

ARCHITECTURE FOR NETWORK-ENABLED TOOLS

BACKGROUND

5 [0001] The present disclosure relates generally to network architecture and more particularly to network enabled tools. Still more particularly, the present disclosure relates to a data processing system for monitoring and managing network enabled tools.

10

 [0002] Tools used in production programs are often equipped with computer controls that electronically control process parameters for the tools. These process parameters require the tools to be checked into a remotely located tool crib and
15 updated manually when process parameters change. A tool crib is a room or physical location in a production environment where tools are stored when not in use. The tool crib, or tool library, manages the inventory of the tools when they are checked into the tool crib.

20 [0003] When a tool is checked into the tool crib, process data from operation of the tool in the production environment may be downloaded or retrieved for storage and analysis. If a process parameter has been updated since the last time the tool was checked into the tool crib, the tool may receive a process
25 update manually at the tool crib. If a process parameter changes while a tool is in operation in the production environment, the process update is not applied to the tool until the tool is next checked into the tool crib.

 [0004] Some point-to-point wired Ethernet solutions exist in
30 limited scenarios to capture process data from a tool during operation, storing the captured data on a server. This stored data may be automatically or manually captured by the tool operator. Manual method of capture includes downloading of information onto a laptop computer that is connected to the tool

by Ethernet. The data stored on the laptop hard disk drive resides there until deleted or it may be further downloaded to a portable memory media such as a CD-ROM. Automated solutions incorporate software to create a point-to-point connection with
5 a remote database on the network or a data repository within or nearby the machine.

[0005] Therefore, it would be advantageous to have a method and apparatus that addresses one or more of the issues discussed above.

10

SUMMARY

[0006] The different advantageous embodiments provide a system comprising a wireless network, a number of tools, and a central server. The number of tools is in communication with the wireless network. The number of tools is configured to perform a number of tasks and collect process data while performing the number of tasks. The central server is in communication with the wireless network. The central server is configured to monitor the number of tools performing the number of tasks and receive the process data collected by the number of tools over the wireless network.

[0007] The different advantageous embodiments further provide a method for tool management. A number of tools is implemented in a wireless network. Tool processes for the number of tools is monitored at a central location using the wireless network. Process data is received at the central location for the number of tools during the tool processes using the wireless network.

[0008] The different advantageous embodiments further provide a method for tool process updates. A tool mechanism is controlled to perform a task using a number of process parameters. Process data is collected during task performance. The process data collected is sent to a remote location.

[0009] The features, functions, and advantages can be achieved independently in various embodiments of the present disclosure or may be combined in yet other embodiments in which further details can be seen with reference to the following description and drawings.

[0009a] According to an aspect, there is provided a system comprising: an enterprise-level network architecture including a communication system; a number of tools in communication with the communication system and configured to perform a number of tasks using a number of process parameters, and collect process data while performing the number of tasks; wherein the number of tools further comprises a sensor system configured to collect the process data while the number of tools perform the number of tasks; and a central server in communication with the communication system and configured to: provide the process parameters for the number of tools; monitor the number of tools as the number of tools perform the number of tasks; receive the process data collected by the number of tools as the number of tools perform the number of tasks using the communication system; analyze the process data received from the number of tools as the number of tools perform the number of tasks; and determine whether a process parameter update is needed while the number of tools perform the number of tasks; wherein the central server is configured to send a number of process parameter updates to the number of tools using the communication system; and wherein the number of tools further comprises a data processing system configured to receive the number of process parameter updates from the central server while the number of tools perform the number of tasks.

[0009b] According to another aspect, there is provided a method for tool management within an enterprise-level network architecture including a communication system, the method comprising: monitoring tool processes for a number of tools in communication with the communication system as the number of tools perform a number of tasks, wherein the number of tools collect process data while performing the number of tasks, and wherein the monitoring is performed at a central location using the communication system; providing process parameters for the number of tools; receiving the process data for the number of tools during the tool processes at the central location using the communication system; wherein the number of tools includes a number of sensor systems, and wherein the process data is collected by the number of tools using the number of sensor systems while the number of tools perform the number of tasks; analyzing the process data collected from the number of tools during the tool processes; determining whether a process parameter update is needed for the number of tools while the number of tools perform the number of tasks; and responsive to a determination that the process parameter update is needed, sending a number of process parameter updates to the number of tools from the central location using the communication system, wherein the number of process parameter updates is sent to the number of tools while the number of tools are in operation.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The novel features believed characteristic of the advantageous embodiments are set forth in the appended claims.

5 The advantageous embodiments, however, as well as a preferred mode of use, further objectives and advantages thereof, will best be understood by reference to the following detailed description of an advantageous embodiment of the present disclosure when read in conjunction with the accompanying
10 drawings, wherein:

[0011] Figure 1 is an illustration of a smart tool environment in which an advantageous embodiment may be implemented;

[0012] Figure 2 is an illustration of a data processing
15 system in accordance with an advantageous embodiment;

[0013] Figure 3 is an illustration of a tool in accordance with an advantageous embodiment;

[0014] Figure 4 is an illustration of a network enabled tool environment in accordance with an advantageous embodiment;

20 [0015] Figure 5 is an illustration of a flowchart illustrating a process for tool management in accordance with an advantageous embodiment; and

[0016] Figure 6 is an illustration of a flowchart illustrating a process for tool process updates in accordance
25 with an advantageous embodiment.

DETAILED DESCRIPTION

[0017] With reference now to the figures and, in particular, with reference to **Figure 1**, an illustration of a smart tool environment is depicted in which an advantageous embodiment may be implemented. Smart tool environment 100 may be any type of environment in which tools operate. For example, smart tool environment 100 may be a production environment, a manufacturing environment, a maintenance environment, a service environment, and/or any other suitable environment.

[0018] Smart tool environment 100 includes network 102, back office 104, and factory 106. Network 102 provides wireless access between components of factory 106 and back office 104. Network 102 may be a secure wireless network with protected access using certification protocols, such as, for example, without limitation, WPA, WPA2, and/or any other suitable security protocol. Back office 104 may include central data server 108, dynamic host communication protocol (DHCP) server 110, certificate authority 112, database 114, and user interface 116. In one advantageous embodiment, back office 104 may be a computer implementing processes, such as central data server 108, dynamic host communication protocol (DHCP) server 110, and certificate authority 112, for example. In another advantageous embodiment, back office 104 may be a remote location having one or more computers, such as central data server 108, dynamic host communication protocol (DHCP) server 110, and certificate authority 112, for example.

[0019] Central data server 108 provides process parameters for tools, such as number of tools 120 in factory 106. Central data server 108 receives process data from number of tools 120, analyzes the process data, and determines whether one or more process updates are needed for number of tools 120. Central data server 108 is a central source for monitoring and managing number of tools 120.

[0020] Dynamic host communication protocol (DHCP) server 110 allows number of tools 120 to request IP addresses and log onto network 102, in this illustrative example. Certificate authority 112 processes a number of certificates for security and manages the secure interaction of number of tools 120 with back office 104 over network 102. As used herein, a number of items means one or more items. For example number of tools 120 is one more tools.

[0021] Factory 106 includes number of access points 118 and number of tools 120. Number of access points 118 may be a number of wireless access points at a number of locations throughout factory 106 that enable number of tools 120 to have network access throughout factory 106. A wireless access point may be a device that allows wireless communication devices to connect to a wireless network using Wi-Fi, Bluetooth, or related standards. A wireless access point may connect to a router to relay data between the wireless devices, such as number of tools 120, and other devices on the network, such as back office 104, for example. Number of wireless access points 118 may be stand-alone devices or integrated devices with number of tools 120, for example. Number of tools 120 is a tool that electronically controls process parameters for tool mechanisms, for example.

[0022] Tool 122 is an illustrative example of one implementation of number of tools 120. Tool 122 includes data processing system 124, gateway 126, and controller 128. Data processing system 124 manages data collected from tool 122 during operation and data received from back office 104 during operation, for example. Gateway 126 provides access for data processing system 124 to network 102 using number of access points 118, for example. Gateway 126 may be, for example, without limitation, a dedicated computer, a router, a Wi-Fi serial bridge, and/or any other suitable gateway device.

[0023] Controller 128 controls operation of tool 122 using process parameters received from data processing system 124. Controller 128 may be, for example, without limitation, a motion controller.

5 [0024] Number of tools 120 may be any type of computer aided tool, such as, without limitation, contour drilling machine 130, drill motor 132, fastener 134, and/or any other suitable tool.

[0025] The illustration of smart tool environment 100 in Figure 1 is not meant to imply physical or architectural
10 embodiments may be implemented. Other components in addition and/or in place of the ones illustrated may be used. Some components may be unnecessary in some advantageous embodiments. Also, the blocks are presented to illustrate some functional components. One or more of these blocks may be combined and/or
15 divided into different blocks when implemented in different advantageous embodiments.

[0026] In some advantageous embodiments, back office 104 may be implemented as a data processing system, such as data processing system 200 in Figure 2, below. In other advantageous
20 embodiments, back office 104 may be implemented as a number of data processing systems connected over network 102, for example. In an illustrative example, back office 104 may be local, such as an office adjacent to factory 106, or remote, such as a server farm.

25 [0027] With reference now to Figure 2, an illustration of a data processing system is depicted in accordance with an advantageous embodiment. Data processing system 200 may be an example of a computer, such as central data server 108 and/or data processing system 124 in Figure 1, in which computer usable
30 program code or instructions implementing the processes may be located for the advantageous embodiments.

[0028] In this illustrative example, data processing system 200 includes communications fabric 202, which provides communications between processor unit 204, memory 206,

persistent storage 208, communications unit 210, input/output (I/O) unit 212, and display 214.

[0029] Processor unit 204 serves to execute instructions for software that may be loaded into memory 206. Processor unit 204
5 may be a set of one or more processors or may be a multi-processor core, depending on the particular implementation. Further, processor unit 204 may be implemented using one or more heterogeneous processor systems in which a main processor is present with secondary processors on a single chip. As another
10 illustrative example, processor unit 204 may be a symmetric multi-processor system containing multiple processors of the same type.

[0030] Memory 206 and persistent storage 208 are examples of storage devices 216. A storage device is any piece of hardware
15 that is capable of storing information, such as, for example without limitation, data, program code in functional form, and/or other suitable information either on a temporary basis and/or a permanent basis. Memory 206, in these examples, may be, for example, a random access memory or any other suitable
20 volatile or non-volatile storage device. Persistent storage 208 may take various forms depending on the particular implementation. For example, persistent storage 208 may contain one or more components or devices. For example, persistent storage 208 may be a hard drive, a flash memory, a rewritable
25 optical disk, a rewritable magnetic tape, or some combination of the above. The media used by persistent storage 208 also may be removable. For example, a removable hard drive may be used for persistent storage 208.

[0031] Communications unit 210, in these examples, provides
30 for communications with other data processing systems or devices. In these examples, communications unit 210 is a network interface card. Communications unit 210 may provide communications through the use of either or both physical and wireless communications links.

[0032] Input/output unit 212 allows for input and output of data with other devices that may be connected to data processing system 200. For example, input/output unit 212 may provide a connection for user input through a keyboard, a mouse, and/or
5 some other suitable input device. Further, input/output unit 212 may send output to a printer. Display 214 provides a mechanism to display information to a user.

[0033] Instructions for the operating system, applications and/or programs may be located in storage devices 216, which are
10 in communication with processor unit 204 through communications fabric 202. In these illustrative examples, the instructions are in a functional form on persistent storage 208. These instructions may be loaded into memory 206 for execution by processor unit 204. The processes of the different embodiments
15 may be performed by processor unit 204 using computer implemented instructions, which may be located in a memory, such as memory 206.

[0034] These instructions are referred to as program code, computer usable program code, or computer readable program code
20 that may be read and executed by a processor in processor unit 204. The program code in the different embodiments may be embodied on different physical or tangible computer readable media, such as memory 206 or persistent storage 208.

[0035] Program code 218 is located in a functional form on
25 computer readable media 220 that is selectively removable and may be loaded onto or transferred to data processing system 200 for execution by processor unit 204. Program code 218 and computer readable media 220 form computer program product 222 in these examples. In one example, computer readable media 220 may
30 be computer readable storage media 224 or computer readable signal media 226. Computer readable storage media 224 may include, for example, an optical or magnetic disc that is inserted or placed into a drive or other device that is part of persistent storage 208 for transfer onto a storage device, such

as a hard drive that is part of persistent storage 208.

Computer readable storage media 224 also may take the form of a persistent storage, such as a hard drive, a thumb drive, or a flash memory that is connected to data processing system 200.

5 In some instances, computer readable storage media 224 may not be removable from data processing system 200. In these illustrative examples, computer readable storage media 224 is a non-transitory computer readable storage medium.

[0036] Alternatively, program code 218 may be transferred to
10 data processing system 200 from computer readable media 220 using computer readable signal media 226. Computer readable signal media 226 may be, for example, a propagated data signal containing program code 218. For example computer readable signal media 226 may be an electro-magnetic signal, an optical
15 signal, and/or any other suitable type of signal. These signals may be transmitted over communications links, such as wireless communications links, optical fiber cable, coaxial cable, a wire, and/or any other suitable type of communications link. In other words, the communications link and/or the connection may
20 be physical or wireless in the illustrative examples.

[0037] In some illustrative embodiments, program code 218 may be downloaded over a network to persistent storage 208 from another device or data processing system through computer readable signal media 226 for use within data processing system
25 200. For instance, program code stored in a computer readable storage medium in a server data processing system may be downloaded over a network from the server to data processing system 200. The data processing system providing program code 218 may be a server computer, a client computer, or some other
30 device capable of storing and transmitting program code 218.

[0038] The different components illustrated for data processing system 200 are not meant to provide architectural limitations to the manner in which different embodiments may be implemented. The different illustrative embodiments may be

implemented in a data processing system including components in addition to or in place of those illustrated for data processing system 200. Other components shown in Figure 2 can be varied from the illustrative examples shown. The different embodiments
5 may be implemented using any hardware device or system capable of executing program code. As one example, the data processing system may include organic components integrated with inorganic components and/or may be comprised entirely of organic components excluding a human being. For example, a storage
10 device may be comprised of an organic semiconductor.

[0039] As another example, a storage device in data processing system 200 is any hardware apparatus that may store data. Memory 206, persistent storage 208 and computer readable media 220 are examples of storage devices in a tangible form.

15 [0040] In another example, a bus system may be used to implement communications fabric 202 and may be comprised of one or more buses, such as a system bus or an input/output bus. Of course, the bus system may be implemented using any suitable type of architecture that provides for a transfer of data
20 between different components or devices attached to the bus system. Additionally, a communications unit may include one or more devices used to transmit and receive data, such as a modem or a network adapter. Further, a memory may be, for example, memory 206 or a cache such as found in an interface and memory
25 controller hub that may be present in communications fabric 202.

[0041] As used herein, the phrase "at least one of", when used with a list of items, means that different combinations of one or more of the items may be used and only one of each item in the list may be needed. For example, "at least one of item
30 A, item B, and item C" may include, for example, without limitation, item A or item A and item B. This example also may include item A, item B, and item C or item B and item C.

[0042] As used herein, when a first component is connected to a second component, the first component may be connected to the

second component without any additional components. The first component also may be connected to the second component by one or more other components. For example, one electronic device may be connected to another electronic device without any additional electronic devices between the first electronic device and the second electronic device. In some cases, another electronic device may be present between the two electronic devices connected to each other.

[0043] The different advantageous embodiments recognize and take into account that current systems for updating tool process parameters require the tools to be checked into a remotely located tool crib and updated manually when process parameters change. These updates occur locally when the tool is idle or inactive and placed into storage at the tool crib until needed again or until updates are complete. Remote updates may be sent to the tool crib, but the tool does not receive the update until it is checked into the crib after operations are finished whereupon the tool must be manually updated by a technician.

[0044] The different advantageous embodiments further recognize and take into account that current methods for monitoring tool processes require human monitoring of an individual tool for visual inspection of results or tool parameter performance, for example. Some limited computer monitoring may be provided, with the results of tool processes analyzed after the processes are complete and the information has been downloaded or manually retrieved for analysis.

[0045] Thus, the different advantageous embodiments provide a system comprising a wireless network, a number of tools, and a central server. The number of tools is connected to the wireless network. The number of tools is configured to perform a number of tasks and collect process data while performing the number of tasks. The central server is connected to the wireless network. The central server is configured to monitor the number of tools performing the number of tasks and receive

the process data collected by the number of tools over the wireless network.

[0046] The different advantageous embodiments further provide a method for tool management. A number of tools is implemented in a wireless network. Tool processes for the number of tools is monitored at a central location using the wireless network. Process data is received at the central location for the number of tools during the tool processes using the wireless network.

[0047] The different advantageous embodiments further provide a method for tool process updates. A tool mechanism is controlled to perform a task using a number of process parameters. Process data is collected during task performance. The process data collected is sent to a remote location.

[0048] With reference now to **Figure 3**, an illustration of a tool is depicted in accordance with an advantageous embodiment. Tool 300 is an illustrative example of one implementation of tool 122 in **Figure 1**.

[0049] Tool 300 includes control case 302, tool mechanism 304, housing 306, and power supply 308. Control case 302 is the processing and control center for tool 300. Tool mechanism 304 is the mechanical component of tool 300 controlled by components of control case 302. Tool mechanism 304 may be, for example, without limitation, drilling mechanism 310, fastening mechanism 312, and/or any other suitable mechanism for performing tool operations.

[0050] Housing 306 may encase tool mechanism 304 and/or control case 302 to contain components of tool 300 in a single implementation, for example. Power supply 308 may include, without limitation, factory-supplied alternating current (AC) or direct current (DC) electric power, a battery, ultra-capacitor, fuel cell, gas powered generator, photo cells, and/or any other suitable electrical power connected to the source by hardwire or wireless connection. In an illustrative embodiment, tool 300 may be powered by power 308 using a cable, for example.

[0051] Control case 302 includes data processing system 314, database 316, sensor system 318, gateway 320, and controller 322. Data processing system 314 may be implemented using data processing system 200 in Figure 2, for example. Data processing system 314 includes data manager 324. Data manager 324 receives collected data from sensor system 318 and manages transmission of the collected data to a centralized location, such as back office 104 in Figure 1, for example. Data manager 324 also receives data from the centralized location and uses the data received to update process parameters and network parameters for tool 300.

[0052] Database 316 includes process parameters 326 and security certificates 328, in these illustrative examples. Process parameters 326 may be received by data processing system 314 from a centralized location, such as back office 104 in Figure 1, and stored in database 316 for use by controller 322 in operation of tool 300. Process parameters 326 may be, for example, without limitation, an amount of torque applied to install a fastener, a degree of acceptable vibration in drilling operations, a size of a hole for drilling operations, a feed rate of a cutter into a material for drilling and milling operations, spindle speed, pilot hole formation, countersink depth, and/or any other suitable process parameters for tool 300.

[0053] Security certificates 328 provide a current authentication frame for gateway 320 to authenticate with the network, such as network 102 in Figure 1. Authentication with the network is controlled by an authentication control server on the network, such as network 102 in Figure 1. The certificate is sent to the network via access points, such as number of access points 118 in Figure 1, for example.

[0054] Controller 322 operates to control tool mechanism 304 of tool 300. Controller 322 may include tool operation process 330, for example. Tool operation process 330 may process

instructions and/or process parameters, such as process parameters 326, to use in controlling operation of tool mechanism 304.

[0055] Sensor system 318 collects process data 332 during operation of tool 300. Sensory system 318 may include, without limitation, number of cameras 334, video server 336, image processor 338, and/or any other suitable sensor. Other suitable sensors may include, for example, without limitation, accelerometers to collect vibration, thermocouples to detect temperature, strain gauges to detect flexion, humidity sensors, hole probes, tachometers, and Hall effect sensors. Number of cameras 334 may include, for example, without limitation, a color camera, a black and white camera, a digital camera, an infrared camera, a video camera, and/or any other suitable camera. Video server 336 may be any type of computer device dedicated to processing video images from number of cameras 334, for example. Image processor 338 gathers luminance and chrominance information from individual pixels generated for an image by number of cameras 334 and uses the information to computer the correct color and brightness values for each pixels using a number of algorithms. Luminance is a photometric measure of the luminous intensity per unit area of light traveling in a given direction. Luminance describes the amount of light that passes through or is emitted from a particular area, and falls within a given solid angle. Chrominance conveys the color information for an area or object. Video server 336 and/or image processor 338 may generate image data, for example. Image data 340 is an illustrative example of one implementation of process data 332.

[0056] Data manager 324 receives process data 332 from sensor system 318 and uses process data 332 to generate process reports 342. Process reports 342 may be transmitted by data processing system 314 to a centralized server using gateway 320. Gateway 320 is an illustrative example of one implementation of gateway

126 in **Figure 1**. Process reports 342 may be analyzed by a number of applications at the centralized server to determine whether process parameters 326 require updates.

[0057] In some advantageous embodiments, gateway 320 may also
5 manage network authentication for tool 300 using security certificates 328. Data manager 324 may generate security report 344 when authentication issues are detected by gateway 320, for example. Further, data integrity is addressed on the network media to prevent collection of erroneous data or to prevent
10 valuable data loss.

[0058] Data manager 324 also receives process updates 346 and security updates 348 from the centralized location, such as back office 104 in **Figure 1**. Process updates 346 may be used to update process parameters 326 in database 316. Process updates
15 346 may be received while tool 300 is in operation. Controller 322 may detect updated process parameters and adjust tool operation process 330 accordingly during operation of tool 300, for example.

[0059] The illustration of tool 300 in **Figure 3** is not meant
20 to imply physical or architectural limitations to the manner in which different advantageous embodiments may be implemented. Other components in addition and/or in place of the ones illustrated may be used. Some components may be unnecessary in some advantageous embodiments. Also, the blocks are presented
25 to illustrate some functional components. One or more of these blocks may be combined and/or divided into different blocks when implemented in different advantageous embodiments.

[0060] With reference now to **Figure 4**, an illustration of a network enabled tool environment is depicted in accordance with
30 an advantageous embodiment. Network enabled tool environment 400 is an illustrative example of one implementation of smart tool environment 100 in **Figure 1**.

[0061] Factory 401 is an illustrative example of factory 106 in **Figure 1**. Factory 401 includes drill 402. Drill 402 is an

illustrative example of one implementation of number of tools 120 in Figure 1. Drill 402 includes control case 404 and drilling mechanism 406. Control case 404 is an illustrative example of one implementation of control case 302 in Figure 3.

5 Drilling mechanism 406 is an illustrative example of one implementation of tool mechanism 304 in Figure 3.

[0062] Drill 402 may be used to drill holes in workpiece 408, for example. Factory 401 includes access point 410, which provides network access to drill 402, in this illustrative
10 example. Access point 410 is an illustrative example of one implementation of number of access points 118 in Figure 1. Control case 404 may interact with back office 412 over network 414 using access point 410, for example. Control case 404 collects process data during operation of drill 402 and sends
15 the process data collected to back office 412 using wireless network 414.

[0063] User interface 416 is connected to central data server 418, in this illustrative example. User 420 may use user interface 416 to access the process data collected and sent by
20 drill 402. User 420 and/or central data server 418 may analyze the process data to determine whether process updates are needed for control case 404 of drill 402. If process updates are needed, the process updates may be generated at central data server 418 and sent wirelessly over network 414 to control case
25 404 of drill 402. These process updates may be received by drill 402 while drill 402 is in operation within factory 401, for example.

[0064] The illustration of network enabled tool environment 400 in Figure 4 is not meant to imply physical or architectural
30 limitations to the manner in which different advantageous embodiments may be implemented. Other components in addition and/or in place of the ones illustrated may be used. Some components may be unnecessary in some advantageous embodiments. Also, the blocks are presented to illustrate some functional

components. One or more of these blocks may be combined and/or divided into different blocks when implemented in different advantageous embodiments.

5 [0065] With reference now to **Figure 5**, an illustration of a flowchart illustrating a process for tool management is depicted in accordance with an advantageous embodiment. The process in **Figure 5** may be implemented by a component such as back office 104 in **Figure 1**, for example.

10 [0066] The process begins by implementing a number of tools in an enterprise network architecture (operation 502). Tools on the network may be nodes that are available to interact with other network endpoints, such as data integration appliances, enterprise process middleware, and database software, for example. The number of tools may perform the role of data 15 server or behave as a client to centralized configuration management services. The process monitors tool processes using a network (operation 504). The process may monitor tool processes over a wireless network, such as network 102 in **Figure 1**, for example. The process collects process data from the 20 number of tools using the network (operation 506). The process data may be collected by sensor systems implemented on the tools, such as sensor system 318 of tool 300 in **Figure 3**, for example.

[0067] The process analyzes the process data from the number 25 of tools (operation 508). The process data may be analyzed by a centralized server, such as central data server 108 in **Figure 1**, and/or by a user accessing the process data on the centralized server, such as user 420 in **Figure 4**. The process determines whether a process parameter update is needed (operation 510). A 30 process parameter update may be needed if process data indicates that a tool is not operating within a certain specified parameter, or a tool is not operating to produce a result within a certain specified parameter, for example.

[0068] If a determination is made that a process parameter update is not needed, the process returns to operation 504. If a determination is made that a process parameter update is needed, the process sends a number of process parameter updates
5 to the number of tools using the network (operation 512), and then returns to operation 504.

[0069] The process in **Figure 5** may iteratively repeat as long as a tool is in operation, for example. The process may terminate when a tool is idle or inactive for a given period of
10 time, for example.

[0070] With reference now to **Figure 6**, an illustration of a flowchart illustrating a process for tool process updates is depicted in accordance with an advantageous embodiment. The process in **Figure 6** may be implemented by a component such as
15 number of tools 120 in **Figure 1** and/or tool 300 in **Figure 3**.

[0071] The process begins by controlling a tool mechanism to perform a task using a number of process parameters (operation 602). A tool mechanism may be, for example, without limitation, a drill bit or fastener mechanism. The task performed may be,
20 for example, without limitation, drilling a hole, fastening one component to another component, and/or any other suitable task. The process parameters may provide specified parameters for tool operation and/or for results from tool operation, for example.

[0072] The process collects process data during task
25 performance (operation 604). The process data may be, for example, without limitation, visual data of the tool process and/or tool process results, such as the size of a hole drilled, for example. The process data may also be information about torque expended during a fastening operation by the tool, for
30 example.

[0073] The process sends the process data collected to a remote location (operation 606). The remote location may be, for example, back office 104 in **Figure 1**. The process may send

the process data over a wireless network, such as network 102 in Figure 1, for example.

5 [0074] The process determines whether a process update is received (operation 608). A process update may be received over the wireless network from a centralized, remote location, such as back office 104 in Figure 1, for example. A process update may be received as a result of operation 512 of Figure 5, for example. If a determination is made that a process update is not received, the process returns to operation 604.

10 [0075] If a determination is made that a process update is received, the process updates the number of process parameters using the process update received to form new process parameters (operation 610). The process then controls the tool mechanism to perform the task using the new process parameters (operation 15 612), with the process terminating thereafter.

[0076] The process in Figure 6 may iteratively repeat as long as the tool is in operation, for example, receiving a number of process updates as needed during operation of the tool.

20 [0077] The flowcharts and block diagrams in the different depicted embodiments illustrate the architecture, functionality, and operation of some possible implementations of apparatus, methods and computer program products. In this regard, each block in the flowchart or block diagrams may represent a module, segment, or portion of computer usable or readable program code, 25 which comprises one or more executable instructions for implementing the specified function or functions. In some alternative implementations, the function or functions noted in the block may occur out of the order noted in the figures. For example, in some cases, two blocks shown in succession may be 30 executed substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved.

[0078] The different advantageous embodiments can take the form of an entirely hardware embodiment, an entirely software

embodiment, or an embodiment containing both hardware and software elements. Some embodiments are implemented in software, which includes but is not limited to forms, such as, for example, firmware, resident software, and microcode.

5 [0079] Furthermore, the different embodiments can take the form of a computer program product accessible from a computer usable or computer readable medium providing program code for use by or in connection with a computer or any device or system that executes instructions. For the purposes of this

10 disclosure, a computer usable or computer readable medium can generally be any tangible apparatus that can contain, store, communicate, propagate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device.

15 [0080] The computer usable or computer readable medium can be, for example, without limitation an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, or a propagation medium. Non limiting examples of a computer readable medium include a semiconductor or solid state memory, 20 magnetic tape, a removable computer diskette, a random access memory (RAM), a read-only memory (ROM), a rigid magnetic disk, and an optical disk. Optical disks may include compact disk - read only memory (CD-ROM), compact disk - read/write (CD-R/W) and DVD.

25 [0081] Further, a computer usable or computer readable medium may contain or store a computer readable or usable program code such that when the computer readable or usable program code is executed on a computer, the execution of this computer readable or usable program code causes the computer to transmit another 30 computer readable or usable program code over a communications link. This communications link may use a medium that is, for example without limitation, physical or wireless.

[0082] A data processing system suitable for storing and/or executing computer readable or computer usable program code will

include one or more processors coupled directly or indirectly to memory elements through a communications fabric, such as a system bus. The memory elements may include local memory employed during actual execution of the program code, bulk
5 storage, and cache memories which provide temporary storage of at least some computer readable or computer usable program code to reduce the number of times code may be retrieved from bulk storage during execution of the code.

[0083] Input/output or I/O devices can be coupled to the
10 system either directly or through intervening I/O controllers. These devices may include, for example, without limitation to keyboards, touch screen displays, and pointing devices. Different communications adapters may also be coupled to the system to enable the data processing system to become coupled to
15 other data processing systems or remote printers or storage devices through intervening private or public networks. Non-limiting examples are modems and network adapters are just a few of the currently available types of communications adapters.

[0084] The different advantageous embodiments recognize and
20 take into account that current systems for updating tool process parameters require the tools to be checked into a remotely located tool crib and updated manually when process parameters change. These updates occur locally when the tool is idle or inactive and placed into storage at the tool crib until needed
25 again or until updates are complete. Remote updates may be sent to the tool crib, but the tool does not receive the update until it is checked into the crib after operations are finished.

[0085] The different advantageous embodiments further
recognize and take into account that current methods for
30 monitoring tool processes require human monitoring of an individual tool for visual inspection of results or tool parameter performance, for example. Some limited computer monitoring may be provided, with the results of tool processes

analyzed after the processes are complete and the information has been downloaded or manually retrieved for analysis.

5 [0086] Thus, the different advantageous embodiments provide an enterprise-level network architecture for deployment of numerous smart tools, tools that electronically control process parameters and are equipped with sensors to report the resulting process parameters, to existing production programs. This system creates bi-directional communication between tools and a centralized administration system for uniform updates to process parameters and monitoring of tool processes.

10 [0087] The description of the different advantageous embodiments has been presented for purposes of illustration and description, and is not intended to be exhaustive or limited to the embodiments in the form disclosed. Many modifications and variations will be apparent to those of ordinary skill in the art. Further, different advantageous embodiments may provide different advantages as compared to other advantageous embodiments. The embodiment or embodiments selected are chosen and described in order to best explain the principles of the
15 embodiments, the practical application, and to enable others of ordinary skill in the art to understand the disclosure for various embodiments with various modifications as are suited to the particular use contemplated.

CLAIMS:

What is claimed is:

1. A system comprising:

an enterprise-level network architecture including a communication system;

a number of tools (120) in communication with the communication system and configured to perform a number of tasks using a number of process parameters (326), and collect process data while performing the number of tasks; wherein

the number of tools (120) further comprises a sensor system (318) configured to collect the process data while the number of tools (120) perform the number of tasks; and

a central server (108) in communication with the communication system and configured to:

provide the process parameters (326) for the number of tools (120);

monitor the number of tools (120) as the number of tools (120) perform the number of tasks;

receive the process data collected by the number of tools (120) as the number of tools (120) perform the number of tasks using the communication system;

analyze the process data received from the number of tools (120) as the number of tools (120) perform the number of tasks; and

determine whether a process parameter update is needed while the number of tools (120) perform the number of tasks;

wherein the central server (108) is configured to send a number of process parameter updates to the number of tools (120) using the communication system; and

wherein the number of tools (120) further comprises a data processing system (124) configured to receive the number of process parameter updates from the central server (108) while the number of tools (120) perform the number of tasks.

2. The system of claim 1, wherein the data processing system (124) is configured to send the process data to the central server (108).

3. The system of any preceding claim, wherein the number of tools (120) is selected from at least one of a contour drilling machine, a drill motor, and a fastener.

4. A method for tool management within an enterprise-level network architecture including a communication system, the method comprising:

monitoring tool processes for a number of tools (120) in communication with the communication system as the number of tools (120) perform a number of tasks, wherein the number of tools (120) collect process data while performing the number of tasks, and wherein the monitoring is performed at a central location (104) using the communication system;

providing process parameters (326) for the number of tools (120);

receiving the process data for the number of tools (120) during the tool processes at the central location using the communication system; wherein

the number of tools (120) includes a number of sensor systems (318), and wherein the process data (332)

is collected by the number of tools (120) using the number of sensor systems (318) while the number of tools (120) perform the number of tasks;

analyzing the process data (332) collected from the number of tools (120) during the tool processes;

determining whether a process parameter update is needed for the number of tools (120) while the number of tools (120) perform the number of tasks; and

responsive to a determination that the process parameter update is needed, sending a number of process parameter updates to the number of tools (120) from the central location (104) using the communication system, wherein the number of process parameter updates is sent to the number of tools (120) while the number of tools are in operation.

5. The method of claim 4, wherein the process parameter update is needed if the process data (332) indicates that a tool (300) in the number of tools (120) is not operating within a given parameter.

6. The method of claim 4 further comprising:

determining whether a process update (346) is received at the number of tools (120); and

responsive to a determination that the process update is received, updating a number of process parameters (326) at the number of tools (120).

7. The method of claim 6 further comprising:

controlling a tool mechanism (300) to perform a task using the number of updated process parameters (326);

collecting process data (332) during task performance; and

sending the process data (332) collected to the central location (104) using the communication system.

8. The method of claims 4-7, wherein the number of tools (120) is selected from at least one of a contour drilling machine (130), a drill motor (130), and a fastener (134), and wherein the number of tools is a number of nodes on the network (102).

9. The system of claim 1, wherein the enterprise-level network architecture further includes an authentication control server (102), and wherein the system further comprises: a number of access points (118) providing communication between the number of tools (120) and the authentication control server (102), wherein a number of security certificates (328) are each associated with each of the number of tools (120), and wherein the authentication control server (102) is configured to authenticate each of the number of tools (120) using the respective security certificate (328).

10. The method of claim 4, wherein the enterprise-level network architecture further includes an authentication control server (102) in communication with the communication system, and wherein the method further comprises: providing communication between each of the number of tools (120) and the authentication control server (102) using a number of access points (118), wherein a number of security certificates (328) are each associated with each of the number of tools (120); authenticating each of the number of tools (120) using the respective security certificate (328) and the authentication control server (102).