



# **MTConnect<sup>®</sup> Standard**

## **Part 2.0 – Device Information Model**

### **Version 2.0.0**

Prepared for: MTConnect Institute

Prepared from: MTConnectSysMLModel.xml

Prepared on: May 24, 2022

MTConnect<sup>®</sup> 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 MTConnect Specification or Material. AMT grants to you a non-exclusive, non-transferable, revocable, non-sublicensable, fully-paid-up copyright license to reproduce, copy and redistribute this MTConnect Specification or Material, provided that you may only copy or redistribute the MTConnect Specification or Material in the form in which you received it, without modifications, and with all copyright notices and other notices and disclaimers contained in the MTConnect Specification or Material.

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

The normative XMI is located at the following URL: `MTConnectSysMLModel.xml`

# Table of Contents

<b>1</b>	<b>Purpose of This Document</b>	<b>2</b>
<b>2</b>	<b>Terminology and Conventions</b>	<b>3</b>
2.1	General Terms . . . . .	3
2.2	Information Model Terms . . . . .	9
2.3	Protocol Terms . . . . .	10
2.4	HTTP Terms . . . . .	12
2.5	XML Terms . . . . .	14
2.6	MTConnect Terms . . . . .	15
2.7	Acronyms . . . . .	16
2.8	MTConnect References . . . . .	28
<b>3</b>	<b>Device Information Model</b>	<b>29</b>
3.1	Device . . . . .	30
3.1.1	Value Properties of Device . . . . .	30
3.1.2	Part Properties of Device . . . . .	31
3.1.3	Commonly Observed DataItem Types for Device . . . . .	32
<b>4</b>	<b>Components Model</b>	<b>33</b>
4.1	Components . . . . .	35
4.1.1	Component . . . . .	35
4.1.2	Description . . . . .	39
4.2	Devices . . . . .	40
4.2.1	Agent . . . . .	40
4.3	Component Types . . . . .	41
4.3.1	Actuator . . . . .	41
4.3.2	Adapter . . . . .	42
4.3.3	Auxiliary . . . . .	42
4.3.4	Axis . . . . .	44
4.3.5	Chuck . . . . .	47
4.3.6	Door . . . . .	47
4.3.7	Lock . . . . .	47
4.3.8	Part . . . . .	48
4.3.9	Path . . . . .	48
4.3.10	<<deprecated>>Power . . . . .	49
4.3.11	Process . . . . .	49
4.3.12	Resource . . . . .	50
4.3.13	Sensor . . . . .	51
4.3.14	Structure . . . . .	51
4.3.15	System . . . . .	52
4.4	Component Organizer Types . . . . .	55

4.4.1	<<organizer>>Adapters . . . . .	55
4.4.2	<<organizer>>Auxiliaries . . . . .	55
4.4.3	<<organizer>>Axes . . . . .	55
4.4.4	<<organizer>>Controllers . . . . .	55
4.4.5	<<organizer>>Interfaces . . . . .	55
4.4.6	<<organizer>>Parts . . . . .	55
4.4.7	<<organizer>>Processes . . . . .	56
4.4.8	<<organizer>>Resources . . . . .	56
4.4.9	<<organizer>>Structures . . . . .	56
4.4.10	<<organizer>>Systems . . . . .	56
<b>5</b>	<b>Compositions Model</b>	<b>57</b>
5.1	Compositions . . . . .	57
5.1.1	Composition . . . . .	57
5.2	Composition Types . . . . .	59
5.2.1	Actuator . . . . .	59
5.2.2	Amplifier . . . . .	60
5.2.3	Ballscrew . . . . .	60
5.2.4	Belt . . . . .	60
5.2.5	Brake . . . . .	60
5.2.6	Chain . . . . .	60
5.2.7	Chopper . . . . .	60
5.2.8	Chuck . . . . .	60
5.2.9	Chute . . . . .	61
5.2.10	CircuitBreaker . . . . .	61
5.2.11	Clamp . . . . .	61
5.2.12	Compressor . . . . .	61
5.2.13	CoolingTower . . . . .	61
5.2.14	Door . . . . .	61
5.2.15	Drain . . . . .	61
5.2.16	Encoder . . . . .	62
5.2.17	ExpiredPot . . . . .	62
5.2.18	ExposureUnit . . . . .	62
5.2.19	ExtrusionUnit . . . . .	62
5.2.20	Fan . . . . .	62
5.2.21	Filter . . . . .	62
5.2.22	Galvanomotor . . . . .	62
5.2.23	Gripper . . . . .	63
5.2.24	Hopper . . . . .	63
5.2.25	LinearPositionFeedback . . . . .	63
5.2.26	Motor . . . . .	63
5.2.27	Oil . . . . .	63

5.2.28	Pot . . . . .	63
5.2.29	PowerSupply . . . . .	64
5.2.30	Pulley . . . . .	64
5.2.31	Pump . . . . .	64
5.2.32	Reel . . . . .	64
5.2.33	RemovalPot . . . . .	64
5.2.34	ReturnPot . . . . .	64
5.2.35	SensingElement . . . . .	64
5.2.36	Spreader . . . . .	65
5.2.37	StagingPot . . . . .	65
5.2.38	Station . . . . .	65
5.2.39	StorageBattery . . . . .	65
5.2.40	Switch . . . . .	65
5.2.41	Table . . . . .	65
5.2.42	Tank . . . . .	65
5.2.43	Tensioner . . . . .	66
5.2.44	TransferArm . . . . .	66
5.2.45	TransferPot . . . . .	66
5.2.46	Transformer . . . . .	66
5.2.47	Valve . . . . .	66
5.2.48	Vat . . . . .	66
5.2.49	Water . . . . .	66
5.2.50	Wire . . . . .	67
5.2.51	Workpiece . . . . .	67
<b>6</b>	<b>DataItems Model</b>	<b>68</b>
6.1	DataItems . . . . .	68
6.1.1	DataItem . . . . .	68
6.2	Properties of DataItem . . . . .	88
6.2.1	ResetTrigger . . . . .	88
6.2.2	Source . . . . .	89
6.2.3	InitialValue . . . . .	90
6.2.4	Filter . . . . .	90
6.2.5	MinimumDeltaFilter . . . . .	91
6.2.6	PeriodFilter . . . . .	91
6.2.7	Constraints . . . . .	91
6.2.8	Definition . . . . .	93
6.3	Properties of Definition . . . . .	94
6.3.1	CellDefinition . . . . .	94
6.3.2	EntryDefinition . . . . .	95
6.4	Relationship Types for DataItem . . . . .	97
6.4.1	SpecificationRelationship . . . . .	97

6.4.2	DataItemRelationship . . . . .	98
6.5	DataItem Types . . . . .	99
6.5.1	Condition . . . . .	100
6.5.2	Event . . . . .	101
6.5.3	Sample . . . . .	109
<b>7</b>	<b>References Model</b>	<b>116</b>
7.1	References . . . . .	116
7.1.1	Reference . . . . .	116
7.1.2	DataItemRef . . . . .	118
7.1.3	ComponentRef . . . . .	118
<b>8</b>	<b>Configurations Model</b>	<b>119</b>
8.1	Configurations . . . . .	119
8.1.1	Configuration . . . . .	119
8.2	CoordinateSystems . . . . .	121
8.2.1	CoordinateSystem . . . . .	121
8.2.2	Origin . . . . .	126
8.2.3	Transformation . . . . .	126
8.2.4	Rotation . . . . .	127
8.2.5	Translation . . . . .	127
8.3	Motion . . . . .	127
8.3.1	Motion . . . . .	127
8.3.2	Axis . . . . .	131
8.4	Relationships . . . . .	132
8.4.1	Relationship . . . . .	132
8.4.2	ComponentRelationship . . . . .	133
8.4.3	DeviceRelationship . . . . .	135
8.5	Sensor . . . . .	136
8.5.1	SensorConfiguration . . . . .	138
8.5.2	Channel . . . . .	142
8.6	SolidModel . . . . .	144
8.6.1	SolidModel . . . . .	145
8.6.2	Scale . . . . .	148
8.7	Specifications . . . . .	148
8.7.1	Specification . . . . .	148
8.7.2	ProcessSpecification . . . . .	153
8.7.3	ControlLimits . . . . .	153
8.7.4	AlarmLimits . . . . .	154
8.7.5	SpecificationLimits . . . . .	155
8.7.6	UpperWarning . . . . .	156
8.7.7	UpperLimit . . . . .	156

8.7.8	Maximum . . . . .	157
8.7.9	LowerLimit . . . . .	157
8.7.10	LowerWarning . . . . .	157
8.7.11	Minimum . . . . .	157
8.7.12	Nominal . . . . .	157
<b>9</b>	<b>Profile</b>	<b>158</b>
9.1	DataTypes . . . . .	158
9.1.1	boolean . . . . .	158
9.1.2	ID . . . . .	158
9.1.3	string . . . . .	158
9.1.4	float . . . . .	158
9.1.5	dateTime . . . . .	159
9.1.6	integer . . . . .	159
9.1.7	xlinktype . . . . .	159
9.1.8	xslang . . . . .	159
9.1.9	SECOND . . . . .	159
9.1.10	IDREF . . . . .	159
9.1.11	xlinkhref . . . . .	159
9.1.12	x509 . . . . .	160
9.1.13	int32 . . . . .	160
9.1.14	int64 . . . . .	160
9.1.15	version . . . . .	160
9.1.16	uInt32 . . . . .	160
9.1.17	uInt64 . . . . .	160
9.2	Stereotypes . . . . .	160
9.2.1	organizer . . . . .	160
9.2.2	deprecated . . . . .	161
9.2.3	extensible . . . . .	161
9.2.4	informative . . . . .	161
9.2.5	valueType . . . . .	161
9.2.6	normative . . . . .	161
9.2.7	observes . . . . .	161
	<b>Appendices</b>	<b>163</b>
A	Bibliography . . . . .	163
B	XML Schema Diagrams . . . . .	165
B.1	Components Schema Diagrams . . . . .	165
B.2	DataItems Schema Diagrams . . . . .	165
B.3	References Schema Diagrams . . . . .	165
B.4	Configuration Schema Diagrams . . . . .	165
C	XML Examples . . . . .	166

C.1	Device Entity Hierarchy Example . . . . .	166
C.2	Component Example . . . . .	167
C.3	Component with Compositions Example . . . . .	168
C.4	CoordinateSystem Example . . . . .	169
C.5	Motion Example . . . . .	170
C.6	Relationship Example . . . . .	171
C.7	SolidModel Example . . . . .	172
C.8	Specification Example . . . . .	173
C.9	Example of sensing element provided as data item associated with a Component . . . . .	174
C.10	Example of Sensor for rotary axis . . . . .	174
C.11	Example of sensor unit with sensing element . . . . .	175
C.12	Example of configuration data for Sensor . . . . .	175



## Table of Figures

<b>Figure 1: Device Entity Hierarchy Example</b> . . . . .	33
<b>Figure 2: Components</b> . . . . .	35
<b>Figure 3: Component Example</b> . . . . .	36
<b>Figure 4: Component with Compositions Example</b> . . . . .	58
<b>Figure 5: DataItems</b> . . . . .	69
<b>Figure 6: References</b> . . . . .	117
<b>Figure 7: Configuration</b> . . . . .	119
<b>Figure 8: CoordinateSystem</b> . . . . .	122
<b>Figure 9: CoordinateSystem Example</b> . . . . .	123
<b>Figure 10: Motion</b> . . . . .	128
<b>Figure 11: Motion Example</b> . . . . .	129
<b>Figure 12: Relationships</b> . . . . .	132
<b>Figure 13: Relationship Example</b> . . . . .	133
<b>Figure 14: Sensing Element provided as a DataItem Example</b> . . . . .	137
<b>Figure 15: Sensor for Rotary Axis Example</b> . . . . .	139
<b>Figure 16: Sensor unit with Sensing Elements Example</b> . . . . .	140
<b>Figure 17: Sensor Configuration Example</b> . . . . .	140
<b>Figure 18: SensorConfiguration</b> . . . . .	141
<b>Figure 19: SolidModel</b> . . . . .	144
<b>Figure 20: SolidModel Example</b> . . . . .	145
<b>Figure 21: Specifications</b> . . . . .	148
<b>Figure 22: Specification Example</b> . . . . .	150
<b>Figure 23: DataTypes</b> . . . . .	158
<b>Figure 24: Stereotypes</b> . . . . .	162
<b>Figure 25: Components Schema</b> . . . . .	165
<b>Figure 26: Component Description Schema</b> . . . . .	166
<b>Figure 27: DataItems Schema</b> . . . . .	177
<b>Figure 28: Source Schema</b> . . . . .	178
<b>Figure 29: Constraints Schema</b> . . . . .	179
<b>Figure 30: Filter Schema</b> . . . . .	179
<b>Figure 31: Definition Schema</b> . . . . .	180
<b>Figure 32: References Schema</b> . . . . .	181
<b>Figure 33: ComponentRef Schema</b> . . . . .	181
<b>Figure 34: DataItemRef Schema</b> . . . . .	182
<b>Figure 35: Configuration Schema</b> . . . . .	183
<b>Figure 36: CoordinateSystem Schema</b> . . . . .	184
<b>Figure 37: Motion Schema</b> . . . . .	184
<b>Figure 38: Relationships Schema</b> . . . . .	185
<b>Figure 39: ComponentRelationship Schema</b> . . . . .	186
<b>Figure 40: DeviceRelationship Schema</b> . . . . .	186

<b>Figure 41:SensorConfiguration Schema</b> . . . . .	187
<b>Figure 42:SolidModel Schema</b> . . . . .	188
<b>Figure 43:Specifications Schema</b> . . . . .	189
<b>Figure 44:ProcessSpecification Schema</b> . . . . .	190

## List of Tables

<b>Table 1: Value Properties of Device</b>	30
<b>Table 2: Part Properties of Device</b>	31
<b>Table 3: Commonly Observed DataItem Types for Device</b>	32
<b>Table 4: Value Properties of Component</b>	36
<b>Table 5: Reference Properties of Component</b>	37
<b>Table 6: Part Properties of Component</b>	38
<b>Table 7: Value Properties of Description</b>	40
<b>Table 8: Commonly Observed DataItem Types for Door</b>	47
<b>Table 9: Commonly Observed DataItem Types for Lock</b>	48
<b>Table 10: Commonly Observed DataItem Types for PartOccurrence</b>	48
<b>Table 11: Commonly Observed DataItem Types for Path</b>	49
<b>Table 12: Commonly Observed DataItem Types for ProcessOccurrence</b>	50
<b>Table 13: Part Properties of Controller</b>	52
<b>Table 14: Value Properties of Composition</b>	58
<b>Table 15: Part Properties of Composition</b>	59
<b>Table 16: Value Properties of DataItem</b>	70
<b>Table 17: Reference Properties of DataItem</b>	86
<b>Table 18: Part Properties of DataItem</b>	87
<b>Table 19: Value Properties of Source</b>	89
<b>Table 20: Value Properties of Filter</b>	90
<b>Table 21: Value Properties of Constraints</b>	92
<b>Table 22: Part Properties of Constraints</b>	92
<b>Table 23: Part Properties of Definition</b>	93
<b>Table 24: Value Properties of CellDefinition</b>	94
<b>Table 25: Part Properties of CellDefinition</b>	95
<b>Table 26: Value Properties of EntryDefinition</b>	96
<b>Table 27: Part Properties of EntryDefinition</b>	97
<b>Table 28: Value Properties of SpecificationRelationship</b>	98
<b>Table 29: Value Properties of DataItemRelationship</b>	98
<b>Table 30: Value Properties of Condition</b>	100
<b>Table 31: Value Properties of Event</b>	101
<b>Table 32: Value Properties of Sample</b>	110
<b>Table 33: Value Properties of Reference</b>	116
<b>Table 34: Part Properties of Configuration</b>	120
<b>Table 35: Value Properties of CoordinateSystem</b>	121
<b>Table 36: Part Properties of CoordinateSystem</b>	125
<b>Table 37: Part Properties of Transformation</b>	126
<b>Table 38: Value Properties of Motion</b>	128
<b>Table 39: Part Properties of Motion</b>	131
<b>Table 40: Value Properties of ComponentRelationship</b>	134

<b>Table 41: Value Properties of DeviceRelationship</b>	135
<b>Table 42: Value Properties of SensorConfiguration</b>	138
<b>Table 43: Part Properties of SensorConfiguration</b>	142
<b>Table 44: Value Properties of Channel</b>	143
<b>Table 45: Part Properties of Channel</b>	143
<b>Table 46: Value Properties of SolidModel</b>	146
<b>Table 47: Part Properties of SolidModel</b>	147
<b>Table 48: Value Properties of Specification</b>	149
<b>Table 49: Part Properties of Specification</b>	152
<b>Table 50: Part Properties of ProcessSpecification</b>	153
<b>Table 51: Part Properties of ControlLimits</b>	154
<b>Table 52: Part Properties of AlarmLimits</b>	155
<b>Table 53: Part Properties of SpecificationLimits</b>	156

## 1 1 Purpose of This Document

2 This document, *MTConnect Standard: Part 2.0 - Device Information Model* of the MT-  
3 Connect Standard, establishes the rules and terminology to be used by designers to de-  
4 scribe the function and operation of a piece of equipment and to define the data that is  
5 provided by an *agent* from the equipment. The *Device Information Model* also defines the  
6 structure for the *response document* that is returned from an *agent* in response to a *probe*  
7 *request*.

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

11 Note: See *MTConnect Standard: Part 3.0 - Observation Information Model*  
12 of the MTConnect Standard for details on the *response document* that are  
13 returned from an *agent* in response to a *sample request* or *current request*.

## 14 2 Terminology and Conventions

15 Refer to *MTConnect Standard Part 1.0 - Fundamentals* for a dictionary of terms, reserved  
16 language, and document conventions used in the MTConnect Standard.

### 17 2.1 MTConnect References

18 [MTConnect Part 1.0] *MTConnect Standard Part 1.0 - Fundamentals*. Version 2.0.

19 [MTConnect Part 2.0] *MTConnect Standard: Part 2.0 - Device Information Model*. Ver-  
20 sion 2.0.

21 [MTConnect Part 3.0] *MTConnect Standard: Part 3.0 - Observation Information Model*.  
22 Version 2.0.

23 [MTConnect Part 4.0] *MTConnect Standard: Part 4.0 - Asset Information Model*. Ver-  
24 sion 2.0.

25

## 26 3 Device Information Model

27 The *Device Information Model* provides a representation of the physical and logical con-  
28 figuration for a piece of equipment used for a manufacturing process or for any other  
29 purpose. It also provides the definition of data that may be reported by that equipment.

30 Using information defined in the *Device Information Model*, a software application can  
31 determine the configuration and reporting capabilities of a piece of equipment. To do this,  
32 the software application issues a *probe request* (defined in *MTConnect Standard Part 1.0 -*  
33 *Fundamentals*) to an *agent* associated with a piece of equipment. An *agent* responds to the  
34 *probe request* with an *MTConnectDevices Response Document* that contains information  
35 describing both the physical and logical structure of the piece of equipment and a detailed  
36 description of each *Observation* that can be reported by the *agent* associated with the  
37 piece of equipment. This information allows the client software application to interpret  
38 the document and to extract the data with the same meaning, value, and context that it had  
39 at its original source.

40 The *MTConnectDevices Response Document* is comprised of two sections: `Header` and  
41 `Devices`.

42 The `Header` section contains protocol related information as defined in *MTConnect Stan-*  
43 *dard Part 1.0 - Fundamentals*.

44 The `Devices` section of the *MTConnectDevices Response Document* contains a `Device`  
45 entity for each piece of equipment described in the document. Each `Device` is comprised  
46 of two primary types of entities - *Components* and *DataItems*.

47 *Components* organize information that represents the physical and logical parts and sub-  
48 parts of a piece of equipment (See *Section 4.1 - Components* for more details).

49 *DataItems* describe data that can be reported by a piece of equipment. In the *Device Infor-*  
50 *mation Model*, *DataItems* are defined as `DataItem` entities (See *Section 6.5 - DataItem*  
51 *Types*).

52 The *Components* and *DataItems* in the *MTConnectDevices Response Document* provide  
53 information representing the physical and logical structure for a piece of equipment and  
54 the types of data that the piece of equipment can report relative to that structure. The *MT-*  
55 *ConnectDevices Response Document* does not contain values for the data types reported  
56 by the piece of equipment. The *MTConnectStreams Response Document* defined in *MT-*  
57 *Connect Standard: Part 3.0 - Observation Information Model* provides the data values that  
58 are reported by the piece of equipment.

59 Note: The MTConnect Standard also defines the information model for assets.  
 60 An asset is something that is used in the manufacturing process, but is not  
 61 permanently associated with a single piece of equipment, can be removed  
 62 from the piece of equipment without compromising its function, and can be  
 63 associated with other pieces of equipment during its lifecycle. See *MTConnect*  
 64 *Standard: Part 4.0 - Asset Information Model* for more details on assets.

## 65 3.1 Device

66 Component composed of a piece of equipment that produces observations about itself.

67 A Device *organizes* its parts as Component entities.

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

69 A Device **MUST** have the following DataItems: Availability, AssetChanged,  
 70 and AssetRemoved.

71 See *Section 4.1 - Components* for more details on the properties of Device.

72 See *Section 3.1.2 - Part Properties of Device* for a list of *top level* Compo-  
 73 nent types for a Device.

### 74 3.1.1 Value Properties of Device

75 *Table 1* lists the Value Properties of Device.

Value Property name	Value Property type	Multiplicity
<<deprecated>> iso841Class	string	0..1
uuid	ID	1
mtconnectVersion	string	0..1
name	string	1

**Table 1:** Value Properties of Device

76 Descriptions for Value Properties of Device:

- 77 • <<deprecated>> iso841Class
- 78 **DEPRECATED** in *MTConnect Version 1.2*.



- `mtconnectVersion`  
MTConnect version of the *Device Information Model* used to configure the information to be published for a piece of equipment in an *MTConnect Response Document*.

## 3.1.2 Part Properties of Device

Table 2 lists the Part Properties of Device.

Part Property name	Multiplicity
Auxiliary (organized by <<organizer>> Auxiliaries)	0..*
Controller (organized by <<organizer>> Controllers)	0..*
Interface (organized by <<organizer>> Interfaces)	0..*
Resource (organized by <<organizer>> Resources)	0..*
Structure (organized by <<organizer>> Structures)	0..*
System (organized by <<organizer>> Systems)	0..*
Axis (organized by <<organizer>> Axes)	0..*
Adapter (organized by <<organizer>> Adapters)	0..*

**Table 2:** Part Properties of Device

Descriptions for Part Properties of Device:

- **Auxiliary**  
abstract Component composed of removable part(s) of a piece of equipment that provides supplementary or extended functionality.
- **Controller**  
System that provides regulation or management of a system or component. *Ref ISO 16484-5:2017*
- **Interface**  
abstract Component that coordinates actions and activities between pieces of equipment.
- **Resource**  
abstract Component composed of material or personnel involved in a manufacturing process.

- 97     • **Structure**
- 98       Component composed of part(s) comprising the rigid bodies of the piece of equip-
- 99       ment.
- 100    • **System**
- 101      abstract Component that is permanently integrated into the piece of equipment.
- 102    • **Axis**
- 103      abstract Component composed of a motion system that provides linear or rota-
- 104      tional motion for a piece of equipment.
- 105    • **Adapter**
- 106      Component that provides information about the data source for an *MTConnect*
- 107      Agent.

### 108 3.1.3 Commonly Observed DataItem Types for Device

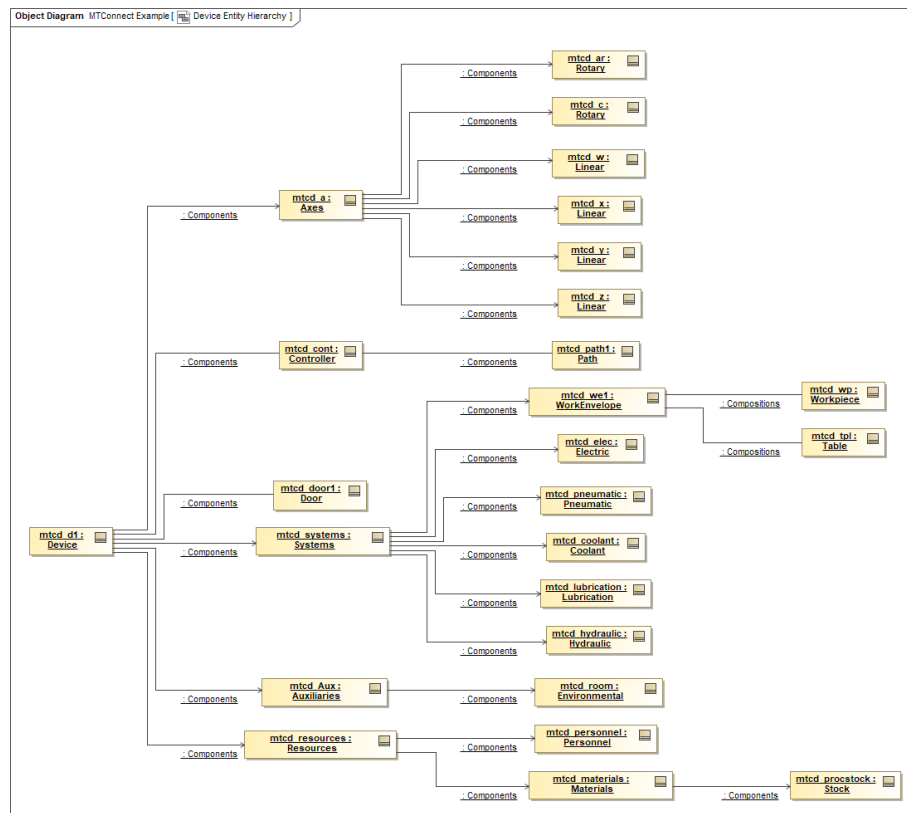
109 Table 3 lists the Commonly Observed DataItem Types for Device.

Commonly Observed DataItem Types	Multiplicity
Availability	1
AssetChanged	1
AssetRemoved	1

**Table 3:** Commonly Observed DataItem Types for Device

## 110 4 Components Model

111 MTConnectDevices provides the physical and logical architecture of a piece of equip-  
 112 ment. Figure 1 provides an overview of the entities used in an example of an MTCon-  
 113 nectDevices entity.



**Figure 1: Device Entity Hierarchy Example**

114 Note 1 to entry: See *Example 1* for the Extensible Markup Language (XML)  
 115 representation of the same example.

116 Note 2 to entry: Example above only shows the Component and Composi-  
 117 tion level hierarchy. For a complete semantics on each kind see *Section 4.1*  
 118 - *Components* and *Section 5.1 - Compositions*.

119 A variety of entities are defined to describe a piece of equipment. Some of these entities  
 120 **MUST** always be defined for an MTConnectDevices entity, while others are optional  
 121 and **MAY** be used, as required, to provide additional structure.

122 The first, or highest level, entity defined for an MTConnectDevices entity is De-  
 123 vices. Devices is used to group one or more pieces of equipment into a single docu-  
 124 ment. Devices **MUST** always be defined for an MTConnectDevices entity.

125 Device is the next entity defined for the MTConnectDevices entity. A separate De-  
 126 vice entity is used to identify each piece of equipment for an MTConnectDevices  
 127 entity. Each Device provides information on the physical and logical structure of the  
 128 piece of equipment and the data associated with that equipment. Device can also repre-  
 129 sent any logical grouping of pieces of equipment that function as a unit or any other data  
 130 source that provides data through an *agent*.

131 One or more Device entities **MUST** always be defined for an MTConnectDevices  
 132 entity.

133 Components is the next entity defined for the MTConnectDevices entity. Com-  
 134 ponents is used to group information describing lower level physical parts or logical  
 135 functions of a piece of equipment.

136 Component is the next level of entity defined for the MTConnectDevices entity.  
 137 Component is both an abstract type entity and an *organizer* type entity.

138 As an abstract entity, Component will always be realized by a specific Component type  
 139 defined in *Section 4.3 - Component Types*. Each Component can also be used to organize  
 140 information describing *lower level* entities or *DataItems* associated with the Component.

141 If *lower level* entities are described, these entities are by definition child Component  
 142 entities of a parent Component. At this next level, the *lower level* child Component  
 143 entities are grouped by Components.

144 A Component **MAY** be further decomposed into Composition entities that are grouped  
 145 by Compositions. These describe the lowest level basic structural or functional build-  
 146 ing blocks contained within a Component. Data provided for a Component provides  
 147 more specific meaning when it is associated with one of the Composition entities of  
 148 the Component. The different Composition types that **MAY** be defined for the MT-  
 149 ConnectDevices entity are defined in *Section 5.1 - Compositions*.

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

152 Note: See Figure 1 for an example.

## 153 4.1 Components

154 This section provides semantic information for the Component entity. Figure 2 shows  
 155 the Component model.

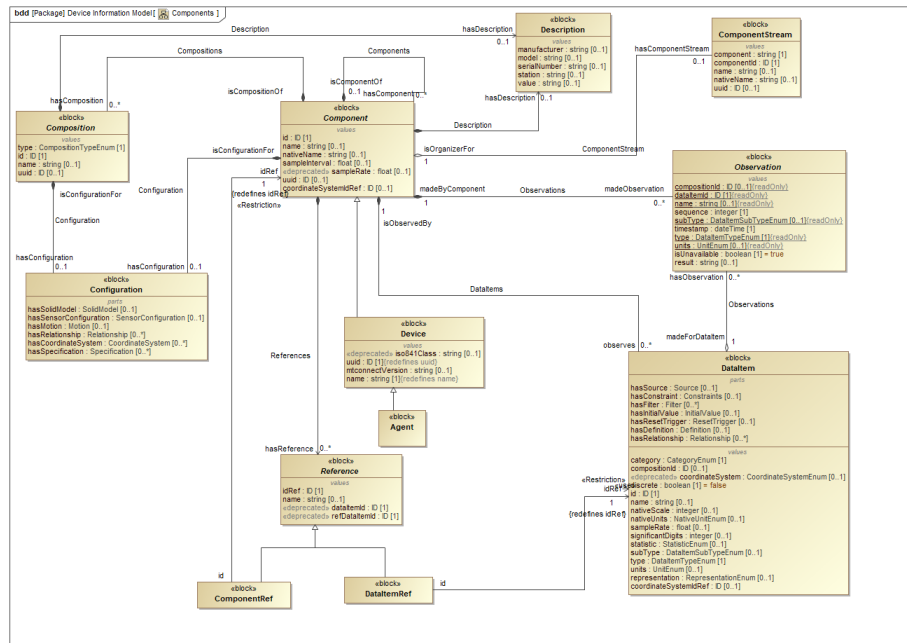


Figure 2: Components

156 Note: See *Section B.1 - Components Schema Diagrams* for XML schema.

### 157 4.1.1 Component

158 logical or physical entity that provides a capability.

159 Component is an abstract entity and will be realized by specific Component types for  
 160 an MTConnectDevices entity. See *Section 4.3 - Component Types* for more details on  
 161 the Component types.

162 Component also provides structure for describing the *lower level* entities associated with  
 163 it.

164 At least one of Component, DataItem, or Reference entities **MUST** be provided  
 165 for a Component.

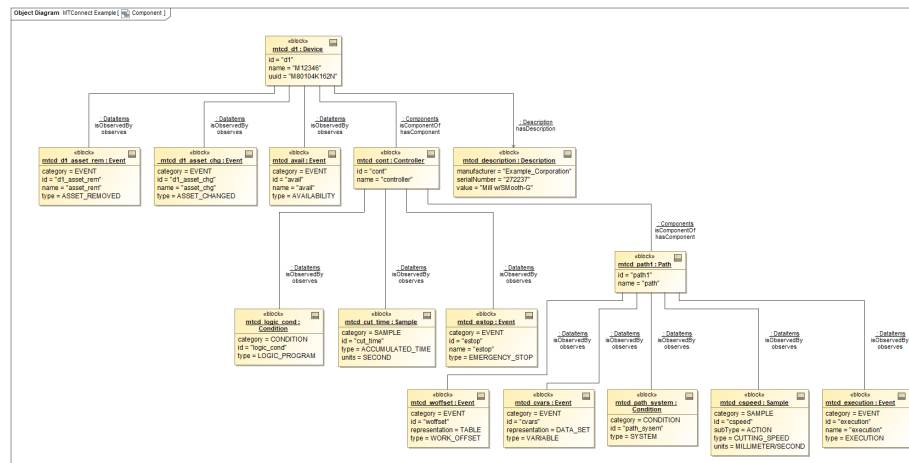


Figure 3: Component Example

Note: See *Example 2* for the XML representation of the same example.

#### 4.1.1.1 Value Properties of Component

Table 4 lists the Value Properties of Component.

Value Property name	Value Property type	Multiplicity
id	ID	1
name	string	0..1
nativeName	string	0..1
sampleInterval	float	0..1
<<deprecated>> sampleRate	float	0..1
uuid	ID	0..1
coordinateSystemIdRef	ID	0..1

Table 4: Value Properties of Component

Descriptions for Value Properties of Component:

- id  
unique identifier for the Component.
- name  
name of the Component.  
When provided, name **MUST** be unique for all child Component entities of a parent Component.

- 176 • `nativeName`
- 177 common name associated with `Component`.
- 178 • `sampleInterval`
- 179 interval in milliseconds between the completion of the reading of the data associated
- 180 with the `Component` until the beginning of the next sampling of that data.
- 181 This information may be used by client software applications to understand how
- 182 often information from a `Component` is expected to be refreshed.
- 183 The refresh rate for data from all child `Component` entities will be the same as for
- 184 the parent `Component` element unless specifically overridden by another sam-
- 185 pleInterval provided for the child `Component`.
- 186 • `<<deprecated>> sampleRate`
- 187 **DEPRECATED** in *MTConnect Version 1.2*. Replaced by `sampleInterval`, `Component`.
- 188 • `uuid`
- 189 universally unique identifier for the `Component`.
- 190 • `coordinateSystemIdRef`
- 191 specifies the `CoordinateSystem` for this `Component` and its children.

#### 192 4.1.1.2 Reference Properties of Component

193 *Table 5* lists the Reference Properties of `Component`.

Reference Property name	Multiplicity
<code>ComponentStream</code>	0..1

**Table 5:** Reference Properties of `Component`

194 Descriptions for Reference Properties of `Component`:

- 195 • `ComponentStream`
- 196 organizes the data associated with each `Component` entity defined for a `Device`
- 197 in the associated *MTConnectDevices Response Document*.
- 198 `ComponentStream` *organizes* the `Observation` entities associated with the
- 199 `Component`.
- 200 See *MTConnect Standard: Part 3.0 - Observation Information Model* for the Com-
- 201 ponentStream model.

202 Note 1 to entry: In the XML representation, `ComponentStream` enti-  
 203 ties **MUST NOT** appear in the *MTConnectDevices Response Document*.

204 Note 2 to entry: In the XML representation, `ComponentStream` enti-  
 205 ties **MUST** appear only in the *MTConnectStreams Response Document*.

### 206 4.1.1.3 Part Properties of Component

207 *Table 6* lists the Part Properties of Component.

Part Property name	Multiplicity
Description	0..1
Composition (organized by Compositions)	0..*
Component (organized by Components)	0..*
Configuration	0..1
DataItem (organized by DataItems)	0..*
Observation (organized by Observations)	0..*
Reference (organized by References)	0..*

**Table 6:** Part Properties of Component

208 Descriptions for Part Properties of Component:

- 209 • Description  
 210 descriptive content.  
 211 See *Section 4.1.2 - Description*.
- 212 • Composition  
 213 functional part of a piece of equipment contained within a Component.  
 214 `Compositions` groups one or more `Composition` entities. See *Section 5.1 -*  
 215 *Compositions*.
- 216 • Component  
 217 logical or physical entity that provides a capability.  
 218 `Components` groups one or more `Component` entities.
- 219 • Configuration  
 220 technical information about an entity describing its physical layout, functional char-  
 221 acteristics, and relationships with other entities.  
 222 See *Section 8.1 - Configurations*.



- 223     • `DataItem`  
224       information reported about a piece of equipment.  
225       `DataItems` groups one or more `DataItem` entities. See *Section 6.1 - DataItems*.
- 226     • `Observation`  
227       abstract entity that provides telemetry data for a `DataItem` at a point in time.  
228       `Observations` groups one or more `Observations` made by the Component  
229       entity.  
230       Component make `Observations` about observed `DataItems`.  
231       See *MTConnect Standard: Part 3.0 - Observation Information Model* for the Ob-  
232       servation model.
- 233       Note 1 to entry: In the XML representation, `Observation` entities  
234       **MUST NOT** appear in the *MTConnectDevices Response Document*.
- 235       Note 2 to entry: In the XML representation, `Observation` entities  
236       **MUST** appear only in the *MTConnectStreams Response Document*.
- 237     • `Reference`  
238       pointer to information that is associated with another entity defined elsewhere in the  
239       `MTConnectDevices` entity for a piece of equipment.  
240       `References` groups one or more `Reference` entities associated with the Com-  
241       ponent. See *Section 7.1 - References*.

## 242 4.1.2 Description

243 descriptive content.

244     Note 1 to entry: See Figure 3 for an example.

245     Note 2 to entry: See *Example 2* for the XML representation of the same ex-  
246     ample.

247 The value of `Description` **MUST** be string.

### 248 4.1.2.1 Value Properties of Description

249 *Table 7* lists the Value Properties of `Description`.

Value Property name	Value Property type	Multiplicity
manufacturer	string	0..1
model	string	0..1
serialNumber	string	0..1
station	string	0..1

**Table 7:** Value Properties of Description

250 Descriptions for Value Properties of Description:

- 251 • `manufacturer`  
252 name of the manufacturer of the physical or logical part of a piece of equipment  
253 represented by this element.
- 254 • `model`  
255 model description of the physical part or logical function of a piece of equipment  
256 represented by this element.
- 257 • `serialNumber`  
258 serial number associated with a piece of equipment.
- 259 • `station`  
260 station where the physical part or logical function of a piece of equipment is located  
261 when it is part of a manufacturing unit or cell with multiple stations.

## 262 4.2 Devices

263 This section provides semantic information for the *Device* types.

### 264 4.2.1 Agent

265 *Device* composed of an *MTConnect Agent* and all its connected data sources.

266 An Agent **MUST** be provided by all *MTConnect Agent* implementations.

267 An Agent **MUST** provide notifications when devices are added or changed.

268 An Agent **MUST** provide connection information for each data source currently supply-  
269 ing data to the *MTConnect Agent*.

270 An Agent **MAY** provide information about telemetry relating to data sources.

271 An Agent **MAY** provide information about the *MTConnect Agent* resource utilization.

## 272 4.3 Component Types

273 This section provides semantic information for the types of Component.

274 Note: In the XML representation, Component entities are defined into two  
275 major categories:

- 276 • *top level* Component entities that *organizes* the most significant physi-  
277 cal or logical functions of a piece of equipment (see *Section 3.1.2 - Part*  
278 *Properties of Device*). They **MAY** also be used as *lower level* Com-  
279 ponent entities; as required. See *Section 4.4 - Component Organizer*  
280 *Types*.

- 281 • *lower level* Component entities composed of the sub-parts of the parent  
282 Component to provide more clarity and granularity to the physical or  
283 logical structure of the *top level* Component entities.

284 This section provides guidance for the most common relationships between Component  
285 types. However, all Component types **MAY** be used in any configuration, as required, to  
286 fully describe a piece of equipment.

287 As described in *Section 4.1 - Components*, Component is an abstract entity and will be  
288 always realized by a specific Component type.

### 289 4.3.1 Actuator

290 Component composed of a physical apparatus that moves or controls a mechanism or  
291 system.

292 It takes energy usually provided by air, electric current, or liquid and converts the energy  
293 into some kind of motion.

## 294 **4.3.2 Adapter**

295 `Component` that provides information about the data source for an *MTConnect Agent*.

296 It **MAY** contain connectivity state of the data source and additional telemetry about the  
297 data source and source-specific information.

## 298 **4.3.3 Auxiliary**

299 abstract `Component` composed of removable part(s) of a piece of equipment that pro-  
300 vides supplementary or extended functionality.

### 301 **4.3.3.1 Deposition**

302 `Auxiliary` that manages the addition of material or state change of material being per-  
303 formed in an additive manufacturing process.

304 For example, this could describe the portion of a piece of equipment that manages a mate-  
305 rial extrusion process or a vat polymerization process.

### 306 **4.3.3.2 Environmental**

307 `Auxiliary` that monitors, manages, or conditions the environment around or within a  
308 piece of equipment.

#### 309 **4.3.3.2.1 Heating**

310 `System` that delivers controlled amounts of heat to achieve a target temperature at a spec-  
311 ified heating rate.

312       Note: As an example, Energy Delivery Method can be either through Electric  
313       heaters or Gas burners.

#### 314 **4.3.3.2.2 Vacuum**

315 `System` that evacuates gases and liquids from an enclosed and sealed space to a controlled  
316 negative pressure or a molecular density below the prevailing atmospheric level.

317 **4.3.3.2.3 Cooling**

318 `System` that extracts controlled amounts of heat to achieve a target temperature at a spec-  
319 ified cooling rate.

320       Note: As an example, Energy Extraction Method can be via cooling water  
321       pipes running through the chamber.

322 **4.3.3.2.4 Pressure**

323 `System` that delivers compressed gas or fluid and controls the pressure and rate of pres-  
324 sure change to a desired target set-point.

325       Note: For example, Delivery Method can be a Compressed Air or N2 tank  
326       that is piped via an inlet valve to the chamber.

327 **4.3.3.3 Loader**

328 `Auxiliary` that provides movement and distribution of materials, parts, tooling, and  
329 other items to or from a piece of equipment.

330 **4.3.3.3.1 BarFeeder**

331 `Loader` that delivers bar stock to a piece of equipment.

332 **4.3.3.4 ToolingDelivery**

333 `Auxiliary` that manages, positions, stores, and delivers tooling within a piece of equip-  
334 ment.

335 **4.3.3.4.1 GangToolBar**

336 `ToolingDelivery` composed of a tool mounting mechanism that holds any number of  
337 tools.

338 Tools are located in `Station` entities. Tools are positioned for use in the manufacturing  
339 process by linearly positioning the `GangToolBar`.

340 **4.3.3.4.2 AutomaticToolChanger**

341 ToolingDelivery composed of a tool delivery mechanism that moves tools between  
342 a ToolMagazine and a *spindle* a Turret.

343 AutomaticToolChanger may also transfer tools between a location outside of a piece  
344 of equipment and a ToolMagazine or Turret.

345 **4.3.3.4.3 ToolMagazine**

346 ToolingDelivery composed of a tool storage mechanism that holds any number of  
347 tools.

348 Tools are located in Pots. Pots are moved into position to transfer tools into or out of the  
349 ToolMagazine by an AutomaticToolChanger.

350 **4.3.3.4.4 ToolRack**

351 ToolingDelivery composed of a linear or matrixed tool storage mechanism that holds  
352 any number of tools.

353 Tools are located in Station entities.

354 **4.3.3.4.5 Turret**

355 ToolingDelivery composed of a tool mounting mechanism that holds any number of  
356 tools.

357 Tools are positioned for use in the manufacturing process by rotating the Turret.

358 **4.3.3.5 WasteDisposal**

359 Auxiliary that removes manufacturing byproducts from a piece of equipment.

360 **4.3.4 Axis**

361 abstract Component composed of a motion system that provides linear or rotational mo-  
362 tion for a piece of equipment.

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

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

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

372 Examples of cartesian coordinate equipment are CNC Machine Tools, Coordinate mea-  
 373 surement machines, as specified in ISO 841, and 3D Printers. Examples of articulated  
 374 automation equipment are Robotic systems as specified in ISO 8373.

375 The following sections define the designation of names for the axes and additional guid-  
 376 ance when selecting the correct scheme to use for a given piece of equipment.

#### 377 **4.3.4.1 Cartesian Coordinate Naming Conventions**

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

381 Axes name **SHOULD** comply with ISO 841, if possible.

##### 382 **4.3.4.1.1 Linear Motion**

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

##### 387 **4.3.4.1.2 Rotary Motion**

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

#### 4.3.4.2 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.

#### 4.3.4.3 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.

#### 4.3.4.4 Linear

`Axis` that provides prismatic motion along a fixed axis.

#### 4.3.4.5 Rotary

`Axis` that provides rotation about a fixed axis.

#### 4.3.4.6 <<deprecated>>Spindle

Component that provides an axis of rotation for the purpose of rapidly rotating a part or a tool to provide sufficient surface speed for cutting operations.

`Spindle` was **DEPRECATED** in *MTConnect Version 1.1* and was replaced by `RotaryMode`.



### 420 4.3.5 Chuck

421 Component composed of a mechanism that holds a part or stock material in place.

### 422 4.3.6 Door

423 Component composed of a mechanical mechanism or closure that can cover a physical  
424 access portal into a piece of equipment allowing or restricting access to other parts of the  
425 equipment.

426 The closure can be opened or closed to allow or restrict access to other parts of the equip-  
427 ment.

428 Door **MUST** have DoorState data item to indicate if the door is OPEN, CLOSED, or  
429 UNLATCHED. A Component **MAY** contain multiple Door entities.

#### 430 4.3.6.1 Commonly Observed DataItem Types for Door

431 Table 8 lists the Commonly Observed DataItem Types for Door.

Commonly Observed DataItem Types	Multiplicity
DoorState	1

**Table 8:** Commonly Observed DataItem Types for Door

### 432 4.3.7 Lock

433 Component that physically prohibits a Device or Component from opening or oper-  
434 ating.

#### 435 4.3.7.1 Commonly Observed DataItem Types for Lock

436 Table 9 lists the Commonly Observed DataItem Types for Lock.

Commonly Observed DataItem Types	Multiplicity
LockState	0..1

**Table 9:** Commonly Observed DataItem Types for Lock**437 4.3.8 Part**

438 abstract Component composed of a *part* being processed by a piece of equipment.

**439 4.3.8.1 PartOccurrence**

440 Part that exists at a specific place and time, such as a specific instance of a bracket at a  
441 specific timestamp.

442 PartId **MUST** be defined for PartOccurrence.

**443 4.3.8.2 Commonly Observed DataItem Types for PartOccurrence**

444 *Table 10* lists the Commonly Observed DataItem Types for PartOccurrence.

Commonly Observed DataItem Types	Multiplicity
PartId	1
PartUniqueId	0..1
PartGroupId	0..1
PartKindId	0..1
PartCount	0..1
PartStatus	0..1
ProcessOccurrenceId	0..1
ProcessTime	0..1
User	0..1

**Table 10:** Commonly Observed DataItem Types for PartOccurrence**445 4.3.9 Path**

446 Component that organizes an independent operation or function within a Controller.

447 For many types of equipment, Path organizes a set of Axes, one or more Program el-  
448 ements, and the data associated with the motion of a control point as it moves through

449 space. However, it **MAY** also represent any independent function within a `Controller`  
 450 that has unique data associated with that function.

451 `Path` **SHOULD** provide an `Execution` data item to define the operational state of the  
 452 `Controller` of the piece of equipment.

453 If the `Controller` is capable of performing more than one independent operation or  
 454 function simultaneously, a separate `Path` **MUST** be used to organize the data associated  
 455 with each independent operation or function.

#### 456 4.3.9.1 Commonly Observed DataItem Types for Path

457 *Table 11* lists the Commonly Observed DataItem Types for `Path`.

Commonly Observed DataItem Types	Multiplicity
<code>Execution</code>	0..1

**Table 11:** Commonly Observed DataItem Types for Path

#### 458 4.3.10 <<deprecated>>Power

459 `Power` was **DEPRECATED** in *MTConnect Version 1.1* and was replaced by `Avail-`  
 460 `ability` data item type.

#### 461 4.3.11 Process

462 abstract `Component` composed of a manufacturing process being executed on a piece of  
 463 equipment.

##### 464 4.3.11.1 ProcessOccurrence

465 `Process` that takes place at a specific place and time, such as a specific instance of part-  
 466 milling occurring at a specific timestamp.

467 `ProcessOccurrenceId` **MUST** be defined for `ProcessOccurrence`.

### 4.3.11.2 Commonly Observed DataItem Types for ProcessOccurrence

Table 12 lists the Commonly Observed DataItem Types for ProcessOccurrence.

Commonly Observed DataItem Types	Multiplicity
ProcessOccurrenceId	1
ProcessAggregateId	0..1
ProcessTime	0..1
ProcessKindId	0..1
User	0..1
Program	0..1
PartUniqueId	0..1

**Table 12:** Commonly Observed DataItem Types for ProcessOccurrence

## 4.3.12 Resource

abstract Component composed of material or personnel involved in a manufacturing process.

### 4.3.12.1 Material

Resource composed of material that is consumed or used by the piece of equipment for production of parts, materials, or other types of goods.

#### 4.3.12.1.1 Stock

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.

### 4.3.12.2 Personnel

Resource composed of an individual or individuals who either control, support, or otherwise interface with a piece of equipment.

### 485 4.3.13 Sensor

486 Component that responds to a physical stimulus and transmits a resulting impulse or  
487 value from a sensing unit.

488 If modeling individual sensors, then sensor should be associated with the Component  
489 that the measured value is most closely associated.

490 When modeled as an Auxiliary, sensor **SHOULD** represent an integrated sensor unit  
491 system that provides signal processing, conversion, and communications. A sensor unit  
492 may have multiple *sensing elements*.

493 See SensorConfiguration for more details on the use and configuration of a Sen-  
494 sor.

#### 495 4.3.13.1 <<deprecated>>Thermostat

496 Component composed of a sensor or an instrument that measures temperature.

497 Thermostat was **DEPRECATED** in *MTConnect Version 1.2* and was replaced by  
498 Temperature.

#### 499 4.3.13.2 <<deprecated>>Vibration

500 Component composed of a sensor or an instrument that measures the amount and/or  
501 frequency of vibration within a system.

502 Vibration was **DEPRECATED** in *MTConnect Version 1.2* and was replaced by Dis-  
503 placement, Frequency etc.

### 504 4.3.14 Structure

505 Component composed of part(s) comprising the rigid bodies of the piece of equipment.

#### 506 4.3.14.1 Link

507 Structure that provides a connection between Component entities.

## 508 4.3.15 System

509 abstract `Component` that is permanently integrated into the piece of equipment.

### 510 4.3.15.1 Controller

511 System that provides regulation or management of a system or component. *Ref ISO*  
512 *16484-5:2017*

513 Typical types of controllers for a piece of equipment include CNC (Computer Numerical  
514 Control), PAC (Programmable Automation Control), IPC (Industrialized Computer), or IC  
515 (Imbedded Computer).

516 Note: In XML representation, `Controller` is a *top level* element.

### 517 4.3.15.2 Part Properties of Controller

518 *Table 13* lists the Part Properties of `Controller`.

Part Property name	Multiplicity
Path	0..*

**Table 13:** Part Properties of Controller

519 Descriptions for Part Properties of `Controller`:

- 520 • Path
- 521 `Component` that organizes an independent operation or function within a Con-  
522 troller.

### 523 4.3.15.3 Coolant

524 System that provides distribution and management of fluids that remove heat from a  
525 piece of equipment.

### 526 4.3.15.4 Dielectric

527 System that manages a chemical mixture used in a manufacturing process being per-  
528 formed at that piece of equipment.

529 For example, this could describe the dielectric system for an EDM process or the chemical  
530 bath used in a plating process.

#### 531 **4.3.15.5 Electric**

532 System composed of the main power supply for the piece of equipment that provides  
533 distribution of that power throughout the equipment.

534 The electric system will provide all the data with regard to electric current, voltage, fre-  
535 quency, etc. that applies to the piece of equipment as a functional unit. Data regard-  
536 ing electric power that is specific to a `Component` will be reported for that specific  
537 block(`Component`).

#### 538 **4.3.15.6 Enclosure**

539 System composed of a structure that is used to contain or isolate a piece of equipment or  
540 area.

541 `Enclosure` may provide information regarding access to the internal components of a  
542 piece of equipment or the conditions within the enclosure. For example, `Door` may be  
543 defined as a *lower level* `Component` or `Composition` entity of the `Enclosure`.

#### 544 **4.3.15.7 EndEffector**

545 System composed of functions that form the last link segment of a piece of equipment.

546 It is the part of a piece of equipment that interacts with the manufacturing process.

#### 547 **4.3.15.8 Feeder**

548 System that manages the delivery of materials within a piece of equipment.

549 For example, this could describe the wire delivery system for an EDM or welding process;  
550 conveying system or pump and valve system distributing material to a blending station; or  
551 a fuel delivery system feeding a furnace.

552 **4.3.15.9 Hydraulic**

553 System that provides movement and distribution of pressurized liquid throughout the  
554 piece of equipment.

555 **4.3.15.10 Lubrication**

556 System that provides distribution and management of fluids used to lubricate portions of  
557 the piece of equipment.

558 **4.3.15.11 Pneumatic**

559 System that uses compressed gasses to actuate components or do work within the piece  
560 of equipment.

561 Note: Actuation is usually performed using a cylinder.

562 **4.3.15.12 ProcessPower**

563 System composed of a power source associated with a piece of equipment that supplies  
564 energy to the manufacturing process separate from the Electric system.

565 For example, this could be the power source for an EDM machining process, an electro-  
566 plating line, or a welding system.

567 **4.3.15.13 Protective**

568 System that provides functions used to detect or prevent harm or damage to equipment  
569 or personnel.

570 Protective does not include the information relating to the Enclosure.

571 **4.3.15.14 WorkEnvelope**

572 System composed of the physical process execution space within a piece of equipment.

573 WorkEnvelope **MAY** provide information regarding the physical workspace and the  
574 conditions within that workspace.



## 575 4.4 Component Organizer Types

576 This section provides semantic information for the types of Component that are used to  
577 *organize* other Component types.

### 578 4.4.1 <<organizer>>Adapters

579 Component that *organizes* Adapter types.

### 580 4.4.2 <<organizer>>Auxiliaries

581 Component that *organizes* Auxiliary types.

### 582 4.4.3 <<organizer>>Axes

583 Component that *organizes* Axis types.

### 584 4.4.4 <<organizer>>Controllers

585 Component that *organizes* Controller entities.

### 586 4.4.5 <<organizer>>Interfaces

587 Component that *organizes* Interface types.

### 588 4.4.6 <<organizer>>Parts

589 Component that *organizes* Part types.

590 **4.4.7 <<organizer>>Processes**

591 Component that *organizes* Process types.

592 **4.4.8 <<organizer>>Resources**

593 Component that *organizes* Resource types.

594 **4.4.8.1 <<organizer>>Materials**

595 Resources that *organizes* Material types.

596 **4.4.9 <<organizer>>Structures**

597 Component that *organizes* Structure types.

598 **4.4.10 <<organizer>>Systems**

599 Component that *organizes* System types.

## 600 5 Compositions Model

601 Composition entities are used to describe the lowest level physical building blocks of  
 602 a piece of equipment contained within a Component. By referencing a specific Com-  
 603 position entity, further clarification and meaning to data associated with a specific  
 604 Component can be achieved.

605 Both Component and Composition entities are *lower level* entities representing the  
 606 sub-parts of the parent Component. However, there are distinct differences between  
 607 Component and Composition type entities.

608 Component entities may be further defined with *lower level* Component entities and  
 609 may have associated *DataItems*.

610 Composition entities represent the lowest level physical part of a piece of equipment.  
 611 They **MUST NOT** be further defined with *lower level* Component entities and they  
 612 **MUST NOT** have *DataItems* directly associated with them. They do provide additional  
 613 information that can be used to enhance the specificity of *DataItems* associated with the  
 614 parent Component.

### 615 5.1 Compositions

616 This section provides semantic information for the Composition entity.

617 See Figure 2 for the Composition model diagram.

618 Note: See *Example 3* for the XML representation of the same example.

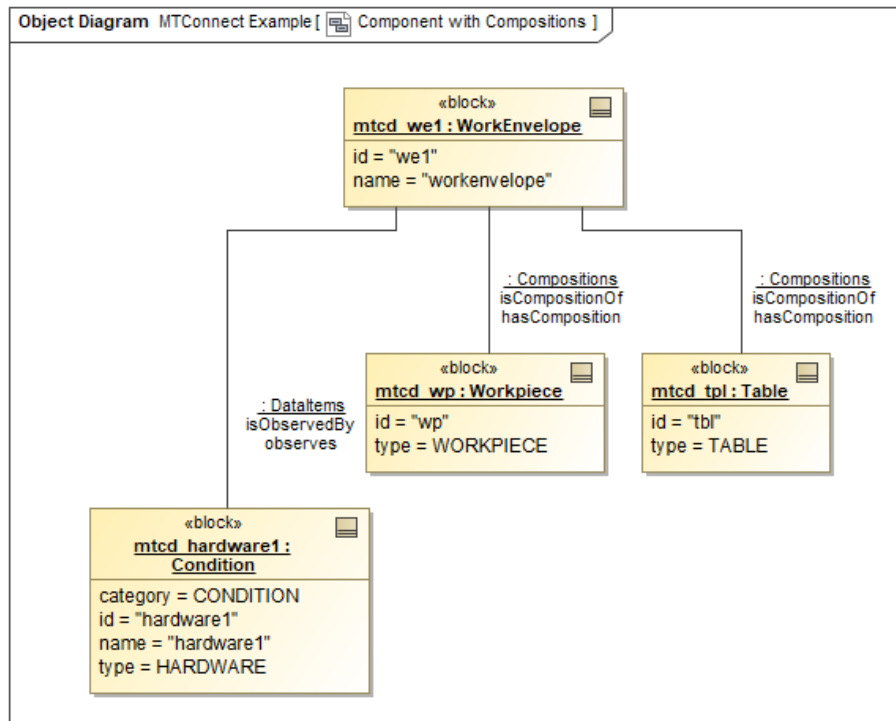
#### 619 5.1.1 Composition

620 functional part of a piece of equipment contained within a Component.

621 Composition **MUST NOT** have child Component, Composition, or DataItems  
 622 elements.

##### 623 5.1.1.1 Value Properties of Composition

624 *Table 14* lists the Value Properties of Composition.



**Figure 4:** Component with Compositions Example

Value Property name	Value Property type	Multiplicity
type	CompositionTypeEnum	1
id	ID	1
name	string	0..1
uuid	ID	0..1

**Table 14:** Value Properties of Composition

625 Descriptions for Value Properties of Composition:

- 626     • type
- 627         type of Composition. See *Section 5.2 - Composition Types*.
- 628         The value of type **MUST** be one of the CompositionTypeEnum enumeration.
- 629     • id
- 630         unique identifier for the Composition element.
- 631     • name
- 632         name of the Composition element.

- 633     • uuid
- 634         universally unique identifier for the Composition.

### 635   **5.1.1.2   Part Properties of Composition**

636   *Table 15* lists the Part Properties of Composition.

Part Property name	Multiplicity
Description	0..1
Configuration	0..1

**Table 15:** Part Properties of Composition

637   Descriptions for Part Properties of Composition:

- 638     • Description
- 639         descriptive content.
- 640         See *Section 4.1.2 - Description*.
- 641     • Configuration
- 642         technical information about an entity describing its physical layout, functional characteristics, and relationships with other entities.
- 643         See *Section 8 - Configurations Model*.

## 645   **5.2   Composition Types**

646   This section provides semantic information for the types of Composition that are currently available to describe sub-parts of a Component.

### 648   **5.2.1   Actuator**

649   Composition that moves or controls a mechanism or system.

650   It takes energy usually provided by air, electric current, or liquid and converts the energy into some kind of motion.

## 652 **5.2.2 Amplifier**

653 `Composition` composed of an electronic component or circuit that amplifies power,  
654 electric current, or voltage.

## 655 **5.2.3 Ballscrew**

656 `Composition` composed of a mechanical structure that transforms rotary motion into  
657 linear motion.

## 658 **5.2.4 Belt**

659 `Composition` composed of an endless flexible band that transmits motion for a piece of  
660 equipment or conveys materials and objects.

## 661 **5.2.5 Brake**

662 `Composition` that slows or stops a moving object by the absorption or transfer of the  
663 energy of momentum, usually by means of friction, electrical force, or magnetic force.

## 664 **5.2.6 Chain**

665 `Composition` composed of interconnected series of objects that band together and are  
666 used to transmit motion for a piece of equipment or to convey materials and objects.

## 667 **5.2.7 Chopper**

668 `Composition` that breaks material into smaller pieces.

## 669 **5.2.8 Chuck**

670 `Composition` that holds a part, stock material, or any other item in place.

**671 5.2.9 Chute**

672 `Composition` composed of an inclined channel that conveys material.

**673 5.2.10 CircuitBreaker**

674 `Composition` that interrupts an electric circuit.

**675 5.2.11 Clamp**

676 `Composition` that strengthens, support, or fastens objects in place.

**677 5.2.12 Compressor**

678 `Composition` composed of a pump or other mechanism that reduces volume and in-  
679 creases pressure of gases in order to condense the gases to drive pneumatically powered  
680 pieces of equipment.

**681 5.2.13 CoolingTower**

682 `Composition` composed of a heat exchange system that uses a fluid to transfer heat to  
683 the atmosphere.

**684 5.2.14 Door**

685 `Composition` that can cover a physical access portal into a piece of equipment allowing  
686 or restricting access to other parts of the equipment.

**687 5.2.15 Drain**

688 `Composition` that allows material to flow for the purpose of drainage from, for example,  
689 a vessel or tank.

**690 5.2.16 Encoder**

691 `Composition` that measures position.

**692 5.2.17 ExpiredPot**

693 `Pot` for a tool that is no longer usable for removal from a `ToolMagazine` or `Turret`.

**694 5.2.18 ExposureUnit**

695 `Composition` that emits a type of radiation.

**696 5.2.19 ExtrusionUnit**

697 `Composition` that dispenses liquid or powered materials.

**698 5.2.20 Fan**

699 `Composition` that produces a current of air.

**700 5.2.21 Filter**

701 `Composition` through which liquids or gases are passed to remove suspended impurities  
702 or to recover solids.

**703 5.2.22 Galvanomotor**

704 `Composition` composed of an electromechanical actuator that produces deflection of a  
705 beam of light or energy in response to electric current through its coil in a magnetic field.



**706 5.2.23 Gripper**

707 Composition that holds a part, stock material, or any other item in place.

**708 5.2.24 Hopper**

709 Composition composed of a chamber or bin in which materials are stored temporarily,  
710 typically being filled through the top and dispensed through the bottom.

**711 5.2.25 LinearPositionFeedback**

712 Composition that measures linear motion or position.

713 **DEPRECATION WARNING** : May be deprecated in the future. Recommend using  
714 Encoder.

**715 5.2.26 Motor**

716 Composition that converts electrical, pneumatic, or hydraulic energy into mechanical  
717 energy.

**718 5.2.27 Oil**

719 Composition composed of a viscous liquid.

**720 5.2.28 Pot**

721 Composition composed of a tool storage location associated with a ToolMagazine  
722 or AutomaticToolChanger.

**723 5.2.29 PowerSupply**

724 Composition that provides power to electric mechanisms.

**725 5.2.30 Pulley**

726 Composition composed of a mechanism or wheel that turns in a frame or block and  
727 serves to change the direction of or to transmit force.

**728 5.2.31 Pump**

729 Composition that raises, drives, exhausts, or compresses fluids or gases by means of a  
730 piston, plunger, or set of rotating vanes.

**731 5.2.32 Reel**

732 Composition composed of a rotary storage unit for material.

**733 5.2.33 RemovalPot**

734 Pot for a tool that has to be removed from a ToolMagazine or Turret to a location  
735 outside of the piece of equipment.

**736 5.2.34 ReturnPot**

737 Pot for a tool that has been removed from *spindle* or Turret and awaiting for return to  
738 a ToolMagazine.

**739 5.2.35 SensingElement**

740 Composition that provides a signal or measured value.

**741 5.2.36 Spreader**

742 Composition that flattens or spreading materials.

**743 5.2.37 StagingPot**

744 Pot for a tool that is awaiting transfer to a ToolMagazine or Turret from outside of  
745 the piece of equipment.

**746 5.2.38 Station**

747 Composition composed of a storage or mounting location for a tool associated with a  
748 Turret, GangToolBar, or ToolRack.

**749 5.2.39 StorageBattery**

750 Composition composed of one or more cells in which chemical energy is converted  
751 into electricity and used as a source of power.

**752 5.2.40 Switch**

753 Composition that turns on or off an electric current or makes or breaks a circuit.

**754 5.2.41 Table**

755 Composition composed of a surface for holding an object or material.

**756 5.2.42 Tank**

757 Composition composed of a receptacle or container that holds material.

**758 5.2.43 Tensioner**

759 *Composition* that provides or applies a stretch or strain to another mechanism.

**760 5.2.44 TransferArm**

761 *Composition* that physically moves a tool from one location to another.

**762 5.2.45 TransferPot**

763 Pot for a tool that is awaiting transfer from a ToolMagazine to *spindle* or Turret.

**764 5.2.46 Transformer**

765 *Composition* that transforms electric energy from a source to a secondary circuit.

**766 5.2.47 Valve**

767 *Composition* that halts or controls the flow of a liquid, gas, or other material through a  
768 passage, pipe, inlet, or outlet.

**769 5.2.48 Vat**

770 *Composition* composed of a container that holds liquid or powdered materials.

**771 5.2.49 Water**

772 *Composition* composed of  $H_2O$ .

**773 5.2.50 Wire**

774 `Composition` composed of a string like piece or filament of relatively rigid or flexible  
775 material provided in a variety of diameters.

**776 5.2.51 Workpiece**

777 `Composition` composed of an object or material on which a form of work is performed.

## 778 6 DataItems Model

779 For an `MTConnectDevices` entity, *DataItems* describe data that can be reported by a  
 780 piece of equipment and are associated with `Device` and `Component` entities. While the  
 781 *DataItems* describe the data that can be reported by a piece of equipment as an `MTCon-`  
 782 `nectDevices` entity, the actual data values are provided by the `MTConnectStreams`  
 783 entity in the *MTConnectStreams Response Document*. See *MTConnect Standard: Part 3.0*  
 784 - *Observation Information Model* for detail on the reported values.

785 Each *DataItem* **SHOULD** be modeled for the `MTConnectDevices` entity such that it  
 786 is associated with the entity that the reported data directly applies.

787 *DataItem* describes specific types of *DataItems* that represent a numeric value, a func-  
 788 tioning state, or a health status reported by a piece of equipment. *DataItem* provides a  
 789 detailed description for each *DataItem* that is reported; it defines the type of data being  
 790 reported and an array of optional attributes that further describe that data. The different  
 791 types of *DataItem* elements are defined in *Section 6.5 - DataItem Types*.

### 792 6.1 DataItems

793 This section provides semantic information for the *DataItem* entity. Figure 5 shows the  
 794 *DataItem* model.

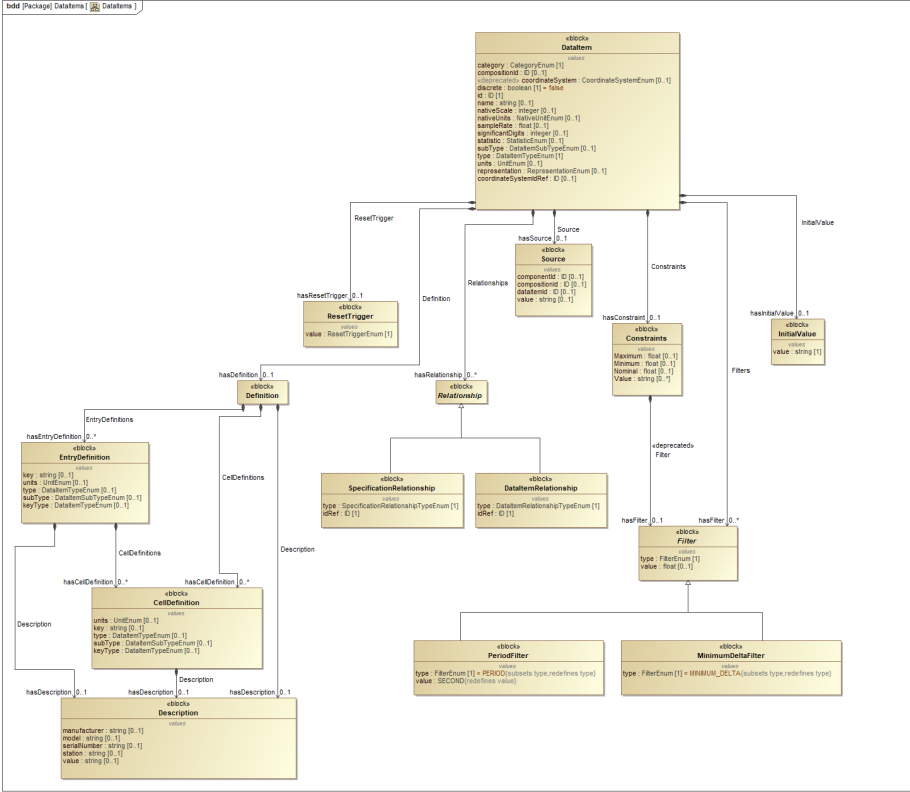
795 Note: See Figure 27 for XML schema.

#### 796 6.1.1 DataItem

797 information reported about a piece of equipment.

##### 798 6.1.1.1 Value Properties of DataItem

799 *Table 16* lists the Value Properties of *DataItem*.



### Figure 5: DataItems

Value Property name	Value Property type	Multiplicity
category	CategoryEnum	1
compositionId	ID	0..1
<<deprecated>> coordinateSystem	CoordinateSystemEnum	0..1
discrete	boolean	1
id	ID	1
name	string	0..1
nativeScale	integer	0..1
nativeUnits	NativeUnitEnum	0..1
sampleRate	float	0..1
significantDigits	integer	0..1
statistic	StatisticEnum	0..1
subType	DataItemSubTypeEnum	0..1
type	DataItemTypeEnum	1
units	UnitEnum	0..1
representation	RepresentationEnum	0..1
coordinateSystemIdRef	ID	0..1

**Table 16: Value Properties of DataItem**



## 800 Descriptions for Value Properties of `DataItem`:

- 801     • `category`  
 802         specifies the kind of information provided by a data item.  
 803         `CategoryEnum` Enumeration:  
 804             – `CONDITION`  
 805                 information about the health of a piece of equipment and its ability to function.  
 806             – `EVENT`  
 807                 discrete piece of information from the piece of equipment.  
 808             – `SAMPLE`  
 809                 continuously variable or analog data value.  
 810                 A continuous value can be measured at any point-in-time and will always pro-  
 811                 duce a result.
- 812     • `compositionId`  
 813         identifier attribute of the `Composition` that the reported data is most closely as-  
 814         sociated.
- 815     • `<<deprecated>> coordinateSystem`  
 816         for measured values relative to a coordinate system like `Position`, the coordinate  
 817         system used may be reported.  
 818         **DEPRECATED** in *Version 2.0*. Replaced by `coordinateSystemIdRef`.
- 819     • `discrete`  
 820         indication signifying whether each value reported for the *Observation* is significant  
 821         and whether duplicate values are to be suppressed.  
 822         If a value is not defined for `discrete`, the default value **MUST** be `false`.
- 823     • `id`  
 824         unique identifier for this data item.
- 825     • `name`  
 826         name of the data item.
- 827     • `nativeScale`  
 828         used to convert the reported value to represent the original measured value.

- 829 • `nativeUnits`
- 830 native units of measurement for the reported value of the data item.
- 831 <<extensible>> `NativeUnitEnum` Enumeration:
- 832     – `BAR`
- 833         pressure in Bar.
- 834     – `CENTIPOISE`
- 835         measure of viscosity.
- 836     – `DEGREE/MINUTE`
- 837         rotational velocity in degrees per minute.
- 838     – `FAHRENHEIT`
- 839         temperature in Fahrenheit.
- 840     – `FOOT`
- 841         feet.
- 842     – `FOOT/MINUTE`
- 843         feet per minute.
- 844     – `FOOT/SECOND`
- 845         feet per second.
- 846     – `FOOT/SECOND2`
- 847         acceleration in feet per second squared.
- 848     – `FOOT_3D`
- 849         point in space identified by X, Y, and Z positions and represented by a space-
- 850         delimited set of numbers each expressed in feet.
- 851     – `GALLON/MINUTE`
- 852         gallons per minute.
- 853     – `HOOR`
- 854         measurement of time in hours.
- 855     – `INCH`
- 856         inches.
- 857     – `INCH/MINUTE`
- 858         inches per minute.
- 859     – `INCH/SECOND`
- 860         inches per second.
- 861     – `INCH/SECOND2`
- 862         acceleration in inches per second squared.

- 863       – INCH\_3D
- 864       point in space identified by X, Y, and Z positions and represented by a space-
- 865       delimited set of numbers each expressed in inches.
- 866       – INCH\_POUND
- 867       measure of torque in inch pounds.
- 868       – KELVIN
- 869       measurement of temperature.
- 870       – KILOWATT
- 871       measurement in kilowatt.
- 872       – KILOWATT\_HOUR
- 873       kilowatt hours which is 3.6 mega joules.
- 874       – LITER/MINUTE
- 875       measurement of rate of flow of a fluid.
- 876       – MILLIMETER/MINUTE
- 877       velocity in millimeters per minute.
- 878       – MILLIMETER\_MERCURY
- 879       pressure in Millimeter of Mercury (mmHg).
- 880       – MINUTE
- 881       measurement of time in minutes.
- 882       – OTHER
- 883       unsupported unit.
- 884       – PASCAL/MINUTE
- 885       pascal per minute.
- 886       – POUND
- 887       US pounds.
- 888       – POUND / INCH<sup>2</sup>
- 889       pressure in pounds per square inch (PSI).
- 890       – RADIAN
- 891       angle in radians.
- 892       – RADIAN/MINUTE
- 893       velocity in radians per minute.
- 894       – RADIAN/SECOND
- 895       rotational acceleration in radian per second squared.
- 896       – RADIAN/SECOND<sup>2</sup>
- 897       rotational acceleration in radian per second squared.

- 898       – TORR
- 899       pressure in Torr.
- 900       • `sampleRate`
- 901       rate at which successive samples of a data item are recorded by a piece of equipment.
- 902       • `significantDigits`
- 903       number of significant digits in the reported value.
- 904       • `statistic`
- 905       type of statistical calculation performed on a series of data samples to provide the
- 906       reported data value.
- 907       StatisticEnum Enumeration:
- 908       – AVERAGE
- 909       mathematical average value calculated for the data item during the calculation
- 910       period.
- 911       – <<deprecated>> KURTOSIS
- 912       **DEPRECATED** in *Version 1.6*. ~~A measure of the “peakedness” of a prob-
- 913       ability distribution; i.e., the shape of the distribution curve.~~
- 914       – MAXIMUM
- 915       maximum or peak value recorded for the data item during the calculation pe-
- 916       riod.
- 917       – MEDIAN
- 918       middle number of a series of numbers.
- 919       – MINIMUM
- 920       minimum value recorded for the data item during the calculation period.
- 921       – MODE
- 922       number in a series of numbers that occurs most often.
- 923       – RANGE
- 924       difference between the maximum and minimum value of a data item during the
- 925       calculation period. Also represents Peak-to-Peak measurement in a waveform.
- 926       – `ROOT_MEAN_SQUARE`
- 927       mathematical Root Mean Square (RMS) value calculated for the data item dur-
- 928       ing the calculation period.
- 929       – `STANDARD_DEVIATION`
- 930       statistical Standard Deviation value calculated for the data item during the cal-
- 931       culation period.

- 932 • subType
- 933 sub-categorization of the data item type.
- 934 <<extensible>> DataItemSubTypeEnum Enumeration:
- 935     – ABORTED
- 936         actions or activities that were attempted, but terminated before they could be
- 937         completed.
- 938     – ABSOLUTE
- 939         relating to or derived in the simplest manner from the fundamental units or
- 940         measurements.
- 941     – ACTION
- 942         indication of the operating state of a mechanism.
- 943     – ACTIVE
- 944         relating to logic or motion program currently executing.
- 945     – ACTIVITY
- 946         phase or segment of a recipe or program.
- 947     – ACTUAL
- 948         measured or reported value of an observation.
- 949     – ALL
- 950         all actions, items, or activities being counted independent of the outcome.
- 951     – <<deprecated>> ALTERNATING
- 952         measurement of alternating voltage or current. If not specified further in statis-
- 953         tic, defaults to RMS voltage.
- 954         **DEPRECATED** in *Version 1.6*.
- 955     – AUXILIARY
- 956         when multiple locations on a piece of bar stock being feed by a bar feeder are
- 957         referenced as the indication of whether the end of that piece of bar stock has
- 958         been reached.
- 959     – A\_SCALE
- 960         A-Scale weighting factor on the frequency scale.
- 961     – BAD
- 962         actions, items, or activities being counted that do not conform to specification
- 963         or expectation.
- 964     – BATCH
- 965         group of parts produced in a batch.

- 966           – BRINELL
- 967           scale to measure the resistance to deformation of a surface.
- 968           – B\_SCALE
- 969           B-Scale weighting factor on the frequency scale.
- 970           – COMMANDED
- 971           directive value including adjustments such as an offset or overrides.
- 972           – COMPLETE
- 973           associated with the completion of an activity or event.
- 974           – CONSUMED
- 975           amount of material consumed from an object or container during a manufacturing process.
- 976
- 977           – CONTROL
- 978           state of the enabling signal or control logic that enables or disables the function
- 979           or operation of the entity.
- 980           – C\_SCALE
- 981           C-Scale weighting factor on the frequency scale.
- 982           – DELAY
- 983           elapsed time of a temporary halt of action.
- 984           – <<deprecated>> DIRECT
- 985           DC current or voltage.
- 986           **DEPRECATED** in *Version 1.6*.
- 987           – DRY\_RUN
- 988           setting or operator selection used to execute a test mode to confirm the execution
- 989           of machine functions.
- 990           – D\_SCALE
- 991           D-Scale weighting factor on the frequency scale.
- 992           – ENDED
- 993           boundary when an activity or an event terminates.
- 994           – EXPIRATION
- 995           relating to the expiration or end of useful life for a material or other physical
- 996           item.
- 997           – FAILED
- 998           actions or activities that were attempted , but failed to complete or resulted in
- 999           an unexpected or unacceptable outcome.
- 1000          – FIRST\_USE
- 1001          relating to the first use of a material or other physical item.

- 1002       – GATEWAY
- 1003       Gateway for the Component network.
- 1004       – GOOD
- 1005       actions, items, or activities being counted that conform to specification or ex-
- 1006       pectation.
- 1007       – HEAT\_TREAT
- 1008       material heat number.
- 1009       – INCREMENTAL
- 1010       relating to or derived from the last observation.
- 1011       – INSTALL\_DATE
- 1012       date the hardware or software was installed.
- 1013       – IPV4\_ADDRESS
- 1014       IPV4 network address of the Component.
- 1015       – IPV6\_ADDRESS
- 1016       IPV6 network address of the Component.
- 1017       – ISO\_STEP\_EXECUTABLE
- 1018       reference to a ISO 10303 Executable.
- 1019       – JOG
- 1020       relating to momentary activation of a function or a movement.
- 1021       **DEPRECATION WARNING:** May be deprecated in the future.
- 1022       – LATERAL
- 1023       indication of the position of a mechanism that may move in a lateral direction.
- 1024       – LEEB
- 1025       scale to measure the elasticity of a surface.
- 1026       – LENGTH
- 1027       reference to a length type tool offset variable.
- 1028       – LICENSE
- 1029       license code to validate or activate the hardware or software.
- 1030       – LINE
- 1031       state of the power source.
- 1032       – LINEAR
- 1033       direction of motion of a linear motion.
- 1034       – LOADED
- 1035       indication that the subparts of a piece of equipment are under load.

- 1036       – LOT
- 1037       group of parts tracked as a lot.
- 1038       – MACHINE\_AXIS\_LOCK
- 1039       setting or operator selection that changes the behavior of the controller on a
- 1040       piece of equipment.
- 1041       – MAC\_ADDRESS
- 1042       Media Access Control Address. The unique physical address of the network
- 1043       hardware.
- 1044       – MAIN
- 1045       relating to the primary logic or motion program currently being executed.
- 1046       – MAINTENANCE
- 1047       relating to maintenance on the piece of equipment.
- 1048       – MANUAL\_UNCLAMP
- 1049       indication of the state of an operator controlled interlock that can inhibit the
- 1050       ability to initiate an unclamp action of an electronically controlled chuck.
- 1051       – MANUFACTURE
- 1052       related to the production of a material or other physical item.
- 1053       – MANUFACTURER
- 1054       corporate identity for the maker of the hardware or software.
- 1055       – MAXIMUM
- 1056       maximum value.
- 1057       – MINIMUM
- 1058       minimum value.
- 1059       – MOHS
- 1060       scale to measure the resistance to scratching of a surface.
- 1061       – MOTION
- 1062       indication of the open or closed state of a mechanism.
- 1063       – NO\_SCALE
- 1064       no weighting factor on the frequency scale.
- 1065       – OPERATING
- 1066       piece of equipment that is powered or performing any activity.
- 1067       – OPERATION
- 1068       step of a discrete manufacturing process.
- 1069       – OPERATOR
- 1070       relating to the person currently responsible for operating the piece of equip-
- 1071       ment.



- 1072       – OPTIONAL\_STOP
- 1073       setting or operator selection that changes the behavior of the controller on a
- 1074       piece of equipment.
- 1075       – ORDER\_NUMBER
- 1076       authorization of a process occurrence.
- 1077       – OVERRIDE
- 1078       overridden value.
- 1079       – PART
- 1080       amount included in the *part*.
- 1081       – PART\_FAMILY
- 1082       group of parts having similarities in geometry, manufacturing process, and/or
- 1083       functions.
- 1084       – PART\_NAME
- 1085       word or set of words by which a part is known, addressed, or referred to.
- 1086       – PART\_NUMBER
- 1087       particular part design or model.
- 1088       – POWERED
- 1089       piece of equipment is powered and functioning or Component that are re-
- 1090       quired to remain on are powered.
- 1091       – PRIMARY
- 1092       main or most important location of a piece of bar stock.
- 1093       – PROBE
- 1094       position provided by a measurement probe.
- 1095       **DEPRECATION WARNING:** May be deprecated in the future.
- 1096       – PROCESS
- 1097       relating to production of a part or product on a piece of equipment.
- 1098       – PROCESS\_NAME
- 1099       word or set of words by which a process being executed (process occurrence)
- 1100       by the device is known, addressed, or referred to.
- 1101       – PROCESS\_PLAN
- 1102       process plan that a process occurrence belongs to.
- 1103       – PROCESS\_STEP
- 1104       step in the process plan that this occurrence corresponds to.
- 1105       – PROGRAMMED
- 1106       directive value without offsets and adjustments.

- 1107       – RADIAL
- 1108       reference to a radial type tool offset variable.
- 1109       – RAPID
- 1110       performing an operation faster or in less time than nominal rate.
- 1111       – RAW\_MATERIAL
- 1112       material that is used to produce parts.
- 1113       – RECIPE
- 1114       process as part of product production; can be a subprocess of a larger process.
- 1115       – RELEASE\_DATE
- 1116       date the hardware or software was released for general use.
- 1117       – REMAINING
- 1118       remaining measure or count of an action, object or activity.
- 1119       – REQUEST
- 1120       *request* by an Interface for a task.
- 1121       – RESPONSE
- 1122       *response* by an Interface to a *request* for a task.
- 1123       – ROCKWELL
- 1124       scale to measure the resistance to deformation of a surface.
- 1125       – ROTARY
- 1126       direction of a rotary motion using the right hand rule convention.
- 1127       – SCHEDULE
- 1128       identity of a control program that is used to specify the order of execution of
- 1129       other programs.
- 1130       – SEGMENT
- 1131       phase of a recipe process.
- 1132       – SERIAL\_NUMBER
- 1133       serial number that uniquely identifies a specific part.
- 1134       – SET\_UP
- 1135       relating to the preparation of a piece of equipment for production or restoring
- 1136       the piece of equipment to a neutral state after production.
- 1137       – SHORE
- 1138       scale to measure the resistance to deformation of a surface.
- 1139       – SINGLE\_BLOCK
- 1140       setting or operator selection that changes the behavior of the controller on a
- 1141       piece of equipment.

- 1142       – STANDARD
- 1143       standard measure of an object or an action.
- 1144       – START
- 1145       boundary when an activity or an event commences.
- 1146       – SUBNET\_MASK
- 1147       SubNet mask for the Component network.
- 1148       – SWITCHED
- 1149       indication of the activation state of a mechanism represented by a Composi-
- 1150       tion.
- 1151       – TARGET
- 1152       goal of the operation or process.
- 1153       – TARGET\_COMPLETION
- 1154       relating to the end or completion of an activity or event.
- 1155       – TOOL\_CHANGE\_STOP
- 1156       setting or operator selection that changes the behavior of the controller on a
- 1157       piece of equipment.
- 1158       – USEABLE
- 1159       remaining usable measure of an object or action.
- 1160       – UUID
- 1161       universally unique identifier as specified in ISO 11578 or RFC 4122.
- 1162       – VERSION
- 1163       version of the hardware or software.
- 1164       – VERTICAL
- 1165       indication of the position of a mechanism that may move in a vertical direction.
- 1166       – VICKERS
- 1167       scale to measure the resistance to deformation of a surface.
- 1168       – VLAN\_ID
- 1169       layer2 Virtual Local Network (VLAN) ID for the Component network.
- 1170       – WASTE
- 1171       amount discarded.
- 1172       – WIRELESS
- 1173       identifies whether the connection type is wireless.
- 1174       – WORKING
- 1175       piece of equipment performing any activity, the equipment is active and per-
- 1176       forming a function under load or not.

- 1177 • `type`
- 1178     type of data being measured. See *Section 6.5 - DataItem Types*.
- 1179     The value of `type` **MUST** be one of the `DataItemTypeEnum` enumeration.
- 1180 • `units`
- 1181     unit of measurement for the reported value of the data item.
- 1182     <<extensible>> `UnitEnum` Enumeration:
- 1183         – `AMPERE`
- 1184             amps.
- 1185         – `CELSIUS`
- 1186             degrees Celsius.
- 1187         – `COUNT`
- 1188             count of something.
- 1189         – `COUNT/SECOND`
- 1190             counts per second.
- 1191         – `CUBIC_MILLIMETER`
- 1192             geometric volume in millimeters.
- 1193         – `CUBIC_MILLIMETER/SECOND`
- 1194             change of geometric volume per second.
- 1195         – `CUBIC_MILLIMETER/SECOND2`
- 1196             change in geometric volume per second squared.
- 1197         – `DECIBEL`
- 1198             sound level.
- 1199         – `DEGREE`
- 1200             angle in degrees.
- 1201         – `DEGREE/SECOND`
- 1202             angular degrees per second.
- 1203         – `DEGREE/SECOND2`
- 1204             angular acceleration in degrees per second squared.
- 1205         – `DEGREE_3D`
- 1206             space-delimited, floating-point representation of the angular rotation in degrees
- 1207             around the X, Y, and Z axes relative to a cartesian coordinate system respec-
- 1208             tively in order as A, B, and C.
- 1209             If any of the rotations is not known, it **MUST** be zero (0).

1210	– GRAM
1211	gram.
1212	– GRAM/CUBIC_METER
1213	gram per cubic meter.
1214	– HERTZ
1215	frequency measured in cycles per second.
1216	– JOULE
1217	measurement of energy.
1218	– KILOGRAM
1219	kilograms.
1220	– LITER
1221	measurement of volume of a fluid.
1222	– LITER/SECOND
1223	liters per second.
1224	– MICRO_RADIAN
1225	measurement of tilt.
1226	– MILLIGRAM
1227	milligram.
1228	– MILLIGRAM/CUBIC_MILLIMETER
1229	milligram per cubic millimeter.
1230	– MILLILITER
1231	milliliter.
1232	– MILLIMETER
1233	millimeters.
1234	– MILLIMETER/REVOLUTION
1235	millimeters per revolution.
1236	– MILLIMETER/SECOND
1237	millimeters per second.
1238	– MILLIMETER/SECOND <sup>2</sup>
1239	acceleration in millimeters per second squared.
1240	– MILLIMETER_3D
1241	point in space identified by X, Y, and Z positions and represented by a space-
1242	delimited set of numbers each expressed in millimeters.
1243	– NEWTON
1244	force in Newtons.

1245	– NEWTON_METER
1246	torque, a unit for force times distance.
1247	– OHM
1248	measure of electrical resistance.
1249	– PASCAL
1250	pressure in Newtons per square meter.
1251	– PASCAL/SECOND
1252	pascal per second.
1253	– PASCAL_SECOND
1254	measurement of viscosity.
1255	– PERCENT
1256	percentage.
1257	– PH
1258	measure of the acidity or alkalinity of a solution.
1259	– REVOLUTION/MINUTE
1260	revolutions per minute.
1261	– REVOLUTION/SECOND
1262	rotational velocity in revolution per second.
1263	– REVOLUTION/SECOND <sup>2</sup>
1264	revolutions per second squared.
1265	– SECOND
1266	measurement of time.
1267	– SIEMENS/METER
1268	measurement of electrical conductivity.
1269	– UNIT_VECTOR_3D
1270	3D Unit Vector.
1271	Space delimited list of three floating point numbers.
1272	– VOLT
1273	volts.
1274	– VOLT_AMPERE
1275	measurement of the apparent power in an electrical circuit, equal to the product
1276	of root-mean-square (RMS) voltage and RMS current (commonly referred to
1277	as VA).
1278	– VOLT_AMPERE_REACTIVE
1279	measurement of reactive power in an AC electrical circuit (commonly referred
1280	to as VAR).

- 1281       – WATT
- 1282        watts.
- 1283       – WATT\_SECOND
- 1284        measurement of electrical energy, equal to one Joule.
- 1285   • `representation`
- 1286       description of a means to interpret data consisting of multiple data points or samples
- 1287       reported as a single value.
- 1288       If `representation` is not specified, it **MUST** be determined to be `VALUE`.
- 1289       `RepresentationEnum` Enumeration:
- 1290       – `DATA_SET`
- 1291        reported value(s) are represented as a set of *key-value pairs*.
- 1292        Each reported value in the *data set* **MUST** have a unique key.
- 1293       – `<<deprecated>> DISCRETE`
- 1294        **DEPRECATED** as a *representation* in *MTConnect Version 1.5*. Re-
- 1295        placed by the `discrete, DataItem` attribute of a `DataItem`.
- 1296       – `TABLE`
- 1297        two dimensional set of *key-value pairs* where the `Entry` represents a row, and
- 1298        the value is a set of *key-value pair* `Cell` elements.
- 1299        A *table* follows the same behavior as the *data set* for change tracking, clearing,
- 1300        and history. When an `Entry` changes, all `Cell` elements update as a single
- 1301        unit following the behavior of a *data set*.
- 1302        Note: It is best to use the `Variable DataItem` type if the `Cell`
- 1303        elements represent multiple semantic types.
- 1304        Each `Entry` in the *table* **MUST** have a unique key. Each `Cell` of each `En-`
- 1305        try in the *table* **MUST** have a unique key.
- 1306        See *Representation* in *MTConnect Standard: Part 3.0 - Observation In-*
- 1307        formation Model, for a description of `Entry` and `Cell` elements.
- 1308       – `TIME_SERIES`
- 1309        series of sampled data.
- 1310        The data is reported for a specified number of samples and each sample is
- 1311        reported with a fixed period.
- 1312       – `VALUE`
- 1313        measured value of the sample data.
- 1314        If no `representation, DataItem` is specified for a data item, the *rep-*
- 1315        resentation, `DataItem` **MUST** be determined to be `VALUE`.

- 1316 • coordinateSystemIdRef
- 1317 associated CoordinateSystem context for the DataItem.

### 1318 6.1.1.2 Reference Properties of DataItem

1319 *Table 17* lists the Reference Properties of DataItem.

Reference Property name	Multiplicity
Observation (organized by Observations)	0..*

**Table 17:** Reference Properties of DataItem

1320 Descriptions for Reference Properties of DataItem:

- 1321 • Observation
- 1322 abstract entity that provides telemetry data for a DataItem at a point in time.
- 1323 Observations groups one or more Observation entities made for the DataItem
- 1324 entity.
- 1325 Component observes DataItem entities to create Observation entities for
- 1326 the DataItem entities.
- 1327 See *MTConnect Standard: Part 3.0 - Observation Information Model* for the Ob-
- 1328 servation model.
- 1329 Note 1 to entry: In the XML representation, Observation entities
- 1330 **MUST NOT** appear in the *MTConnectDevices Response Document*.
- 1331 Note 2 to entry: In the XML representation, Observation entities
- 1332 **MUST** appear only in the *MTConnectStreams Response Document*.

### 1333 6.1.1.3 Part Properties of DataItem

1334 *Table 18* lists the Part Properties of DataItem.



Part Property name	Multiplicity
Source	0..1
Constraints	0..1
Filter (organized by Filters)	0..*
InitialValue	0..1
ResetTrigger	0..1
Definition	0..1
Relationship (organized by Relationships)	0..*

**Table 18:** Part Properties of DataItem

1335 Descriptions for Part Properties of DataItem:

1336 • Source

1337 identifies the Component, DataItem, or Composition from which a mea-  
1338 sured value originates.

1339 See *Section 6.2.2 - Source*.

1340 • Constraints

1341 organizes a set of expected values that can be reported for a DataItem.

1342 Constraints organizes a set of expected values that can be reported for the  
1343 DataItem. See *Section 6.2.7 - Constraints*.

1344 • Filter

1345 provides a means to control when an *agent* records updated information for a DataItem.

1346 Filters groups one or more Filter entities associated with the DataItem.

1347 • InitialValue

1348 starting value for a DataItem as well as the value to be set for the DataItem  
1349 after a reset event.

1350 • ResetTrigger

1351 type of event that may cause a reset to occur.

1352 • Definition

1353 defines the meaning of Entry and Cell elements associated with the DataItem  
1354 when the representation is either DATA or TABLE.

1355 See *Section 6.2.8 - Definition*.

- 1356 • Relationship
- 1357 association between two pieces of equipment that function independently but to-
- 1358 gether perform a manufacturing operation.
- 1359 Relationships groups one or more `DataItemRelationship` and `Spec-`
- 1360 `ificationRelationship`. See *Section 8.4 - Relationships*.

## 1361 6.2 Properties of `DataItem`

1362 This section provides additional semantic information for the Part Properties of `DataItem`.

1363 Note: See *Section B.2 - DataItems Schema Diagrams* for XML schema of the

1364 elements for `DataItem`.

### 1365 6.2.1 `ResetTrigger`

1366 type of event that may cause a reset to occur.

1367 `ResetTriggerEnum` Enumeration:

- 1368 • `ACTION_COMPLETE`
- 1369 observation of the `DataItem` that is measuring an action or operation is to be reset
- 1370 upon completion of that action or operation.
- 1371 • `ANNUAL`
- 1372 observation of the `DataItem` is to be reset at the end of a 12-month period.
- 1373 • `DAY`
- 1374 observation of the `DataItem` is to be reset at the end of a 24-hour period.
- 1375 • `LIFE`
- 1376 observation of the `DataItem` is not reset and accumulates for the entire life of the
- 1377 piece of equipment.
- 1378 • `MAINTENANCE`
- 1379 observation of the `DataItem` is to be reset upon completion of a maintenance
- 1380 event.

- 1381     • MONTH
- 1382       observation of the `DataItem` is to be reset at the end of a monthly period.
- 1383     • POWER\_ON
- 1384       observation of the `DataItem` is to be reset when power was applied to the piece of
- 1385       equipment after a planned or unplanned interruption of power has occurred.
- 1386     • SHIFT
- 1387       observation of the `DataItem` is to be reset at the end of a work shift.
- 1388     • WEEK
- 1389       observation of the `DataItem` is to be reset at the end of a 7-day period.

## 1390 6.2.2 Source

1391 identifies the `Component`, `DataItem`, or `Composition` from which a measured value  
 1392 originates.

1393 The value of `Source` **MUST** be `string`.

### 1394 6.2.2.1 Value Properties of Source

1395 *Table 19* lists the Value Properties of `Source`.

Value Property name	Value Property type	Multiplicity
<code>componentId</code>	ID	0..1
<code>compositionId</code>	ID	0..1
<code>dataItemId</code>	ID	0..1

**Table 19:** Value Properties of `Source`

1396 Descriptions for Value Properties of `Source`:

- 1397     • `componentId`
- 1398       identifier of the `Component` that represents the physical part of a piece of equip-
- 1399       ment where the data represented by the `DataItem` originated.
- 1400     • `compositionId`
- 1401       identifier of the `Composition` that represents the physical part of a piece of equip-
- 1402       ment where the data represented by the `DataItem` originated.

- 1403 • `dataItemId`
- 1404 identifier of the `DataItem` that represents the originally measured value of the data
- 1405 referenced by this `DataItem`.

### 1406 6.2.3 InitialValue

1407 starting value for a `DataItem` as well as the value to be set for the `DataItem` after a  
 1408 reset event.

1409 The value of `InitialValue` **MUST** be string.

### 1410 6.2.4 Filter

1411 provides a means to control when an *agent* records updated information for a `DataItem`.

1412 The value of `Filter` **MUST** be float.

#### 1413 6.2.4.1 Value Properties of Filter

1414 *Table 20* lists the Value Properties of `Filter`.

Value Property name	Value Property type	Multiplicity
type	FilterEnum	1

**Table 20:** Value Properties of Filter

1415 Descriptions for Value Properties of `Filter`:

- 1416 • type
- 1417 type of `Filter`.
- 1418 `FilterEnum` Enumeration:
  - 1419 – MINIMUM\_DELTA
  - 1420 new value **MUST NOT** be reported for a data item unless the measured value
  - 1421 has changed from the last reported value by at least the delta given as the value
  - 1422 of this element.
  - 1423 The value of `Filter` **MUST** be an absolute value using the same units as the
  - 1424 reported data.

1425           – PERIOD  
 1426           data reported for a data item is provided on a periodic basis. The PERIOD for  
 1427           reporting data is defined in the value of the Filter.  
 1428           The value of Filter **MUST** be an absolute value reported in seconds repre-  
 1429           senting the time between reported samples of the value of the data item.

## 1430 **6.2.5 MinimumDeltaFilter**

1431 new value **MUST NOT** be reported for a data item unless the measured value has changed  
 1432 from the last reported value by at least the delta given as the value of this element.  
 1433 The value of Filter **MUST** be an absolute value using the same units as the reported  
 1434 data.

## 1435 **6.2.6 PeriodFilter**

1436 data reported for a data item is provided on a periodic basis. The PERIOD for reporting  
 1437 data is defined in the value of the Filter.  
 1438 The value of Filter **MUST** be an absolute value reported in seconds representing the  
 1439 time between reported samples of the value of the data item.  
 1440 The value of PeriodFilter **MUST** be SECOND. See *Section 9.1.9 - SECOND*.

## 1441 **6.2.7 Constraints**

1442 *organizes* a set of expected values that can be reported for a DataItem.

### 1443 **6.2.7.1 Value Properties of Constraints**

1444 *Table 21* lists the Value Properties of Constraints.

Value Property name	Value Property type	Multiplicity
Maximum	float	0..1
Minimum	float	0..1
Nominal	float	0..1
Value	string	0..*

**Table 21:** Value Properties of Constraints

1445 Descriptions for Value Properties of Constraints:

- 1446     • Maximum
- 1447         numeric upper constraint.
- 1448         If the data reported for a data item is a range of numeric values, the expected value
- 1449         reported **MAY** be described with an upper limit defined by this constraint.
- 1450     • Minimum
- 1451         numeric lower constraint.
- 1452         If the data reported for a data item is a range of numeric values, the expected value
- 1453         reported **MAY** be described with a lower limit defined by this constraint.
- 1454     • Nominal
- 1455         numeric target or expected value.
- 1456     • Value
- 1457         single data value that is expected to be reported for a `DataItem`.
- 1458         Value **MUST NOT** be used in conjunction with any other Constraint ele-
- 1459         ments.

#### 1460 6.2.7.2 Part Properties of Constraints

1461 *Table 22* lists the Part Properties of Constraints.

Part Property name	Multiplicity
<<deprecated>> Filter	0..1

**Table 22:** Part Properties of Constraints

1462 Descriptions for Part Properties of Constraints:

- 1463 • Filter
- 1464 provides a means to control when an *agent* records updated information for a `DataItem`.
- 1465 **DEPRECATED** in *MTConnect Version 1.4*. Moved to the `Filters`. See *Section 6.2 - Properties of DataItem*.
- 1466

## 1467 6.2.8 Definition

- 1468 defines the meaning of `Entry` and `Cell` elements associated with the `DataItem` when  
 1469 the representation is either `DATA` or `TABLE`.

### 1470 6.2.8.1 Part Properties of Definition

- 1471 *Table 23* lists the Part Properties of Definition.

Part Property name	Multiplicity
CellDefinition (organized by CellDefinitions)	0..*
Description	0..1
EntryDefinition (organized by EntryDefinitions)	0..*

**Table 23:** Part Properties of Definition

- 1472 Descriptions for Part Properties of Definition:

- 1473 • CellDefinition
- 1474 semantic definition of a `Cell`.
- 1475 `CellDefinitions` groups one or more `CellDefinition` entities. See *Section 6.3.1 - CellDefinition*.
- 1476
- 1477 • Description
- 1478 descriptive content.
- 1479 See *Section 4.1.2 - Description*.
- 1480 • EntryDefinition
- 1481 semantic definition of an `Entry`.
- 1482 `EntryDefinitions` groups one or more `EntryDefinition` entities. See *Section 6.3.2 - EntryDefinition*.
- 1483

## 1484 6.3 Properties of Definition

1485 This section provides semantic information for the elements of the `Definition` for a  
1486 `DataItem`.

### 1487 6.3.1 CellDefinition

1488 semantic definition of a `Cell`.

#### 1489 6.3.1.1 Value Properties of CellDefinition

1490 *Table 24* lists the Value Properties of `CellDefinition`.

Value Property name	Value Property type	Multiplicity
units	UnitEnum	0..1
key	string	0..1
type	DataItemTypeEnum	0..1
subType	DataItemSubTypeEnum	0..1
keyType	DataItemTypeEnum	0..1

**Table 24:** Value Properties of `CellDefinition`

1491 Descriptions for Value Properties of `CellDefinition`:

- 1492 • units
- 1493     same as `DataItem` units. See *Section 6.1.1.1 - Value Properties of DataItem*.
- 1494     The value of `units` **MUST** be one of the `UnitEnum` enumeration.
- 1495 • key
- 1496     unique identification of the `Cell` in the `Definition`.
- 1497     The description applies to all `Cell` observations having this key.
- 1498 • type
- 1499     same as `DataItem` type. See *Section 6.5 - DataItem Types*.
- 1500     The value of `type` **MUST** be one of the `DataItemTypeEnum` enumeration.



- 1501     • subType
- 1502         same as DataItem subType. See *Section 6.1.1 - DataItem*.
- 1503         The value of subType **MUST** be one of the DataItemSubTypeEnum enumer-
- 1504         ation.
- 1505     • keyType
- 1506         DataItem type that defines the meaning of the key.
- 1507         The value of keyType **MUST** be one of the DataItemTypeEnum enumeration.

### 1508   **6.3.1.2   Part Properties of CellDefinition**

1509   *Table 25* lists the Part Properties of CellDefinition.

Part Property name	Multiplicity
Description	0..1

**Table 25:** Part Properties of CellDefinition

1510   Descriptions for Part Properties of CellDefinition:

- 1511     • Description
- 1512         descriptive content.
- 1513         See *Section 4.1.2 - Description*.

### 1514   **6.3.2   EntryDefinition**

1515   semantic definition of an Entry.

#### 1516   **6.3.2.1   Value Properties of EntryDefinition**

1517   *Table 26* lists the Value Properties of EntryDefinition.

Value Property name	Value Property type	Multiplicity
key	string	0..1
units	UnitEnum	0..1
type	DataItemTypeEnum	0..1
subType	DataItemSubTypeEnum	0..1
keyType	DataItemTypeEnum	0..1

**Table 26:** Value Properties of EntryDefinition

1518 Descriptions for Value Properties of EntryDefinition:

- 1519     • key
- 1520         unique identification of the Entry in the Definition.
- 1521         The description applies to all Entry observations having this key.
- 1522     • units
- 1523         same as DataItem units. See *Section 6.1.1.1 - Value Properties of DataItem*.
- 1524         The value of units **MUST** be one of the UnitEnum enumeration.
- 1525     • type
- 1526         same as DataItem type. See *Section 6.5 - DataItem Types*.
- 1527         The value of type **MUST** be one of the DataItemTypeEnum enumeration.
- 1528     • subType
- 1529         same as DataItem subType. See *Section 6.1.1 - DataItem*.
- 1530         The value of subType **MUST** be one of the DataItemSubTypeEnum enumer-
- 1531         ation.
- 1532     • keyType
- 1533         DataItem type that defines the meaning of the key.
- 1534         The value of keyType **MUST** be one of the DataItemTypeEnum enumeration.

### 1535 6.3.2.2 Part Properties of EntryDefinition

1536 Table 27 lists the Part Properties of EntryDefinition.

Part Property name	Multiplicity
Description	0..1
CellDefinition (organized by CellDefinitions)	0..*

**Table 27:** Part Properties of EntryDefinition

1537 Descriptions for Part Properties of EntryDefinition:

- 1538     • Description  
1539         descriptive content.  
1540         *See Section 4.1.2 - Description.*
- 1541     • CellDefinition  
1542         semantic definition of a Cell.  
1543         CellDefinitions groups one or more CellDefinition entities if the rep-  
1544         resentation, DataItem of DataItem is TABLE. *See Section 6.3.1 - CellDef-*  
1545         *inition.*

## 1546 6.4 Relationship Types for DataItem

1547 This section provides semantic information for the types of Relationship that can be  
1548 defined for a DataItem.

1549 *See Section 8.4 - Relationships* for details on the Relationship model.

### 1550 6.4.1 SpecificationRelationship

1551 Relationship that provides a semantic reference to another Specification de-  
1552 scribed by the type and idRef property.

#### 1553 6.4.1.1 Value Properties of SpecificationRelationship

1554 *Table 28* lists the Value Properties of SpecificationRelationship.

Value Property name	Value Property type	Multiplicity
type	SpecificationRelationshipTypeEnum	1
idRef	ID	1

**Table 28:** Value Properties of SpecificationRelationship

1555 Descriptions for Value Properties of SpecificationRelationship:

- 1556     • type
- 1557         specifies how the Specification is related.
- 1558         SpecificationRelationshipTypeEnum Enumeration:
- 1559             – LIMIT
- 1560             referenced Specification provides process limits.
- 1561     • idRef
- 1562         reference to the related Specification id.

## 1563 6.4.2 DataItemRelationship

1564 Relationship that provides a semantic reference to another DataItem described by

1565 the type property.

### 1566 6.4.2.1 Value Properties of DataItemRelationship

1567 Table 29 lists the Value Properties of DataItemRelationship.

Value Property name	Value Property type	Multiplicity
type	DataItemRelationshipTypeEnum	1
idRef	ID	1

**Table 29:** Value Properties of DataItemRelationship

1568 Descriptions for Value Properties of DataItemRelationship:

- 1569     • type
- 1570         specifies how the DataItem is related.
- 1571         DataItemRelationshipTypeEnum Enumeration:

- 1572       – ATTACHMENT
- 1573       reference to a `DataItem` that associates the values with an external entity.
- 1574       – COORDINATE\_SYSTEM
- 1575       referenced `DataItem` provides the `id` of the effective Coordinate System.
- 1576       – LIMIT
- 1577       referenced `DataItem` provides process limits.
- 1578       – OBSERVATION
- 1579       referenced `DataItem` provides the observed values.
- 1580       • `idRef`
- 1581       reference to the related `DataItem` `id`.

## 1582 6.5 `DataItem` Types

1583 This section provides semantic information for the types of a `DataItem`.

1584 In the MTConnect Standard, `DataItem` elements are defined and organized based upon  
 1585 the category and type attributes. The category attribute provides a high level  
 1586 grouping for `DataItem` elements based on the kind of information that is reported by  
 1587 the data item.

1588 These categories are:

- 1589       • **SAMPLE**: A **SAMPLE** reports a continuously variable or analog data value.
- 1590       • **EVENT**: An **EVENT** reports information representing a functional state, with two or  
 1591       more discrete values, associated with a component or it contains a message. The  
 1592       data provided may be a numeric value or text.
- 1593       • **CONDITION**: A **CONDITION** reports information about the health of a piece of  
 1594       equipment and its ability to function.

1595 The `type` attribute specifies the specific kind of data that is reported. For some types of  
 1596 data items, a `subType` attribute may also be used to differentiate between multiple data  
 1597 items of the same `type` where the information reported by the data item has a different,  
 1598 but related, meaning.

1599 Many types of data items provide two forms of data: a value (reported as either a **SAMPLE**  
 1600 or **EVENT**) and a health status (reported as a **CONDITION**). These `DataItem` types **MAY**  
 1601 be defined in more than one category based on the data that they report.

## 1602 6.5.1 Condition

1603 abstract DataItem that is about an entity's status regarding its ability to operate or it  
1604 provides an indication whether the data reported for the entity is within an expected range.

1605 Condition is reported differently than Sample or Event. Condition **MUST** be  
1606 reported as Normal, Warning, or Fault.

1607 All Samples **MAY** have associated Condition states. Condition states indicate  
1608 whether the value for the data is within an expected range and **MUST** be reported as  
1609 Normal, or the value is unexpected or out of tolerance for the data and a Warning or  
1610 Fault **MUST** be provided.

### 1611 6.5.1.1 Value Properties of Condition

1612 Table 30 lists the Value Properties of Condition.

Value Property name	Value Property type	Multiplicity
category	CONDITION	1
type	ConditionEnum	1

**Table 30:** Value Properties of Condition

1613 Descriptions for Value Properties of Condition:

- 1614 • type
- 1615 <<extensible>> ConditionEnum Enumeration:
- 1616 – ACTUATOR  
1617 indication of a fault associated with an actuator.
  - 1618 – COMMUNICATIONS  
1619 indication that the piece of equipment has experienced a communications fail-  
1620 ure.
  - 1621 – DATA\_RANGE  
1622 indication that the value of the data associated with a measured value or a  
1623 calculation is outside of an expected range.
  - 1624 – LOGIC\_PROGRAM  
1625 indication that an error occurred in the logic program or programmable logic  
1626 controller (PLC) associated with a piece of equipment.

- 1627       – MOTION\_PROGRAM
- 1628       indication that an error occurred in the motion program associated with a piece
- 1629       of equipment.
- 1630       – SYSTEM
- 1631       general purpose indication associated with an electronic component of a piece
- 1632       of equipment or a controller that represents a fault that is not associated with
- 1633       the operator, program, or hardware.

## 1634 6.5.2 Event

1635 abstract `DataItem` that is a discrete piece of information from a piece of equipment. It  
 1636 does not have intermediate values that vary over time.

1637 An `Event` is information that, when provided at any specific point in time, represents the  
 1638 current state of the piece of equipment.

### 1639 6.5.2.1 Value Properties of Event

1640 *Table 31* lists the Value Properties of `Event`.

Value Property name	Value Property type	Multiplicity
category	EVENT	1
type	EventEnum	1

**Table 31:** Value Properties of Event

1641 Descriptions for Value Properties of `Event`:

- 1642       • type
- 1643       <<extensible>> `EventEnum` Enumeration:
  - 1644           – ACTIVATION\_COUNT
  - 1645           accumulation of the number of times a function has attempted to, or is planned
  - 1646           to attempt to, activate or be performed.
  - 1647           – ACTIVE\_AXES
  - 1648           set of axes currently associated with a `Path` or `Controller`.
  - 1649           – ACTUATOR\_STATE
  - 1650           operational state of an apparatus for moving or controlling a mechanism or
  - 1651           system.

- 1652       – ADAPTER\_SOFTWARE\_VERSION
- 1653           originator's software version of the *adapter*.
- 1654       – ADAPTER\_URI
- 1655           Uniform Resource Identifier (URI) of the *adapter*.
- 1656       – <<deprecated>> ALARM
- 1657           **DEPRECATED:** Replaced with CONDITION category data items in Version
- 1658           1.1.0.
- 1659       – ALARM\_LIMIT
- 1660           set of limits used to trigger warning or alarm indicators.
- 1661       – APPLICATION
- 1662           application on a Component.
- 1663       – ASSET\_CHANGED
- 1664           assetId of the *Asset* that has been added or changed.
- 1665       – ASSET\_COUNT
- 1666           data set of the number of *Assets* of a given type for a *Device*.
- 1667       – ASSET\_REMOVED
- 1668           assetId of the *Asset* that has been removed.
- 1669       – AVAILABILITY
- 1670           agent's ability to communicate with the data source.
- 1671       – AXIS\_COUPLING
- 1672           describes the way the axes will be associated to each other.
- 1673           This is used in conjunction with COUPLED\_AXES to indicate the way they are
- 1674           interacting.
- 1675       – AXIS\_FEEDRATE\_OVERRIDE
- 1676           value of a signal or calculation issued to adjust the feedrate of an individual
- 1677           linear type axis.
- 1678       – AXIS\_INTERLOCK
- 1679           state of the axis lockout function when power has been removed and the axis
- 1680           is allowed to move freely.
- 1681       – AXIS\_STATE
- 1682           state of a Linear or Rotary component representing an axis.
- 1683       – BLOCK
- 1684           line of code or command being executed by a Controller entity.
- 1685       – BLOCK\_COUNT
- 1686           total count of the number of blocks of program code that have been executed
- 1687           since execution started.



- 1688       – CHUCK\_INTERLOCK
- 1689       state of an interlock function or control logic state intended to prevent the as-
- 1690       sociated *Chuck* component from being operated.
- 1691       – CHUCK\_STATE
- 1692       operating state of a mechanism that holds a part or stock material during a
- 1693       manufacturing process.
- 1694       It may also represent a mechanism that holds any other mechanism in place
- 1695       within a piece of equipment.
- 1696       – CLOCK\_TIME
- 1697       time provided by a timing device at a specific point in time.
- 1698       – <<deprecated>> CODE
- 1699       programmatic code being executed.
- 1700       **DEPRECATED** in *Version 1.1*.
- 1701       – COMPOSITION\_STATE
- 1702       operating state of a mechanism represented by a *Composition* entity.
- 1703       – CONNECTION\_STATUS
- 1704       status of the connection between an *adapter* and an *agent*.
- 1705       – CONTROLLER\_MODE
- 1706       current mode of the *Controller* component.
- 1707       – CONTROLLER\_MODE\_OVERRIDE
- 1708       setting or operator selection that changes the behavior of a piece of equipment.
- 1709       – CONTROL\_LIMIT
- 1710       set of limits used to indicate whether a process variable is stable and in control.
- 1711       – COUPLED\_AXES
- 1712       set of associated axes.
- 1713       – CYCLE\_COUNT
- 1714       accumulation of the number of times a cyclic function has attempted to, or is
- 1715       planned to attempt to execute.
- 1716       – DATE\_CODE
- 1717       time and date code associated with a material or other physical item.
- 1718       – DEACTIVATION\_COUNT
- 1719       accumulation of the number of times a function has attempted to, or is planned
- 1720       to attempt to, deactivate or cease.
- 1721       – DEVICE\_ADDED
- 1722       Universally Unique Identifier (UUID) of new device added to an *MTConnect*
- 1723       *Agent*.

- 1724       – DEVICE\_CHANGED
- 1725        UUID of the device whose *metadata* has changed.
- 1726       – DEVICE\_REMOVED
- 1727        UUID of a device removed from an *MTConnect Agent*.
- 1728       – DEVICE\_UUID
- 1729        identifier of another piece of equipment that is temporarily associated with a
- 1730        component of this piece of equipment to perform a particular function.
- 1731       – DIRECTION
- 1732        direction of motion.
- 1733       – DOOR\_STATE
- 1734        operational state of a `Door` component or composition element.
- 1735       – EMERGENCY\_STOP
- 1736        state of the emergency stop signal for a piece of equipment, controller path, or
- 1737        any other component or subsystem of a piece of equipment.
- 1738       – END\_OF\_BAR
- 1739        indication of whether the end of a piece of bar stock being feed by a bar feeder
- 1740        has been reached.
- 1741       – EQUIPMENT\_MODE
- 1742        indication that a piece of equipment, or a sub-part of a piece of equipment, is
- 1743        performing specific types of activities.
- 1744       – EXECUTION
- 1745        execution status of the `Component`.
- 1746       – FIRMWARE
- 1747        embedded software of a `Component` .
- 1748       – FIXTURE\_ID
- 1749        identifier for a fixture.
- 1750       – FUNCTIONAL\_MODE
- 1751        current intended production status of the `Component`.
- 1752       – HARDNESS
- 1753        hardness of a material.
- 1754       – HARDWARE
- 1755        hardware of a `Component`.
- 1756       – LIBRARY
- 1757        software library on a `Component`

- 1758       – <<deprecated>> LINE
- 1759       current line of code being executed.
- 1760       **DEPRECATED** in *Version 1.4.0*.
- 1761       – LINE\_LABEL
- 1762       identifier for a Block of code in a Program.
- 1763       – LINE\_NUMBER
- 1764       position of a block of program code within a control program.
- 1765       – LOAD\_COUNT
- 1766       accumulation of the number of times an operation has attempted to, or is
- 1767       planned to attempt to, load materials, parts, or other items.
- 1768       – LOCK\_STATE
- 1769       state or operating mode of a Lock.
- 1770       – MAINTENANCE\_LIST
- 1771       actions or activities to be performed in support of a piece of equipment.
- 1772       – MATERIAL
- 1773       identifier of a material used or consumed in the manufacturing process.
- 1774       – MATERIAL\_LAYER
- 1775       identifies the layers of material applied to a part or product as part of an additive
- 1776       manufacturing process.
- 1777       – MESSAGE
- 1778       information to be transferred from a piece of equipment to a client software
- 1779       application.
- 1780       – MTCONNECT\_VERSION
- 1781       reference version of the MTConnect Standard supported by the *adapter*.
- 1782       – NETWORK
- 1783       network details of a Component.
- 1784       – OPERATING\_MODE
- 1785       state of Component or Composition that describes the automatic or man-
- 1786       ual operation of the entity.
- 1787       – OPERATING\_SYSTEM
- 1788       Operating System (OS) of a Component.
- 1789       – OPERATOR\_ID
- 1790       identifier of the person currently responsible for operating the piece of equip-
- 1791       ment.
- 1792       – PALLET\_ID
- 1793       identifier for a pallet.

- 1794       – PART\_COUNT
- 1795       aggregate count of parts.
- 1796       – PART\_COUNT\_TYPE
- 1797       interpretation of PART\_COUNT.
- 1798       – PART\_DETECT
- 1799       indication designating whether a part or work piece has been detected or is
- 1800       present.
- 1801       – PART\_GROUP\_ID
- 1802       identifier given to a collection of individual parts.
- 1803       – PART\_ID
- 1804       identifier of a part in a manufacturing operation.
- 1805       – PART\_KIND\_ID
- 1806       identifier given to link the individual occurrence to a class of parts, typically
- 1807       distinguished by a particular part design.
- 1808       – <<deprecated>> PART\_NUMBER
- 1809       identifier of a part or product moving through the manufacturing process.
- 1810       **DEPRECATED** in *Version 1.7*. PART\_NUMBER is now a subType of PART\_KIND\_ID.
- 1811       – PART\_PROCESSING\_STATE
- 1812       particular condition of the part occurrence at a specific time.
- 1813       – PART\_STATUS
- 1814       state or condition of a part.
- 1815       – PART\_UNIQUE\_ID
- 1816       identifier given to a distinguishable, individual part.
- 1817       – PATH\_FEEDRATE\_OVERRIDE
- 1818       value of a signal or calculation issued to adjust the feedrate for the axes asso-
- 1819       ciated with a Path component that may represent a single axis or the coordi-
- 1820       nated movement of multiple axes.
- 1821       – PATH\_MODE
- 1822       describes the operational relationship between a Path entity and another Path
- 1823       entity for pieces of equipment comprised of multiple logical groupings of con-
- 1824       trolled axes or other logical operations.
- 1825       – POWER\_STATE
- 1826       indication of the status of the source of energy for an entity to allow it to per-
- 1827       form its intended function or the state of an enabling signal providing permis-
- 1828       sion for the entity to perform its functions.

- 1829       – <<deprecated>> POWER\_STATUS
- 1830       status of the Component.
- 1831       **DEPRECATED** in *Version 1.1.0*.
- 1832       – PROCESS\_AGGREGATE\_ID
- 1833       identifier given to link the individual occurrence to a group of related occur-
- 1834       rences, such as a process step in a process plan.
- 1835       – PROCESS\_KIND\_ID
- 1836       identifier given to link the individual occurrence to a class of processes or
- 1837       process definition.
- 1838       – PROCESS\_OCCURRENCE\_ID
- 1839       identifier of a process being executed by the device.
- 1840       – PROCESS\_STATE
- 1841       particular condition of the process occurrence at a specific time.
- 1842       – PROCESS\_TIME
- 1843       time and date associated with an activity or event.
- 1844       – PROGRAM
- 1845       name of the logic or motion program being executed by the Controller
- 1846       component.
- 1847       – PROGRAM\_COMMENT
- 1848       comment or non-executable statement in the control program.
- 1849       – PROGRAM\_EDIT
- 1850       indication of the status of the Controller components program editing
- 1851       mode.
- 1852       A program may be edited while another is executed.
- 1853       – PROGRAM\_EDIT\_NAME
- 1854       name of the program being edited.
- 1855       This is used in conjunction with ProgramEdit when in ACTIVE state.
- 1856       – PROGRAM\_HEADER
- 1857       non-executable header section of the control program.
- 1858       – PROGRAM\_LOCATION
- 1859       URI for the source file associated with Program.
- 1860       – PROGRAM\_LOCATION\_TYPE
- 1861       defines whether the logic or motion program defined by Program is being
- 1862       executed from the local memory of the controller or from an outside source.

- 1863       – PROGRAM\_NEST\_LEVEL
- 1864           indication of the nesting level within a control program that is associated with
- 1865           the code or instructions that is currently being executed.
- 1866       – ROTARY\_MODE
- 1867           current operating mode for a Rotary type axis.
- 1868       – ROTARY\_VELOCITY\_OVERRIDE
- 1869           percentage change to the velocity of the programmed velocity for a Rotary
- 1870           axis.
- 1871       – ROTATION
- 1872           three space angular rotation relative to a coordinate system.
- 1873       – SENSOR\_ATTACHMENT
- 1874           *attachment* between a sensor and an entity.
- 1875       – SERIAL\_NUMBER
- 1876           serial number associated with a Component, Asset, or Device.
- 1877       – SPECIFICATION\_LIMIT
- 1878           set of limits defining a range of values designating acceptable performance for
- 1879           a variable.
- 1880       – SPINDLE\_INTERLOCK
- 1881           indication of the status of the spindle for a piece of equipment when power has
- 1882           been removed and it is free to rotate.
- 1883       – TOOL\_ASSET\_ID
- 1884           identifier of an individual tool asset.
- 1885       – TOOL\_GROUP
- 1886           identifier for the tool group associated with a specific tool. Commonly used to
- 1887           designate spare tools.
- 1888       – <<deprecated>> TOOL\_ID
- 1889           identifier of the tool currently in use for a given Path.
- 1890           **DEPRECATED** in *Version 1.2.0*. See TOOL\_ASSET\_ID.
- 1891       – TOOL\_NUMBER
- 1892           identifier assigned by the Controller component to a cutting tool when in
- 1893           use by a piece of equipment.
- 1894       – TOOL\_OFFSET
- 1895           reference to the tool offset variables applied to the active cutting tool associated
- 1896           with a Path in a Controller type component.

- 1897       – `TRANSFER_COUNT`
- 1898       accumulation of the number of times an operation has attempted to, or is
- 1899       planned to attempt to, transfer materials, parts, or other items from one lo-
- 1900       cation to another.
- 1901       – `TRANSLATION`
- 1902       three space linear translation relative to a coordinate system.
- 1903       – `UNLOAD_COUNT`
- 1904       accumulation of the number of times an operation has attempted to, or is
- 1905       planned to attempt to, unload materials, parts, or other items.
- 1906       – `USER`
- 1907       identifier of the person currently responsible for operating the piece of equip-
- 1908       ment.
- 1909       – `VALVE_STATE`
- 1910       state of a valve is one of open, closed, or transitioning between the states.
- 1911       – `VARIABLE`
- 1912       data whose meaning may change over time due to changes in the operation of
- 1913       a piece of equipment or the process being executed on that piece of equipment.
- 1914       – `WAIT_STATE`
- 1915       indication of the reason that `Execution` is reporting a value of `WAIT`.
- 1916       – `WIRE`
- 1917       identifier for the type of wire used as the cutting mechanism in Electrical Dis-
- 1918       charge Machining or similar processes.
- 1919       – `WORKHOLDING_ID`
- 1920       identifier for the current workholding or part clamp in use by a piece of equip-
- 1921       ment.
- 1922       – `WORK_OFFSET`
- 1923       offset variables for a work piece or part associated with a `Path` in a `Con-`
- 1924       troller

### 1925 6.5.3 Sample

- 1926   abstract `DataItem` that is continuously changing or analog data value.
- 1927   This data can be measured at any point-in-time and will always produce a result.
- 1928   The `units` for `Sample` **MUST** always be specified.

### 6.5.3.1 Value Properties of Sample

Table 32 lists the Value Properties of Sample.

Value Property name	Value Property type	Multiplicity
category	SAMPLE	1
type	SampleEnum	1

**Table 32:** Value Properties of Sample

Descriptions for Value Properties of Sample:

- type

<<extensible>> SampleEnum Enumeration:

- ACCELERATION

positive rate of change of velocity.

- ACCUMULATED\_TIME

accumulated time for an activity or event.

- <<deprecated>> AMPERAGE

strength of electrical current.

**DEPRECATED** in *Version 1.6*. Replaced by AMPERAGE\_AC and AMPERAGE\_DC.

- AMPERAGE\_AC

electrical current that reverses direction at regular short intervals.

- AMPERAGE\_DC

electric current flowing in one direction only.

- ANGLE

angular position.

- ANGULAR\_ACCELERATION

positive rate of change of angular velocity.

- ANGULAR\_DECELERATION

negative rate of change of angular velocity.

- ANGULAR\_VELOCITY

rate of change of angular position.

- ASSET\_UPDATE\_RATE

average rate of change of values for assets in the MTConnect streams.

The average is computed over a rolling window defined by the implementation.



1957	–	AXIS_FEEDRATE
1958		feedrate of a linear axis.
1959	–	CAPACITY_FLUID
1960		fluid capacity of an object or container.
1961	–	CAPACITY_SPATIAL
1962		geometric capacity of an object or container.
1963	–	CONCENTRATION
1964		percentage of one component within a mixture of components.
1965	–	CONDUCTIVITY
1966		ability of a material to conduct electricity.
1967	–	CUTTING_SPEED
1968		speed difference (relative velocity) between the cutting mechanism and the
1969		surface of the workpiece it is operating on.
1970	–	DECELERATION
1971		negative rate of change of velocity.
1972	–	DENSITY
1973		volumetric mass of a material per unit volume of that material.
1974	–	DEPOSITION_ACCELERATION_VOLUMETRIC
1975		rate of change in spatial volume of material deposited in an additive manufac-
1976		turing process.
1977	–	DEPOSITION_DENSITY
1978		density of the material deposited in an additive manufacturing process per unit
1979		of volume.
1980	–	DEPOSITION_MASS
1981		mass of the material deposited in an additive manufacturing process.
1982	–	DEPOSITION_RATE_VOLUMETRIC
1983		rate at which a spatial volume of material is deposited in an additive manufac-
1984		turing process.
1985	–	DEPOSITION_VOLUME
1986		spatial volume of material to be deposited in an additive manufacturing pro-
1987		cess.
1988	–	DIAMETER
1989		dimension of a diameter.
1990	–	DISPLACEMENT
1991		change in position of an object.

1992	– ELECTRICAL_ENERGY
1993	Wattage used or generated by a component over an interval of time.
1994	– EQUIPMENT_TIMER
1995	amount of time a piece of equipment or a sub-part of a piece of equipment has
1996	performed specific activities.
1997	– FILL_LEVEL
1998	amount of a substance remaining compared to the planned maximum amount
1999	of that substance.
2000	– FLOW
2001	rate of flow of a fluid.
2002	– FREQUENCY
2003	number of occurrences of a repeating event per unit time.
2004	– <<deprecated>> GLOBAL_POSITION
2005	position in three-dimensional space.
2006	<b>DEPRECATED</b> in Version 1.1.
2007	– HUMIDITY_ABSOLUTE
2008	amount of water vapor expressed in grams per cubic meter.
2009	– HUMIDITY_RELATIVE
2010	amount of water vapor present expressed as a percent to reach saturation at the
2011	same temperature.
2012	– HUMIDITY_SPECIFIC
2013	ratio of the water vapor present over the total weight of the water vapor and air
2014	present expressed as a percent.
2015	– LENGTH
2016	length of an object.
2017	– <<deprecated>> LEVEL
2018	level of a resource.
2019	<b>DEPRECATED</b> in <i>Version 1.2</i> . See FILL_LEVEL.
2020	– LINEAR_FORCE
2021	<i>force</i> applied to a mass in one direction only.
2022	– LOAD
2023	actual versus the standard rating of a piece of equipment.
2024	– MASS
2025	mass of an object(s) or an amount of material.

- 2026       – OBSERVATION\_UPDATE\_RATE
- 2027       average rate of change of values for data items in the MTConnect streams. The
- 2028       average is computed over a rolling window defined by the implementation.
- 2029       – OPENNESS
- 2030       percentage open where 100% is fully open and 0% is fully closed.
- 2031       – ORIENTATION
- 2032       measured or calculated orientation of a plane or vector relative to a cartesian
- 2033       coordinate system.
- 2034       – PATH\_FEEDRATE
- 2035       feedrate for the axes, or a single axis, associated with a Path component.
- 2036       – PATH\_FEEDRATE\_PER\_REVOLUTION
- 2037       feedrate for the axes, or a single axis.
- 2038       – PATH\_POSITION
- 2039       measured or calculated position of a control point associated with a Con-
- 2040       troller entity, or Path entity if provided, of a piece of equipment.
- 2041       – PH
- 2042       acidity or alkalinity of a solution.
- 2043       – POSITION
- 2044       measured or calculated position of a Component element as reported by a
- 2045       piece of equipment.
- 2046       – POWER\_FACTOR
- 2047       ratio of real power flowing to a load to the apparent power in that AC circuit.
- 2048       – PRESSURE
- 2049       force per unit area measured relative to atmospheric pressure.
- 2050       Commonly referred to as gauge pressure.
- 2051       – PRESSURE\_ABSOLUTE
- 2052       force per unit area measured relative to a vacuum.
- 2053       – PRESSURIZATION\_RATE
- 2054       change of pressure per unit time.
- 2055       – PROCESS\_TIMER
- 2056       amount of time a piece of equipment has performed different types of activities
- 2057       associated with the process being performed at that piece of equipment.
- 2058       – RESISTANCE
- 2059       degree to which a substance opposes the passage of an electric current.
- 2060       – ROTARY\_VELOCITY
- 2061       rotational speed of a rotary axis.

- 2062       – SOUND\_LEVEL
- 2063       sound level or sound pressure level relative to atmospheric pressure.
- 2064       – <<deprecated>> SPINDLE\_SPEED
- 2065       rotational speed of the rotary axis.
- 2066       **DEPRECATED** in *Version 1.2*. Replaced by ROTARY\_VELOCITY.
- 2067       – STRAIN
- 2068       amount of deformation per unit length of an object when a load is applied.
- 2069       – TEMPERATURE
- 2070       degree of hotness or coldness measured on a definite scale.
- 2071       – TENSION
- 2072       force that stretches or elongates an object.
- 2073       – TILT
- 2074       angular displacement.
- 2075       – TORQUE
- 2076       turning force exerted on an object or by an object.
- 2077       – VELOCITY
- 2078       rate of change of position of a Component.
- 2079       – VISCOSITY
- 2080       fluid's resistance to flow.
- 2081       – <<deprecated>> VOLTAGE
- 2082       electrical potential between two points.
- 2083       **DEPRECATED** in *Version 1.6*. Replaced by VOLTAGE\_AC and VOLTAGE\_DC.
- 2084       – VOLTAGE\_AC
- 2085       electrical potential between two points in an electrical circuit in which the cur-
- 2086       rent periodically reverses direction.
- 2087       – VOLTAGE\_DC
- 2088       electrical potential between two points in an electrical circuit in which the cur-
- 2089       rent is unidirectional.
- 2090       – VOLT\_AMPERE
- 2091       apparent power in an electrical circuit, equal to the product of root-mean-
- 2092       square (RMS) voltage and RMS current (commonly referred to as VA).
- 2093       – VOLT\_AMPERE\_REACTIVE
- 2094       reactive power in an AC electrical circuit (commonly referred to as VAR).
- 2095       – VOLUME\_FLUID
- 2096       fluid volume of an object or container.

- 2097       – VOLUME\_SPATIAL
- 2098       geometric volume of an object or container.
- 2099       – WATTAGE
- 2100       power flowing through or dissipated by an electrical circuit or piece of equip-
- 2101       ment.
- 2102       – X\_DIMENSION
- 2103       dimension of an entity relative to the X direction of the referenced coordinate
- 2104       system.
- 2105       – Y\_DIMENSION
- 2106       dimension of an entity relative to the Y direction of the referenced coordinate
- 2107       system.
- 2108       – Z\_DIMENSION
- 2109       dimension of an entity relative to the Z direction of the referenced coordinate
- 2110       system.

## 2111 7 References Model

2112 References *organizes* pointers to information defined elsewhere within the MTCon-  
 2113 nectDevices entity for a piece of equipment. It is an efficient method to associate  
 2114 information with an element without duplicating any of the data or structure.

### 2115 7.1 References

2116 This section provides semantic information for the Reference entity. Figure 6 shows  
 2117 the Reference model.

2118 Reference may be modeled as part of a Device, Component or Interface type.

2119 Note: See *Section B.3 - References Schema Diagrams* for XML schema of  
 2120 Reference and its types.

#### 2121 7.1.1 Reference

2122 pointer to information that is associated with another entity defined elsewhere in the MT-  
 2123 ConnectDevices entity for a piece of equipment.

2124 Reference is an abstract entity and will be realized by a specific Reference type for  
 2125 an MTConnectDevices entity. See *Section 7.1.3 - ComponentRef* and *Section 7.1.2 -*  
 2126 *DataItemRef*.

##### 2127 7.1.1.1 Value Properties of Reference

2128 Table 33 lists the Value Properties of Reference.

Value Property name	Value Property type	Multiplicity
idRef	ID	1
name	string	0..1

**Table 33:** Value Properties of Reference

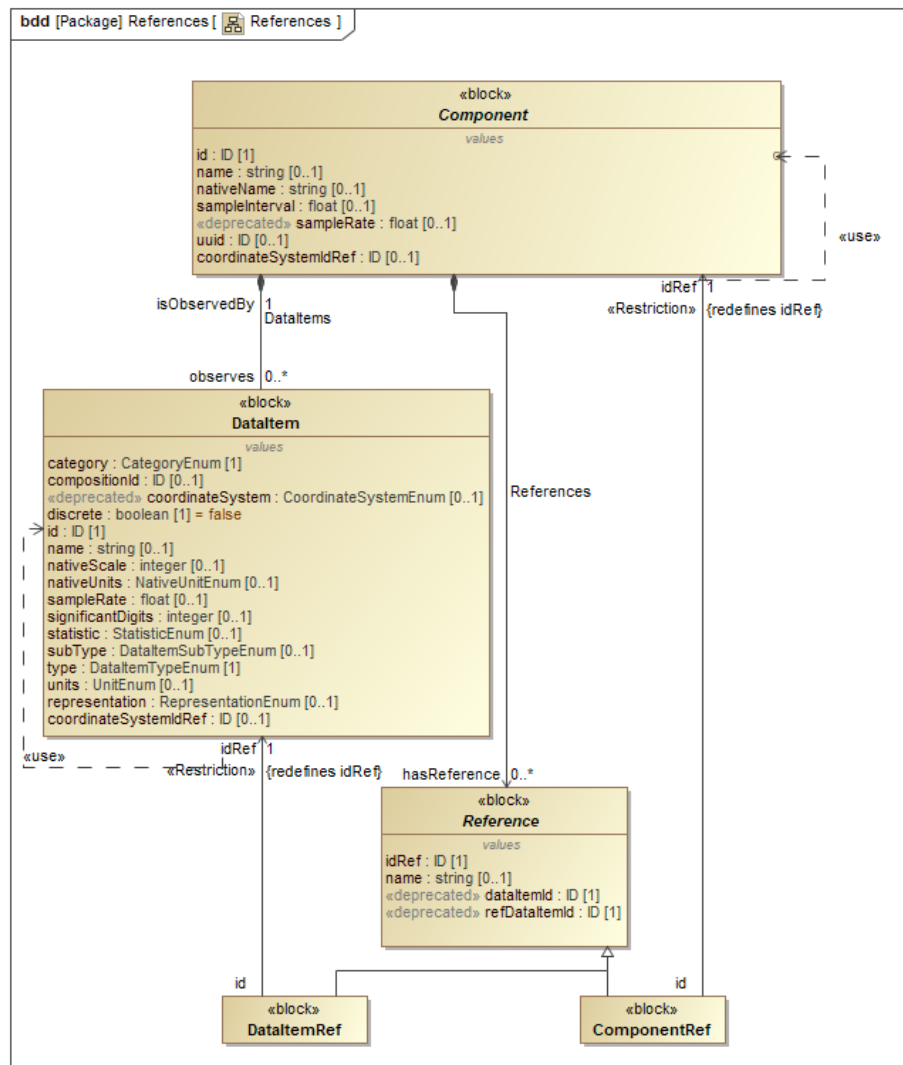


Figure 6: References

2129 Descriptions for Value Properties of Reference:

- 2130 • idRef
- 2131 pointer to the id of an entity that contains the information to be associated with this
- 2132 entity.
- 2133 • name
- 2134 name of an element or a piece of equipment.

### 2135 **7.1.2 DataItemRef**

2136 Reference that is a pointer to a `DataItem` associated with another entity defined for  
2137 a piece of equipment.

2138 `DataItemRef` allows the data associated with a `DataItem` defined in another entity to  
2139 be directly associated with this entity.

### 2140 **7.1.3 ComponentRef**

2141 Reference that is a pointer to all of the information associated with another entity de-  
2142 fined for a piece of equipment.

2143 `ComponentRef` allows all of the information of (*lower level* `Component` entities that  
2144 is associated with the other entity to be directly associated with this entity.



## 2145 8 Configurations Model

2146 This section provides semantic information for the Configuration entity that is used  
2147 to model technical information about a Component.

### 2148 8.1 Configurations

2149 Figure 7 shows the abstract Configuration and its types.

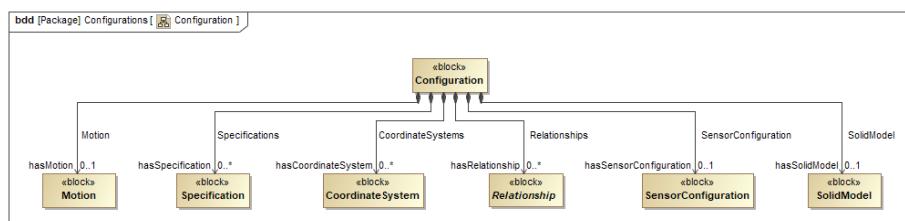


Figure 7: Configuration

2150 Note: See Figure 35 for XML schema.

#### 2151 8.1.1 Configuration

2152 technical information about an entity describing its physical layout, functional character-  
2153 istics, and relationships with other entities.

##### 2154 8.1.1.1 Part Properties of Configuration

2155 Table 34 lists the Part Properties of Configuration.

Part Property name	Multiplicity
SolidModel	0..1
SensorConfiguration	0..1
Motion	0..1
Relationship (organized by Relationships)	0..*
CoordinateSystem (organized by CoordinateSystems)	0..*
Specification (organized by Specifications)	0..*

**Table 34:** Part Properties of Configuration

#### 2156 Descriptions for Part Properties of Configuration:

- 2157 • SolidModel
- 2158 references to a file with the three-dimensional geometry of the Component or
- 2159 Composition.
- 2160 See *Section 8.6 - SolidModel*.
- 2161 • SensorConfiguration
- 2162 configuration for a Sensor.
- 2163 See SensorConfiguration.
- 2164 • Motion
- 2165 movement of the component relative to a coordinate system.
- 2166 See *Section 8.3 - Motion*.
- 2167 • Relationship
- 2168 association between two pieces of equipment that function independently but to-
- 2169 gether perform a manufacturing operation.
- 2170 Relationships groups one or more Relationship types. See *Section 8.4 -*
- 2171 *Relationships*.
- 2172 • CoordinateSystem
- 2173 reference system that associates a unique set of n parameters with each point in an
- 2174 n-dimensional space. *Ref ISO 10303-218:2004*
- 2175 CoordinateSystems groups one or more CoordinateSystem entities. See
- 2176 *Section 8.2 - CoordinateSystems*.
- 2177 • Specification
- 2178 design characteristics for a piece of equipment.

2179 Specifications groups one or more Specification entities. See *Sec-*  
 2180 *tion 8.7 - Specifications.*

## 2181 8.2 CoordinateSystems

2182 This section provides semantic information for the CoordinateSystem entity.

2183 Note: See Figure 36 for XML schema.

2184 Note: See *Example 4* for the XML representation of the same example.

### 2185 8.2.1 CoordinateSystem

2186 reference system that associates a unique set of n parameters with each point in an n-  
 2187 dimensional space. *Ref ISO 10303-218:2004*

2188 At most only one of Origin or Transformation **MUST** be defined for a Coordi-  
 2189 nateSystem.

#### 2190 8.2.1.1 Value Properties of CoordinateSystem

2191 *Table 35* lists the Value Properties of CoordinateSystem.

Value Property name	Value Property type	Multiplicity
id	ID	1
name	string	0..1
nativeName	string	0..1
parentIdRef	ID	0..1
type	CoordinateSystemTypeEnum	1
Description	string	0..1

**Table 35:** Value Properties of CoordinateSystem

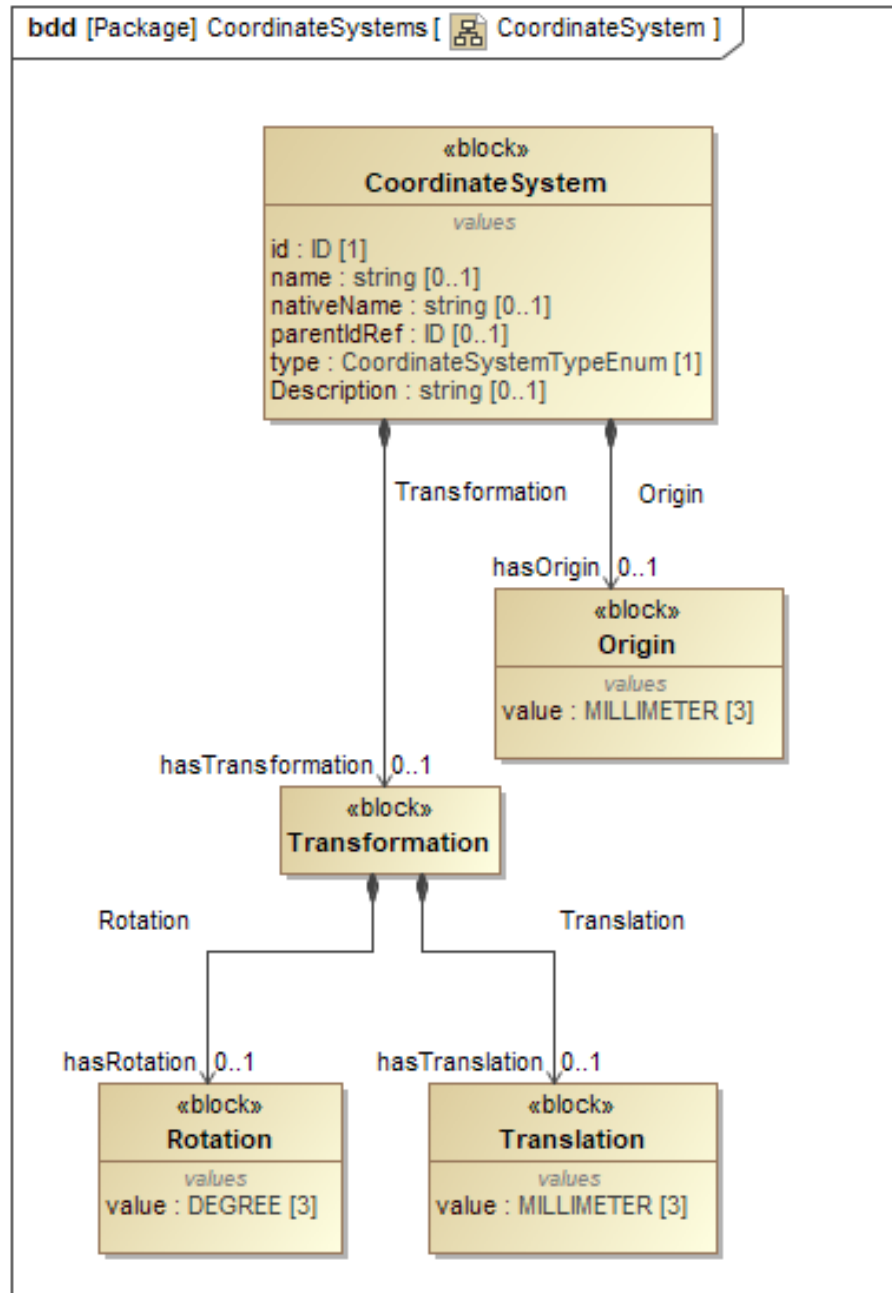


Figure 8: CoordinateSystem

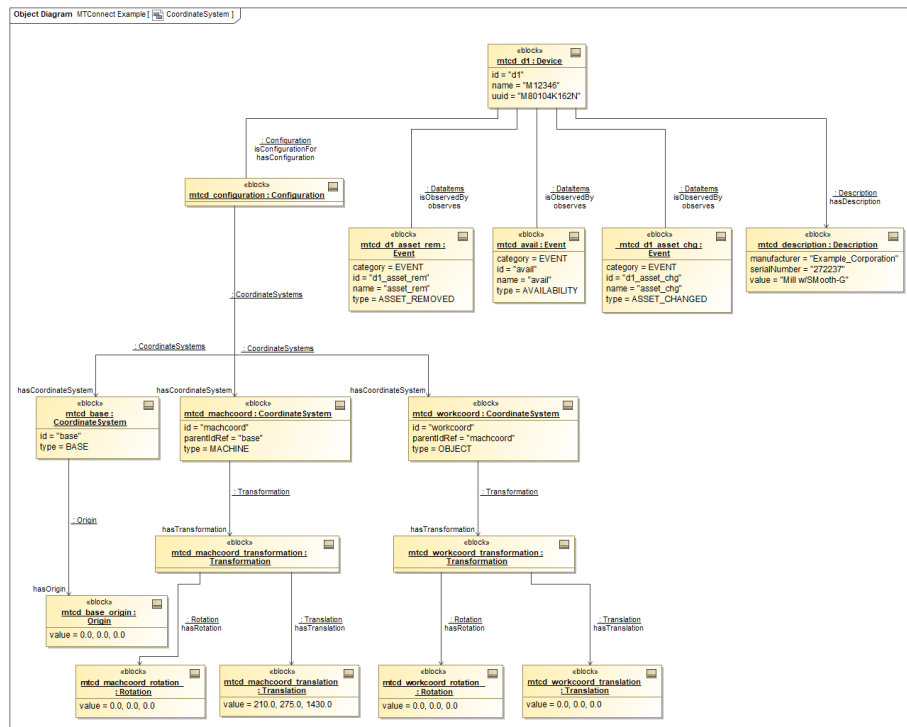


Figure 9: CoordinateSystem Example

2192 Descriptions for Value Properties of `CoordinateSystem`:

- 2193     • `id`
- 2194         unique identifier for this element.
- 2195     • `name`
- 2196         name of the coordinate system.
- 2197     • `nativeName`
- 2198         manufacturer's name or users name for the coordinate system.
- 2199     • `parentIdRef`
- 2200         pointer to the `id` attribute of the parent `CoordinateSystem`.
- 2201     • `type`
- 2202         type of coordinate system.

2203 `CoordinateSystemTypeEnum` Enumeration:

- 2204     – `BASE`
- 2205         coordinate system referenced to the base mounting surface. *Ref ISO 9787:2013*
- 2206         A base mounting surface is a connection surface between the arm and its sup-
- 2207         porting structure. *Ref ISO 9787:2013*
- 2208         For non-robotic devices, it is the connection surface between the device and its
- 2209         supporting structure.
- 2210     – `CAMERA`
- 2211         coordinate system referenced to the sensor which monitors the site of the task.
- 2212         *Ref ISO 9787:2013*
- 2213     – `MACHINE`
- 2214         coordinate system referenced to the home position and orientation of the pri-
- 2215         mary axes of a piece of equipment.
- 2216     – `MECHANICAL_INTERFACE`
- 2217         coordinate system referenced to the mechanical interface. *Ref ISO 9787:2013*
- 2218     – `MOBILE_PLATFORM`
- 2219         coordinate system referenced to one of the components of a mobile platform.
- 2220         *Ref ISO 8373:2012*
- 2221     – `OBJECT`
- 2222         coordinate system referenced to the object. *Ref ISO 9787:2013*

- 2223       – TASK
- 2224       coordinate system referenced to the site of the task. *Ref ISO 9787:2013*
- 2225       – TOOL
- 2226       coordinate system referenced to the tool or to the end effector attached to the
- 2227       mechanical interface. *Ref ISO 9787:2013*
- 2228       – WORLD
- 2229       stationary coordinate system referenced to earth, which is independent of the
- 2230       robot motion. *Ref ISO 9787:2013*
- 2231       For non-robotic devices, stationary coordinate system referenced to earth, which
- 2232       is independent of the motion of a piece of equipment.

- 2233       • Description
- 2234       natural language description of the `CoordinateSystem`.

### 2235   **8.2.1.2   Part Properties of `CoordinateSystem`**

2236   *Table 36* lists the Part Properties of `CoordinateSystem`.

Part Property name	Multiplicity
Origin	0..1
Transformation	0..1

**Table 36:** Part Properties of `CoordinateSystem`

2237   Descriptions for Part Properties of `CoordinateSystem`:

- 2238       • Origin
- 2239       coordinates of the origin position of a coordinate system.
- 2240       See *Section 8.2.2 - Origin*.
- 2241       • Transformation
- 2242       process of transforming to the origin position of the coordinate system from a parent
- 2243       coordinate system using `Translation` and `Rotation`.
- 2244       See *Section 8.2.3 - Transformation*.

## 2245 8.2.2 Origin

2246 coordinates of the origin position of a coordinate system.

2247 The value of `Origin` **MUST** be reported in `MILLIMETER_3D`.

## 2248 8.2.3 Transformation

2249 process of transforming to the origin position of the coordinate system from a parent co-  
2250 ordinate system using `Translation` and `Rotation`.

2251 At a minimum, a `Translation` or a `Rotation` **MUST** be defined for a `Transfor-`  
2252 `mation`.

### 2253 8.2.3.1 Part Properties of Transformation

2254 *Table 37* lists the Part Properties of `Transformation`.

Part Property name	Multiplicity
<code>Translation</code>	0..1
<code>Rotation</code>	0..1

**Table 37:** Part Properties of Transformation

2255 Descriptions for Part Properties of `Transformation`:

2256 • `Translation`

2257 translations along X, Y, and Z axes are expressed as x,y, and z respectively within a  
2258 3-dimensional vector.

2259 See *Section 8.2.5 - Translation*.

2260 • `Rotation`

2261 rotations about X, Y, and Z axes are expressed in A, B, and C respectively within a  
2262 3-dimensional vector.

2263 See *Section 8.2.4 - Rotation*.



## 2264 8.2.4 Rotation

2265 rotations about X, Y, and Z axes are expressed in A, B, and C respectively within a 3-  
2266 dimensional vector.

2267 The value of `Rotation` **MUST** be reported in `DEGREE_3D`.

## 2268 8.2.5 Translation

2269 translations along X, Y, and Z axes are expressed as x,y, and z respectively within a 3-  
2270 dimensional vector.

2271 The value of `Translation` **MUST** be reported in `MILLIMETER_3D`.

## 2272 8.3 Motion

2273 This section provides semantic information for the `Motion` entity.

2274 Note: See Figure 37 for XML schema.

2275 Note: See *Example 5* for the XML representation of the same example.

### 2276 8.3.1 Motion

2277 movement of the component relative to a coordinate system.

2278 `Motion` specifies the kinematic chain of the `component` entities.

2279 At most only one of `Origin` or `Transformation` **MUST** be defined for a `Motion`.

#### 2280 8.3.1.1 Value Properties of Motion

2281 *Table 38* lists the Value Properties of `Motion`.

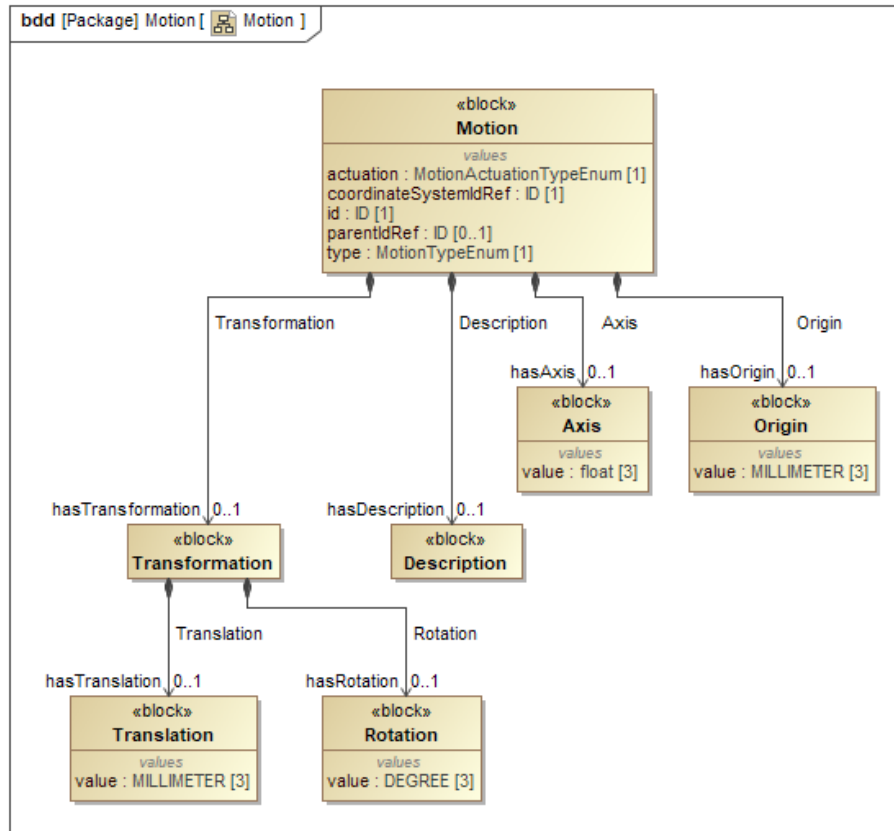


Figure 10: Motion

Value Property name	Value Property type	Multiplicity
actuation	MotionActuationTypeEnum	1
coordinateSystemIdRef	ID	1
id	ID	1
parentIdRef	ID	0..1
type	MotionTypeEnum	1

Table 38: Value Properties of Motion

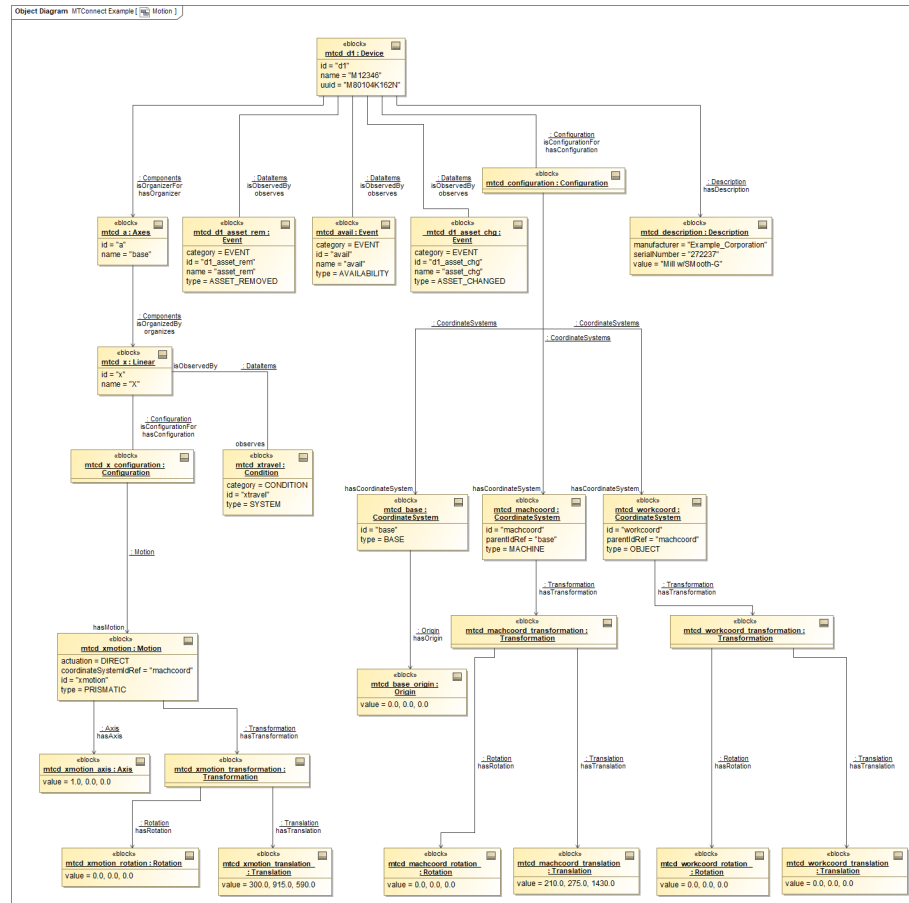


Figure 11: Motion Example

## 2282 Descriptions for Value Properties of Motion:

- 2283 • `actuation`
- 2284 describes if this component is actuated directly or indirectly as a result of other
- 2285 motion.

### 2286 `MotionActuationTypeEnum` Enumeration:

- 2287 – `DIRECT`
- 2288 movement is initiated by the component.
- 2289 – `NONE`
- 2290 no actuation of this axis.

2291 Note: Actuation of `NONE` can be either a derived `REVOLUTE` or

2292 `PRISMATIC` motion or static `FIXED` relationship.

- 2293 – `VIRTUAL`
- 2294 motion is computed and is used for expressing an imaginary movement.

- 2295 • `coordinateSystemIdRef`
- 2296 coordinate system within which the kinematic motion occurs.

- 2297 • `id`
- 2298 unique identifier for this element.

- 2299 • `parentIdRef`
- 2300 pointer to the `id` attribute of the parent `Motion`.

2301 The kinematic chain connects all components using the parent relations. All motion

2302 is connected to the motion of the parent. The first node in the chain will not have a

2303 parent.

- 2304 • `type`
- 2305 type of motion.

### 2306 `MotionTypeEnum` Enumeration:

- 2307 – `CONTINUOUS`
- 2308 revolves around an axis with a continuous range of motion.
- 2309 – `FIXED`
- 2310 axis does not move.
- 2311 – `PRISMATIC`
- 2312 sliding linear motion along an axis with a fixed range of motion.
- 2313 – `REVOLUTE`
- 2314 rotates around an axis with a fixed range of motion.

### 2315 8.3.1.2 Part Properties of Motion

2316 *Table 39* lists the Part Properties of Motion.

Part Property name	Multiplicity
Axis	0..1
Origin	0..1
Transformation	0..1
Description	0..1

**Table 39:** Part Properties of Motion

2317 Descriptions for Part Properties of Motion:

- 2318 • Axis
- 2319 axis along or around which the Component moves relative to a coordinate system.
- 2320 See *Section 4.3.4 - Axis*.
- 2321 • Origin
- 2322 coordinates of the origin position of a coordinate system.
- 2323 See *Section 8.2.2 - Origin*.
- 2324 • Transformation
- 2325 process of transforming to the origin position of the coordinate system from a parent
- 2326 coordinate system using Translation and Rotation.
- 2327 See *Section 8.2.3 - Transformation*.
- 2328 • Description
- 2329 descriptive content.
- 2330 See *Section 4.1.2 - Description*.

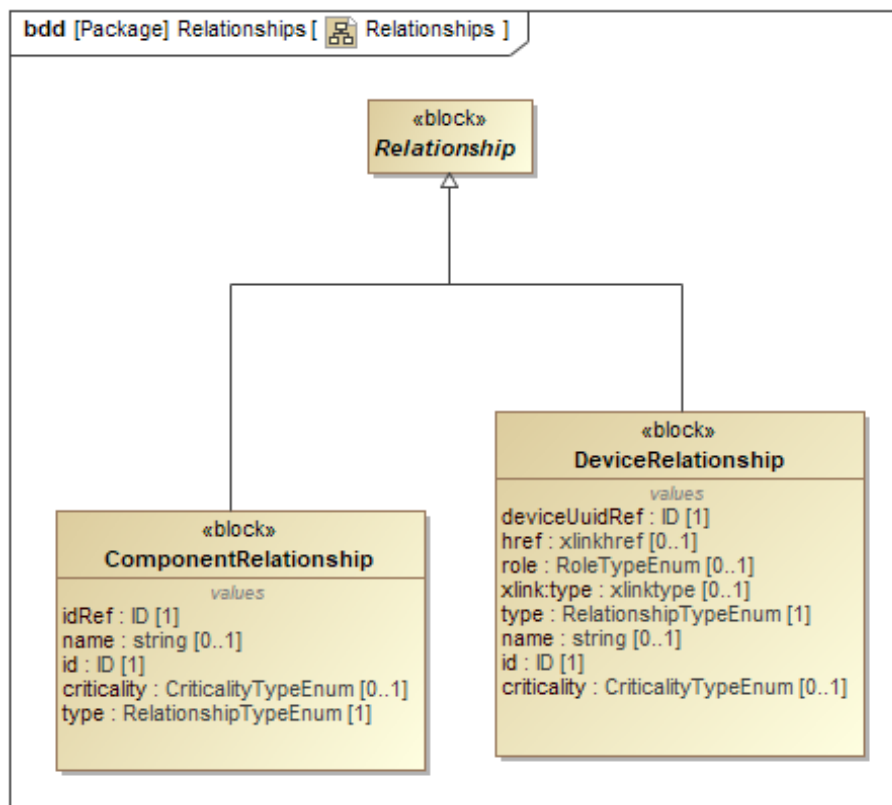
### 2331 8.3.2 Axis

2332 axis along or around which the Component moves relative to a coordinate system.

2333 The value of Axis **MUST** be a list of float of size 3.

## 2334 8.4 Relationships

2335 This section provides semantic information for the Relationship entity.



**Figure 12:** Relationships

2336 Note: See Figure 38 for XML schema.

2337 Note: See *Example 6* for the XML representation of the same example.

### 2338 8.4.1 Relationship

2339 association between two pieces of equipment that function independently but together  
 2340 perform a manufacturing operation.

2341 Relationship is an abstract entity and hence will be realized by specific Relation-  
 2342 ship types in an MTConnectDevices entity. See *Section 8.4.2 - ComponentRelation-*  
 2343 *ship*, *Section 8.4.3 - DeviceRelationship* and *Section 6.4 - Relationship Types for DataItem*.

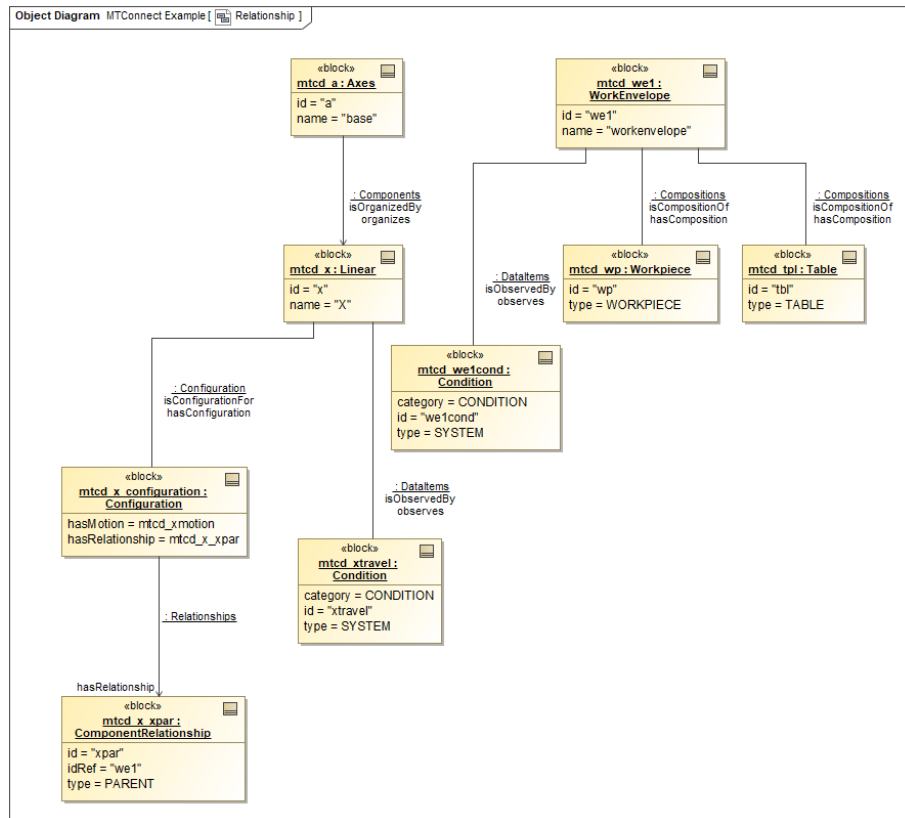


Figure 13: Relationship Example

## 2344 8.4.2 ComponentRelationship

2345 Relationship that describes the association between two components within a piece of  
 2346 equipment that function independently but together perform a capability or service within  
 2347 a piece of equipment.

### 2348 8.4.2.1 Value Properties of ComponentRelationship

2349 Table 40 lists the Value Properties of ComponentRelationship.

Value Property name	Value Property type	Multiplicity
idRef	ID	1
name	string	0..1
id	ID	1
criticality	CriticalityTypeEnum	0..1
type	RelationshipTypeEnum	1

**Table 40:** Value Properties of ComponentRelationship

2350 Descriptions for Value Properties of ComponentRelationship:

- 2351     • idRef
- 2352         reference to the associated Component element.
- 2353     • name
- 2354         name associated with this Relationship.
- 2355     • id
- 2356         unique identifier for this Relationship.
- 2357     • criticality
- 2358         defines whether the services or functions provided by the associated piece of equip-
- 2359         ment is required for the operation of this piece of equipment.
- 2360         CriticalityTypeEnum Enumeration:
- 2361             – CRITICAL
- 2362                 services or functions provided by the associated element is required for the
- 2363                 operation of this element.
- 2364             – NONCRITICAL
- 2365                 services or functions provided by the associated element is not required for the
- 2366                 operation of this element.
- 2367     • type
- 2368         defines the authority that this piece of equipment has relative to the associated piece
- 2369         of equipment.
- 2370         RelationshipTypeEnum Enumeration:
- 2371             – CHILD
- 2372                 functions as a child in the relationship with the associated element.



- 2373       – PARENT  
 2374       functions as a parent in the relationship with the associated element.
- 2375       – PEER  
 2376       functions as a peer which provides equal functionality and capabilities in the  
 2377       relationship with the associated element.

### 2378 8.4.3 DeviceRelationship

2379 Relationship that describes the association between two pieces of equipment that  
 2380 function independently but together perform a manufacturing operation.

#### 2381 8.4.3.1 Value Properties of DeviceRelationship

2382 *Table 41* lists the Value Properties of DeviceRelationship.

Value Property name	Value Property type	Multiplicity
deviceUuidRef	ID	1
href	xlink:href	0..1
role	RoleTypeEnum	0..1
xlink:type	xlink:type	0..1
type	RelationshipTypeEnum	1
name	string	0..1
id	ID	1
criticality	CriticalityTypeEnum	0..1

**Table 41:** Value Properties of DeviceRelationship

2383 Descriptions for Value Properties of DeviceRelationship:

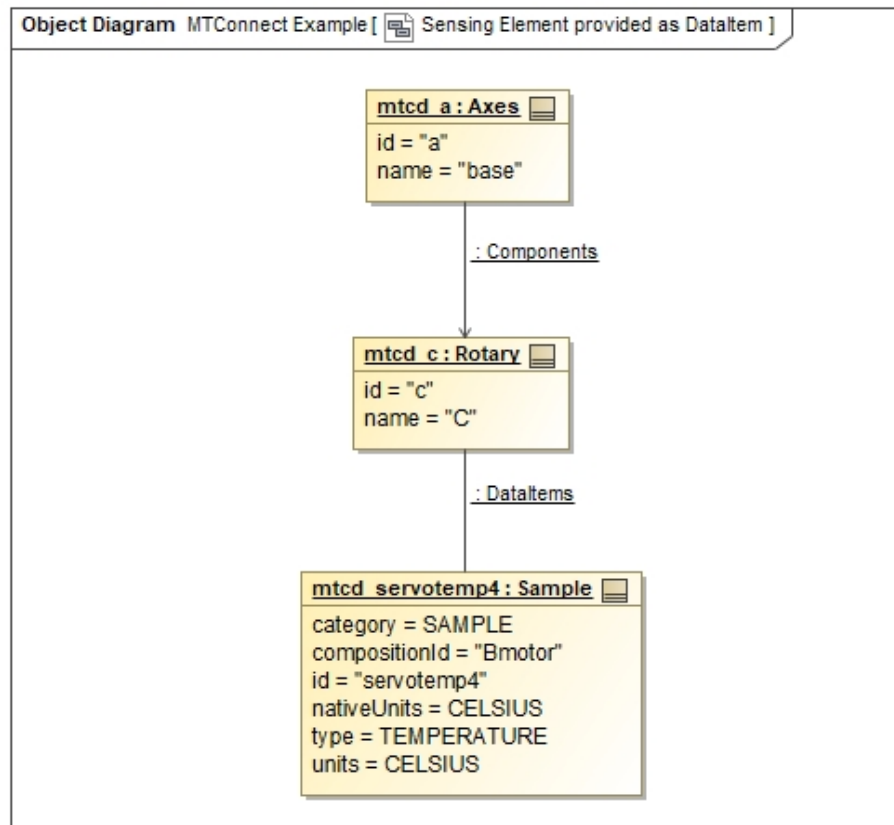
- 2384       • deviceUuidRef  
 2385       reference to the `uuid` attribute of the `Device` element of the associated piece of  
 2386       equipment.
- 2387       • href  
 2388       URI identifying the *agent* that is publishing information for the associated piece of  
 2389       equipment.

- 2390 • `role`
- 2391 defines the services or capabilities that the referenced piece of equipment provides
- 2392 relative to this piece of equipment.
- 2393 `RoleTypeEnum` Enumeration:
- 2394     – `AUXILIARY`
- 2395         associated element performs the functions as an `Auxiliary` for this element.
- 2396     – `SYSTEM`
- 2397         associated element performs the functions of a `System` for this element.
- 2398 • `xlink:type`
- 2399 `xlink:type` **MUST** have a fixed value of `locator` as defined in W3C XLink
- 2400 1.1 *Ref <https://www.w3.org/TR/xlink11/>.*
- 2401 • `type`
- 2402 defines the authority that this piece of equipment has relative to the associated piece
- 2403 of equipment.
- 2404 The value of `type` **MUST** be one of the `RelationshipTypeEnum` enumera-
- 2405 tion.
- 2406 • `name`
- 2407 `name` associated with this `Relationship`.
- 2408 • `id`
- 2409 unique identifier for this `Relationship`.
- 2410 • `criticality`
- 2411 defines whether the services or functions provided by the associated piece of equip-
- 2412 ment is required for the operation of this piece of equipment.
- 2413 The value of `criticality` **MUST** be one of the `CriticalityTypeEnum`
- 2414 enumeration.

## 2415 8.5 Sensor

- 2416 This section provides semantic information for the `SensorConfiguration` entity.
- 2417 `sensor` is a unique type of a piece of equipment. A sensor is typically comprised of two
- 2418 major components: a sensor unit that provides signal processing, conversion, and commu-
- 2419 nications and the *sensing elements* that provides a signal or measured value.

2420 The sensor unit is modeled as a *lower level* Component called Sensor. The *sensing*  
 2421 *element* may be modeled as a Composition element of a Sensor element and the  
 2422 measured value would be modeled as a DataItem (See *Section 6.1 - DataItems* for more  
 2423 information on DataItem elements). Each sensor unit may have multiple *sensing ele-*  
 2424 *ments*; each representing the data for a variety of measured values.



**Figure 14:** Sensing Element provided as a DataItem Example

2425 Note: See *Example 9* for an XML example.

2426 When a sensor unit is modeled as a Component or as a separate piece of equipment, it  
 2427 may provide additional configuration information for the sensor elements and the sensor  
 2428 unit itself.

2429 Note: If a Sensor provides vibration measurement data for the spindle on a  
 2430 piece of equipment, it could be modeled as a Sensor for rotary axis named  
 2431 C. See *Example 10* for an XML example.

2432 Note: If a `Sensor` provides measurement data for multiple `Component` el-  
 2433 ements within a piece of equipment and is not associated with any particular  
 2434 `Component`, it **MAY** be modeled as an independent `Component` and the  
 2435 data associated with measurements are associated with their associated `Com-`  
 2436 ponent entities. See *Example 11* for an XML example.

2437 Configuration data provides information required for maintenance and support of the  
 2438 sensor.

2439 When `Sensor` represents the sensor unit for multiple *sensing element(s)*, each sensing  
 2440 element is represented by a `Channel`. The sensor unit itself and each `Channel` repre-  
 2441 senting one *sensing element* **MAY** have its own configuration data.

2442 Note: See *Example 12* for an XML example.

## 2443 8.5.1 SensorConfiguration

2444 configuration for a `Sensor`.

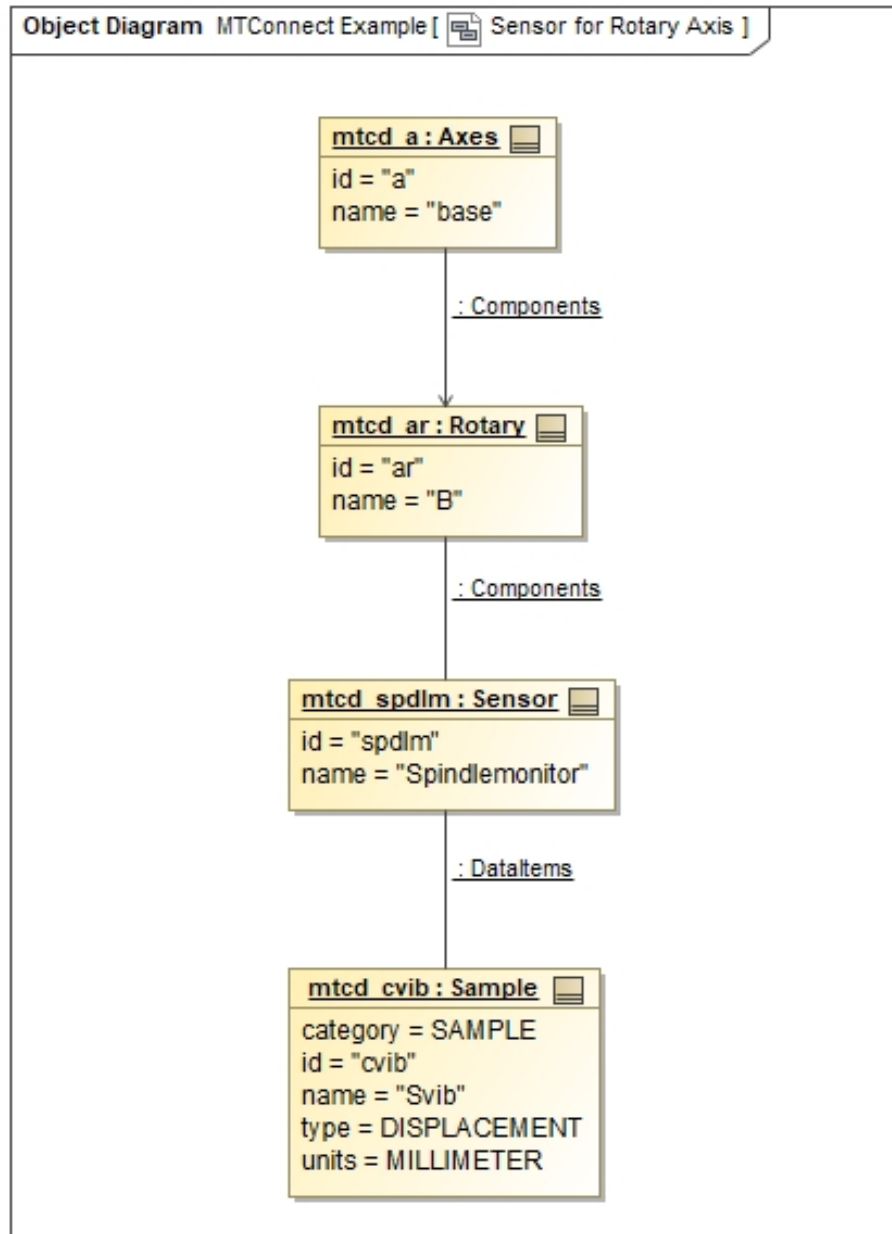
2445 Note: See Figure 41 for XML schema.

### 2446 8.5.1.1 Value Properties of SensorConfiguration

2447 *Table 42* lists the Value Properties of `SensorConfiguration`.

Value Property name	Value Property type	Multiplicity
CalibrationDate	dateTime	0..1
CalibrationInitials	string	0..1
FirmwareVersion	string	1
NextCalibrationDate	dateTime	0..1

**Table 42:** Value Properties of `SensorConfiguration`

**Figure 15:** Sensor for Rotary Axis Example

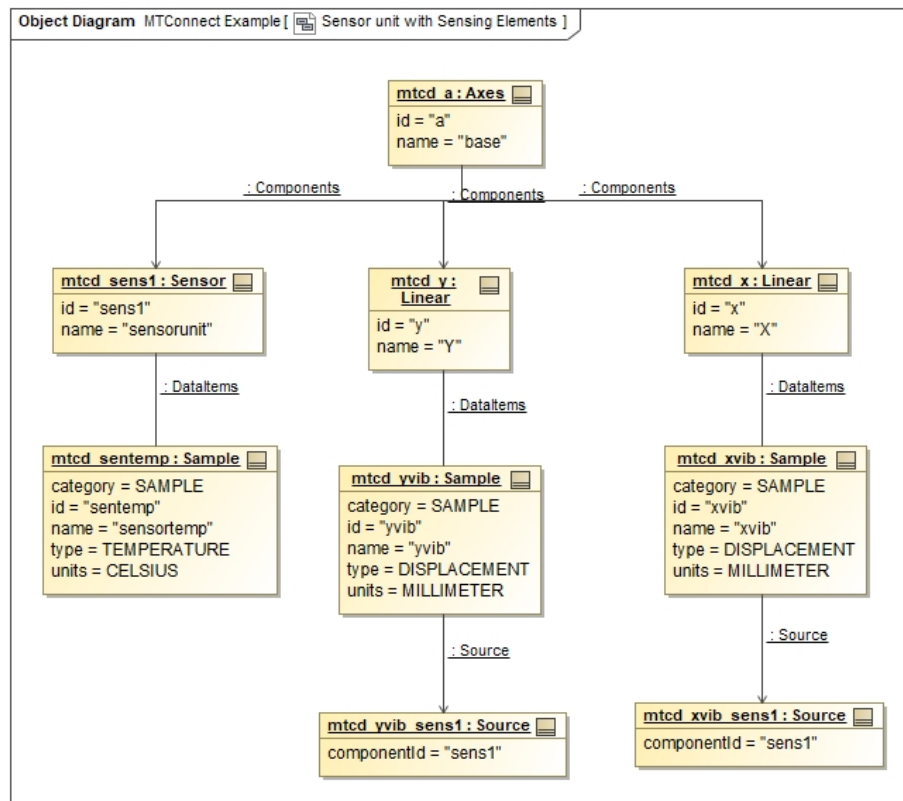


Figure 16: Sensor unit with Sensing Elements Example

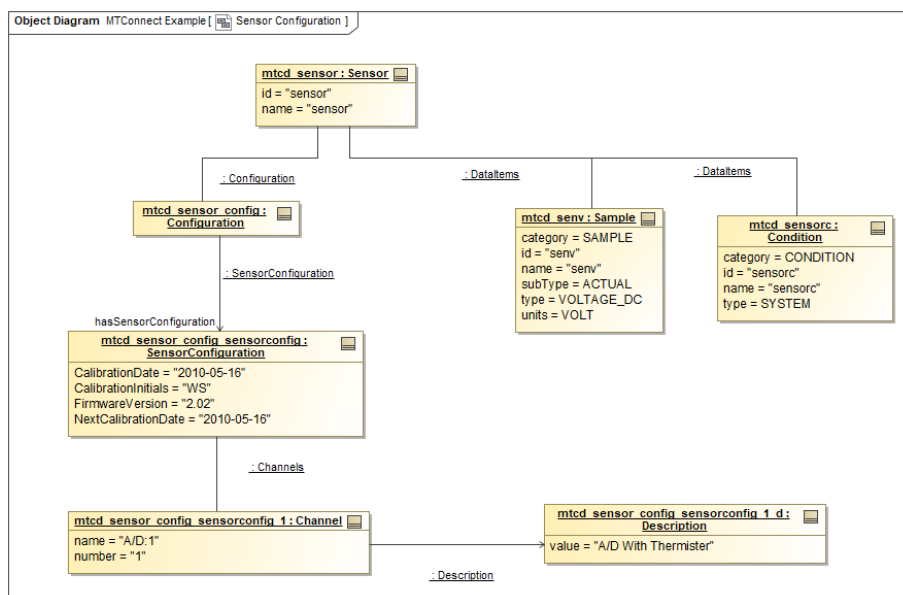
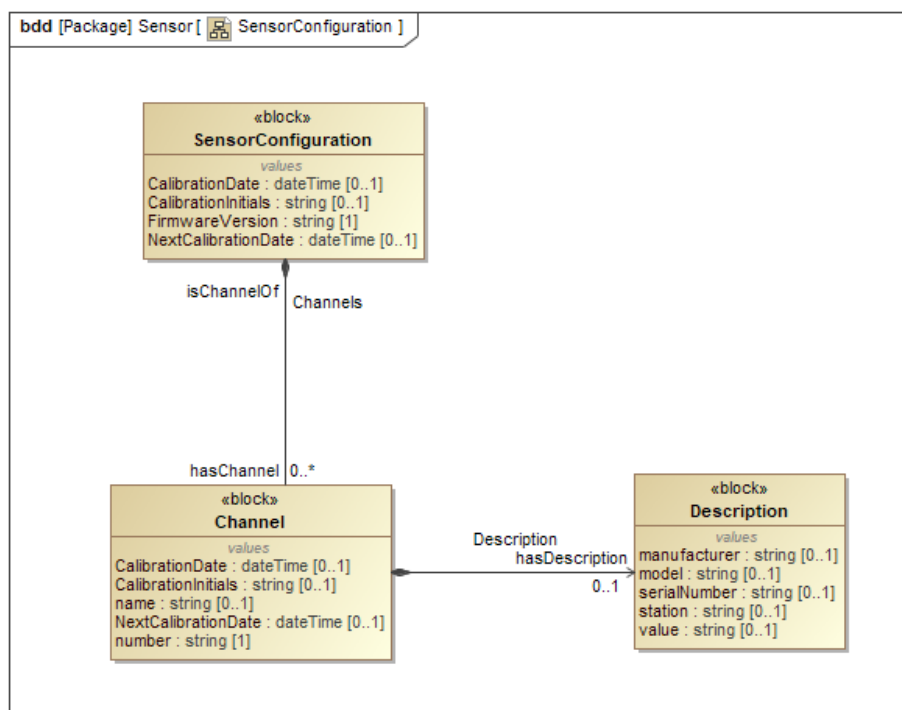


Figure 17: Sensor Configuration Example

**Figure 18:** SensorConfiguration

2448 Descriptions for Value Properties of SensorConfiguration:

- 2449 • CalibrationDate
- 2450     Date upon which the sensor unit was last calibrated.
- 2451 • CalibrationInitials
- 2452     The initials of the person verifying the validity of the calibration data.
- 2453 • FirmwareVersion
- 2454     Version number for the sensor unit as specified by the manufacturer.
- 2455 • NextCalibrationDate
- 2456     Date upon which the sensor unit is next scheduled to be calibrated.

### 2457 8.5.1.2 Part Properties of SensorConfiguration

2458 Table 43 lists the Part Properties of SensorConfiguration.

Part Property name	Multiplicity
Channel (organized by Channels)	0..*

**Table 43:** Part Properties of SensorConfiguration

2459 Descriptions for Part Properties of SensorConfiguration:

- 2460 • Channel
- 2461     *sensing element* of a Sensor.
- 2462     Channels groups one or more Channel entities. See Channel.

## 2463 8.5.2 Channel

2464 *sensing element* of a Sensor.

2465 When Sensor has multiple *sensing elements*, each *sensing element* is modeled as a  
 2466 Channel for the Sensor.

### 2467 8.5.2.1 Value Properties of Channel

2468 Table 44 lists the Value Properties of Channel.



Value Property name	Value Property type	Multiplicity
CalibrationDate	dateTime	0..1
CalibrationInitials	string	0..1
name	string	0..1
NextCalibrationDate	dateTime	0..1
number	string	1

**Table 44:** Value Properties of Channel

2469 Descriptions for Value Properties of Channel:

- 2470 • CalibrationDate
- 2471     Date upon which the sensor unit was last calibrated to the sensor element.
- 2472 • CalibrationInitials
- 2473     The initials of the person verifying the validity of the calibration data.
- 2474 • name
- 2475     name of the specific *sensing element*.
- 2476 • NextCalibrationDate
- 2477     Date upon which the sensor element is next scheduled to be calibrated with the
- 2478     sensor unit.
- 2479 • number
- 2480     unique identifier that will only refer to a specific *sensing element*.

### 2481 8.5.2.2 Part Properties of Channel

2482 *Table 45* lists the Part Properties of Channel.

Part Property name	Multiplicity
Description	0..1

**Table 45:** Part Properties of Channel

2483 Descriptions for Part Properties of Channel:

- 2484 • Description
- 2485     descriptive content.
- 2486     See *Section 4.1.2 - Description*.

## 2487 8.6 SolidModel

2488 This section provides semantic information for the SolidModel entity.

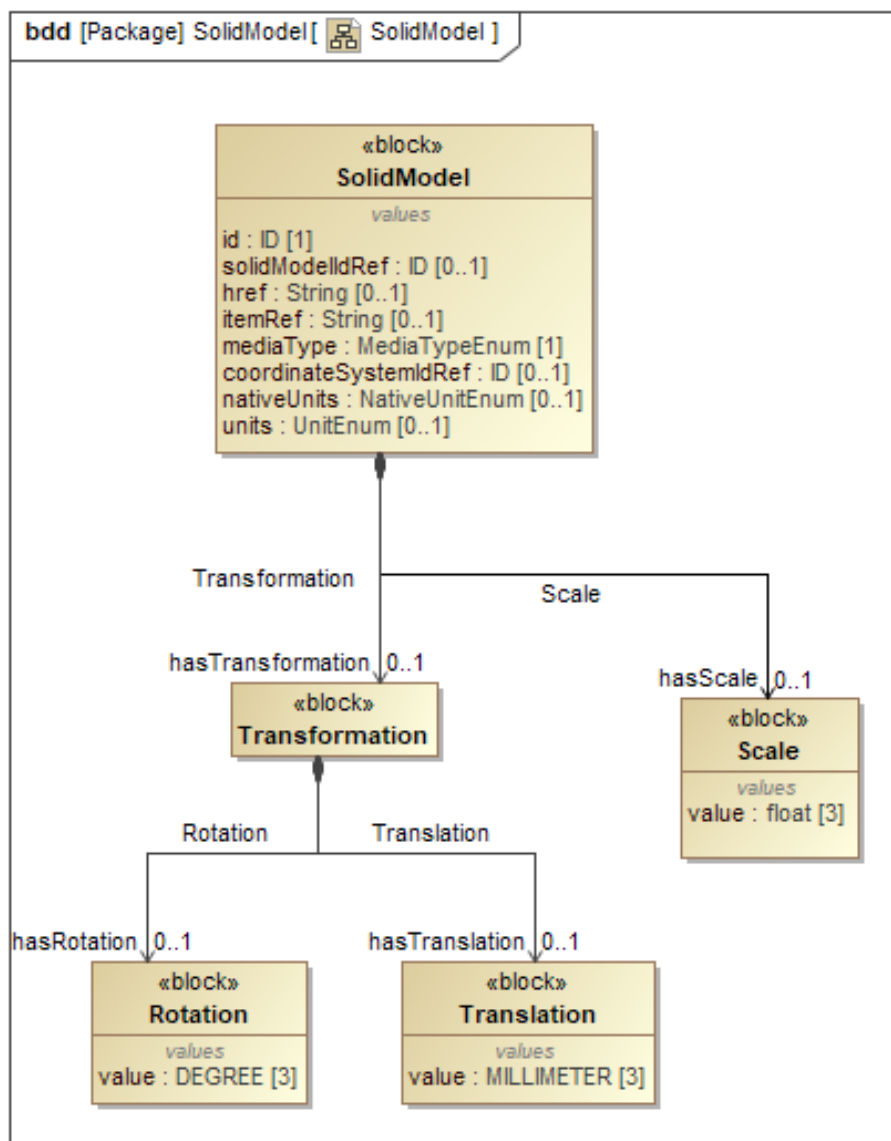
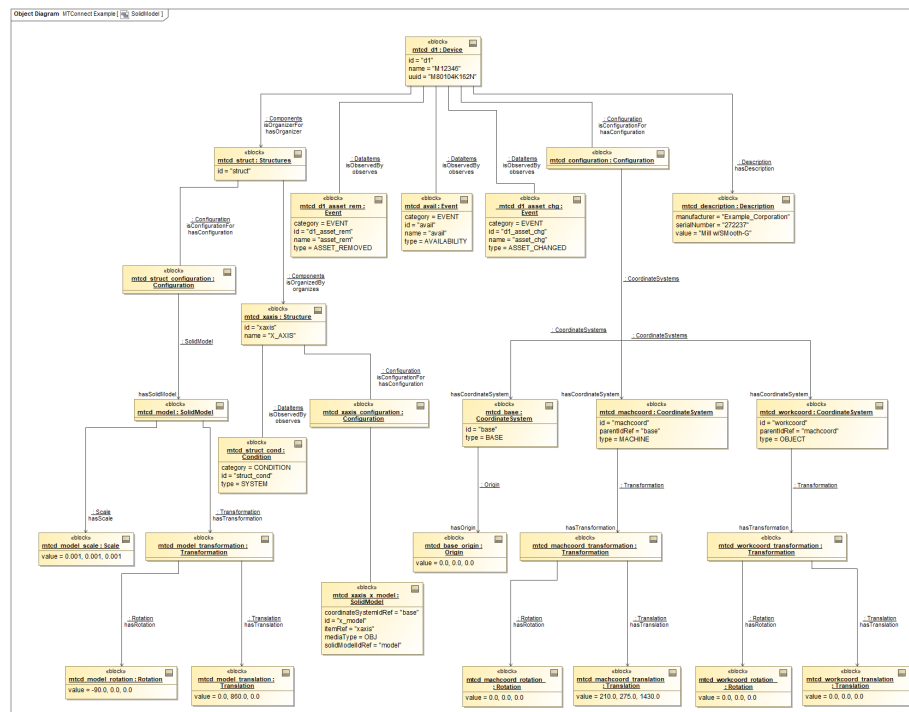


Figure 19: SolidModel

2489 Note: See Figure 42 for XML schema.

2490 Note: See *Example 7* for the XML representation of the same example.



### Figure 20: SolidModel Example

## 2491 8.6.1 SolidModel

2492 references to a file with the three-dimensional geometry of the Component or Compo-  
2493 sition.

2494 The geometry **MAY** have a transformation and a scale to position the Component with  
2495 respect to the other Components. A geometry file can contain a set of assembled items,  
2496 in this case, the `SolidModel` reference the `id, SolidModel` of the assembly model  
2497 file and the specific item within that file.

2498 The SolidModel **MAY** provide a translation, rotation, and scale to correctly place it  
2499 relative to the other geometries in the machine. If the Component can move and has  
2500 a Motion Configuration, the SolidModel will move when the Component or  
2501 Composition moves.

2502 Either an href, SolidModel or a modelIdRef, SolidModel and an itemRef, SolidModel  
2503 **MUST** be specified.

### 2504 8.6.1.1 Value Properties of SolidModel

2505 *Table 46* lists the Value Properties of `SolidModel`.

Value Property name	Value Property type	Multiplicity
<code>id</code>	ID	1
<code>solidModelIdRef</code>	ID	0..1
<code>mediaType</code>	MediaTypeEnum	1
<code>coordinateSystemIdRef</code>	ID	0..1
<code>nativeUnits</code>	NativeUnitEnum	0..1
<code>units</code>	UnitEnum	0..1

**Table 46:** Value Properties of `SolidModel`

2506 Descriptions for Value Properties of `SolidModel`:

- 2507     • `id`
- 2508         unique identifier for this element.
- 2509     • `solidModelIdRef`
- 2510         associated model file if an item reference is used.
- 2511     • `mediaType`
- 2512         format of the referenced document.
- 2513         MediaTypeEnum Enumeration:
- 2514             – 3DS
- 2515                 Autodesk file format.
- 2516             – ACIS
- 2517                 Dassault file format.
- 2518             – COLLADA
- 2519                 ISO 17506.
- 2520             – GDML
- 2521                 Geometry Description Markup Language.
- 2522             – IGES
- 2523                 Initial Graphics Exchange Specification.
- 2524             – OBJ
- 2525                 Wavefront OBJ file format.
- 2526             – STEP
- 2527                 ISO 10303 STEP AP203 or AP242 format.

- 2528       – STL
- 2529       STereoLithography file format.
- 2530       – X\_T
- 2531       Parasolid XT Siemens data interchange format.
- 2532       • coordinateSystemIdRef
- 2533       reference to the coordinate system for this SolidModel.
- 2534       • nativeUnits
- 2535       same as DataItem nativeUnits. See *Section 6.1.1 - DataItem*.
- 2536       The value of nativeUnits **MUST** be one of the NativeUnitEnum enumera-
- 2537       tion.
- 2538       • units
- 2539       same as DataItem units. See *Section 6.1.1 - DataItem*.
- 2540       The value of units **MUST** be one of the UnitEnum enumeration.

#### 2541 **8.6.1.2 Part Properties of SolidModel**

2542 *Table 47* lists the Part Properties of SolidModel.

Part Property name	Multiplicity
Transformation	0..1
Scale	0..1

**Table 47:** Part Properties of SolidModel

2543 Descriptions for Part Properties of SolidModel:

- 2544       • Transformation
- 2545       process of transforming to the origin position of the coordinate system from a parent
- 2546       coordinate system using Translation and Rotation.
- 2547       See *Section 8.2.3 - Transformation*.
- 2548       • Scale
- 2549       either a single multiplier applied to all three dimensions or a three space multiplier
- 2550       given in the X, Y, and Z dimensions in the coordinate system used for the Solid-
- 2551       Model.
- 2552       See *Section 8.6.2 - Scale*.

## 2553 8.6.2 Scale

2554 either a single multiplier applied to all three dimensions or a three space multiplier given  
2555 in the X, Y, and Z dimensions in the coordinate system used for the `SolidModel`.

2556 The value of `Scale` **MUST** be a list of `float` of size 3.

## 2557 8.7 Specifications

2558 This section provides semantic information for the `Specification` entity.

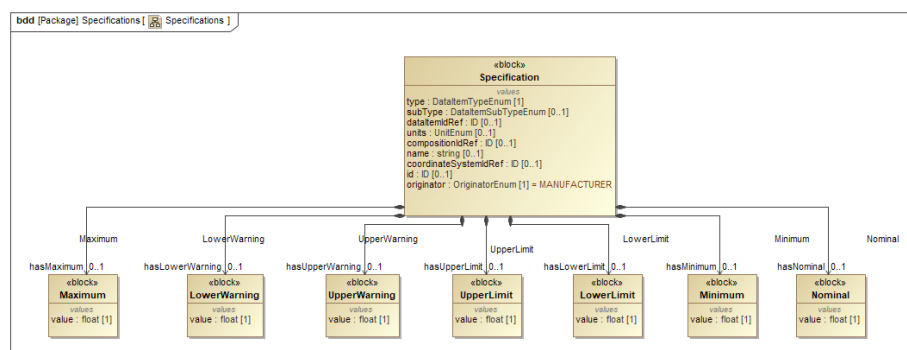


Figure 21: Specifications

2559 Note: See Figure 43 for XML schema.

2560 Note: See *Example 8* for the XML representation of the same example.

### 2561 8.7.1 Specification

2562 design characteristics for a piece of equipment.

#### 2563 8.7.1.1 Value Properties of Specification

2564 *Table 48* lists the Value Properties of `Specification`.

Value Property name	Value Property type	Multiplicity
type	DataTypeEnum	1
subType	DataTypeSubTypeEnum	0..1
dataItemIdRef	ID	0..1
units	UnitEnum	0..1
compositionIdRef	ID	0..1
name	string	0..1
coordinateSystemIdRef	ID	0..1
id	ID	0..1
originator	OriginatorEnum	1

**Table 48:** Value Properties of Specification

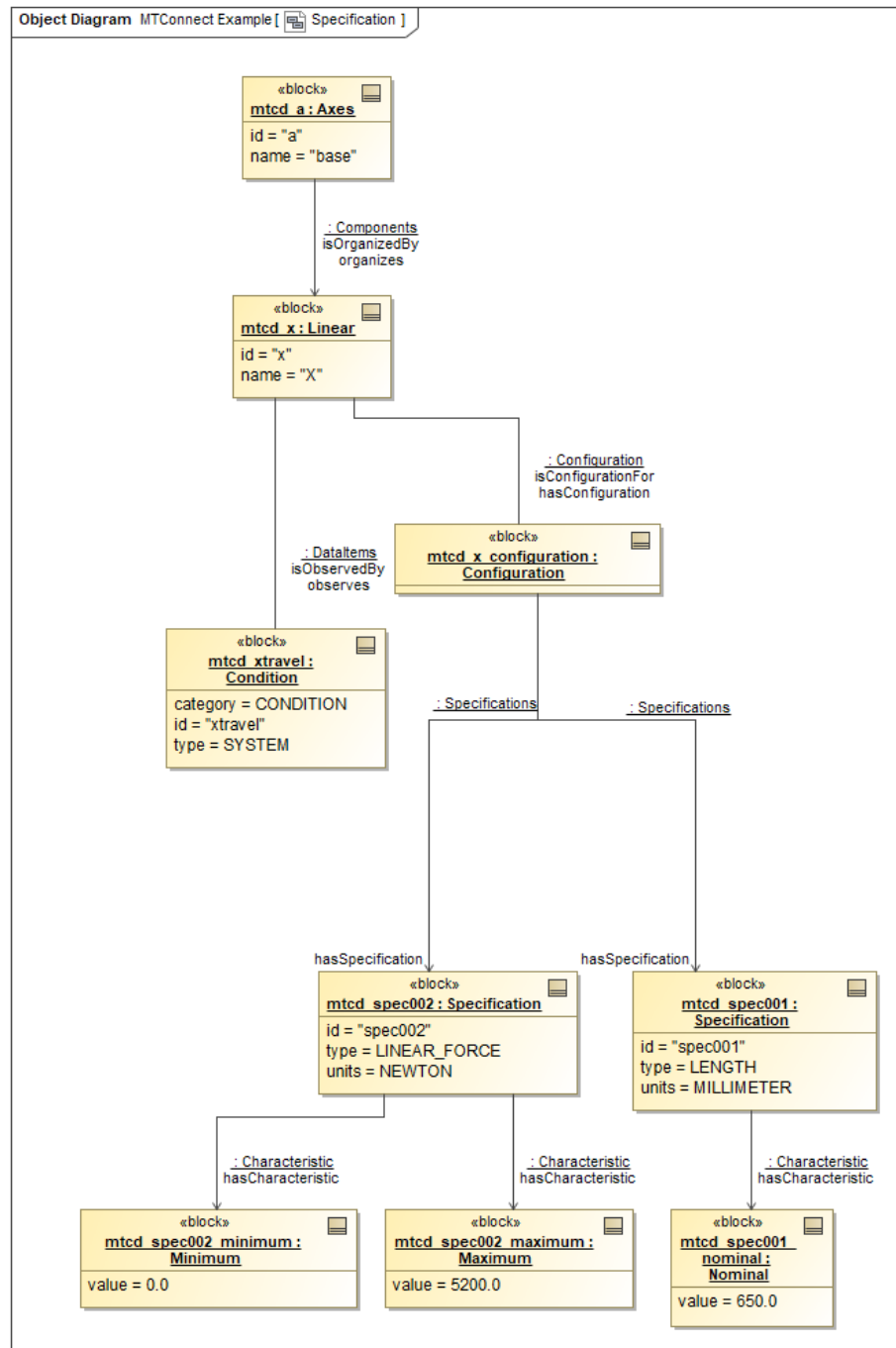


Figure 22: Specification Example



## 2565 Descriptions for Value Properties of Specification:

- 2566 • type
  - 2567 same as DataItem type. See *Section 6.5 - DataItem Types*.
  - 2568 The value of type **MUST** be one of the DataItemTypeEnum enumeration.
- 2569 • subType
  - 2570 same as DataItem subType. See *Section 6.1.1 - DataItem*.
  - 2571 The value of subType **MUST** be one of the DataItemSubTypeEnum enumer-
  - 2572 ation.
- 2573 • dataItemIdRef
  - 2574 reference to the id attribute of the DataItem associated with this element.
- 2575 • units
  - 2576 same as DataItem units. See *Section 6.1.1 - DataItem*.
  - 2577 The value of units **MUST** be one of the UnitEnum enumeration.
- 2578 • compositionIdRef
  - 2579 reference to the id attribute of the Composition associated with this element.
- 2580 • name
  - 2581 name provides additional meaning and differentiates between Specification
  - 2582 elements.
- 2583 • coordinateSystemIdRef
  - 2584 references the CoordinateSystem for geometric Specification elements.
- 2585 • id
  - 2586 unique identifier for this Specification.
- 2587 • originator
  - 2588 reference to the creator of the Specification.
  - 2589 OriginatorEnum Enumeration:
    - 2590 – MANUFACTURER
      - 2591 manufacturer of a piece of equipment or Component.
    - 2592 – USER
      - 2593 owner or implementer of a piece of equipment or Component.

### 2594 8.7.1.2 Part Properties of Specification

2595 *Table 49* lists the Part Properties of Specification.

Part Property name	Multiplicity
Maximum	0..1
UpperLimit	0..1
LowerWarning	0..1
LowerLimit	0..1
UpperWarning	0..1
Nominal	0..1
Minimum	0..1

**Table 49:** Part Properties of Specification

2596 Descriptions for Part Properties of Specification:

- 2597     • Maximum
- 2598         numeric upper constraint.
- 2599     • UpperLimit
- 2600         upper conformance boundary for a variable.
- 2601         Note: immediate concern or action may be required.
- 2602     • LowerWarning
- 2603         lower boundary indicating increased concern and supervision may be required.
- 2604     • LowerLimit
- 2605         lower conformance boundary for a variable.
- 2606         Note: immediate concern or action may be required.
- 2607     • UpperWarning
- 2608         upper boundary indicating increased concern and supervision may be required.
- 2609     • Nominal
- 2610         numeric target or expected value.
- 2611     • Minimum
- 2612         numeric lower constraint.

## 2613 8.7.2 ProcessSpecification

2614 Specification that provides information used to assess the conformance of a variable  
2615 to process requirements.

### 2616 8.7.2.1 Part Properties of ProcessSpecification

2617 Table 50 lists the Part Properties of ProcessSpecification.

Part Property name	Multiplicity
SpecificationLimits	0..1
ControlLimits	0..1
AlarmLimits	0..1

**Table 50:** Part Properties of ProcessSpecification

2618 Descriptions for Part Properties of ProcessSpecification:

- 2619 • SpecificationLimits  
2620 set of limits that define a range of values designating acceptable performance for a  
2621 variable.  
2622 See Section 8.7.5 - SpecificationLimits.
- 2623 • ControlLimits  
2624 set of limits that is used to indicate whether a process variable is stable and in control.  
2625 See Section 8.7.3 - ControlLimits.
- 2626 • AlarmLimits  
2627 set of limits that is used to trigger warning or alarm indicators.  
2628 See Section 8.7.4 - AlarmLimits.

## 2629 8.7.3 ControlLimits

2630 set of limits that is used to indicate whether a process variable is stable and in control.

### 2631 8.7.3.1 Part Properties of ControlLimits

2632 Table 51 lists the Part Properties of ControlLimits.

Part Property name	Multiplicity
UpperLimit	0..1
UpperWarning	0..1
LowerWarning	0..1
Nominal	0..1
LowerLimit	0..1

**Table 51:** Part Properties of ControlLimits

2633 Descriptions for Part Properties of ControlLimits:

2634 • UpperLimit

2635 upper conformance boundary for a variable.

2636 Note: immediate concern or action may be required.

2637 • UpperWarning

2638 upper boundary indicating increased concern and supervision may be required.

2639 • LowerWarning

2640 lower boundary indicating increased concern and supervision may be required.

2641 • Nominal

2642 numeric target or expected value.

2643 • LowerLimit

2644 lower conformance boundary for a variable.

2645 Note: immediate concern or action may be required.

## 2646 8.7.4 AlarmLimits

2647 set of limits that is used to trigger warning or alarm indicators.

### 2648 8.7.4.1 Part Properties of AlarmLimits

2649 Table 52 lists the Part Properties of AlarmLimits.

Part Property name	Multiplicity
UpperLimit	0..1
UpperWarning	0..1
LowerLimit	0..1
LowerWarning	0..1

**Table 52: Part Properties of AlarmLimits**

2650 Descriptions for Part Properties of AlarmLimits:

2651     • UpperLimit

2652         upper conformance boundary for a variable.

2653             Note: immediate concern or action may be required.

2654     • UpperWarning

2655         upper boundary indicating increased concern and supervision may be required.

2656     • LowerLimit

2657         lower conformance boundary for a variable.

2658             Note: immediate concern or action may be required.

2659     • LowerWarning

2660         lower boundary indicating increased concern and supervision may be required.

## 2661 8.7.5 SpecificationLimits

2662 set of limits that define a range of values designating acceptable performance for a variable.

### 2663 8.7.5.1 Part Properties of SpecificationLimits

2664 *Table 53* lists the Part Properties of SpecificationLimits.

Part Property name	Multiplicity
UpperLimit	0..1
Nominal	0..1
LowerLimit	0..1

**Table 53:** Part Properties of SpecificationLimits

2665 Descriptions for Part Properties of SpecificationLimits:

2666     • UpperLimit

2667         upper conformance boundary for a variable.

2668             Note: immediate concern or action may be required.

2669     • Nominal

2670         numeric target or expected value.

2671     • LowerLimit

2672         lower conformance boundary for a variable.

2673             Note: immediate concern or action may be required.

## 2674 8.7.6 UpperWarning

2675     upper boundary indicating increased concern and supervision may be required.

2676     The value of UpperWarning **MUST** be float.

## 2677 8.7.7 UpperLimit

2678     upper conformance boundary for a variable.

2679             Note: immediate concern or action may be required.

2680     The value of UpperLimit **MUST** be float.

**2681 8.7.8 Maximum**

2682 numeric upper constraint.

2683 The value of `Maximum` **MUST** be float.

**2684 8.7.9 LowerLimit**

2685 lower conformance boundary for a variable.

2686 Note: immediate concern or action may be required.

2687 The value of `LowerLimit` **MUST** be float.

**2688 8.7.10 LowerWarning**

2689 lower boundary indicating increased concern and supervision may be required.

2690 The value of `LowerWarning` **MUST** be float.

**2691 8.7.11 Minimum**

2692 numeric lower constraint.

2693 The value of `Minimum` **MUST** be float.

**2694 8.7.12 Nominal**

2695 numeric target or expected value.

2696 The value of `Nominal` **MUST** be float.

## 2697 9 Profile

2698 MTConnect Profile is a *profile* that extends the Systems Modeling Language (SysML)  
 2699 metamodel for the MTConnect domain using additional data types and *stereotypes*.

### 2700 9.1 DataTypes

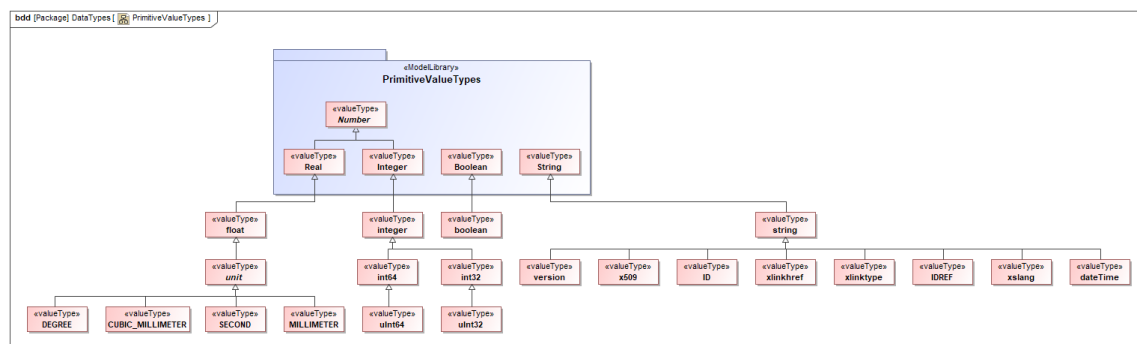


Figure 23: DataTypes

#### 2701 9.1.1 boolean

2702 primitive type.

#### 2703 9.1.2 ID

2704 string that represents an identifier (ID).

#### 2705 9.1.3 string

2706 primitive type.

#### 2707 9.1.4 float

2708 primitive type.



**2709 9.1.5 dateTime**

2710 string that represents timestamp in ISO 8601 format.

**2711 9.1.6 integer**

2712 primitive type.

**2713 9.1.7 xlinktype**

2714 string that represents the type of an XLink element. See [https://www.w3.org/TR/](https://www.w3.org/TR/xlink11/)  
2715 [xlink11/](https://www.w3.org/TR/xlink11/).

**2716 9.1.8 xslang**

2717 string that represents a language tag. See [http://www.ietf.org/rfc/rfc4646.](http://www.ietf.org/rfc/rfc4646.txt)  
2718 [txt](http://www.ietf.org/rfc/rfc4646.txt).

**2719 9.1.9 SECOND**

2720 float that represents time in seconds.

**2721 9.1.10 IDREF**

2722 string that represents a reference to an ID.

**2723 9.1.11 xlinkhref**

2724 string that represents the locator attribute of an XLink element. See [https://www.w3.](https://www.w3.org/TR/xlink11/)  
2725 [org/TR/xlink11/](https://www.w3.org/TR/xlink11/).

**2726 9.1.12 x509**

2727 string that represents an `x509` data block. *Ref ISO/IEC 9594-8:2020.*

**2728 9.1.13 int32**

2729 32-bit integer.

**2730 9.1.14 int64**

2731 64-bit integer.

**2732 9.1.15 version**

2733 series of four numeric values, separated by a decimal point, representing a *major*, *minor*,  
2734 and *revision* number of the MTConnect Standard and the revision number of a specific  
2735 *schema*.

**2736 9.1.16 uInt32**

2737 32-bit unsigned integer.

**2738 9.1.17 uInt64**

2739 64-bit unsigned integer.

**2740 9.2 Stereotypes**

**2741 9.2.1 organizer**

2742 element that *organizes* other elements of a type.

**2743 9.2.2 deprecated**

2744 element that has been deprecated.

**2745 9.2.3 extensible**

2746 enumeration that can be extended.

**2747 9.2.4 informative**

2748 element that is descriptive and non-normative.

**2749 9.2.5 valueType**

2750 extends SysML <<ValueType>> to include `Class` as a value type.

**2751 9.2.6 normative**

2752 element that has been added to the standard.

**2753 9.2.7 observes**

2754 association in which a *Component* makes *Observations* about an observable *DataItem*.

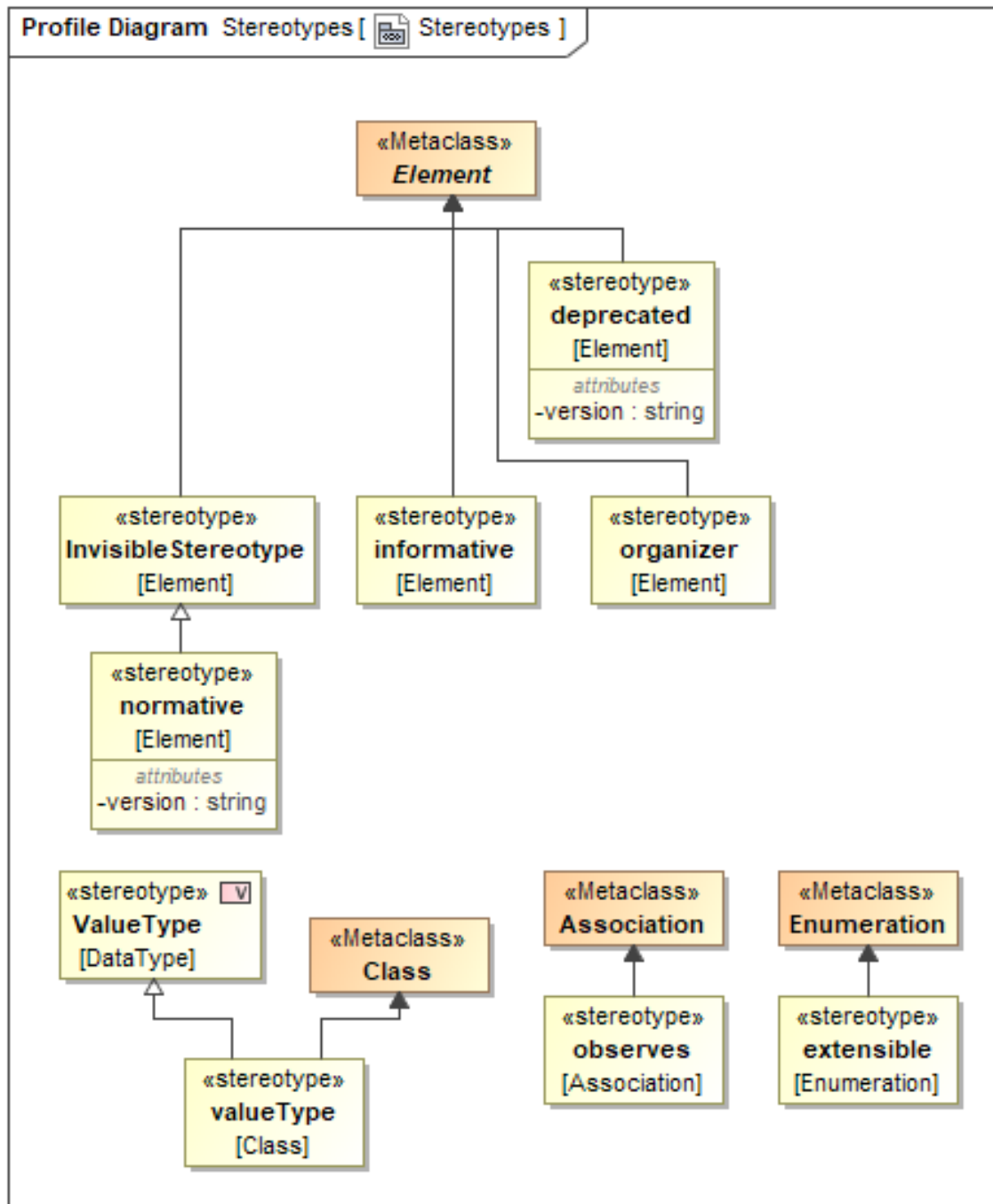


Figure 24: Stereotypes

## 2755 Appendices

### 2756 A Bibliography

2757 Engineering Industries Association. EIA Standard - EIA-274-D, Interchangeable Variable,  
2758 Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically  
2759 Controlled Machines. Washington, D.C. 1979.

2760 ISO TC 184/SC4/WG3 N1089. ISO/DIS 10303-238: Industrial automation systems and  
2761 integration Product data representation and exchange Part 238: Application Protocols: Ap-  
2762 plication interpreted model for computerized numerical controllers. Geneva, Switzerland,  
2763 2004.

2764 International Organization for Standardization. ISO 14649: Industrial automation sys-  
2765 tems and integration – Physical device control – Data model for computerized numerical  
2766 controllers – Part 10: General process data. Geneva, Switzerland, 2004.

2767 International Organization for Standardization. ISO 14649: Industrial automation sys-  
2768 tems and integration – Physical device control – Data model for computerized numerical  
2769 controllers – Part 11: Process data for milling. Geneva, Switzerland, 2000.

2770 International Organization for Standardization. ISO 6983/1 – Numerical Control of ma-  
2771 chines – Program format and definition of address words – Part 1: Data format for posi-  
2772 tioning, line and contouring control systems. Geneva, Switzerland, 1982.

2773 Electronic Industries Association. ANSI/EIA-494-B-1992, 32 Bit Binary CL (BCL) and  
2774 7 Bit ASCII CL (ACL) Exchange Input Format for Numerically Controlled Machines.  
2775 Washington, D.C. 1992.

2776 National Aerospace Standard. Uniform Cutting Tests - NAS Series: Metal Cutting Equip-  
2777 ment Specifications. Washington, D.C. 1969.

2778 International Organization for Standardization. ISO 10303-11: 1994, Industrial automa-  
2779 tion systems and integration Product data representation and exchange Part 11: Descrip-  
2780 tion methods: The EXPRESS language reference manual. Geneva, Switzerland, 1994.

2781 International Organization for Standardization. ISO 10303-21: 1996, Industrial automa-  
2782 tion systems and integration – Product data representation and exchange – Part 21: Imple-  
2783 mentation methods: Clear text encoding of the exchange structure. Geneva, Switzerland,  
2784 1996.

2785 H.L. Horton, F.D. Jones, and E. Oberg. Machinery's Handbook. Industrial Press, Inc.

- 2786 New York, 1984.
- 2787 International Organization for Standardization. ISO 841-2001: Industrial automation sys-  
2788 tems and integration - Numerical control of machines - Coordinate systems and motion  
2789 nomenclature. Geneva, Switzerland, 2001.
- 2790 ASME B5.57: Methods for Performance Evaluation of Computer Numerically Controlled  
2791 Lathes and Turning Centers, 1998.
- 2792 ASME/ANSI B5.54: Methods for Performance Evaluation of Computer Numerically Con-  
2793 trolled Machining Centers. 2005.
- 2794 OPC Foundation. OPC Unified Architecture Specification, Part 1: Concepts Version 1.00.  
2795 July 28, 2006.
- 2796 IEEE STD 1451.0-2007, Standard for a Smart Transducer Interface for Sensors and Ac-  
2797 tuators – Common Functions, Communication Protocols, and Transducer Electronic Data  
2798 Sheet (TEDS) Formats, IEEE Instrumentation and Measurement Society, TC-9, The In-  
2799 stitute of Electrical and Electronics Engineers, Inc., New York, N.Y. 10016, SH99684,  
2800 October 5, 2007.
- 2801 IEEE STD 1451.4-1994, Standard for a Smart Transducer Interface for Sensors and Ac-  
2802 tuators – Mixed-Mode Communication Protocols and Transducer Electronic Data Sheet  
2803 (TEDS) Formats, IEEE Instrumentation and Measurement Society, TC-9, The Institute of  
2804 Electrical and Electronics Engineers, Inc., New York, N.Y. 10016, SH95225, December  
2805 15, 2004.

## 2806 B XML Schema Diagrams

### 2807 B.1 Components Schema Diagrams

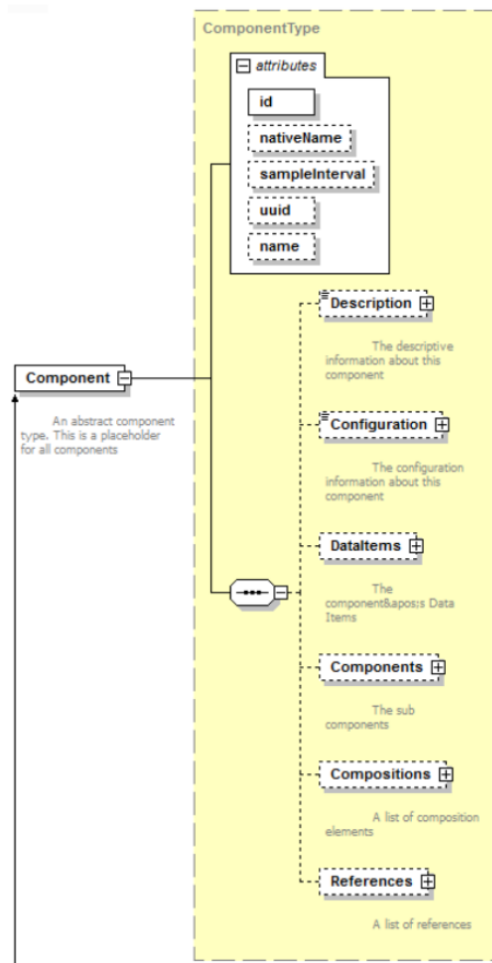


Figure 25: Components Schema

### 2808 B.2 DataItems Schema Diagrams

### 2809 B.3 References Schema Diagrams

### 2810 B.4 Configuration Schema Diagrams

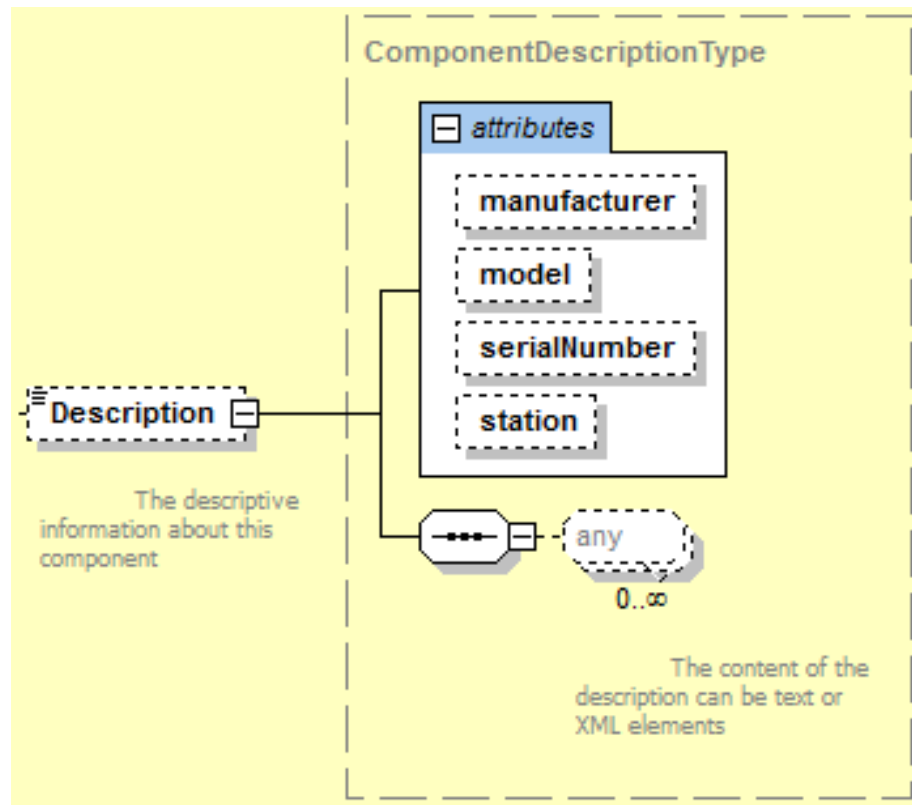


Figure 26: Component Description Schema

## 2811 C XML Examples

### 2812 C.1 Device Entity Hierarchy Example

#### Example 1: Device Entity Hierarchy Example

```

2813 1 <Devices>
2814 2   <Device id="d1" name="M12346" uuid="M80104K162N">
2815 3     <Components>
2816 4       <Axes id="a" name="base">
2817 5         <Components>
2818 6           <Linear id="x" name="X"/>
2819 7           <Linear id="y" name="Y"/>
2820 8           <Linear id="z" name="Z"/>
2821 9           <Rotary id="ar" name="B"/>
2822 10          <Linear id="w" name="Z3" nativeName="W"/>
2823 11          <Rotary id="c" name="C"/>
2824 12        </Components>
2825 13      </Axes>
2826 14      <Controller id="cont" name="controller">
2827 15        <Components>

```



```

2828 16      <Path id="path1" name="path"/>
2829 17    </Components>
2830 18  </Controller>
2831 19    <Door id="door1" name="door"/>
2832 20    <Systems id="systems" name="systems">
2833 21      <Components>
2834 22        <WorkEnvelope id="wel" name="workenv">
2835 23          <Compositions>
2836 24            <Composition type="WORKPIECE" id="wp"/>
2837 25            <Composition type="TABLE" id="tbl"/>
2838 26          </Compositions>
2839 27        </WorkEnvelope>
2840 28        <Electric id="elec" name="electric"/>
2841 29        <Hydraulic id="hydraulic" name="hydraulic"/>
2842 30        <Coolant id="coolant" name="coolant"/>
2843 31        <Pneumatic id="pneumatic" name="pneumatic"/>
2844 32        <Lubrication id="lubrication" name="lubrication"/>
2845 33      </Components>
2846 34    </Systems>
2847 35    <Auxiliaries id="Aux" name="auxiliaries">
2848 36      <Components>
2849 37        <Environmental id="room" name="environmental"/>
2850 38      </Components>
2851 39    </Auxiliaries>
2852 40    <Resources id="resources" name="resources">
2853 41      <Components>
2854 42        <Personnel id="personnel" name="personnel"/>
2855 43        <Materials id="materials" name="materials">
2856 44          <Components>
2857 45            <Stock id="procstock" name="stock"/>
2858 46          </Components>
2859 47        </Materials/>
2860 48      </Components>
2861 49    </Resources>
2862 50  </Components>
2863 51 </Device>
2864 52 </Devices>

```

## 2865 C.2 Component Example

### Example 2: Component Example

```

2866 1 <Devices>
2867 2   <Device id="d1" name="M12346" uuid="M80104K162N">
2868 3     <Description manufacturer="Example\textunderscore_Corporation"
2869 4       serialNumber="272237"> Mill w/SMooth-G
2870 5   </Description>
2871 6   <DataItems>

```

```

2872 7      <DataItem category="EVENT" id="avail" name="avail" type="
2873 AVAILABILITY"/>
2874 8      <DataItem category="EVENT" id="dl\textunderscore_asset\
2875 textunderscore_chg" name="asset\textunderscore_chg" type="
2876 ASSET\textunderscore_CHANGED"/>
2877 9      <DataItem category="EVENT" id="dl\textunderscore_asset\
2878 textunderscore_rem" name="asset\textunderscore_rem" type="
2879 ASSET\textunderscore_REMOVED"/>
2880 10     </DataItems>
2881 11     <Components>
2882 12     <Controller id="cont" name="controller">
2883 13         <DataItems>
2884 14         <DataItem category="EVENT" id="estop" name="estop" type="
2885 EMERGENCY\textunderscore_STOP"/>
2886 15         <DataItem category="CONDITION" id="logic\textunderscore_cond"
2887 type="LOGIC\textunderscore_PROGRAM"/>
2888 16         <DataItem category="CONDITION" id="cont\textunderscore_system"
2889 type="SYSTEM"/>
2890 17         <DataItem category="SAMPLE" id="cut\textunderscore_time" type="
2891 "ACCUMULATED\textunderscore_TIME" units="SECOND"/>
2892 18     </DataItems>
2893 19     <Components>
2894 20         <Path id="path1" name="path">
2895 21             <DataItems>
2896 22             <DataItem category="EVENT" id="execution" name="execution"
2897 type="EXECUTION"/>
2898 23             <DataItem category="SAMPLE" id="cspeed" subType="ACTUAL"
2899 type="CUTTING\textunderscore_SPEED" units="MILLIMETER/
2900 SECOND"/>
2901 24             <DataItem category="CONDITION" id="path\textunderscore_
2902 system" type="SYSTEM"/>
2903 25             <DataItem category="EVENT" id="cvars" representation="DATA\
2904 textunderscore_SET" type="VARIABLE"/>
2905 26             <DataItem category="EVENT" id="woffset" representation="
2906 TABLE" type="WORK\textunderscore_OFFSET"/>
2907 27             </DataItems>
2908 28         </Path>
2909 29     </Components>
2910 30 </Controller>
2911 31 </Components>
2912 32 </Device>
2913 33 </Devices>

```

### 2914 C.3 Component with Compositions Example

2915 In XML, Composition types are represented differently than Component types. For  
2916 Component types, the element name is Pascal Case of the Component type name.

2917 Whereas, the element name for all Compositiontypes is Composition and the type  
 2918 is defined by the type attribute of the element (see example below).

### Example 3: Component with Compositions Example

```

2919 1 <WorkEnvelope id="we1" name="workenv">
2920 2   <DataItems>
2921 3     <DataItem category="CONDITION" id="hardware1" name="hardware1"
2922 4       type="HARDWARE"/>
2923 4   </DataItems>
2924 5   <Compositions>
2925 6     <Composition type="WORKPIECE" id="wp"/>
2926 7     <Composition type="TABLE" id="tbl"/>
2927 8   </Compositions>
2928 9 </WorkEnvelope>

```

## 2929 C.4 CoordinateSystem Example

### Example 4: CoordinateSystem Example

```

2930 1 <Devices>
2931 2   <Device id="d1" name="M12346" uuid="M80104K162N">
2932 3     <Description manufacturer="Example\textunderscore_Corporation"
2933 4       serialNumber="272237"> Mill w/SMooth-G
2934 5     </Description>
2935 6     <DataItems>
2936 7       <DataItem id="avail" type="AVAILABILITY" category="EVENT"/>
2937 8       <DataItem category="EVENT" id="d1\textunderscore_asset\
2938 9         textunderscore_chg" name="asset\textunderscore_chg" type="
2939 10        ASSET\textunderscore_CHANGED"/>
2940 11      <DataItem category="EVENT" id="d1\textunderscore_asset\
2941 12        textunderscore_rem" name="asset\textunderscore_rem" type="
2942 13        ASSET\textunderscore_REMOVED"/>
2943 14    </DataItems>
2944 15    <Configuration>
2945 16      <CoordinateSystems>
2946 17        <CoordinateSystem id="base" type="BASE">
2947 18          <Origin>0 0 0</Origin>
2948 19        </CoordinateSystem>
2949 20        <CoordinateSystem id="machcoord" type="MACHINE" parentIdRef=
2950 21          "base">
2951 22          <Transformation>
2952 23            <Translation>210 275 1430</Translation>
2953 24            <Rotation>0 0 0</Rotation>
2954 25          </Transformation>
2955 26        </CoordinateSystem>
2956 27        <CoordinateSystem id="workcoord" type="OBJECT" parentIdRef="
2957 28          machcoord">
2958 29          <Transformation>

```

```

2959 24      <Translation>0 0 0</Translation>
2960 25      <Rotation>0 0 0</Rotation>
2961 26      </Transformation>
2962 27      </CoordinateSystem>
2963 28      </CoordinateSystems>
2964 29      </Configuration>
2965 30      </Device>
2966 31      </Devices>

```

## 2967 C.5 Motion Example

### Example 5: Motion Example

```

2968 1  <Devices>
2969 2  <Device id="d1" name="M12346" uuid="M80104K162N">
2970 3      <Description manufacturer="Example\textunderscore_Corporation"
2971 4          serialNumber="272237"> Mill w/SMooth-G
2972 5      </Description>
2973 6      <DataItems>
2974 7          <DataItem id="avail" type="AVAILABILITY" category="EVENT"/>
2975 8          <DataItem category="EVENT" id="d1\textunderscore_asset\
2976          textunderscore_chg" name="asset\textunderscore_chg" type="
2977          ASSET\textunderscore_CHANGED"/>
2978 9          <DataItem category="EVENT" id="d1\textunderscore_asset\
2979          textunderscore_rem" name="asset\textunderscore_rem" type="
2980          ASSET\textunderscore_REMOVED"/>
2981 10     </DataItems>
2982 11     <Configuration>
2983 12         <CoordinateSystems>
2984 13             <CoordinateSystem id="base" type="BASE">
2985 14                 <Origin>0 0 0</Origin>
2986 15             </CoordinateSystem>
2987 16             <CoordinateSystem id="machcoord" type="MACHINE" parentIdRef=
2988                 "base">
2989 17                 <Transformation>
2990 18                     <Translation>210 275 1430</Translation>
2991 19                     <Rotation>0 0 0</Rotation>
2992 20                 </Transformation>
2993 21             </CoordinateSystem>
2994 22             <CoordinateSystem id="workcoord" type="OBJECT" parentIdRef="
2995                 machcoord">
2996 23                 <Transformation>
2997 24                     <Translation>0 0 0</Translation>
2998 25                     <Rotation>0 0 0</Rotation>
2999 26                 </Transformation>
3000 27             </CoordinateSystem>
3001 28         </CoordinateSystems>
3002 29     </Configuration>

```

```

3003 30      <Components>
3004 31          <Axes id="a" name="base">
3005 32              <Components>
3006 33                  <Linear id="x" name="X">
3007 34                      <Configuration>
3008 35                          <Motion id="xmotion" coordinateSystemIdRef="machcoord"
3009 36                          type="PRISMATIC" actuation="DIRECT">
3010 37                              <Transformation>
3011 38                                  <Translation>300 915 590</Translation>
3012 39                                  <Rotation>0 0 0</Rotation>
3013 40                              </Transformation>
3014 41                              <Axis>1.0 0 0</Axis>
3015 42                          </Motion>
3016 43                      </Configuration>
3017 44                  </Linear>
3018 45                  <DataItems>
3019 46                      <DataItem id="xtravel" type="SYSTEM" category="CONDITION
3020 47                      ">
3021 48                          </DataItems>
3022 49                  </Components>
3023 50              </Axes>
3024 51          </Components>
3025 52      </Device>
3026 53  </Devices>

```

## 3027 C.6 Relationship Example

### Example 6: Relationship Example

```

3028 1  <Components>
3029 2      <Axes id="a" name="base">
3030 3          <Components>
3031 4              <Linear id="x" name="X">
3032 5                  <Configuration>
3033 6                      <Relationships>
3034 7                          <ComponentRelationship id="xpar" type="PARENT" idRef="
3035 8                          we1"/>
3036 9                      </Relationships>
3037 10                  </Configuration>
3038 11              </Linear>
3039 12          <DataItems>
3040 13              <DataItem id="xtravel" type="SYSTEM" category="CONDITION">
3041 14              </DataItems>
3042 15          </Components>
3043 16      </Axes>
3044 17      <Systems id="systems" name="systems">
3045 18          <Components>
3046 19              <WorkEnvelope id="we1" name="workenv">

```

```

3047 19      <Compositions>
3048 20          <Composition type="WORKPIECE" id="wp"/>
3049 21          <Composition type="TABLE" id="tbl"/>
3050 22      </Compositions>
3051 23      <DataItems>
3052 24          <DataItem id="welcond" type="SYSTEM" category="CONDITION">
3053 25      </DataItems>
3054 26  </WorkEnvelope>
3055 27  </Components>
3056 28  </Systems>
3057 29  </Components>

```

## 3058 C.7 SolidModel Example

### Example 7: SolidModel Example

```

3059 1  <Devices>
3060 2  <Device id="d1" name="M12346" uuid="M80104K162N">
3061 3      <Description manufacturer="Example\textunderscore_Corporation"
3062 4          serialNumber="272237"> Mill w/SMooth-G
3063 5  </Description>
3064 6  <DataItems>
3065 7      <DataItem id="avail" type="AVAILABILITY" category="EVENT"/>
3066 8      <DataItem category="EVENT" id="d1\textunderscore_asset\
3067          textunderscore_chg" name="asset\textunderscore_chg" type="
3068          ASSET\textunderscore_CHANGED"/>
3069 9      <DataItem category="EVENT" id="d1\textunderscore_asset\
3070          textunderscore_rem" name="asset\textunderscore_rem" type="
3071          ASSET\textunderscore_REMOVED"/>
3072 10 </DataItems>
3073 11 <Configuration>
3074 12 <CoordinateSystems>
3075 13 <CoordinateSystem id="base" type="BASE">
3076 14 <Origin>0 0 0</Origin>
3077 15 </CoordinateSystem>
3078 16 <CoordinateSystem id="machcoord" type="MACHINE" parentIdRef=
3079     "base">
3080 17 <Transformation>
3081 18 <Translation>210 275 1430</Translation>
3082 19 <Rotation>0 0 0</Rotation>
3083 20 </Transformation>
3084 21 </CoordinateSystem>
3085 22 <CoordinateSystem id="workcoord" type="OBJECT" parentIdRef="
3086     machcoord">
3087 23 <Transformation>
3088 24 <Translation>0 0 0</Translation>
3089 25 <Rotation>0 0 0</Rotation>
3090 26 </Transformation>

```

```

3091 27         </CoordinateSystem>
3092 28     </CoordinateSystems>
3093 29 </Configuration>
3094 30 <Components>
3095 31     <Structures id="struct">
3096 32         <Configuration>
3097 33             <SolidModel id="model" mediaType="OBJ" href="/objs/mazak.
3098 obj" coordinateSystemIdRef="base">
3099 34                 <Transformation>
3100 35                     <Translation>0 860 0</Translation>
3101 36                     <Rotation>-90 0 0</Rotation>
3102 37                 </Transformation>
3103 38                 <Scale>0.001 0.001 0.001</Scale>
3104 39             </SolidModel>
3105 40         </Configuration>
3106 41     <Components>
3107 42         <Structure id="xaxis" name="X\textunderscore_AXIS">
3108 43             <Configuration>
3109 44                 <SolidModel id="x\textunderscore_model" mediaType="OBJ
3110 " itemRef="xaxis" solidModelIdRef="model" coordinateSystemIdRef="
3111 base"/>
3112 45             </Configuration>
3113 46             <DataItems>
3114 47                 <DataItem type="SYSTEM" category="CONDITION" id="
3115 struct\textunderscore_cond"/>
3116 48             </DataItems>
3117 49         </Structure>
3118 50     </Components>
3119 51 </Structures>
3120 52 </Components>
3121 53 </Device>
3122 54 </Devices>

```

## 3123 C.8 Specification Example

### Example 8: Specification Example

```

3124 1 <Components>
3125 2     <Axes id="a" name="base">
3126 3         <Components>
3127 4             <Linear id="x" name="X">
3128 5                 <Configuration>
3129 6                     <Specifications>
3130 7                         <Specification id="spec001" type="LENGTH" units="
3131 MILLIMETER">
3132 8                             <Nominal>650</Nominal>
3133 9                         </Specification>
3134 10                        <Specification id="spec002" type="LINEAR\textunderscore_
3135 FORCE" units="NEWTON">

```

```

3136 11         <Maximum>5200</Maximum>
3137 12         <Minimum>0</Minimum>
3138 13     </Specification>
3139 14 </Specifications>
3140 15 </Configuration>
3141 16 <DataItems>
3142 17     <DataItem id="xtravel" type="SYSTEM" category="CONDITION">
3143 18 </DataItems>
3144 19 </Linear>
3145 20 </Components>
3146 21 </Axes>
3147 22 </Components>

```

## 3148 C.9 Example of sensing element provided as data item associated with 3149 a Component

**Example 9:** Example of sensing element provided as data item associated with a Component

```

3150 1 <Components>
3151 2     <Axes id="a" name="base"
3152 3         <Components>
3153 4             <Rotary id="c" name="C">
3154 5                 <DataItems>
3155 6                     <DataItem type="TEMPERATURE" id="servotemp4"
3156 7                         category="SAMPLE" nativeUnits="CELSIUS"
3157 8                         compositionId="Bmotor" units="CELSIUS"/>
3158 9                 </DataItems>
3159 10            </Rotary>
3160 11        </Components>
3161 12    </Axes>
3162 13 </Components>

```

## 3163 C.10 Example of Sensor for rotary axis

**Example 10:** Example of Sensor for rotary axis

```

3164 1 <Components>
3165 2     <Axes id="a" name="base"
3166 3         <Components>
3167 4             <Rotary id="ar" name="B">
3168 5                 <Components>
3169 6                     <Sensor id="spdlm" name="Spindlemonitor">
3170 7                         <DataItems>
3171 8                             <DataItem type="DISPLACEMENT" id="cvib"

```



```

3172 9          category="SAMPLE" name="Svib"
3173 10          units="MILLIMETER"/>
3174 11      </DataItems>
3175 12  </Sensor >
3176 13  <Components>
3177 14      </Rotary>
3178 15      </Components>
3179 16  </Axes>
3180 17 </Components>

```

## 3181 C.11 Example of sensor unit with sensing element

### Example 11: Example of sensor unit with sensing element

```

3182 1 <Axes id="a" name="base"
3183 2   <Components>
3184 3   <Sensor id="sens1" name="Sensorunit">
3185 4       <DataItems>
3186 5           <DataItem type="TEMPERATURE" id="sentemp"
3187 6               category="SAMPLE" name="Sensortemp"
3188 7               units="CELSIUS"/>
3189 8       </DataItems>
3190 9   </Sensor >
3191 10  <Linear id="x" name="X">
3192 11      <DataItems>
3193 12          <DataItem type="DISPLACEMENT" id="xvib"
3194 13              category="SAMPLE" name="xvib"
3195 14              units="MILLIMETER">
3196 15              <Source componentId="sens1"/>
3197 16          </DataItem>
3198 17      </DataItems>
3199 18  </Rotary>
3200 19  <Linear id="y" name="Y">
3201 20      <DataItems>
3202 21          <DataItem type="DISPLACEMENT" id="yvib"
3203 22              category="SAMPLE" name="yvib"
3204 23              units="MILLIMETER">
3205 24              <Source componentId="sens1"/>
3206 25          </DataItem>
3207 26      </DataItems>
3208 27  </Linear>
3209 28  <Components>
3210 29 </Axes>

```

## 3211 C.12 Example of configuration data for Sensor

**Example 12:** Example of configuration data for Sensor

```

3212 1  <Sensor id="sensor" name="sensor">
3213 2    <Configuration>
3214 3      <SensorConfiguration>
3215 4        <FirmwareVersion>2.02</FirmwareVersion>
3216 5        <CalibrationDate>2010-05-16</CalibrationDate>
3217 6        <NextCalibrationDate>2010-05-16</NextCalibrationDate>
3218 7        <CalibrationInitials>WS</CalibrationInitials>
3219 8        <Channels>
3220 9          <Channel number="1" name="A/D:1">
3221 10            <Description>A/D With Thermister</Description>
3222 11          </Channel>
3223 12        </Channels>
3224 13      </SensorConfiguration>
3225 14    </Configuration>
3226 15    <DataItems>
3227 16      <DataItem category="CONDITION" id="sensorc"
3228 17        name="sensorc" type="SYSTEM" />
3229 18      <DataItem category="SAMPLE" id="senv" name="sensorc"
3230 19        type="VOLTAGE\textunderscore_DC" units="VOLT" subType="ACTUAL"
3231 20        />
3232 21    </DataItems>
3233 22  </Sensor>

```

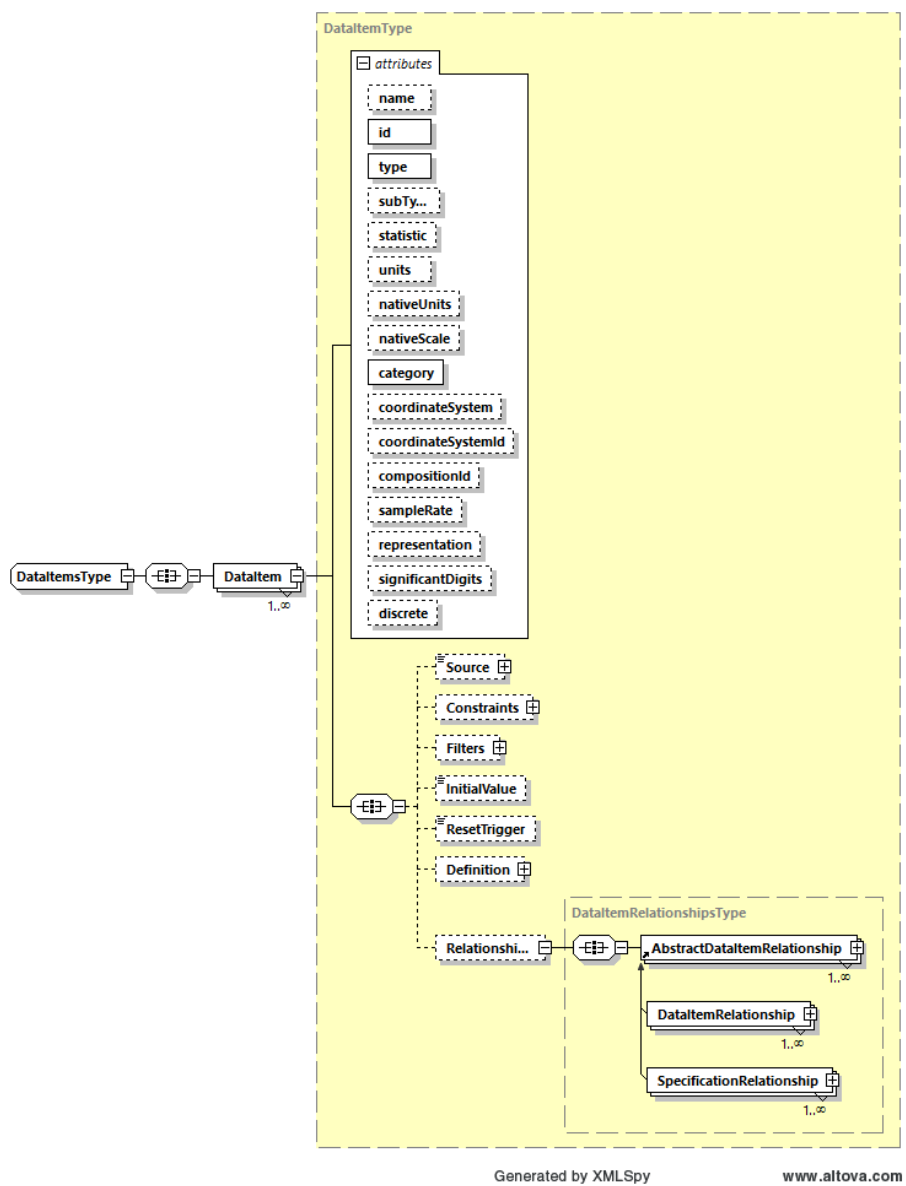
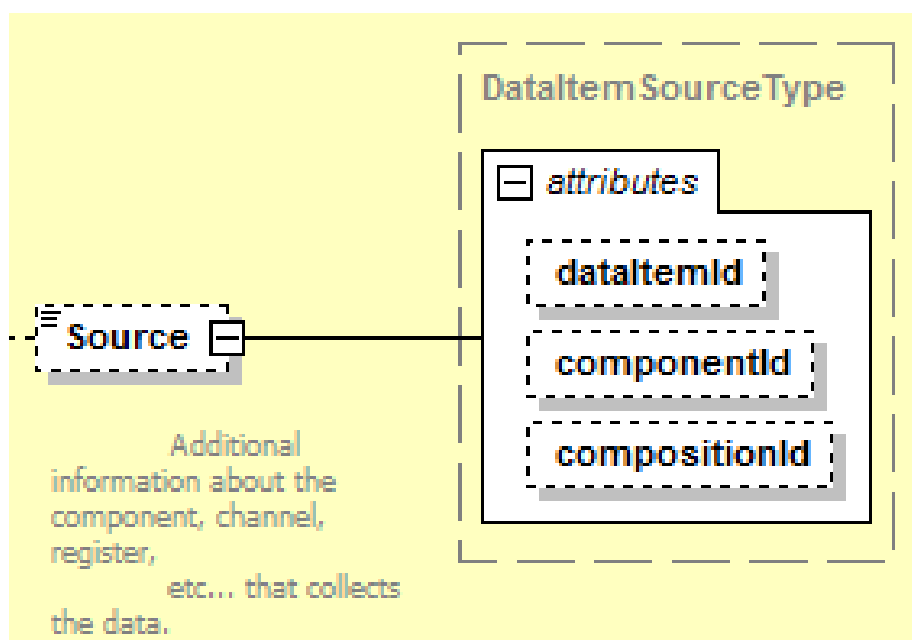
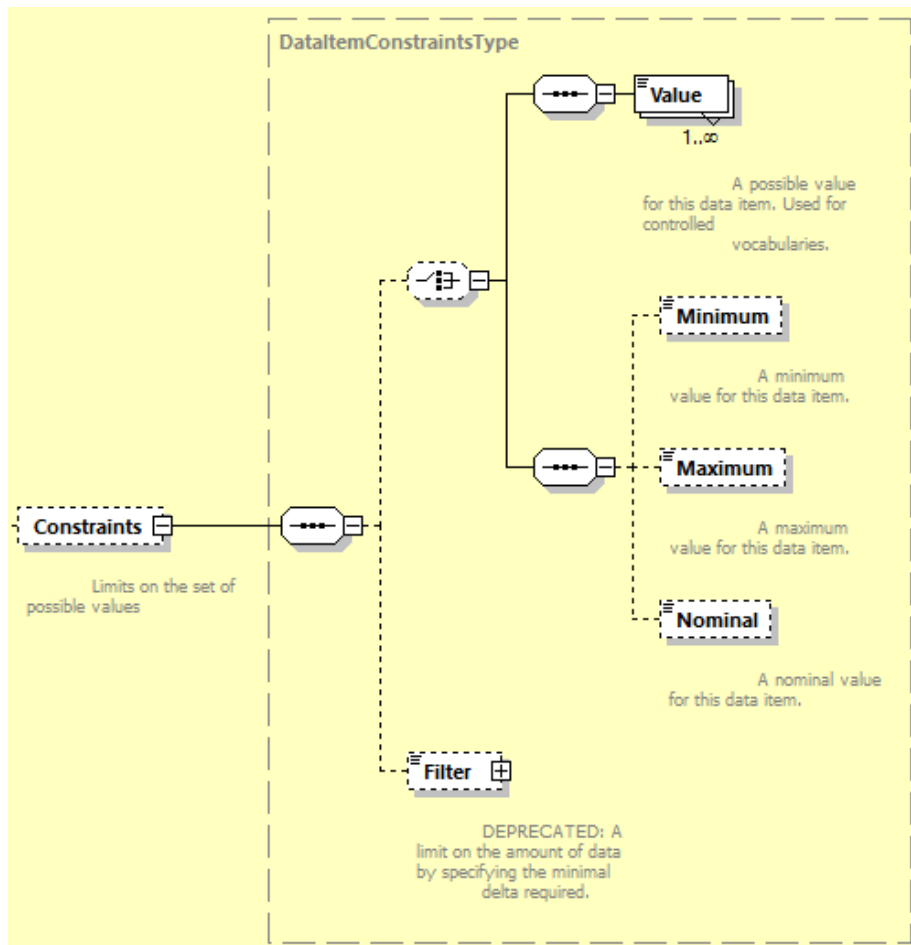


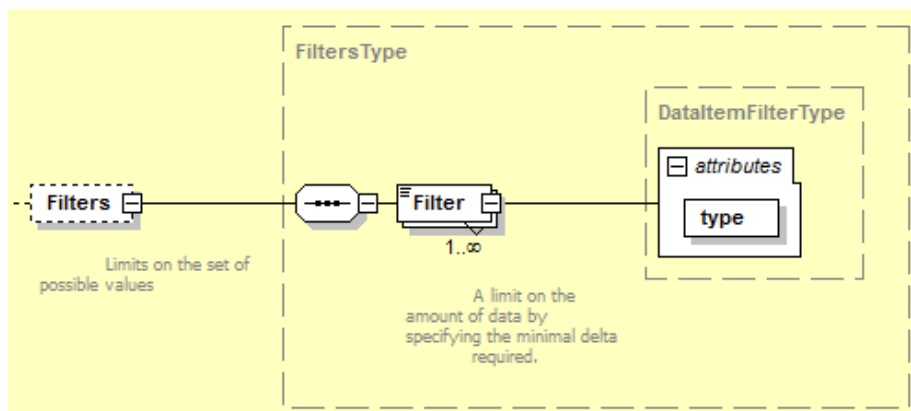
Figure 27: DataItems Schema



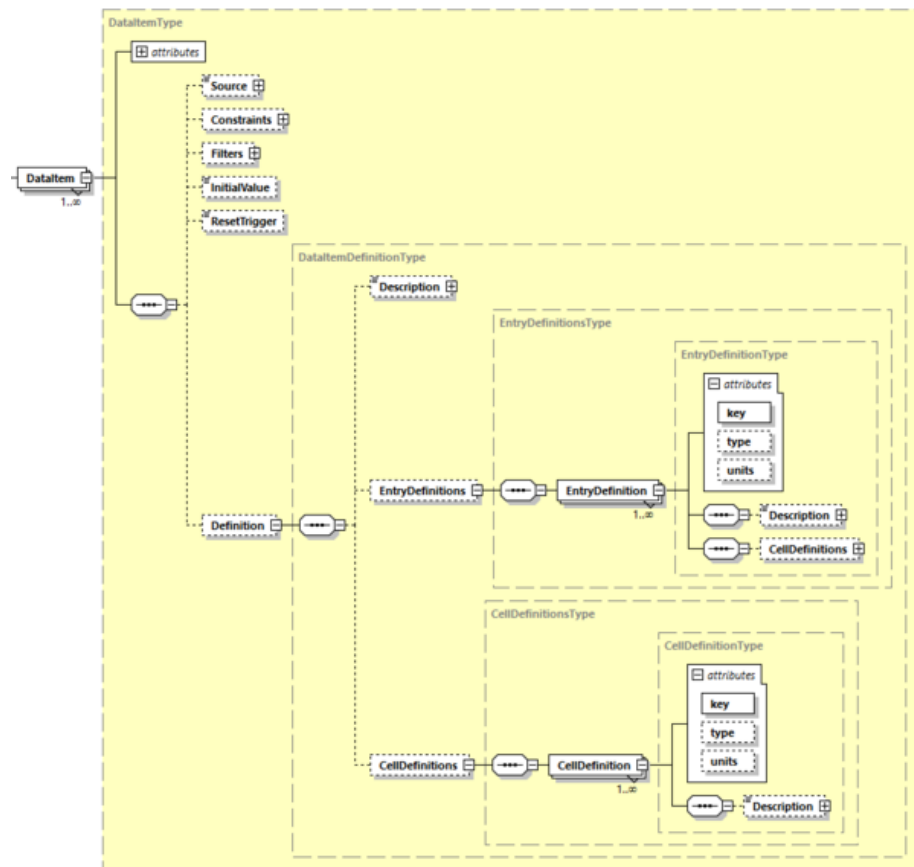
**Figure 28:** Source Schema



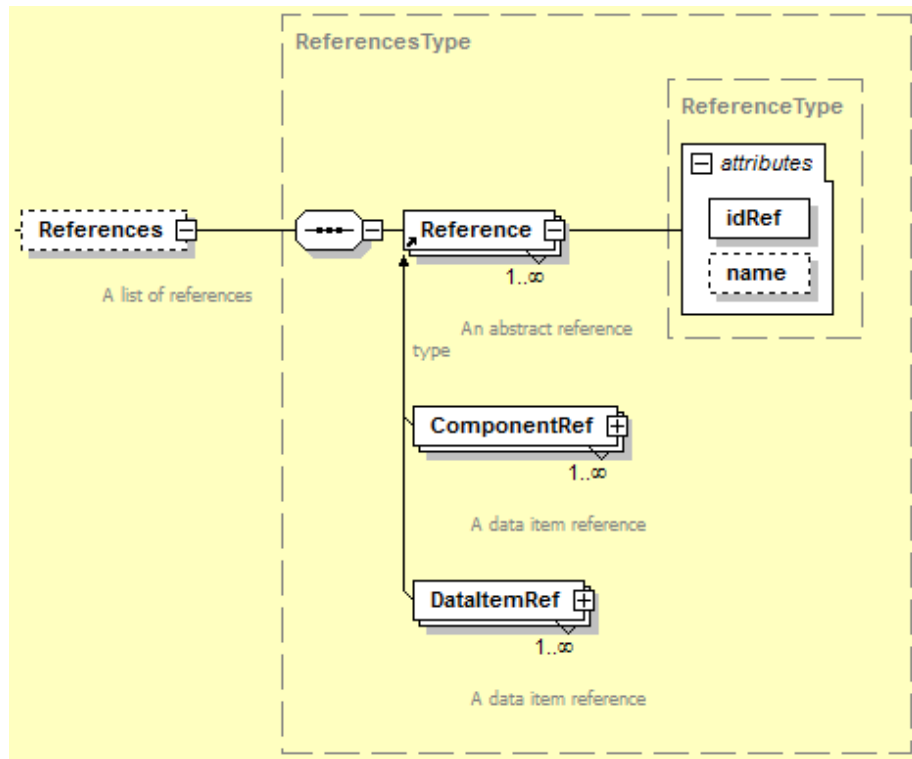
**Figure 29: Constraints Schema**



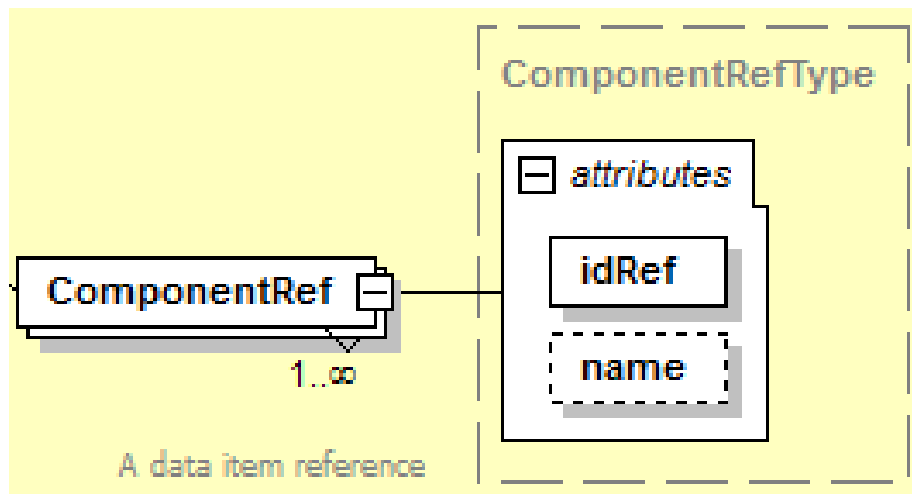
**Figure 30: Filter Schema**



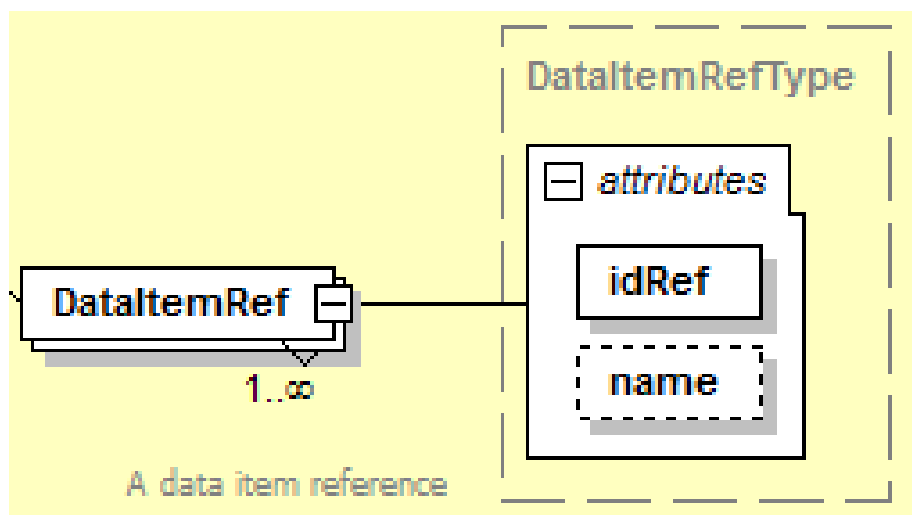
**Figure 31:** Definition Schema



**Figure 32: References Schema**

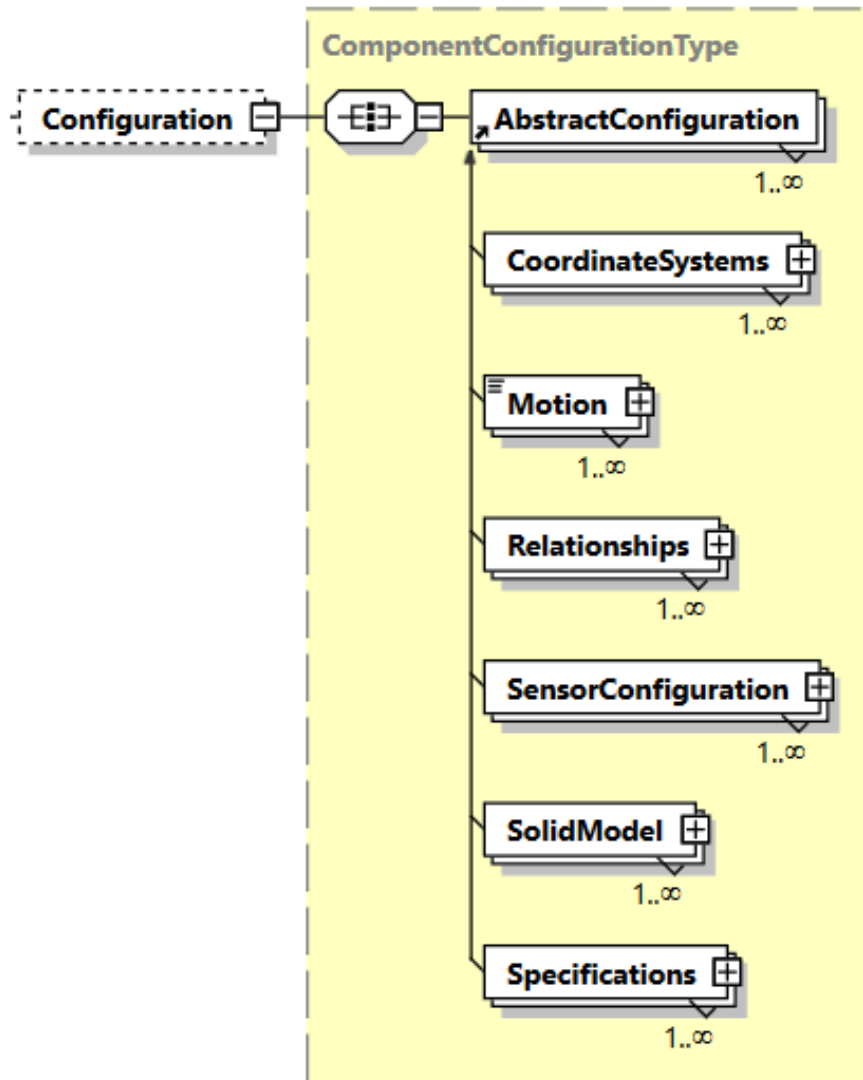


**Figure 33: ComponentRef Schema**

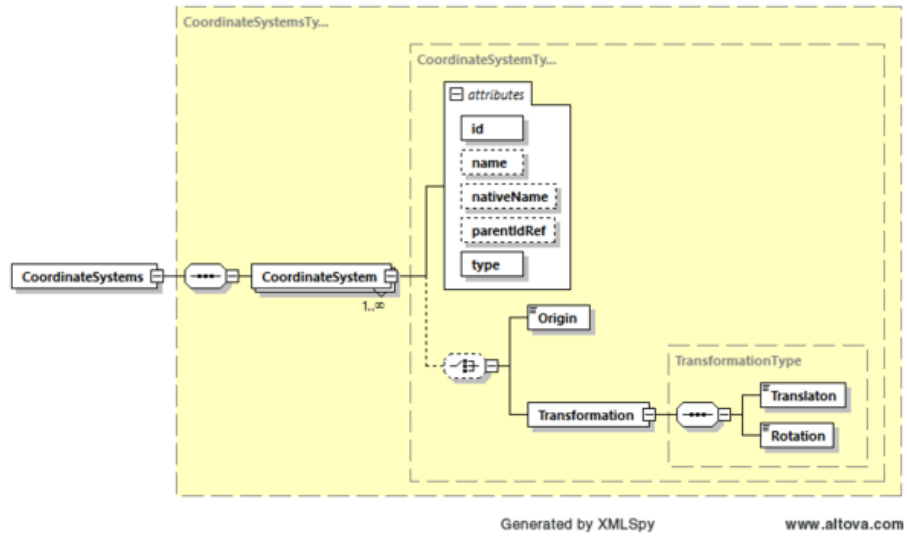


**Figure 34:** DataItemRef Schema

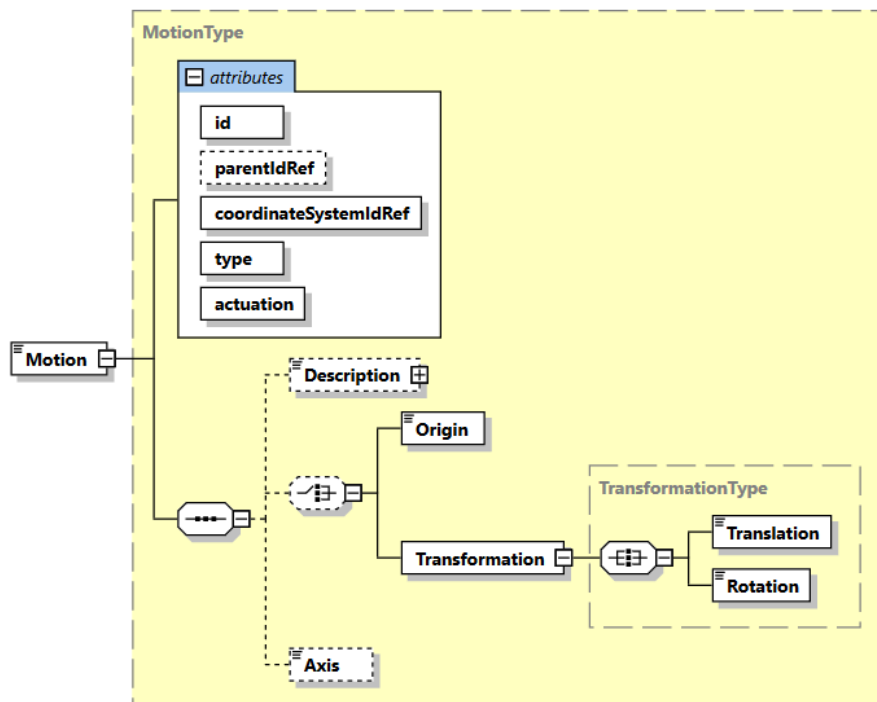




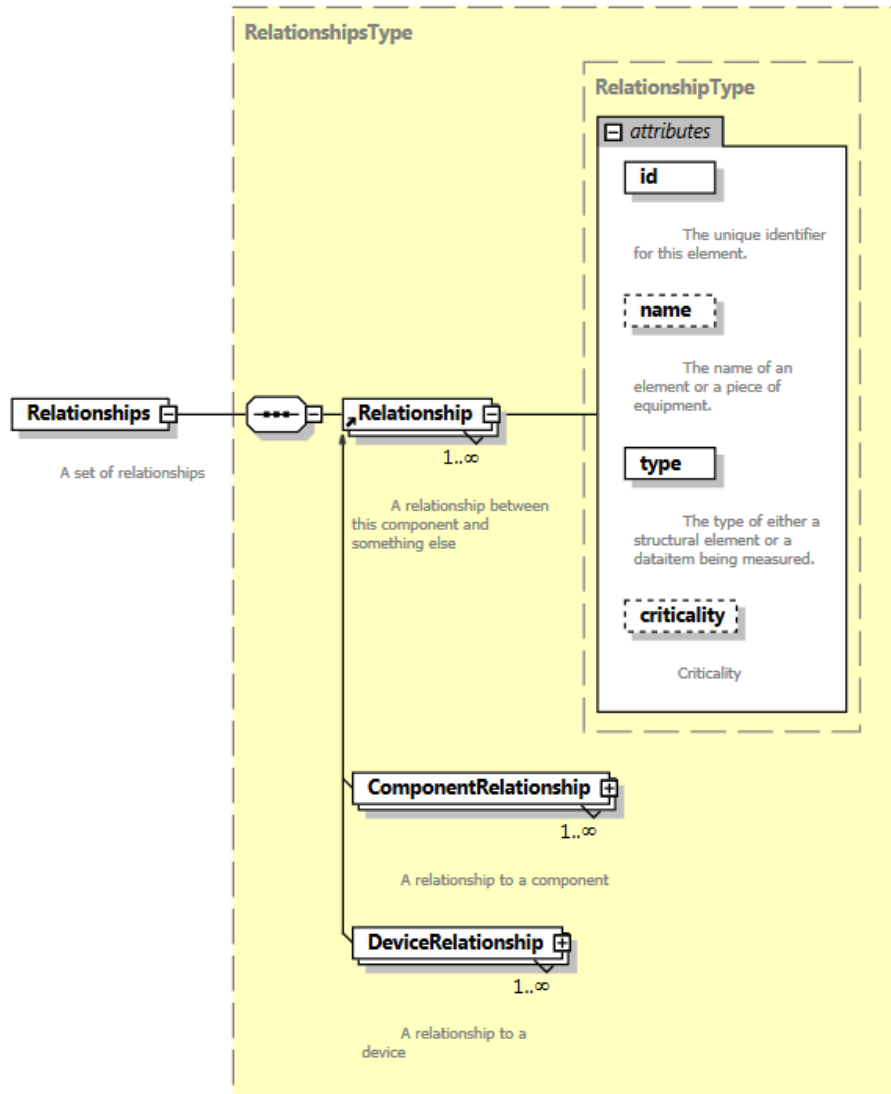
**Figure 35:** Configuration Schema



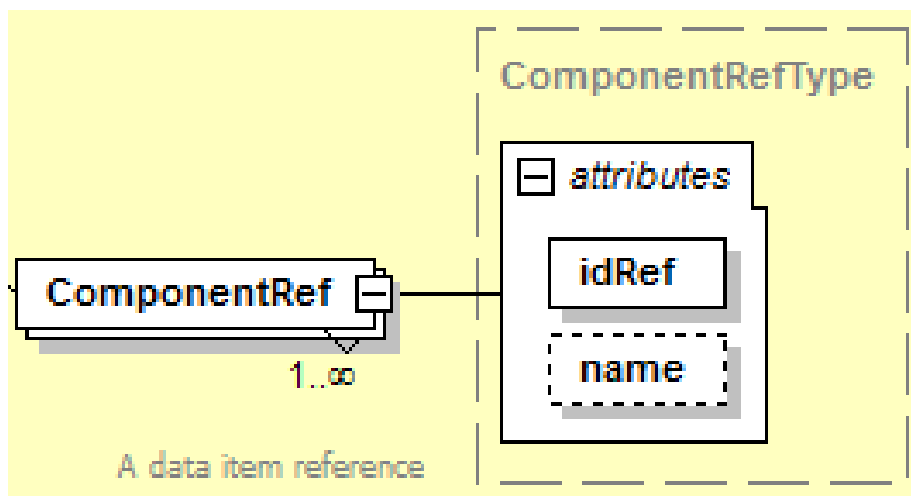
**Figure 36:** CoordinateSystem Schema



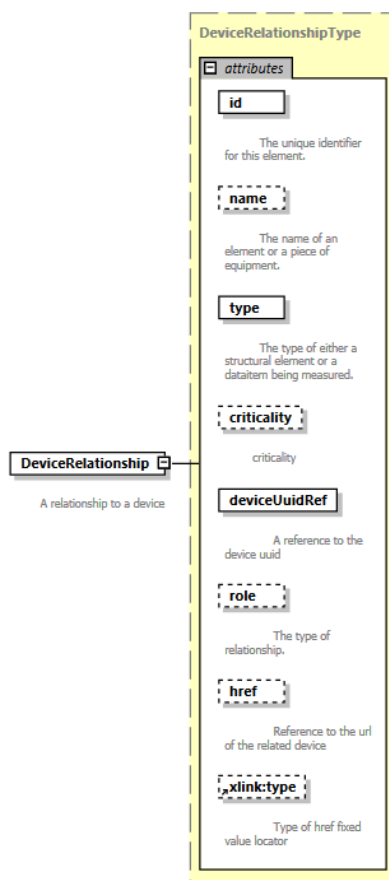
**Figure 37:** Motion Schema



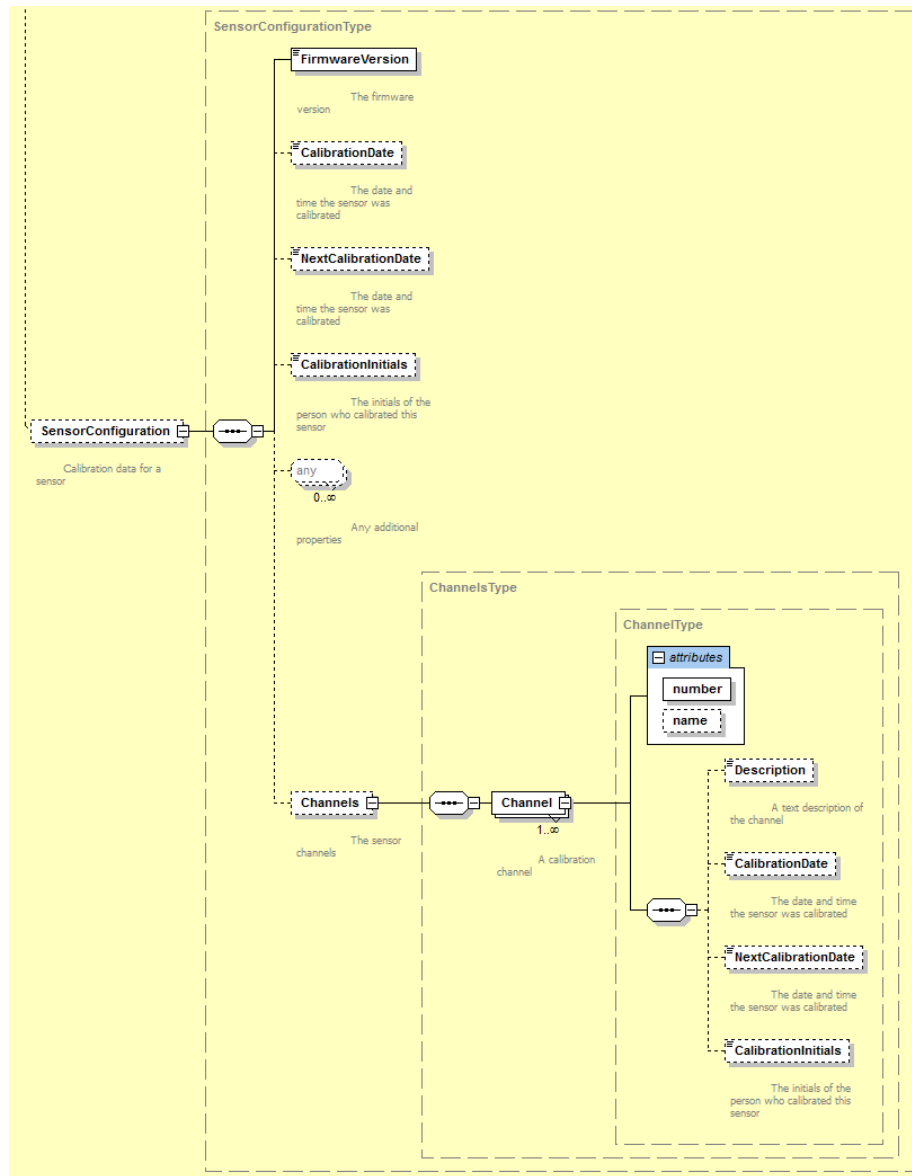
**Figure 38:** Relationships Schema



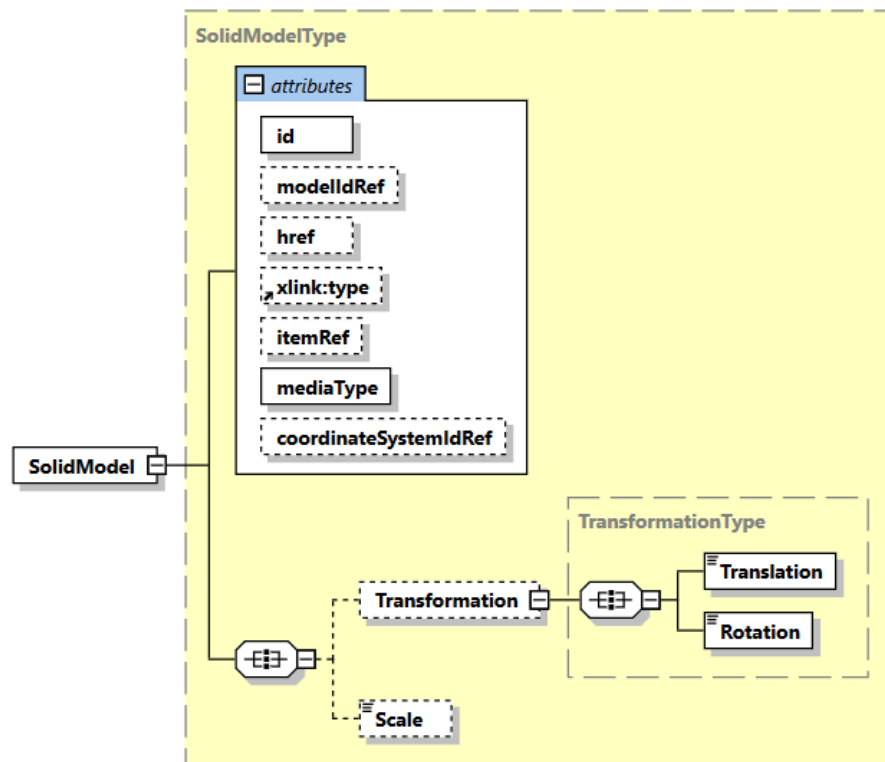
**Figure 39:** ComponentRelationship Schema



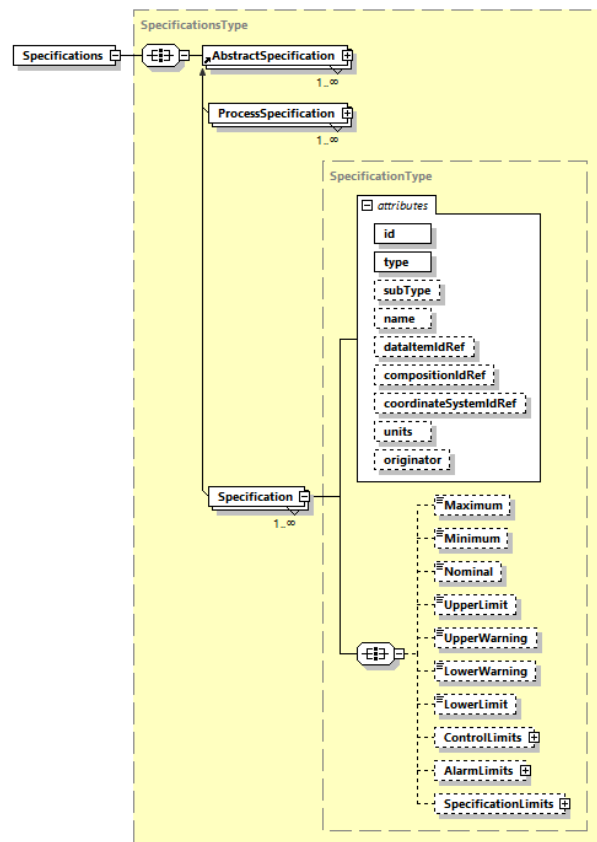
**Figure 40:** DeviceRelationship Schema



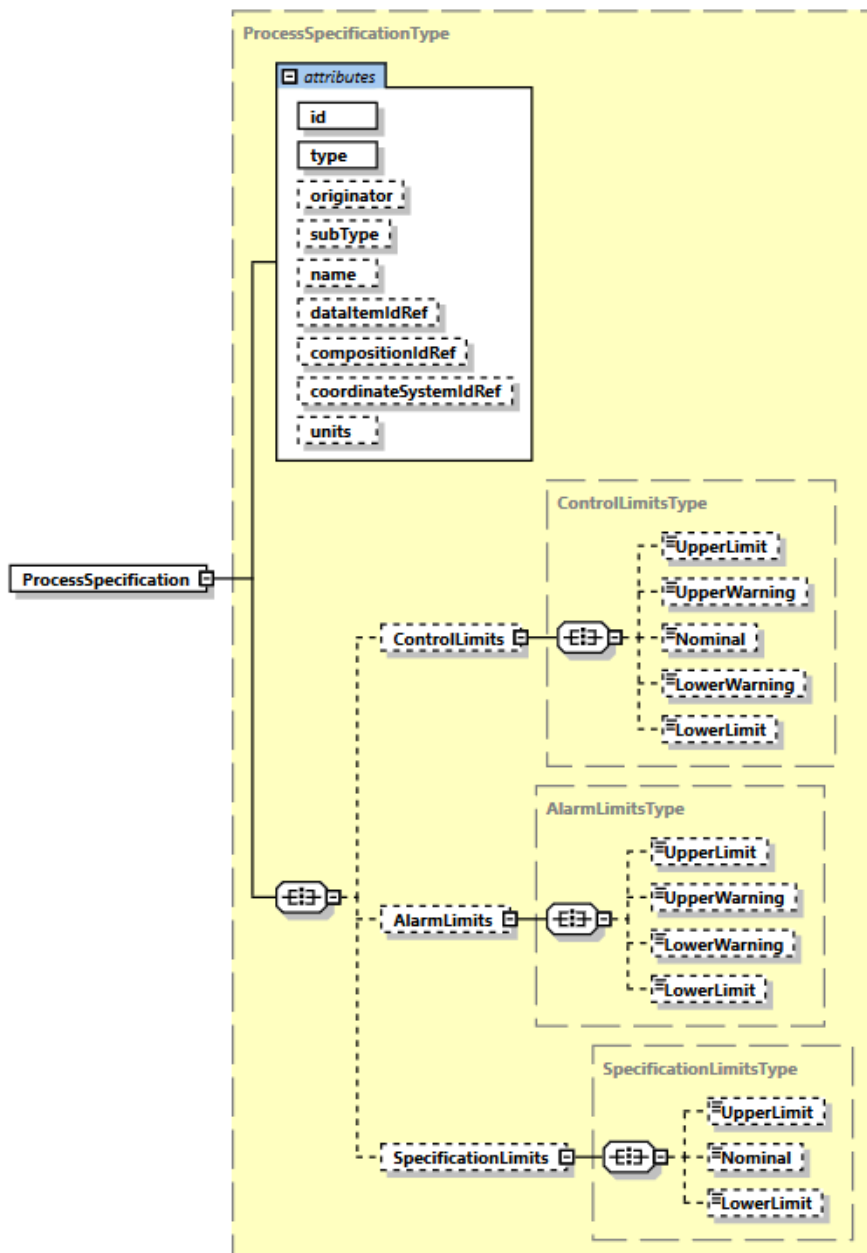
**Figure 41: SensorConfiguration Schema**



**Figure 42:** SolidModel Schema



**Figure 43:** Specifications Schema



**Figure 44:** ProcessSpecification Schema