MTConnect® Standard
Part 2.0 – Devices Information Model
Version 1.6.0

Prepared for: MTConnect Institute
Prepared on: July 15, 2020

MTConnect® is a registered trademark of AMT - The Association for Manufacturing Technology. Use of MTConnect is limited to use as specified on [http://www.mtconnect.org/]
MTConnect Specification and Materials

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

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

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

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

1 Purpose of This Document .............................................. 2

2 Terminology and Conventions .......................................... 3
   2.1 Glossary .................................................................. 3
   2.2 Acronyms .................................................................. 11
   2.3 MTConnect References ............................................. 11

3 Devices Information Model ............................................... 12

4 Structural Elements for MTConnectDevices ......................... 14
   4.1 Devices .................................................................... 17
   4.2 Device ...................................................................... 18
   4.3 Components ............................................................. 18
   4.4 Component ............................................................... 18
      4.4.1 XML Schema Structure for Component ................... 19
      4.4.2 Attribute for Component ........................................ 21
      4.4.3 Elements of Component ........................................ 24
         4.4.3.1 Description for Component ............................... 24
         4.4.3.2 Configuration for Component .......................... 26
         4.4.3.3 DataItems for Component ................................. 27
         4.4.3.4 Components within Component ......................... 28
         4.4.3.5 Compositions for Component ............................. 28
         4.4.3.6 References for Component .................................. 28
   4.5 Compositions ............................................................ 28
   4.6 Composition ............................................................. 29
      4.6.1 XML Schema Structure for Composition ................ 30
      4.6.2 Attributes for Composition .................................... 31
      4.6.3 Elements of Composition ..................................... 32
         4.6.3.1 Description for Composition .............................. 33
   4.7 References ................................................................ 34
   4.8 Reference .................................................................. 35
      4.8.1 ComponentRef ..................................................... 36
      4.8.2 DataItemRef ......................................................... 37

5 Component Structural Elements ......................................... 39
   5.1 Axes ....................................................................... 41
      5.1.1 Cartesian Coordinate Naming Conventions ............... 41
         5.1.1.1 Linear Motion .................................................. 42
         5.1.1.2 Rotary Motion .................................................. 42
      5.1.2 Articulated Machine Control Systems ..................... 42
      5.1.3 Articulated Machine Axis Names ............................. 42
5.1.4 Rotary Component ........................................ 43
5.1.5 Linear Component ......................................... 43
5.2 Controller ..................................................... 43
5.2.1 Path ......................................................... 43
5.3 Systems ....................................................... 44
5.3.1 Hydraulic System .......................................... 44
5.3.2 Pneumatic System ........................................ 44
5.3.3 Coolant System ........................................... 44
5.3.4 Lubrication System ....................................... 45
5.3.5 Electric System ........................................... 45
5.3.6 Enclosure System ......................................... 45
5.3.7 Protective System ......................................... 45
5.3.8 ProcessPower System ..................................... 45
5.3.9 Feeder System ............................................. 46
5.3.10 Dielectric System ........................................ 46
5.3.11 EndEffector System ...................................... 46
5.3.12 WorkEnvelope System ................................... 46
5.4 Auxiliaries .................................................... 46
5.4.1 Loader System ............................................. 47
5.4.2 WasteDisposal System .................................... 47
5.4.3 ToolingDelivery System .................................. 47
5.4.4 BarFeeder System ......................................... 47
5.4.5 Environmental System .................................... 47
5.4.6 Sensor System ............................................. 48
5.4.7 Deposition System ........................................ 48
5.5 Resources .................................................... 48
5.5.1 Materials .................................................. 48
5.5.1.1 Stock .................................................... 49
5.5.2 Personnel ................................................... 49
5.6 Interfaces ..................................................... 49
5.7 Other Components ........................................... 49
5.7.1 Actuator .................................................... 50
5.7.2 Door ......................................................... 50
5.7.3 Sensor ....................................................... 50

6 Composition Type Structural Elements ........................................ 51

7 Data Entities for Device ........................................ 55
7.1 Datatypes ..................................................... 56
7.2 Datatitem .................................................... 56
7.2.1 XML Schema Structure for DataItem .................. 56
7.2.2 Attributes for Datatitem ............................... 58
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Example Device Structural Elements</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Example Composition Structural Elements</td>
<td>17</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Component Diagram</td>
<td>20</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Description of Component Diagram</td>
<td>25</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Component Configuration Diagram</td>
<td>27</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Composition Diagram</td>
<td>31</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Description of Composition Diagram</td>
<td>33</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Reference Diagram</td>
<td>36</td>
</tr>
<tr>
<td>Figure 9</td>
<td>ComponentRef Diagram</td>
<td>36</td>
</tr>
<tr>
<td>Figure 10</td>
<td>DataItemRef Diagram</td>
<td>37</td>
</tr>
<tr>
<td>Figure 11</td>
<td>Example Data Entities for Device (DataItem)</td>
<td>55</td>
</tr>
<tr>
<td>Figure 12</td>
<td>DataItem Diagram</td>
<td>57</td>
</tr>
<tr>
<td>Figure 13</td>
<td>Source Diagram</td>
<td>77</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Constraints Diagram</td>
<td>79</td>
</tr>
<tr>
<td>Figure 15</td>
<td>Filter Diagram</td>
<td>82</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Definition Schema Diagram</td>
<td>86</td>
</tr>
<tr>
<td>Figure 17</td>
<td>Configuration Element</td>
<td>144</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Sensor Data Associations</td>
<td>146</td>
</tr>
<tr>
<td>Figure 19</td>
<td>SensorConfiguration Diagram</td>
<td>150</td>
</tr>
<tr>
<td>Figure 20</td>
<td>Relationship Diagram</td>
<td>157</td>
</tr>
<tr>
<td>Figure 21</td>
<td>DeviceRelationship Diagram</td>
<td>159</td>
</tr>
<tr>
<td>Figure 22</td>
<td>ComponentRelationship Diagram</td>
<td>163</td>
</tr>
<tr>
<td>Figure 23</td>
<td>Specifications Diagram</td>
<td>166</td>
</tr>
<tr>
<td>Figure 24</td>
<td>CoordinateSystems Diagram</td>
<td>168</td>
</tr>
</tbody>
</table>
List of Tables

Table 1: MTConnect Devices Element ........................................... 17
Table 2: MTConnect Components Element ..................................... 18
Table 3: MTConnect Component Element ....................................... 19
Table 4: Attributes for Component .............................................. 21
Table 5: Elements for Component ................................................ 24
Table 6: Attributes for Description for Component ......................... 25
Table 7: MTConnect Configuration Element for Component ............... 26
Table 8: MTConnect Compositions Element ................................... 29
Table 9: MTConnect Composition Element .................................... 30
Table 10: Attributes for Composition .......................................... 31
Table 11: Elements for Composition ............................................ 33
Table 12: Attributes for Description for Composition ...................... 34
Table 13: MTConnect References Element .................................... 35
Table 14: Attributes for ComponentRef ......................................... 37
Table 15: Attributes for DataItemRef ............................................ 38
Table 16: Top Level Component Elements ..................................... 39
Table 17: Composition type Elements .......................................... 51
Table 18: MTConnect DataItems Element ..................................... 56
Table 19: MTConnect DataItem Element ....................................... 56
Table 20: Attributes for DataItem ................................................ 58
Table 21: DataItem attribute statistic type .................................... 65
Table 22: DataItem attribute units type ........................................ 66
Table 23: DataItem attribute nativeunits type ................................. 68
Table 24: DataItem attribute coordinateSystem type ......................... 71
Table 25: DataItem attribute representation type ............................ 72
Table 26: Elements for DataItem .................................................. 75
Table 27: Attributes for Source ................................................... 77
Table 28: Elements for Constraints ............................................. 80
Table 29: MTConnect Filters Element ......................................... 81
Table 30: DataItem Element Filter type ....................................... 82
Table 31: MTConnect ResetTrigger Element .................................. 84
Table 32: DataItem Element ResetTrigger type ............................... 84
Table 33: Elements for Definition ............................................... 87
Table 34: Elements for EntryDefinitions ...................................... 87
Table 35: Attributes for EntryDefinition ....................................... 88
Table 36: Elements for EntryDefinition ....................................... 88
Table 37: Elements for CellDefinitions ........................................ 89
Table 38: Attributes for CellDefinition ........................................ 89
Table 39: Elements for CellDefinition ......................................... 89
Table 40: DataItem type subType for category SAMPLE ................... 91
1 Purpose of This Document

This document, MTConnect Standard: Part 2.0 - Devices Information Model of the MTConnect Standard, establishes the rules and terminology to be used by designers to describe the function and operation of a piece of equipment and to define the data that is provided by an Agent from the equipment. The Devices Information Model also defines the structure for the XML document that is returned from an Agent in response to a Probe Request.

In the MTConnect Standard, equipment represents any tangible property that is used in the operations of a manufacturing facility. Examples of equipment are machine tools, ovens, sensor units, workstations, software applications, and bar feeders.

Note: See MTConnect Standard: Part 3.0 - Streams Information Model of the MTConnect Standard for details on the XML documents that are returned from an Agent in response to a Sample Request or Current Request.
2 Terminology and Conventions

Refer to Section 3 of MTConnect Standard Part 1.0 - Overview and Fundamentals for a dictionary of terms, reserved language, and document conventions used in the MTConnect Standard.

2.1 Glossary

CDATA

General meaning:

An abbreviation for Character Data.

CDATA is used to describe a value (text or data) published as part of an XML element.

For example, "This is some text" is the CDATA in the XML element:

<Message ...>This is some text</Message>

Appears in the documents in the following form: CDATA

HTTP

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

Note: HTTP is an IETF standard and is defined in RFC 7230.

NMTOKEN

The data type for XML identifiers.

Note: The identifier must start with a letter, an underscore "_" or a colon. The next character must be a letter, a number, or one of the following ".", ",", ",", ",":. The identifier must not have any spaces or special characters.

 Appears in the documents in the following form: NMTOKEN

XML

Stands for eXtensible Markup Language.

XML defines a set of rules for encoding documents that both a human-readable and machine-readable.

XML is the language used for all code examples in the MTConnect Standard.

Refer to http://www.w3.org/XML for more information about XML.
Agent

Refers to an MTConnect Agent. Software that collects data published from one or more piece(s) of equipment, organizes that data in a structured manner, and responds to requests for data from client software systems by providing a structured response in the form of a [Response Document] that is constructed using the [semantic data models] defined in the Standard.

Appears in the documents in the following form: Agent

Asset

General meaning:

Typically referred to as an MTConnect Asset.

An MTConnect Asset is something that is used in the manufacturing process, but is not permanently associated with a single piece of equipment, can be removed from the piece of equipment without compromising its function, and can be associated with other pieces of equipment during its lifecycle.

Used to identify a storage area in an Agent.

See description of [buffer].

Used as an [Information Model]

Used to describe an [Information Model] that contains the rules and terminology that describe information that may be included in electronic documents representing MTConnect Assets.

The Asset Information Models defines the structure for the Assets Response Document.

Individual [Information Models] describe the structure of the [Asset Documents] represent each type of MTConnect Asset. Appears in the documents in the following form: Asset Information Models or (asset type) [Information Model]

Used when referring to an MTConnect Asset:

Refers to the information related to an MTConnect Asset or a group of MTConnect Assets.

Appears in the documents in the following form: Asset or Assets

Used as an XML container or element:

- When used as an XML container that consists of one or more types of Asset XML elements.

Appears in the documents in the following form: Assets.
• When used as an abstract XML element. It is replaced in the XML document by types of Asset elements representing individual Asset entities.

Appears in the documents in the following form: Asset.

Used to describe information stored in an Agent.

Identifies an electronic document published by a data source and stored in the assets buffer of an Agent.

Appears in the documents in the following form: Asset Document.

Used as an XML representation of an MTConnect Response Document:

Identifies an electronic document encoded in XML and published by an Agent in response to a Request for information from a client software application relating to MTConnect Assets.

Appears in the documents in the following form: MTConnectAssets.

Used as an MTConnect Request.

Represents a specific type of communications request between a client software application and an Agent regarding MTConnect Assets.

Appears in the documents in the following form: Asset Request.

Used as part of an HTTP Request.

Used in the path portion of an HTTP Request Line, by a client software application, to initiate an Asset Request to an Agent to publish an MTConnectAssets document.

Appears in the documents in the following form: asset.

Asset Document

An electronic document published by an Agent in response to a Request for information from a client software application relating to Assets.

buffer

General meaning:

A section of an Agent that provides storage for information published from pieces of equipment.

Used relative to Streaming Data:

A section of an Agent that provides storage for information relating to individual pieces of Streaming Data.

Appears in the documents in the following form: buffer.

Used relative to MTConnect Assets:
A section of an Agent that provides storage for Asset Documents. Appears in the documents in the following form: assets buffer.

**Child Element**

A portion of a data modeling structure that illustrates the relationship between an element and the higher-level Parent Element within which it is contained. Appears in the documents in the following form: Child Element.

**Controlled Vocabulary**

A restricted set of values that may be published as the Valid Data Value for a Data Entity. Appears in the documents in the following form: Controlled Vocabulary.

**Current Request**

An HTTP request to the Agent for returning latest known values for the Data Item as an MTConnectStreams XML document.

**Data Entity**

A primary data modeling element that represents all elements that either describe data items that may be reported by an Agent or the data items that contain the actual data published by an Agent. Appears in the documents in the following form: Data Entity.

**Data Set**

A set of key-value pairs where each entry is uniquely identified by the key.

**Devices Information Model**

A set of rules and terms that describes the physical and logical configuration for a piece of equipment and the data that may be reported by that equipment. Appears in the documents in the following form: Devices Information Model.

**Document**

General meaning:

A piece of written, printed, or electronic matter that provides information. Used to represent an MTConnect Document. Refers to printed or electronic document(s) that represent a Part(s) of the MTConnect Standard. Appears in the documents in the following form: MTConnect Document.
Used to represent a specific representation of an MTConnect Document. Refers to electronic document(s) associated with an Agent that are encoded using XML: [Response Documents] or [Asset Documents]. Appears in the documents in the following form: MTConnect XML Document.

Used to describe types of information stored in an Agent. In an implementation, the electronic documents that are published from a data source and stored by an Agent. Appears in the documents in the following form: Asset Document.

Used to describe information published by an Agent. A document published by an Agent based upon one of the semantic data models defined in the MTConnect Standard in response to a request from a client. Appears in the documents in the following form: Response Document.

engineering units

A quantity, dimension, or magnitude used in engineering adopted as a standard in terms of which the magnitude of other quantities of the same kind can be expressed or calculated.

Equipment Metadata

See Metadata

HTTP Request

In the MTConnect Standard, a communications command issued by a client software application to an Agent requesting information defined in the HTTP Request Line. Appears in the documents in the following form: HTTP Request.

HTTP Request Line

In the MTConnect Standard, the first line of an HTTP Request describing a specific Response Document to be published by an Agent. Appears in the documents in the following form: HTTP Request Line.

Information Model

The rules, relationships, and terminology that are used to define how information is structured.

For example, an information model is used to define the structure for each MTConnect Response Document; the definition of each piece of information within those documents and the relationship between pieces of information. Appears in the documents in the following form: Information Model.
Interaction Model
The definition of information exchanged to support the interactions between pieces of equipment collaborating to complete a task.
Appears in the documents in the following form: **Interaction Model**

Interface
**General meaning:**
The exchange of information between pieces of equipment and/or software systems.
Appears in the documents in the following form: **interface**.
**Used as an Interaction Model:**
An **Interaction Model** that describes a method for inter-operations between pieces of equipment.
Appears in the documents in the following form: **Interface**.
**Used as an XML container or element:**
- When used as an XML container that consists of one or more types of **Interface** XML elements.
Appears in the documents in the following form: **Interfaces**.
- When used as an abstract XML element. It is replaced in the XML document by types of **Interface** elements.
Appears in the documents in the following form: **Interface**

key
A unique identifier in a **key-value pair** association.

key-value pair
An association between an identifier referred to as the **key** and a value which taken together create a **key-value pair**. When used in a set of **key-value pairs**, each **key** is unique and will only have one value associated with it at any point in time.

Lower Level
A nested element that is below a higher level element.

Metadata
Data that provides information about other data.
For example, **Equipment Metadata** defines both the **Structural Elements** that represent the physical and logical parts and sub-parts of each piece of equipment, the
relationships between those parts and sub-parts, and the definitions of the Data Entities associated with that piece of equipment.

Appears in the documents in the following form: Metadata or Equipment Metadata.

**MTConnect Document**

See Document.

**MTConnect Request**

A communication request for information issued from a client software application to an Agent.

Appears in the documents in the following form: MTConnect Request.

**MTConnect XML Document**

See Document.

**observation**

The observed value of a property at a point in time.

**organize**

The act of containing and owning one or more elements.

**Parent Element**

An XML element used to organize Lower Level child elements that share a common relationship to the Parent Element.

Appears in the documents in the following form: Parent Element.

**Request**

A communications method where a client software application transmits a message to an Agent. That message instructs the Agent to respond with specific information.

Appears in the documents in the following form: Request.

**Response Document**

See Document.

**Sample Request**

A request from the Agent for a stream of time series data.
**semantic data model**

A methodology for defining the structure and meaning for data in a specific logical way.

It provides the rules for encoding electronic information such that it can be interpreted by a software system.

Appears in the documents in the following form: **semantic data model**

**Streaming Data**

The values published by a piece of equipment for the **Data Entities** defined by the **Equipment Metadata**.

Appears in the documents in the following form: **Streaming Data**

**Streams Information Model**

The rules and terminology (**semantic data model**) that describes the **Streaming Data** returned by an **Agent** from a piece of equipment in response to a **Sample Request** or a **Current Request**.

Appears in the documents in the following form: **Streams Information Model**

**Structural Element**

General meaning:

An XML element that organizes information that represents the physical and logical parts and sub-parts of a piece of equipment.

Appears in the documents in the following form: **Structural Element**

Used to indicate hierarchy of Components:

When used to describe a primary physical or logical construct within a piece of equipment.

Appears in the documents in the following form: **Top Level Structural Element**

When used to indicate a **Child Element** which provides additional detail describing the physical or logical structure of a **Top Level Structural Element**.

Appears in the documents in the following form: **Lower Level Structural Element**

**Table**

A two dimensional set of values given by a set of **key-value pairs** **Table Entries**. Each **Table Entry** contains a set of **key-value pairs** of **Table Cells**. The Entry and Cell elements comprise a tabular representation of the information.

**Table Cell**

A subdivision of a **Table Entry** representing a singular value.
Table Entry

A subdivision of a Table containing a set of key-value pairs representing Table Cells.

Top Level

Structural Elements that represent the most significant physical or logical functions of a piece of equipment.

Valid Data Value

One or more acceptable values or constrained values that can be reported for a Data Entity. Appears in the documents in the following form: Valid Data Value(s).

XML Schema

In the MTConnect Standard, an instantiation of a schema defining a specific document encoded in XML.

2.2 Acronyms

AMT

The Association for Manufacturing Technology

2.3 MTConnect References


[MTConnect Part 5.0] MTConnect Standard: Part 5.0 - Interfaces. Version 1.5.0.
The Devices Information Model provides a representation of the physical and logical configuration for a piece of equipment used for a manufacturing process or for any other purpose. It also provides the definition of data that may be reported by that equipment.

Using information defined in the Devices Information Model, a software application can determine the configuration and reporting capabilities of a piece of equipment. To do this, the software application issues a Probe Request (defined in MTConnect Standard Part 1.0 - Overview and Fundamentals Section 8.1.1) to an Agent associated with a piece of equipment. An Agent responds to the Probe Request with an MTConnectDevices XML document that contains information describing both the physical and logical structure of the piece of equipment and a detailed description of each Data Entity that can be reported by the Agent associated with the piece of equipment. This information allows the client software application to interpret the document and to extract the data with the same meaning, value, and context that it had at its original source.

The MTConnectDevices XML document is comprised of two sections: Header and Devices.

The Header section contains protocol related information as defined in MTConnect Standard Part 1.0 - Overview and Fundamentals Section 6.5.1.

The Devices section of the MTConnectDevices document contains a Device XML container for each piece of equipment described in the document. Each Device container is comprised of two primary types of XML elements - Structural Elements and Data Entities.

Structural Elements are defined as XML elements that organize information that represents the physical and logical parts and sub-parts of a piece of equipment (See Section 4 - Structural Elements for MTConnectDevices for more details).

Data Entities are defined as XML elements that describe data that can be reported by a piece of equipment. In the Devices Information Model, Data Entities are defined as DataItem elements (See Section 7 - Data Entities for Device and Section 8 - Listing of Data Items).

The Structural Elements and Data Entities in the MTConnectDevices document provide information representing the physical and logical structure for a piece of equipment and the types of data that the piece of equipment can report relative to that structure. The MTConnectDevices document does not contain values for the data types reported by the piece of equipment. The MTConnectStreams document defined in MTConnect
Standard: Part 3.0 - Streams Information Model provides the data values that are reported by the piece of equipment. As such, most Structural Elements and Data Entities in the MTConnectDevices document do not contain CDATA XML elements that provide values or information in the CDATA will be specifically identified in Section 4 - Structural Elements for MTConnectDevices, Section 7 - Data Entities for Device, and Section 9.1 - Sensor.

Note: The MTConnect Standard also defines the information model for Assets. An Asset is something that is used in the manufacturing process, but is not permanently associated with a single piece of equipment, can be removed from the piece of equipment without compromising its function, and can be associated with other pieces of equipment during its lifecycle. See MTConnect Standard: Part 4.0 - Assets Information Model for more details on Assets.
4 Structural Elements for MTConnectDevices

Structural Elements are XML elements that form the logical structure for the MTConnectDevices XML document. These elements are used to organize information that represents the physical and logical architecture of a piece of equipment. Refer to Figure 1 for an overview of the Structural Elements used in an MTConnectDevices document.

A variety of Structural Elements are defined to describe a piece of equipment. Some of these elements MUST always appear in the MTConnectDevices XML document, while others are optional and MAY be used, as required, to provide additional structure.

The first, or highest level, Structural Element in a MTConnectDevices XML document is Devices. Devices is a container type XML element used to group one or more pieces of equipment into a single XML document. Devices MUST always appear in the MTConnectDevices document.

Device is the next Structural Element in the MTConnectDevices XML document. Device is also a container type XML element. A separate Device container is used to identify each piece of equipment represented in the MTConnectDevices document. Each Device container provides information on the physical and logical structure of the piece of equipment and the data associated with that equipment. Device can also represent any logical grouping of pieces of equipment that function as a unit or any other data source that provides data through an Agent.

One or more Device element(s) MUST always appear in an MTConnectDevices document.

Components is the next Structural Element in the MTConnectDevices XML document. Components is also a container type XML element. Components is used to group information describing Lower Level physical parts or logical functions of a piece of equipment.

If the Components container appears in the XML document, it MUST contain one or more Component type XML elements.

Component is the next level of Structural Element in the MTConnectDevices XML document. Component is both an abstract type XML element and a container type element.

As an abstract type element, Component will never appear in the XML document describing a piece of equipment and will be replaced by a specific Component type defined in Section 5 - Component Structural Elements. Each Component type is also a container type element. As a container, the Component type element is used to organize infor-
If **Lower Level** Structural Elements are described, these elements are by definition child Component elements of a parent Component. At this next level, the **Lower Level** child Component elements are grouped into an XML container called Components.

This **Lower Level** Components container is comprised of one or more child Component elements representing the sub-parts of the parent Component. Just like the parent Component element, the child Component element is an abstract type XML element and will never appear in the XML document – only the different **Lower Level** child Component types will appear.

This parent-child relationship can continue to any depth required to fully define a piece of equipment.

**Example 1** illustrates the relationship between a parent Component and **Lower Level** child components:

```
1 <Devices>
2  <Device>
3   <Components>
4    <Axes>  Parent Component
5    <Components>
6     <Rotary>  Child component of Axes and Parent component of **Lower Level** components
7    <Components>
8     <Chuck>  Child Component of Rotary
```

**Figure 1** demonstrates the various Structural Elements provided to describe a piece of equipment and the relationship between these elements.
Component type XML elements MAY be further decomposed into Composition type XML elements. Composition elements describe the lowest level basic structural or functional building blocks contained within a Component. Any number of Composition elements MAY be used. Data provided for a Component provides more specific meaning when it is associated with one of the Composition elements of the Component. The different Composition types that MAY appear in the XML document are defined in Section 6 - Composition Type Structural Elements.

The Composition elements are organized into a Compositions container. The Compositions container MAY appear in the XML document further describing a Component. If one or more Composition element(s) is provided to describe a Component, a Compositions container MUST be defined for the Component.

Example 2 represents an XML document structure that demonstrates the relationship between a parent Component and its Composition elements.

Example 2: Component levels with Composition

Figure 1: Example Device Structural Elements

Figure 2 demonstrates this relationship between a Component and some of its potential Composition elements.
4.1 Devices

Devices MUST organize one or more Device elements.

Table 1: MTConnect Devices Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Devices</td>
<td>The first, or highest level, [Structural Element] in a MTConnectDevices document. Devices is a container type XML element.</td>
<td>1</td>
</tr>
</tbody>
</table>
4.2 Device

A Device is a Component that represents a piece of equipment that produces observations about itself. It organizes its parts as Components.

A Device MUST have a name and uuid attribute to identify itself.

A Device MUST have the following DataItems: AVAILABILITY, ASSET_CHANGED, and ASSET_REMOVED.

See Section 4.4 - Component for details on the Device model.

4.3 Components

Components is an XML container used to group information describing physical parts or logical functions of a piece of equipment. Components contains one or more Component XML elements.

Table 2: MTConnect Components Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Components</td>
<td>An XML container that consists of one or more types of Component XML elements.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>If a Components XML element is provided, then only one Components element MUST be defined for a Device element.</td>
<td></td>
</tr>
</tbody>
</table>

4.4 Component

A Component XML element is a container type XML element used to organize information describing a physical part or logical function of a piece of equipment. It also provides structure for describing the Lower Level Structural Elements associated with the Component. Component is an abstract type XML element and will never appear directly in the MTConnect XML document. As an abstract type XML element, Component will be replaced in the XML document by specific Component types. XML elements representing Component are described in Section 5 - Component Structural Elements and include elements such as Axes, Controller, and Systems.
Table 3: MTConnect Component Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component</td>
<td>An abstract XML element. Replaced in the XML document by types of Component elements representing physical parts and logical functions of a piece of equipment. There can be multiple types of Component XML elements in the document.</td>
<td>1..*</td>
</tr>
</tbody>
</table>

4.4.1 XML Schema Structure for Component

Figure 3 represents the structure of a Component XML element showing the attributes defined for Component and the elements that MAY be associated with Component.
Figure 3: Component Diagram
### 4.4.2 Attribute for Component

Table 4 defines the attributes that may be used to provide additional information for a Component type XML element.

**Table 4: Attributes for Component**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The unique identifier for this element. id is a required attribute.</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>An id MUST be unique across all the id attributes in the document.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>An XML ID-type.</td>
<td></td>
</tr>
<tr>
<td>nativeName</td>
<td>The common name normally associated with a specific physical or logical part of a piece of equipment. nativeName is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Occurrence</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>sampleInterval</td>
<td>An optional attribute that is an indication provided by a piece of equipment describing the interval in milliseconds between the completion of the reading of the data associated with the Component element until the beginning of the next sampling of that data. This indication is reported as the number of milliseconds between data captures.</td>
<td>0..1 ††</td>
</tr>
<tr>
<td></td>
<td>This information may be used by client software applications to understand how often information from a piece of equipment for a specific Component element is expected to be refreshed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The refresh rate for data from all Lower Level Component elements will be the same as for the parent Component element unless specifically overridden by another sampleInterval provided for the Lower Level Component element.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If the value of sampleInterval is less than one millisecond, the value will be represented as a floating-point number. For example, an interval of 100 microseconds would be 0.1.</td>
<td></td>
</tr>
<tr>
<td>sampleRate</td>
<td><strong>DEPRECATED</strong> in MTConnect Version 1.2. Replaced by sampleInterval.</td>
<td>0..1 †††</td>
</tr>
</tbody>
</table>
Continuation of Table 4

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>uuid</td>
<td>A unique identifier for this XML element. uuid is an optional attribute. The value provided for the uuid MUST be unique amongst all uuid identifiers used in an MTConnect installation. For example, this may be a combination of the manufacturer’s code and serial number. The uuid SHOULD be alphanumeric and not exceed 255 characters. An NMTOKEN XML type.</td>
<td>0..1 †</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Component element. name is an optional attribute. However, if there are multiple Lower Level components that have the same parent and are of the same component type (example Linear), then the name attribute MUST be provided for all Lower Level components of the same element type to differentiate between the similar components. When provided, name MUST be unique for all Lower Level components of a parent Component. An NMTOKEN XML type.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Notes: †While uuid MUST be provided for the Device element, it is optional for Component elements. ††The sampleInterval is used to aid a client software application in interpreting values provided by some Data Entities. This is the desired sample interval and may vary depending on the capabilities of the piece of equipment. †††Remains in schema for backwards compatibility.
4.4.3 Elements of Component

Table 5 lists the elements defined to provide additional information for a Component type XML element.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>An element that can contain any descriptive content.</td>
<td>0..1</td>
</tr>
<tr>
<td>Configuration</td>
<td>An XML element that contains technical information about a piece of equipment describing its physical layout or functional characteristics.</td>
<td>0..1</td>
</tr>
<tr>
<td>DataItems</td>
<td>A container for the Data Entities (defined in Section 8 - Listing of Data Items) associated with this Component element.</td>
<td>0..1 †</td>
</tr>
<tr>
<td>Components</td>
<td>A container for Lower Level Component XML elements associated with this parent Component.</td>
<td>0..1 †</td>
</tr>
<tr>
<td>Compositions</td>
<td>A container for the Composition elements (defined in Section 6 - Composition Type Structural Elements) associated with this Component element.</td>
<td>0..1</td>
</tr>
<tr>
<td>References</td>
<td>A container for the Reference elements associated with this Component element.</td>
<td>0..1 †</td>
</tr>
</tbody>
</table>

Note: †At least one of Components, DataItems, or References MUST be provided.

4.4.3.1 Description for Component

Figure 4 illustrates the structure of the Description XML element showing the attributes defined for Description. Description can contain any descriptive content of this Component. This element is defined to contain mixed content and additional XML elements (indicated by the any element) MAY be added to extend the schema for Description.
Table 6 lists the attributes defined for the Description XML element.

**Table 6: Attributes for Description for Component**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>manufacturer</td>
<td>The name of the manufacturer of the physical or logical part of a piece of equipment represented by the Component element. manufacturer is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>model</td>
<td>The model description of the physical part or logical function of a piece of equipment represented by the Component element. model is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>serialNumber</td>
<td>The serial number associated with the physical part or logical function of a piece of equipment represented by the Component element. serialNumber is an optional attribute.</td>
<td>0..1</td>
</tr>
</tbody>
</table>
Continuation of Table 6

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>station</td>
<td>The station where the physical part or logical function of a piece of equipment represented by the Component element is located when it is part of a manufacturing unit or cell with multiple stations.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

station is an optional attribute.

The content of Description MAY include any additional descriptive information the implementer chooses to include regarding the Component element. This content SHOULD be limited to information not included elsewhere in the MTConnectDevices XML document.

Example 3: Example of Description

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

4.4.3.2 Configuration for Component

The Configuration XML element contains technical information about a component. Configuration MAY include any information describing the physical layout or functional characteristics of a component, such as capabilities, testing, installation, operation, calibration, or maintenance. Configuration MAY also include information representing the inter-relationships between components within a piece of equipment.

Table 7: MTConnect Configuration Element for Component

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration</td>
<td>An XML element that contains technical information about a component describing its physical layout, functional characteristics, and relationships with other components within a piece of equipment.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Configuration data for Component is structured in the MTConnectDevices XML.
AbstractConfiguration is an abstract type XML element. It will never appear in the XML document representing a piece of equipment. When Configuration is provided for a component, that type of Configuration will appear in the XML document.

See Section 9 - Configuration for details on the types of Configuration.

![Component Configuration Diagram](image)

**Figure 5:** Component Configuration Diagram

### 4.4.3.3 DataItems for Component

DataItems is an XML container that provides structure for organizing the data reported by a piece of equipment that is associated with the Component.

See Section 7 - Data Entities for Device for details on the DataItems XML element.
4.4.3.4 Components within Component

The use of the XML container Components within a Component element provides the ability to further break down the structure of a Component element into even Lower Level physical and logical sub-parts. These Lower Level elements can add more clarity and granularity to the physical or logical structure of a piece of equipment and the data associated with that equipment.

This parent-child relationship can be extended down to any level necessary to fully describe a piece of equipment. These Lower Level Component elements use the same XML structure as Component defined in Section 4.4.1 - XML Schema Structure for Component.

Example 4: Example of parent Component and Child Elements

```
<Devices>
  <Device>
    <Components>
      <Axes> (Component)
      <Components>
        <Linear> (Component)
        <Components>
          <Etc. > (Component)
    </Components>
  </Components>
</Device>
</Devices>
```

4.4.3.5 Compositions for Component

Compositions is an XML container used to organize the lowest level structural building blocks contained within a Component as defined below.

4.4.3.6 References for Component

References is an XML container used to organize Reference elements associated with a Component element. See Section 4.7 - References for details on References.

4.5 Compositions

Compositions is an XML container that defines the lowest level structural building blocks contained within a Component element.

Compositions contains one or more Composition XML elements.
Table 8: MTConnect Compositions Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compositions</td>
<td>An XML container consisting of one or more types of Composition XML elements. Only one Compositions container MAY appear for a Component element.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

4.6 Composition

Composition XML elements are used to describe the lowest level physical building blocks of a piece of equipment contained within a Component.

Composition provides the ability to organize information describing parts of its parent Component. A Composition MUST NOT have child Components, Compositions, or DataItems elements.

Composition elements are used to add more clarity and granularity to the data being retrieved from a piece of equipment. The meaning of the data associated with a Component may be enhanced by designating a specific Composition element associated with that data.

An example of the additional detail provided when using Composition elements would be:

A TEMPERATURE associated with a Linear type axis may be further clarified by referencing the MOTOR or AMPLIFIER type Composition element associated with that axis, which differentiates the temperature of the motor from the temperature of the amplifier.

Composition is a typed XML element and will always define a specific type of structural building block contained within a Component. XML elements representing the types of Composition elements are described in Section 6 - Composition Type Structural Elements and include elements describing such basic building blocks as motors, amplifiers, filters, and pumps.

Example 5: Example of parent Component and child Composition elements

```xml
<Devices>
  <Device>
    <Components>
      <Axes> (Component)
    </Components>
  </Device>
</Devices>
```
### Table 9: MTConnect Composition Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Composition</td>
<td>An XML element used to describe the lowest level structural building blocks contained within a Component element. Composition is a typed XML element. There can be multiple types of Composition XML elements defined for a Component element.</td>
<td>1..*</td>
</tr>
</tbody>
</table>

#### 4.6.1 XML Schema Structure for Composition

*Figure 6* illustrates a Composition XML element showing the attributes defined for Composition and the elements that may be associated with Composition type XML elements.
4.6.2 Attributes for Composition

Table 10 defines the attributes that may be used to provide additional information for a Composition type XML element.

Table 10: Attributes for Composition

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The unique identifier for this element. *id is a required attribute. *An id MUST be unique across all the id attributes in the document. *An XML ID-type.</td>
<td>1</td>
</tr>
</tbody>
</table>
Continuation of Table 10

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>uuid</td>
<td>A unique identifier for this XML element. uuid is an optional attribute. The uuid MUST be unique amongst all uuid identifiers used in an MTConnect installation. For example, this may be a combination of the manufacturer’s code and serial number. The uuid SHOULD be alphanumeric and not exceed 255 characters. An NMTOKEN XML type.</td>
<td>0..1</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Composition element. If more than one Composition elements have the same type for the same Component, then the name attribute MUST be provided. Otherwise, the name attribute is optional. If provided, name MUST be unique within a Component element. name is an NMTOKEN XML type</td>
<td>0..1</td>
</tr>
<tr>
<td>type</td>
<td>The type of Composition element. type is a required attribute. Examples of types are MOTOR, FILTER, PUMP, and AMPLIFIER. Refer to Section 6.- Composition Type Structural Elements for a list of currently defined types.</td>
<td>1</td>
</tr>
</tbody>
</table>

4.6.3 Elements of Composition

Table 11 lists the elements defined to provide additional information for a Composition type XML element.
Table 11: Elements for Composition

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>An element that can contain any descriptive content.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

4.6.3.1 Description for Composition

Figure 7 represents the structure of the Description XML element showing the attributes defined for Description. Description can contain any descriptive content for this Composition element. This element is defined to contain mixed content and additional XML elements (indicated by the any element) MAY be added to extend the schema for Description.

![Composition Diagram](image)

Figure 7: Description of Composition Diagram

Table 12 lists the attributes defined for the Description XML element.
Table 12: Attributes for Description for Composition

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>manufacturer</td>
<td>The name of the manufacturer of the physical part of a piece of equipment represented by the Composition element. manufacturer is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>model</td>
<td>The model description of the physical part of a piece of equipment represented by the Composition element. model is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>serialNumber</td>
<td>The serial number associated with the physical part of a piece of equipment represented by the Composition element. serialNumber is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>station</td>
<td>The station where the physical part of a piece of equipment represented by the Composition element is located when it is part of a manufacturing unit or cell with multiple stations. station is an optional attribute.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

566 The content of Description MAY include any additional descriptive information the implementer chooses to include regarding the Composition element. This content SHOULD be limited to information not included elsewhere in the MTConnectDevices XML document.

Example 6: Example of Description

```
<Description manufacturer="Example Co"
serialNumber="A124FFF" station="2"> Spindle motor associated with Path 2. </Description>
```

4.7 References

574 References is an XML container that organizes pointers to information defined elsewhere within the XML document for a piece of equipment.
References may be modeled as part of a Device, Component or Interface type Structural Element.

References contains one or more Reference XML elements.

**Table 13: MTConnect References Element**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>References</td>
<td>An XML container consisting of one or more types of Reference XML elements. Only one References container MUST appear for a Device, Component, or Interface element.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

4.8 Reference

Reference is a pointer to information that is associated with another Structural Element defined elsewhere in the XML document for a piece of equipment. That information may be data from the other element or the entire structure of that element.

Reference is an efficient method to associate information with an element without duplicating any of the data or structure. For example, a Bar Feeder System may make a request for the BarFeederInterface and receive all the relevant data for the interface and the associated spindle (Rotary element) that is referenced as part of the BarFeederInterface.

Reference is an abstract type XML element and will never appear directly in the MTConnect XML document. As an abstract type XML element, Reference will be replaced in the XML document by a specific Reference type. The current supported types of Reference are DataItemRef and ComponentRef XML elements.

Figure 8 represents the structure of the Reference XML element.
4.8.1 ComponentRef

ComponentRef XML element is a pointer to all of the information associated with another Structural Element defined elsewhere in the XML document for a piece of equipment. ComponentRef allows all of the information (Lower Level Components and all Data Entities) that is associated with the other Structural Element to be directly associated with this XML element.

Figure 9 represents the structure of a ComponentRef XML element showing the attributes defined for ComponentRef.
Table 14 lists the attributes defined for the ComponentRef element.

**Table 14: Attributes for ComponentRef**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>idRef</td>
<td>A pointer to the id attribute of the Component that contains the information to be associated with this XML element. idRef is a required attribute.</td>
<td>1</td>
</tr>
<tr>
<td>name</td>
<td>The optional name of the ComponentRef. Only informative. name is an NMTOKEN XML type.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### 4.8.2 DataItemRef

DataItemRef XML element is a pointer to a Data Entity associated with another Structural Element defined elsewhere in the XML document for a piece of equipment. DataItemRef allows the data associated with a data item defined in another Structural Element to be directly associated with this XML element.

*Figure 10* represents the structure of a DataItemRef XML element showing the attributes defined for DataItemRef.

*Figure 10: DataItemRef Diagram*

Table 15 lists the attributes defined for the DataItemRef element.
**Table 15: Attributes for DataItemRef**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>idRef</td>
<td>A pointer to the id attribute of the DataItem that contains the information to be associated with this XML element. idRef is a required attribute.</td>
<td>1</td>
</tr>
<tr>
<td>name</td>
<td>The optional name of the DataItemRef. Only informative. name is an NMTOKEN XML type.</td>
<td>0..1</td>
</tr>
</tbody>
</table>
5 Component Structural Elements

Component Structural Elements are XML containers used to represent physical parts or logical functions of a piece of equipment.

Component Structural Elements are defined into two major categories:

- **Top Level** Component elements are used to group the Structural Elements representing the most significant physical or logical functions of a piece of equipment. The Top Level Component elements provided in an MTConnectDevices document SHOULD be restricted to those defined in Table 16. However, these Top Level Component elements MAY also be used as Lower Level Component elements; as required.

- **Lower Level** Component elements are used to describe the sub-parts of the parent Component to provide more clarity and granularity to the physical or logical structure of the Top Level Component elements.

This section of the Devices Information Model provides guidance for the most common relationships between Top Level Component elements and Lower Level child components. However, all Component elements MAY be used in any configuration, as required, to fully describe a piece of equipment.

As described in Section 4 - Structural Elements for MTConnectDevices, Component is an abstract type Structural Element within the Devices Information Model and will never appear directly in the MTConnectDevices XML document. As abstract type XML elements, Component will be replaced in the XML document by a specific Component type.

Table 16 defines the Top Level Component elements available to describe a piece of equipment.

<table>
<thead>
<tr>
<th>Top Level Component Element ††</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axes</td>
<td>An XML container used to organize the Structural Elements of a piece of equipment that perform linear or rotational motion.</td>
</tr>
<tr>
<td>Controller</td>
<td>An XML container used to organize information about an intelligent or computational function within a piece of equipment.</td>
</tr>
</tbody>
</table>
### Continuation of Table 16

<table>
<thead>
<tr>
<th>Top Level Component Element ††</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systems</td>
<td>An XML container used to organize information for <strong>Lower Level</strong> elements representing the major sub-systems that are permanently integrated into a piece of equipment.</td>
</tr>
<tr>
<td>Auxiliaries</td>
<td>An XML container used to organize information for <strong>Lower Level</strong> elements representing functional sub-systems that provide supplementary or extended capabilities for a piece of equipment, but they are not required for the basic operation of the equipment.</td>
</tr>
<tr>
<td>Resources</td>
<td>An XML container used to organize information for <strong>Lower Level</strong> elements representing types of items, materials, and personnel that support the operation of a piece of equipment or work to be performed at a location. Resources also represents materials or other items consumed or transformed by a piece of equipment for production of parts or other types of goods.</td>
</tr>
<tr>
<td>Interfaces</td>
<td>An XML container that organizes information used to coordinate actions and activities between pieces of equipment that communicate information between each other.</td>
</tr>
</tbody>
</table>

**Note:** †† The following components have been relocated or redefined since they are not classified as restricted **Top Level** components:
- Power was **DEPRECATED** in MTConnect Version 1.1 and was replaced by the **Data Entity** called **AVAILABILITY**.
- Door has been redefined as a **Lower Level** component of a parent **Component** element or as a **Composition** element.
- Actuator, due to its uniqueness, has been redefined as a piece of equipment with the ability to be represented as a **Lower Level** component of a parent **Component** element or as a **Composition** element.
- Sensor, due to its uniqueness, has been redefined as a piece of equipment with the ability to be represented as a **Lower Level** component of a parent **Component** element (See Section 9.1 - Sensor for further detail).
- Stock has been redefined as a **Lower Level** component of the **Resources** **Top Level** Component element.
The common relationship between the Top Level Component elements and the Lower Level child Component elements are described below. It should be noted that as the MT-Connect Standard evolves, more Component types will be added to organize information for new types of equipment and/or new physical or logical sub-parts of equipment.

5.1 Axes

Axes is a top-level Component that organizes information representing linear or rotational motion for a piece of equipment. The Linear axis Component represents linear motion, and the Rotary axis Component represents rotational motion.

In robotics, the term Axis is synonymous with Joint. A Joint is the connection between two parts of the structure that move in relation to each other.

Linear and Rotary components MUST have a name attribute that MUST follow the conventions described below. Use the nativeName attribute for the manufacturer’s name of the axis if it differs from the assigned name.

MTConnect has two high-level classes for automation equipment as follows: (1) Equipment that controls cartesian coordinate axes and (2) Equipment that controls articulated axes. There are ambiguous cases where some machines exhibit both characteristics; when this occurs, the primary control system’s configuration determines the classification.

Examples of cartesian coordinate equipment are CNC Machine Tools, Coordinate measurement machines, as specified in ISO 841, and 3D Printers. Examples of articulated automation equipment are Robotic systems as specified in ISO 8373.

The following sections define the designation of names for the axes and additional guidance when selecting the correct scheme to use for a given piece of equipment.

5.1.1 Cartesian Coordinate Naming Conventions

A Three-Dimensional Cartesian Coordinate control system organizes its axes orthogonally relative to a machine coordinate system where the manufacturer of the equipment specifies the origin.

Axes name SHOULD comply with ISO 841, if possible.
5.1.1 Linear Motion

A piece of equipment MUST represent prismatic motion using a Linear axis Component and assign its name using the designations X, Y, and Z. A Linear axis name MUST append a monotonically increasing suffix when there are more than one parallel axes; for example, X2, X3, and X4.

5.1.2 Rotary Motion

MTConnect MUST assign the name to Rotary axes exhibiting rotary motion using A, B, and C. A Rotary axis name MUST append a monotonically increasing suffix when more than one Rotary axis rotates around the same Linear axis; for example, A2, A3, and A4.

5.1.2 Articulated Machine Control Systems

An articulated control system’s axes represent the connecting linkages between two adjacent rigid members of an assembly. The Linear axis represents prismatic motion, and the Rotary axis represents the rotational motion of the two related members. The control organizes the axes in a kinematic chain from the mounting surface (base) to the end-effector or tooling.

5.1.3 Articulated Machine Axis Names

The axes of articulated machines represent forward kinematic relationships between mechanical linkages. Each axis is a connection between linkages, also referred to as joints, and MUST be named using a J followed by a monotonically increasing number; for example, J1, J2, J3. The numbering starts at the base axis connected or closest to the mounting surface, J1, incrementing to the mechanical interface, Jn, where n is the number of the last axis. The chain forms a parent-child relationship with the parent being the axis closest to the base.

A machine having an axis with more than one child MUST number each branch using its numeric designation followed by a branch number and a monotonically increasing number. For example, if J2 has two children, the first child branch MUST be named J2.1.1 and the second child branch J2.2.1. A child of the first branch MUST be named J2.1.2, incrementing to J2.1.n, where J2.1.n is the number of the last axis in that branch.
5.1.4 Rotary Component

A Rotary axis represents rotation about a fixed axis.

5.1.5 Linear Component

A Linear axis represents prismatic motion along a fixed axis.

5.2 Controller

Controller is a Top Level container that organizes information for an intelligent part of a piece of equipment that monitors and calculates information to alter the operating conditions of the equipment. Typical types of controllers for a piece of equipment include CNC (Computer Numerical Control), PAC (Programmable Automation Control), IPC (Industrialized Computer), or IC (Imbedded Computer).

Controller is a component that organizes and provides information regarding the execution of a control program(s), the mode of operation of the piece of equipment, and fault information regarding the operation of the equipment.

Note: MTConnect Version 1.1.0 and later implementations SHOULD use a Lower Level Component element called Path to represent an individual tool path or other independent function within a Controller element. When the Controller element is capable of executing more than one simultaneous and independent programs, the implementation MUST specify a Lower Level Path element representing each of the independent functions of the Controller.

5.2.1 Path

Path is an XML container that represents the information for an independent operation or function within a Controller. For many types of equipment, Path represents a set of Axes, one or more Program elements, and the data associated with the motion of a control point as it moves through space. However, it MAY also represent any independent function within a Controller that has unique data associated with that function.

Path SHOULD provide an EXECUTION data item to define the operational state of the Controller component of the piece of equipment.
If the Controller is capable of performing more than one independent operation or function simultaneously, a separate Path component MUST be used to organize the data associated with each independent operation or function.

5.3 Systems

Systems is a Top Level XML container that provides structure for the information describing one or more Lower Level functional systems that perform as discrete operating modules of the equipment or provide utility type services to support the operation of the equipment. These systems are required for the piece of equipment to perform its intended function and are permanently integrated into the piece of equipment.

Since these systems operate as separate functional units, they are represented in the MT-ConnectDevices XML document as individual Lower Level Component elements of Systems based on the function or service provided.

5.3.1 Hydraulic System

Hydraulic is an XML container that represents the information for a system comprised of all the parts involved in moving and distributing pressurized liquid throughout the piece of equipment.

5.3.2 Pneumatic System

Pneumatic is an XML container that represents the information for a system comprised of all the parts involved in moving and distributing pressurized gas throughout the piece of equipment.

5.3.3 Coolant System

Coolant is an XML container that represents the information for a system comprised of all the parts involved in distribution and management of fluids that remove heat from a piece of equipment.
5.3.4 Lubrication System

Lubrication is an XML container that represents the information for a system comprised of all the parts involved in distribution and management of fluids used to lubricate portions of the piece of equipment.

5.3.5 Electric System

Electric is an XML container that represents the information for the main power supply for device piece of equipment and the distribution of that power throughout the equipment. The electric system will provide all the data with regard to electric current, voltage, frequency, etc. that applies to the piece of equipment as a functional unit. Data regarding electric power that is specific to a Component will be reported as Data Entities for that specific Component.

5.3.6 Enclosure System

Enclosure is an XML container that represents the information for a structure used to contain or isolate a piece of equipment or area. The Enclosure system may provide information regarding access to the internal components of a piece of equipment or the conditions within the enclosure. For example, Door may be defined as a Lower Level Component or Composition element of the Enclosure system.

5.3.7 Protective System

Protective is an XML container that represents the information for those functions that detect or prevent harm or damage to equipment or personnel. Protective does not include the information relating to the Enclosure system.

5.3.8 ProcessPower System

ProcessPower is an XML container that represents the information for a power source associated with a piece of equipment that supplies energy to the manufacturing process separate from the Electric system. For example, this could be the power source for an EDM machining process, an electroplating line, or a welding system.
5.3.9 Feeder System

Feeder is an XML container that represents the information for a system that manages the delivery of materials within a piece of equipment. For example, this could describe the wire delivery system for an EDM or welding process; conveying system or pump and valve system distributing material to a blending station; or a fuel delivery system feeding a furnace.

5.3.10 Dielectric System

Dielectric is an XML container that represents the information for a system that manages a chemical mixture used in a manufacturing process being performed at that piece of equipment. For example, this could describe the dielectric system for an EDM process or the chemical bath used in a plating process.

5.3.11 EndEffector System

EndEffector is an XML container that represents the information for those functions that form the last link segment of a piece of equipment. It is the part of a piece of equipment that interacts with the manufacturing process.

5.3.12 WorkEnvelope System

WorkEnvelope organizes information about the physical process execution space within a piece of equipment. The WorkEnvelope MAY provide information regarding the physical workspace and the conditions within that workspace.

5.4 Auxiliaries

Auxiliaries is a Top Level XML container that provides structure for the information describing one or more Lower Level functional systems that provide supplementary or additional capabilities for the operation of a piece of equipment. These systems extend the capabilities of a piece of equipment, but are not required for the equipment to function.

Since these systems operate as independent units or are only temporarily associated with a piece of equipment, they are represented in the MTConnectDevices XML document as
individual Component elements of Auxiliaries based on the function or service provided to the equipment.

5.4.1 Loader System

Loader is an XML container that represents the information for a unit comprised of all the parts involved in moving and distributing materials, parts, tooling, and other items to or from a piece of equipment.

5.4.2 WasteDisposal System

WasteDisposal is an XML container that represents the information for a unit comprised of all the parts involved in removing manufacturing byproducts from a piece of equipment.

5.4.3 ToolingDelivery System

ToolingDelivery is an XML container that represents the information for a unit involved in managing, positioning, storing, and delivering tooling within a piece of equipment.

5.4.4 BarFeeder System

BarFeeder is an XML container that represents the information for a unit involved in delivering bar stock to a piece of equipment.

5.4.5 Environmental System

Environmental is an XML container that represents the information for a unit or function involved in monitoring, managing, or conditioning the environment around or within a piece of equipment.
5.4.6 Sensor System

Sensor is an XML container that represents the information for a piece of equipment that responds to a physical stimulus and transmits a resulting impulse or value from a sensing unit. When modeled as a component of Auxiliaries, sensor SHOULD represent an integrated sensor unit system that provides signal processing, conversion, and communications. A sensor unit may have multiple sensing elements, each representing the data for a variety of measured values. See Section 9.1.2 - Sensor Unit for more details on sensor unit.

Note: If modeling an individual sensor, then sensor should be associated with the component that the measured value is most closely associated. See Section 5.7.3 - Sensor.

5.4.7 Deposition System

Deposition is an XML container that represents the information for a system that manages the addition of material or state change of material being performed in an additive manufacturing process. For example, this could describe the portion of a piece of equipment that manages a material extrusion process or a vat polymerization process.

5.5 Resources

Resources is a Top Level XML container that groups items that support the operation of a piece of equipment. Resources also represents materials or other items consumed, transformed, or used for production of parts, materials, or other types of goods by a piece of equipment.

5.5.1 Materials

Materials is an XML container that provides information about materials or other items consumed or used by the piece of equipment for production of parts, materials, or other types of goods. Materials also represents parts or part stock that are present at a piece of equipment or location to which work is applied to transform the part or stock material into a more finished state.
5.5.1.1 Stock

Stock is an XML container that represents the information for the material that is used in a manufacturing process and to which work is applied in a machine or piece of equipment to produce parts.

Stock may be either a continuous piece of material from which multiple parts may be produced or it may be a discrete piece of material that will be made into a part or a set of parts.

5.5.2 Personnel

Personnel is an XML container that provides information about an individual or individuals who either control, support, or otherwise interface with a piece of equipment.

5.6 Interfaces

Interfaces is a Top Level XML Structural Element in the MTConnectDevices XML document. Interfaces organizes the information provided by a piece of equipment used to coordinate activities with other pieces of equipment. As such, Interfaces represents the inter-device communication information between a piece of equipment and other pieces of equipment.

See MTConnect Standard: Part 5.0 - Interfaces for detailed information on Interfaces.

5.7 Other Components

While most component elements SHOULD be modeled in a specific manner, there are some types of component elements that are used ubiquitously in equipment and MAY be associated with any number of different types of parent component elements.

These components MAY be modeled as Lower Level components of the Parent Element.
5.7.1 Actuator

Actuator is an XML container that represents the information for an apparatus for moving or controlling a mechanism or system. It takes energy usually provided by air, electric current, or liquid and converts the energy into some kind of motion.

5.7.2 Door

Door is an XML container that represents the information for a mechanical mechanism or closure that can cover, for example, a physical access portal into a piece of equipment. The closure can be opened or closed to allow or restrict access to other parts of the equipment.

When Door is represented as a Component, it MUST have a data item called DOOR_STATE to indicate if the door is OPEN, CLOSED, or UNLATCHED. A Component MAY contain multiple Door components.

5.7.3 Sensor

Sensor is an XML container that represents the information for a piece of equipment that responds to a physical stimulus and transmits a resulting impulse or value. If modeling individual sensors, then sensor should be associated with the component that the measured value is most closely associated.

See Section 9.1 - Sensor for more details on the use of Sensor.
6 Composition Type Structural Elements

Composition Structural Elements are used to describe the lowest level physical building blocks of a piece of equipment contained within a Component. By referencing a specific Composition element, further clarification and meaning to data associated with a specific Component can be achieved.

Both Component and Composition elements are Lower Level child Component XML elements representing the sub-parts of the parent Component. However, there are distinct differences between Component and Composition type elements.

Component elements may be further defined with Lower Level Component elements and may have associated Data Entities.

Composition elements represent the lowest level physical part of a piece of equipment. They MUST NOT be further defined with Lower Level Component elements and they MUST NOT have Data Entities directly associated with them. They do provide additional information that can be used to enhance the specificity of Data Entities associated with the parent Component.

Table 17 defines Composition type elements that are currently available to describe sub-parts of a Component element.

Table 17: Composition type Elements

<table>
<thead>
<tr>
<th>Element Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTUATOR</td>
<td>A mechanism for moving or controlling a mechanical part of a piece of equipment. It takes energy usually provided by air, electric current, or liquid and converts the energy into some kind of motion.</td>
</tr>
<tr>
<td>AMPLIFIER</td>
<td>An electronic component or circuit for amplifying power, electric current, or voltage.</td>
</tr>
<tr>
<td>BALLSCREW</td>
<td>A mechanical structure for transforming rotary motion into linear motion.</td>
</tr>
<tr>
<td>BELT</td>
<td>An endless flexible band used to transmit motion for a piece of equipment or to convey materials and objects.</td>
</tr>
<tr>
<td>Element Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BRAKE</td>
<td>A mechanism for slowing or stopping a moving object by the absorption or transfer of the energy of momentum, usually by means of friction, electrical force, or magnetic force.</td>
</tr>
<tr>
<td>CHAIN</td>
<td>An interconnected series of objects that band together and are used to transmit motion for a piece of equipment or to convey materials and objects.</td>
</tr>
<tr>
<td>CHOPPER</td>
<td>A mechanism used to break material into smaller pieces.</td>
</tr>
<tr>
<td>CHUCK</td>
<td>A mechanism that holds a part, stock material, or any other item in place.</td>
</tr>
<tr>
<td>CHUTE</td>
<td>An inclined channel for conveying material.</td>
</tr>
<tr>
<td>CIRCUIT_BREAKER</td>
<td>A mechanism for interrupting an electric circuit.</td>
</tr>
<tr>
<td>CLAMP</td>
<td>A mechanism used to strengthen, support, or fasten objects in place.</td>
</tr>
<tr>
<td>COMPRESSOR</td>
<td>A pump or other mechanism for reducing volume and increasing pressure of gases in order to condense the gases to drive pneumatically powered pieces of equipment.</td>
</tr>
<tr>
<td>DOOR</td>
<td>A mechanical mechanism or closure that can cover a physical access portal into a piece of equipment allowing or restricting access to other parts of the equipment.</td>
</tr>
<tr>
<td>DRAIN</td>
<td>A mechanism that allows material to flow for the purpose of drainage from, for example, a vessel or tank.</td>
</tr>
<tr>
<td>ENCODER</td>
<td>A mechanism to measure position.</td>
</tr>
<tr>
<td>EXPOSURE_UNIT</td>
<td>A mechanism for emitting a type of radiation</td>
</tr>
<tr>
<td>EXTRUSION_UNIT</td>
<td>A mechanism for dispensing liquid or powered materials</td>
</tr>
<tr>
<td>FAN</td>
<td>Any mechanism for producing a current of air.</td>
</tr>
<tr>
<td>Element Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FILTER</td>
<td>Any substance or structure through which liquids or gases are passed to remove suspended impurities or to recover solids.</td>
</tr>
<tr>
<td>GALVANOMOTOR</td>
<td>An electromechanical actuator that produces deflection of a beam of light or energy in response to electric current through its coil in a magnetic field.</td>
</tr>
<tr>
<td>GRIPPER</td>
<td>A mechanism that holds a part, stock material, or any other item in place.</td>
</tr>
<tr>
<td>HOPPER</td>
<td>A chamber or bin in which materials are stored temporarily, typically being filled through the top and dispensed through the bottom.</td>
</tr>
<tr>
<td>LINEAR_POSITION_FEEDBACK</td>
<td>A mechanism that measures linear motion or position.</td>
</tr>
<tr>
<td></td>
<td><strong>DEPRECIATION WARNING</strong>: May be deprecated in the future. Recommend using ENCODER.</td>
</tr>
<tr>
<td>MOTOR</td>
<td>A mechanism that converts electrical, pneumatic, or hydraulic energy into mechanical energy.</td>
</tr>
<tr>
<td>OIL</td>
<td>A viscous liquid.</td>
</tr>
<tr>
<td>POWER_SUPPLY</td>
<td>A unit that provides power to electric mechanisms.</td>
</tr>
<tr>
<td>PULLEY</td>
<td>A mechanism or wheel that turns in a frame or block and serves to change the direction of or to transmit force.</td>
</tr>
<tr>
<td>PUMP</td>
<td>An apparatus raising, driving, exhausting, or compressing fluids or gases by means of a piston, plunger, or set of rotating vanes.</td>
</tr>
<tr>
<td>REEL</td>
<td>A rotary storage unit for material</td>
</tr>
<tr>
<td>SENSING_ELEMENT</td>
<td>A mechanism that provides a signal or measured value.</td>
</tr>
</tbody>
</table>
Continuation of Table 17

<table>
<thead>
<tr>
<th>Element Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPREADER</td>
<td>A mechanism for flattening or spreading materials</td>
</tr>
<tr>
<td>STORAGE_BATTERY</td>
<td>A component consisting of one or more cells, in which chemical energy is converted into electricity and used as a source of power.</td>
</tr>
<tr>
<td>SWITCH</td>
<td>A mechanism for turning on or off an electric current or for making or breaking a circuit.</td>
</tr>
<tr>
<td>TABLE</td>
<td>A surface for holding an object or material</td>
</tr>
<tr>
<td>TANK</td>
<td>A receptacle or container for holding material.</td>
</tr>
<tr>
<td>TENSIONER</td>
<td>A mechanism that provides or applies a stretch or strain to another mechanism.</td>
</tr>
<tr>
<td>TRANSFORMER</td>
<td>A mechanism that transforms electric energy from a source to a secondary circuit.</td>
</tr>
<tr>
<td>VALVE</td>
<td>Any mechanism for halting or controlling the flow of a liquid, gas, or other material through a passage, pipe, inlet, or outlet.</td>
</tr>
<tr>
<td>VAT</td>
<td>A container for liquid or powdered materials</td>
</tr>
<tr>
<td>WATER</td>
<td>A fluid.</td>
</tr>
<tr>
<td>WIRE</td>
<td>A string like piece or filament of relatively rigid or flexible material provided in a variety of diameters.</td>
</tr>
<tr>
<td>WORKPIECE</td>
<td>An object or material on which a form of work is performed.</td>
</tr>
</tbody>
</table>

Note: As the MTConnect Standard evolves, more composition types will be added.
7 Data Entities for Device

In the MTConnectDevices XML document, Data Entities are XML elements that describe data that can be reported by a piece of equipment and are associated with Device and Component Structural Elements. While the Data Entities describe the data that can be reported by a piece of equipment in the MTConnectDevices document, the actual data values are provided in the Streams Information Model. See MTConnect Standard: Part 3.0 - Streams Information Model for detail on the reported values.

Each Data Entity SHOULD be modeled in the MTConnectDevices document such that it is associated with the Structural Element that the reported data directly applies.

When Data Entities are associated with a Structural Element, they are organized in a DataItems XML element. DataItems is a container type XML element. DataItems provides the structure for organizing individual DataItem elements that represent each Data Entity. The DataItems container is comprised of one or more DataItem type XML element(s).

DataItem describes specific types of Data Entities that represent a numeric value, a functioning state, or a health status reported by a piece of equipment. DataItem provides a detailed description for each Data Entity that is reported; it defines the type of data being reported and an array of optional attributes that further describe that data. The different types of DataItem elements are defined in Section 8 - Listing of Data Items.

Figure 11 demonstrates the relationship between Data Entities (DataItem) and the various Structural Elements in the MTConnectDevices XML document.

Figure 11: Example Data Entities for Device (DataItem)
7.1 DataItems

The DataItems XML element is the first, or highest, level container for the Data Entities associated with a Device or Component XML element. DataItems MUST contain only DataItem type elements. DataItems MUST contain at least one DataItem type element, but MAY contain multiple DataItem type elements.

Table 18: MTConnect DataItems Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataItems</td>
<td>An XML container consisting of one or more types of DataItem XML elements. Only one DataItems container MUST appear for each Structural Element in the XML document.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

7.2 DataItem

A DataItem XML element represents each Data Entity that MAY be reported by a piece of equipment through an Agent. DataItem provides a detailed description for each Data Entity that is reported and defines the type of data being reported along with an array of optional attributes that further define that data. XML elements representing DataItem will include elements such as TEMPERATURE, PRESSURE, and VELOCITY.

Table 19: MTConnect DataItem Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataItem</td>
<td>Data Entity describing a piece of information reported about a piece of equipment.</td>
<td>1..*</td>
</tr>
</tbody>
</table>

7.2.1 XML Schema Structure for DataItem

Figure 12 represents the structure of a DataItem XML element showing the attributes defined for DataItem and the elements that may be associated with DataItem type XML elements.
Figure 12: DataItem Diagram
### 7.2.2 Attributes for DataItem

*Table* 20 lists the attributes defined to provide information for a `DataItem` type XML element.

DataItem **MUST** specify the type of data being reported, the id of the DataItem, and the category of the DataItem.

**Table 20: Attributes for DataItem**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The name of the data item. <code>name</code> is provided as an additional human readable identifier for this data item in addition to the <code>id</code>. <code>name</code> is an optional attribute and will be implementation dependent. An NMTOKEN XML type.</td>
<td>0..1</td>
</tr>
<tr>
<td>id</td>
<td>The unique identifier for this element. <code>id</code> is a required attribute. The <code>id</code> attribute <strong>MUST</strong> be unique within the <code>MTConnectDevices</code> document. An XML ID-type.</td>
<td>1</td>
</tr>
<tr>
<td>type</td>
<td>The type of data being measured. <code>type</code> is a required attribute. Examples of types are <code>POSITION</code>, <code>VELOCITY</code>, <code>ANGLE</code>, <code>BLOCK</code>, and <code>ROTARY_VELOCITY</code>.</td>
<td>1</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Occurrence</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>subType</td>
<td>A sub-categorization of the data item type. &lt;br&gt;<strong>subType</strong> is an optional attribute. &lt;br&gt;**For example, the subType of <strong>POSITION</strong> can be <strong>ACTUAL</strong> or <strong>COMMANDED</strong>. &lt;br&gt;Not all type attributes have a subType.</td>
<td>0..1</td>
</tr>
<tr>
<td>statistic</td>
<td>Describes the type of statistical calculation performed on a series of data samples to provide the reported data value. &lt;br&gt;<strong>statistic</strong> is an optional attribute. &lt;br&gt;<strong>Examples of statistic are</strong> <strong>AVERAGE</strong>, <strong>MINIMUM</strong>, <strong>MAXIMUM</strong>, <strong>ROOT_MEAN_SQUARE</strong>, <strong>RANGE</strong>, <strong>MEDIAN</strong>, <strong>MODE</strong>, <strong>and</strong> <strong>STANDARD_DEVIATION</strong>.</td>
<td>0..1</td>
</tr>
<tr>
<td>units</td>
<td>The unit of measurement for the reported value of the data item. &lt;br&gt;<strong>units</strong> is an optional attribute. &lt;br&gt;Data items in the Sample category <strong>MUST</strong> report the standard units for the measured values. &lt;br&gt;&lt;br&gt;<strong>See Section 7.2.2.5 - units Attribute for DataItem</strong> for a list of available standard units identified in the MTConnect Standard.</td>
<td>0..1</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Occurrence</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>nativeUnits</td>
<td>The native units of measurement for the reported value of the data item. nativeUnits is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>nativeUnits</td>
<td>See Section 7.2.2.6 - nativeUnits Attribute for DataItem for a list of available native units identified in the MTConnect Standard.</td>
<td></td>
</tr>
<tr>
<td>nativeScale</td>
<td>The nativeUnits may not be scaled to directly represent the original measured value. nativeScale MAY be used to convert the reported value to represent the original measured value.</td>
<td>0..1</td>
</tr>
<tr>
<td>nativeScale</td>
<td>nativeScale is an optional attribute.</td>
<td></td>
</tr>
<tr>
<td>nativeScale</td>
<td>As an example, the nativeUnits may be reported as GALLON/MINUTE. The measured value may actually be in 1000 GALLON/MINUTE. The value of the reported data MAY be divided by the nativeScale to convert the reported value to its original measured value and units.</td>
<td></td>
</tr>
<tr>
<td>nativeScale</td>
<td>If provided, the value MUST be numeric.</td>
<td></td>
</tr>
<tr>
<td>category</td>
<td>Specifies the kind of information provided by a data item. category is a required attribute. The available options are Sample, Event, or Condition.</td>
<td>1</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Occurrence</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>coordinateSystem</td>
<td>For measured values relative to a coordinate system like POSITION, the coordinate system being used may be reported. coordinateSystem is an optional attribute. The available values for coordinateSystem are WORK and MACHINE.</td>
<td>0..1</td>
</tr>
<tr>
<td>compositionId</td>
<td>The identifier attribute of the Composition element that the reported data is most closely associated. compositionId is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>sampleRate</td>
<td>The rate at which successive samples of a data item are recorded by a piece of equipment. sampleRate is an optional attribute. sampleRate is expressed in terms of samples per second. If the sampleRate 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</td>
<td>0..1</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Occurrence</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>representation</td>
<td>Description of a means to interpret data consisting of multiple data points or as a single value.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>representation is an optional attribute.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>representation defines the unique format for each set of data.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>representation for TIME_SERIES, DISCRETE (DEPRECATED in Version 1.5), DATA_SET, TABLE, and VALUE are defined in Section 7.2.2.12 - representation Attribute for DataItem</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If representation is not specified, it MUST be determined to be VALUE.</td>
<td></td>
</tr>
<tr>
<td>significantDigits</td>
<td>The number of significant digits in the reported value.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>significantDigits is an optional attribute.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>This SHOULD be specified for all numeric values.</td>
<td></td>
</tr>
</tbody>
</table>
### Continuation of Table 20

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
</table>
| discrete                | An indication signifying whether each value reported for the Data Entity is significant and whether duplicate values are to be suppressed.  

The value defined **MUST** be either **true** or **false** - an XML boolean type.  

**true** indicates that each update to the Data Entity's value is significant and duplicate values **MUST NOT** be suppressed.  

**false** indicates that duplicated values **MUST** be suppressed.  

If a value is not defined for `discrete`, the default value **MUST** be false. |
| coordinateSystemIdRef   | The associated CoordinateSystem context for the DataItem.                    | 0..1         |

#### 7.2.2.1 name Attribute for DataItem

The attribute `name` is provided as an additional human readable identifier for a data item. It is not required and is implementation dependent.

#### 7.2.2.2 id Attribute for DataItem

Each DataItem element **MUST** be identified with an `id`. The `id` attribute **MUST** be unique across the entire MTConnectDevices document for a piece of equipment, including the identifiers for all ` Structural Elements`. This unique `id` provides the information required by a client software application to uniquely identify each Data Entity.

For example, an XML document may provide three different Data Entities representing the position of the axes on a machine (x axis position, y axis position, and z axis position). All three may be modeled in the XML document as POSITION type data items for the Axes components. The unique `id` allows the client software application to distinguish the data for each of the axes.
7.2.2.3 type and subType Attributes for DataItem

The attribute type specifies the kind of data that is represented by the data item.

The attribute type MUST be specified for every data item.

A data item MAY further qualify the data being reported by specifying a subType. subType is required for certain data item types. For example, POSITION has the subType of ACTUAL and PROGRAMMED. Both data values can be represented in the document as two separate and different DataItem XML elements – POSITION with subType ACTUAL and POSITION with subType PROGRAMMED.

The type and subType SHOULD be used to further identify the meaning of the DataItem associated with a Component element when a subType is applicable. There SHOULD NOT be more than one DataItem with the same type, subType, and compositionId within a Component element.

Section 8 - Listing of Data Items provides a detailed listing of the data item type and subType elements defined for each category of data item available for a piece of equipment: SAMPLE, EVENT, and CONDITION.

7.2.2.4 statistic Attribute for DataItem

A piece of equipment may further process some data types using a statistical calculation like average, mean, or square root. In this case, the statistic attribute MAY be used to indicate how the data was processed.

statistic may be defined for any SAMPLE type DataItem. All statistic data is reported in the standard units of the DataItem.

statistic data is always the result of a calculation using data that has been measured over a specified period of time.

The value of statistic may be periodically reset. When a piece of equipment reports a DataItem with a value that is a statistic, the information provided in the XML document for that Data Entity MUST include an additional attribute called duration. The attribute duration defines the period of time over which the statistic has been calculated. See MTConnect Standard: Part 3.0 - Streams Information Model for more information about duration.

Table 21 shows the statistic calculations that can be defined for a DataItem.
Table 21: DataItem attribute statistic type

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVERAGE</td>
<td>Mathematical Average value calculated for the data item during the calculation period.</td>
</tr>
<tr>
<td>KURTOSIS</td>
<td><strong>DEPRECATED in Version 1.6.</strong> A measure of the &quot;peakedness&quot; of a probability distribution; i.e., the shape of the distribution curve.</td>
</tr>
<tr>
<td>MAXIMUM</td>
<td>Maximum or peak value recorded for the data item during the calculation period.</td>
</tr>
<tr>
<td>MEDIAN</td>
<td>The middle number of a series of numbers.</td>
</tr>
<tr>
<td>MINIMUM</td>
<td>Minimum value recorded for the data item during the calculation period.</td>
</tr>
<tr>
<td>MODE</td>
<td>The number in a series of numbers that occurs most often.</td>
</tr>
<tr>
<td>RANGE</td>
<td>Difference between the maximum and minimum value of a data item during the calculation period. Also represents Peak-to-Peak measurement in a waveform.</td>
</tr>
<tr>
<td>ROOT_MEAN_SQUARE</td>
<td>Mathematical Root Mean Square (RMS) value calculated for the data item during the calculation period.</td>
</tr>
<tr>
<td>STANDARD DEVIATION</td>
<td>Statistical Standard Deviation value calculated for the data item during the calculation period.</td>
</tr>
</tbody>
</table>
Table 22 lists the units that are defined as the standard unit of measure for each type of DataItem. All SAMPLE type data items MUST report data values in standard units.

Table 22: DataItem attribute units type

<table>
<thead>
<tr>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMPERE</td>
<td>Amps</td>
</tr>
<tr>
<td>CELSIUS</td>
<td>Degrees Celsius</td>
</tr>
<tr>
<td>COUNT</td>
<td>A count of something.</td>
</tr>
<tr>
<td>CUBIC_MILLIMETER</td>
<td>Geometric volume in millimeters</td>
</tr>
<tr>
<td>CUBIC_MILLIMETER/SECOND</td>
<td>Change of geometric volume per second</td>
</tr>
<tr>
<td>CUBIC_MILLIMETER/SECOND²</td>
<td>Change in geometric volume per second squared</td>
</tr>
<tr>
<td>DECIBEL</td>
<td>Sound Level</td>
</tr>
<tr>
<td>DEGREE</td>
<td>Angle in degrees</td>
</tr>
<tr>
<td>DEGREE/SECOND</td>
<td>Angular degrees per second</td>
</tr>
<tr>
<td>DEGREE/SECOND²</td>
<td>Angular acceleration in degrees per second squared</td>
</tr>
<tr>
<td>DEGREE_3D</td>
<td>A space-delimited, floating-point representation of the angular rotation in degrees around the X, Y, and Z axes relative to a cartesian coordinate system respectively in order as A, B, and C. If any of the rotations is not known, it MUST be zero (0).</td>
</tr>
<tr>
<td>GRAM/CUBIC_METER</td>
<td>Gram per cubic meter.</td>
</tr>
<tr>
<td>HERTZ</td>
<td>Frequency measured in cycles per second</td>
</tr>
<tr>
<td>JOULE</td>
<td>A measurement of energy.</td>
</tr>
<tr>
<td>KILOGRAM</td>
<td>Kilograms</td>
</tr>
<tr>
<td>LITER</td>
<td>Measurement of volume of a fluid</td>
</tr>
<tr>
<td>LITER/SECOND</td>
<td>Liters per second</td>
</tr>
<tr>
<td>MICRO_RADIAN</td>
<td>Measurement of Tilt</td>
</tr>
<tr>
<td>Units</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MILLIGRAM</td>
<td>Milligram</td>
</tr>
<tr>
<td>MILLIGRAM/CUBIC_MILLITER</td>
<td>Milligram per cubic millimeter</td>
</tr>
<tr>
<td>MILLILITER</td>
<td>Milliliter</td>
</tr>
<tr>
<td>MILLIMETER</td>
<td>Millimeters</td>
</tr>
<tr>
<td>MILLIMETER/REVOLUTION</td>
<td>Millimeters per revolution.</td>
</tr>
<tr>
<td>MILLIMETER/SECOND</td>
<td>Millimeters per second</td>
</tr>
<tr>
<td>MILLIMETER/SECOND²</td>
<td>Acceleration in millimeters per second squared</td>
</tr>
<tr>
<td>MILLIMETER_3D</td>
<td>A point in space identified by X, Y, and Z positions and represented by a space-delimited set of numbers each expressed in millimeters.</td>
</tr>
<tr>
<td>NEWTON</td>
<td>Force in Newtons</td>
</tr>
<tr>
<td>NEWTON_METER</td>
<td>Torque, a unit for force times distance.</td>
</tr>
<tr>
<td>OHM</td>
<td>Measure of Electrical Resistance</td>
</tr>
<tr>
<td>PASCAL</td>
<td>Pressure in Newtons per square meter.</td>
</tr>
<tr>
<td>PASCAL_SECON</td>
<td>Measurement of Viscosity</td>
</tr>
<tr>
<td>PERCENT</td>
<td>Percentage</td>
</tr>
<tr>
<td>PH</td>
<td>A measure of the acidity or alkalinity of a solution.</td>
</tr>
<tr>
<td>REVOLUTION/MINUTE</td>
<td>Revolutions per minute</td>
</tr>
<tr>
<td>REVOLUTION/SECOND</td>
<td>Revolutions per second.</td>
</tr>
<tr>
<td>REVOLUTION/SECOND²</td>
<td>Revolutions per second squared.</td>
</tr>
<tr>
<td>SECOND</td>
<td>A measurement of time.</td>
</tr>
<tr>
<td>SIEMENS/METER</td>
<td>A measurement of Electrical Conductivity</td>
</tr>
<tr>
<td>VOLT</td>
<td>Volts</td>
</tr>
<tr>
<td>VOLT_AMPERE</td>
<td>Volt-Ampere (VA)</td>
</tr>
<tr>
<td>VOLT_AMPERE_REACTIVE</td>
<td>Volt-Ampere Reactive (VAR)</td>
</tr>
<tr>
<td>WATT</td>
<td>Watts</td>
</tr>
</tbody>
</table>
Continuation of Table 22

<table>
<thead>
<tr>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WATT SECOND</td>
<td>Measurement of electrical energy, equal to one Joule</td>
</tr>
</tbody>
</table>

### 7.2.2.6 nativeUnits Attribute for DataItem

The `DataItem` **MAY** specify the *engineering units* used by the information source using the optional attribute `nativeUnits`. The `nativeUnits` are inclusive of the *engineering units* for the `units` attribute (See Table 22). One **MAY** use a prefixed value, for example `nativeUnits="x:MILE"`, to extend the [Controlled Vocabulary](#) with a namespace.

*MTConnect* specifies the following [Controlled Vocabulary](#) for `nativeUnits` in Table 23:

#### Table 23: DataItem attribute nativeunits type

<table>
<thead>
<tr>
<th>Native Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CENTIPOISE</td>
<td>A measure of Viscosity</td>
</tr>
<tr>
<td>DEGREE/ MINUTE</td>
<td>Rotational velocity in degrees per minute</td>
</tr>
<tr>
<td>FAHRENHEIT</td>
<td>Temperature in Fahrenheit</td>
</tr>
<tr>
<td>FOOT</td>
<td>Feet</td>
</tr>
<tr>
<td>FOOT/ MINUTE</td>
<td>Feet per minute</td>
</tr>
<tr>
<td>FOOT/ SECOND</td>
<td>Feet per second</td>
</tr>
<tr>
<td>FOOT/ SECOND$^2$</td>
<td>Acceleration in feet per second squared</td>
</tr>
<tr>
<td>FOOT_3D</td>
<td>A point in space identified by X, Y, and Z positions and represented by a space-delimited set of numbers each expressed in feet.</td>
</tr>
<tr>
<td>GALLON/ MINUTE</td>
<td>Gallons per minute.</td>
</tr>
<tr>
<td>HOUR</td>
<td>A measurement of time in hours</td>
</tr>
<tr>
<td>INCH</td>
<td>Inches</td>
</tr>
<tr>
<td>INCH/ MINUTE</td>
<td>Inches per minute</td>
</tr>
<tr>
<td>INCH/ SECOND</td>
<td>Inches per second</td>
</tr>
</tbody>
</table>
Continuation of Table 23

<table>
<thead>
<tr>
<th>Native Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INCH/SECOND²</td>
<td>Acceleration in inches per second squared</td>
</tr>
<tr>
<td>INCH_3D</td>
<td>A point in space identified by X, Y, and Z positions and represented by a space-delimited set of numbers each expressed in inches.</td>
</tr>
<tr>
<td>INCH_POUND</td>
<td>A measure of torque in inch pounds.</td>
</tr>
<tr>
<td>KELVIN</td>
<td>A measurement of temperature</td>
</tr>
<tr>
<td>KILOWATT</td>
<td>A measurement in kilowatt.</td>
</tr>
<tr>
<td>KILOWATT_HOUR</td>
<td>Kilowatt hours which is 3.6 mega joules.</td>
</tr>
<tr>
<td>LITER</td>
<td>Measurement of volume of a fluid</td>
</tr>
<tr>
<td>LITER/MINUTE</td>
<td>Measurement of rate of flow of a fluid</td>
</tr>
<tr>
<td>MILLIMETER/MINUTE</td>
<td>Velocity in millimeters per minute</td>
</tr>
<tr>
<td>MINUTE</td>
<td>A measurement of time in minutes</td>
</tr>
<tr>
<td>OTHER</td>
<td>Unsupported units</td>
</tr>
<tr>
<td>POUND</td>
<td>US pounds</td>
</tr>
<tr>
<td>POUND/INCH²</td>
<td>Pressure in pounds per square inch (PSI).</td>
</tr>
<tr>
<td>RADIANT</td>
<td>Angle in radians</td>
</tr>
<tr>
<td>RADIANT/MINUTE</td>
<td>Velocity in radians per minute.</td>
</tr>
<tr>
<td>RADIANT/SECOND</td>
<td>Rotational acceleration in radian per second squared</td>
</tr>
<tr>
<td>RADIANT/SECOND²</td>
<td>Rotational acceleration in radian per second squared</td>
</tr>
<tr>
<td>REVOLUTION/SECOND</td>
<td>Rotational velocity in revolution per second</td>
</tr>
</tbody>
</table>

7.2.2.7 nativeScale Attribute for DataItem

The units of measure for some measured values may be different from the nativeUnits defined in Section 7.2.2.8 - category Attribute for DataItem. In the cases where the units of measure use a different weighting or range than is provided by nativeUnits, the nativeScale attribute can be used to define the original units of measure.

As an example, a velocity measured in units of 100 ft/min can be represented as nativeUnits="FEET/MINUTE" and nativeScale="100".
### 7.2.2.8 category Attribute for DataItem

Many DataItem types provide two forms of data, a value (reported as either a SAMPLE or EVENT category) and a health status (reported as a CONDITION category). Therefore, each occurrence of a DataItem in the XML document MUST report a category attribute. This category attribute provides the information required by a client software application to determine the specific meaning of the data provided.

Each Data Entity provided by a piece of equipment MUST be identified with one of the following: SAMPLE, EVENT, CONDITION.

A SAMPLE is the reading of the value of a continuously variable or analog data value. A continuous value can be measured at any point-in-time and will always produce a result. An example of a continuous data value is the position of a linear axis called X.

The data provided for a SAMPLE category data item is always a floating point number or integers that have an infinite number of possible values. This is different from a state or discrete type data item that has a limited number of possible values. A data item of category SAMPLE MUST also provide the units attribute.

An EVENT is a data item representing a discrete piece of information from the piece of equipment. EVENT does not have intermediate values that vary over time, as does SAMPLE. An EVENT is information that, when provided at any specific point in time, represents the current state of the piece of equipment.

There are two types of EVENT: those representing state, with two or more discrete values, and those representing messages that contain plain text data.

An example of a state type EVENT is the value of the data item DOOR_STATE, which can be OPEN, CLOSED, or UNLATCHED. (Note: No other values are valid to represent the value of DOOR_STATE.)

An example of a message type EVENT is the value for a data item PROGRAM. The value representing PROGRAM can be any valid string of characters.

A CONDITION is a data item that communicates information about the health of a piece of equipment and its ability to function. A valid value for a data item in the category CONDITION can be one of Normal, Warning, or Fault.

A data item of category CONDITION MAY report multiple values (CONDITION) at one time whereas a data item of category SAMPLE or EVENT can only have a single value at any one point in time.
7.2.2.9 coordinateSystem Attribute for DataItem

The values reported by a piece of equipment for some types of data will be associated to a specific positioning measurement system used by the equipment. The coordinateSystem attribute MAY be used to specify the coordinate system used for the measured value.

The coordinateSystem attribute is used by a client software application to interpret the spatial relationship between values reported by a piece of equipment.

If coordinateSystem is not provided, all values representing positional data for Axes MUST be interpreted using the MACHINE coordinate system and all values representing positional data for Path MUST be interpreted using the WORK coordinate system.

Table 24 defines the types of coordinateSystem currently supported by the MTConnectDevices XML document:

<table>
<thead>
<tr>
<th>Coordinate System</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MACHINE</td>
<td>An unchangeable coordinate system that has machine zero as its origin.</td>
</tr>
<tr>
<td>WORK</td>
<td>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 piece of equipment, the MACHINE coordinates will be used.</td>
</tr>
</tbody>
</table>

7.2.2.10 compositionId Attribute for DataItem

compositionId attribute identifies the id of the Composition element where the reported data is most closely associated.

An example would be a TEMPERATURE associated with a Linear type axis may be further clarified by referencing the MOTOR or AMPLIFIER type Composition element associated with that axis, which differentiates the temperature of the motor from the temperature of the amplifier.
The compositionId attribute provides the information required by a client software application to interpret the data with a greater specificity and to disambiguate between multiple Data Entities of the same data type associated with a Component element.

7.2.2.11 sampleRate Attribute for DataItem

The value for some data types provided by a piece of equipment may be reported as a single set of data containing a series of values that have been recorded at a fixed sample rate. When such data is reported, the sampleRate defines the rate at which successive samples of data were recorded.

The sampleRate attribute provides the information required by a client software application to interpret the data and the sampling time relationship between successive values contained in the set of data.

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

7.2.2.12 representation Attribute for DataItem

Some data types provide data that may consist of a series of values or a file of data, not a single value. Other data types provide a series of data values that may require additional information so that the data may be correctly understood by a client software application.

When such data is provided, the representation attribute MUST be used to define the format for the data provided.

The types of representation defined are provided in Table 25.

Note: See MTConnect Standard: Part 3.0 - Streams Information Model for more information on the structure and format of each representation.

Table 25: DataItem attribute representation type

<table>
<thead>
<tr>
<th>Representation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA_SET</td>
<td>The reported value(s) are represented as a set of key-value pairs. Each reported value in the Data Set MUST have a unique key.</td>
</tr>
</tbody>
</table>
Continuation of Table 25

<table>
<thead>
<tr>
<th>Representation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISCRETE</td>
<td><strong>DEPRECATED</strong> as a representation in MTConnect Version 1.5. Replaced by the discrete attribute for a <a href="#">Data Entity</a> – <a href="#">Section 7.2.2.14 - discrete Attribute for DataItem</a>. A Data Entity where each discrete occurrence of the data may have the same value as the previous occurrence of the data. There is no reported state change between occurrences of the data. In this case, duplicate occurrences of the same data value SHOULD NOT be suppressed. An example of a DISCRETE data type would be a parts counter that reports the completion of each part versus the accumulation of parts. Another example would be a Message that does not typically have a reset state and may re-occur each time a specific message is triggered.</td>
</tr>
<tr>
<td>TIME_SERIES</td>
<td>A series of sampled data. The data is reported for a specified number of samples and each sample is reported with a fixed period.</td>
</tr>
<tr>
<td>VALUE</td>
<td>The measured value of the sample data. If no representation is specified for a data item, the representation <strong>MUST</strong> be determined to be <strong>VALUE</strong>.</td>
</tr>
<tr>
<td>Representation</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>TABLE</td>
<td>A <strong>Table</strong> is a two dimensional set of <strong>key-value pairs</strong> where the <strong>Entry</strong> represents a row, and the value is a set of <strong>key-value pairs</strong> <strong>Cell</strong> elements. The <strong>Table</strong> follows the same behavior as the <strong>Data Set</strong> for change tracking, clearing, and history. When an <strong>Entry</strong> changes, all <strong>Cell</strong> elements update as a single unit following the behavior of a <strong>Data Set</strong>. Note: It is best to use the <strong>VARIABLE DataItem</strong> type if the <strong>Cell</strong> elements represent multiple semantic types. Each <strong>Entry</strong> in the <strong>Table</strong> <strong>MUST</strong> have a unique key. Each <strong>Cell</strong> of each <strong>Entry</strong> in the <strong>Table</strong> <strong>MUST</strong> have a unique key. See Section 5.6.5 of MTConnect Standard: Part 3.0 - Streams Information Model, for a description of <strong>Entry</strong> and <strong>Cell</strong> elements.</td>
</tr>
</tbody>
</table>

### 7.2.2.13 significantDigits Attribute for DataItem

**significantDigits** is used to specify the level of precision (number of significant digits) for the value provided for a data item.

**significantDigits** attribute is not required for a data item, but it is recommended and **SHOULD** be used for any data item reporting a numeric value.

### 7.2.2.14 discrete Attribute for DataItem

An indication signifying whether each value reported for the **Data Entity** is significant and whether duplicate values are to be suppressed.

The value defined **MUST** be either true or false - an **XML** boolean type.

true indicates that each update to the **Data Entity**’s value is significant and duplicate values **MUST NOT** be suppressed.

false indicates that duplicated values **MUST** be suppressed.
If a value is not defined for discrete, the default value **MUST** be false.

### 7.2.3 Elements for DataItem

Table 26 lists the elements defined to provide additional information for a DataItem type XML element.

#### Table 26: Elements for DataItem

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Source is an optional XML element that identifies the Component, DataItem, or Composition representing the area of the piece of equipment from which a measured value originates. Additionally, Source <strong>MAY</strong> provide information relating to the identity of a measured value. This information is reported as CDATA for Source. (example, a PLC tag)</td>
<td>0..1</td>
</tr>
<tr>
<td>Constraints</td>
<td>Constraints is an optional container that provides a set of expected values that can be reported for this DataItem. Constraints are used by a software application to evaluate the validity of the reported data.</td>
<td>0..1</td>
</tr>
<tr>
<td>Filters</td>
<td>An optional container for the Filter elements associated with this DataItem element.</td>
<td>0..1</td>
</tr>
<tr>
<td>InitialValue</td>
<td>InitialValue is an optional XML element that defines the starting value for a data item as well as the value to be set for the data item after a reset event. Only one InitialValue element may be defined for a data item. The value will be constant and cannot change. If no InitialValue element is defined for a data item that is periodically reset, then the starting value for the data item <strong>MUST</strong> be a value of 0.</td>
<td>0..1</td>
</tr>
</tbody>
</table>
### Continuation of Table 26

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResetTrigger</td>
<td>ResetTrigger is an optional XML element that identifies the type of event that may cause a reset to occur. It is additional information regarding the meaning of the data that establishes an understanding of the time frame that the data represents so that the data may be correctly understood by a client software application.</td>
<td>0..1</td>
</tr>
<tr>
<td>Definition</td>
<td>The Definition defines the meaning of Entry and Cell elements associated with the DataItem when the representation is either DATA_SET or TABLE.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

#### 7.2.3.1 Source Element for DataItem

Source is an optional XML element that may be used to identify the physical part of a piece of equipment where the data represented by DataItem originated and/or it may be used to identify a complex name or an alternate name used to identify the data where it originated (e.g. a PLC tag name).

As an example, data related to a servo motor on an Axes component may actually originate from a measurement made in the Controller element.

In the case where the real name associated with a DataItem element is either complex or does not meet the format requirements of a NMTOKEN XML type, the real name of the element may not be able to be expressed in the name attribute. Additionally, a second or alternate name may be required to describe a piece of data. An example of this case would be the identity of the bit address in a PLC that represents this piece of data (PLC address I0015.4). When these cases occur, the alternate name can be provided as the value for the CDATA for Source.

The XML schema in *Figure 13* represents the structure of the Source XML element showing the attributes defined for Source.
1123  **7.2.3.1.1 Attributes for Source**

1124 *Table 27* identifies the attributes available to identify Source for a measured value:

**Table 27: Attributes for Source**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>componentId</td>
<td>The identifier attribute of the Component element that represents the physical part of a piece of equipment where the data represented by the DataItem element originated.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td><strong>A Valid Data Value reported for componentId MUST be the value of the id attribute for the Component element identified. componentId is an optional attribute.</strong></td>
<td></td>
</tr>
<tr>
<td>dataItemId</td>
<td>The identifier attribute of the DataItem that represents the originally measured value of the data referenced by this data item.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td><strong>A Valid Data Value reported for dataItemId MUST be the value of the id attribute for the DataItem element identified. dataItemId is an optional attribute.</strong></td>
<td></td>
</tr>
</tbody>
</table>
Continuation of Table 27

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>compositionId</td>
<td>The identifier attribute of the Composition element that represents the physical part of a piece of equipment where the data represented by the DataItem element originated.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>A <strong>Valid Data Value</strong> reported for <strong>compositionId</strong> <strong>MUST</strong> be the value of the id attribute for the Composition element identified.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>compositionId</strong> is an optional attribute.</td>
<td></td>
</tr>
</tbody>
</table>

Note: †One of componentID, compositionId, or dataItemId MUST be provided.

7.2.3.2 Constraints Element for DataItem

For some types of DataItem elements, the expected value(s) for the data reported for the DataItem **MAY** be restricted to specific values or a range of values.

Constraints is an optional XML element that provides a way to define the expected value(s) or the upper and lower limits for the range of values that are expected to be reported in response to a **Current Request** or **Sample Request**.

Constraints are used by a software application to evaluate the validity of the data reported.

The value associated with each Constraint element is reported in the CDATA for that element.

7.2.3.2.1 Schema for Constraints

The XML schema in *Figure 14* represents the structure of the Constraints XML element and the elements defined for Constraints.
Figure 14: Constraints Diagram

Table 28 identifies the elements available to identify Constraints for a measured value:
Table 28: Elements for Constraints

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>Value represents a single data value that is expected to be reported for a DataItem element. The data value is provided in the CDATA for this element and may be any numeric or text content. When there are multiple data values that may be expected to be reported for a DataItem element, multiple Value elements may be defined. In the case where only one Value element is defined, the data returned in response to a Current Request or Sample Request request MUST be the data value defined for Value element. Value MUST NOT be used in conjunction with any other Constraint elements.</td>
<td>0..*</td>
</tr>
<tr>
<td>Maximum</td>
<td>If the data reported for a data item is a range of numeric values, the expected value reported MAY be described with an upper limit defined by this constraint. The data value is provided in the CDATA for this element and MUST be a value using the same units as the reported data.</td>
<td>0..1</td>
</tr>
<tr>
<td>Minimum</td>
<td>If the data reported for a data item is a range of numeric values, the expected value reported MAY be described with a lower limit defined by this constraint. The data value is provided in the CDATA for this element and MUST be a value using the same units as the reported data.</td>
<td>0..1</td>
</tr>
<tr>
<td>Nominal</td>
<td>The target or expected value for this data item. The data value is provided in the CDATA for this element and MUST be a value using the same units as the reported data.</td>
<td>0..1</td>
</tr>
</tbody>
</table>
Continuation of Table 28

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter</td>
<td><strong>DEPRECATED</strong> in Version 1.4 – Moved to the Filters element of a DataItem.</td>
<td>0..1 †</td>
</tr>
</tbody>
</table>

If the data reported for a DataItem is a numeric value, a new value MUST NOT be reported if the change from the last reported value is less than the delta given as the CDATA of this element. Filter is an abstract type XML element. As such, Filter will never appear in the XML document, but will be replaced by a Filter type. The only currently supported Filter type is MINIMUM_DELTA. The CDATA MUST be an absolute value using the same Units as the reported data. Additional filter types MAY be supported in the future.

Note: †Remains in schema for backwards compatibility.

### 7.2.3.3 Filters Element for DataItem

Filters is an optional XML container that organizes the Filter elements for DataItem.

Filters contains one or more Filter XML elements.

**Table 29: MTConnect Filters Element**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filters</td>
<td>An XML container consisting of one or more types of Filter XML elements. Only one Filters container MAY appear for a DataItem element.</td>
<td>0..1</td>
</tr>
</tbody>
</table>
7.2.3.3.1 Filter

Filter provides a means to control when an Agent records updated information for a data item. Currently, there are two types of Filter elements defined in the MTConnect Standard - MINIMUM_DELTA and PERIOD. More Filter types may be added in the future.

The value associated with each Filter element is reported in the CDATA for that element.

Figure 15 represents the structure for Filter XML element.

Table 30 describes the types of Filter defined for a DataItem element and the expected behavior of an Agent when a Filter is applied to DataItem element.

<table>
<thead>
<tr>
<th>type</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>MINIMUM_DELTA</td>
<td>For a MINIMUM_DELTA type Filter, a new value <strong>MUST NOT</strong> be reported for a data item unless the measured value has changed from the last reported value by at least the delta given as the CDATA of this element. The CDATA <strong>MUST</strong> be an absolute value using the same units as the reported data.</td>
<td>0..1 †</td>
</tr>
<tr>
<td>Type</td>
<td>Description</td>
<td>Occurrence</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>PERIOD</td>
<td>For a PERIOD type Filter, the data reported for a data item is provided on a periodic basis. The PERIOD for reporting data is defined in the CDATA for the Filter. The CDATA MUST be an absolute value reported in seconds representing the time between reported samples of the value of the data item. If the PERIOD is smaller than one second, the number can be represented as a floating point number. For example, a PERIOD of 100 milliseconds would be 0.1.</td>
<td>0..1†</td>
</tr>
</tbody>
</table>

†Note: Either MINIMUM_DELTA or PERIOD can be defined, not both.

7.2.3.4 InitialValue Element for DataItem

InitialValue is an XML element that defines the value to be set for the data item after a reset event.

The value associated with the InitialValue element is reported in the CDATA for this element and MUST be an absolute value using the same units as the reported data.

7.2.3.5 ResetTrigger Element for DataItem

The value of some data types is periodically reset to the value of the InitialValue element. These reset events may be based upon a specific elapsed time or may be triggered by a physical or logical reset action that causes the reset to occur. ResetTrigger provides additional information regarding the meaning of the data – establishing an understanding of the time frame that the data represents so that the data may be correctly understood by a client software application.
Table 31: MTConnect ResetTrigger Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResetTrigger</td>
<td>ResetTrigger is an XML element that describes the reset action that causes a reset to occur. It is additional information regarding the meaning of the data that establishes an understanding of the time frame that the data represents so that the data may be correctly understood by a client software application.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

The reset action that **MAY** cause a reset to occur is provided in the CDATA for this element.

The reset actions that may cause a reset to occur are described in Table 32.

Table 32: DataItem Element ResetTrigger type

<table>
<thead>
<tr>
<th>Reset Actions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTION_COMPLETE</td>
<td>The value of the Data Entity that is measuring an action or operation is to be reset upon completion of that action or operation.</td>
</tr>
<tr>
<td>ANNUAL</td>
<td>The value of the Data Entity is to be reset at the end of a 12-month period.</td>
</tr>
<tr>
<td>DAY</td>
<td>The value of the Data Entity is to be reset at the end of a 24-hour period.</td>
</tr>
<tr>
<td>LIFE</td>
<td>The value of the Data Entity is not reset and accumulates for the entire life of the piece of equipment.</td>
</tr>
<tr>
<td>MAINTENANCE</td>
<td>The value of the Data Entity is to be reset upon completion of a maintenance event.</td>
</tr>
<tr>
<td>MONTH</td>
<td>The value of the Data Entity is to be reset at the end of a monthly period.</td>
</tr>
<tr>
<td>POWER_ON</td>
<td>The value of the Data Entity is to be reset when power was applied to the piece of equipment after a planned or unplanned interruption of power has occurred.</td>
</tr>
</tbody>
</table>
Continuation of Table 32

<table>
<thead>
<tr>
<th>Reset Actions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHIFT</td>
<td>The value of the <strong>Data Entity</strong> is to be reset at the end of a work shift.</td>
</tr>
<tr>
<td>WEEK</td>
<td>The value of the <strong>Data Entity</strong> is to be reset at the end of a 7-day period.</td>
</tr>
</tbody>
</table>

### 7.2.3.6 Definition Element for DataItem

*Figure 16* represents the **XML Schema** structure for **Definition** element.
Figure 16: Definition Schema Diagram
The Definition provides additional descriptive information for any DataItem representations. When the representation is either DATA_SET or TABLE, it gives the specific meaning of a key and MAY provide a Description, type, and units for semantic interpretation of data.

### Table 33: Elements for Definition

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The Description of the Definition. See Component Description</td>
<td>0..1</td>
</tr>
<tr>
<td>EntryDefinitions</td>
<td>The EntryDefinitions aggregates EntryDefinition.</td>
<td>0..1</td>
</tr>
<tr>
<td>CellDefinitions</td>
<td>The CellDefinitions aggregates CellDefinition.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### 7.2.3.6.1 EntryDefinitions Element for Definition

The EntryDefinitions aggregates EntryDefinition for Definition.

Elements for EntryDefinitions

### Table 34: Elements for EntryDefinitions

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryDefinition</td>
<td>The semantic definition of an Entry</td>
<td>1..*</td>
</tr>
</tbody>
</table>

### 7.2.3.6.2 EntryDefinition Element for Definition

When the representation is DATA_SET, the EntryDefinition provides the Description, units, and type of each Entry identified by a unique key.

When the representation is TABLE, the EntryDefinition provides a Description and a set of CellDefinitions for an Entry identified by a unique key.

The key for the EntryDefinition MUST be unique for a given DataItem Definition.
### Attributes for EntryDefinition

**Table 35**: Attributes for EntryDefinition

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>key</td>
<td>The unique identification of the Entry in the Definition. The description applies to all Entry observations having this key.</td>
<td>1</td>
</tr>
<tr>
<td>units</td>
<td>Same as DataItem units. See Section 7.2.2.5 - units Attribute for DataItem. Only valid for representation of DATA_SET.</td>
<td>0..1</td>
</tr>
<tr>
<td>type</td>
<td>Same as DataItem type. See Section 8 - Listing of Data Items</td>
<td>0..1</td>
</tr>
<tr>
<td>subType</td>
<td>Same as DataItem subType. See Section 8 - Listing of Data Items</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### Elements for EntryDefinition

**Table 36**: Elements for EntryDefinition

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The Description of the EntryDefinition. See Component Description</td>
<td>0..1</td>
</tr>
<tr>
<td>CellDefinitions</td>
<td>The CellDefinitions aggregates CellDefinition if the representation is TABLE.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

#### 7.2.3.6.3 CellDefinitions Element for Definition

The CellDefinitions aggregates CellDefinition declarations.
**Elements for CellDefinitions**

**Table 37: Elements for CellDefinitions**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>CellDefinition</td>
<td>The semantic definition of a Cell.</td>
<td>1..*</td>
</tr>
</tbody>
</table>

**7.2.3.6.4 CellDefinition Element for CellDefinitions**

When the representation is TABLE, the CellDefinition provides the Description and the units associated each Cell by key.

The key for the CellDefinition MUST be unique for a given Definition or EntryDefinition.

**Attributes for CellDefinition**

**Table 38: Attributes for CellDefinition**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>key</td>
<td>The unique identification of the Entry in the Definition. The description applies to all Entry observations having this key.</td>
<td>1</td>
</tr>
<tr>
<td>units</td>
<td>Same as DataItem units. See Section 7.2.2.5 - units for DataItem.</td>
<td>0..1</td>
</tr>
<tr>
<td>type</td>
<td>Same as DataItem type. See Section 8 - Listing of Data Items.</td>
<td>0..1</td>
</tr>
<tr>
<td>subType</td>
<td>Same as DataItem subType. See Section 8 - Listing of Data Items.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Elements for CellDefinition**

**Table 39: Elements for CellDefinition**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The Description of the CellDefinition. See Component Description</td>
<td>0..1</td>
</tr>
</tbody>
</table>
8 Listing of Data Items

In the MTConnect Standard, `DataItem` elements are defined and organized based upon the `category` and `type` attributes. The `category` attribute provides a high level grouping for `DataItem` elements based on the kind of information that is reported by the data item.

These categories are:

- **SAMPLE**
  
  A `SAMPLE` reports a continuously variable or analog data value.

- **EVENT**
  
  An `EVENT` reports information representing a functional state, with two or more discrete values, associated with a component or it contains a message. The data provided may be a numeric value or text.

- **CONDITION**
  
  A `CONDITION` reports information about the health of a piece of equipment and its ability to function.

The `type` attribute specifies the specific kind of data that is reported. For some types of data items, a `subType` attribute may also be used to differentiate between multiple data items of the same `type` where the information reported by the data item has a different, but related, meaning.

Many types of data items provide two forms of data: a value (reported as either a `SAMPLE` or `EVENT`) and a health status (reported as a `CONDITION`). These `DataItem` types **MAY** be defined in more than one `category` based on the data that they report.
## 8.1 Data Items in category SAMPLE

The types of DataItem elements in the SAMPLE category report data representing a continuously changing or analog data value. This data can be measured at any point-in-time and will always produce a result. The data provided may be a scalar floating point number or integers that have an infinite number of possible values. The units attribute MUST be defined and reported for each DataItem in this category.

Table 40 defines the types and subtypes of DataItem elements defined for the SAMPLE category. The subtypes are indented below their associated types.

### Table 40: DataItem type subType for category SAMPLE

<table>
<thead>
<tr>
<th>DataItem type/subType</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCELERATION</td>
<td>Rate of change of velocity.</td>
<td>MILLIMETER/SECOND²</td>
</tr>
<tr>
<td>ACCUMULATED_TIME</td>
<td>The measurement of accumulated time for an activity or event.</td>
<td>SECOND</td>
</tr>
<tr>
<td></td>
<td><strong>DEPRECATION WARNING</strong>: May be deprecated in the future. Recommend using PROCESS_TIMER and EQUIPMENT_TIMER.</td>
<td></td>
</tr>
<tr>
<td>AMPERAGE - ACTUAL</td>
<td>The measured amperage being delivered from a power source.</td>
<td>AMPERE</td>
</tr>
<tr>
<td></td>
<td><strong>DEPRECATED in Version 1.6.</strong> Replaced by AMPERAGE_AC and AMPERAGE_DC.</td>
<td></td>
</tr>
<tr>
<td>AMPERAGE - ALTERNATING</td>
<td>The measurement of alternating current. If not specified further in statistic, defaults to RMS voltage.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>AMPERAGE - DIRECT</td>
<td>The measurement of DC current.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>TARGET</td>
<td>The desired or preset amperage to be delivered from a power source.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>AMPERAGE_AC</td>
<td>The measurement of an electrical current that reverses direction at regular short intervals. A subType MUST always be specified. If not specified further in statistic, defaults to RMS amperage.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured amperage within an electrical circuit.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The value for a current as specified by a component.</td>
<td>AMPERE</td>
</tr>
<tr>
<td></td>
<td>The COMMANDED current is a calculated value that includes adjustments and overrides.</td>
<td></td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The value for a current as specified by a logic or motion program or set by a switch.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>AMPERAGE_DC</td>
<td>The measurement of an electric current flowing in one direction only.</td>
<td>AMPERE</td>
</tr>
<tr>
<td></td>
<td>A subType MUST always be specified.</td>
<td></td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured amperage within an electrical circuit.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The value for a current as specified by a component.</td>
<td>AMPERE</td>
</tr>
<tr>
<td></td>
<td>The <strong>COMMANDED</strong> current is a calculated value that includes adjustments and overrides.</td>
<td></td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The value for a current as specified by a logic or motion program or set by a switch.</td>
<td>AMPERE</td>
</tr>
<tr>
<td>ANGLE</td>
<td>The measurement of angular position.</td>
<td>DEGREE</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The actual angular position as read from the physical component.</td>
<td>DEGREE</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>A calculated value for angular position computed by the <strong>Controller type</strong> component.</td>
<td>DEGREE</td>
</tr>
<tr>
<td>ANGULAR ACCELERATION</td>
<td>Rate of change of angular velocity.</td>
<td>DEGREE/SECOND²</td>
</tr>
<tr>
<td>ANGULAR VELOCITY</td>
<td>Rate of change of angular position.</td>
<td>DEGREE/SECOND</td>
</tr>
<tr>
<td>AXIS FEEDRATE</td>
<td>The feedrate of a linear axis.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value of the feedrate of a linear axis.</td>
<td>MILLIMETER/SECOND</td>
</tr>
</tbody>
</table>
### Continuation of Table 40: DataItem type subType for category SAMPLE

<table>
<thead>
<tr>
<th>DataItem type/subType</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMANDED</td>
<td>The feedrate of a linear axis as specified by the Controller type component. The COMMANDED feedrate is a calculated value that includes adjustments and overrides.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>JOG</td>
<td>The feedrate specified by a logic or motion program, by a pre-set value, or set by a switch as the feedrate for a linear axis when operating in a manual state or method (jogging).</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>OVERRIDE</td>
<td>The operator’s overridden value. Percent of commanded. <strong>DEPRECATED</strong> in Version 1.3. See EVENT category data items.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The feedrate specified by a logic or motion program or set by a switch for a linear axis.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>RAPID</td>
<td>The feedrate specified by a logic or motion program, by a pre-set value, or set by a switch as the feedrate for a linear axis when operating in a rapid positioning mode.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>CAPACITY_FLUID</td>
<td>The fluid capacity of an object or container.</td>
<td>MILLILITER</td>
</tr>
<tr>
<td>CAPACITY_SPATIAL</td>
<td>The geometric capacity of an object or container.</td>
<td>CUBIC_MILLIMETER</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>CLOCK_TIME</td>
<td>The value provided by a timing device at a specific point in time. <strong>CLOCK_TIME MUST</strong> be reported in W3C ISO 8601 format.</td>
<td>yyyy-mm-ddth:mm:ss.ffff</td>
</tr>
<tr>
<td>CONCENTRATION</td>
<td>Percentage of one component within a mixture of components.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>CONDUCTIVITY</td>
<td>The ability of a material to conduct electricity.</td>
<td>SIEMENS/METER</td>
</tr>
<tr>
<td>CUTTING_SPEED</td>
<td>The speed difference (relative velocity) between the cutting mechanism and the surface of the workpiece it is operating on.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value between the cutting mechanism and the surface of the workpiece it is operating on.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded value between the cutting mechanism and the surface of the workpiece it is operating on.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The programmed value between the cutting mechanism and the surface of the workpiece it is operating on.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>DENSITY</td>
<td>The volumetric mass of a material per unit volume of that material.</td>
<td>MILLIGRAM/CUBIC_-MILLIMETER</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>DEPOSITION_ACCELERATION_VOLUMETRIC</td>
<td>The rate of change in spatial volume of material deposited in an additive manufacturing process.</td>
<td>CUBIC_- MILLIMETER/SECOND^2</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured rate of change in spatial volume of material deposited in an additive manufacturing process.</td>
<td>CUBIC_- MILLIMETER/SECOND^2</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded rate of change in spatial volume of material to be deposited in an additive manufacturing process.</td>
<td>CUBIC_- MILLIMETER/SECOND^2</td>
</tr>
<tr>
<td>DEPOSITION_DENSITY</td>
<td>The density of the material deposited in an additive manufacturing process per unit of volume.</td>
<td>MILLIGRAM/CUBIC_- MILLIMETER</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured density of the material deposited in an additive manufacturing process.</td>
<td>MILLIGRAM/CUBIC_- MILLIMETER</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded density of material to be deposited in an additive manufacturing process.</td>
<td>MILLIGRAM/CUBIC_- MILLIMETER</td>
</tr>
<tr>
<td>DEPOSITION_MASS</td>
<td>The mass of the material deposited in an additive manufacturing process.</td>
<td>MILLIGRAM</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured mass of the material deposited in an additive manufacturing process.</td>
<td>MILLIGRAM</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded mass of the material to be deposited in an additive manufacturing process.</td>
<td>MILLIGRAM</td>
</tr>
<tr>
<td>DEPOSITION_RATE_-VOLUMETRIC</td>
<td>The rate at which a spatial volume of material is deposited in an additive manufacturing process.</td>
<td>CUBIC_-MILLIMETER/SECOND</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured rate at which a spatial volume of material is deposited in an additive manufacturing process.</td>
<td>CUBIC_-MILLIMETER/SECOND</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The programmed rate at which a spatial volume of material is to be deposited in an additive manufacturing process.</td>
<td>CUBIC_-MILLIMETER/SECOND</td>
</tr>
<tr>
<td>DEPOSITION_VOLUME</td>
<td>The spatial volume of material to be deposited in an additive manufacturing process.</td>
<td>CUBIC_MILLIMETER</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured spatial volume of material deposited.</td>
<td>CUBIC_MILLIMETER</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The target spatial volume of material to be deposited.</td>
<td>CUBIC_MILLIMETER</td>
</tr>
<tr>
<td>DIAMETER</td>
<td>The measured dimension of a diameter.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>DISPLACEMENT</td>
<td>The change in position of an object.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>ELECTRICAL_ENERGY</td>
<td>The measurement of electrical energy consumption by a component.</td>
<td>WATT_SECOND</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>EQUIPMENT_TIMER</td>
<td>The measurement of the amount of time a piece of equipment or a sub-part of a piece of equipment has performed specific activities. Often used to determine when maintenance may be required for the equipment. Multiple subTypes of EQUIPMENT_TIMER MAY be defined. A subType MUST always be specified.</td>
<td>SECOND</td>
</tr>
<tr>
<td>DELAY</td>
<td>Measurement of the time that a piece of equipment is waiting for an event or an action to occur.</td>
<td>SECOND</td>
</tr>
<tr>
<td>LOADED</td>
<td>Measurement of the time that the sub-parts of a piece of equipment are under load. Example: For traditional machine tools, this is a measurement of the time that the cutting tool is assumed to be engaged with the part.</td>
<td>SECOND</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>OPERATING</td>
<td>Measurement of the time that the major sub-parts of a piece of equipment are powered or performing any activity whether producing a part or product or not. Example: For traditional machine tools, this includes WORKING, plus idle time.</td>
<td>SECOND</td>
</tr>
<tr>
<td>POWERED</td>
<td>The measurement of time that primary power is applied to the piece of equipment and, as a minimum, the controller or logic portion of the piece of equipment is powered and functioning or components that are required to remain on are powered. Example: Heaters for an extrusion machine that are required to be powered even when the equipment is turned off</td>
<td>SECOND</td>
</tr>
<tr>
<td>WORKING</td>
<td>Measurement of the time that a piece of equipment is performing any activity the equipment is active and performing a function under load or not. Example: For traditional machine tools, this includes LOADED, plus rapid moves, tool changes, etc.</td>
<td>SECOND</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>FILL_LEVEL</td>
<td>The measurement of the amount of a substance remaining compared to the planned maximum amount of that substance.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>FLOW</td>
<td>The rate of flow of a fluid.</td>
<td>LITER/SECOND</td>
</tr>
<tr>
<td>FREQUENCY</td>
<td>The measurement of the number of occurrences of a repeating event per unit time.</td>
<td>Hertz</td>
</tr>
<tr>
<td>GLOBAL_POSITION</td>
<td>DEPRECATED in Version 1.1</td>
<td>None</td>
</tr>
<tr>
<td>HUMIDITY_ABSOLUTE</td>
<td>The amount of water vapor expressed in grams per cubic meter.</td>
<td>GRAM/CUBIC_METER</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value.</td>
<td>GRAM/CUBIC_METER</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded value.</td>
<td>GRAM/CUBIC_METER</td>
</tr>
<tr>
<td>HUMIDITY_RELATIVE</td>
<td>The amount of water vapor present expressed as a percent to reach saturation at the same temperature.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded value.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>HUMIDITY_SPECIFIC</td>
<td>The ratio of the water vapor present over the total weight of the water vapor and air present expressed as a percent.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded value.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>LENGTH</td>
<td>The length of an object.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>REMAINING</td>
<td>The remaining total length of an object.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>STANDARD</td>
<td>The standard or original length of an object.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>USEABLE</td>
<td>The remaining useable length of an object.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>LEVEL</td>
<td><strong>DEPRECATED</strong> in Version 1.2. See FILL_LEVEL</td>
<td>None</td>
</tr>
<tr>
<td>LINEAR_FORCE</td>
<td>The measurement of the push or pull introduced by an actuator or exerted on an object.</td>
<td>NEWTON</td>
</tr>
<tr>
<td>LOAD</td>
<td>The measurement of the actual versus the standard rating of a piece of equipment.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>MASS</td>
<td>The measurement of the mass of an object(s) or an amount of material.</td>
<td>KILOGRAM</td>
</tr>
<tr>
<td>ORIENTATION</td>
<td>A measured or calculated orientation of a plane or vector relative to a cartesian coordinate system.</td>
<td>DEGREE_3D</td>
</tr>
<tr>
<td></td>
<td><strong>ORIENTATION</strong> SHOULD have a coordinateSystemIdRef or a coordinateSystem attribute, otherwise the coordinateSystem attribute MUST default to WORK coordinates.</td>
<td></td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value.</td>
<td>DEGREE_3D</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded value.</td>
<td>DEGREE_3D</td>
</tr>
<tr>
<td>PATH_FEEDRATE</td>
<td>The feedrate for the axes, or a single axis, associated with a Path component—a vector.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value of the feedrate of the axes, or a single axis, associated with a path component.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The feedrate as specified by the Controller type component for the axes, or a single axis, associated with a Path component. The COMMANDED feedrate is a calculated value that includes adjustments and overrides.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>JOG</td>
<td>The feedrate specified by a logic or motion program, by a pre-set value, or set by a switch as the feedrate for the axes, or a single axis, associated with a Path when operating in a manual state or method (jogging).</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>OVERRIDE</td>
<td>The operator’s overridden value. Percent of commanded. DEPRECATED in Version 1.3. See EVENT category data items.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The feedrate specified by a logic or motion program or set by a switch as the feedrate for the axes, or a single axis, associated with a Path.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>RAPID</td>
<td>The feedrate specified by a logic or motion program, by a pre-set value, or set by a switch as the feedrate for the axes, or a single axis, associated with a Path when operating in a rapid positioning mode.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>PATH_FEEDRATE_-PER_REVOLUTION</td>
<td>The feedrate for the axes, or a single axis.</td>
<td>MILLIMETER/REVO-LUTION</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value of the feedrate of the axes, or a single axis.</td>
<td>MILLIMETER/REVO-LUTION</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The feedrate as specified by the Controller for the axes, or a single axis. The COMMANDED feedrate is a calculated value that includes adjustments and overrides.</td>
<td>MILLIMETER/REVO-LUTION</td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The feedrate specified by a logic or motion program or set by a switch as the feedrate for the axes, or a single axis.</td>
<td>MILLIMETER/REVO-LUTION</td>
</tr>
</tbody>
</table>
Continuation of Table 40: Data Item type subType for category SAMPLE

<table>
<thead>
<tr>
<th>Data Item type/subType</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>PATH_POSITION</td>
<td>A measured or calculated position of a control point associated with a piece of equipment. The control point <strong>MUST</strong> be reported as a set of space-delimited floating-point numbers representing a point in 3-D space. The position of the control point <strong>MUST</strong> be reported in units of MILLIMETER and listed in order of X, Y, and Z referenced to the coordinate system of the piece of equipment. Any control point representing a position in 1-D or 2-D space <strong>MAY</strong> be represented in terms of 3-D space by setting any undefined coordinate to zero (0). <strong>PATH_POSITION SHOULD</strong> be further defined with a coordinateSystem attribute. If a coordinateSystem attribute is not specified, the position of the control point <strong>MUST</strong> be reported in WORK coordinates.</td>
<td>MILLIMETER_3D</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured position of the current program control point as reported by the piece of equipment.</td>
<td>MILLIMETER_3D</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The position of the control point specified by a logic or motion program.</td>
<td>MILLIMETER_3D</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The position computed by the Controller type component.</td>
<td>MILLIMETER_3D</td>
</tr>
<tr>
<td>PROBE</td>
<td>The position provided by a measurement probe.</td>
<td>MILLIMETER_3D</td>
</tr>
<tr>
<td>TARGET</td>
<td>The desired end position for a movement or a series of movements. Multiple discrete movements may need to be completed to achieve the final TARGET position.</td>
<td>MILLIMETER_3D</td>
</tr>
<tr>
<td>PH</td>
<td>The measurement of the acidity or alkalinity.</td>
<td>PH</td>
</tr>
<tr>
<td>POSITION</td>
<td>A measured or calculated position of a Component element as reported by a piece of equipment. POSITION SHOULD be further defined with a coordinateSystem attribute. If a coordinateSystem attribute is not specified, the position of the control point MUST be reported in MACHINE coordinates.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The physical measured position of the control point for a Component.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>A position calculated by the <strong>Controller</strong> type component for a discrete movement.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The position of the control point for a <strong>Component</strong> specified by a logic or motion program.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>TARGET</td>
<td>The desired end position of the control point for a <strong>Component</strong> resulting from a movement or a series of movements. Multiple discrete movements may need to be completed to achieve the final TARGET position.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>POWER_FACTOR</td>
<td>The measurement of the ratio of real power flowing to a load to the apparent power in that AC circuit.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>PRESSURE</td>
<td>The force per unit area exerted by a gas or liquid.</td>
<td>PASCAL</td>
</tr>
</tbody>
</table>
Continuation of Table 40: DataItem type subType for category SAMPLE

<table>
<thead>
<tr>
<th>DataItem type/subType</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROCESS_TIMER</td>
<td>The measurement of the amount of time a piece of equipment has performed different types of activities associated with the process being performed at that piece of equipment. Multiple subtypes of PROCESS_TIMER may be defined. Typically, PROCESS_TIMER <strong>SHOULD</strong> be modeled as a data item for the Device element, but <strong>MAY</strong> be modeled for either a Controller or Path <em>Structural Element</em> in the <em>XML</em> document. A subType <strong>MUST</strong> always be specified.</td>
<td>SECOND</td>
</tr>
<tr>
<td>DELAY</td>
<td>Measurement of the time that a process is waiting and unable to perform its intended function.</td>
<td>SECOND</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PROCESS</td>
<td>The measurement of the time from the beginning of production of a part or product on a piece of equipment until the time that production is complete for that part or product on that piece of equipment. This includes the time that the piece of equipment is running, producing parts or products, or in the process of producing parts.</td>
<td>SECOND</td>
</tr>
<tr>
<td>RESISTANCE</td>
<td>The degree to which a substance opposes the passage of an electric current.</td>
<td>OHM</td>
</tr>
<tr>
<td>ROTARY_VELOCITY</td>
<td>The rotational speed of a rotary axis.</td>
<td>REVOLUTION/MINUTE</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value of rotational speed that the rotary axis is spinning.</td>
<td>REVOLUTION/MINUTE</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The rotational speed as specified by the Controller type component.</td>
<td>REVOLUTION/MINUTE</td>
</tr>
<tr>
<td></td>
<td>The COMMANDED velocity is a calculated value that includes adjustments and overrides.</td>
<td></td>
</tr>
<tr>
<td>OVERRIDE</td>
<td>The operator's overridden value. Percent of commanded. DEPRECATED in Version 1.3. See EVENT category data items.</td>
<td>PERCENT</td>
</tr>
</tbody>
</table>
Continuation of Table 40: DataItem type subType for category SAMPLE

<table>
<thead>
<tr>
<th>DataItem type/subType</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROGRAMMED</td>
<td>The rotational velocity specified by a logic or motion program or set by a switch.</td>
<td>REVOLUTION/MINUTE</td>
</tr>
<tr>
<td>SOUND_LEVEL</td>
<td>The measurement of a sound level or sound pressure level relative to atmospheric pressure.</td>
<td>DECIBEL</td>
</tr>
<tr>
<td>A_SCALE</td>
<td>A Scale weighting factor. This is the default weighting factor if no factor is specified</td>
<td>DECIBEL</td>
</tr>
<tr>
<td>B_SCALE</td>
<td>B Scale weighting factor</td>
<td>DECIBEL</td>
</tr>
<tr>
<td>C_SCALE</td>
<td>C Scale weighting factor</td>
<td>DECIBEL</td>
</tr>
<tr>
<td>D_SCALE</td>
<td>D Scale weighting factor</td>
<td>DECIBEL</td>
</tr>
<tr>
<td>NO_SCALE</td>
<td>No weighting factor on the frequency scale</td>
<td>DECIBEL</td>
</tr>
<tr>
<td>SPINDLE_SPEED</td>
<td><strong>DEPRECATED</strong> in Version 1.2. Replaced by ROTARY VELOCITY</td>
<td>REVOLUTION/MINUTE</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The rotational speed of a rotary axis. ROTARY_MODE MUST be SPINDLE.</td>
<td>REVOLUTION/MINUTE</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The rotational speed the as specified by the Controller type Component.</td>
<td>REVOLUTION/MINUTE</td>
</tr>
<tr>
<td>OVERRIDE</td>
<td>The operator’s overridden value. Percent of commanded.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>STRAIN</td>
<td>The amount of deformation per unit length of an object when a load is applied.</td>
<td>PERCENT</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>TEMPERATURE</td>
<td>The measurement of temperature.</td>
<td>CELSIUS</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured value.</td>
<td>CELSIUS</td>
</tr>
<tr>
<td>COMMANDED</td>
<td>The commanded value.</td>
<td>CELSIUS</td>
</tr>
<tr>
<td>TENSION</td>
<td>The measurement of a force that stretches or elongates an object.</td>
<td>NEWTON</td>
</tr>
<tr>
<td>TILT</td>
<td>The measurement of angular displacement.</td>
<td>MICRO_RADIAN</td>
</tr>
<tr>
<td>TORQUE</td>
<td>The turning force exerted on an object or by an object.</td>
<td>NEWTON_METER</td>
</tr>
<tr>
<td>VELOCITY</td>
<td>The rate of change of position.</td>
<td>MILLIMETER/SECOND</td>
</tr>
<tr>
<td>VISCOSITY</td>
<td>The measurement of a fluids resistance to flow.</td>
<td>PASCAL_SECOND</td>
</tr>
<tr>
<td>VOLTAGE</td>
<td><strong>DEPRECATED</strong> in Version 1.6. Replaced by VOLTAGE_AC and VOLTAGE_DC.</td>
<td>VOLT</td>
</tr>
<tr>
<td>-ACTUAL-</td>
<td>The measured voltage being delivered from a power source.</td>
<td>VOLT</td>
</tr>
<tr>
<td>-ALTERNATING-</td>
<td>The measurement of alternating voltage. If not specified further in statistic, defaults to RMS voltage.</td>
<td>VOLT</td>
</tr>
<tr>
<td>-DIRECT-</td>
<td>The measurement of DC voltage.</td>
<td>VOLT</td>
</tr>
<tr>
<td>-TARGET-</td>
<td>The desired or preset voltage to be delivered from a power source.</td>
<td>VOLT</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
</tbody>
</table>
| VOLTAGE_AC            | The measurement of the electrical potential between two points in an electrical circuit in which the current periodically reverses direction.  
  A subType **MUST** be specified.  
  If not specified further in statistic, defaults to RMS voltage. | VOLT |
| ACTUAL                | The measured voltage within an electrical circuit. | VOLT |
| COMMANDED             | The value for a voltage as specified by a Controller component.  
  The **COMMANDED** voltage is a calculated value that includes adjustments and overrides. | VOLT |
| PROGRAMMED            | The value for a voltage as specified by a logic or motion program or set by a switch. | VOLT |
| VOLTAGE_DC            | The measurement of the electrical potential between two points in an electrical circuit in which the current is unidirectional.  
  A subType **MUST** be specified. | VOLT |
<p>| ACTUAL                | The measured voltage within an electrical circuit. | VOLT |</p>
<table>
<thead>
<tr>
<th>DataItem type/subType</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMANDED</td>
<td>The value for a voltage as specified by a Controller component.</td>
<td>VOLT</td>
</tr>
<tr>
<td></td>
<td>The COMMANDED voltage is a calculated value that includes adjustments and overrides.</td>
<td></td>
</tr>
<tr>
<td>PROGRAMMED</td>
<td>The value for a voltage as specified by a logic or motion program or set by a switch.</td>
<td>VOLT</td>
</tr>
<tr>
<td>VOLT_AMPERE</td>
<td>The measurement 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).</td>
<td>VOLT_AMPERE</td>
</tr>
<tr>
<td>VOLT_AMPERE_-REACTIVE</td>
<td>The measurement of reactive power in an AC electrical circuit (commonly referred to as VAR).</td>
<td>VOLT_AMPERE_-REACTIVE</td>
</tr>
<tr>
<td>VOLUME_FLUID</td>
<td>The fluid volume of an object or container.</td>
<td>MILLILITER</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The amount of fluid currently present in an object or container.</td>
<td>MILLILITER</td>
</tr>
<tr>
<td>CONSUMED</td>
<td>The amount of fluid material consumed from an object or container during a manufacturing process.</td>
<td>MILLILITER</td>
</tr>
<tr>
<td>VOLUME_SPATIAL</td>
<td>The geometric volume of an object or container.</td>
<td>CUBIC_MILLIMETER</td>
</tr>
<tr>
<td>DataItem type/subType</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The amount of bulk material currently present in an object or container.</td>
<td>CUBIC_MILLIMETER</td>
</tr>
<tr>
<td>CONSUMED</td>
<td>The amount of bulk material consumed from an object or container during a manufacturing process.</td>
<td>CUBIC_MILLIMETER</td>
</tr>
<tr>
<td>WATTAGE</td>
<td>The measurement of power flowing through or dissipated by an electrical circuit or piece of equipment.</td>
<td>WATT</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The measured wattage being delivered from a power source.</td>
<td>WATT</td>
</tr>
<tr>
<td>TARGET</td>
<td>The desired or preset wattage to be delivered from a power source.</td>
<td>WATT</td>
</tr>
<tr>
<td>X_DIMENSION</td>
<td>Measured dimension of an entity relative to the X direction of the referenced coordinate system.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>Y_DIMENSION</td>
<td>Measured dimension of an entity relative to the Y direction of the referenced coordinate system.</td>
<td>MILLIMETER</td>
</tr>
<tr>
<td>Z_DIMENSION</td>
<td>Measured dimension of an entity relative to the Z direction of the referenced coordinate system.</td>
<td>MILLIMETER</td>
</tr>
</tbody>
</table>
### 8.2 Data Items in category EVENT

DataItem types in the EVENT category represent a discrete piece of information from a piece of equipment. EVENT does not have intermediate values that vary over time.

An EVENT is information that, when provided at any specific point in time, represents the current state of the piece of equipment.

There are two types of EVENT: those representing state, with two or more discrete values, and those representing messages that contain plain text data.

*Table 41* defines the DataItem types and subtypes defined for the EVENT category. The subtypes are indented below their associated types.

**Table 41: DataItem type subType for category EVENT**

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
</table>
| ACTIVE_AXES           | The set of axes currently associated with a Path or Controller *Structural Element*.  
If this DataItem is not provided, it will be assumed that all axes are currently associated with the Controller *Structural Element* and with an individual Path.  

The [Valid Data Value](https://example.com) for ACTIVE_AXES SHOULD be a space-delimited set of axes reported as the value of the name attribute for each axis. If name is not available, the piece of equipment MUST report the value of the nativeName attribute for each axis. |
| ACTUATOR_STATE        | Represents the operational state of an apparatus for moving or controlling a mechanism or system.  

The [Valid Data Value](https://example.com) MUST be ACTIVE or INACTIVE. |
<p>| ALARM                 | DEPRECATED in Version 1.1. Replaced with CONDITION category. |</p>
<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLICATION</td>
<td>The application on a component. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string. A <strong>subType</strong> <strong>MUST</strong> always be specified.</td>
</tr>
<tr>
<td>LICENSE</td>
<td>The license code to validate or activate the hardware or software.</td>
</tr>
<tr>
<td>VERSION</td>
<td>The version of the hardware or software.</td>
</tr>
<tr>
<td>RELEASE_DATE</td>
<td>The date the hardware or software was released for general use.</td>
</tr>
<tr>
<td>INSTALL_DATE</td>
<td>The date the hardware or software was installed.</td>
</tr>
<tr>
<td>MANUFACTURER</td>
<td>The corporate identity for the maker of the hardware or software.</td>
</tr>
<tr>
<td>AVAILABILITY</td>
<td>Represents the <strong>Agent</strong>’s ability to communicate with the data source. This <strong>MUST</strong> be provided for a <strong>Device Element</strong> and <strong>MAY</strong> be provided for any other <strong>Structural Element</strong>. The <strong>Valid Data Value</strong> <strong>MUST</strong> be <strong>AVAILABLE</strong> or <strong>UNAVAILABLE</strong>.</td>
</tr>
<tr>
<td>AXIS_COUPLING</td>
<td>Describes the way the axes will be associated to each other. This is used in conjunction with <strong>COUPLED_AXES</strong> to indicate the way they are interacting. The <strong>Valid Data Value</strong> <strong>MUST</strong> be <strong>TANDEM</strong>, <strong>SYNCHRONOUS</strong>, <strong>MASTER</strong>, and <strong>SLAVE</strong>. The coupling <strong>MUST</strong> be viewed from the perspective of a specific axis. Therefore, a <strong>MASTER</strong> coupling indicates that this axis is the master for the <strong>COUPLED_AXES</strong>.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AXIS_FEEDRATE_OVERRIDE</td>
<td>The value of a signal or calculation issued to adjust the feedrate of an individual linear type axis.</td>
</tr>
<tr>
<td></td>
<td>The value provided for AXIS_FEEDRATE_OVERRIDE is expressed as a percentage of the designated feedrate for the axis.</td>
</tr>
<tr>
<td></td>
<td>When AXIS_FEEDRATE_OVERRIDE is applied, the resulting commanded feedrate for the axis is limited to the value of the original feedrate multiplied by the value of the AXIS_FEEDRATE_OVERRIDE.</td>
</tr>
<tr>
<td></td>
<td>There MAY be different subtypes of AXIS_FEEDRATE_OVERRIDE; each representing an override value for a designated subtype of feedrate depending on the state of operation of the axis. The subtypes of operation of an axis are currently defined as PROGRAMMED, JOG, and RAPID.</td>
</tr>
<tr>
<td>JOG</td>
<td>The value of a signal or calculation issued to adjust the feedrate of an individual linear type axis when that axis is being operated in a manual state or method (jogging).</td>
</tr>
<tr>
<td></td>
<td>When the JOG subtype of AXIS_FEEDRATE_OVERRIDE is applied, the resulting commanded feedrate for the axis is limited to the value of the original JOG subtype of the AXIS_FEEDRATE multiplied by the value of the JOG subtype of AXIS_FEEDRATE_OVERRIDE.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
</table>
| PROGRAMMED            | The value of a signal or calculation issued to adjust the feedrate of an individual linear type axis that has been specified by a logic or motion program or set by a switch.  

When the PROGRAMMED subtype of AXIS_FEEDRATE_OVERRIDE is applied, the resulting commanded feedrate for the axis is limited to the value of the original PROGRAMMED subtype of the AXIS_FEEDRATE multiplied by the value of the PROGRAMMED subtype of AXIS_FEEDRATE_OVERRIDE. |
| RAPID                 | The value of a signal or calculation issued to adjust the feedrate of an individual linear type axis that is operating in a rapid positioning mode.  

When the RAPID subtype of AXIS_FEEDRATE_OVERRIDE is applied, the resulting commanded feedrate for the axis is limited to the value of the original RAPID subtype of the AXIS_FEEDRATE multiplied by the value of the RAPID subtype of AXIS_FEEDRATE_OVERRIDE. |
| AXIS_INTERLOCK        | An indicator of the state of the axis lockout function when power has been removed and the axis is allowed to move freely.  

The [Valid Data Value] MUST be ACTIVE or INACTIVE. |
| AXIS_STATE            | An indicator of the controlled state of a Linear or Rotary component representing an axis.  

The [Valid Data Value] MUST be HOME, TRAVEL, PARKED, or STOPPED. |
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCK</td>
<td>The line of code or command being executed by a Controller. The value reported for Block MUST include the entire expression for a line of program code, including all parameters.</td>
</tr>
<tr>
<td>BLOCK_COUNT</td>
<td>The total count of the number of blocks of program code that have been executed since execution started. BLOCK_COUNT counts blocks of program code executed regardless of program structure (e.g., looping or branching within the program). The starting value for BLOCK_COUNT MAY be established by an initial value provided in the Constraint element defined for the data item.</td>
</tr>
<tr>
<td>CHUCK_INTERLOCK</td>
<td>An indication of the state of an interlock function or control logic state intended to prevent the associated CHUCK component from being operated. The Valid Data Value MUST be ACTIVE or INACTIVE.</td>
</tr>
<tr>
<td>MANUAL_UNCLAMP</td>
<td>An indication of the state of an operator controlled interlock that can inhibit the ability to initiate an unclamp action of an electronically controlled chuck. The Valid Data Value MUST be ACTIVE or INACTIVE. When MANUAL_UNCLAMP is ACTIVE, it is expected that a chuck cannot be unclamped until MANUAL_UNCLAMP is set to INACTIVE.</td>
</tr>
<tr>
<td>DataItem type subType</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| CHUCK_STATE           | An indication of the operating state of a mechanism that holds a part or stock material during a manufacturing process. It may also represent a mechanism that holds any other mechanism in place within a piece of equipment.  
The **Valid Data Value** **MUST** be **OPEN**, **CLOSED**, or **UNLATCHED**. |
| CODE                  | DEPRECATED in Version 1.1. |
| COMPOSITION_STATE     | An indication of the operating condition of a mechanism represented by a Composition type element.  
A subType **MUST** always be specified.  
A compositionId **MUST** always be specified. |
| ACTION                | An indication of the operating state of a mechanism represented by a Composition type component.  
The operating state indicates whether the Composition element is activated or disabled.  
The **Valid Data Value** **MUST** be **ACTIVE** or **INACTIVE**. |
| LATERAL               | An indication of the position of a mechanism that may move in a lateral direction. The mechanism is represented by a Composition type component.  
The position information indicates whether the Composition element is positioned to the right, to the left, or is in transition.  
The **Valid Data Value** **MUST** be **RIGHT**, **LEFT**, **TRANSITIONING**. |
### DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MOTION</strong></td>
<td>An indication of the open or closed state of a mechanism. The mechanism is represented by a Composition type component. The operating state indicates whether the state of the Composition element is open, closed, or unlatched. The <strong>Valid Data Value</strong> MUST be OPEN, UNLATCHED, or CLOSED.</td>
</tr>
<tr>
<td><strong>SWITCHED</strong></td>
<td>An indication of the activation state of a mechanism represented by a Composition type component. The activation state indicates whether the Composition element is activated or not. The <strong>Valid Data Value</strong> MUST be ON or OFF.</td>
</tr>
<tr>
<td><strong>VERTICAL</strong></td>
<td>An indication of the position of a mechanism that may move in a vertical direction. The mechanism is represented by a Composition type component. The position information indicates whether the Composition element is positioned to the top, to the bottom, or is in transition. The <strong>Valid Data Value</strong> MUST be UP, DOWN, or TRANSITIONING.</td>
</tr>
<tr>
<td><strong>CONTROLLER_MODE</strong></td>
<td>The current mode of the Controller component. The <strong>Valid Data Value</strong> MUST be AUTOMATIC, MANUAL, MANUAL_DATA_INPUT, SEMI_AUTOMATIC, or EDIT.</td>
</tr>
<tr>
<td><strong>CONTROLLER_MODE_OVERRIDE</strong></td>
<td>A setting or operator selection that changes the behavior of a piece of equipment. A subType MUST always be specified.</td>
</tr>
<tr>
<td>DataItem type subType</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DRY_RUN</td>
<td>A setting or operator selection used to execute a test mode to confirm the execution of machine functions. The Valid Data Value MUST be ON or OFF. When DRY_RUN is ON, the equipment performs all of its normal functions, except no part or product is produced. If the equipment has a spindle, spindle operation is suspended.</td>
</tr>
<tr>
<td>MACHINE_AXIS_LOCK</td>
<td>A setting or operator selection that changes the behavior of the controller on a piece of equipment. The Valid Data Value MUST be ON or OFF. When MACHINE_AXIS_LOCK is ON, program execution continues normally, but no equipment motion occurs</td>
</tr>
<tr>
<td>OPTIONAL_STOP</td>
<td>A setting or operator selection that changes the behavior of the controller on a piece of equipment. The Valid Data Value MUST be ON or OFF. The program execution is stopped after a specific program block is executed when OPTIONAL_STOP is ON. In the case of a G-Code program, a program BLOCK containing a M01 code designates the command for an OPTIONAL_STOP. EXECUTION MUST change to OPTIONAL_STOP after a program block specifying an optional stop is executed and the OPTIONAL_STOP selection is ON.</td>
</tr>
</tbody>
</table>
### Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SINGLE_BLOCK</td>
<td>A setting or operator selection that changes the behavior of the controller on a piece of equipment. <strong>The [Valid Data Value] MUST be ON or OFF.</strong> Program execution is paused after each BLOCK of code is executed when SINGLE_BLOCK is ON.</td>
</tr>
<tr>
<td>TOOL_CHANGE_STOP</td>
<td>A setting or operator selection that changes the behavior of the controller on a piece of equipment. <strong>The [Valid Data Value] MUST be ON or OFF.</strong> Program execution is paused when a command is executed requesting a cutting tool to be changed.</td>
</tr>
<tr>
<td>COUPLED_AXES</td>
<td>Refers to the set of associated axes. <strong>The [Valid Data Value] for COUPLED_AXES SHOULD be a space-delimited set of axes reported as the value of the name attribute for each axis. If name is not available, the piece of equipment MUST report the value of the nativeName attribute for each axis.</strong></td>
</tr>
<tr>
<td>DATE_CODE</td>
<td>The time and date code associated with a material or other physical item. <strong>DATE_CODE MUST be reported in ISO 8601 format.</strong></td>
</tr>
</tbody>
</table>
## DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MANUFACTURE</td>
<td>The time and date code relating to the production of a material or other physical item.</td>
</tr>
<tr>
<td>EXPIRATION</td>
<td>The time and date code relating to the expiration or end of useful life for a material or other physical item.</td>
</tr>
<tr>
<td>FIRST_USE</td>
<td>The time and date code relating the first use of a material or other physical item.</td>
</tr>
<tr>
<td>DEVICE_UUID</td>
<td>The identifier of another piece of equipment that is temporarily associated with a component of this piece of equipment to perform a particular function. The <strong>Valid Data Value</strong> MUST be a <code>NMTOKEN</code> XML type.</td>
</tr>
<tr>
<td>DIRECTION</td>
<td>The direction of motion. A subType MUST always be specified</td>
</tr>
<tr>
<td>LINEAR</td>
<td>The direction of linear motion. The <strong>Valid Data Value</strong> MUST be <code>POSTIVE</code>, <code>NEGATIVE</code>, or <code>NONE</code>.</td>
</tr>
<tr>
<td>ROTARY</td>
<td>The direction of rotary motion using the right-hand rule convention. The <strong>Valid Data Value</strong> MUST be <code>CLOCKWISE</code>, <code>COUNTER_CLOCKWISE</code>, or <code>NONE</code>.</td>
</tr>
<tr>
<td>DOOR_STATE</td>
<td>The operational state of a DOOR type component or composition element. The <strong>Valid Data Value</strong> MUST be <code>OPEN</code>, <code>UNLATCHED</code>, or <code>CLOSED</code>.</td>
</tr>
<tr>
<td>DataItem type subType</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>EMERGENCY_STOP</strong></td>
<td>The current state of the emergency stop signal for a piece of equipment, controller path, or any other component or subsystem of a piece of equipment. The <em>Valid Data Value</em> <strong>MUST</strong> be <strong>ARMED</strong> (the circuit is complete and the device is allowed to operate) or <strong>TRIGGERED</strong> (the circuit is open and the device must cease operation).</td>
</tr>
<tr>
<td><strong>END_OF_BAR</strong></td>
<td>An indication of whether the end of a piece of bar stock being feed by a bar feeder has been reached. The <em>Valid Data Value</em> <strong>MUST</strong> be expressed as a Boolean expression of <strong>YES</strong> or <strong>NO</strong>.</td>
</tr>
<tr>
<td><strong>AUXILIARY</strong></td>
<td>When multiple locations on a piece of bar stock are referenced as the indication for the <strong>END_OF_BAR</strong>, the additional location(s) <strong>MUST</strong> be designated as <strong>AUXILIARY</strong> indication(s) for the <strong>END_OF_BAR</strong>.</td>
</tr>
<tr>
<td><strong>PRIMARY</strong></td>
<td>Specific applications <strong>MAY</strong> reference one or more locations on a piece of bar stock as the indication for the <strong>END_OF_BAR</strong>. The main or most important location <strong>MUST</strong> be designated as the <strong>PRIMARY</strong> indication for the <strong>END_OF_BAR</strong>. If no subType is specified, <strong>PRIMARY</strong> <strong>MUST</strong> be the default <strong>END_OF_BAR</strong> indication.</td>
</tr>
<tr>
<td><strong>EQUIPMENT_MODE</strong></td>
<td>An indication that a piece of equipment, or a sub-part of a piece of equipment, is performing specific types of activities. <strong>EQUIPMENT_MODE</strong> <strong>MAY</strong> have more than one subtype defined. A subType <strong>MUST</strong> always be specified.</td>
</tr>
<tr>
<td><strong>DELAY</strong></td>
<td>An indication that a piece of equipment is waiting for an event or an action to occur.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
</table>
| LOADED                | An indication that the sub-parts of a piece of equipment are under load.  
Example: For traditional machine tools, this is an indication that the cutting tool is assumed to be engaged with the part.  
The [Valid Data Value] MUST be ON or OFF. |
| OPERATING             | An indication that the major sub-parts of a piece of equipment are powered or performing any activity whether producing a part or product or not.  
Example: For traditional machine tools, this includes when the piece of equipment is WORKING or it is idle.  
The [Valid Data Value] MUST be ON or OFF. |
| POWERED               | An indication that primary power is applied to the piece of equipment and, as a minimum, the controller or logic portion of the piece of equipment is powered and functioning or components that are required to remain on are powered.  
Example: Heaters for an extrusion machine that required to be powered even when the equipment is turned off.  
The [Valid Data Value] MUST be ON or OFF. |
| WORKING               | An indication that a piece of equipment is performing any activity the equipment is active and performing a function under load or not.  
Example: For traditional machine tools, this includes when the piece of equipment is LOADED, making rapid moves, executing a tool change, etc.  
The [Valid Data Value] MUST be ON or OFF. |
## Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXECUTION</td>
<td>The execution status of the component.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Valid Data Value MUST be</strong> READY, ACTIVE, INTERRUPTED, WAIT, FEED_HOLD, STOPPED, OPTIONAL_STOP, PROGRAM_STOPPED, or PROGRAM_COMPLETED.</td>
</tr>
<tr>
<td>FIRMWARE</td>
<td>The embedded software of a component.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Valid Data Value MUST be</strong> a text string.</td>
</tr>
<tr>
<td></td>
<td>A subType <strong>MUST</strong> always be specified.</td>
</tr>
<tr>
<td>LICENSE</td>
<td>The license code to validate or activate the hardware or software.</td>
</tr>
<tr>
<td>VERSION</td>
<td>The version of the hardware or software.</td>
</tr>
<tr>
<td>RELEASE_DATE</td>
<td>The date the hardware or software was released for general use.</td>
</tr>
<tr>
<td>INSTALL_DATE</td>
<td>The date the hardware or software was installed.</td>
</tr>
<tr>
<td>MANUFACTURER</td>
<td>The corporate identity for the maker of the hardware or software.</td>
</tr>
<tr>
<td>FUNCTIONAL_MODE</td>
<td>The current intended production status of the device or component.</td>
</tr>
<tr>
<td></td>
<td>Typically, the FUNCTIONAL_MODE <strong>SHOULD be</strong> modeled as a data item for the Device element, but <strong>MAY be</strong> modeled for any Structural Element in the XML document.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Valid Data Value MUST be</strong> PRODUCTION, SETUP, TEARDOWN, MAINTENANCE, or PROCESS_DEVELOPMENT.</td>
</tr>
<tr>
<td>HARDNESS</td>
<td>The measurement of the hardness of a material.</td>
</tr>
<tr>
<td></td>
<td>The measurement does not provide a unit.</td>
</tr>
<tr>
<td></td>
<td>A subType <strong>MUST</strong> always be specified to designate the hardness scale associated with the measurement.</td>
</tr>
</tbody>
</table>
### Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BRINELL</strong></td>
<td>A scale to measure the resistance to deformation of a surface.</td>
</tr>
<tr>
<td><strong>LEEB</strong></td>
<td>A scale to measure the elasticity of a surface.</td>
</tr>
<tr>
<td><strong>MOHS</strong></td>
<td>A scale to measure the resistance to scratching of a surface.</td>
</tr>
<tr>
<td><strong>ROCKWELL</strong></td>
<td>A scale to measure the resistance to deformation of a surface.</td>
</tr>
<tr>
<td><strong>SHORE</strong></td>
<td>A scale to measure the resistance to deformation of a surface.</td>
</tr>
<tr>
<td><strong>VICKERS</strong></td>
<td>A scale to measure the resistance to deformation of a surface.</td>
</tr>
<tr>
<td><strong>HARDWARE</strong></td>
<td>The hardware of a component. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string. A subType <strong>MUST</strong> always be specified.</td>
</tr>
<tr>
<td><strong>LICENSE</strong></td>
<td>The license code to validate or activate the hardware or software.</td>
</tr>
<tr>
<td><strong>VERSION</strong></td>
<td>The version of the hardware or software.</td>
</tr>
<tr>
<td><strong>RELEASE_DATE</strong></td>
<td>The date the hardware or software was released for general use.</td>
</tr>
<tr>
<td><strong>INSTALL_DATE</strong></td>
<td>The date the hardware or software was installed.</td>
</tr>
<tr>
<td><strong>MANUFACTURER</strong></td>
<td>The corporate identity for the maker of the hardware or software.</td>
</tr>
<tr>
<td><strong>INTERFACE_STATE</strong></td>
<td>The current functional or operational state of an Interface type element indicating whether the interface is active or is not currently functioning. The <strong>Valid Data Value</strong> <strong>MUST</strong> be ENABLED or DISABLED.</td>
</tr>
<tr>
<td><strong>LIBRARY</strong></td>
<td>The software library on a component. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string. A subType <strong>MUST</strong> always be specified.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LICENSE</td>
<td>The license code to validate or activate the hardware or software.</td>
</tr>
<tr>
<td>VERSION</td>
<td>The version of the hardware or software.</td>
</tr>
<tr>
<td>RELEASE_DATE</td>
<td>The date the hardware or software was released for general use.</td>
</tr>
<tr>
<td>INSTALL_DATE</td>
<td>The date the hardware or software was installed.</td>
</tr>
<tr>
<td>MANUFACTURER</td>
<td>The corporate identity for the maker of the hardware or software.</td>
</tr>
<tr>
<td>LINE</td>
<td>The current line of code being executed. The data will be an alpha numeric value representing the line number of the current line of code being executed. <strong>DEPRECATED</strong> in Version 1.4.0.</td>
</tr>
<tr>
<td>MAXIMUM</td>
<td>The maximum line number of the code being executed.</td>
</tr>
<tr>
<td>MINIMUM</td>
<td>The minimum line number of the code being executed.</td>
</tr>
<tr>
<td>LINE_LABEL</td>
<td>An optional identifier for a BLOCK of code in a PROGRAM.</td>
</tr>
<tr>
<td>LINE_NUMBER</td>
<td>A reference to the position of a block of program code within a control program. The line number <strong>MAY</strong> represent either an absolute position starting with the first line of the program or an incremental position relative to the occurrence of the last LINE_LABEL. <strong>LINE_NUMBER</strong> does not change subject to any looping or branching in a control program. A <strong>subType</strong> <strong>MUST</strong> be defined.</td>
</tr>
<tr>
<td>ABSOLUTE</td>
<td>The position of a block of program code relative to the beginning of the control program.</td>
</tr>
<tr>
<td>INCREMENTAL</td>
<td>The position of a block of program code relative to the occurrence of the last LINE_LABEL encountered in the control program.</td>
</tr>
</tbody>
</table>
### Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MATERIAL</td>
<td>The identifier of a material used or consumed in the manufacturing process. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string.</td>
</tr>
<tr>
<td>MATERIAL_LAYER</td>
<td>Identifies the layers of material applied to a part or product as part of an additive manufacturing process. The <strong>Valid Data Value</strong> <strong>MUST</strong> be an integer.</td>
</tr>
<tr>
<td>ACTUAL</td>
<td>The current number of layers of material applied to a part or product during an additive manufacturing process.</td>
</tr>
<tr>
<td>TARGET</td>
<td>The target or planned number layers of material applied to a part or product during an additive manufacturing process.</td>
</tr>
<tr>
<td>MESSAGE</td>
<td>Any text string of information to be transferred from a piece of equipment to a client software application.</td>
</tr>
<tr>
<td>NETWORK</td>
<td>Network details of a component. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string. A <strong>subType</strong> <strong>MUST</strong> always be specified. If the <strong>subType</strong> is <strong>WIRELESS</strong>, the <strong>Valid Data Value</strong> <strong>MUST</strong> be <strong>YES</strong> or <strong>NO</strong>.</td>
</tr>
<tr>
<td>IPV4_ADDRESS</td>
<td>The IPV4 network address of the component.</td>
</tr>
<tr>
<td>IPV6_ADDRESS</td>
<td>The IPV6 network address of the component.</td>
</tr>
<tr>
<td>GATEWAY</td>
<td>The Gateway for the component network.</td>
</tr>
<tr>
<td>SUBNET_MASK</td>
<td>The SubNet mask for the component network.</td>
</tr>
<tr>
<td>VLAN_ID</td>
<td>The layer2 Virtual Local Network (VLAN) ID for the component network.</td>
</tr>
<tr>
<td>MAC_ADDRESS</td>
<td>Media Access Control Address. The unique physical address of the network hardware.</td>
</tr>
<tr>
<td>WIRELESS</td>
<td>Identifies whether the connection type is wireless.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPERATING_SYSTEM</td>
<td>The Operating System of a component. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string. A <strong>subType</strong> <strong>MUST</strong> always be specified.</td>
</tr>
<tr>
<td>LICENSE</td>
<td>The license code to validate or activate the hardware or software.</td>
</tr>
<tr>
<td>VERSION</td>
<td>The version of the hardware or software.</td>
</tr>
<tr>
<td>RELEASE_DATE</td>
<td>The date the hardware or software was released for general use.</td>
</tr>
<tr>
<td>INSTALL_DATE</td>
<td>The date the hardware or software was installed.</td>
</tr>
<tr>
<td>MANUFACTURER</td>
<td>The corporate identity for the maker of the hardware or software.</td>
</tr>
<tr>
<td>OPERATOR_ID</td>
<td>The identifier of the person currently responsible for operating the piece of equipment. <strong>DEPRECIATION WARNING</strong>: May be deprecated in the future. See USER below.</td>
</tr>
<tr>
<td>PALLET_ID</td>
<td>The identifier for a pallet. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string.</td>
</tr>
<tr>
<td>PART_COUNT</td>
<td>The aggregate count of parts. Use the discrete attribute with value <strong>true</strong> to report non-aggregate part count. See Section <a href="#">7.2.3.5 - ResetTrigger Element for DataItem</a> to reset the count. The <strong>Valid Data Value</strong> <strong>MUST</strong> be numeric.</td>
</tr>
<tr>
<td>ALL</td>
<td>The number of parts produced. <strong>ALL</strong> is the default <strong>subType</strong>.</td>
</tr>
<tr>
<td>BAD</td>
<td>The number of parts produced that do not conform to specification.</td>
</tr>
<tr>
<td>GOOD</td>
<td>The number of parts produced that conform to specification.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REMAINING</td>
<td>The number of remaining or in-stock parts to be produced.</td>
</tr>
<tr>
<td>TARGET</td>
<td>The number of projected or planned parts to be produced.</td>
</tr>
<tr>
<td>PART_DETECT</td>
<td>An indication designating whether a part or work piece has been detected or is present. The Valid Data Value MUST be PRESENT or NOT_PRESENT.</td>
</tr>
<tr>
<td>PART_ID</td>
<td>An identifier of a part in a manufacturing operation. The Valid Data Value MUST be a text string.</td>
</tr>
<tr>
<td>PART_NUMBER</td>
<td>An identifier of a part or product moving through the manufacturing process. The Valid Data Value MUST be a text string. DEPRECATED WARNING: May be deprecated in the future.</td>
</tr>
<tr>
<td>DataItem type subType</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PATH_FEEDRATE_OVERRIDE</td>
<td>The value of a signal or calculation issued to adjust the feedrate for the axes associated with a Path component that may represent a single axis or the coordinated movement of multiple axes. The value provided for PATH_FEEDRATE_OVERRIDE is expressed as a percentage of the designated feedrate for the path. When PATH_FEEDRATE_OVERRIDE is applied, the resulting commanded feedrate for the path is limited to the value of the original feedrate multiplied by the value of the PATH_FEEDRATE_OVERRIDE. There MAY be different subtypes of PATH_FEEDRATE_OVERRIDE; each representing an override value for a designated subtype of feedrate depending on the state of operation of the path. The states of operation of a path are currently defined as PROGRAMMED, JOG, and RAPID.</td>
</tr>
<tr>
<td>JOG</td>
<td>The value of a signal or calculation issued to adjust the feedrate of the axes associated with a Path component when the axes, or a single axis, are being operated in a manual mode or method (jogging). When the JOG subtype of PATH_FEEDRATE_OVERRIDE is applied, the resulting commanded feedrate for the axes, or a single axis, associated with the path are limited to the value of the original JOG subtype of the PATH_FEEDRATE multiplied by the value of the JOG subtype of PATH_FEEDRATE_OVERRIDE.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROGRAMMED</td>
<td>The value of a signal or calculation issued to adjust the feedrate of the axes associated with a Path component when the axes, or a single axis, are operating as specified by a logic or motion program or set by a switch. When the PROGRAMMED subtype of PATH_FEEDRATE OVERRIDE is applied, the resulting commanded feedrate for the axes, or a single axis, associated with the path are limited to the value of the original PROGRAMMED subtype of the PATH_FEEDRATE multiplied by the value of the PROGRAMMED subtype of PATH_FEEDRATE OVERRIDE.</td>
</tr>
<tr>
<td>RAPID</td>
<td>The value of a signal or calculation issued to adjust the feedrate of the axes associated with a Path component when the axes, or a single axis, are being operated in a rapid positioning mode or method (rapid). When the RAPID subtype of PATH_FEEDRATE OVERRIDE is applied, the resulting commanded feedrate for the axes, or a single axis, associated with the path are limited to the value of the original RAPID subtype of the PATH_FEEDRATE multiplied by the value of the RAPID subtype of PATH_FEEDRATE_OVERRIDE.</td>
</tr>
<tr>
<td>PATH_MODE</td>
<td>Describes the operational relationship between a Path Structural Element and another Path Structural Element for pieces of equipment comprised of multiple logical groupings of controlled axes or other logical operations. The Valid Data Value MUST be INDEPENDENT, MASTER, SYNCHRONOUS, or MIRROR. The default value MUST be INDEPENDENT if PATH_MODE is not specified.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POWER_STATE</td>
<td>The indication of the status of the source of energy for a Structural Element to allow it to perform its intended function or the state of an enabling signal providing permission for the Structural Element to perform its functions. The Valid Data Value MUST be ON or OFF. <strong>DEPRECATION WARNING</strong>: May be deprecated in the future.</td>
</tr>
<tr>
<td>CONTROL</td>
<td>The state of the enabling signal or control logic that enables or disables the function or operation of the Structural Element.</td>
</tr>
<tr>
<td>LINE</td>
<td>The state of the power source for the Structural Element.</td>
</tr>
<tr>
<td>POWER_STATUS</td>
<td><strong>DEPRECATED</strong> in Version 1.1.0.</td>
</tr>
<tr>
<td>PROCESS_TIME</td>
<td>The time and date associated with an activity or event. PROCESS_TIME MUST be reported in ISO 8601 format.</td>
</tr>
<tr>
<td>START</td>
<td>The time and date associated with the beginning of an activity or event.</td>
</tr>
<tr>
<td>COMPLETE</td>
<td>The time and date associated with the completion of an activity or event.</td>
</tr>
<tr>
<td>TARGET_COMPLETION</td>
<td>The projected time and date associated with the end or completion of an activity or event.</td>
</tr>
<tr>
<td>PROGRAM</td>
<td>The identity of the logic or motion program being executed by the piece of equipment. The Valid Data Value MUST be a text string.</td>
</tr>
<tr>
<td>SCHEDULE</td>
<td>The identity of a control program that is used to specify the order of execution of other programs.</td>
</tr>
<tr>
<td>DataItem type subType</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>MAIN</td>
<td>The identity of the primary logic or motion program currently being executed. It is the starting nest level in a call structure and may contain calls to sub programs.</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>The identity of the logic or motion program currently executing.</td>
</tr>
<tr>
<td>PROGRAM_COMMENT</td>
<td>A comment or non-executable statement in the control program. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string.</td>
</tr>
<tr>
<td>SCHEDULE</td>
<td>The identity of a control program that is used to specify the order of execution of other programs.</td>
</tr>
<tr>
<td>MAIN</td>
<td>The identity of the primary logic or motion program currently being executed. It is the starting nest level in a call structure and may contain calls to sub programs.</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>The identity of the logic or motion program currently executing.</td>
</tr>
<tr>
<td>PROGRAM_EDIT</td>
<td>An indication of the status of the Controller components program editing mode. On many controls, a program can be edited while another program is currently being executed. The <strong>Valid Data Value</strong> <strong>MUST</strong> be:</td>
</tr>
<tr>
<td></td>
<td>ACTIVE: The controller is in the program edit mode.</td>
</tr>
<tr>
<td></td>
<td>READY: The controller is capable of entering the program edit mode and no function is inhibiting a change of mode.</td>
</tr>
<tr>
<td></td>
<td>NOT_READY: A function is inhibiting the controller from entering the program edit mode.</td>
</tr>
<tr>
<td>DataItem type subType</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PROGRAM_EDIT_NAME</td>
<td>The name of the program being edited. This is used in conjunction with</td>
</tr>
<tr>
<td></td>
<td>PROGRAM_EDIT when in ACTIVE state.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string.</td>
</tr>
<tr>
<td>PROGRAM_HEADER</td>
<td>The non-executable header section of the control program.</td>
</tr>
<tr>
<td></td>
<td>If not specified, the default <strong>subType</strong> is <strong>MAIN</strong>.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string.</td>
</tr>
<tr>
<td>SCHEDULE</td>
<td>The identity of a control program that is used to specify the order of</td>
</tr>
<tr>
<td></td>
<td>execution of other programs.</td>
</tr>
<tr>
<td>MAIN</td>
<td>The identity of the primary logic or motion program currently being executed.</td>
</tr>
<tr>
<td></td>
<td>It is the starting nest level in a call structure and may contain calls to</td>
</tr>
<tr>
<td></td>
<td>sub programs.</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>The identity of the logic or motion program currently executing.</td>
</tr>
<tr>
<td>PROGRAM_LOCATION</td>
<td>The Uniform Resource Identifier (URI) for the source file associated with</td>
</tr>
<tr>
<td></td>
<td>PROGRAM.</td>
</tr>
<tr>
<td>SCHEDULE</td>
<td>An identity of a control program that is used to specify the order of</td>
</tr>
<tr>
<td></td>
<td>execution of other programs.</td>
</tr>
<tr>
<td>MAIN</td>
<td>The identity of the primary logic or motion program currently being executed.</td>
</tr>
<tr>
<td></td>
<td>It is the starting nest level in a call structure and may contain calls to</td>
</tr>
<tr>
<td></td>
<td>sub programs.</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>The identity of the logic or motion program currently executing.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROGRAM_LOCATION_TYPE</td>
<td>Defines whether the logic or motion program defined by PROGRAM is being executed from the local memory of the controller or from an outside source. The Valid Data Value MUST be LOCAL or EXTERNAL.</td>
</tr>
<tr>
<td>SCHEDULE</td>
<td>An identity of a control program that is used to specify the order of execution of other programs.</td>
</tr>
<tr>
<td>MAIN</td>
<td>The identity of the primary logic or motion program currently being executed. It is the starting nest level in a call structure and may contain calls to sub programs.</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>The identity of the logic or motion program currently executing.</td>
</tr>
<tr>
<td>PROGRAM_NEST_LEVEL</td>
<td>An indication of the nesting level within a control program that is associated with the code or instructions that is currently being executed. If an initial value is not defined, the nesting level associated with the highest or initial nesting level of the program MUST default to zero (0). The value reported for PROGRAM_NEST_LEVEL MUST be an integer.</td>
</tr>
<tr>
<td>ROTARY_MODE</td>
<td>The current operating mode for a Rotary type axis.</td>
</tr>
<tr>
<td></td>
<td>The Valid Data Value MUST be SPINDLE, INDEX, or CONTOUR.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROTARY VELOCITY OVERRIDE</td>
<td>The value of a command issued to adjust the programmed velocity for a Rotary type axis. This command represents a percentage change to the velocity calculated by a logic or motion program or set by a switch for a Rotary type axis. ROTARY_VELOCITY_OVERRIDE is expressed as a percentage of the programmed ROTARY_VELOCITY.</td>
</tr>
<tr>
<td>ROTATION</td>
<td>A three space angular rotation relative to a coordinate system. When the DataItem has a coordinateSystemIdRef attribute and the CoordinateSystem does not specify a Rotation, the value of the observation is the rotation of the the referenced CoordinateSystem. The units MUST be DEGREE_3D.</td>
</tr>
<tr>
<td>SERIAL_NUMBER</td>
<td>The serial number associated with a Component, Asset, or Device. The Valid Data Value MUST be a text string.</td>
</tr>
<tr>
<td>SPINDLE_INTERLOCK</td>
<td>An indication of the status of the spindle for a piece of equipment when power has been removed and it is free to rotate. The Valid Data Value MUST be: ACTIVE if power has been removed and the spindle cannot be operated. INACTIVE if power to the spindle has not been deactivated.</td>
</tr>
<tr>
<td>TOOL_ASSET_ID</td>
<td>The identifier of an individual tool asset. The Valid Data Value MUST be a text string.</td>
</tr>
<tr>
<td>DataItem type subType</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>TOOL_GROUP</td>
<td>An identifier for the tool group associated with a specific tool. Commonly used to designate spare tools.</td>
</tr>
<tr>
<td>TOOL_ID</td>
<td><strong>DEPRECATED</strong> in Version 1.2.0. See TOOL_ASSET_ID. The identifier of the tool currently in use for a given Path.</td>
</tr>
<tr>
<td>TOOL_NUMBER</td>
<td>The identifier assigned by the Controller component to a cutting tool when in use by a piece of equipment. The <strong>Valid Data Value</strong> MUST be a text string.</td>
</tr>
<tr>
<td>TOOL_OFFSET</td>
<td>A reference to the tool offset variables applied to the active cutting tool. The <strong>Valid Data Value</strong> MUST be a text string. The reported value returned for TOOL_OFFSET identifies the location in a table or list where the actual tool offset values are stored. <strong>DEPRECATED</strong> in V1.5 A subType MUST always be specified.</td>
</tr>
<tr>
<td>LENGTH</td>
<td>A reference to a length type tool offset.</td>
</tr>
<tr>
<td>RADIAL</td>
<td>A reference to a radial type tool offset.</td>
</tr>
<tr>
<td>TRANSLATION</td>
<td>A three space linear translation relative to a coordinate system. <strong>When the DataItem has a coordinateSystemIdRef attribute and the CoordinateSystem does not specify a Translation, the value of the observation is the translation of the referenced CoordinateSystem.</strong> The units <strong>MUST</strong> be MILLIMETER_3D</td>
</tr>
</tbody>
</table>
## DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER</td>
<td>The identifier of the person currently responsible for operating the piece of equipment. A subType <strong>MUST</strong> always be specified.</td>
</tr>
<tr>
<td>MAINTENANCE</td>
<td>The identifier of the person currently responsible for performing maintenance on the piece of equipment.</td>
</tr>
<tr>
<td>OPERATOR</td>
<td>The identifier of the person currently responsible for operating the piece of equipment.</td>
</tr>
<tr>
<td>SET_UP</td>
<td>The identifier of the person currently responsible for preparing a piece of equipment for production or restoring the piece of equipment to a neutral state after production.</td>
</tr>
<tr>
<td>VARIABLE</td>
<td>A data value whose meaning may change over time due to changes in the operation of a piece of equipment or the process being executed on that piece of equipment.</td>
</tr>
<tr>
<td>WAIT_STATE</td>
<td>An indication of the reason that EXECUTION is reporting a value of WAIT. The <strong>Valid Data Value</strong> MUST be POWERING_UP, POWERING_DOWN, PART_LOAD, PART_UNLOAD, TOOL_LOAD, TOOL_UNLOAD, MATERIAL_LOAD, MATERIAL_UNLOAD, SECONDARY_PROCESS, PAUSING, or RESUMING.</td>
</tr>
<tr>
<td>WIRE</td>
<td>The identifier for the type of wire used as the cutting mechanism in Electrical Discharge Machining or similar processes. The <strong>Valid Data Value</strong> MUST be a text string.</td>
</tr>
</tbody>
</table>
Continuation of Table 41: DataItem type subType for category EVENT

<table>
<thead>
<tr>
<th>DataItem type subType</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORKHOLDING_ID</td>
<td>The identifier for the current workholding or part clamp in use by a piece of equipment. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string.</td>
</tr>
<tr>
<td>WORK_OFFSET</td>
<td>A reference to the offset variables for a work piece or part associated with a Path in a Controller type component. The <strong>Valid Data Value</strong> <strong>MUST</strong> be a text string. The reported value returned for WORK_OFFSET identifies the location in a table or list where the actual work offset values are stored.</td>
</tr>
</tbody>
</table>
8.3 Data Items in category CONDITION

CONDITION category data items report data representing a Structural Element's status regarding its ability to operate or it provides an indication whether the data reported for the Structural Element is within an expected range.

CONDITION is reported differently than SAMPLE or EVENT. CONDITION MUST be reported as Normal, Warning, or Fault.

All DataItem types in the SAMPLE category MAY have associated CONDITION states. CONDITION states indicate whether the value for the data is within an expected range and MUST be reported as Normal, or the value is unexpected or out of tolerance for the data and a Warning or Fault MUST be provided.

Some DataItem types in the EVENT category MAY have associated CONDITION states.

Additional CONDITION types are provided to represent the health and fault status of Structural Elements. Table 42 defines these additional DataItem types.

CONDITION type data items are unlike other data item types since they MAY have multiple concurrently active values at any point in time.

### Table 42: DataItem type for category CONDITION

<table>
<thead>
<tr>
<th>DataItem type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTUATOR</td>
<td>An indication of a fault associated with an actuator.</td>
</tr>
<tr>
<td>CHUCK_INTERLOCK</td>
<td>An indication of the operational condition of the interlock function for an electronically controller chuck.</td>
</tr>
<tr>
<td>COMMUNICATIONS</td>
<td>An indication that the piece of equipment has experienced a communications failure.</td>
</tr>
<tr>
<td>DATA_RANGE</td>
<td>An indication that the value of the data associated with a measured value or a calculation is outside of an expected range.</td>
</tr>
<tr>
<td>DIRECTION</td>
<td>An indication of a fault associated with the direction of motion of a Structural Element</td>
</tr>
<tr>
<td>END_OF_BAR</td>
<td>An indication that the end of a piece of bar stock has been reached.</td>
</tr>
<tr>
<td>HARDWARE</td>
<td>An indication of a fault associated with the hardware subsystem of the Structural Element</td>
</tr>
</tbody>
</table>
### Continuation of Table 42

<table>
<thead>
<tr>
<th>DataItem type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTERFACE_STATE</td>
<td>An indication of the operation condition of an Interface component.</td>
</tr>
<tr>
<td>LOGIC_PROGRAM</td>
<td>An indication that an error occurred in the logic program or programmable logic controller (PLC) associated with a piece of equipment.</td>
</tr>
<tr>
<td>MOTION_PROGRAM</td>
<td>An indication that an error occurred in the motion program associated with a piece of equipment.</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>An indication of a fault associated with a piece of equipment or component that cannot be classified as a specific type.</td>
</tr>
</tbody>
</table>
Configuration contains technical information about a component describing its physical layout, functional characteristics, and relationships with other components within a piece of equipment.

Table 43 lists the types of Configuration defined for a Component.

**Table 43: Types of Configuration**

<table>
<thead>
<tr>
<th>type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationships</td>
<td>Relationships organizes Relationship elements for a Component.</td>
</tr>
</tbody>
</table>
9.1 Sensor

Sensor is a unique type of a piece of equipment. A Sensor is typically comprised of two major components: a sensor unit that provides signal processing, conversion, and communications and the sensing elements that provides a signal or measured value.

The sensor unit is modeled as a Lower Level Component called Sensor. The sensing element may be modeled as a Composition element of a Sensor element and the measured value would be modeled as a DataItem (See Section 8 - Listing of Data Items for more information on DataItem elements). Each sensor unit may have multiple sensing elements; each representing the data for a variety of measured values.

Example: A pressure transducer could be modeled as a Sensor (Component) with a name = Pressure Transducer B and its measured value could be modeled as a PRESSURE type DataItem.

While a Sensor may be modeled in the XML document in different ways, it will always be modeled to associate the information measured by each sensing element with the Structural Element to which the measured value is most closely associated.

9.1.1 Sensor Data

The most basic implementation of a sensor occurs when the sensing element itself is not identified in the data model, but the data that is measured by the sensing element is provided as a data item associated with a Component. An example would be the measured value of the temperature of a spindle motor. This would be represented as a DataItem called TEMPERATURE that is associated with the Rotary type axis element called "C" as shown in Example 7.
A sensor may measure values associated with any Component or Device element. Some examples of how sensor data may be modeled are represented in Figure 18:

**Figure 18: Sensor Data Associations**

**9.1.2 Sensor Unit**

A *sensor unit* is an intelligent piece of equipment that manages the functions of one or more *sensing elements*.

Typical functions of the *sensor unit* include:

- convert low level signals from the *sensing elements* into data that can be used by other pieces of equipment. (Example: Convert a non-linear millivolt signal from a temperature sensor into a scaled temperature value that can be transmitted to another piece of equipment.)
• process *sensing element* data into calculated values. (Example: temperature sensor data is converted into calculated values of average temperature, maximum temperature, minimum temperature, etc.)

• provide calibration and configuration information associated with each *sensing element*

• monitor the health and integrity of the *sensing elements* and the *sensor unit*. (Example: The *sensor unit* may provide diagnostics on each *sensing element* (e.g., open wire detection) and itself (e.g., measure internal temperature of the *sensor unit*).

Depending on how the *sensor unit* is used, it may be considered as either an independent piece of equipment and modeled in the **XML** document as a *Device*, or it may be modeled as a **Top Level** *Component* called *Sensor* if it is integral to a piece of equipment.

A *Sensor* MAY have its own *uuid* so it can be tracked throughout its lifetime.

The following examples demonstrate how a *Sensor* may be modeled in the **XML** document differently based on how the *Sensor* functions within the overall piece of equipment.

**Example#1:** If the *Sensor* provides vibration measurement data for the spindle on a piece of equipment, it could be modeled as a *Sensor* for rotary axis named C.

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

**Example#2:** If a *Sensor* provides measurement data for multiple *Component* elements within a piece of equipment and is not associated with any particular *Component* element, it MAY be modeled in the **XML** document as an independent **Lower Level** *Component*.

```xml
<Components>
  <Axes>
    <Rotary id="c" name="C">
      <Components>
        <Sensor id="spdlm" name="Spindlemonitor">
          <DataItems>
            <DataItem type="DISPLACEMENT" id="cvib" category="SAMPLE" name="Svib" units="MILLIMETER"/>
          </DataItems>
        </Sensor>
      </Components>
    </Rotary>
  </Axes>
</Components>
```
ponent and the data associated with measurements are associated with their associated
Component elements.

This example represents a sensor unit with two sensing elements, one measures spindle
vibration and the other measures the temperature for the X axis. The sensor unit also has
a sensing element measuring the internal temperature of the sensor unit.

**Example 9:** Example of Sensor Unit with Sensing Element

```xml
<Device id="d1" uuid="HM1" name="HMC_3Axis">
  <Description>3 Axis Mill</Description>
  <Components>
    <Axes>
      <Sensor id="sens1" name="Sensorunit">
        <DataItems>
          <DataItem type="TEMPERATURE" id="sentemp" category="SAMPLE" name="Sensortemp" units="DEGREE"/>
        </DataItems>
      </Sensor>
      <Rotary id="c" name="C">
        <DataItems>
          <DataItem type="DISPLACEMENT" id="cvib" category="SAMPLE" name="Svib" units="MILLIMETER"/>
          <Source componentId="sens1"/>
        </DataItems>
      </Rotary>
      <Linear id="x" name="X">
        <DataItems>
          <DataItem type="TEMPERATURE" id="xt" category="SAMPLE" name="Xtemp" units="DEGREE"/>
          <Source componentId="sens1"/>
        </DataItems>
      </Linear>
    </Axes>
  </Components>
</Device>
```

### 9.1.3 Sensor Configuration

When a Sensor unit is modeled in the XML document as a Component or as a separate	piece of equipment, it may provide additional configuration information for the sensor
elements and the sensor unit itself.

Configuration data provides information required for maintenance and support of the sensor.

Configuration data is only available when the Sensor unit is modeled as a Component or a separate piece of equipment. For details on the modeling of configuration data in the XML document, see Section 4.4.3.2 - Configuration for Component.

When Sensor represents the sensor unit for multiple sensing element(s), each sensing element is represented by a Channel. The sensor unit itself and each Channel representing one sensing element MAY have its own configuration data.

SensorConfiguration can contain any descriptive content for a sensor unit. This element is defined to contain mixed content and additional XML elements (indicated by the any element in Figure 19) MAY be added to extend the schema for SensorConfiguration.

Figure 19 represents the structure of the SensorConfiguration XML element showing the attributes defined for SensorConfiguration.
Figure 19: SensorConfiguration Diagram
**Table 44: MTConnect SensorConfiguration Element**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SensorConfiguration</strong></td>
<td>An element that can contain descriptive content defining the configuration information for Sensor. For Sensor, the valid configuration is SensorConfiguration which 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). MTConnect does not support all of the data represented in the TEDS data, nor does it duplicate the function of the TEDS data sheets.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**9.1.3.1 Elements for SensorConfiguration**

*Table 45* defines the configuration elements available for SensorConfiguration:

**Table 45: Elements for SensorConfiguration**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FirmwareVersion</strong></td>
<td>Version number for the sensor unit as specified by the manufacturer. FirmwareVersion is a required element if SensorConfiguration is used. The data value for FirmwareVersion is provided in the CDATA for this element and MAY be any numeric or text content.</td>
<td>1</td>
</tr>
</tbody>
</table>
### Continuation of Table 45

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalibrationDate</td>
<td>Date upon which the <em>sensor unit</em> was last calibrated.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>The data value for CalibrationDate is provided in the CDATA for this element and <strong>MUST</strong> be represented in the W3C ISO 8601 format.</td>
<td></td>
</tr>
<tr>
<td>NextCalibrationDate</td>
<td>Date upon which the <em>sensor unit</em> is next scheduled to be calibrated.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>The data value for NextCalibrationDate is provided in the CDATA for this element and <strong>MUST</strong> be represented in the W3C ISO 8601 format.</td>
<td></td>
</tr>
<tr>
<td>CalibrationInitials</td>
<td>The initials of the person verifying the validity of the calibration data.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>The data value for CalibrationInitials is provided in the CDATA for this element and <strong>MAY</strong> be any numeric or text content.</td>
<td></td>
</tr>
<tr>
<td>Channels</td>
<td><strong>When</strong> Sensor represents multiple <em>sensing elements</em>, each <em>sensing element</em> is represented by a Channel for the Sensor.</td>
<td>0..1</td>
</tr>
<tr>
<td></td>
<td>Channels is an XML container used to organize information for the <em>sensing elements</em>.</td>
<td></td>
</tr>
</tbody>
</table>

**9.1.3.1.1 Attributes for Channel**

Channel represents each *sensing element* connected to a *sensor unit*. Table 46 defines the attributes for Channel:
Table 46: Attributes for Channel

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
</table>
| number    | A unique identifier that will only refer to a specific *sensing element.*
|           | *number* *is a required attribute.*
|           | For example, this can be the manufacturer code and the serial number.
|           | *number* *SHOULD* be alphanumeric and not exceeding 255 characters.
|           | An [NMTOKEN XML](https://www.w3.org/TR/xml/#character-classes) type. |
| name      | The *name* of the *sensing element.*
|           | *name* is an optional attribute.
|           | *name* *SHOULD* be unique within the *sensor unit* to allow for easier data integration.
|           | An [NMTOKEN XML](https://www.w3.org/TR/xml/#character-classes) type. |

<table>
<thead>
<tr>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>0..1</td>
</tr>
</tbody>
</table>

9.1.3.1.2 Elements for Channel

Table 47 describes the elements provided for Channel.

Table 47: Elements for Channel

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>An [XML](<a href="https://www.w3.org/international/publications/">https://www.w3.org/international/publications/</a> morbids.html) element that can contain any descriptive content.</td>
</tr>
<tr>
<td></td>
<td>The <a href="https://www.w3.org/TR/xml/#cdatasection">CDATA</a> of <em>Description</em> <em>MAY</em> include any additional descriptive information the implementer chooses to include regarding a <em>sensor element.</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>0..1</td>
</tr>
</tbody>
</table>
Continuation of Table 47

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalibrationDate</td>
<td>Date upon which the sensor unit was last calibrated to the sensor element. The data value for CalibrationDate is provided in the CDATA for this element and MUST be represented in the W3C ISO 8601 format.</td>
<td>0..1</td>
</tr>
<tr>
<td>NextCalibrationDate</td>
<td>Date upon which the sensor element is next scheduled to be calibrated with the sensor unit. The data value for NextCalibrationDate is provided in the CDATA for this element and MUST be represented in the W3C ISO 8601 format.</td>
<td>0..1</td>
</tr>
<tr>
<td>CalibrationInitials</td>
<td>The initials of the person verifying the validity of the calibration data. The data value for CalibrationInitials is provided in the CDATA for this element and MAY be any numeric or text content.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Example 10 is an example of the configuration data for Sensor that is modeled as a Component. It has Configuration data for the sensor unit, one Channel named A/D:1, and two DataItems – Voltage (as a SAMPLE) and Voltage (as a CONDITION or alarm).

Example 10: Example of configuration data for Sensor

```
1406 1  <Sensor id="sensor" name="sensor">
1407 2  <Configuration>
1408 3  <SensorConfiguration>
1409 4  <FirmwareVersion>2.02</FirmwareVersion>
1410 5  <CalibrationDate>2010-05-16</CalibrationDate>
1411 6  <NextCalibrationDate>2010-05-16</NextCalibrationDate>
1412 7  <CalibrationInitials>WS</CalibrationInitials>
1413 8  <Channels>
1414 9  <Channel number="1" name="A/D:1">
1415 10 <Description>A/D With Thermister</Description>
1416 11 </Channel>
```
9.2 Relationships

Relationships is an XML container that organizes information defining the association between pieces of equipment that function independently but together perform a manufacturing operation. Relationships may also define the association between components within a piece of equipment.

Relationships may be modeled as part of a Device or a Component Structural Element.

Relationships contains one or more Relationship XML elements.

Table 48: MTConnect Relationships Element

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationships</td>
<td>XML container consisting of one or more Relationship XML elements. Only one Relationships container MUST appear for a Device or a Component element.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

9.2.1 Relationship

Relationship is an XML element that describes the association between two pieces of equipment that function independently but together perform a manufacturing operation. Relationship may also be used to define the association between two components within a piece of equipment.

Relationship is an abstract type XML element. Relationship will be replaced in
the XML document by specific Relationship types. XML elements representing Relationship are described in Section 9.2.1.1 - DeviceRelationship and Section 9.2.1.2 - ComponentRelationship.

A separate Relationship type element MAY be defined to describe each pair of associations with a piece of equipment or between Component elements within a piece of equipment.

Pieces of equipment may only be associated with other pieces of equipment and Component elements may only be associated with other Component elements within a specific piece of equipment.

The XML schema diagram in Figure 20 represents the structure of the Relationship XML element.
Figure 20: Relationship Diagram
9.2.1.1 DeviceRelationship

DeviceRelationship describes the association between two pieces of equipment that function independently but together perform a manufacturing operation.

The XML schema diagram in Figure 27 represents the structure of a DeviceRelationship XML element showing the attributes defined for DeviceRelationship.
Figure 21: DeviceRelationship Diagram
Table 49 lists the attributes defined for the DeviceRelationship element.

### Table 49: Attributes for DeviceRelationship

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The unique identifier for this DeviceRelationship. id is a required attribute. The id attribute MUST be unique within the MTConnectDevices document. An <a href="#">XML ID-type.</a></td>
<td>1</td>
</tr>
<tr>
<td>name</td>
<td>The name associated with this DeviceRelationship. name is provided as an additional human readable identifier for this DeviceRelationship. name is an optional attribute. An <a href="#">NMTOKEN XML type.</a></td>
<td>0..1</td>
</tr>
<tr>
<td>type</td>
<td>Defines the authority that this piece of equipment has relative to the associated piece of equipment. type is a required attribute. The value provided for type MUST be one of the following values: PARENT: This piece of equipment functions as a parent in the relationship with the associated piece of equipment. CHILD: This piece of equipment functions as a child in the relationship with the associated piece of equipment. PEER: This piece of equipment functions as a peer which provides equal functionality and capabilities in the relationship with the associated piece of equipment.</td>
<td>1</td>
</tr>
</tbody>
</table>
### Table 49 (Continued)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>criticality</td>
<td>Defines whether the services or functions provided by the associated piece of equipment is required for the operation of this piece of equipment. criticality is an optional attribute. The value provided for criticality MUST be one of the following values: CRITICAL: The services or functions provided by the associated piece of equipment is required for the operation of this piece of equipment. NONCRITICAL: The services or functions provided by the associated piece of equipment is not required for the operation of this piece of equipment.</td>
<td>0..1</td>
</tr>
<tr>
<td>deviceUuidRef</td>
<td>A reference to the associated piece of equipment. The value provided for deviceUuidRef MUST be the value provided for the uuid attribute of the Device element of the associated piece of equipment. deviceUuidRef is a required attribute. An <code>NMTOKEN</code> XML type.</td>
<td>1</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Occurrence</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td>role</td>
<td>Defines the services or capabilities that the referenced piece of equipment provides relative to this piece of equipment. role is an optional attribute. The value provided for role MUST be one of the following values: SYSTEM: The associated piece of equipment performs the functions of a System for this piece of equipment. In MTConnect, System provides utility type services to support the operation of a piece of equipment and these services are required for the operation of a piece of equipment. AUXILIARY: The associated piece of equipment performs the functions as an Auxiliary for this piece of equipment. In MTConnect, Auxiliary extends the capabilities of a piece of equipment, but is not required for the equipment to function.</td>
<td>0..1</td>
</tr>
<tr>
<td>href</td>
<td>A URI identifying the Agent that is publishing information for the associated piece of equipment. href MUST also include the UUID for that specific piece of equipment. href is of type xlink:href from the W3C XLink specification: (<a href="https://www.w3.org/TR/xlink11/">https://www.w3.org/TR/xlink11/</a>). href is an optional attribute.</td>
<td>0..1</td>
</tr>
<tr>
<td>xlink:type</td>
<td>The XLink type attribute MUST have a fixed value of locator as defined in W3C XLink 1.1 <a href="https://www.w3.org/TR/xlink11/">https://www.w3.org/TR/xlink11/</a> section 5.4 Locator Attribute (&lt;href&gt;). If the href attribute is provided, it MUST conform to the URI syntactic rules as defined in IETF RFC 3986 for Uniform Resource Identifiers. (<a href="https://www.ietf.org/rfc/rfc3986.txt">https://www.ietf.org/rfc/rfc3986.txt</a>)</td>
<td>0..1</td>
</tr>
</tbody>
</table>
9.2.1.2 ComponentRelationship

ComponentRelationship describes the association between two components within a piece of equipment that function independently but together perform a capability or service within a piece of equipment.

The XML schema in Figure 22 represents the structure of a ComponentRelationship XML element showing the attributes defined for ComponentRelationship.

Figure 22: ComponentRelationship Diagram

The Table lists the attributes defined for the ComponentRelationship element.
Table 50: Attributes for ComponentRelationship

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The unique identifier for this ComponentRelationship. id is a required attribute. The id attribute <strong>MUST</strong> be unique within the MTConnectDevices document. An [XML] ID-type.</td>
<td>1</td>
</tr>
<tr>
<td>name</td>
<td>The name associated with this ComponentRelationship. name is provided as an additional human readable identifier for this ComponentRelationship. name is an optional attribute. An [NMTOKEN] XML type.</td>
<td>0..1</td>
</tr>
</tbody>
</table>
| type      | Defines the authority that this component element has relative to the associated component element. type is a required attribute. The value provided for type **MUST** be one of the following values:  

- **PARENT**: This component functions as a parent in the relationship with the associated component element.  
- **CHILD**: This component functions as a child in the relationship with the associated component element.  
- **PEER**: This component functions as a peer which provides equal functionality and capabilities in the relationship with the associated component element. | 1 |
### Continuation of Table 50

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>criticality</td>
<td>Defines whether the services or functions provided by the associated component element is required for the operation of this piece of equipment. &lt;br&gt;criticality is an optional attribute. &lt;br&gt;The value provided for criticality MUST be one of the following values: &lt;br&gt;<strong>CRITICAL</strong>: The services or functions provided by the associated component element is required for the operation of this piece of equipment. &lt;br&gt;<strong>NONCRITICAL</strong>: The services or functions provided by the associated component element is not required for the operation of this piece of equipment.</td>
</tr>
<tr>
<td>idRef</td>
<td>A reference to the associated component element. &lt;br&gt;The value provided for idRef MUST be the value provided for the id attribute of the associated Component element. &lt;br&gt;idRef is a required attribute. An <code>NM_TOKEN</code> XML type.</td>
</tr>
<tr>
<td></td>
<td>Occurrence: 0..1</td>
</tr>
</tbody>
</table>

### 9.3 Specifications

Specifications is an XML container in the Configuration of a Component that contains one or more Specification elements describing the design characteristics for a piece of equipment.
Figure 23: Specifications Diagram
9.3.1 Specification

Specification elements define information describing the design characteristics for a piece of equipment.

9.3.1.1 Attributes for Specification

Table 51 lists the attributes defined to provide information for a Specification element.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Same as DataItem type. See Section 8 - Listing of Data Items</td>
<td>1</td>
</tr>
<tr>
<td>subType</td>
<td>Same as DataItem subtypes. See Section 8 - Listing of Data Items</td>
<td>0..1</td>
</tr>
<tr>
<td>dataItemIdRef</td>
<td>A reference to the id attribute of the DataItem associated with this element.</td>
<td>0..1</td>
</tr>
<tr>
<td>units</td>
<td>Same as DataItem units. See Section 7.2.2.5 - units Attribute for DataItem</td>
<td>0..1</td>
</tr>
<tr>
<td>compositionIdRef</td>
<td>A reference to the id attribute of the Composition associated with this element.</td>
<td>0..1</td>
</tr>
<tr>
<td>name</td>
<td>The name provides additional meaning and differentiates between Specifications. A name MUST exist when two Specifications have the same type and subType within a Component.</td>
<td>0..1</td>
</tr>
<tr>
<td>coordinateSystemIdRef</td>
<td>References the CoordinateSystem for geometric Specification elements.</td>
<td>0..1</td>
</tr>
</tbody>
</table>
9.3.1.2 Elements for Specification

Table 52 lists the elements defined to provide information for a Specification element.

Table 52: Elements for Specification

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>A numeric upper limit constraint.</td>
<td>0..1</td>
</tr>
<tr>
<td>Minimum</td>
<td>A numeric lower limit constraint.</td>
<td>0..1</td>
</tr>
<tr>
<td>Nominal</td>
<td>The numeric target or expected value.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

9.4 CoordinateSystems

CoordinateSystems aggregates CoordinateSystem configurations for a Component.

Figure 24: CoordinateSystems Diagram
9.4.1 CoordinateSystem

A CoordinateSystem is a reference system that associates a unique set of n parameters with each point in an n-dimensional space. Ref: ISO 10303-218:2004

9.4.1.1 Attributes for CoordinateSystem

Table 53 lists the attributes defined to provide information for a CoordinateSystem element.

Table 53: Attributes for CoordinateSystem

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The unique identifier for this element.</td>
<td>1</td>
</tr>
<tr>
<td>name</td>
<td>The name of the coordinate system. If more than one CoordinateSystem elements have the same type for the same Component, then the name attribute MUST be provided. Otherwise, the name attribute is optional. name provides as an additional human-readable identifier in addition to the id.</td>
<td>0..1</td>
</tr>
<tr>
<td>nativeName</td>
<td>The manufacturer’s name or users name for the coordinate system.</td>
<td>0..1</td>
</tr>
<tr>
<td>parentIdRef</td>
<td>A pointer to the id attribute of the parent CoordinateSystem.</td>
<td>0..1</td>
</tr>
<tr>
<td>type</td>
<td>The type of coordinate system.</td>
<td>1</td>
</tr>
</tbody>
</table>

9.4.1.1.1 CoordinateSystem types

Table 54 defines the various types of coordinate systems.
Table 54: CoordinateSystem types

<table>
<thead>
<tr>
<th>type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORLD</td>
<td>stationary coordinate system referenced to earth, which is independent of the robot motion. <em>Ref:</em> ISO 9787:2013 For non-robotic devices, stationary coordinate system referenced to earth, which is independent of the motion of a piece of equipment.</td>
</tr>
<tr>
<td>BASE</td>
<td>coordinate system referenced to the base mounting surface. <em>Ref:</em> ISO 9787:2013 A base mounting surface is a connection surface between the arm and its supporting structure. <em>Ref:</em> ISO 9787:2013 For non-robotic devices, it is the connection surface between the device and its supporting structure.</td>
</tr>
<tr>
<td>OBJECT</td>
<td>coordinate system referenced to the object. <em>Ref:</em> ISO 9787:2013</td>
</tr>
<tr>
<td>TASK</td>
<td>coordinate system referenced to the site of the task. <em>Ref:</em> ISO 9787:2013</td>
</tr>
<tr>
<td>MECHANICAL INTERFACE</td>
<td>coordinate system referenced to the mechanical interface. <em>Ref:</em> ISO 9787:2013</td>
</tr>
<tr>
<td>TOOL</td>
<td>coordinate system referenced to the tool or to the end effector attached to the mechanical interface. <em>Ref:</em> ISO 9787:2013</td>
</tr>
<tr>
<td>MOBILE PLATFORM</td>
<td>coordinate system referenced to one of the components of a mobile platform. <em>Ref:</em> ISO 8373:2012</td>
</tr>
<tr>
<td>MACHINE</td>
<td>coordinate system referenced to the home position and orientation of the primary axes of a piece of equipment.</td>
</tr>
<tr>
<td>CAMERA</td>
<td>coordinate system referenced to the sensor which monitors the site of the task. <em>Ref:</em> ISO 9787:2013</td>
</tr>
</tbody>
</table>
### 9.4.1.2 Elements for CoordinateSystem

Table [55] lists the elements defined to provide information for a CoordinateSystem element.

**Table 55: Elements for CoordinateSystem**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin</td>
<td>The coordinates of the origin position of a coordinate system. The coordinate <strong>MUST</strong> be in MILLIMETER_3D.</td>
<td>0..1</td>
</tr>
<tr>
<td>Transformation</td>
<td>The process of transforming to the origin position of the coordinate system from a parent coordinate system using Translation and Rotation.</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Notes: Only one of Location or Transformation can be defined for a CoordinateSystem.

### 9.4.1.2.1 Elements for Transformation

Table [56] lists the elements defined to provide information for a Transformation element.

**Table 56: Elements for Transformation**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRANSLATION</td>
<td>Translations along X, Y, and Z axes are expressed as x, y, and z respectively within a 3-dimensional vector. The values <strong>MUST</strong> be given in MILLIMETER_3D.</td>
<td>0..1</td>
</tr>
<tr>
<td>ROTATION</td>
<td>Rotations about X, Y, and Z axes are expressed in A, B, and C respectively within a 3-dimensional vector. The values <strong>MUST</strong> be given in DEGREE_3D. Positive A, B, and C are in the directions to advance right-hand screws in the positive X, Y, and Z directions, respectively. <strong>Ref:</strong> ISO 9787:2013</td>
<td>0..1</td>
</tr>
</tbody>
</table>
Appendices

A Bibliography


