASME B5.59-2 Version 9c: Data Specification for Properties of Machine Tools for Milling and Turning. 2005.*

13. *ASME/ANSI B5.54: Methods for Performance Evaluation of Computer Numerically Controlled Lathes and Turning Centers. 2005.*

14. OPC Foundation. *OPC Unified Architecture Specification, Part 1: Concepts Version 1.00. July 28, 2006.*

B. Machine Tool Modeling

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

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

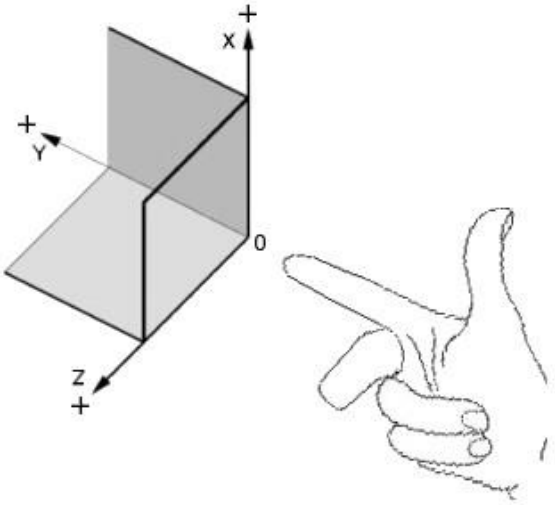


Figure 7: Right Hand Rule Coordinate Planes

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

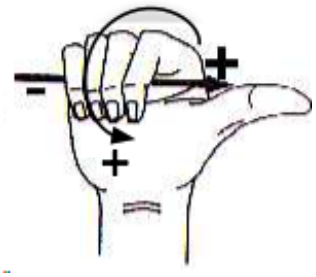


Figure 8: Rotational Right Hand Rule

B.1. Vertical Three Axis Mill

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

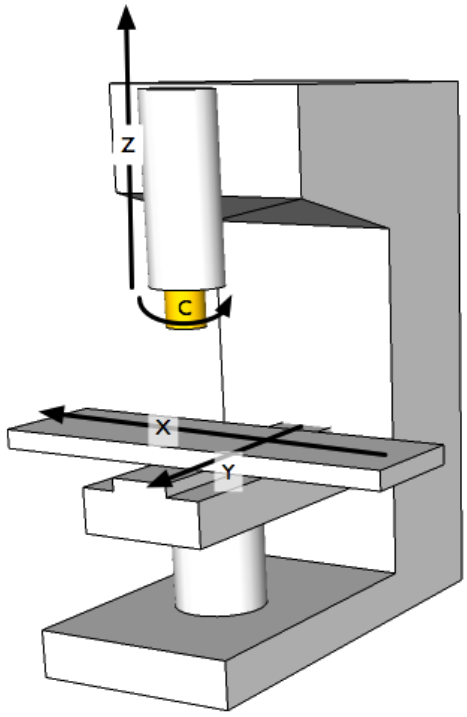


Figure 9: Three Axis Mill

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

```
1.  <?xml version="1.0" encoding="UTF-8"?>
2.  <MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1 MTConnectDevices.xsd">
3.    <Header bufferSize="130000" instanceId="1" creationTime="2009-11-13T02:31:40" sender="local"
    version="1.1"/>
4.    <Devices>
5.      <Device id="d1" uuid="HM1" name="HMC_3Axis">
6.      <Description>3 Axis Mill</Description>
7.      <Components>
8.        <Axes id="a" name="base">
9.          <Components>
10.           <Linear id="y" name="Y">
```

```
931   11.             <DataItems>
932   12.               <DataItem type="POSITION" subType="ACTUAL" id="yp" category="SAMPLE"
933       name="Yact" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
934   13.             </DataItems>
935   14.           </Linear>
936   15.           <Linear id="x" name="X">
937   16.             <DataItems>
938   17.               <DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"
939       name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
940   18.             </DataItems>
941   19.           </Linear>
942   20.           <Linear id="z" name="Z">
943   21.             <DataItems>
944   22.               <DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact" sub-
945       Type="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
946   23.             </DataItems>
947   24.           </Linear>
948   25.           <Rotary id="c" name="C">
949   26.             <DataItems>
950   27.               <DataItem type="SPINDLE_SPEED" id="cspd" category="SAMPLE" name="Sspeed"
951       subType="ACTUAL" units="REVOLUTION/MINUTE" nativeUnits="REVOLUTION/MINUTE"/>
952   28.               <DataItem type="SPINDLE_SPEED" id="cso" category="SAMPLE" name="Sovr" sub-
953       Type="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>
954   29.               <DataItem type="ROTARY_MODE" id="rf" category="EVENT" name="rfunc">
955   30.                 <Constraints>
956   31.                   <Value>SPINDLE</Value>
957   32.                 </Constraints>
958   33.               </DataItem>
959   34.             </DataItems>
960   35.           </Rotary>
961   36.         </Components>
962   37.       </Axes>
963   38.       <Controller id="cont" name="controller">
964   39.         <Components>
965   40.           <Path id="path" name="path">
966   41.             <DataItems>
967   42.               <DataItem type="PROGRAM" id="pgm" category="EVENT" name="program"/>
968   43.               <DataItem type="BLOCK" id="blk" category="EVENT" name="block"/>
969   44.               <DataItem type="LINE" id="ln" category="EVENT" name="line"/>
970   45.               <DataItem type="PATH_FEEDRATE" id="pf" category="SAMPLE" name="Fact"
971       units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL"/>
972   46.               <DataItem type="PATH_FEEDRATE" id="pfo" category="SAMPLE" name="Fovr"
973       units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>
974   47.               <DataItem type="PATH_POSITION" id="pp" category="SAMPLE" name="Ppos"
975       units="MILLIMETER_3D" nativeUnits="FOOT_3D" coordinateSystem="WORK"/>
976   48.               <DataItem type="EXECUTION" id="exec" category="EVENT" name="execution"/>
977   49.               <DataItem type="CONTROLLER_MODE" id="cm" category="EVENT" name="mode"/>
978   50.             </DataItems>
979   51.           </Path>
980   52.         </Components>
981   53.       </Controller>
982   54.     </Components>
983   55.   </Device>
984   56. </Devices>
985   57. </MTConnectDevices>
```

B.2. Two Axis Lathe

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

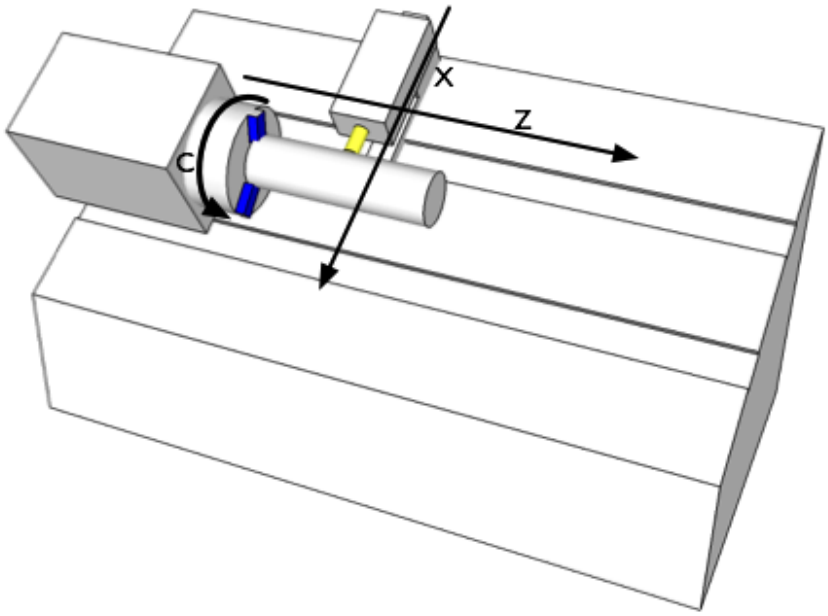


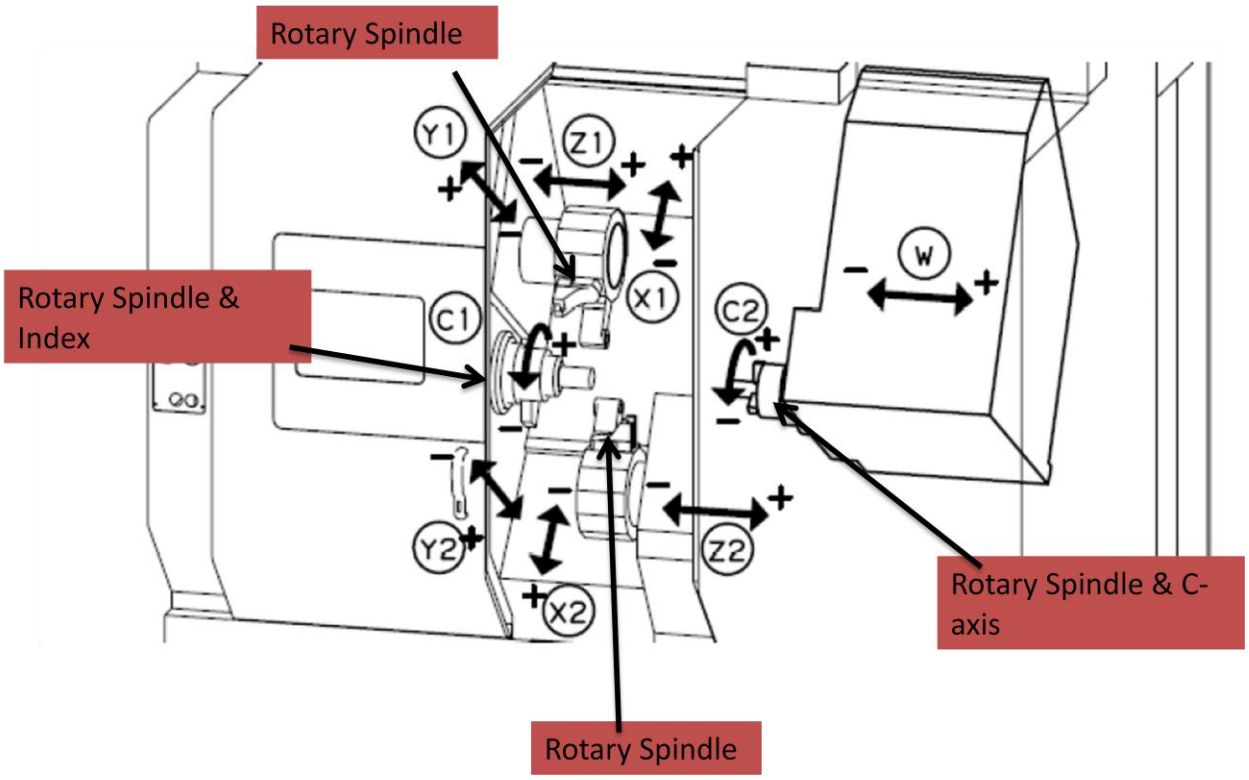
Figure 10: Two Axis Lathe

```
1.  <?xml version="1.0" encoding="UTF-8"?>
2.  <MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1 MTConnectDevices.xsd">
3.    <Header bufferSize="130000" instanceId="1" creationTime="2009-11-13T02:31:40" sender="local"
    version="1.1"/>
4.    <Devices>
5.      <Device id="d1" uuid="HM1" name="HMC_3Axis">
6.      <Description>3 Axis Mill</Description>
7.      <Components>
8.        <Axes id="a" name="base">
9.          <Components>
10.           <Linear id="x" name="X">
11.             <DataItems>
12.               <DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"
    name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
13.             </DataItems>
14.           </Linear>
15.           <Linear id="z" name="Z">
16.             <DataItems>
17.               <DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact" sub-
    Type="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
18.             </DataItems>
```

```
19.          </Linear>
20.          <Rotary id="c" name="C">
21.            <DataItems>
22.              <DataItem type="SPINDLE_SPEED" id="cspd" category="SAMPLE" name="Sspeed"
     subType="ACTUAL" units="REVOLUTION/MINUTE" nativeUnits="REVOLUTION/MINUTE"/>
23.              <DataItem type="SPINDLE_SPEED" id="cso" category="SAMPLE" name="Sovr" sub-
     Type="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>
24.              <DataItem type="ROTARY_MODE" id="rf" category="EVENT" name="rfunc">
25.                <Constraints>
26.                  <Value>SPINDLE</Value>
27.                  <Value>INDEX</Value>
28.                </Constraints>
29.              </DataItem>
30.            </DataItems>
31.          </Rotary>
32.        </Components>
33.      </Axes>
34.      <Controller id="cont" name="controller">
35.        <Components>
36.          <Path id="path" name="path">
37.            <DataItems>
38.              <DataItem type="PROGRAM" id="pgm" category="EVENT" name="program"/>
39.              <DataItem type="BLOCK" id="blk" category="EVENT" name="block"/>
40.              <DataItem type="LINE" id="ln" category="EVENT" name="line"/>
41.              <DataItem type="PATH_FEEDRATE" id="pf" category="SAMPLE" name="Fact"
     units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL"/>
42.              <DataItem type="PATH_FEEDRATE" id="pfo" category="SAMPLE" name="Fovr"
     units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>
43.              <DataItem type="PATH_POSITION" id="pp" category="SAMPLE" name="Ppos"
     units="MILLIMETER_3D" nativeUnits="FOOT_3D" coordinateSystem="WORK"/>
44.              <DataItem type="EXECUTION" id="exec" category="EVENT" name="execution"/>
45.              <DataItem type="CONTROLLER_MODE" id="cm" category="EVENT" name="mode"/>
46.            </DataItems>
47.          </Path>
48.        </Components>
49.      </Controller>
50.    </Components>
51.  </Device>
52. </Devices>
53. </MTConnectDevices>
```

B.3. HyperQuadrex

Mazak - HyperQuadrex




```
<?xml version="1.0" encoding="UTF-8"?>
<MTConnectDevices xmlns="urn:mtconnect.com:MTConnectDevices:1.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:mtconnect.com:MTConnectDevices:1.1 ../MTConnectDevices.xsd">
  <Header bufferSize="130000" instanceId="1" creationTime="2009-11-13T02:31:40" sender="local" ver-
sion="1.1"/>
  <Devices>
    <Device id="d1" uuid="HM1" name="HyperQuadrex">
      <Description>Mazak - HyperQuadrex</Description>
      <Components>
        <Axes id="a" name="base">
          <Components>
            <Linear id="x" name="X" nativeName="X1">
              <DataItems>
                <DataItem type="POSITION" subType="ACTUAL" id="xp" category="SAMPLE"
name="Xact" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE">
                  <Source>X1pos</Source>
                </DataItem>
                <DataItem type="LOAD" id="xl" category="SAMPLE" name="Xload" units="NEWTON">
                  <Source>X1load</Source>
                </DataItem>
              </DataItems>
```

```
1081            </Linear>
1082            <Linear id="y" name="Y" nativeName="Y1">
1083              <DataItems>
1084                <DataItem type="POSITION" subType="ACTUAL" id="yp" category="SAMPLE"
1085    name="Yact" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE">
1086                  <Source>Y1pos</Source>
1087                </DataItem>
1088                <DataItem type="LOAD" id="yl" category="SAMPLE" name="Yload" units="NEWTON">
1089                  <Source>Y1load</Source>
1090                </DataItem>
1091              </DataItems>
1092            </Linear>
1093            <Linear id="z" name="Z" nativeName="Z1">
1094              <DataItems>
1095                <DataItem type="POSITION" id="zp" category="SAMPLE" name="Zact" sub-
1096    Type="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE">
1097                  <Source>Z1pos</Source>
1098                </DataItem>
1099                <DataItem type="LOAD" id="zl" category="SAMPLE" name="Zload" units="NEWTON">
1100                  <Source>Z1load</Source>
1101                </DataItem>
1102              </DataItems>
1103            </Linear>
1104            <Linear id="x2" name="X2" >
1105              <DataItems>
1106                <DataItem type="POSITION" subType="ACTUAL" id="x2p" category="SAMPLE"
1107    name="X2act" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
1108                <DataItem type="LOAD" id="x2l" category="SAMPLE" name="X2load" units="NEWTON">
1109                  <Source>X2load</Source>
1110                </DataItem>
1111              </DataItems>
1112            </Linear>
1113            <Linear id="y2" name="Y2">
1114              <DataItems>
1115                <DataItem type="POSITION" subType="ACTUAL" id="y2p" category="SAMPLE"
1116    name="Y2act" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE"/>
1117                <DataItem type="LOAD" id="y2l" category="SAMPLE" name="Y2load" units="NEWTON"/>
1118              </DataItems>
1119            </Linear>
1120            <Linear id="z2" name="Z2">
1121              <DataItems>
1122                <DataItem type="POSITION" id="z2p" category="SAMPLE" name="Z2act" sub-
1123    Type="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE">
1124                  <Source>Z2pos</Source>
1125                </DataItem>
1126                <DataItem type="LOAD" id="z2l" category="SAMPLE" name="Z2load" units="NEWTON"/>
1127              </DataItems>
1128            </Linear>
1129            <Linear id="z3" name="Z3" nativeName="W">
1130              <DataItems>
1131                <DataItem type="POSITION" id="z3p" category="SAMPLE" name="Z3act" sub-
1132    Type="ACTUAL" units="MILLIMETER" nativeUnits="MILLIMETER" coordinateSystem="MACHINE">
1133                  <Source>Wpos</Source>
1134                </DataItem>
1135                <DataItem type="LOAD" id="z3l" category="SAMPLE" name="Z3load" units="NEWTON">
1136                  <Source>Wload</Source>
```

```
1137              </DataItem>
1138            </DataItems>
1139          </Linear>
1140          <Rotary id="c" name="C " nativeName="C1">
1141            <DataItems>
1142              <DataItem type="LOAD" id="Cl" category="SAMPLE" name="Cload" units="NEWTON"/>
1143              <DataItem type="SPINDLE_SPEED" id="cspd" category="SAMPLE" name="Sspeed" sub-
1144  Type="ACTUAL" units="REVOLUTION/MINUTE" nativeUnits="REVOLUTION/MINUTE"/>
1145              <DataItem type="SPINDLE_SPEED" id="cso" category="SAMPLE" name="Sovr" sub-
1146  Type="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>
1147              <DataItem type="DIRECTION" id="cdir" category="EVENT" name="Sdir"/>
1148              <DataItem type="ANGLE" id="cpos" category="SAMPLE" name="Cpos" sub-
1149  Type="ACTUAL" units="DEGREE" nativeUnits="DEGREE" nativeScale="-1.0"/>
1150              <DataItem type="ROTARY_MODE" id="rf" category="EVENT" name="rfunc">
1151                <Constraints>
1152                  <Value>SPINDLE</Value>
1153                  <Value>INDEX</Value>
1154                </Constraints>
1155              </DataItem>
1156            </DataItems>
1157          </Rotary>
1158          <Rotary id="c2" name="C2">
1159            <DataItems>
1160              <DataItem type="LOAD" id="C2l" category="SAMPLE" name="C2load" units="NEWTON"/>
1161              <DataItem type="SPINDLE_SPEED" id="c2spd" category="SAMPLE" name="Sspeed" sub-
1162  Type="ACTUAL" units="REVOLUTION/MINUTE" nativeUnits="REVOLUTION/MINUTE"/>
1163              <DataItem type="SPINDLE_SPEED" id="c2so" category="SAMPLE" name="Sovr" sub-
1164  Type="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>
1165              <DataItem type="DIRECTION" id="c2dir" category="EVENT" name="S2dir"/>
1166              <DataItem type="ROTARY_MODE" id="rf2" category="EVENT" name="rfunc">
1167                <Constraints>
1168                  <Value>SPINDLE</Value>
1169                </Constraints>
1170              </DataItem>
1171            </DataItems>
1172          </Rotary>
1173          <Rotary id="b" name="B" nativeName="S1">
1174            <DataItems>
1175              <DataItem type="LOAD" id="bl" category="SAMPLE" name="Bload" units="NEWTON"/>
1176              <DataItem type="SPINDLE_SPEED" id="bspd" category="SAMPLE" name="Sspeed" sub-
1177  Type="ACTUAL" units="REVOLUTION/MINUTE" nativeUnits="REVOLUTION/MINUTE"/>
1178              <DataItem type="SPINDLE_SPEED" id="bso" category="SAMPLE" name="Sovr" sub-
1179  Type="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>
1180              <DataItem type="DIRECTION" id="bdir" category="EVENT" name="S3dir"/>
1181              <DataItem type="ROTARY_MODE" id="brf" category="EVENT" name="rfunc">
1182                <Constraints>
1183                  <Value>SPINDLE</Value>
1184                </Constraints>
1185              </DataItem>
1186            </DataItems>
1187          </Rotary>
1188          <Rotary id="b2" name="B2" nativeName="S2">
1189            <DataItems>
1190              <DataItem type="LOAD" id="b2l" category="SAMPLE" name="B2load" units="NEWTON"/>
1191              <DataItem type="SPINDLE_SPEED" id="b2spd" category="SAMPLE" name="Sspeed" sub-
1192  Type="ACTUAL" units="REVOLUTION/MINUTE" nativeUnits="REVOLUTION/MINUTE"/>
```

```
1193              <DataItem type="SPINDLE_SPEED" id="b2so" category="SAMPLE" name="Sovr" sub-
1194    Type="OVERRIDE" units="PERCENT" nativeUnits="PERCENT"/>
1195              <DataItem type="DIRECTION" id="b2dir" category="EVENT" name="S3dir"/>
1196              <DataItem type="ROTARY_MODE" id="b2rf" category="EVENT" name="rfunc">
1197                <Constraints>
1198                  <Value>SPINDLE</Value>
1199                </Constraints>
1200              </DataItem>
1201            </DataItems>
1202          </Rotary>
1203        </Components>
1204      </Axes>
1205      <Controller id="cont" name="controller">
1206        <Components>
1207          <Path id="path1" name="path1">
1208            <DataItems>
1209              <DataItem type="ACTIVE_AXES" category="EVENT" name="axes" id="act_axes1"/>
1210              <DataItem type="PROGRAM" id="pgm1" category="EVENT" name="program"/>
1211              <DataItem type="BLOCK" id="blk1" category="EVENT" name="block"/>
1212              <DataItem type="LINE" id="ln1" category="EVENT" name="line"/>
1213              <DataItem type="PATH_FEEDRATE" id="pf1" category="SAMPLE" name="Fact"
1214    units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL" coordinateSys-
1215    tem="WORK"/>
1216              <DataItem type="PATH_FEEDRATE" id="pfo1" category="SAMPLE" name="Fovr"
1217    units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>
1218              <DataItem type="PATH_POSITION" id="pp1" category="SAMPLE" name="Ppos"
1219    units="MILLIMETER_3D" nativeUnits="MILLIMETER_3D" coordinateSystem="WORK"/>
1220              <DataItem type="TOOL_ID" id="tid1" category="EVENT" name="Tid"/>
1221              <DataItem type="PART_ID" id="pid1" category="EVENT" name="Pid"/>
1222              <DataItem type="EXECUTION" id="exec1" category="EVENT" name="execution"/>
1223              <DataItem type="CONTROLLER_MODE" id="cm1" category="EVENT" name="mode"/>
1224            </DataItems>
1225          </Path>
1226          <Path id="path2" name="path2">
1227            <DataItems>
1228              <DataItem type="ACTIVE_AXES" category="EVENT" name="axes" id="act_axes2"/>
1229              <DataItem type="PROGRAM" id="pgm2" category="EVENT" name="program"/>
1230              <DataItem type="BLOCK" id="blk2" category="EVENT" name="block"/>
1231              <DataItem type="LINE" id="ln2" category="EVENT" name="line"/>
1232              <DataItem type="PATH_FEEDRATE" id="pf2" category="SAMPLE" name="Fact"
1233    units="MILLIMETER/SECOND" nativeUnits="FOOT/MINUTE" subType="ACTUAL" coordinateSys-
1234    tem="WORK"/>
1235              <DataItem type="PATH_FEEDRATE" id="pfo2" category="SAMPLE" name="Fovr"
1236    units="PERCENT" nativeUnits="PERCENT" subType="OVERRIDE"/>
1237              <DataItem type="PATH_POSITION" id="pp2" category="SAMPLE" name="Ppos" units="
1238    MILLIMETER_3D" nativeUnits=" MILLIMETER_3D" coordinateSystem="WORK"/>
1239              <DataItem type="TOOL_ID" id="tid2" category="EVENT" name="Tid"/>
1240              <DataItem type="PART_ID" id="pid2" category="EVENT" name="Pid"/>
1241              <DataItem type="EXECUTION" id="exec2" category="EVENT" name="execution"/>
1242              <DataItem type="CONTROLLER_MODE" id="cm2" category="EVENT" name="mode"/>
1243            </DataItems>
1244          </Path>
1245        </Components>
1246      </Controller>
1247      <Door id="d" name="door">
1248        <DataItems>
```

```
1249            <DataItem id="ds" category="EVENT" name="door" type="DOOR_STATE"/>
1250          </DataItems>
1251        </Door>
1252      </Components>
1253    </Device>
1254  </Devices>
1255 </MTConnectDevices>
1256
1257
```