Getting Started with MTConnect

Connectivity Guide

MTConnect® Specifications or Materials

Table of Contents

SECTION I – INTRODUCTION

Background

A major challenge to the wide adoption of the MTConnect® standard on the shop floor is the lack of a clear understanding of what is required to implement the standard and how to connect applications to the machine tools in existence on the shop floor today. AMT has estimated there are over 1.2 million machine tools installed in the US that could be enabled with MTConnect. Industry experts stated in 2010 that only four to five percent of all machine tools worldwide were connected to a data collection system. The need for connectivity to existing machines, in addition to new machines, was recognized from the beginning of the development of the standard. To address this issue, a proposal was made to the MTConnect Technical Advisory Group during the International Manufacturing Technology Show (IMTS) 2010 for the creation of a working group to address this issue. This committee was named the Shop Floor Connectivity Working Group and consisted of members from the following organizations:

Memex Automation, Inc. (Co-Chair)

FA Consulting & Technology, LLC (Co-Chair)

Advanced Technologies Services, Inc.

FANUC FA America

General Dynamics – OTS

Georgia Tech Factory Information Systems Lab

Nexas Networks Inc.

Pinaka Systems, Inc.

Predator Software, Inc.

Real Time Development Corp.

Remmele Engineering, Inc.

SCADAware, Inc.

Southern Manufacturing Technologies, Inc.

System Insights, Inc.

TechSolve, Inc.

Virtual Photons Electrons, LLC

Objective

The efforts of this working group are contained in this document outlining key considerations for the implementation of the MTConnect standard to a wide variety of manufacturing equipment (e.g. machine tools) representing many different generations of technology.

This guide does not address requirements for implementing software applications that use data conforming to the MTConnect standard. It does, however, discuss a number of issues to be considered in planning your project and for preparation for discussions with third-party solution providers who implement solutions compliant to the MTConnect standard.

No vendor specific hardware or software solutions will be highlighted in this Connectivity Guide - only guidelines that should be evaluated by anyone considering the implementation of MTConnect on their production floor have been described.

Note: For the balance of this guide, the terms “Device” and “Devices” will be used to describe any piece, or multiple pieces, of equipment on the shop floor. Device(s) may be a machine tool or other piece(s) of process equipment.

Who Will Find Benefit from this Paper

This paper is intended to assist those individuals or companies who are interested in implementing shop floor software information systems and tools incorporating the use of the MTConnect standard. This paper specifically focuses on information for those who are considering adopting the MTConnect standard for both new and existing machine tools. It provides information to help in understanding the capabilities and limitations of your equipment and a variety of technology requirements that must be addressed by the decision makers, as well as the support personnel, who will be responsible for the implementation of the software information systems and tools on your shop floor. From the business side, this will include the owners and top management of the business as well as the operations and engineering management. From the implementation and support side, this normally includes the organization’s IT staff, Manufacturing Engineering/Manufacturing staff and shop floor personnel who will be supporting the implementation. It is the objective of this document to provide information to all levels of the organization that need a clear understanding of the benefits and technology limitations of integrating the MTConnect standard to your equipment.

SECTION II – MTConnect Overview

Why MTConnect

MTConnect is a universal factory floor communications protocol that many experts and business owners have agreed is necessary. Those considering implementation may need to know more about the implementation strategies before adopting this standard into their operations.

MTConnect is designed specifically for the shop floor environment. While there are numerous communication solutions available for the shop floor, MTConnect offers one very distinct difference. MTConnect is the first standard to define a "dictionary" for manufacturing data. This means that data from multiple machines will have a common definition – name, units, values, and context.

Traditionally when installing software applications that use shop floor data, the description of the data (name, scaling, meaning, etc.) is defined at the point of use – the application. This means that if you use the same data for more than one application, you must repeat the efforts to define your data for each application. This increases installation costs and time, increases your long term costs to maintain each application, and introduces a level of complexity that leads to errors and system failures.

With MTConnect, the data is defined only once at the MTConnect compliant interface to the device or machine tool. Once the data is defined based on the MTConnect standard protocol, it can then easily be used by all MTConnect compliant software applications. This eliminates the need to redefine the data within each application.

This fundamental difference significantly reduces startup time, overall project costs, and long term maintenance of software system interfaces. MTConnect compliant devices process information locally and then provide that data in a consistent format to any application - ERP, MES, Production Management Systems, Maintenance Systems, browsers, spreadsheets, and countless other applications. This approach leads to a plug-and-play atmosphere that mimics the PC computer arena.

Basics of MTConnect

MTConnect is an Open, Extensible, and Royalty Free standard.

As an open standard, MTConnect is based on well defined, commercially available technologies. The MTConnect protocol is based on standard Internet technologies – HTTP and XML (Extensible Mark-Up Language) – the underlying language of most web sites).

Extensibility is a key feature of the MTConnect standard. The MTConnect Institute clearly understands a standard cannot address every conceivable data type needed on the shop floor. MTConnect provides a clearly defined method for adding new data types which can be exchanged between machines and applications; providing the flexibility to meet the demands of any environment. The most common extensions implemented by industry will be considered for additions to future releases of the MTConnect Standard – continually improving and expanding the standard.

As a royalty free standard, no one is required to pay a licensing fee to the MTConnect Institute for use of the standard. Companies who agree to the MTConnect licensing agreement can integrate MTConnect with their products. Costs associated with products or services related to MTConnect apply to the products or services themselves – not an additional license fee for the use of the MTConnect standard. Additionally, from an MTConnect Institute perspective, there is no cost for a company to implement the MTConnect standard for their own equipment. This does not mean that companies cannot charge for their implementation of MTConnect. That is entirely up to those companies.

A system implemented using the MTConnect standard is made up of five fundamental components (see Figure 1 below).

Device – commonly a machine tool, but can be any piece of equipment or data source.

Adapter – An optional piece of software (and sometimes hardware) that provides a link or conversion from the data source and proprietary data definition in the device to the MTConnect Data definition. This can be thought of as a translator. The Adapter is not needed for devices that use MTConnect as their native language.

Agent – A piece of software that collects, arranges, and stores data from the device or adapter. It receives requests for data from applications, processes those requests, and then transmits the required data. The function of the Agent, and its associated data dictionary, is defined by the MTConnect Standard. It is the only part of the communication system that is specifically defined by the standard. The other components (as illustrated in Figure 1 below) represent your equipment, your plant or shop communications network, and your software application.

Network – The physical connection between a data source (device) and the data consumer (application). Typically, this is an Ethernet network. The communication on the network normally uses standard network communications methods – HTTP protocol. It should be noted that the MTConnect structure is adaptable and can be implemented in conjunction with other networking solutions other than Ethernet and Internet protocols.

Application (Client) – The application is the actual requestor and consumer of MTConnect data. Typical functions of the application are to request, store, manipulate and display data. The Application includes a function called the Client which initiates all requests for MTConnect data. The Client is a software function in the application that actually requests data from the Agent and translates that data into the format required for the application.

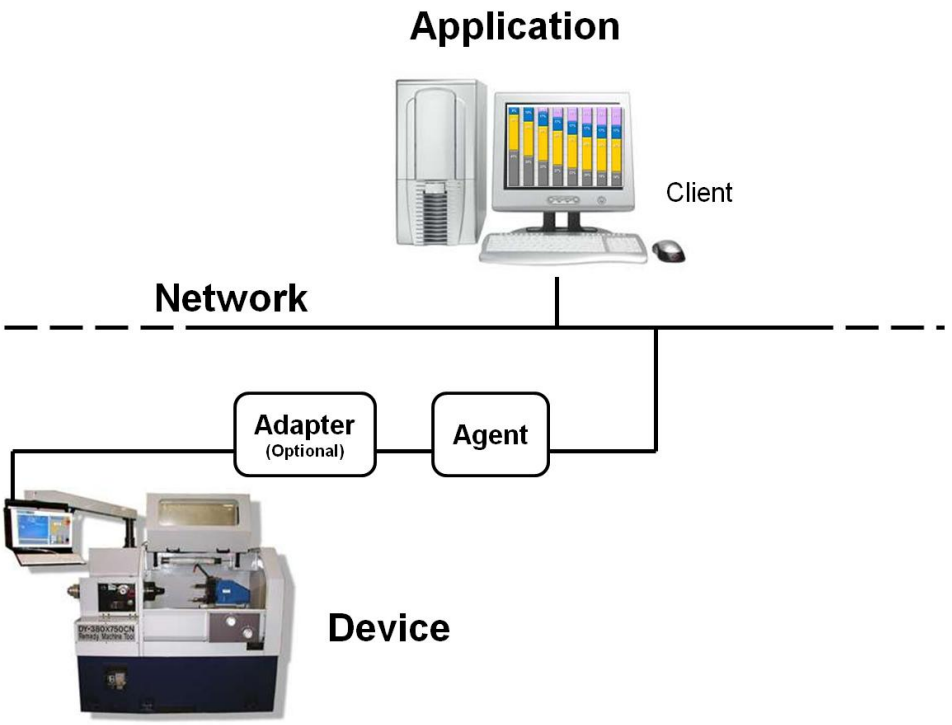

Figure 1 – Overview

The MTConnect Standard does not restrict the physical implementation of how a system using the MTConnect standard may be implemented.

- The Network may be a physical implementation like an Ethernet network. It can also be implemented using wireless or other technologies. (MTConnect is network neutral).

- Using the Network Protocol (HTTP) does not mean that your machine is automatically connected outside of your plant to the Internet. This is a communications method only. Protection of your data is controlled by your network security standards, such as firewalls or other popular security methods. If you should choose to make your machine data available over the Internet (maybe for remote connection to your machine builder's service department), it is much easier to do since the underlying communications standard is based on the Internet protocol

- The MTConnect Standard does not specify where the Adapter and Agent functions are located. They typically are located at the device. However, they can be placed anywhere in the networking architecture on, for example, a separate computer. Also, they do not need to be located together. It is acceptable to have the Adapter installed at the device and the Agent installed along with the Client. The location of these functions should be considered when implementing MTConnect since they will impact the level of data flow on different segments of your network.

Connecting a Device to a Network

While it may be straight forward to integrate a new machine tool that natively speaks MTConnect into your plant, how do you connect equipment that does not natively speak MTConnect into your network? Connection of the 1.2 million machine tools already installed in production shops in the US (many more than that internationally) is the major premise for this Connectivity Guide.

There are a variety of factors that impact the viability of communicating data from either new or older devices to an application. These factors are primarily defined by the capabilities of your equipment and the requirements of your software application(s). Understanding the capabilities of your existing machine to communicate data is a good starting point for your assessment. This paper cannot address every machine and technology in your plant. However, you can make an initial assessment of your equipment since most machines will naturally fall into one of three general groups – MTConnect Native Devices, MTConnect Translation Dependant Devices, or MTConnect Data Connection Dependant Devices. All machines can provide data using one of the methods described below.

MTConnect Native Devices (Figure 2) are devices (machines or equipment) that are provided with MTConnect functions and the data dictionary integrated into the device and it is capable to be connected to a network and MTConnect compliant software applications without the addition of other components. No additional functions must be added to enable communications. Basically, the Device naturally communicates using the MTConnect Standard.

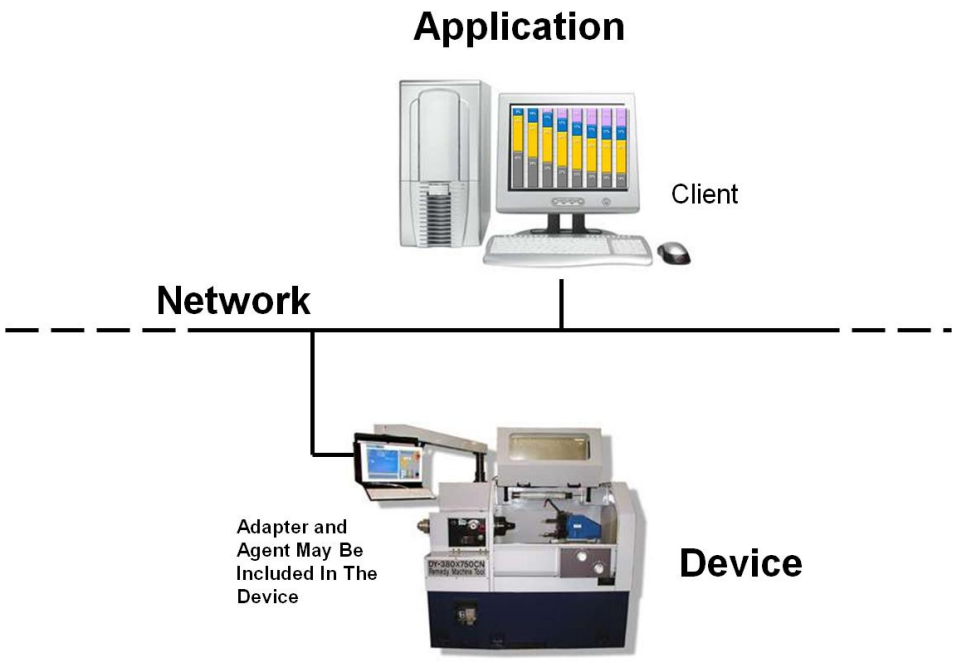

Figure 2 – MTConnect Native Device

MTConnect Translation Dependent Devices (Figure 3) are devices (machines or equipment) that require a separate function, or Translation Unit, (software and/or hardware) to translate data from the native language of the device into MTConnect compliant data. Data is available to a software application from the device but it needs to be translated into the MTConnect standard dictionary names via an Adapter. The Translation Unit, as defined for MTConnect Translation Dependent Devices, performs the function of linking two different communication systems so that they can exchange information with each other. This translation may include both physical changes (e.g. serial communications converted to Ethernet) and logical changes to translate data from the native language of the device to MTConnect compliant data.

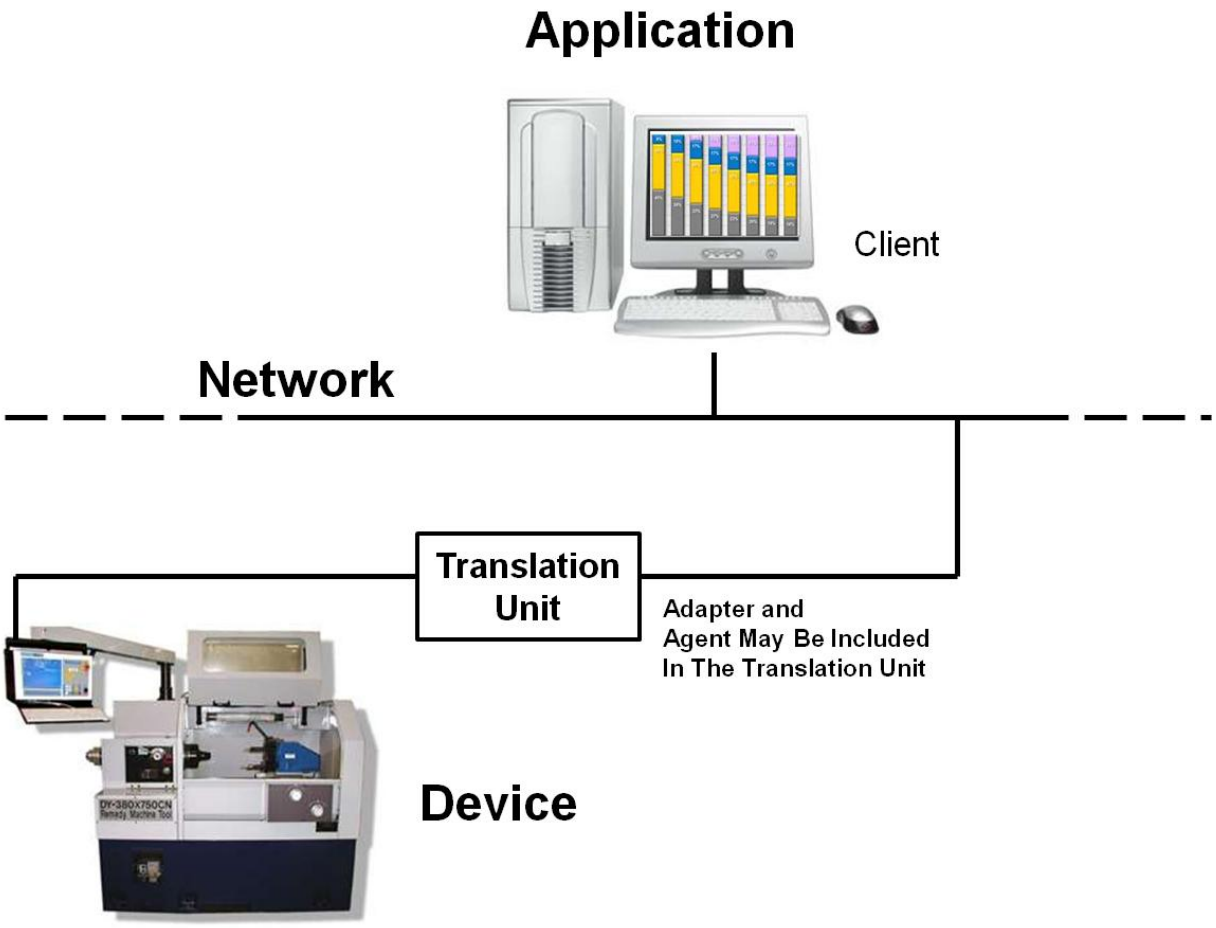

Figure 3 – MTConnect Translation Dependent Device

MTConnect Data Connection Dependent Devices (Figure 4) are devices (machines or equipment) that do not normally provide the data required by a software application through any standard connection method. Many of these devices do not naturally publish any data or the data may be limited. Machines in this category require a separate Connection Unit (software and/or hardware) to collect data from the device. The Connection Unit also includes the functionality of a Translation Unit to convert that data into MTConnect compliant data.

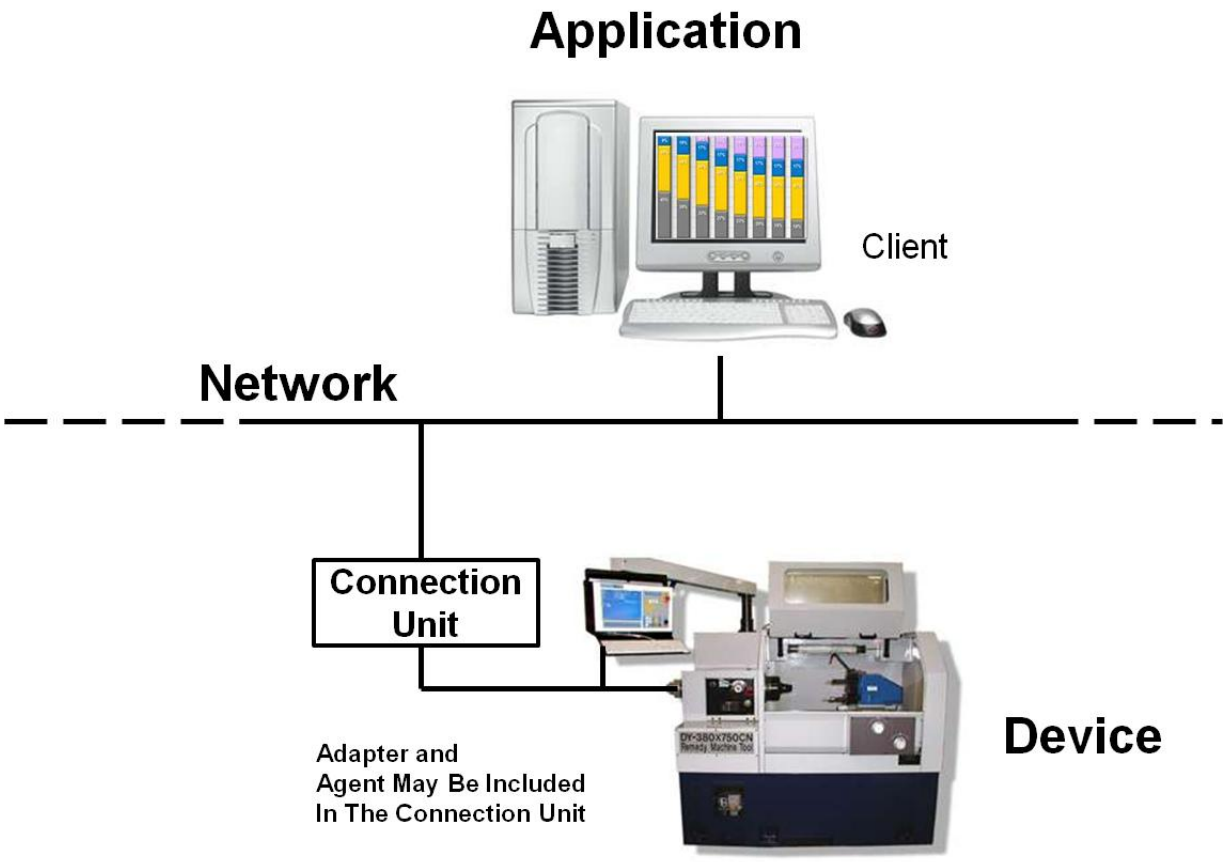

Figure 4 – MTConnect Connection Dependent Device

SECTION III – What Problems Do Companies Solve With MTConnect?

There are essentially limitless possibilities of how companies can use data from their shop floor to improve operations, track production, and justify decisions that impact shop operations. Some of the more common uses for shop floor data include:

Production Dashboard or Monitor:

A dashboard is an overview display showing the current state of each machine on the shop floor. It provides a high level view of the production facility and identifies problems on the shop floor that may require attention. Typically the dashboard is provided as a display on a computer on the shop floor or in a production office. Machine state information can include differing levels of detail but normally is restricted to Running, Idle, and Failed (Green/Yellow/Red). Many software systems also provide a "drill down" feature to access additional details on a specific machine through secondary screens dedicated to that machine.

Alerts:

Alerts are notifications of changes on the shop floor that require the attention of management, maintenance, etc. Alerts provide immediate identification of issues associated with the machines and related processes. These can include actual problems or pending problems that require attention before a process interruption occurs. Examples include: 1) Machine Down due to a specific fault or alarm; 2) Parts Running Low – Additional parts required; 3) Filters becoming clogged – Maintenance required; etc. Alerts typically are provided by alarm displays on a computer, email or text notices, or annunciation on a marquee or dashboard.

Equipment Availability and Usage:

Analysis of each machine showing how effectively the machine is being used. Multiple machine states are typically pre-defined based on machine and process requirements. The machines are monitored by the data collection system and the time that a machine is in each state is accumulated for reporting. Reports typically allow the data to be displayed based on multiple parameters – Daily, Weekly, Monthly, Shift, Operation, Part, etc. Report formats vary, but typically are represented either in table format, bar charts, or line charts. Measuring Machine Utilization is an effective parameter to evaluate the overall effectiveness of projects or operational changes aimed at improving the productivity on the shop floor.

Machine Downtime Analysis:

From the machine utilization information described above, each non-productive machine state can be further analyzed to determine the causes for lost production time. The Downtime Analysis further segregates Planned and Unplanned Downtime. The Unplanned Downtime can then be segregated into specific causes for production losses. The identification of the causes for production losses can be used to define projects to eliminate or reduce these losses.

Overall Equipment Effectiveness:

Overall Equipment Effectiveness (OEE) combines machine utilization information with quality measurements to provide a tool to measure the end-to-end effectiveness of any process or machining operation. OEE is used to measure how a production operation changes with time, determine the impact of changes made on the shop floor, and to compare the effectiveness of different processes or machines.

Production Reporting/Tracking:

Production data can be gathered directly from the shop floor. Software applications can provide tools to monitor production levels, manage product flow through the production process, track inventory and raw material queues, and lots of others. Production data is typically provided in both "near real-time" and periodic reports that can be displayed based on multiple parameters – Daily, Weekly, Monthly, Shift, Operation, Part, etc. Report formats vary, but typically are represented either in table format, bar charts, or line charts. Drill-down capabilities provide detailed analysis of production data based on individual machines, operators, work orders, shifts, parts, and production lots.

Maintenance Tracking/Planning:

Maintenance issues typically fall into two categories – Machine/Process Faults and Preventative Maintenance. Software information systems can provide significant benefits to both categories. Using Alerts, maintenance teams can be advised immediately of problems on the shop floor. Many times early identification of a problem or change on the shop floor can avoid more significant problems before they develop. Preventative Maintenance plans are often based strictly on calendar time (monthly, quarterly, or annual) with no consideration of actual production status or machine usage. Measuring operating times or number of operations directly from equipment can provide information to more effectively manage preventative maintenance schedules.

These examples of software applications using shop floor data are representative of some of the technologies available from system integrators and software vendors. The possible uses of this type of data are nearly limitless. The primary reason this is true is the combination of the nature of data and the fact that the data is easy to use based on how it is formatted by MTConnect. Information on services provided by integration and software providers is addressed in more detail below.

SECTION IV – How to Begin

Whether a company may be considering implementing an MTConnect solution or may be evaluating a third-party system integrator to implement an MTConnect solution, there are some basic steps that can be followed to aide in preparation for your project.

Appendix B includes a Worksheet that can guide you through the preparation process.

Step #1 – Define the Problem That You Are Trying to Solve

The most important task for establishing a solid foundation for your project is to clearly define the manufacturing or business problem that you are trying to solve. Specifics are most important. A clear definition provides a framework for assessing alternatives and driving to specific solutions. It will also help in your discussions with a third-party integrator or implementer – helping you select the right integrator/implementer and shortening the time for that supplier to define a solution to address your needs.

Often problem definitions do not clearly define the real issues to be addressed. Asking "Why do I want to know this" is a helpful tool to lead to the underlying issue that needs to be addressed (this question may need to be asked multiple times to reach the details associated with the root cause issue). As an example, it is common to state the objective for a project to be "I need to understand what is happening with the equipment on my shop floor". While this is true, it is not overly useful for defining a technical solution to address the underlying problem.

Asking the "Why do I want to know this" question will typically lead to a more precise statement of the problem. In the example above, it would typically lead to "I am not getting the production levels that I expect from my machines." Asking the "Why" question again would lead to "I have too much downtime on my machines and I don't know why." Going deeper yet will lead to "I need to know the causes of downtime on my machines and how much time is being lost for each of the causes." Now you have a very specific problem to address. Selection of a software data collection solution to address this problem can be easily identified and a targeted solution implemented.

The Worksheet in Appendix B will help lead you through this definition process.

Step #2 – What Does Success Look Like?

It is important that the project team and any 3rd party implementers/integrators have a clear understanding of the end goal for the project. If the project team's view of success is different from that of the management team, then the likelihood is that the project will not be successful – either by undershooting the expectations for success or overshooting and making a project too complicated, time consuming, and expensive. Stating both the minimum expectations for success and some optional stretch targets can be very effective. Periodic reviews during the planning and implementation stages of the project to re-assess progress towards the project goals are highly recommended.

Setting optional stretch targets and conducting periodic reviews can have the additional benefit of identifying information that may be available as a standard by-product of a software information system that can provide incremental value to a business at no or little additional cost. Many times, simply asking the question, "what else can we do with this information" opens up additional opportunities to increase overall productivity.

Step #3 – Define Your Equipment

Whether your project involves one piece of equipment or hundreds of pieces of equipment, there is some fundamental information that will need to be understood for any project. As part of the implementation of a software information system, significant details will be required for each piece of equipment that will provide data. However, these details can normally be defined later in the project by the project implementation team or a 3rd party integrator.

In the planning stages of your project, you will want to define any data that is readily available and depend on your implementation team to gather additional data as it is needed.

Appendix B includes an "Equipment Data Sheet" that can be used to document data for each piece of equipment as that information becomes available. One "Equipment Data Sheet" should be completed for each piece of equipment that will provide data from the shop floor. The Equipment Data Sheet(s) should be attached to your MTConnect Project Worksheet as part of your overall project documentation package.

When defining your machines, be aware that different parts of an organization may have different naming conventions for your equipment – Machine 54697 may also be referred to as XYZ Slant-bed lathe, for example. Including these alternate designations for your machine will reduce confusion within a project team and will be very beneficial to 3rd party integrators.

Information about your equipment will include details about the control system implemented on each piece of equipment since the control system is normally the source for data to be collected. You may or may not know details about the control system. Initially gather all the information that is readily available about the control system and include it in the Equipment Data Sheet(s). The implementation team or 3rd party implementer may require support from the equipment or control system manufacturer to gather additional details as required.

Step #4 – Define Limitations/Restraints That Impact Your Project

Many times there are pre-defined limitations or restraints that can have a definitive impact on the scope, implementation strategies, and outcome of a project. It is important to document these as early in the project process as possible. It is also important to update project documents to reflect any changes in these restraints and limitations.

These restraints and limitations may automatically disqualify certain technical solutions from your project. Documenting these limitations is especially valuable to 3rd party integrators and implementers who may be unfamiliar with the detailed operations of your company.

Limitations and restraints can take many forms. They include both topics that cannot be considered as well as those that must be included in the project. Some examples include:

- Budgetary Limitations
- Restrictions on access to specific equipment or personnel
- Approved or disapproved technologies (networking, communications, software, etc.)
- Networking standards, Internet accessibility, etc.
- Privacy and Non-Disclosure Issues
- Security Issues
- "Must Have" Capabilities or Functions
- Future Requirements

Any identified limitations and restrictions should be documented in the MTConnect Project Worksheet.

Step #5 – List the People in Your Organization Who Will Impact the Project

Projects implementing Shop Floor software information systems and tools often impact many people within an organization. Identifying all the individuals impacting a project and documenting their roles is beneficial as both a communications tool and for keeping a project on track. This information will also be valuable to any 3rd party integrators or implementers involved in the project.

For Shop Floor software information systems, it is most common that production and shop floor management personnel are identified as key contributors to the project. However, consideration should be given for the project's impact on the maintenance department, all corporate or outside support organizations, IT and network support teams, and training departments, as examples.

Everyone either directly or indirectly associated with a project should be listed, along with their roles, in Section #5 of the MTConnect Project Worksheet.

SECTION V – I Know What I Need – How Do I Get It Done?

Once the project has been clearly defined, you need to determine the resources required to implement the project. The type of resources and the expertise and knowledge required will depend greatly on the specific equipment that will provide the data, the networking system available, and the complexity of the software system (application) to be implemented.

If all of the equipment is relatively new and has MTConnect functions already implemented, then the major task remaining is the selection of a software application that provides the functions needed. There may still be plenty of effort required to complete the configuration of the software, but the collection of data from the equipment should be a relatively small task. You may consider implementing a software system using your own resources or you may want to contract with a software integrator to implement your software system.

If the equipment does not have MTConnect functions already implemented (MTConnect Translation Devices or MTConnect Data Connection Devices as defined above), then a strategy for implementing the MTConnect standard for each piece of equipment must be considered. Many machine builders and independent 3rd party implementers and integrators either have products that translate the information from your equipment into the MTConnect data structure or have expertise to create such a translation capability. You will typically want to contract with your machinery supplier or a 3rd party to implement your software system.

There are many resources available to help you identify a 3rd party system integrator or implementer for your project. The most extensive resource available is the Directory of Resources provided on the MTConnect website (http://www.mtconnect.org/). You may also want to consider contacting any MTConnect Institute member or submit questions at "Contact Us" on the MTConnect website for more information on integrators and implementers.

SECTION VI – Information (Data) Available using MTConnect

The information or data available from a device that communicates using the MTConnect standard is often impacted by how the Device is connected to the network – as described above. This is not the only consideration. The designer of the MTConnect interface for each device makes a decision on what data will be available from that specific device. There are many considerations that impact this decision. For MTConnect Native devices, a list of the standard data types for the device should be available from the supplier of the MTConnect Agent associated with that device. For MTConnect Translation Dependent Devices that require a Translation Unit or MTConnect Data Connection Dependent Devices that use a Connection Unit, you will have to contact suppliers of these interfaces to determine available data types they can provide through their product offerings.

If you cannot determine the information available for a MTConnect enabled device, it can be collected directly from the device by anyone who is knowledgeable about connecting a computer to the device. A request, called a Probe (See the MTConnect Specification Part 1 for more details), can be sent to any MTConnect Agent. A common way to collect the Probe data is to make a request to the Agent using a standard internet browser on a personal computer – *http//:device_name/probe*. The Agent will respond with a list of the supported data from the device in XML format. The information returned by the Agent provides a complete listing of all available data, its scaling, tag names, units, and the relationship of each data item to a model of the overall device (equipment). The Probe function can also be used by an application to determine the data available from a device and for automatically setting up a data collection routine.

An important consideration for the owner of a piece of equipment, or someone considering purchase of a piece of equipment, that is required to communicate data using the MTConnect Standard is to understand what data is supported by the equipment and how additional data types can be added in the future. Again, this information should be available from your machine or system supplier. The method for adding and configuring new data types to your machine's MTConnect interface could be one factor in differentiating between suppliers – your MTConnect interface should be able to adapt to your future business needs.

Note: By definition, a MTConnect capable device produces data consistent with the MTConnect Specification. The minimum data required by all devices is the `DataItem AVAILABILITY` – essentially an indicator that the device is alive and able to communicate. All other data is optional and is subject to decisions made by the system/machine implementer and the capabilities of the device.

Appendix C provides a listing of many of the data types available in the MTConnect dictionary. This information will be helpful to determine whether the data required by your application is defined in the MTConnect Standard or whether an integrator or implementer may be required to extend the data model to include the data required.

SECTION VII – Connecting a MTConnect Device to a Network

As discussed previously, most machines and shop floor devices will fall into one of three technology groups with respect to MTConnect connectivity. The following is a more detailed description of each of these groups and a discussion of issues that need to be considered for each when connecting the devices to a software application using the MTConnect standard.

MTConnect Native Devices (Figure 5) are devices (machines or equipment) that are provided with MTConnect functions and the data dictionary integrated into the device and it is capable to be connected to a network and MTConnect compliant software applications without the addition of other components. No additional functions must be added to enable communications. Basically, the MTConnect Native Device naturally communicates using the MTConnect Standard and networking technologies.

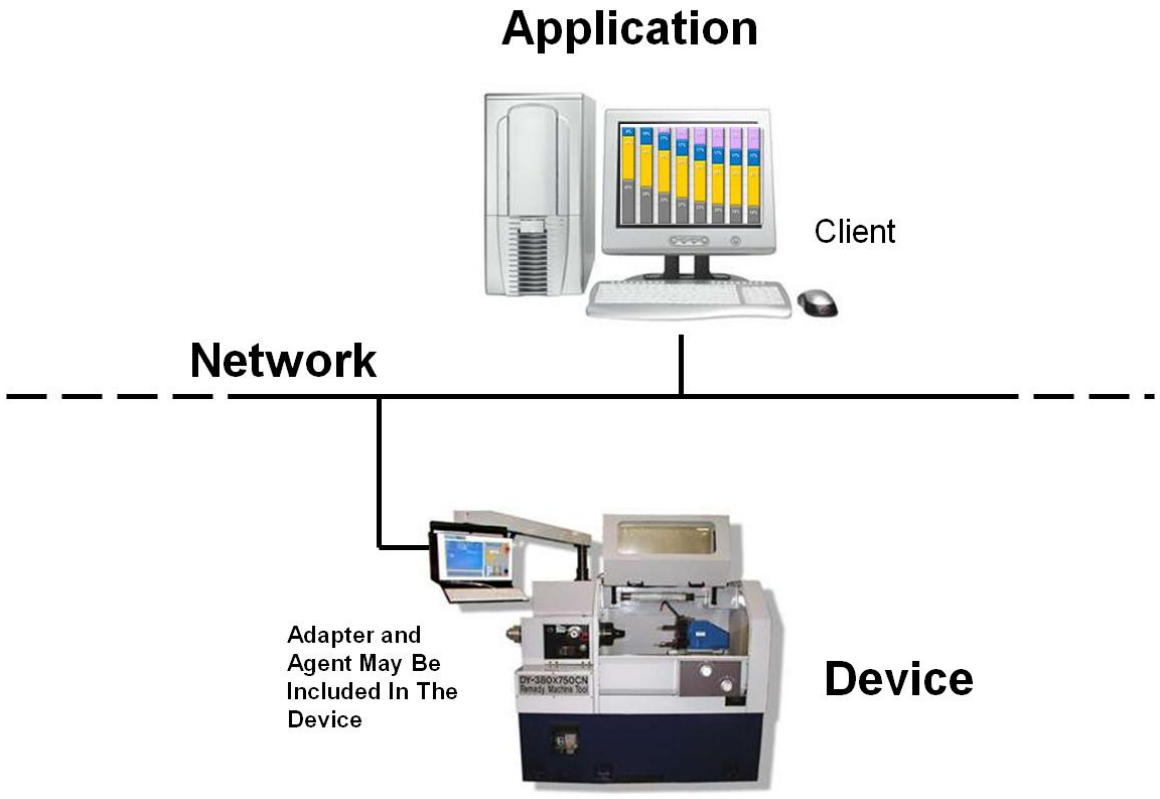

Figure 5 – MTConnect Native Device

Note: The Agent and Adapter for the above device may or may not be located at the Device. It is acceptable that the Agent and Adapter be installed elsewhere in the network architecture. The Agent and Adapter will be configured and provided by the device supplier.

For MTConnect Native devices, little additional information is required before you can begin collecting data using a MTConnect compliant software application. One basic piece of information you will need to know is the IP address for your machine. As described above, you will also need to know what data is supported by your machine currently and how you may or may not be able to add data types in the future.

Armed with this information, you should have all the core information needed to implement your software information system using data compliant with the MTConnect standard. Your remaining task will be to select one or multiple MTConnect compliant applications that will use the data produced by your equipment.

MTConnect Translation Dependent Devices (Figure 6) are devices (machines or equipment) that required a separate function, or Translation Unit, (software and/or hardware) to translate data from the native language of the device into MTConnect compliant data. Data is available to a software application(s) from the device but it needs to be translated into the MTConnect standard dictionary names via an Adapter. The Translation Unit, as defined for MTConnect Translation Dependent Devices, performs the function of linking two different communication systems so that they can exchange information with each other. This translation may include both physical changes (e.g. serial communications converted to Ethernet) and logical changes to translate data from the native language of the device to MTConnect compliant data.

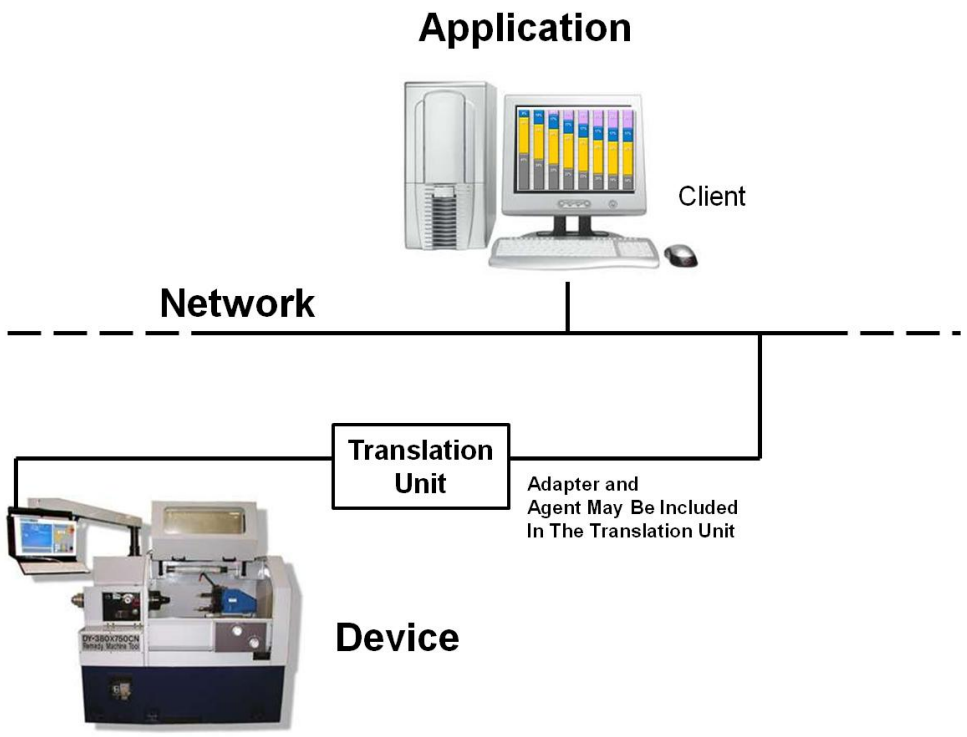

Figure 6 – MTConnect Translation Dependent Device

The Translation Unit may be available from your machine supplier or a 3rd party solution provider who specializes in implementing MTConnect systems.

If a commercially available Translation Unit is not available for your machine, one may be able to be developed by an MTConnect Solution provider.

Depending upon architecture of your equipment's controller, the Translation Unit function can be embedded in the controller or it can be installed external to your machine. For example, equipment with PC-based controllers can typically have the Translation Unit function software embedded in the controller itself. For machines that do not allow additional software to be added, the Translation Unit function can be installed on an attached hardware device – typically a small PC device located in or near your machine.

The development of the Translation Unit function will require knowledge of the data provided by the equipment in its native language and typically detailed knowledge of where that data is stored in the controller of the equipment.

An MTConnect compliant application only needs to communicate with the Agent of the Translation Unit, not directly with the equipment. The Translation Unit effectively isolates an application from the proprietary communications requirements of the equipment and provides data in a format compliant with the standard MTConnect standard.

As with the Native MTConnect devices, little additional information is required before you can begin collecting data. You will need to know the IP address for the device where your Translation Unit function is located. As described above, you will also need to know what data is supported by the Translation Unit and how, or if, you may be able to add data item types in the future.

MTConnect Data Connection Dependent Devices (Figure 7) are devices (machines or equipment) that do not normally provide the data required by a software application through any standard connection method. Many of these devices do not naturally publish any data or the data may be limited. Machines in this category require a separate Connection Unit (software and/or hardware) to collect data from the device and make that data available to software applications. The Connection Unit also includes the functionality of a Translation Unit to convert that data into MTConnect compliant data.

Machines with no, or very limited, communications ability are by far the largest group of machines installed in plants today. A number of implementers and integrators are developing solutions to gather data from these machines and making that data available in a format compatible with the MTConnect standard.

With the appropriate Data Connection interface, these machines can now be integrated into a software information system – significantly increasing the potential to gain increased value from these assets.

There are various technical alternatives for implementing a Data Connection interface. You should contact your equipment manufacturer or a MTConnect system integrator to determine the solutions available for your specific piece of equipment. More information on System Integrators can be found in the directory located on the MTConnect website (http://www.mtconnect.org/).

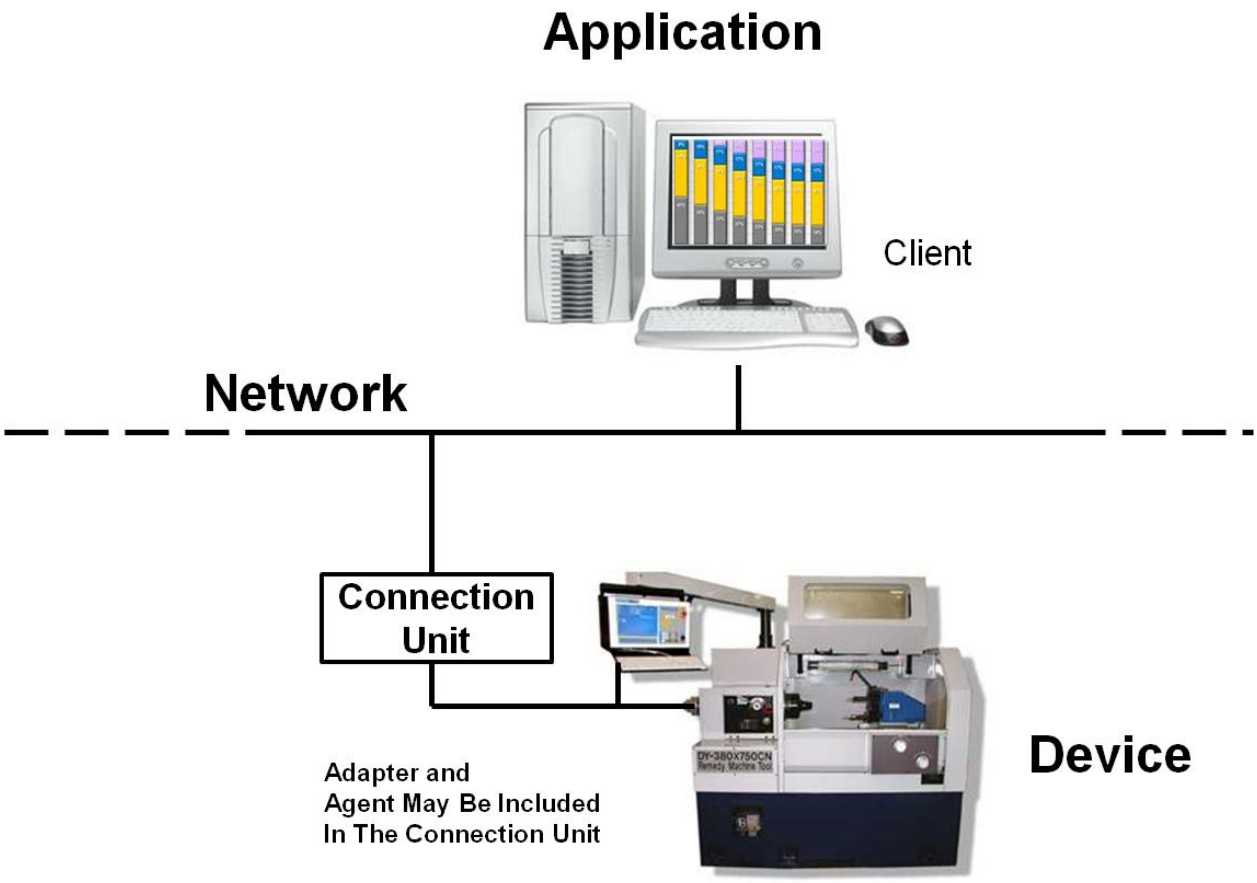

Figure 7 – MTConnect Connection Dependent Device

As with the equipment using other connection types, some additional information is required before you can begin collecting data using a MTConnect compliant software application. You will need to know the IP address for your Connection Unit. As described above, you will also need to know what data is supported by your Connection Unit currently and how you may or may not be able to add new data types in the future. Since the connection methods to the machine and the data sources within the machines vary significantly for this category of machines, the issue of changing or expanding the available data types may be more complex or restrictive. You should coordinate closely with your 3rd party integrator or implementer to fully evaluate both your current and future data needs and how these needs can be addressed using the data connection method available for your device.

SECTION VI – Conclusions

MTConnect provides a valuable link to provide data from the shop floor to software applications used to run your business. While focused initially on connectivity to machine tools, MTConnect is designed to be applied to any type of equipment or device on your shop floor.

As described above, machines normally fit into one of three levels of technology. There are solutions to connect any of them to a compatible software application – MTConnect Native, MTConnect Translation Dependent, or MTConnect Data Connection Dependent devices. Whether you have new equipment or older equipment, it typically can be configured to provide data using the MTConnect standard.

For more information on the MTConnect Institute, its partners, and the MTConnect Standard, please visit http://www.mtconnect.org/ . On this site, you can also find a listing of technology companies (integrators and implementers) who provide products and services to enable your equipment to communicate data using the MTConnect Standard and offer software applications that use MTConnect data - http://mtconnect.org/images/stories/WebSite%20Listing%20of%20Resources_04-26_11.pdf.

Should you need additional information, please contact:

Mr. Paul Warndorf
Vice President of Technology
Association for Manufacturing Technology
pwarndorf@MTConnect.hyperoffice.com

The MTConnect Standard is not a static document. A significant number of Committees are continuing the development and extension of the standard to address new application areas for the shop floor. Committees are teams of industry experts cooperating together to extend the MTConnect Standard for the general benefit of the entire industry. These Committees are presently working on Robotics/Machine Loading, Sensors, Tooling, EDM machining, Machine Accessories, Alarming and Notifications, and Software Application Developments. The Institute encourages new members to join and to participate in these Committees or to form new Committees to extend the MTConnect standard for applications of specific interest to the group.

The MTConnect standard can only deliver benefits when it is installed on your shop floor. This means that you need to request MTConnect compliant equipment and software applications from your suppliers on future projects – driving down installation time and support costs. The availability of equipment and software applications based on MTConnect will continue to expand based on the level of demand for these products. Shop owners are encouraged to include connectivity to MTConnect as a standard part of your future equipment purchase requirements.

Appendix A – MTConnect Glossary of Terms

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

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

Alarm An alarm indicates an event that requires attention and indicates a deviation from normal operation. Alarms are reported in MTConnect as `Condition`.

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

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

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

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

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

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

Data Item A data item provides the descriptive information regarding something that can be collected by the *Agent*.

Device A piece of equipment capable of performing an operation. A device may be composed of a set of components that provide data to the application. The device is a separate entity with at least one component or data item providing information about the device.

Discovery Discovery is a service that allows the application to locate *Agents* for devices in the manufacturing environment. The discovery service is also referred to as the *Name Service.*

Event An event represents a change in state that occurs at a point in time. Note: An event does not occur at predefined frequencies.

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

Instance	When used in software engineering, the word *instance* is used to define a single physical example of that type. In object-oriented models, there is the class that describes the thing and the instance that is an example of that thing.
LDAP	Lightweight Directory Access Protocol, better known as Active Directory in Microsoft Windows. This protocol provides resource location and contact information in a hierarchal structure.
MIME	Multipurpose Internet Mail Extensions. A format used for encoding multipart mail and http content with separate sections separated by a fixed boundary.
Probe	A request to determine the configuration and reporting capabilities of the device.
REST	REpresentational State Transfer. A software architecture where the client and server move through a series of state transitions based solely on the request from the client and the response from the server.
Results	A general term for the `Samples`, `Events`, and `Condition` contained in a `ComponentStream` as a response from a `sample` or `current` request.
Sample	A sample is a data point from within a continuous series of data points. An example of a Sample is the position of an axis.
Socket	When used concerning inter-process communication, it refers to a connection between two end-points (usually processes). Socket communication most often uses TCP/IP as the underlying protocol.
Stream	A collection of `Events, Samples,` and `Condition` organized by devices and components.
Service	An application that provides necessary functionality.
Tag	Used to reference an instance of an XML element.
TCP/IP	TCP/IP is the most prevalent stream-based protocol for inter-process communication. It is based on the IP stack (Internet Protocol) and provides the flow-control and reliable transmission layer on top of the IP routing infrastructure.
URI	Universal Resource Identifier. This is the official name for a web address as seen in the address bar of a browser.
UUID	Universally unique identifier.
XPath	XPath is a language for addressing parts of an XML Document. See the XPath specification for more information. http://www.w3.org/TR/xpath
XML	Extensible Markup Language. http://www.w3.org/XML/
XML Schema	The definition of the XML structure and vocabularies used in the XML Document.

XML Document An instance of an XML Schema which has a single root XML element and conforms to the XML specification and schema.

XML Element An element is the central building block of any XML Document. For example, in MTConnect® the Device XML element is specified as <**Device**>...</**Device**>

XML NMTOKEN The data type for XML identifiers. It **MUST** start with a letter, an underscore "_" or a colon ":" and then it **MUST** be followed by a letter, a number, or one of the following ".", "-", "_", or ":". An NMTOKEN cannot have any spaces or special characters.

Appendix B – MTConnect Project Work Sheet

The attached MTConnect Project Work Sheet is provided as a guideline for gathering the documentation that would be useful for planning a project for implementing a Shop Floor software information system and for preparing for a discussion with 3rd Party implementers or integrators. The shop or plant will benefit by collecting as much information as possible to properly define the capabilities of the manufacturing equipment. Where information cannot be determined, it should be noted for future reference.

MTConnect

Project Planning Worksheet

Step #1 – Define the Problem That You Are Trying to Solve

Provide a clear and concise description of the problem to be solved. Ask the "Why do I want to know this" question repeatedly until the description of the problem you are trying to solve becomes clear to everyone and is easy to articulate.

_____
_____
_____
_____
_____
_____

Step #2 – What Does Success Look Like?

Clearly define the minimum functionality or information provided by a software information system or the business outcome that defines success for your project.

_____
_____
_____
_____

Define optional stretch targets for additional information or functionality that would be beneficial to the business.

_____
_____
_____
_____
_____
_____
_____
_____
_____

Step #3 – Define Your Equipment

Complete an Equipment Data Sheet for each piece of equipment to be monitored. Provide as much detail as is readily available. Also use the Equipment Data Sheet to incorporate additional data as it becomes available.

Attach a copy of the Equipment Data Sheet to this Project Planning Worksheet for each piece of equipment that will be required to provide data for this project.

Step #4 – Define Limitations/Restraints That Impact Your Project

List any known limitations and restraints that will impact your project. Take care to include both topics that cannot be included in the scope of the project as well as those "Must Haves" that must be included.

_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

Step #5 – List the People in Your Organization Who Will Impact the Project

List all individuals in the organization that will be either directly or indirectly involved in the project. It is also important to document their role as it applies to the project.

Key Person	Project Role
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

MTConnect
Equipment Data Sheet

Equipment Name: _____ Alternate ID: _____

Equipment Manufacturer: _____ Serial Number: _____

Equipment Description: _____ Year of Manufacture: _____

Control System Information:

Manufacturer: _____
Model Number: _____
Serial Number: _____
Version Number: _____
Communication Connection Method: _____(i.e. Ethernet, RS232, etc.)__
Other Control System Data:
_____
_____
_____
_____

Data Types:

Data Type (Item)	Required	Available
Operating Mode (Sample)	X	X

Appendix C – MTConnect Data Model and Data Types

MTConnect presently supports a wide variety of data types required for communicating data from the equipment on your shop floor.

MTConnect uses an XML Data Model to describe a piece of equipment and to provide relevance for each piece of data communicated using the MTConnect Standard. The major equipment components presently defined in the MTConnect standard include:

- Controller
  - Path
- Linear Axes
- Rotary Axes
- Tooling
- Actuators
- Doors
- Sensors
- Systems
  - Hydraulic
  - Pneumatic
  - Lubrication
  - Coolant
  - Electric

In addition to the XML Data Model for a piece of equipment, the MTConnect data dictionary provides three (3) categories of data types used to describe information about the equipment and the machining process. Each piece of information defined in the MTConnect data dictionary is called a `DataItem`.

The first category of data type is called `Sample`. A `Sample` is the reading of the value of a continuously variable or analog `DataItem`. A continuous value can be sampled at any point-in-time and will always produce a result. An example of a continuous `DataItem` is the Linear X axis position.

`Sample` category data items that are presently defined in the MTConnect data dictionary include:

Data Item type/subtype	Description
ACCELERATION	**Rate of change of velocity**
ACCUMULATED_TIME	**The measurement of accumulated time associated with a** `Component`
ANGULAR_ACCELERATION	**Rate of change of angular velocity.**
ANGULAR_VELOCITY	**Rate of change of angular position.**
AMPERAGE	**The measurement of AC Current or a DC current**
ALTERNATING	**The measurement of alternating current. If not specified further in** `statistic`**, defaults to RMS Current**
DIRECT	**The measurement of DC current**
ANGLE	**The angular position of a component relative to the parent.**
ACTUAL	**The angular position as read from the physical component.**
COMMANDED	**The angular position computed by the** `Controller`**.**
AXIS_FEEDRATE	**The feedrate of a linear axis.**
ACTUAL	**The actual federate of a linear axis.**
COMMANDED	**The feedrate as specified in the program.**
OVERRIDE	**The operator's overridden value. Percent of commanded.**
CLOCK_TIME	**The reading of a timing device at a specific point in time. Clock time MUST be reported in W3C ISO 8601 format.**
CONCENTRATION	**Percentage of one component within a mixture of components**
CONDUCTIVITY	**The ability of a material to conduct electricity**
DISPLACEMENT	**The displacement as the change in position of an object**
ELECTRICAL_ENERGY	**The measurement of electrical energy consumption by a component**
FILL_LEVEL	**The measurement of the amount of a substance remaining compared to the planned maximum amount of that substance**
FLOW	**The rate of flow of a fluid**
FREQUENCY	**The measurement of the number of occurrences of a repeating event per unit time**
LINEAR_FORCE	**The measure of the push or pull introduced by an actuator or exerted on an object**
LOAD	**The measurement of the percentage of the standard rating of a device**
MASS	**The measurement of the mass of an object(s) or an amount of material**
PATH_FEEDRATE	**The feedrate of the tool path.**
ACTUAL	**The three-dimensional feedrate derived from the** `Controller`**.**
COMMANDED	**The feedrate as specified in the program**

Data Item type/subtype	Description
OVERRIDE	**The operator's overridden value. Percent of commanded.**
PATH_POSITION	**The current program control point or program coordinate in** WORK **coordinates. The coordinate system will revert to** MACHINE **coordinates if** WORK **coordinates are not available.**
ACTUAL	**The position of the** Component **as read from the device.**
COMMANDED	**The position computed by the** Controller**.**
TARGET	**The target position for the movement.**
PROBE	**The position provided by a probe**
PH	**The measure of the acidity or alkalinity.**
POSITION	**The position of the** Component**. Defaults to** MACHINE **coordinates.**
ACTUAL	**The position of the** Component**.**
COMMANDED	**The position as given by the** Controller**.**
TARGET	**The target position for the movement.**
POWER_FACTOR	**The measurement of the ratio of real power flowing to a load to the apparent power in that AC circuit.**
PRESSURE	**The force per unit area exerted by a gas or liquid**
RESISTANCE	**The measurement of the degree to which an object opposes an electric current through it**
ROTARY_VELOCITY	**The rotational speed of a rotary axis.**
ACTUAL	**The rotational speed the rotary axis is spinning at.** ROTARY_MODE **MUST be** SPINDLE**.**
COMMANDED	**The rotational speed as specified in the program.**
OVERRIDE	**The operator's overridden value. Percent of commanded.**
SOUND_LEVEL	**Measurement of a sound level or sound pressure level relative to atmospheric pressure**
NO_SCALE	**No weighting factor on the frequency scale**
A_SCALE	**A Scale weighting factor. This is the default weighting factor if no factor is specified**
B_SCALE	**B Scale weighting factor**
C_SCALE	**C Scale weighting factor**
D_SCALE	**D Scale weighting factor**
STRAIN	**Strain is the amount of deformation per unit length of an object when a load is applied.**
TEMPERATURE	**The measurement of temperature**
TILT	**A measurement of angular displacement**
TORQUE	**The turning force exerted on an object or by an object**

Data Item type/subtype	Description
VOLT_AMPERE	**The measure of the apparent power in an electrical circuit, equal to the product of root-mean-square (RMS) voltage and RMS current' (commonly referred to as VA)**
VOLT_AMPERE_REACTIVE	**The measurement of reactive power in an AC electrical circuit (commonly referred to as var)**
VELOCITY	**The rate of change of position.**
VISCOSITY	**A measurement of a fluid's resistance to flow**
VOLTAGE	**The measurement of electrical potential between two points**
ALTERNATING	**The measurement of alternating voltage. If not specified further in** `statistic`**, defaults to RMS voltage**
DIRECT	**The measurement of DC voltage**
WATTAGE	**The measurement of power consumed or dissipated by an electrical circuit or device**

The next data type is called `Event`. An `Event` comprises discrete information from the device. There are two types of `Event`: those representing state, with two or more discrete values; and those representing messages that contain plain text data. An example of a state type `Event` is a `Door_State` that can be either `OPEN`, `UNLATCHED`, or `CLOSED`. An example of a message type `Event` is a `PROGRAM` that can be any valid string of characters. A `DataItem` of category `Event` does not have intermediate values that vary over time, as does `Sample`. An `Event` can be thought of as information that if taken at any point in time represents the current state of the device.

`Event` category data items that are presently defined in the MTConnect data dictionary include:

Data Item type/subtype	Description
ACTUATOR_STATE	The state of the `Actuator` - `ACTIVE or INACTIVE`.
ACTIVE_AXES	The set of axes associated with a `Path` that the `Controller` is controlling. If this `DataItem` is not provided, it will be assumed the `Controller` is controlling all axes.
AVAILABILITY	Represents the ability of a `Component` to communicate. This **MUST** be provided for a `Device` and **MAY** be provided for any other `Component`. `AVAILABLE or UNAVAILABLE`.
AXIS_COUPLING	Describes the way the axes will be associated to each other. This is used in conjunction with `COUPLED_AXES` to indicate the way they are interacting. The possible values are: `TANDEM`, `SYNCHRONOUS`, `MASTER`, and `SLAVE`. The coupling **MUST** be viewed from the perspective of the axis; therefore a `MASTER` coupling indicates that this axis is the master of the `COUPLED_AXES`.

Data Item type/subtype	Description
BLOCK	The block of code being executed. `Block` contains the entire expression for a line of program code.
CONTROLLER_MODE	The current mode of the `Controller`. `AUTOMATIC`, `MANUAL`, `MANUAL_DATA_INPUT`, or `SEMI_AUTOMATIC`.
COUPLED_AXES	Refers to the set of associated axes. The value will be a space delimited set of axes names.
DIRECTION	The direction of motion. `CLOCKWISE or COUNTER_CLOCKWISE`
ROTARY	The rotational direction of a rotary device using the right hand rule convention. `CLOCKWISE or COUNTER_CLOCKWISE`
LINEAR	The direction of motion of a linear device. `POSTIVE` or `NEGATIVE`
DOOR_STATE	The opened or closed state of the door. `OPEN, UNLATCHED, or CLOSED.`
EMERGENCY_STOP	The current state of the emergency stop actuator. ARMED (the circuit is complete and the `device` is operating) or TRIGGERED (the circuit is open and the device MUST cease operation).
EXECUTION	The execution status of the `Controller`. `READY, ACTIVE, INTERRUPTED, FEED_HOLD, or STOPPED`
LINE	The current line of code being executed
MAXIMUM	The maximum line number of the code being executed.
MINIMUM	The minimum line number of the code being executed.
MESSAGE	An uninterpreted textual notification.
PALLET_ID	The identifier for the pallet currently in use for a given `Path`
PART_COUNT	The current count of parts produced as represented by the `Controller`.
ALL	The count of all the parts produced. If the subtype is not given, this is the default.
GOOD	Indicates the count of correct parts made.
BAD	Indicates the count of incorrect parts produced.
PART_ID	An identifier of the current part in the device
PATH_MODE	The operational mode for this `Path`. `SYNCHRONOUS`, `MIRROR`, or `INDEPENDENT`. Default value is `INDEPENDENT` if not specified.
POWER_STATE	The ON or OFF status of the `Component`.
LINE	The state of the high voltage line.
CONTROL	The state of the low power line.
PROGRAM	The name of the program being executed
ROTARY_MODE	The mode for the `Rotary` axis. `SPINDLE`, `INDEX`, or `CONTOUR`.
TOOL_ASSET_ID	The identifier of the tool currently in use for a given `Path`
WORKHOLDING_ID	The identifier for the workholding currently in use for a given `Path`

The third data type is called `Condition`. A `Condition` communicates the device's health and ability to function. A `DataItem` of category `Condition` can be one of `UNAVAILABLE`, `NORMAL`, `WARNING`, or `FAULT`.

`Condition` category data items that are presently defined in the MTConnect data dictionary include:

Data Item type/ qualifier	Description
ACCELERATION	Rate of Change of Velocity
ACCUMULATED_TIME	The measurement of accumulated time associated with a `Component`
ACTUATOR	An actuator related condition.
AMPERAGE	A high or low condition for the electrical current.
ANGLE	The angular position of a `Component`.
ANGULAR-ACCELERATION	Rate of change of angular velocity.
ANGULAR_VELOCITY	Rate of change of angular position
COMMUNICATIONS	A communications failure indicator.
CONCENTRATION	Percentage of one ingredient within a mixture of ingredients
CONDUCTIVITY	The ability of a material to conduct electricity
DATA_RANGE	Information provided is outside of expected value range
DIRECTION	The direction of motion of a `Component`
DISPLACEMENT	The change in position of an object
ELECTRICAL_ENERGY	The measurement of electrical energy consumption by a `Component`
FILL_LEVEL	Represents the amount of a substance remaining compared to the planned maximum amount of that substance
FLOW	The rate of flow of a fluid
FREQUENCY	The number of occurrences of a repeating event per unit time
HARDWARE	The hardware subsystem of the `Component's` operation condition.
LINEAR_FORCE	The measure of the push or pull introduced by an actuator or exerted by an object
LOAD	The measure of the percentage of the standard rating of a device
LOGIC_PROGRAM	An error occurred in the logic program or PLC (programmable logic controller).
MASS	The measurement of the mass of an object(s) or an amount of material
MOTION_PROGRAM	An error occurred in the motion program.
PATH_FEEDRATE	The federate of the tool path
PATH_POSITION	The current control point of the path
PH	The measure of acidity or alkalinity
POSITION	The position of a `Component`.

Data Item type/ qualifier	Description
POWER_FACTOR	The ratio of real power flowing to a load to the apparent power in that AC circuit.
PRESSURE	The measurement of the force per unit area exerted by a gas or liquid.
RESISTANCE	The measurement of the degree to which an object opposes an electric current through it
ROTARY_VELOCITY	The rotational speed of a rotary axis
SOUND_LEVEL	The measurement of sound pressure level
STRAIN	Indicates the amount of deformation per unit length of an object when a load is applied
SYSTEM	A condition representing something that is not the operator, program, or hardware. This is often used for operating system issues.
TEMPERATURE	Indicates the temperature of a `Component`.
TILT	The measure of angular displacement
TORQUE	The measured of the turning force exerted on an object or by an object
VOLT_AMPERAGE	The measure of the apparent power in an electrical circuit (commonly referred to as VA)
VOLT_AMPERAGE_REACTIVE	The measure of reactive power in an AC electrical power circuit (commonly referred to as var).
VELOCITY	Indicated the velocity of a component.
VISCOSITY	The measure of a fluid's resistance to flow
VOLTAGE	The measurement of electrical potential between two points
WATTAGE	The measurement of power consumed or dissipated by an electrical circuit or device

In addition to the categories of `DataItems` listed above addressing equipment, the MTConnect data dictionary and XML model provides an extensive set of data for defining and modeling cutting tools. This information is applicable to an integrated tool management system linking information from the Tool Crib, Pre-Setters, Tooling, and the Machine Tool into an integrated information model for tools. For more information on the modeling and data items available for tooling, see *Part 4 of the MTConnect Standard*.