



MTConnect[®] Standard

Part 4.0 – Assets

Version 1.3.0

Prepared for: MTConnect Institute
Prepared by: William Sobel
Prepared on: September 30, 2014

MTConnect[®] Specification and Materials

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

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

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

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

Table of Contents

1	Overview	1
1.1	MTConnect® Document Structure	1
2	Purpose of This Document	3
2.1	Terminology.....	3
2.2	Terminology and Conventions.....	5
3	Extension to Part 1, Overview and Protocol	6
4	Extensions to Part 2, Components and Data Items	7
4.1	Data Item Types for EVENT Category.....	7
5	Extensions to Part 3, Streams, Events, Samples, and Condition	7
5.1	Extension to Events section 3.9.....	7
5.1.1	<i>Additional AssetChanged attributes:</i>	8
5.1.2	<i>Additional AssetChanged attributes:</i>	8
6	Assets	9
	Appendices	10
A.	Bibliography	10

Table of Figures

Figure 1: Assets Schema..... 9

1 Overview

MTConnect[®] is a standard based on an open protocol for data integration. MTConnect[®] is not intended to replace the functionality of existing products, but it strives to enhance the data acquisition capabilities of devices and applications and move toward a plug-and-play environment to reduce the cost of integration.

MTConnect[®] is built upon the most prevalent standards in the manufacturing and software industry, maximizing the number of tools available for its implementation and providing the highest level of interoperability with other standards and tools in these industries.

To facilitate this level of interoperability, a number of objectives are being met. Foremost is the ability to transfer data via a standard protocol which includes:

- A device identity (i.e. model number, serial number, calibration data, etc.).
- The identity of all the independent components of the device.
- Possibly a device's design characteristics (i.e. axis length, maximum speeds, device thresholds, etc.).
- Most importantly, data captured in real or near-real-time (i.e. current speed, position data, temperature data, program block, etc.) by a device that can be utilized by other devices or applications (e.g. utilized by maintenance diagnostic systems, management production information systems, CAM products, etc.).

The types of data that may need to be addressed in MTConnect[®] could include:

- Physical and actual device design data
- Measurement or calibration data
- Near-real-time data from the device

To accommodate the vast amount of different types of devices and information that may come into play, MTConnect[®] will provide a common high-level vocabulary and structure.

The first version of MTConnect[®] will focus on a limited set of the characteristics mentioned above that were selected based on the fact that they can have an immediate affect on the efficiency of operations.

1.1 MTConnect[®] Document Structure

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

- Part 1: Overview and Protocol
- Part 2: Components and Data Items
- Part 3: Streams, Events, Samples, and Condition
- Part 4.0: Assets
 - Part 4.1: Cutting Tools

These four documents are considered the bases of the MTConnect standard. Information applicable to basic machine and device types will be included in these documents. Additional parts to the standard will be added to provide information and extensions to the standard focused on specific devices, components, or technologies considered requiring separate emphasis. All

42 information specific to the topic of each additional part **MUST** be included within that document
43 even when it is a subject matter of one of the base parts of the standard.

44

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

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

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

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

50 2 Purpose of This Document

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

- 52
- 53 • define the MTConnect[®] standard;
 - 54 • specify the requirements for compliance with the MTConnect[®] standard;
 - 55 • provide engineers with sufficient information to implement *Agents* for their devices;
 - 56 • provide developers with the necessary guidelines to use the standard to develop applications.

57 Part 1 of the MTConnect Standard provides an overview of the MTConnect Architecture and
58 Protocol; including communication, fault tolerance, connectivity, and error handling require-
59 ments.

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

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

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

73 2.1 Terminology

74	Adapter	An optional software component that connects the Agent to the Device.
75	Agent	A process that implements the MTConnect [®] HTTP protocol, XML generation, 76 and MTConnect protocol.
77	Alarm	An alarm indicates an event that requires attention and indicates a deviation 78 from normal operation. Alarms are reported in MTConnect as <code>Condition</code> .
79	Application	A process or set of processes that access the MTConnect [®] <i>Agent</i> to perform 80 some task.
81	Attribute	A part of an XML element that provides additional information about that 82 XML element. For example, the name XML element of the <code>Device</code> is given 83 as <code><Device name="mill-1">...</Device></code>
84	CDATA	The text in a simple content element. For example, <i>This is some text</i> , 85 in <code><Message ...>This is some text</Message></code> .

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

122	Sample	A sample is a data point from within a continuous series of data points. An
123		example of a Sample is the position of an axis.
124	Socket	When used concerning inter-process communication, it refers to a connection
125		between two end-points (usually processes). Socket communication most
126		often uses TCP/IP as the underlying protocol.
127	Stream	A collection of <code>Events</code> , <code>Samples</code> , and <code>Condition</code> organized by
128		devices and components.
129	Service	An application that provides necessary functionality.
130	Tag	Used to reference an instance of an XML element.
131	TCP/IP	TCP/IP is the most prevalent stream-based protocol for inter-process
132		communication. It is based on the IP stack (Internet Protocol) and provides
133		the flow-control and reliable transmission layer on top of the IP routing
134		infrastructure.
135	URI	Universal Resource Identifier. This is the official name for a web address as
136		seen in the address bar of a browser.
137	UUID	Universally unique identifier.
138	XPath	XPath is a language for addressing parts of an XML Document. See the
139		XPath specification for more information. http://www.w3.org/TR/xpath
140	XML	Extensible Markup Language. http://www.w3.org/XML/
141	XML Schema	The definition of the XML structure and vocabularies used in the XML
142		Document.
143	XML Document	An instance of an XML Schema which has a single root XML element and
144		conforms to the XML specification and schema.
145	XML Element	An element is the central building block of any XML Document. For
146		example, in MTConnect [®] the Device XML element is specified as <code><Device</code>
147		<code>> . . . </Device></code>
148	XML NMTOKEN	The data type for XML identifiers. It MUST start with a letter, an underscore
149		“_” or a colon “:” and then it MUST be followed by a letter, a number, or one
150		of the following “.”, “-”, “_”, “:”. An NMTOKEN cannot have any spaces or
151		special characters.

152 2.2 Terminology and Conventions

153 Please refer to Part 1 “Overview and Protocol” Section 2 for XML Terminology and
154 Documentation conventions.

155 3 Extension to Part 1, Overview and Protocol

156 As documented in Part 1, additional queries will be added to the *Agent* to support the storage and
 157 retrieval of assets. There is more detail in Part 1; what follows is a summary of the protocol
 158 additions:

159 Asset protocol:

- 160 • Request an asset by id:
 - 161 ○ url: `http://example.com/asset/hh1`
 - 162 ○ Returns the `MTConnectAssets` document for asset `hh1`
- 163 • Request multiple assets by id:
 - 164 ○ url: `http://example.com/asset/hh1;cc;123;g5`
 - 165 ○ Returns the `MTConnectAssets` document for asset `hh1`, `cc`, `123`, and `g5`.
- 166 • Request for all the assets in the *Agent*:
 - 167 ○ url: `http://example.com/assets`
 - 168 ○ Returns all available `MTConnect` assets in the *Agent*. `MTConnect` **MAY** return a
 169 limited set if there are too many asset records. The assets **MUST** be added to the
 170 beginning with the most recently modified assets.
- 171 • Request for all assets of a given type in the *Agent*:
 - 172 ○ url: `http://example.com/assets?type=CuttingTool`
 - 173 ○ Returns all available `CuttingTool` assets from the *MTConnect Agent*.
 174 `MTConnect` **MAY** return a limited set if there are too many asset records. The
 175 assets **MUST** be added to the beginning with the most recently modified assets.
- 176 • Request for all assets of a given type in the *Agent* up to a maximum count:
 - 177 ○ url:
 178 `http://example.com/assets?type=CuttingTool&count=1000`
 - 179 ○ Returns all available `CuttingTool` assets from the *MTConnect Agent*.
 180 `MTConnect` **MUST** return up to 1000 assets beginning with the most recently
 181 modified assets if they exist.
- 182 • Request for all assets including assets that have been removed:
 - 183 ○ url:
 184 <http://example.com/assets?type=CuttingTool&removed=true>
 185 [e](http://example.com/assets?type=CuttingTool&removed=true)
 - 186 ○ Returns all available `CuttingTool` assets from the *MTConnect Agent*. With
 187 the removed flag, assets that have been removed but are included in the result set.

188 4 Extensions to Part 2, Components and Data Items

189 This document will add the following data item types to support change notification when an
 190 asset is added or updated. The data item **MUST** be placed in the `DataItems` collection of the
 191 top level device. The device **MUST** be the device that is supplying the asset data.

192 4.1 Data Item Types for EVENT Category

Data Item type/subtype	Description
<code>ASSET_CHANGED</code>	The value of the CDATA for the event MUST be the <code>assetId</code> of the asset that has been added or changed. There will not be a separate message for new assets.
<code>ASSET_REMOVED</code>	The value of the CDATA for the event MUST be the <code>assetId</code> of the asset that has been removed. The asset will still be visible if requested with the <code>includeRemoved</code> parameter as described in the protocol section. When assets are removed they are not moved to the beginning of the most recently modified list.

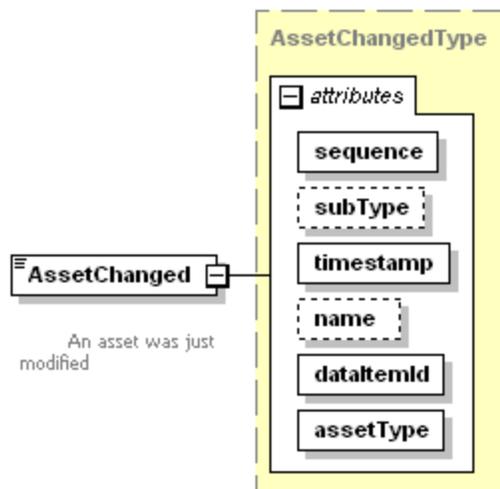
193 5 Extensions to Part 3, Streams, Events, Samples, and 194 Condition

195 The associated modifications **MUST** be added to Part 3 to add the following event to the events
 196 in the streams.

197 5.1 Extension to Events section 3.9

198 The `AssetChanged` element extends the base `Event` type defined in *Part 3, Streams, Events,*
 199 *Samples, and Condition* and adds the `assetType` attribute to the base `Event`. This new event
 200 will signal whenever a new asset is added or the existing definition of an asset is updated. The
 201 `assetId` is provide as the `CDATA` value and can be used to request the asset data from the *Agent*
 202 as described in *Part 1, Overview and Protocol*.

203 The `AssetRemoved` element extends the base `Event` type defined in *Part 3, Streams, Events,*
 204 *Samples, and Condition* and adds the `assetType` attribute to the base `Event`. This new event
 205 will signal whenever an asset has been removed from the agent. The asset will still be available if
 206 requested if the `removed=true` argument is supplied. The `assetId` is provide as the `CDATA` value
 207 and can be used to request the asset data from the *Agent* as described in *Part 1, Overview and*
 208 *Protocol*.



209

Generated by XMLSpy

www.altova.com

210 **AssetChanged** An asset has been added or modified. The **CDATA** for the AssetChanged
 211 element **MUST** be the `assetId` of the asset that has been modified.

212 **5.1.1 Additional AssetChanged attributes:**

Attribute	Description	Occurrence
assetType	The type of asset that changed	1

213

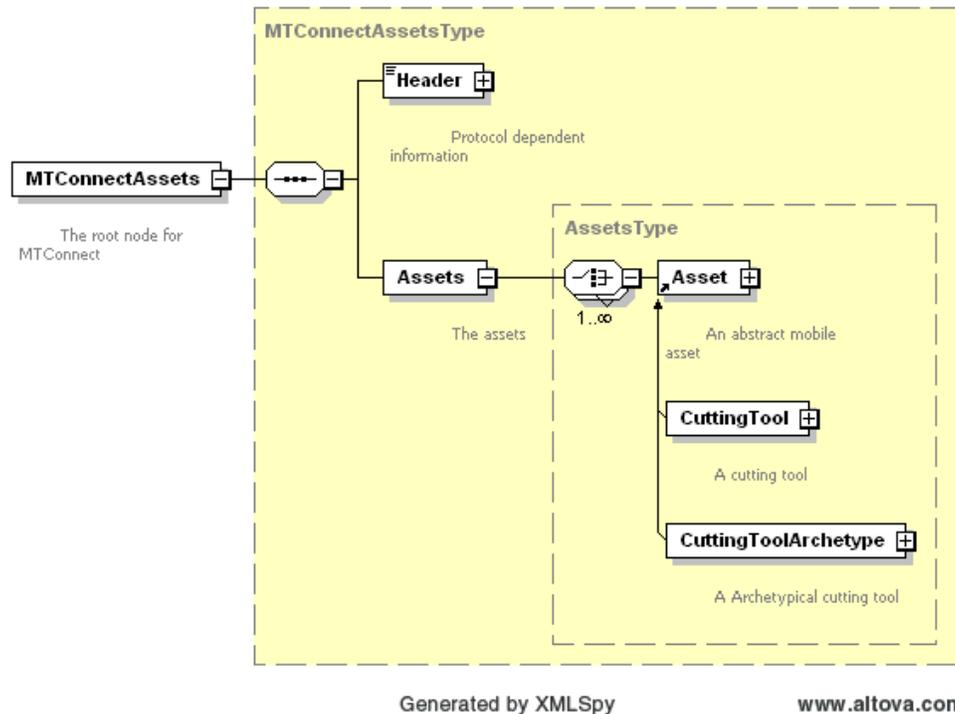
214 *Add diagram..*

215 **AssetRemoved** An asset has been removed. The **CDATA** for the AssetRemoved element
 216 **MUST** be the `assetId` of the asset that has been removed.

217 **5.1.2 Additional AssetChanged attributes:**

Attribute	Description	Occurrence
assetType	The type of asset that changed	1

218

219 **6 Assets**

220

Generated by XMLSpy

www.altova.com

221

Figure 1: Assets Schema

222 An Asset is something that is associated with the manufacturing process that is not a component
 223 of a device, can be removed without detriment to the function of the device, and can be
 224 associated with other devices during their lifecycle. An asset does not have computational
 225 capabilities, but may carry information in some media physically attached to the asset.

226 Concrete examples of Assets are things like Cutting Tools, Workholding Systems, and Fixtures.
 227 Part 4 of the MTConnect standard will concern itself with the modeling of these assets and the
 228 management and communication of asset data using MTConnect.

229 At the top level of the MTConnectAssets document we have a standard header as documented in
 230 Part 1: Overview and Protocol and one or more assets. Each asset is required to have an assetId
 231 that serves as a unique identifier of that asset. The id allows the application to request the asset
 232 data from the agent, as prescribed in Part 1.

233 In the remaining document, we will be discussing Cutting Tools as the first asset type covered by
 234 the standard. The cutting tool must have an assetId that differs from all the other assets tracked
 235 by this agent. There **MUST** never be more than one asset provided by MTConnect with the same
 236 asset Id in the same agent.

237

Appendices

A. Bibliography

- 239 1. Engineering Industries Association. *EIA Standard - EIA-274-D*, Interchangeable Variable,
240 Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically
241 Controlled Machines. Washington, D.C. 1979.
- 242 2. ISO TC 184/SC4/WG3 N1089. *ISO/DIS 10303-238*: Industrial automation systems and
243 integration Product data representation and exchange Part 238: Application Protocols:
244 Application interpreted model for computerized numerical controllers. Geneva,
245 Switzerland, 2004.
- 246 3. International Organization for Standardization. *ISO 14649*: Industrial automation systems
247 and integration – Physical device control – Data model for computerized numerical
248 controllers – Part 10: General process data. Geneva, Switzerland, 2004.
- 249 4. International Organization for Standardization. *ISO 14649*: Industrial automation systems
250 and integration – Physical device control – Data model for computerized numerical
251 controllers – Part 11: Process data for milling. Geneva, Switzerland, 2000.
- 252 5. International Organization for Standardization. *ISO 6983/1* – Numerical Control of
253 machines – Program format and definition of address words – Part 1: Data format for
254 positioning, line and contouring control systems. Geneva, Switzerland, 1982.
- 255 6. Electronic Industries Association. *ANSI/EIA-494-B-1992*, 32 Bit Binary CL (BCL) and 7
256 Bit ASCII CL (ACL) Exchange Input Format for Numerically Controlled Machines.
257 Washington, D.C. 1992.
- 258 7. National Aerospace Standard. *Uniform Cutting Tests* - NAS Series: Metal Cutting
259 Equipment Specifications. Washington, D.C. 1969.
- 260 8. International Organization for Standardization. *ISO 10303-11*: 1994, Industrial
261 automation systems and integration Product data representation and exchange Part 11:
262 Description methods: The EXPRESS language reference manual. Geneva, Switzerland,
263 1994.
- 264 9. International Organization for Standardization. *ISO 10303-21*: 1996, Industrial
265 automation systems and integration -- Product data representation and exchange -- Part
266 21: Implementation methods: Clear text encoding of the exchange structure. Geneva,
267 Switzerland, 1996.
- 268 10. H.L. Horton, F.D. Jones, and E. Oberg. *Machinery's handbook*. Industrial Press, Inc. New
269 York, 1984.
- 270 11. International Organization for Standardization. *ISO 841-2001: Industrial automation*
271 *systems and integration - Numerical control of machines - Coordinate systems and*
272 *motion nomenclature*. Geneva, Switzerland, 2001.

- 273 12. *ASME B5.59-2 Version 9c: Data Specification for Properties of Machine Tools for*
274 *Milling and Turning. 2005.*
- 275 13. *ASME/ANSI B5.54: Methods for Performance Evaluation of Computer Numerically*
276 *Controlled Lathes and Turning Centers. 2005.*
- 277 14. OPC Foundation. *OPC Unified Architecture Specification, Part 1: Concepts Version 1.00.*
278 *July 28, 2006.*
- 279 15. International Organization for Standardization. *ISO 13399: Cutting tool data*
280 *representation and exchange. Geneva, Switzerland, 2000.*
- 281