



MTConnect® Standard
Guide: MTConnect and OPC/UA
Companion Specification
Version 2.0

Prepared for: MTConnect Institute
Prepared by: William Sobel
Prepared on: September 29, 2018

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

1 MTConnect[®] Specification and Materials

2

3 AMT – The Association For Manufacturing Technology (“AMT”) owns the copy-
4 right in this MTConnect[®] Specification or Material. AMT grants to you a non-
5 exclusive, non-transferable, revocable, non-sublicensable, fully-paid-up copyright
6 license to reproduce, copy and redistribute this MTConnect[®] Specification or Ma-
7 terial, provided that you may only copy or redistribute the MTConnect[®] Speci-
8 fication or Material in the form in which you received it, without modifications,
9 and with all copyright notices and other notices and disclaimers contained in the
10 MTConnect[®] Specification or Material.

11 If you intend to adopt or implement an MTConnect[®] Specification or Material
12 in a product, whether hardware, software or firmware, which complies with an
13 MTConnect[®] Specification, you **SHALL** agree to the MTConnect[®] Specifica-
14 tion Implementer License Agreement (“Implementer License”) or to the MTConnect[®]
15 Intellectual Property Policy and Agreement (“IP Policy”). The Implementer Li-
16 cense and IP Policy each sets forth the license terms and other terms of use for
17 MTConnect[®] Implementers to adopt or implement the MTConnect[®] Specifica-
18 tions, including certain license rights covering necessary patent claims for that
19 purpose. These materials can be found at www.MTConnect.org, or by contact-
20 ing Paul Warndorf at [mailto:pwarndorf@mtconnect.hyperoffice.](mailto:pwarndorf@mtconnect.hyperoffice.com)
21 [com](http://www.MTConnect.org).

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

29 This Material and all MTConnect[®] Specifications and Materials are provided “as
30 is” and MTConnect[®] Institute and AMT, and each of their respective members,
31 officers, affiliates, sponsors and agents, make no representation or warranty of
32 any kind relating to these materials or to any implementation of the MTConnect[®]
33 Specifications or Materials in any product, including, without limitation, any ex-
34 pressed or implied warranty of noninfringement, merchantability, or fitness for

September 29, 2018

35 particular purpose, or of the accuracy, reliability, or completeness of information
36 contained herein. In no event shall MTConnect[®] Institute or AMT be liable to
37 any user or implementer of MTConnect[®] Specifications or Materials for the cost
38 of procuring substitute goods or services, lost profits, loss of use, loss of data or
39 any incidental, consequential, indirect, special or punitive damages or other di-
40 rect damages, whether under contract, tort, warranty or otherwise, arising in any
41 way out of access, use or inability to use the MTConnect[®] Specification or other
42 MTConnect[®] Materials, whether or not they had advance notice of the possibility
43 of such damage.

44 Table of Contents

45	1 Introduction	1
46	1.1 Overview	1
47	2 Types	1
48	2.1 Components	1
49	2.1.1 Defintion of ChannelType	1
50	2.1.2 Defintion of DescriptionType	1
51	2.1.3 Defintion of MTComponentType	3
52	2.1.4 Defintion of MTCompositionType	5
53	2.1.5 Defintion of MTConfigurationType	5
54	2.1.6 Defintion of MTDeviceType	5
55	2.1.7 Defintion of SensorConfigurationType	6
56	2.1.8 Defintion of ComponentType	6
57	2.1.9 Defintion of CompositionType	6
58	2.2 Data Items	8
59	2.2.1 Defintion of AssetChangedType	8
60	2.2.2 Defintion of AssetEventType	8
61	2.2.3 Defintion of AssetRemovedType	9
62	2.2.4 Defintion of MTDataItemType	9
63	2.2.5 Defintion of MTEnumeratedEventType	10
64	2.2.6 Defintion of MTFilterType	11
65	2.2.7 Defintion of MTMessageType	11
66	2.2.8 Defintion of MTNumericDataItemType	11
67	2.2.9 Defintion of MTNumericEventType	12
68	2.2.10 Defintion of MTSampleType	12
69	2.2.11 Defintion of MTStringEventType	13
70	2.2.12 Defintion of MinimumDeltaFilterType	14
71	2.2.13 Defintion of PeriodFilterType	14
72	2.2.14 Defintion of DataItemType	14
73	2.3 Conditions	15
74	2.3.1 Defintion of MTExclusiveLimitConditionType	15
75	2.3.2 Defintion of MTNonExclusiveConditionType	15
76	2.3.3 Defintion of ConditionClassType	15
77	2.4 Factories	17
78	2.4.1 Defintion of ComponentObjectFactory	17
79	2.4.2 Defintion of ComponentTypeFactory	17

80	2.4.3	Defintion of CompositionObjectFactory	18
81	2.4.4	Defintion of CompositionTypeFactory	18
82	2.4.5	Defintion of ConditionClassFactory	18
83	2.4.6	Defintion of ConditionObjectFactory	19
84	2.4.7	Defintion of DataItemObjectFactory	19
85	2.4.8	Defintion of DataItemTypeFactory	20
86	2.4.9	Defintion of DeviceObjectFactory	20
87	2.4.10	Defintion of FilterObjectFactory	21
88	2.4.11	Defintion of ObjectFactory	21
89	2.4.12	Defintion of SensorChannelObjectFactory	22
90	2.4.13	Defintion of SensorObjectFactory	22
91	2.4.14	Defintion of TypeFactory	22
92	2.5	MTConnect Device Profile	22
93	2.5.1	Defintion of Dynamic Type	24
94	2.5.2	Defintion of MIXes In	24
95	2.5.3	Defintion of MTConnect XML	24
96	2.5.4	Defintion of MTRelationshipType	25
97	2.5.5	Defintion of Object Factory	25
98	2.5.6	Defintion of Type Factory	25
99	2.5.7	Defintion of bind	25
100	2.5.8	Defintion of constrains	26
101	2.5.9	Defintion of mixin	26
102	2.5.10	Defintion of use	27

103 **List of Figures**

104 **Figure 1: Components Diagram** 2

105 **Figure 2: Data Items Diagram** 8

106 **Figure 3: Conditions Diagram** 16

107 **Figure 4: Factories Diagram** 17

108 **Figure 5: MTConnect Device Profile Diagram** 23

1 Introduction

109 The following conventions will be used throughout the document to provide a
110 clear and consistent understanding of the use of each type of data and information
111 used to define the MTConnect[®] standard and associated data.

1.1 Overview

112 Overview of the standards...

2 Types

2.1 Components

113 The Components documents the Component models and the owned objects.

2.1.1 Defintion of ChannelType

114 Refer to Table 1 for detailed definition.

2.1.2 Defintion of DescriptionType

115 Exact mirror of the MTConnect Type.

116 Refer to Table 2 for detailed definition.

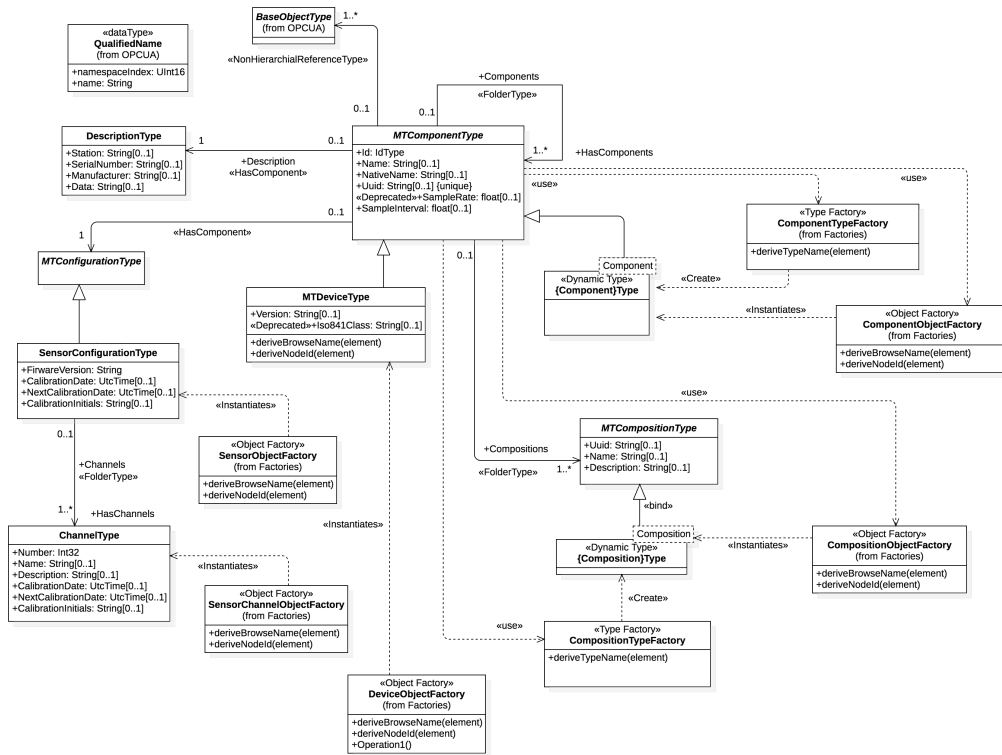


Figure 1: Components Diagram

Table 1: ChannelType Definition

Attribute	Value				
BrowseName	ChannelType				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	TypeDefinition	Modeling Rule
Subtype of BaseObjectType (See OPCUA Documentation)					
HasProperty	Variable	Number	Int32	PropertyType	Mandatory
HasProperty	Variable	Name	String	PropertyType	Optional
HasProperty	Variable	Description	String	PropertyType	Optional
HasProperty	Variable	CalibrationDate	UtcTime	PropertyType	Optional
HasProperty	Variable	NextCalibrationDate	UtcTime	PropertyType	Optional
HasProperty	Variable	CalibrationInitials	String	PropertyType	Optional

Table 2: DescriptionType Definition

Attribute	Value				
BrowseName	DescriptionType				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	TypeDefinition	Modeling Rule
Subtype of BaseObjectType (See OPCUA Documentation)					
HasProperty	Variable	Station	String	PropertyType	Optional
HasProperty	Variable	SerialNumber	String	PropertyType	Optional
HasProperty	Variable	Manufacturer	String	PropertyType	Optional
HasProperty	Variable	Data	String	PropertyType	Optional

2.1.3 Defintion of MTComponentType

117 The base Component Type from which all MTConnect Components are derived
 118 from. The component type factory is used to create the specific OPC/UA types as
 119 subtypes of the MTConnect ‘MTComponentType’. The component types will be
 120 created once for all Component objects of that type based on the ‘QName’ of the
 121 MTConnect XML element.

122 The object factory will instantiate the Component Objects and insert them into
 123 the Components folder with a browse name of the Component QName and the
 124 ‘name’ element if specified surrounded by square brackets, ‘[]’. For example if
 125 the MTConnect Element is:

126 ‘<Linear name=’X’>...</...>’

127 The OPC/UA Object with browse name ‘Linear[X]’ will be created with the
 128 HasTypeDefinition referencing the ‘Linear’ OPC/UA type.

129 The meta data for the component and it’s relationships are static. The dynamic
 130 data will be represented using the _OPC/UA Part 8_

131 Refer to Table 3 for detailed definition.

Table 3: MTComponentType Definition

Attribute	Value				
BrowseName	MTComponentType				
IsAbstract	True				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
HasProperty	Variable	Id	IdType	PropertyType	Mandatory
HasProperty	Variable	Name	String	PropertyType	Optional
HasProperty	Variable	NativeName	String	PropertyType	Optional
HasProperty	Variable	Uuid	String	PropertyType	Optional
HasProperty	Variable	SampleRate	float	PropertyType	Optional
HasProperty	Variable	SampleInterval	float	PropertyType	Optional
HasComponent	Object	Description		DescriptionType	Optional
HasComponent	Object	Configuration		MTConfigurationType	Optional
Organizes	Object	Components	MTComponentType	FolderType	Optional
Organizes	Object	Compositions	MTCompositionType	FolderType	Optional
HasProperty	Variable	<Dynamic>	DataItemType	<Dynamic>	Optional
HasProperty	Variable	<Dynamic>	BaseObjectType	<Dynamic>	Optional
Organizes	Object	Conditions	MTNonExclusiveConditionType	FolderType	Optional
HasProperty	Variable	<Dynamic>	DataItemType	<Dynamic>	Mandatory

Table 4: MTCompositionType Definition

Attribute	Value				
BrowseName	MTCompositionType				
IsAbstract	True				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of BaseObjectType (See OPCUA Documentation)					
HasProperty	Variable	Uuid	String	PropertyType	Optional
HasProperty	Variable	Name	String	PropertyType	Optional
HasProperty	Variable	Description	String	PropertyType	Optional
NonHierarchicalReferenceType	Object	ecomposition	DataItemType	NonHierarchicalReferenceType	Optional

Table 5: MTConfigurationType Definition

Attribute	Value				
BrowseName	MTConfigurationType				
IsAbstract	True				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of BaseObjectType (See OPCUA Documentation)					

2.1.4 Defintion of MTCompositionType

132 Refer to Table 4 for detailed definition.

2.1.5 Defintion of MTConfigurationType

133 Refer to Table 5 for detailed definition.

2.1.6 Defintion of MTDeviceType

134 The MTDevice is a special type whose object will be the root of the device graph.

135 The Device uses the component type factory and the component object factories

Table 6: MTDeviceType Definition

Attribute	Value				
BrowseName	MTDeviceType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTComponentType (see section 2.1.3)					
HasProperty	Variable	Version	String	PropertyType	Optional
HasProperty	Variable	Iso841Class	String	PropertyType	Optional

136 to create each of the first level components.

137 The compositions, relationships, and data items are then recursively created as
 138 one descends the MTConnect information model.

139 Refer to Table 6 for detailed definition.

2.1.7 Definition of SensorConfigurationType

140 The SensorConfiguration browse name will be created as an Object relationship
 141 with the parent component.

142 Refer to Table 7 for detailed definition.

2.1.8 Definition of ComponentType

143 Refer to Table 8 for detailed definition.

2.1.9 Definition of CompositionType

144 Refer to Table 9 for detailed definition.

Table 7: SensorConfigurationType Definition

Attribute	Value				
BrowseName	SensorConfigurationType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTConfigurationType (see section 2.1.5)					
HasProperty	Variable	FirmwareVersion	String	PropertyType	Mandatory
HasProperty	Variable	CalibrationDate	UtcTime	PropertyType	Optional
HasProperty	Variable	NextCalibrationDate	UtcTime	PropertyType	Optional
HasProperty	Variable	CalibrationInitials	String	PropertyType	Optional
Organizes	Object	Channels	ChannelType	FolderType	Optional

Table 8: ComponentType Definition

Attribute	Value				
BrowseName	ComponentType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTComponentType (see section 2.1.3)					

Table 9: CompositionType Definition

Attribute	Value				
BrowseName	CompositionType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTCompositionType (see section 2.1.4)					

Items.png Items.png

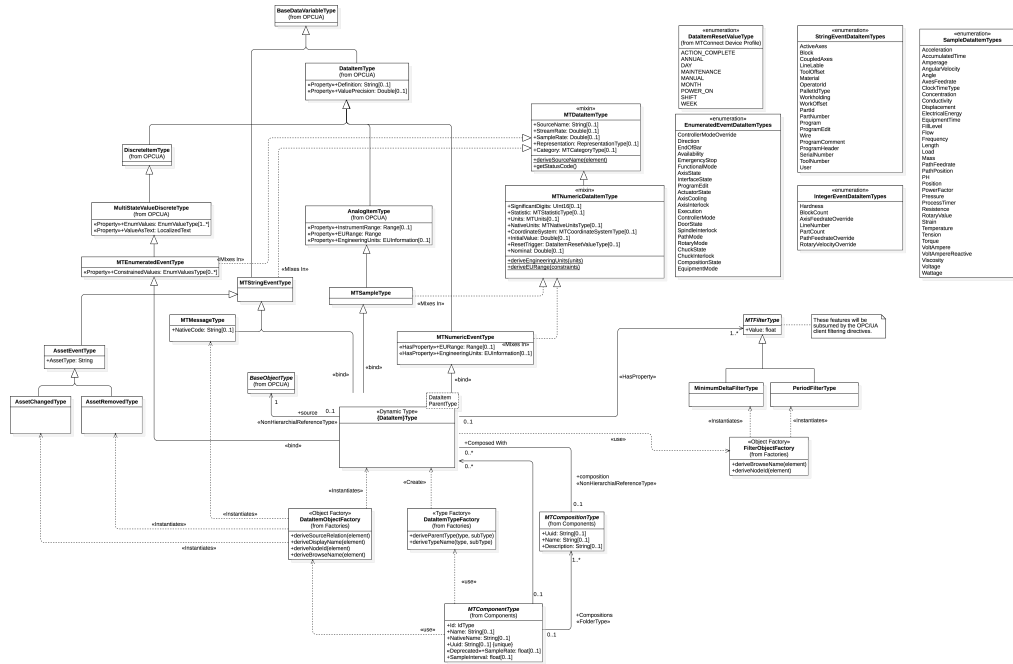


Figure 2: Data Items Diagram

2.2 Data Items

2.2.1 Defintion of AssetChangedType

145 Refer to Table 10 for detailed definition.

2.2.2 Defintion of AssetEventType

146 Refer to Table 11 for detailed definition.

Table 10: AssetChangedType Definition

Attribute	Value				
BrowseName	AssetChangedType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of AssetEventType (see section 2.2.2)					

Table 11: AssetEventType Definition

Attribute	Value				
BrowseName	AssetEventType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTStringEventType (see section 2.2.11)					
HasProperty	Variable	AssetType	String	PropertyType	Mandatory

2.2.3 Defintion of AssetRemovedType

147 Refer to Table 12 for detailed definition.

2.2.4 Defintion of MTDataItemType

148 The data item mixin will inject the properties and the methods into the related
 149 classes. This facility is similar to the Ruby module mixin or the Scala traits.

Table 12: AssetRemovedType Definition

Attribute	Value				
BrowseName	AssetRemovedType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of AssetEventType (see section 2.2.2)					

Table 13: MTDataItemType Definition

Attribute	Value				
BrowseName	MTDataItemType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
HasProperty	Variable	SourceName	String	PropertyType	Optional
HasProperty	Variable	StreamRate	Double	PropertyType	Optional
HasProperty	Variable	SampleRate	Double	PropertyType	Optional
HasProperty	Variable	Representation	RepresentationType	PropertyType	Optional
HasProperty	Variable	Category	MTCategoryType	PropertyType	Optional
HasProperty	Variable	<Dynamic>	MTFilterType	<Dynamic>	Optional
HasComponent	Object	source		BaseObjectType	Optional

Table 14: MTEnumeratedEventType Definition

Attribute	Value				
BrowseName	MTEnumeratedEventType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MultiStateValueDiscreteType (See OPCUA Documentation)					
HasProperty	Variable	ConstrainedValues	EnumValuesType	PropertyType	Mandatory

150 Refer to Table 13 for detailed definition.

2.2.5 Defintion of MTEnumeratedEventType

151 All Data Items with Category EVENT having a Controlled Vocabularies will be
 152 of this type. Otherwise, MTString

153 Refer to Table 14 for detailed definition.

Table 15: MTFilterType Definition

Attribute	Value				
BrowseName	MTFilterType				
IsAbstract	True				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
HasProperty	Variable	Value	float	PropertyType	Mandatory

Table 16: MTMessageType Definition

Attribute	Value				
BrowseName	MTMessageType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTStringEventType (see section 2.2.11)					
HasProperty	Variable	NativeCode	String	PropertyType	Optional

2.2.6 Defintion of MTFilterType

154 These features will be subsumed by the OPC/UA client filtering directives.

155 Refer to Table 15 for detailed definition.

2.2.7 Defintion of MTMessageType

156 Refer to Table 16 for detailed definition.

2.2.8 Defintion of MTNumericDataItemType

157 These are the additional attributes that are relevent to numeric data items. The
 158 factory will evaluate these values and will set the engineering units and the range
 159 associated with the parent entity.

Table 17: MTNumericDataItemType Definition

Attribute	Value				
BrowseName	MTNumericDataItemType				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	TypeDefinition	Modeling Rule
Subtype of MTDataItemType (see section 2.2.4)					
HasProperty	Variable	SignificantDigits	UInt16	PropertyType	Optional
HasProperty	Variable	Statistic	MTStatisticType	PropertyType	Optional
HasProperty	Variable	Units	MTUnits	PropertyType	Optional
HasProperty	Variable	NativeUnits	MTNativeUnitsType	PropertyType	Optional
HasProperty	Variable	CoordinateSystem	MTCoordinateSystemType	PropertyType	Optional
HasProperty	Variable	InitialValue	Double	PropertyType	Optional
HasProperty	Variable	ResetTrigger	DataItemResetValueType	PropertyType	Optional
HasProperty	Variable	Nominal	Double	PropertyType	Optional

160 Refer to Table 17 for detailed definition.

2.2.9 Defintion of MTNumericEventType

161 All data items with category EVENT and a numeric value.

162 Refer to Table 18 for detailed definition.

2.2.10 Defintion of MTSampleType

163 Data Items with category SAMPLE

164 Refer to Table 19 for detailed definition.

Table 18: MTNumericEventType Definition

Attribute	Value				
BrowseName	MTNumericEventType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of DataItemType (See OPCUA Documentation)					
HasProperty	Variable	EURange	Range	PropertyType	Optional
HasProperty	Variable	EngineeringUnits	EUInformation	PropertyType	Optional

Table 19: MTSampleType Definition

Attribute	Value				
BrowseName	MTSampleType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of AnalogItemType (See OPCUA Documentation)					

2.2.11 Defintion of MTStringEventType

- 165 All data items with category EVENT where the data is freeform text. The set_-
 166 data_type constraint derives makes the data type a string for this type.
- 167 Refer to Table 20 for detailed definition.

Table 20: MTStringEventType Definition

Attribute	Value				
BrowseName	MTStringEventType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of BaseDataVariableType (See OPCUA Documentation)					

Table 21: MinimumDeltaFilterType Definition

Attribute	Value				
BrowseName	MinimumDeltaFilterType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTFilterType (see section 2.2.6)					

Table 22: PeriodFilterType Definition

Attribute	Value				
BrowseName	PeriodFilterType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTFilterType (see section 2.2.6)					

2.2.12 Defintion of MinimumDeltaFilterType

168 Refer to Table 21 for detailed definition.

2.2.13 Defintion of PeriodFilterType

169 Refer to Table 22 for detailed definition.

2.2.14 Defintion of DataItemType

170 For each DataItem the Sub Type, and the Type will be composed to be the HasType-
 171 Definition relationship of the object. The BrowseName will also include the Com-
 172 position Type if a composition Id is provided.

173 Refer to Table 23 for detailed definition.

Table 23: DataItemType Definition

Attribute	Value				
BrowseName	DataItemType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of MTNumericEventType (see section 2.2.9)					

Table 24: MTExclusiveLimitConditionType Definition

Attribute	Value				
BrowseName	MTExclusiveLimitConditionType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ExclusiveLimitAlarmType (See OPCUA Documentation)					

2.3 Conditions

2.3.1 Defintion of MTExclusiveLimitConditionType

174 Refer to Table 24 for detailed definition.

2.3.2 Defintion of MTNonExclusiveConditionType

175 Refer to Table 25 for detailed definition.

2.3.3 Defintion of ConditionClassType

176 Refer to Table 26 for detailed definition.

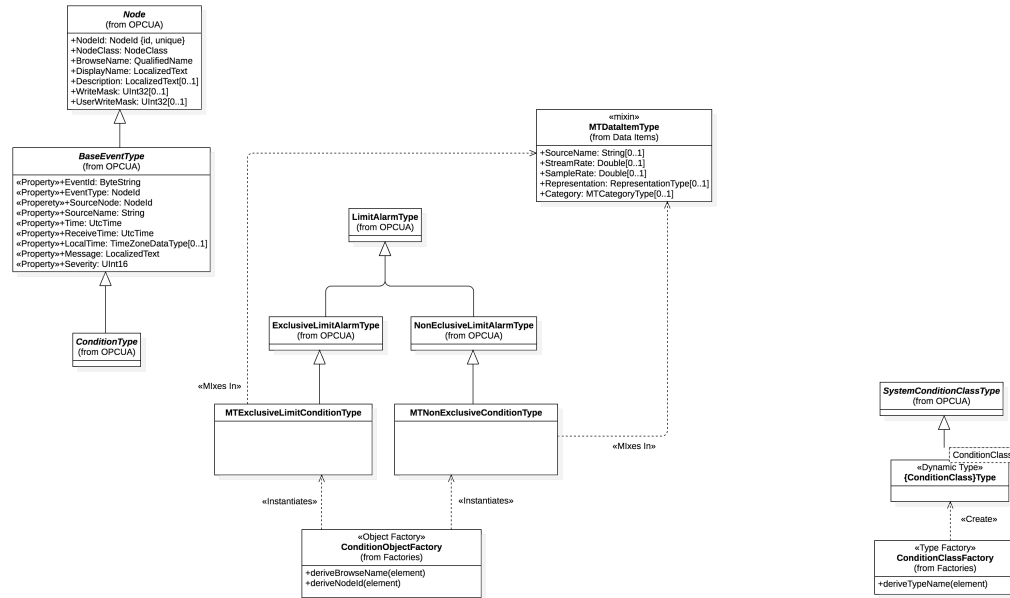


Figure 3: Conditions Diagram

Table 25: MTNonExclusiveConditionType Definition

Attribute	Value				
BrowseName	MTNonExclusiveConditionType				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	Type Definition	Modeling Rule
Subtype of NonExclusiveLimitAlarmType (See OPCUA Documentation)					

Table 26: ConditionClassType Definition

Attribute	Value				
BrowseName	ConditionClassType				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	TypeDefinition	Modeling Rule
Subtype of SystemConditionClassType (See OPCUA Documentation)					

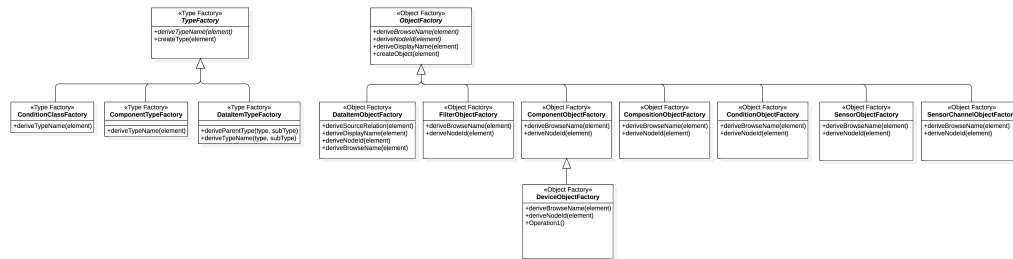


Figure 4: Factories Diagram

2.4 Factories

177 The factories are not part of the OPC/UA information model. They are a set
 178 of helper classes that are used to create dynamic types and objects. Since the
 179 MTConnect information model can be layered on top of the OPC/UA abstractions,
 180 the factories provide the rules for creating the browse and display names for each
 181 type.

182 The factories also create dynamic objects when required for variables of various
 183 classes when they are required, such as the Data Items and the Components. Some
 184 of the relationships are more complex since they require a dynamic super-type
 185 relationship that relies on the correct placement of the MTConnect elements to be
 186 correctly represented using the OPC/UA base types.

187 This is especially evident when mapping the DataItems and the Conditions to the
 188 MTConnect Information Models and providing sufficient definition to allow for
 189 unambiguous implementation.

2.4.1 Definition of ComponentObjectFactory

190 Refer to Table 27 for detailed definition.

Table 27: ComponentObjectFactory Definition

Attribute	Value				
BrowseName	ComponentObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ObjectFactory (see section 2.4.11)					

Table 28: ComponentTypeFactory Definition

Attribute	Value				
BrowseName	ComponentTypeFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of TypeFactory (see section 2.4.14)					

2.4.2 Defintion of ComponentTypeFactory

191 The ‘ComponentTypeFactory’ creates component types using the MTConnect
 192 XML element as an input. The factory takes the ‘QName’ (or qualified name)
 193 of the XML element and then appends ‘Type’. For example an ‘<Controller
 194 id=’...’></...>’ element will create an OPC/UA ‘ControllerType’ type definition
 195 as an extension of the base ‘MTControllerType’.

196 Currently there is no additional abstractions or super types required by the com-
 197 panion specification. The types will be a single level where each Component is a
 198 sub-type of the base ‘MTComponentType’.

199 Refer to Table 28 for detailed definition.

2.4.3 Defintion of CompositionObjectFactory

200 Refer to Table 29 for detailed definition.

Table 29: CompositionObjectFactory Definition

Attribute	Value				
BrowseName	CompositionObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ObjectFactory (see section 2.4.11)					

Table 30: CompositionTypeFactory Definition

Attribute	Value				
BrowseName	CompositionTypeFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

2.4.4 Defintion of CompositionTypeFactory

201 Refer to Table 30 for detailed definition.

2.4.5 Defintion of ConditionClassFactory

202 Refer to Table 31 for detailed definition.

Table 31: ConditionClassFactory Definition

Attribute	Value				
BrowseName	ConditionClassFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of TypeFactory (see section 2.4.14)					

Table 32: ConditionObjectFactory Definition

Attribute	Value				
BrowseName	ConditionObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ObjectFactory (see section 2.4.11)					

Table 33: DataItemObjectFactory Definition

Attribute	Value				
BrowseName	DataItemObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ObjectFactory (see section 2.4.11)					

2.4.6 Defintion of ConditionObjectFactory

203 Refer to Table 32 for detailed definition.

2.4.7 Defintion of DataItemObjectFactory

204 Refer to Table 33 for detailed definition.

2.4.8 Defintion of DataItemTypeFactory

205 Based on the data item category, type, and subType, this class creates a new
 206 OPC/UA type and also provides the template parameter for the ParentType from
 207 which this type is derived.

208 See the Data Item Type Factory.

209 Refer to Table 34 for detailed definition.

Table 34: DataItemTypeFactory Definition

Attribute	Value				
BrowseName	DataItemTypeFactory				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	TypeDefinition	Modeling Rule
Subtype of TypeFactory (see section 2.4.14)					

Table 35: DeviceObjectFactory Definition

Attribute	Value				
BrowseName	DeviceObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	TypeDefinition	Modeling Rule
Subtype of ComponentObjectFactory (see section 2.4.1)					

2.4.9 Defintion of DeviceObjectFactory

210 The model instantiation for MTConnect begins with the ‘Device‘ MTConnect
 211 element and then recursively traverses the sub-elements. The device will the ca-
 212 pabilities in the component factory to generate all the data items and component
 213 types.

214 Refer to Table 35 for detailed definition.

2.4.10 Defintion of FilterObjectFactory

215 Creates filters based on the type attribute of the Filter element.

216 Refer to Table 36 for detailed definition.

Table 36: FilterObjectFactory Definition

Attribute	Value				
BrowseName	FilterObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ObjectFactory (see section 2.4.11)					

Table 37: ObjectFactory Definition

Attribute	Value				
BrowseName	ObjectFactory				
IsAbstract	True				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

2.4.11 Defintion of ObjectFactory

217 Refer to Table 37 for detailed definition.

2.4.12 Defintion of SensorChannelObjectFactory

218 Refer to Table 38 for detailed definition.

Table 38: SensorChannelObjectFactory Definition

Attribute	Value				
BrowseName	SensorChannelObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ObjectFactory (see section 2.4.11)					

Table 39: SensorObjectFactory Definition

Attribute	Value				
BrowseName	SensorObjectFactory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of ObjectFactory (see section 2.4.11)					

Table 40: TypeFactory Definition

Attribute	Value				
BrowseName	TypeFactory				
IsAbstract	True				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

2.4.13 Defintion of SensorObjectFactory

219 Refer to Table 39 for detailed definition.

2.4.14 Defintion of TypeFactory

220 Refer to Table 40 for detailed definition.

2.5 MTConnect Device Profile

221 The device profile documents the common data types and stereotypes that are used
 222 to construct the model. A stereotype is a design or modeling pattern that provides
 223 additional information about the type or the relationship between types.

224 It can also identify the behavior of a property or the role the type or relation will
 225 play in the model.

226 Stereotypes are used throughout the model to provide additional information that

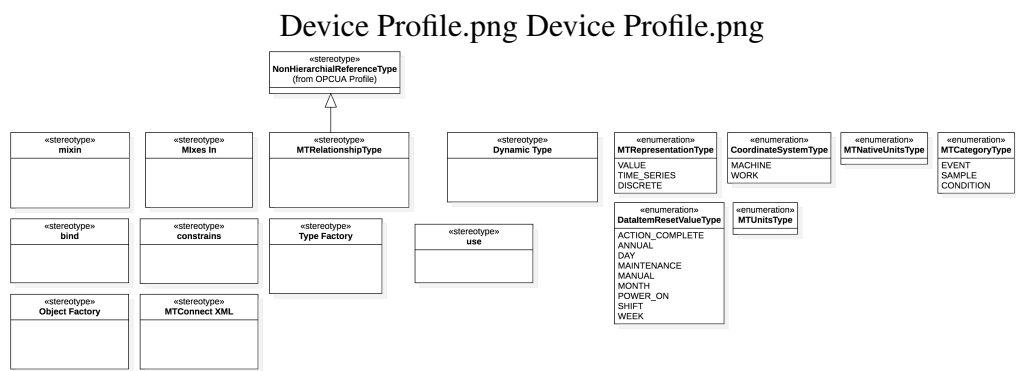


Figure 5: MTConnect Device Profile Diagram

Table 41: Dynamic Type Definition

Attribute	Value				
BrowseName	Dynamic Type				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

227 will help provide context and definition to aid in better understanding the data
228 model.

2.5.1 Defintion of Dynamic Type

229 Refer to Table 41 for detailed definition.

2.5.2 Defintion of Mixes In

230 Refer to Table 42 for detailed definition.

Table 42: Mlxes In Definition

Attribute	Value				
BrowseName	Mlxes In				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

Table 43: MTConnect XML Definition

Attribute	Value				
BrowseName	MTConnect XML				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

2.5.3 Defintion of MTConnect XML

231 Refer to Table 43 for detailed definition.

2.5.4 Defintion of MTRelationshipType

232 Refer to Table 44 for detailed definition.

Table 44: MTRelationshipType Definition

Attribute	Value				
BrowseName	MTRelationshipType				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule
Subtype of NonHierarchialReferenceType (See OPCUA Profile Documentation)					

Table 45: Object Factory Definition

Attribute	Value				
BrowseName	Object Factory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

Table 46: Type Factory Definition

Attribute	Value				
BrowseName	Type Factory				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

2.5.5 Defintion of Object Factory

233 Refer to Table 45 for detailed definition.

2.5.6 Defintion of Type Factory

234 Refer to Table 46 for detailed definition.

2.5.7 Defintion of bind

235 Refer to Table 47 for detailed definition.

2.5.8 Defintion of constrains

236 Refer to Table 48 for detailed definition.

Table 47: bind Definition

Attribute	Value				
BrowseName	bind				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

Table 48: constrains Definition

Attribute	Value				
BrowseName	constrains				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

2.5.9 Defintion of mixin

237 The contents properties and the behavior of the class are combined with another
 238 class.

239 Refer to Table 49 for detailed definition.

2.5.10 Defintion of use

240 The use stereotype indicates that one class uses as a helper to perform a specific
 241 operation or activity. This stereotype is mainly used to indicate that a specific
 242 factory is being employed by another type to create dynamic properties or rela-

Table 49: mixin Definition

Attribute	Value				
BrowseName	mixin				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

Table 50: use Definition

Attribute	Value				
BrowseName	use				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	Modeling Rule

243 tionships.

244 Refer to Table 50 for detailed definition.