

(12) UK Patent

(19) GB

(11) 2593633

(13) B

(45) Date of B Publication

16.02.2022

(54) Title of the Invention: **Streaming data for analytics in process control systems**

(51) INT CL: **G05B 19/418** (2006.01) **G06F 16/25** (2019.01) **H04L 67/08** (2022.01)

(21) Application No: **2107630.2**

(22) Date of Filing: **25.09.2015**

Date Lodged: **28.05.2021**

(30) Priority Data:

(31) **14506863** (32) **06.10.2014** (33) **US**

(62) Divided from Application No
2016319.2 under section 15(9) of the Patents Act 1977

(43) Date of A Publication **29.09.2021**

(72) Inventor(s):

Terrence L Blevins
Mark John Nixon
Ken Beoughter
Daniel Dean Christensen
John Michael Lucas
Paul R Muston

(73) Proprietor(s):

Fisher-Rosemount Systems, Inc
1100 W. Louis Henna Blvd, Bldg 1,
Round Rock 78681, Texas, United States of America

(74) Agent and/or Address for Service:

Forresters IP LLP
Rutland House, 148 Edmund Street, BIRMINGHAM,
B3 2JA, United Kingdom

(56) Documents Cited:

EP 1965316 A1

(58) Field of Search:

As for published application 2593633 A viz:

INT CL **G05B, G06F, H04L**

Other: **WPI, EPODOC**

updated as appropriate

Additional Fields

Other: **None**

GB 2593633 B

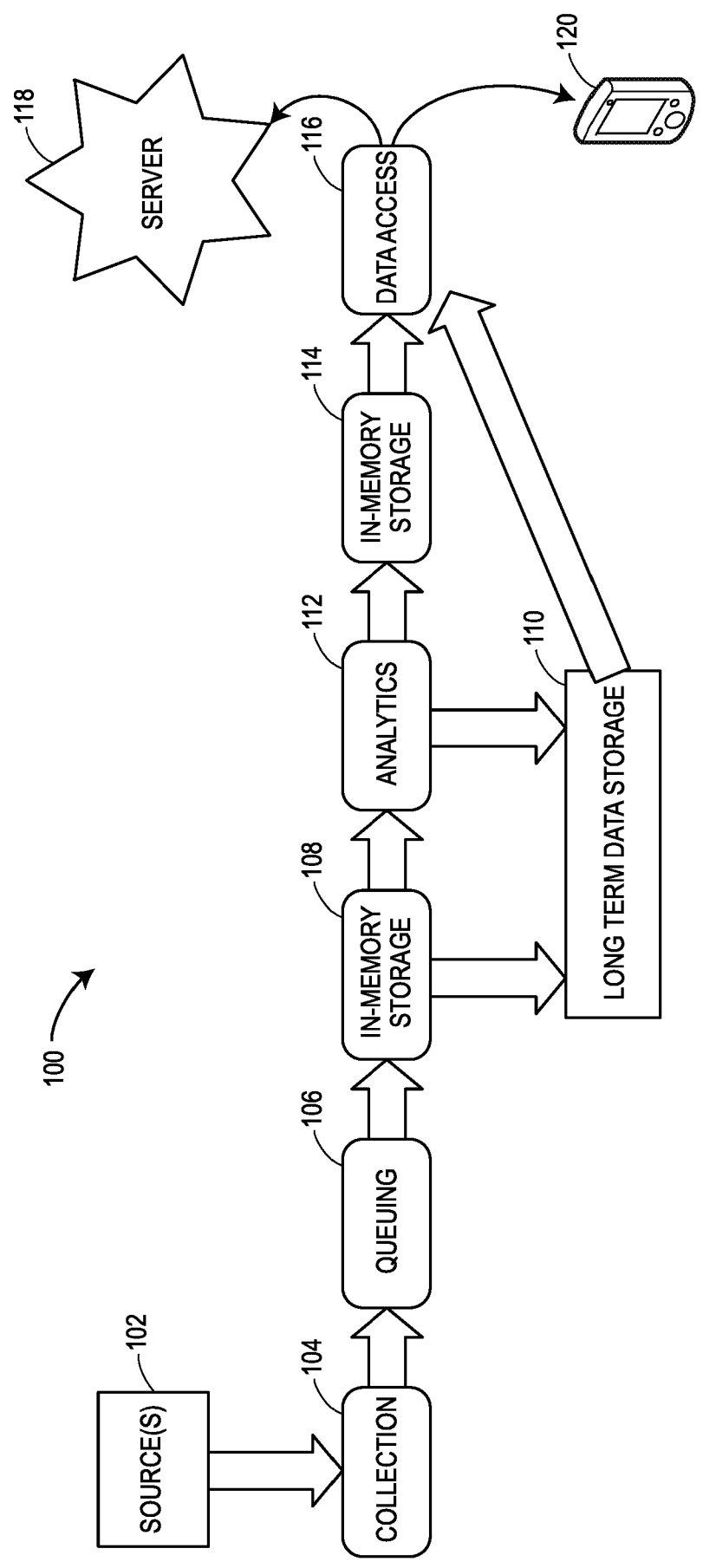


FIG. 1

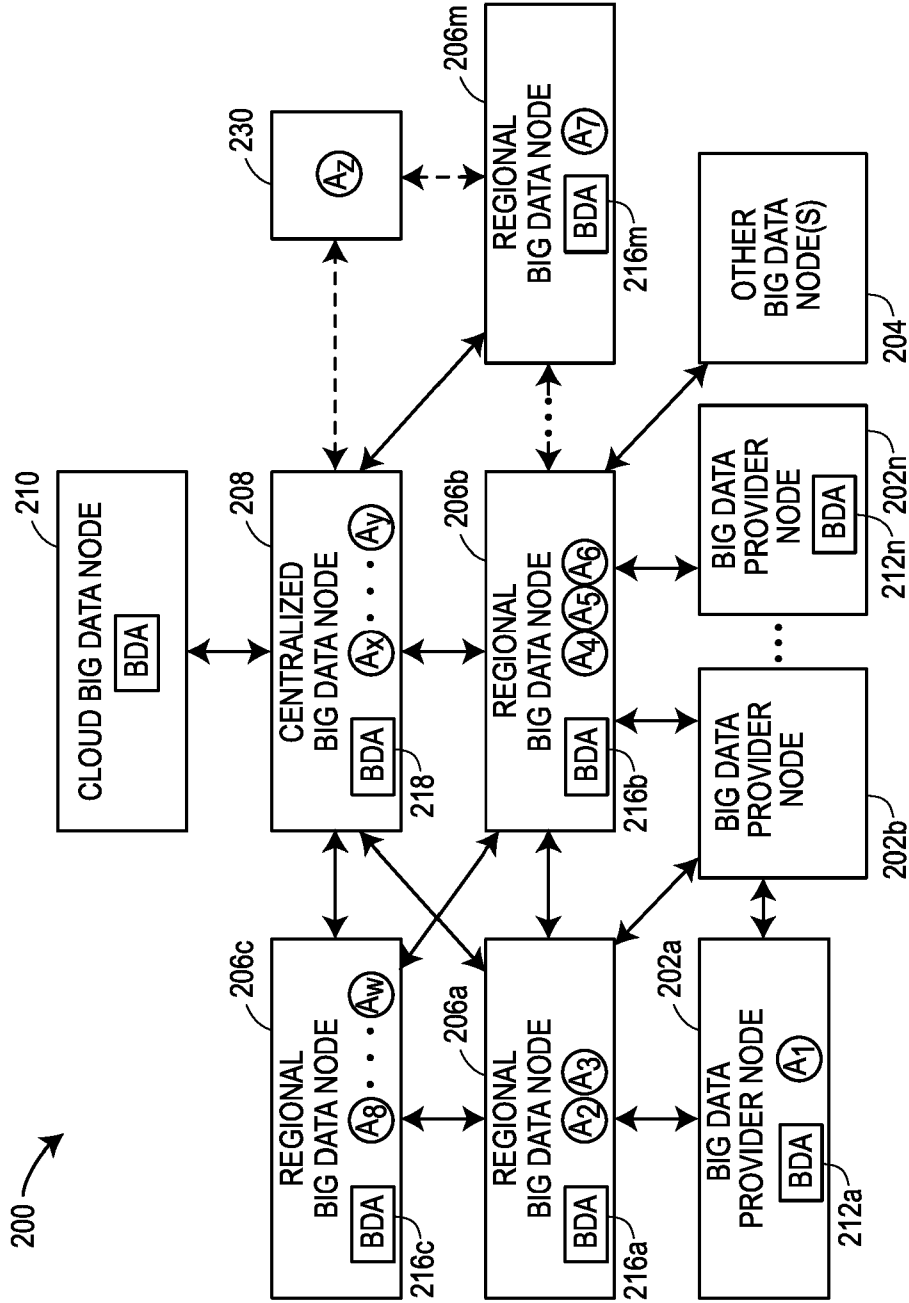


FIG. 2

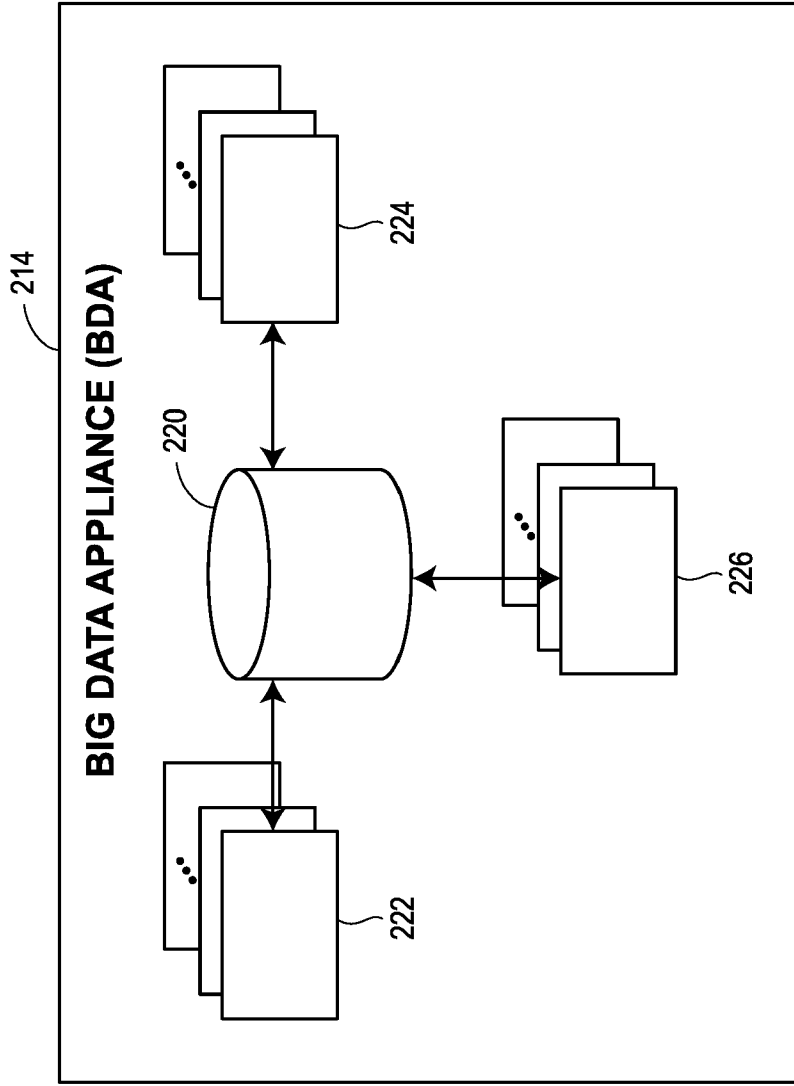


FIG. 3

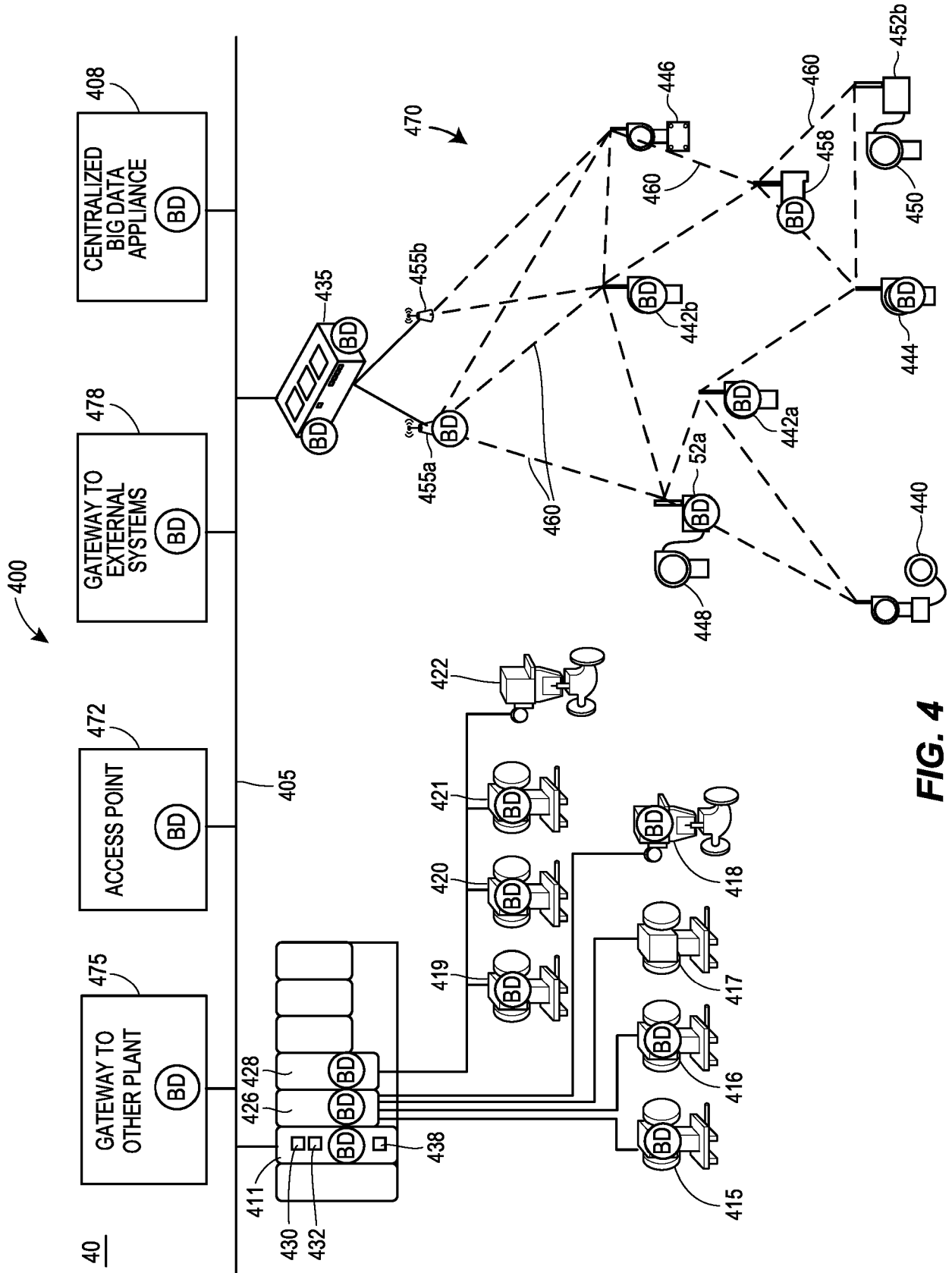


FIG. 4

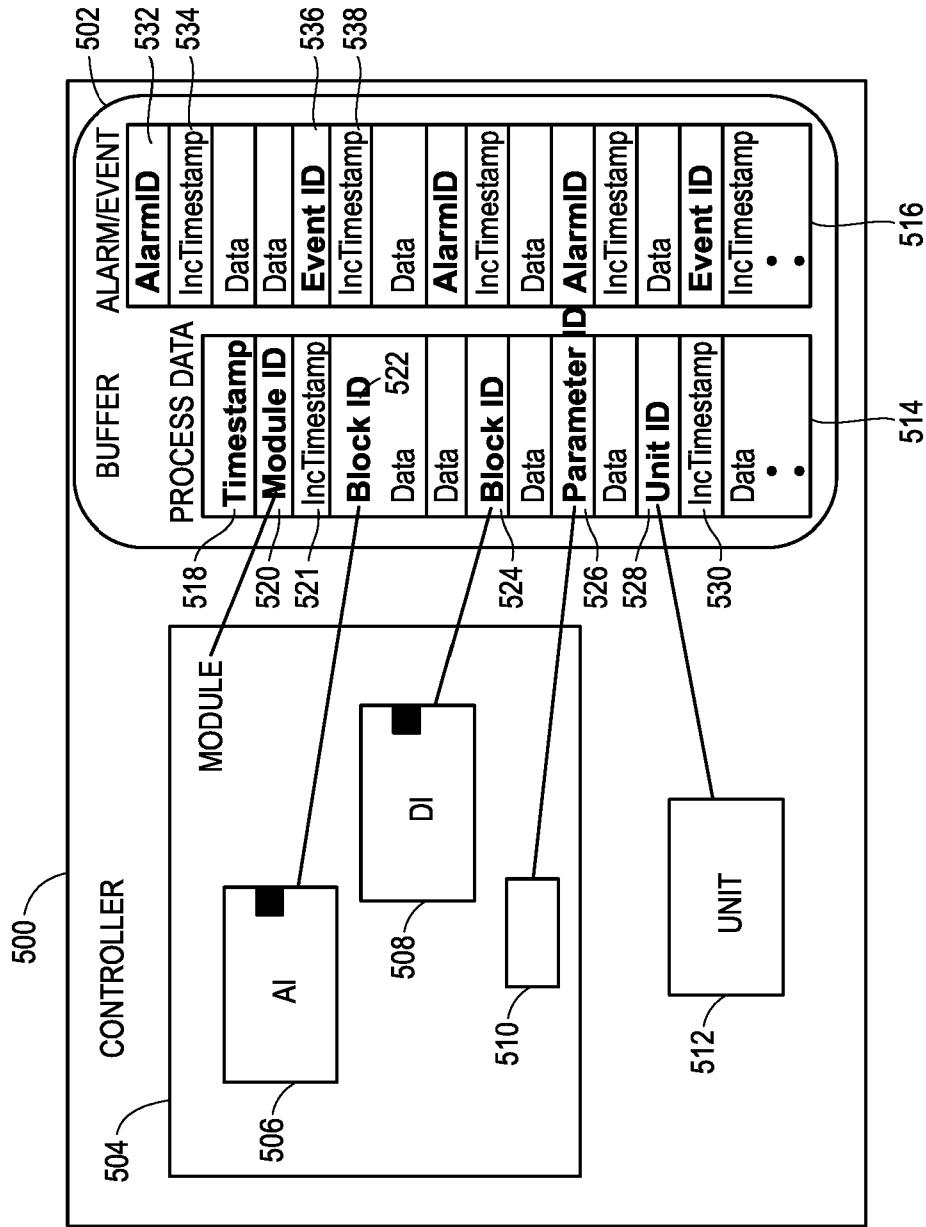


FIG. 5

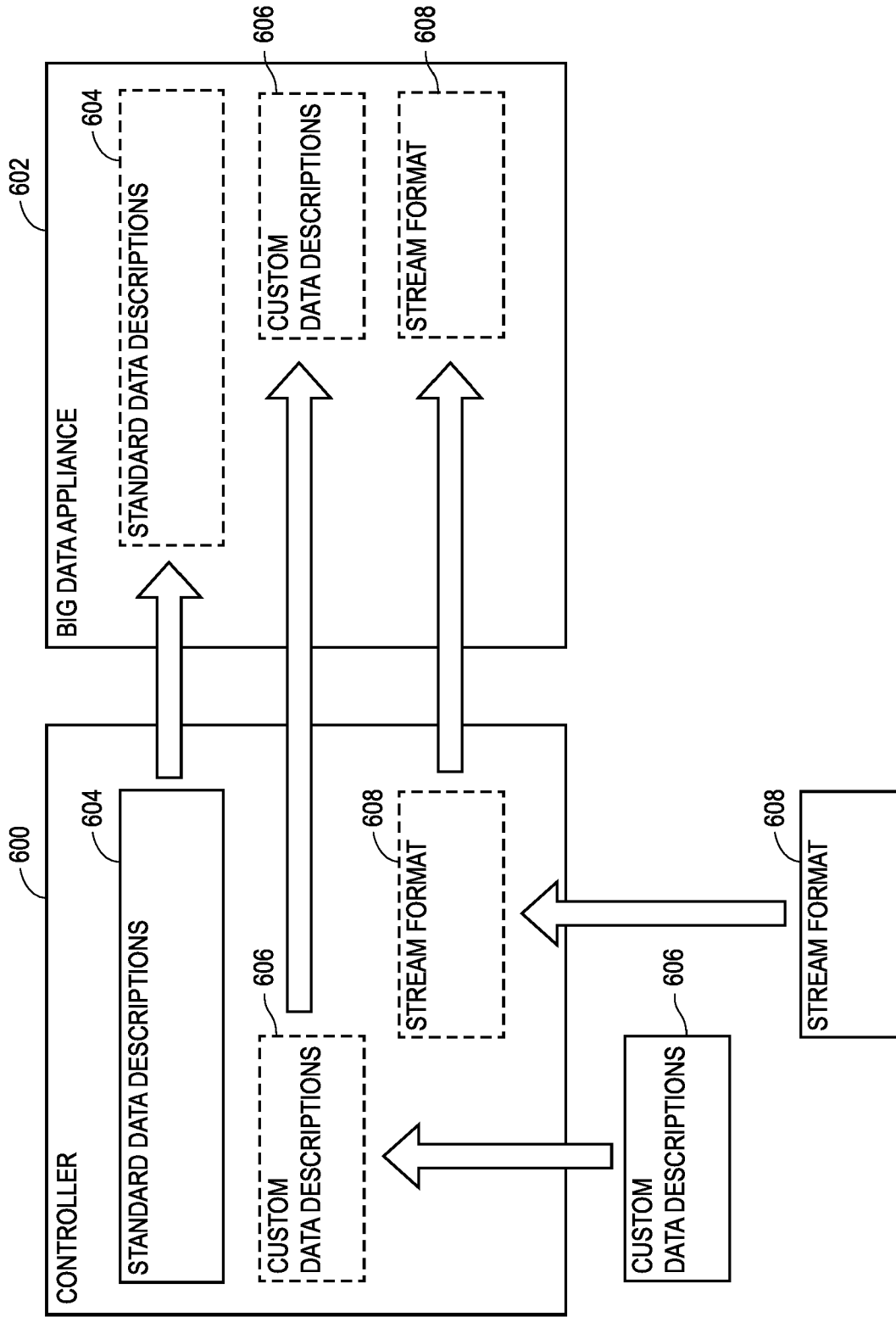


FIG. 6

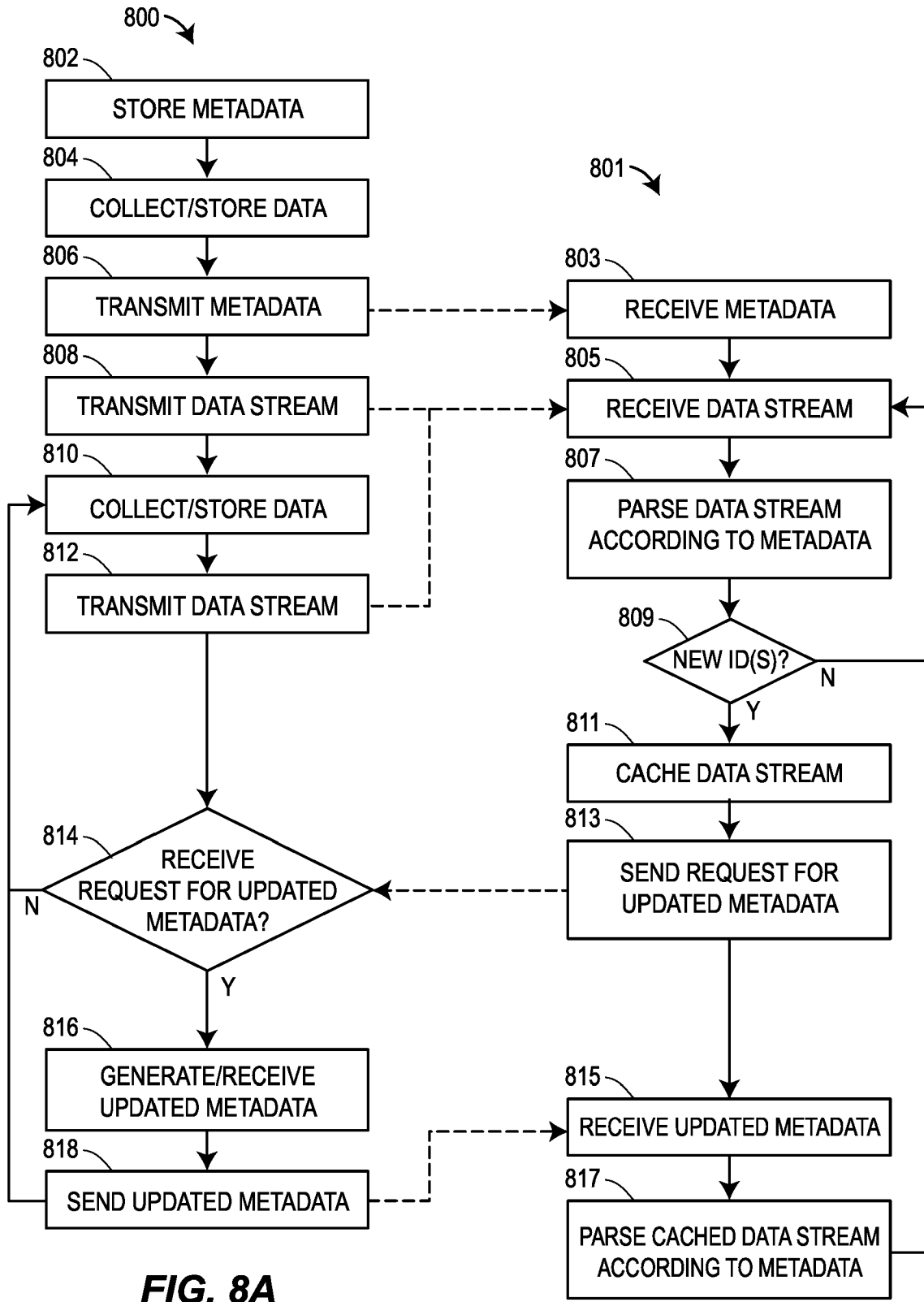


FIG. 8A

FIG. 8B

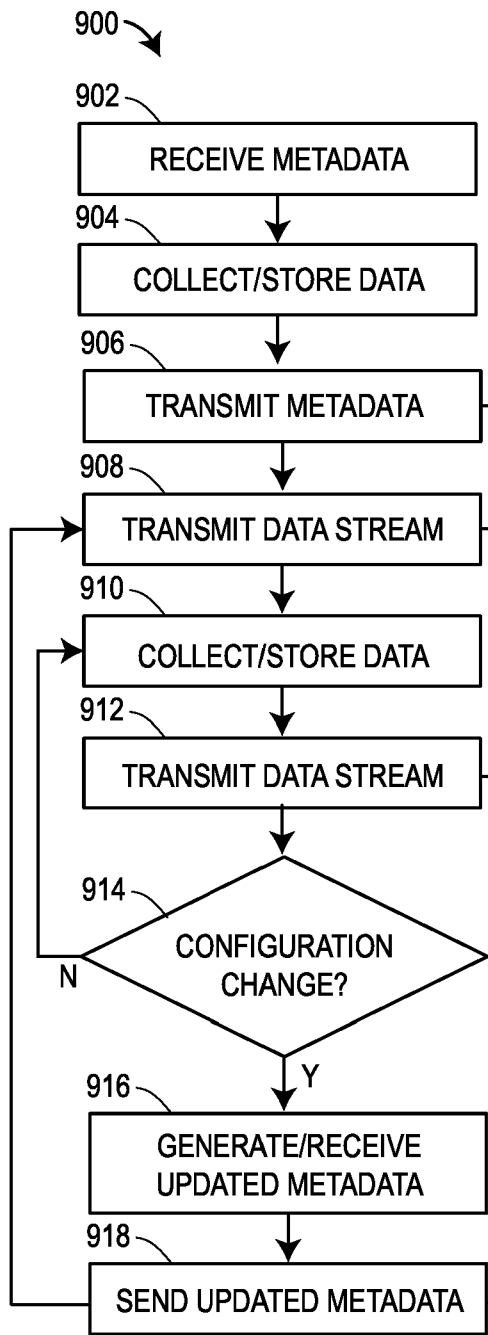


FIG. 9A

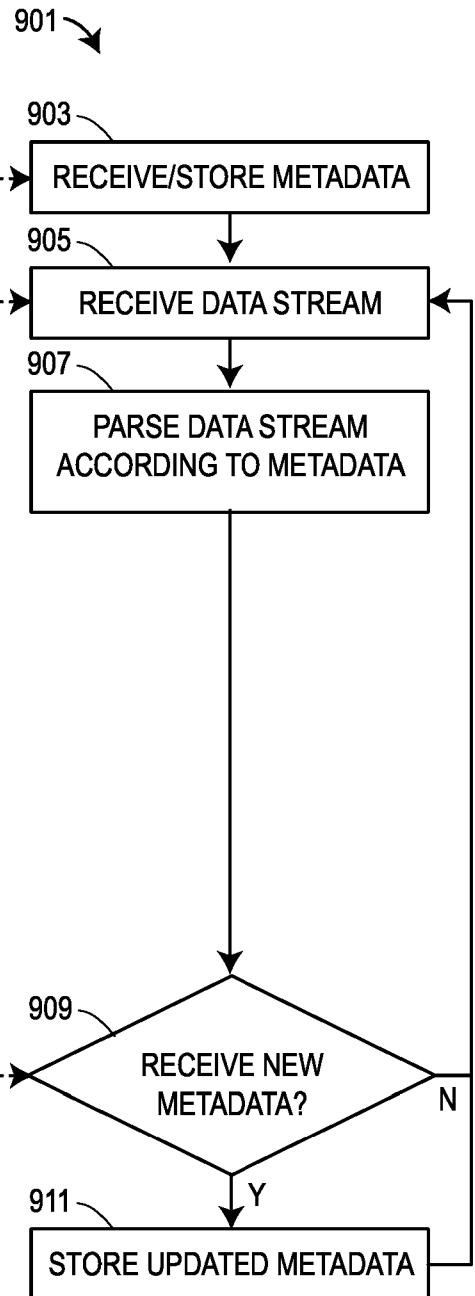


FIG. 9B

STREAMING DATA FOR ANALYTICS IN PROCESS CONTROL SYSTEMS

[0001] The present disclosure is related to U.S. Patent Publication No. US 2014-0250153 (Application No. 13/784,041), entitled "BIG DATA IN PROCESS CONTROL SYSTEMS" and filed March 3, 2013; U.S. Patent Publication No. US 2014-0280497 (Application No. 14/028,785), entitled "METHOD AND APPARATUS FOR CONTROLLING A PROCESS PLANT WITH LOCATION AWARE MOBILE CONTROL DEVICES" and filed on September 17, 2013; U.S. Patent Publication No. US 2014-0277656 (Application No. 14/174,413), entitled "COLLECTING AND DELIVERING DATA TO A BIG DATA MACHINE IN A PROCESS CONTROL SYSTEM" and filed February 6, 2014; U.S. Patent Publication No. US 2014-0277604 (Application No. 14/212,493), entitled "DISTRIBUTED BIG DATA IN A PROCESS CONTROL SYSTEM" and filed March 14, 2014; and U.S. Patent Publication No. US 2015-0261215 (Application No. 14/212,411), entitled "DETERMINING ASSOCIATIONS AND ALIGNMENTS OF PROCESS ELEMENTS AND MEASUREMENTS IN A PROCESS" and filed March 14, 2014, the entire disclosures of each of which are hereby expressly incorporated by reference.

[0002] Additionally, the present disclosure is related to U.S. Patent Publication No. US 2016-0098021 (Application No. 14/507,188) (Attorney Docket No. 06005/593079), entitled "REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS" and filed concurrently herewith; and U.S. Patent Publication No. US 2016-0098647 (Application No. 14/507,252) (Attorney Docket 06005/593086), entitled "AUTOMATIC SIGNAL PROCESSING-BASED LEARNING" and filed concurrently herewith, the entire disclosures of each of which are hereby expressly incorporated by reference.

[0003] The present disclosure relates generally to process plants and to process control systems, and more particularly, to streaming process control data between devices in or related to a process control plant on a continuous and/or as-needed basis.

[0004] Distributed process control systems, like those used in chemical, petroleum, industrial or other process plants to manufacture, refine, transform, generate, or produce physical materials or products typically include one or more process controllers communicatively coupled to one or more field devices via analog, digital or combined analog/digital buses, or via a wireless communication link or network. The field devices, which may be, for example, valves, valve positioners, switches and transmitters (e.g., temperature, pressure, level and flow rate sensors), are located within the process environment and generally perform physical or process control functions such as opening or closing valves, measuring process and/or environmental parameters such as temperature or pressure, etc. to control one or more process executing within the process plant or system. Smart field devices, such as the field devices conforming to the well-known Fieldbus protocol may also perform control calculations, alarming functions, and other control functions commonly implemented within the controller. The process controllers, which are also typically located within the plant environment, receive signals indicative of process measurements made by the field devices and/or other information pertaining to the field devices and execute a controller application that runs, for example, different control modules which make process control decisions, generate control signals based on the received information and coordinate with the control modules or blocks being performed in the field devices, such as HART[®], WirelessHART[®], and FOUNDATION[®] Fieldbus field devices. The control modules in the controller send the control signals over the communication lines or links to the field devices to thereby

16 08 21

control the operation of at least a portion of the process plant or system. For example, the controllers and the field devices control at least a portion of a process being controlled by the process plant or system.

[0005] Information from the field devices and the controller is usually made available over a data highway or communication network to one or more other hardware devices, such as operator workstations, personal computers or computing devices, mobile computing devices, data historians, report generators, centralized databases, or other centralized administrative computing devices that are typically placed in control rooms or other locations away from the harsher plant environment. Each of these hardware devices typically is centralized across the process plant or across a portion of the process plant. These hardware devices run applications that may, for example, enable an operator to perform functions with respect to controlling a process and/or operating the process plant, such as changing settings of the process control routine, modifying the operation of the control modules within the controllers or the field devices, viewing the current state of the process, viewing alarms generated by field devices and controllers, simulating the operation of the process for the purpose of training personnel or testing the process control software, keeping and updating a configuration database, etc. The data highway utilized by the hardware devices, controllers and field devices may include a wired communication path, a wireless communication path, or a combination of wired and wireless communication paths.

[0006] As an example, the DeltaV™ control system, sold by Emerson Process Management(RTM), includes multiple applications stored within and executed by different devices located at diverse places within a process plant. A configuration application, which resides in one or more workstations or computing devices, enables users to create or change process control modules and download these process control modules via a data highway to dedicated distributed controllers. Typically, these control modules are made up of communicatively interconnected function blocks, which are objects in an object oriented programming protocol that perform functions within the control scheme based on inputs thereto and that provide outputs to other function blocks within the control scheme. The configuration application may also allow a configuration designer to create or change operator interfaces which are used by a viewing application to display data to an operator and to enable the operator to change settings, such as set points, within the process control routines. Each dedicated controller and, in some cases, one or more field devices, stores and executes a respective controller application that runs the control modules assigned and downloaded thereto to implement actual process control functionality. The viewing applications, which may be executed on one or more operator workstations (or on one or more remote computing devices in communicative connection with the operator workstations and the data highway), receive data from the controller application via the data highway and display this data to process control system designers, operators, or users using the user interfaces, and may provide any of a number of different views, such as an operator's view, an engineer's view, a technician's view, etc. A data historian application is typically stored in and executed by a data historian device that collects and stores some or all of the data provided across the data highway while a configuration database application may run in a still further computer attached to the data highway to store the current process control routine configuration and data associated therewith. Alternatively, the configuration database may be located in the same workstation as the configuration application.

[0007] In a process plant or process control system, data are generated and/or collected by field devices, controllers, sensors, function blocks, and the like. The generated and/or collected data may be stored in whole or in part, but generally are not available in real time throughout the process control system. Instead,

data are stored and/or processed locally (e.g., in a function block, such as a statistical processing block, of a field device or controller) and the resulting processed data are cached locally and/or transmitted to controllers and/or historians to be stored for later analysis or review. In order to be useful for real-time analysis, for example to predict the future occurrence or present existence of an abnormal condition or a fault in the process control system, the data generated by the many various devices, sensors, function blocks, etc. in the process control system must be available for analysis by expert systems and, for example, by machine learning systems and analysis components such as those described in co-pending application US 2016-0098647 (appl. No. 14/507,252) (attorney docket no. 06005/593086), entitled "AUTOMATIC SIGNAL PROCESSING-BASED LEARNING," and filed concurrently. In current systems, transporting large volumes of data over a network within the process plant requires significant network bandwidth, and processing, storing, and cataloging such large volumes of data requires significant configuration for any device to be able to use the data.

[0008] Simply put, the architecture of currently known process control plants and process control systems is strongly influenced by limited controller and device memory, communications bandwidth, and controller and device processor capability. For example, in currently known process control system architectures, the use of dynamic and static non-volatile memory in the controller is usually minimized or, at the least, managed carefully. As a result, during system configuration (e.g., *a priori*), a user typically must choose which data in the controller is to be archived or saved, the frequency at which it will be saved, and whether or not compression is used, and the controller is accordingly configured with this limited set of data rules. Consequently, data which could be useful in troubleshooting and process analysis is often not archived, and if it is collected, the useful information may have been lost due to data compression.

[0009] The limitations of currently known process plants and process control systems discussed above and other limitations may undesirably manifest themselves in the operation and optimization of process plants or process control systems, for instance, during plant operations, trouble shooting, and/or predictive modeling. For example, such limitations force cumbersome and lengthy work flows that must be performed in order to obtain historical data for troubleshooting, manually feeding the data into stand-alone, off-line signal processing tools, and manually supervising the analysis of the output of the signal processing tools to generate updated models. Even then, the troubleshooting results and models may be incomplete or not fully representative of the actual system, as the inputs to their generation rely on a particular operator's experience and knowledge.

[0010] "Big data" generally refers to a collection of one or more data sets that are so large or complex that traditional database management tools and/or data processing applications (e.g., relational databases and desktop statistic packages) are not able to manage the data sets within a tolerable amount of time. Typically, applications that use big data are transactional and end-user directed or focused. For example, web search engines, social media applications, marketing applications and retail applications may use and manipulate big data. Big data may be supported by a distributed database which allows the parallel processing capability of modern multi-process, multi-core servers to be fully utilized.

SUMMARY

[0011] Techniques, systems, apparatuses, and methods for streaming big data in a process plant are disclosed. Generally, said techniques, systems, apparatuses, and methods facilitate near real-time

movement of data between data sources and data recipients, allowing data to be consumed and analyzed as soon as requested. Data are formatted into a predefined stream structure, and the structure communicated as metadata to the recipient before, and separately, from the data in the stream, reducing network overhead, required bandwidth, and transmission times, and increasing network throughput and storage efficiency. Metadata communicated to the recipient define identifiers associated with the streamed data. When the transmitter of a data stream is configured to include new or different data in the data stream, the transmitter sends new metadata to the recipient before sending the newly formatted data stream. The streamed data include module and block data, parameter data, alarm data, event data, and/or any data capable of being described by metadata. The communication of alarm and event data with the parameter data facilitates storage of the alarm and event data in the same location as the parameter data, which, in turn, facilitates more complex analyses of the relationships between all of the data.

[0012] A method for communicating data in a process control system includes storing a first set of metadata in a device operable to transmit a data stream and storing, in the device operable to transmit the data stream, a set of data for transmitting as the data stream. The method includes transmitting the first set of metadata to a device operable to receive the data stream, and receiving the first set of metadata at the device operable to receive the data stream. Further, the method includes transmitting the data stream to the device operable to receive the data stream, and receiving the data stream at the device operable to receive the data stream, after receiving the first set of metadata. The method also includes parsing the received data stream, at the device operable to receive the data stream, according to the first set of metadata. In various implementations, the first set of metadata includes stream format description metadata describing the structure of the data stream and/or data description metadata describing the content of the data stream.

[0013] In an implementation, the method also includes identifying in the data stream, at the device operable to receive the data stream, an identifier that is not in the first metadata, caching the data stream at the device operable to receive the data stream, sending a request, from the device operable to receive the data stream to the device operable to send the data stream, to provide an updated set of metadata, and receiving the request to provide the updated metadata at the device operable to send the data stream. The method also includes sending the updated set of metadata from the device operable to send the data stream to the device operable to receive the data stream, receiving, at the device operable to receive the data stream, the updated set of metadata, and parsing the cached data stream according to the updated set of metadata.

[0014] In another implementation, the method includes recognizing, in the device operable to send the data stream, a configuration change, updating the metadata according to the recognized configuration change, and sending the updated metadata from the device operable to send the data stream to the device operable to receive the data stream. The method further includes receiving, at the device operable to receive the data stream, the updated set of metadata, sending, from the device operable to send the data stream to the device operable to receive the data stream, the data stream. In the implementation, the data stream is structured according to the updated set of metadata, and parsing the data stream, in the device operable to receive the data stream, according to the updated set of metadata.

[0015] A method of streaming data in a process control system includes storing a first set of metadata in a device operable to transmit a data stream, collecting a set of data for transmitting as the data stream, buffering the set of collected data, and transmitting the first set of metadata to a device operable to receive

the data stream. The method also includes transmitting the data stream to the device operable to receive the data stream. In the implementation, no further metadata are transmitted unless a trigger event causes the device operable to transmit the data stream to transmit additional metadata. In various implementations, the first set of metadata includes stream format description metadata describing the structure of the data stream and/or data description metadata describing the content of the data stream.

[0016] In an implementation of the method, the method also includes receiving a request for updated metadata, generating or downloading updated metadata, and sending updated metadata in response to the request for updated metadata. In another implementation of the method, the method includes recognizing a changed configuration parameter, generating or downloading updated metadata, and sending the updated metadata to the device operable to receive the data stream before sending a data stream according to the updated metadata.

[0017] A method for receiving a data stream in a process control system includes receiving a first set of metadata from a device operable to transmit the data stream, receiving the data stream after receiving the first set of metadata, parsing the received data stream according to the first set of metadata, and continuing to receive streamed data as long as the data stream can be parsed according to the first set of metadata. In an implementation of the method, the method also includes identifying in the data stream an ID that cannot be parsed according to the first set of metadata, caching the data stream, sending a request to the device operable to send the data stream to provide updated metadata, receiving the updated metadata, parsing the cached data stream according to the updated metadata, continuing to receive the data stream, and parsing the data stream according to the updated metadata. In another implementation of the method, the method also includes receiving updated metadata, continuing to receive the data stream, and parsing the data stream according to the updated metadata. In the implementation, the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the first set of metadata.

[0018] An apparatus in a process control system includes a processor, a memory communicatively coupled to the processor and storing a set of metadata, and a data source providing data to the apparatus. The apparatus also includes a queuing routine executing on the processor to buffer data received from the data source and a data streaming routine executing on the processor and cooperating with a communication device. The data streaming routine is operable to transmit the stored set of metadata to a receiving device, assemble the buffered data into a data stream according to the stored set of metadata, and transmit the data stream to the receiving device.

[0019] In an implementation of the apparatus, the apparatus is a controller and the data source is an input receiving data from one or more process control devices. In the implementation, the apparatus also includes one or more routines executing on the processor to control the process according at least in part to the received data, and a collection routine executing on the processor and collecting data received from the one or more process control devices.

[0020] In an implementation of the apparatus, the data streaming routine executing on the processor is also operable to receive a request for updated metadata, generate or download updated metadata, and send the updated metadata in response to the request for updated metadata. In another implementation of the apparatus, the apparatus also includes a metadata updating routine operable to receive or generate updated

metadata, and the data streaming routine executing on the processor is operable to send updated metadata in response to a detected change in a configuration of the apparatus or in response to the generation or reception of updated metadata, and continue sending the data stream. In the implementation, the data stream sent after the updated metadata are sent is structured according to the updated metadata.

[0021] An apparatus for receiving a stream of big data in a process control system includes a processor, a memory communicatively coupled to the processor, a non-transitory memory device for storing big data, and a receiver. The receiver operates to receive a set of metadata, receive a data stream, parse the data stream according to the set of metadata, process data according to the parsing, and continue to receive and process data as long as the data stream can be parsed according to the first set of metadata.

[0022] In an implementation of the apparatus, the apparatus is a big data appliance for storing big data, the non-transitory memory device comprises a high fidelity data storage device, and processing data comprises storing the data in the high fidelity data storage device.

[0023] In an implementation of the apparatus, the receiver is also operable to identify in the data stream an ID that cannot be parsed according to the set of metadata, cache the data stream, and send a request to provide updated metadata. The apparatus can receive the updated metadata, parse the cached data stream according to the updated metadata, and process the data according to the parsing. The apparatus can further continue to receive the data stream, parse the continued data stream according to the updated metadata, and process the data from the continued data stream.

[0024] In another implementation of the apparatus, the receiver is also operable to receive updated metadata, continue to receive the data stream, and parse the data stream according to the updated metadata. Therein, the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the set of metadata.

[0025] An aspect provides a method for receiving a data stream in a process control system, the method comprising: receiving a set of metadata from a device operable to transmit the data stream, the device being a controller communicatively connected to one or more process control devices and executing one or more control routines to control a process according at least in part to data received from the one or more process control devices; receiving, from the device, the data stream after receiving the set of metadata, the data stream including process control data comprising data received by the controller from the one or more process control devices and data indicative of a control signal generated by the one or more control routines; parsing the received data stream according to the set of metadata; and continuing to receive the data stream as long as the continued data stream can be parsed according to the set of metadata.

[0026] The set of metadata may be a first set of metadata, and the method may further comprise: identifying in the data stream an ID that cannot be parsed according to the first set of metadata; caching the data stream; sending a request to the device operable to send the data stream to provide updated metadata; receiving the updated metadata; parsing the cached data stream according to the updated metadata; continuing to receive the data stream; and parsing the continued data stream according to the updated metadata.

[0027] The set of metadata may be a first set of metadata, and the method may further comprise: receiving updated metadata; continuing to receive the data stream; and parsing the continued data stream

according to the updated metadata, wherein the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the first set of metadata.

[0028] The set of metadata may include stream format description metadata describing the structure of the data stream.

[0029] The set of metadata may include data description metadata describing the content of the data stream.

[0030] The method may be performed by a device comprising a big data appliance.

[0031] The method may further comprise performing an analysis on the set of data to generate analysis data.

[0032] The analysis data may be stored in a big data appliance.

[0033] The data stream may comprise a timestamp.

[0034] The data stream may comprise a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the set of data.

[0035] The data stream may comprise a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.

[0036] Another aspect provides an apparatus for receiving a stream of big data in a process control system, the apparatus comprising: a processor; a memory communicatively coupled to the processor; a non-transitory memory device; a receiver operable to: receive a set of metadata from a controller communicatively connected to one or more process control devices and executing one or more control routines to control a process according at least in part to data received from the one or more process control devices; receive a data stream from the controller after receiving the set of metadata, the data stream including process control data comprising data received by the controller from the one or more process control devices and data indicative of a control signal generated by the one or more control routines; parse the data stream according to the set of metadata; process data according to the parsing; continue to receive the data stream and parse the continued data stream according to the set of metadata as long as the continued data stream can be parsed according to the set of metadata.

[0037] The apparatus may be a big data appliance for storing big data, wherein the non-transitory memory device may comprise a high fidelity data storage device, and wherein the receiver may be further operable to store the processed data in the high fidelity data storage device.

[0038] The set of metadata may be a first set of metadata, and the receiver may be further operable to: identify in the data stream an ID that cannot be parsed according to the first set of metadata; cache the data stream; send a request to provide updated metadata; receive the updated metadata; parse the cached data stream according to the updated metadata; process the data according to the parsing; continue to receive the data stream; parse the continued data stream according to the updated metadata; and process the data from the continued data stream.

[0039] The set of metadata may be a first set of metadata, and the receiver may be further operable to: receive updated metadata; continue to receive the data stream; and parse the data stream according to the

updated metadata, wherein the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the first set of metadata.

[0040] The set of metadata may include stream format description metadata describing the structure of the data stream.

[0041] The set of metadata may include data description metadata describing the content of the data stream.

[0042] The apparatus may be operable to stream big data to a downstream device.

[0043] The apparatus may further comprise an analysis module operable to perform an analysis on the data in the data stream to generate analysis data.

[0044] The analysis data may be stored in the non-transitory memory device.

[0045] The data stream may comprise a timestamp.

[0046] The data stream may comprise a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the data stream.

[0047] The data stream may comprise a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.

[0048] Embodiments of the invention are described, by way of example only, with reference to the accompanying drawings, in which:

[0049] FIG. 1 is a block diagram of an example system for streaming process control data in a process plant or process control system;

[0050] FIG. 2 is a block diagram of an example big data network for a process plant or process control system which may support streaming of big data between devices;

[0051] FIG. 3 is a block diagram of an example big data appliance;

[0052] FIG. 4 is a block diagram of an example process plant or process control system in which data are streamed between devices in the process plant;

[0053] FIG. 5 is a block diagram of an example controller implementing data streaming according to the described methods;

[0054] FIG. 6 is a block diagram illustrating the flow of metadata in example components of a process control system implementing data streaming according to the described methods;

[0055] FIG. 7 is a block diagram illustrating the flow of data and metadata in example components of a process control system implementing data streaming;

[0056] FIGS. 8A and 8B are flow charts respectively depicting a method for streaming data to another device and a method for receiving a data stream, and collectively depicting the flow of data between two devices operating according to the respective methods; and

[0057] FIGS. 9A and 9B are flow charts respectively depicting an alternate method for streaming data to another device and an alternate method for receiving a data stream, and collectively depicting the flow of data between two devices operating according to the respective alternate methods.

[0058] Techniques, systems, and methods for streaming data and metadata in a process plant or process control system are disclosed herein. Generally, said techniques, systems, and methods facilitate movement of data between devices in or related to the process plant. More particularly, the techniques, systems, and methods facilitate streaming of various types of process control data, including I/O data, continuous data, event data, batch data, lab data, and data from analyzers and other components from and to various devices in the process plant, and to various mobile and/or external devices that may request such data. In fact, any data capable of being described by metadata may be streamed according to the data streaming techniques described herein. The techniques, systems, and methods additionally simplify the control system to stream the data without requiring complex configuration of the receiving devices. The data can be streamed in real time or on request. That is, in contrast with known process control signal processing techniques in which a requesting device (or user) obtains a *historical* signal feed from a process plant *a posteriori* and which requesting device must be configured to receive precisely the data that will be transmitted to it, the present techniques, systems, and methods allow data to be streamed *in real time* to historians, requesting devices, or other process control devices actively using the data to control the process plant. Thus, by using the techniques, systems, and methods disclosed herein, signal processing may be incorporated or integrated into operating, on-line process plants or systems (e.g., by positioning or locating various signal processing functions near to various signal sources within the process plant or system), *a priori* configuration requirements are reduced, and real-time analytics and learning may be implemented. Other additional advantages will be apparent to those reading the following disclosure.

[0059] As an example of an embodiment described herein in greater detail, data may be produced by various devices and systems (e.g., DCS systems, process control field devices, etc.) and collected via I/O devices, buses, wireless gateways, and the like. The collected data may be queued and streamed to long term data storage devices (e.g., historian devices), to one or more analytics devices performing real-time analysis and/or learning functions, to one or more requesting devices, and/or to one or more devices performing process control using the streamed data as input. Additionally, the collected data may be streamed, directly or from one of the long term data storage devices, to external devices via a web server, an API, etc.

[0060] Metadata of the collected and streamed data may be streamed separately from the streamed data. The metadata are generally transmitted in the stream prior to the data and, in any event, are transmitted to the receiving device before the receiving device may process and/or use the streamed data. The metadata describe the type and organization of the data in the data stream and include, for example, module tag, block names, parameter names, plant area, unit and cell information, in addition to information that defines the stream format such as the identifiers that will indicate the beginning of a particular piece of data.

[0061] These and other techniques, systems, and methods for streaming data in a process control system or plant are discussed in more detail below. It should be expressly noted that, while various descriptions below describe the collection of process control data for storage in a big data historian device, the data

streaming techniques described herein apply to any type of data in a process control system, including lab data, operations data, maintenance data, batch data, event data, alarm data, analysis data, statistical data, etc., so long as the data being streamed can be described by metadata. Additionally, the techniques are applicable to streaming of data between any devices, and is not limited to data streaming between controllers and big data nodes, between process control devices and controllers, or between process control devices and big data nodes. Any process control devices, big data nodes, big data appliances, analytics modules, controllers, workstations, I/O devices, routers, access points, etc., mobile platforms (e.g., smart phones, tablet computers, laptop computers, etc.) in or associated with a process plant (e.g., communicatively connected to the process plant) may be receiver of a data stream, a transmitter of a data stream, or both.

[0062] FIG. 1 illustrates by way of block diagram an example system 100 for streaming process control data in a process plant or process control system. As shown in FIG. 1, the example system 100 includes various sources 102 generating process control data. The sources 102 can include, by way of example, sensors and transmitters; field devices such as valves, actuators, mixers, pumps, etc.; controllers such as those found in a distributed control system (DCS); function blocks operating at any place in the process plant (including within field devices or controllers) to, for instance, receive a parameter or variable and perform a function or operation on the parameter or variable to produce an output; operator or other control/command inputs; I/O devices; communication gateways; or any other source in the process plant. Most generally, the sources 102 include one or more process elements and/or other components or functions corresponding to the process plant or process control system. For example, a source 102 may correspond to an input signal or an output signal of a process control device (e.g., of a controller, field device, or input/output (I/O) device). Additionally or alternatively, a source 102 may be a process variable such as a control variable, a manipulated variable, a disturbance variable, etc. (which may be an input or an output of a process control device). In some cases, the source 102 is a measurement. Examples of measurements may include a process measurement that is directly performed by a field device on some parameter of the process, or that is derived from such a direct measurement; an environmental measurement that is generally independent of control of the process (e.g., an ambient temperature or air pressure, a composition of a raw input material, etc.); a performance measurement (e.g., a process plant communication link bandwidth or delay, a composition of the output of the process plant, etc.), or some other measurement associated with the process plant controlling the process. In some cases, the source 102 is a parameter that is indicative of a balance of mass or a balance of energy within a portion of the process or process plant. In some cases, the source 102 corresponds to an output of another analytics system or module. It is noted that while a data may relate to multiple sources 102 (for example, may reflect the changing output value of an analysis that is being performed on an output of a controller), for ease of readability, the “source” is referred to herein in the singular tense.

[0063] Generally, data may correspond to any parameter or set of parameters that is associated with the process plant and that has a value that varies as a direct or indirect result of the process plant controlling the process, e.g., that varies in real-time or that varies as a direct or indirect result of the process plant (or the portion thereof with which the parameter is associated) is operating on-line.

[0064] The data are collected by a data collection block 104. The collection block 104 receives the data from one or more of the sources 102. While depicted in Fig. 1 as a single data collection block 104, there

may be multiple data collection blocks 104. Each data collection block 104 may correspond to a single source 102, a discrete set of sources 102, or overlapping sets of sources 102. In an embodiment, the collection block 104 and the source 102 are integral. For example, the collection block 104 may be included in a controller or in a field device such as a valve or sensor. In another embodiment, the collection block 104 may be separate from but communicatively connected to the source 102. For instance, the collection block 104 may be physically attached or coupled to the source 102 so that the output of the source 102 traverses the collection block 104 prior to being processed by other blocks, or the collection block 104 may monitor the network link over which the output of the source 102 is transmitted. For example, the collection block 104 may be included in an adjunct device that is coupled to a controller, a field device, or a communication link via which the controller and/or the field device transmits signals.

[0065] The collection block 104 may passively or actively receive or obtain the data from the source 102. For example, when a controller directly sends its output to the collection block 104 (e.g., when the collection block 104 is attached to the output port of the controller, or the controller explicitly addresses its output to be delivered to the collection block 104), the collection block 104 passively obtains the output signal of the source 102. On the other hand, when the collection block 104 monitors the link on which the output of the controller is transmitted, the collection block 104 actively obtains the output signal of the source 102.

[0066] Collected data are queued in a queuing block 106. The data are queued a triggering event causes the queued data to be transmitted. In various embodiments, or in different instances in the same embodiment, the triggering event that causes the queuing block 106 to transmit queued data may be: a full queue, the expiration of a time-to-send timer, or a client request (e.g., from another device) to send the data.

[0067] Once the triggering event causes the queuing block 106 to transmit the collected data, the data are transmitted to a receiving device. The receiving device may be any number of devices, but generally include a big data node, as described below with reference to Fig. 2. The data received by the receiving device are generally stored in memory storage 108 of the receiving device, until further processing or storage occurs. In some embodiments, for example, the data are transferred from in-memory storage 108 of the receiving device to long term data storage 110. The long term data storage 110 may include a big data appliance as described in detail below. In any event, either alternatively or additionally, the data received by the receiving device are transmitted from in-memory storage 108 to an analytics block 112.

[0068] The data may be processed by the analytics block 112 to, by way of example, generate alarms and/or event messages, predictions, and/or recommendations, and/or to provide fault detection for the process plant or for a portion of a process plant. In some embodiments, the analytics block 112 receives or obtains one or more other inputs in conjunction with the data. For example, the analytics block 112 may receive multiple, time-varying input signals from multiple respective sources 102 via multiple collection blocks 104 and/or queuing blocks 106. In some situations, the analytics block 112 receives, in conjunction with the data, one or more additional real-time inputs that are relatively constant over time, such as a setpoint of a control variable or an indication of a static condition within the process plant. In some situations, the analytics block 112 receives, in conjunction with the data (and with any additional real-time inputs, if present), one or more other inputs that are not generated during real-time operations of the process plant, such as measurements that were obtained while the process plant or portion thereof was off-line, an output generated by an off-line, manual analytics tool, data obtained from a model of one or more portions of the process, etc.

[0069] In FIG. 1, the analytics block 112 operates on the data (and any other additional constant or varying inputs, if present) to determine one or more characteristics of the data or one or more operating characteristics of the process plant. The analytics block 112 may, for example, generate additional data, such as statistical data, related to the data received from the collection and queuing blocks 104 and 106. The analytics block 112 may also or alternatively generate information about the state of the process plant or a portion of the process plant, including alarms and event data, prediction data, recommendations, and fault detection/prediction data.

[0070] Indeed, the analytics block 112 automatically or autonomously performs one or more processing functions on the data to determine one or more characteristics of the data. For example, where the data include specific signals (e.g., time-varying signals such as a parameter value) the analytics block 112 may perform filtering, magnitude or amplitude analysis, power analysis, intensity analysis, phase analysis, frequency analysis, spectrum or spectral analysis, correlation analysis, convolution, smoothing, Hilbert transformation, level detection, linear signal processing, non-linear signal processing, and/or any other signal processing technique on the signal within the data. Thus, it follows that the one or more characteristics that are determined by the analytics block 112 correspond to the set of processing techniques that have been applied by the analytics block 112 to the signal or signals within the data. For example, if the analytics block 112 performs spectrum analysis on a signal within the data, the one or more corresponding characteristics of the signal within the data that are determined from the spectrum analysis may include an identification of one or more dominant frequencies within the signal, one or more n -th order frequencies (where n is an integer greater than one), a harmonic, a subharmonic, a bifurcation, bandwidth, distortion, etc. In another example, if the analytics block 112 performs a phase analysis on a signal within the data, the one or more corresponding characteristics of the signal may include an identification of one or more phases of the signal and/or of the presence or absence of any phase shifting. It is noted that while the above spectrum and phase analysis examples illustrate characteristics of one or more repetitive behaviors (e.g., oscillatory or periodic behaviors) that are included in the signal, the analytics block 112 may additionally or alternatively operate on the signal and/or the data to determine any non-repetitive behaviors are present, such as maximum and minimum amplitude over time, impulse responses, etc. In an embodiment, the analytics block 112 may select one or more functions to perform on the data, e.g., based on the source of the data, additional inputs, one or more previously determined characteristics of the data or of a signal in the data, and/or some other criteria.

[0071] The analytics block 112 may also be configured to determine the potential source or sources of the determined characteristics of the signal and/or of the data, e.g., the analytics block 112 may determine the "characteristic source." Specifically, the analytics block 112 may determine a set of process elements that are upstream of the source(s) 102, and identify which of those upstream sources 102 have the most significant impact on the variations in the behavior of the signal and/or data. Additional functionality of the analytics block 112 is described, for example, in co-pending application US 2016-0098647 (appl. no. 14/507,252) (attorney docket no. 06005/593086).

[0072] Regardless of the nature of the output generated by the analytics block 112, the output of the analytics block 112 may be stored in the long term data storage 110 for post-processing, and/or may be stored temporarily in memory storage 114 before, in some embodiments, being made available to a data access block 116. The data access block 116 may be a server (not shown), an Application Program

Interface (API), or the like, that makes some or all of the data in the in-memory storage 114 and/or the long term data storage 110 available to other clients and applications. The data access block 116 may provide the data to a server 118 to be accessed by one or more devices over a network (not shown) or may provide the data directly to a specific device 120 according to a request from the device 120 or according to some predetermined scheme.

[0073] Generally, as referred to herein and as previously discussed, the term “upstream source” refers to a process element, a piece of equipment, or an asset of the process plant that is involved during run-time in causing the process plant to control the process. As used herein, the term “upstream” refers to having an earlier, active participation in or presence during control of the process in real-time, and the term “downstream” refers to having a later, active participation in or presence during control of the process in real-time. For example, a vat that stores raw material for input into the process is upstream of a boiler that heats the raw material, and is upstream of a temperature sensor that measures the temperature of the boiler. The term “process element,” as used herein, generally refers to a physical or logical element that is used to the control of the process or a portion thereof, such as a process control device, a process variable, a measurement, and the like. Thus, using the vat and boiler example, a valve to release the raw material into the boiler is an upstream process element, the temperature sensor is a process element that is downstream of the valve, and another valve to release the heated flow into a pipe for delivery to another piece of equipment is a process element that is downstream of both the first valve and the temperature sensor.

[0074] A “piece of equipment,” “equipment piece,” or “equipment,” as used interchangeably herein, generally refers to a physical element or component which may or may not be directly utilized during control of the process or portion thereof, but nonetheless may be ordered with other equipment pieces and/or process elements with respect to the control or flow of the process. To continue with the above example, the vat and physical supports for the vat are elements that are upstream of the boiler, and the boiler and the supports for the boiler are upstream of the piping that exits the boiler. An “asset” of a process plant, as referred to herein, may be any element corresponding to the process plant having a cost to the provider and/or operator of the process plant. For example, assets of a process plant may include instrumentation, valves, controllers, distributed control systems (DCSs), software, infrastructure, networks, control strategies, applications, configurations, piping, test equipment, configuration equipment, workstations, user interface devices, data storage entities, and the like. Returning again to the vat and boiler example, the vat, boiler, valves, piping, temperature sensor and corresponding supports are assets of the process plant, as are the control modules and strategies that use the valves and temperature sensor to control the heating and the amount of the material in the boiler, as is a portable diagnostic device used to diagnose fault conditions in the vat and/or boiler.

[0075] Any or all of the systems, methods, and techniques disclosed herein may be utilized in any process plant or process control system that is configured to control a process in real-time. Typically, the process is controlled to manufacture, refine, transform, generate, or produce physical materials or products. The process plant may include, for example, one or more wired communication networks and/or one or more wireless communication networks. Similarly, the process plant may include therein one or more wired process elements and/or one or more wireless process elements. The process plant may include centralized databases, such as continuous, batch and other types of historian databases.

[0076] The process plants in which at least portions of the data streaming systems, methods, and techniques disclosed herein are utilized include a process control big data network and process control big data network nodes or devices. For example, at least some of the systems, methods, and techniques disclosed herein may be implemented in a process plant that supports localized, regional, and/or centralized big data, such as described in aforementioned U.S. Publication No. US 2014-0250153 (Application No. 13/784,041), in aforementioned U.S. Patent Publication No. US 2014-0277656 (Application No. 14/174,413), in aforementioned U.S. Publication No. US 2014-0277604 (Application No. 14/212,493), and/or in aforementioned U.S. Publication No. US 2016-0098021 (Application No. 14/507,188) (Attorney Docket No. 06005/593079), entitled "REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS." As such, a process plant that utilizes the data streaming techniques, methods, and systems described herein may include one or more big data devices, at least some of which includes a respective distributed or embedded big data appliance to operate on big data generated by a big data provider node.

[0077] FIG. 2 is a simplified block diagram of an example big data network 200 for a process plant or process control system that controls one or more processes and that supports process control big data and, in particular, that may operate in accordance with the data streaming principles illustrated in FIG. 1. The process control big data network 200 includes one or more process control big data nodes 202-210, each of which collects, observes, generates, stores, analyzes, accesses, transmits, receives, and/or operates on process control big data. The terms "process control big data," "process big data," and "big data," as used interchangeably herein, generally refer to all (or almost all) data that is generated, received, and/or observed by devices included in and associated with the process control system or plant. In an embodiment, all data that is generated by, created by, received at, or otherwise observed by all devices included in and associated with the process plant is collected and stored as big data within the process control big data network 200.

[0078] The example process control big data network 200 includes one or more different types of process control big data nodes or devices 202-210, each of which collects, observes, generates, stores, analyzes, accesses, transmits, receives, and/or operates on process control big data generated from or based on the control of the one or more processes by the process plant or process control system. Each process control big data node or device 202-210 is connected to a process control system big data network backbone (not shown), and may use the backbone to communicate with one or more other process control big data nodes, for example, using the data streaming principles described herein. Accordingly, the process control big data network 200 comprises the process control system big data network backbone and the process control big data nodes 202-210 that are communicatively connected thereto. In an example, the process control big data network 200 includes a plurality of networked computing devices or switches that are configured to route packets to/from various other devices, switches or nodes of the network 200 via the backbone.

[0079] The process control big data network backbone comprises any number of wired communication links and any number of wireless communication links that support one or more suitable routing protocols, including at least a protocol to support streaming of big data as described herein. In addition to the protocol to support streaming of big data as described herein, the process control big data network backbone may support other protocols including, for example, protocols included in the Internet Protocol (IP) suite (e.g., UDP (User Datagram Protocol), TCP (Transmission Control Protocol), Ethernet, etc.), a streaming protocol such as the Stream Control Transmission Protocol (SCTP) and/or another suitable streaming protocol to stream (e.g., transport) data between process control big data nodes, or other suitable routing protocols.

Typically, each node included in the process data big data network 200 may support at least an application layer (and, for some nodes, additional layers) of the routing protocol(s) supported by the backbone. In an embodiment, each process control big data node 202-210 is uniquely identified within the process control system big data network 200, e.g., by a unique network address.

[0080] In an embodiment, at least a portion of the process control system big data network 200 is an ad-hoc network. As such, at least some of the nodes 202-210 (and/or one or more other nodes, such as a user interface device 230) may connect to the network backbone (or to another node of the network 200) in an ad-hoc manner.

[0081] As FIG. 2 is a simplified diagram that depicts communicative connections between various big data nodes 202-210 in the process control big data network 200, the process control network backbone is not explicitly illustrated in FIG. 2. However, an example of such a backbone which may be utilized with any or all of the techniques described herein is described in aforementioned U.S. Patent Publication No. US 2014-0250153 (Application No. 13/784,041). Of course, any or all of the techniques described herein are not limited to being utilized with the backbone described in U.S. Patent Publication No. US 2014-0250153 (Application No. 13/784,041), but may be utilized with any suitable communication network backbone.

[0082] Turning now to the different types of process control big data nodes or devices 202-210, generally, a process control big data node of the network 200 may be a “big data provider” and/or may include a “big data appliance,” as is discussed below.

[0083] The terms “big data provider,” “big data provider node,” or “provider node,” as used interchangeably herein, generally refer to a process control big data node that collects, generates, observes, and/or forwards process control related big data using the process control big data network 200. The process control big data that is generated, collected, observed, and/or forwarded by provider nodes may include data that have been directly utilized in or generated from controlling a process within the plant, e.g., first-order real-time and configuration data that are generated or used by process control devices such as controllers, input/output (I/O) devices, and field devices. Additionally or alternatively, process control big data provider nodes may generate, collect, observe, and/or forward data related to delivering and routing such first-order process control data and other data within the process plant, e.g., data related to network control of the big data network 200 and/or of other communication networks in the plant, data indicative of bandwidth, network access attempts, diagnostic data, etc. Further, some process control big data provider nodes may generate, collect, observe, and/or forward data indicative of results, learning, and/or information that has been learned within the process control big data network 200 by analyzing process control big data that it has collected. Typically, such analytics results, learning, and/or learned information are generated from automatic, autonomous analytics performed by one or more process control big data nodes.

[0084] In most cases, a big data provider node includes multi-core hardware (e.g., multi-core processors) for transmitting and receiving big data in real-time (e.g., streamed) and, in some embodiments, for caching the real-time big data in preparation for streaming or other delivery over the process control big data network 200. A big data provider node may, in some embodiments, also include high-density memory for the caching of the real-time big data. Examples of real-time data that may be transmitted, received, streamed, cached, collected, and/or otherwise observed by big data provider nodes may include process control data such as measurement data, configuration data, batch data, event data, and/or continuous data. For instance, real-

time data corresponding to configurations, batch recipes, setpoints, outputs, rates, control actions, diagnostics, alarms, events and/or changes thereto may be collected. Other examples of real-time data may include process models, statistics, status data, and network and plant management data. In some embodiments, a big data provider node does not cache at least some of the real-time big data that it observes, but instead streams the un-cached data to one or more other big data nodes as the data is observed, received, or generated at the node. Examples of big data provider nodes which may be utilized with any or all of the techniques described herein may be found in aforementioned U.S. Patent Publication No. US 2014-0250153, US 2014-0277656, & US 2014-0277604 (Application Nos. 13/784,041, 14/174,413, and 14/212,493). Of course, any or all of the techniques described herein may be alternatively or additionally utilized with big data provider nodes other than those described in U.S. Publication No. US 2014-0250153, US 2014-0277656, & US 2014-0277604 (Application Nos. 13/784,041, 14/174,413, and 14/212,493).

[0085] On the other hand, the terms “big data appliance,” “big data appliance node,” or “appliance node,” as used interchangeably herein, generally refer to a process control big data node that receives, stores, retrieves, and analyzes process control big data. As such, a process control big data appliance (or “BDA”) generally operates on big data that has been generated or provided by one or more process control big data provider nodes. In some cases, a big data appliance is included in a big data provider node, or is integrally co-resident with a big data provider within a same node or device. In such cases, the big data appliance is referred to as an “embedded big data appliance,” as the appliance is embedded in the provider node or device and operates on the big data that has been received, collected, or generated by the co-resident big data provider. In an example, an embedded big data appliance analyzes big data that has been locally generated and/or provided by the big data provider node on which the embedded big data appliance resides to discover or learn knowledge. This learned knowledge may be stored at the embedded big data appliance, operated on locally by the embedded big data appliance, and/or provided as big data to other big data nodes. Any or all of the techniques described herein may be utilized in conjunction with embedded big data appliances such as described in aforementioned U.S. Patent Publication No. US 2014-0277604 (Application No. 14/212,493) and/or in U.S. Patent Publication No. US 2016-0098021 (Application No. 14/507,188) (Attorney Docket No. 06005/593079), entitled “REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS”, for example, although other suitable embedded big data appliances may additionally or alternatively utilized. Further, it is noted that in embodiments in which a big data provider node includes an embedded big data appliance, the cache of the big data provider node may be reduced in size or omitted, as the embedded big data appliance provides local data storage capabilities.

[0086] In some cases, a big data appliance may be a stand-alone big data node of the process control big data network 200. That is, in these cases, a big data appliance is not embedded in or co-resident with a big data provider node. Thus, a process control big data node that includes a big data appliance may not necessarily itself be a provider of big data.

[0087] FIG. 3 depicts a simplified block diagram of an example big data appliance 214, instances of which may be included in the process control big data network 200 of FIG. 2. Referring to FIG. 3, the example big data appliance 214 includes a big data storage area 220 for historizing or storing received big data, one or more big data appliance receivers 222, and one or more big data appliance request servicers 224. Each of the big data appliance receivers 222 is configured to receive big data packets (which may be streamed from

another node and/or may be generated by a big data provider node on which the appliance 214 resides), process the data packets to retrieve the substantive data and timestamp carried therein, and store the substantive data and timestamp in the big data storage area 220 of the appliance 214, e.g., as time-series data and optionally also as metadata. The big data storage area 220 may comprise multiple local and/or remote physical data drives or storage entities, such as RAID (Redundant Array of Independent Disks) storage, solid-state storage, cloud storage, high-density data storage, and/or any other suitable data storage technology that is suitable for data bank or data center storage, and that has the appearance of a single or unitary logical data storage area or entity to other nodes. Further, each of the big data appliance request servicers 224 is configured to access time-series data and/or metadata stored in the big data appliance storage area 220, e.g., per the request of a requesting entity or application.

[0088] In some instances, a big data appliance 214 includes one or more big data analyzers 226 to perform respective data analytics and/or learning on at least parts of the stored big data, typically in an automatic and/or autonomous manner without using any user input to initiate and/or perform the learning analysis, as described in U.S. Patent Publication No. US 2016-0098647 (Application No. 14/507,252) (Attorney Docket No. 06005/593086), entitled "AUTOMATIC SIGNAL PROCESSING-BASED LEARNING", for example. In an embodiment, the big data analyzers 226 individually and/or collectively perform large scale data analysis on the stored data (e.g., data mining, data discovery, etc.) to discover, detect, or learn new information or knowledge. For example, data mining generally involves the process of examining large quantities of data to extract new or previously unknown interesting data or patterns such as unusual records or multiple groups of data records. The big data analyzers 226 may additionally or alternatively perform large scale data analysis on the stored data (e.g., machine learning analysis, data modeling, pattern recognition, predictive analysis, correlation analysis, etc.) to predict, calculate, or identify implicit relationships or inferences within the stored data. In an embodiment, multiple big data analyzers 226 (and/or multiple instances of at least one big data analyzer 226) operate in parallel and/or cooperatively to analyze the data stored in the big data storage area 220 of the appliance 214. An example of cooperative data analytics which may be utilized with any or all of the techniques described herein is found in aforementioned U.S. Patent Publication No. US 2016-0098021 (Application No. 14/507,188) (Attorney Docket No. 06005/593079) entitled "REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS," although any suitable cooperative data analytics technique or techniques may be utilized with any or all aspects of the present disclosure.

[0089] Typically, each of the big data appliance receivers 222, the big data appliance request servicers 224, and the big data analyzers 226 comprise respective computer-executable instructions stored on one or more non-transitory, tangible memories or data storage devices, and are executable by one or more processor to perform one or more their respective big data functions. In some embodiments, the big data analyzers 226 are not included in the big data appliance 214, but instead are in communicative connection with the big data appliance 214. For example, the big data appliance 214, including the storage area 220, receivers 222 and servicers 125 may be implemented by a first set of computer-executable instructions, and the big data analyzers 226 may be implemented by a second set of computer-executable instructions (which may or may not be stored on the same non-transitory, tangible memories or data storage devices as the first set of computer-executable instructions). Descriptions of various types of example big data appliances and their components which may be utilized with any or all of the techniques described herein may be found in

aforementioned U.S. Patent Publication No. US 2014-0250153, US 2014-0277656, & US 2014-0277604 (Application Nos. 13/784,041, 14/174,413, and 14/212,493), although it is understood that any or all of the techniques described herein may be utilized with other suitable big data appliances.

[0090] Referring again to FIG. 2, the process control big data network 200 may include process control big data provider nodes 202-210 that operate at various levels, tiers, or orders with respect to first-order or primary process related data that is directly generated, routed, and/or used by process control devices such as controllers, I/O devices, field devices, etc. At the lowest order, tier, or level, “local” big data provider nodes or devices 202a-202n that operate nearest to the process to collect, generate, observe, and/or forward primary process big data related to the input, operation, and output of process devices and equipment in the process plant. As such, “local big data provider nodes or devices” 202a-202n typically are nodes and/or devices that generate, route, and/or receive primary process control data to enable the one or more processes to be controlled in real-time in the process plant. Examples of local big data provider nodes 202a-202n include devices whose primary function is directed to generating and/or operating on process control data to control a process, e.g., wired and wireless field devices, controllers, and I/O devices. These process control devices may be communicatively connected to each other and/or to one or more process control communications networks in a distributed manner. For instance, one or more field devices are communicatively connected to one or more I/O devices, which in turn are communicatively connected to one or more controllers, which in turn are communicatively coupled to one or more process control communication networks (e.g., HART[®], WirelessHART[®], process control big data, FOUNDATION[®] Fieldbus, etc.).

[0091] Other examples of local big data provider nodes 202a-202n include devices whose primary function is to provide access to or routing of primary process data through one or more communications networks of the process control system (which may include the process control big data network 200 and/or other communication networks). Examples of such types of local big data provider nodes 202a-202n include access points, routers, interfaces to wired control busses, gateways to wireless communications networks, gateways to external networks or systems, and other such routing and networking devices. Still other examples of local big data provider nodes 202a-202n include devices, such as historian devices, that are configured to temporarily store big data throughout the process control system, e.g., as an overflow cache, way-station, or routing queue.

[0092] In some cases, a local big data provider node includes a respective local big data appliance, as illustrated in FIG. 2 by the nodes 202a, 202n that respectively include the embedded big data appliances 212a, 212n. Each local, embedded big data appliance 212a, 212n receives and stores respective local big data provided by its respective provider 202a, 202n. Further, in some local big data provider nodes, such as in the node 202a, one or more analytics functions, routines, operations, or processes (represented by the encircled **A₁**) may be performed on at least some of the local big data stored in the appliance 212a. In an embodiment, the analytics **A₁** are performed by one or more of the big data analyzers 226 of FIG. 3. The learned information, learnings, and/or the results of the one or more analytics **A₁** may also be stored in the local big data appliance 212a, and at least some of the learned information or results may be provided to another big data node 206a. For example, a local big data provider node that is included in or coupled to a controller that includes an instance of a signal processing module 102, the signal processing module 102

16 08 21

performs a frequency analysis or other signal-processing analysis on the output signal of the controller, and the local big data provider node transmits the results of the analysis to another big data node.

[0093] Some local provider nodes, e.g., as illustrated by the node 202n, include a respective local, embedded big data appliance 212n for local big data collection and historization, however, the resident appliance 212n performs minimal or no analytics. As such, the node 202n merely streams (or otherwise transmits, e.g., upon request or at suitable times) locally stored big data to another node 206b, e.g. for analytical processing or for further forwarding. Some local big data nodes, e.g., the node 202b, do not include any big data appliance at all. Such nodes 202b may stream, in real-time or with the aid of a cache, locally observed big data to one or more other big data nodes 202a, 206b.

[0094] Various types of real-time data, such as process-related data, plant-related data, and other types of data, may be cached, collected, stored, transmitted, and/or streamed as big data by the big data provider nodes or devices 202a-202n. Examples of process-related data include continuous, batch, measurement, and event data that are generated while a process is being controlled in the process plant (and, in some cases, are indicative of an effect of a real-time execution of the process). Further, process-related data may include process definitions, arrangement or set-up data such as configuration data and/or batch recipe data, data corresponding to the configuration, execution and results of process diagnostics, etc.

[0095] Plant-related data, such as data related to the process plant but that may not be generated by applications that directly configure, control, or diagnose a process in the process plant, may also be cached, collected, stored, transmitted, and/or streamed by the big data provider nodes 202a-202n as big data. Examples of plant-related data include vibration data, steam trap data, data indicative of a value of a parameter corresponding to plant safety (e.g., corrosion data, gas detection data, etc.), data indicative of an event corresponding to plant safety, data corresponding to the health of machines, plant equipment and/or devices, data corresponding to the configuration, execution and results of equipment, machine, and/or device diagnostics, and data that is useful for diagnostics and prognostics.

[0096] Further, other types of data including data highway traffic and network management data related to the process control big data network backbone and of various communications networks of the process plant, user-related data such as data related to user traffic, login attempts, queries and instructions, text data (e.g., logs, operating procedures, manuals, etc.), spatial data (e.g., location-based data), and multi-media data (e.g., closed circuit TV, video clips, etc.) may be cached, collected, stored, transmitted, and/or streamed by the big data provider nodes 202a-202n as big data.

[0097] In some embodiments, dynamic measurement and control data may be automatically cached, collected, stored, transmitted, and/or streamed by the big data provider nodes 202a-202n as big data. Examples of dynamic measurement and control data include data specifying changes in a process operation, data specifying changes in operating parameters such as setpoints, records of process and hardware alarms and events such as downloads or communication failures, etc. In addition, static data such as controller configurations, batch recipes, alarms and events may be automatically collected by default when a change is detected or when a controller or other entity is initially added to the big data network 200.

[0098] Moreover, in some scenarios, at least some static metadata that describes or identifies dynamic control and measurement data is captured in the big data provider nodes 202a-202n when a change in the metadata is detected. For example, if a change is made in the controller configuration that impacts the

measurement and control data in modules or units that must be sent by the controller, then an update of the associated metadata is automatically captured by the big data provider nodes 202a-202n. Additionally or alternatively, parameters associated with the special modules used for buffering data from external systems or sources (e.g., weather forecasts, public events, company decisions, etc.), surveillance data, and/or other types of monitoring data may be automatically captured by the big data provider nodes 202a-202n.

[0099] In some situations, added parameters created by end users are automatically captured in the big data provider nodes 202a-202n. For example, an end user may create a special calculation in a module or may add a parameter to a unit that needs to be collected, or the end user may want to collect a standard controller diagnostic parameter that is not communicated by default. Parameters that the end user optionally configures may be communicated in the same manner as the default parameters.

[00100] Referring again to the network 200 of FIG. 2, at one or more levels or tiers above the local big data nodes 202a-202n, the process control big data network 200 may include one or more regional big data nodes 206a-206m. To implement regional big data, the process plant or process control system may be viewed as having a plurality of different areas or regions which may be delineated according to any desired manner such as geographical, physical, functional, logical, etc. In an illustrative but non-limiting example, a process plant may have a first region that receives raw materials and produces a first intermediate product, a second region that receives other raw materials and produces a second intermediate product, and a third region that receives the first and second intermediate products to produce an output product. Each of these three different example regions may be serviced by a respective "regional" big data node 206a, 206b, 206m to operate on big data produced by its respective region. Accordingly, a "regional big data node" provides big data support and services for data that is generated and/or provided by a respective grouping or region of local big data provider nodes 202 and, in some cases, by other big data provider nodes 204. Other big data provider nodes 204 may include, for example, big data nodes that are external to the region of the plant (e.g., a portable diagnostics device or an off-line simulator), user interface devices 230, or data sources that are external to the process plant altogether (e.g., a computing device of a materials supplier, a feed providing a weather forecast, etc.).

[00101] As shown in FIG. 2, a respective grouping or region serviced by a regional big data node 206a-206m may comprise one or more big data provider nodes 202a-202n and/or other big data nodes 204 that are related according to some geographical, physical, functional, logical, or other desired manner. For example, the regional big data node 206a services a region including the local big data provider nodes 202a and 202b, and the regional big data node 206b services a region including the local big data nodes 202b and 202n, as well as another big data node 204. The particular nodes 202, 204 included in a particular region may stream or deliver data to their respective regional big data node 206 for purposes of regional big data storage, access, and/or analysis. Further, any of the big data provider nodes 202a-202n and/or the other big data nodes 204 may communicate with a particular regional big data node 206a-206m to request regionally available services and/or to access regional big data and metadata stored therein, whether such requesting nodes are included in the particular region of the particular regional big data node 206a-206m or not.

[00102] Accordingly, each regional big data node 206a-206m includes a respective regional big data appliance 216a-216m via which big data is received, stored as regional big data, and accessed or requested. Further, each regional big data node 206a-206m may include one or more analytics functions, routines, operations, or processes (e.g., $A_2 - A_w$) that may individually and/or cooperatively operate on at least some

of the regional big data, as generally described in co-pending U.S. patent publication US 2016-0098647 (application no. 14/507,252) (attorney docket no. 06005/593086), entitled "AUTOMATIC SIGNAL PROCESSING-BASED LEARNING," and incorporated herein by reference.

[00103] Big data transmitted and/or received by any of the big data nodes may be streamed according to the methods disclosed herein. In particular, big data (and associated metadata) may be streamed from a first a big data node to a second big data node for one or more of historization, storage, and analysis. Likewise, big data (and associated metadata) may be streamed from a process device to a big data node, and/or from a big data node to a requesting device, such as, for example, a laptop , smartphone, tablet computer, or the like, requesting data via a network connection (e.g., via the Internet). Additionally, one or more of the big data nodes or requesting devices may be in the cloud (a "cloud node").

[00104] In some embodiments, a process plant includes one or more legacy process control devices (not shown) that do not inherently include any big data support. In these embodiments, a gateway node in the plant or an adjunct device directly coupled to a legacy device may convert or translate data messages between a protocol utilized by the legacy device and the protocol utilized by the process control big data network backbone, thereby communicatively connecting the legacy device and the process control big data network 200.

[00105] Typically, big data nodes or devices 202-210 do not have an integral user interface, although some of the big data nodes or devices 202-210 may have the capability to be in communicative connection with one or more user interface devices 230, e.g., by communicating over a wired or wireless communication link, or by plugging a user interface device 230 into a port of the big data nodes or devices 202-210. In FIG. 2, the user interface device 230 is depicted a big data node that is wirelessly connected to the process control big data network 200.

[00106] The user interface device 230 is a device (e.g., a mobile or stationary computing device, a workstation, a handheld device, a surface computing device, a tablet, etc.) that includes one or more integrated user interfaces via which a user or operator may interact with the device and the process control system or process plant to perform activities related to the process plant (e.g., configure, view, monitor, test, diagnose, order, plan, schedule, annotate, and/or other activities). Integrated user interfaces may include a screen, a keyboard, keypad, mouse, buttons, touch screen, touch pad, biometric interface, speakers and microphones, cameras, and/or any other user interface technology. The user interface devices 230 may include a direct wired and/or wireless connection to the process control system big data network backbone, or may include an indirect connection to the backbone, e.g., via an access point or a gateway. In some embodiments, a user interface device 230 may be a requesting device or may otherwise receive streamed big data.

[00107] FIG. 4 is a block diagram including various example aspects of an example process plant or process control system 40 in which the data streaming techniques, methods, systems and apparatuses disclosed herein may be implemented and included. In an embodiment, at least a portion of the process plant 40 is supported by a process control big data network, such as the process control big data network 200 of FIG. 2. However, while the process plant 40 is described below with respect to features illustrated in FIGS. 2-3 for ease of discussion and not for limitation purposes, it is understood that the process plant 40

may utilize a process control big data network other than that described in FIG. 2, or the process plant 400 may omit any process control big data network, nodes, and/or devices.

[00108] In FIG. 4, process control big data nodes or devices are indicated by a “BD” reference that signifies that the node is a process control big data provider node, a big data appliance, or both. For example, nodes or devices indicated by a “BD” reference in FIG. 4 may be, referring to FIG. 2, a local big data provider node and/or appliance 202a-202n, 212a-212n, a regional big data provider node and/or appliance 206a-206m, 216a-216m, a centralized big data provider node 208 and/or appliance 218, or another type of big data node 204.

[00109] In FIG. 4, the process control big data nodes BD are nodes of a process control big data network 400. In an embodiment, the process control big data network 400 is the process control big data network 200 of FIG. 2, and the nodes BD are the nodes 202-208 of the network 200. In FIG. 4, the nodes BD are communicatively connected over the network 400 via a process control system big data network backbone 405. The backbone 405 includes a plurality of networked computing devices or switches that are configured to route packets to/from various process control big data nodes BD. The plurality of networked computing devices of the backbone 405 may be interconnected by any number of wireless and/or wired links, and the big data network backbone 405 may support one or more suitable routing protocols, such as a process control big data streaming protocol, as previously discussed with respect to the network 200.

[00110] As shown in FIG. 4, the process control big data network 400 includes a centralized big data appliance 408 and multiple other big data provider nodes 411, 415, 416, 418, 419, 420, 421, 426, 428, 435, 442a, 442b, 444, 452a, 455a, 458, 472, 475, 478. One of the example big data provider nodes is a big data process controller device 411 that locally collects, analyzes and stores big data of the process control network or plant 40. The controller 411 is communicatively connected to wired field devices 415-422 via input/output (I/O) cards 426 and 428, and is communicatively connected to wireless field devices 440-446 via a wireless gateway 435 and the process control big data network backbone 405. (In another embodiment, though, the controller 411 may be communicatively connected to the wireless gateway 435 using a communications network other than the big data backbone 405, such as by using a process control communications network that includes any number of wired and/or wireless communication links.) In FIG. 4, the controller 411 is a big data provider node BD of the process control system big data network 400, and is directly connected to the process control big data network backbone 405.

[00111] The controller 411, which may be, by way of example, the DeltaV™ controller sold by Emerson Process Management(RTM), may operate to implement a batch process or a continuous process using at least some of the field devices 415-422 and 440-446. In an embodiment, in addition to being communicatively connected to the process control big data network backbone 405, the controller 411 may also be communicatively connected to at least some of the field devices 415-422 and 440-446 using any desired hardware, software, and/or communications links or networks associated with, for example, standard 4-20 mA devices, I/O cards 426, 428, and/or any smart communication protocol such as the FOUNDATION® Fieldbus protocol, the HART® protocol, the WirelessHART® protocol, etc. In an embodiment, the controller 411 may be communicatively connected with at least some of the field devices 415-422 and 440-446 using the process control big data network backbone 405. In FIG. 4, the controller 411, the field devices 415-422 and the I/O cards 426, 428 are wired devices, and the field devices 440-446 are wireless field devices. Of course, the wired field devices 415-422 and wireless field devices 440-446 could conform to any other

desired standard(s) or protocols, such as any wired or wireless protocols, including any standards or protocols developed in the future.

[00112] The process controller device 411 includes a processor 430 that implements or oversees one or more process control routines (e.g., that are stored in a memory 432), which may include control loops. The processor 430 is configured to communicate with the field devices 415-422 and 440-446 and with other process control big data nodes BD that are communicatively connected to the backbone 405. It should be noted that any control routines or modules (including quality prediction and fault detection modules or function blocks) described herein may have parts thereof implemented or executed by different controllers or other devices if so desired. Likewise, the control routines or modules described herein which are to be implemented within the process control system 40 may take any form, including software, firmware, hardware, etc. Control routines may be implemented in any desired software format, such as using object oriented programming, ladder logic, sequential function charts, function block diagrams, or using any other software programming language or design paradigm. The control routines may be stored in any desired type of memory, such as random access memory (RAM), or read only memory (ROM). Likewise, the control routines may be hard-coded into, for example, one or more EPROMs, EEPROMs, application specific integrated circuits (ASICs), or any other hardware or firmware elements. Thus, the controller 411 may be configured to implement a control strategy or control routine in any desired manner.

[00113] In some embodiments, the controller 411 implements a control strategy using what are commonly referred to as function blocks, wherein each function block is an object or other part (e.g., a subroutine) of an overall control routine and operates in conjunction with other function blocks (via communications called links) to implement process control loops within the process control system 40. Control based function blocks typically perform one of an input function, such as that associated with a transmitter, a sensor or other process parameter measurement device, a control function, such as that associated with a control routine that performs PID, fuzzy logic, etc. control, or an output function which controls the operation of some device, such as a valve, to perform some physical function within the process control system 40. Of course, hybrid and other types of function blocks exist. Function blocks may be stored in and executed by the controller 411, which is typically the case when these function blocks are used for, or are associated with standard 4-20 ma devices and some types of smart field devices such as HART(RTM) devices, or may be stored in and implemented by the field devices themselves, which can be the case with Fieldbus devices. The controller 411 may include one or more control routines 438 that may implement one or more control loops. Each control loop is typically referred to as a control module, and may be performed by executing one or more of the function blocks.

[00114] The wired devices 411-422 shown in FIG. 4 include big data wired process control devices 415, 416, and 418-421 and the I/O cards 426, 428. FIG. 4 also shows wired legacy devices 417 and 422, which may operate in conjunction with the wired big data devices 415, 418-421, 426, 428 within the process plant. The wired field devices 415-422 may be any types of devices, such as sensors, valves, transmitters, positioners, etc., while the I/O cards 426 and 428 may be any types of I/O devices conforming to any desired communication or controller protocol. In FIG. 4, the field devices 415-418 are standard 4-20 mA devices or HART(RTM) devices that communicate over analog lines or combined analog and digital lines to the I/O card 426, while the field devices 419-422 are smart devices, such as FOUNDATION[®] Fieldbus field devices, that communicate over a digital bus to the I/O card 428 using a Fieldbus communications protocol. In some

embodiments, though, at least some of the big data wired field devices 415, 416 and 418-421 and/or at least some of the big data I/O cards 426, 428 additionally or alternatively communicate with the controller 411 using the big data network backbone 405.

[00115] The wireless field devices 440-446 shown in FIG. 4 include examples of wireless big data nodes or devices BD (e.g., devices 442a, 442b, 444). FIG. 4 also includes an example of a legacy wireless device (e.g., device 446). The wireless field devices 440-446 communicate in a wireless network 470 using a wireless protocol, such as the WirelessHART(RTM) protocol. Such wireless field devices 440-446 may directly communicate with one or more other big data devices or nodes BD of the process control big data network 400 that are also configured to communicate wirelessly (using a wireless streaming protocol, for example). To communicate with one or more other big data nodes that are not configured to communicate wirelessly, the wireless field devices 440-446 may utilize a wireless gateway 435 connected to the backbone 405 or to another process control communications network. Any number of wireless field devices that support big data may be utilized in the process plant 40.

[00116] The wireless gateway 435, as illustrated in FIG. 4, is another example of a big data node BD included in the process control plant or system 40, and provides access to/from the various wireless devices 440-458 of a wireless communications network 470. In particular, the wireless gateway 435 provides communicative coupling between the wireless devices 440-458, the wired devices 411-428, and/or other nodes or devices of the process control big data network 400 (including the controller 411 of FIG. 4). For example, the wireless gateway 435 may provide communicative coupling by using the big data network backbone 405 and/or by using one or more other communications networks of the process plant 40.

[00117] The wireless gateway 435 provides communicative coupling, in some cases, by the routing, buffering, and timing services to lower layers of the wired and wireless protocol stacks (e.g., address conversion, routing, packet segmentation, prioritization, etc.) while tunneling a shared layer or layers of the wired and wireless protocol stacks. In other cases, the wireless gateway 435 may translate commands between wired and wireless protocols that do not share any protocol layers. In addition to protocol and command conversion, the wireless gateway 435 may provide synchronized clocking used by time slots and superframes (sets of communication time slots spaced equally in time) of a scheduling scheme associated with the wireless protocol implemented in the wireless network 470. Furthermore, the wireless gateway 435 may provide network management and administrative functions for the wireless network 470, such as resource management, performance adjustments, network fault mitigation, monitoring traffic, security, and the like.

[00118] Similar to the wired field devices 415-422, the wireless field devices 440-446 of the wireless network 470 may perform physical control functions within the process plant 40, e.g., opening or closing valves or take measurements of process parameters. The wireless field devices 440-446, however, are configured to communicate using the wireless protocol of the network 470. As such, the wireless field devices 440-446, the wireless gateway 435, and other wireless nodes 452-458 of the wireless network 470 are producers and consumers of wireless communication packets.

[00119] In some scenarios, the wireless network 470 may include non-wireless devices, which may or may not be big data devices. For example, a field device 448 of FIG. 4 may be a legacy 4-20 mA device and a field device 450 may be a traditional wired HART(RTM) device. To communicate within the network 470,

the field devices 448 and 450 may be connected to the wireless communications network 470 via a wireless adaptor (WA) 452a or 452b. In FIG. 4, the wireless adaptor 452b is shown as being a legacy wireless adaptor that communicates using the wireless protocol, and the wireless adaptor 452a is shown as supporting big data and thus is communicatively connected to the big data network backbone 405. Additionally, the wireless adaptors 452a, 452b may support other communication protocols such as Foundation[®] Fieldbus, PROFIBUS, DeviceNet(RTM), etc. Furthermore, the wireless network 470 may include one or more network access points 455a, 455b, which may be separate physical devices in wired communication with the wireless gateway 435 or may be provided with the wireless gateway 435 as an integral device. In FIG. 4, the network access point 455a is illustrated as being a big data device BD, while the network access point 455b is a legacy access point. The wireless network 470 may also include one or more routers 458 to forward packets from one wireless device to another wireless device within the wireless communications network 470, each of which may or may not support distributed big data in the process control system 40. The wireless devices 440-446 and 452-458 may communicate with each other and with the wireless gateway 435 over wireless links 460 of the wireless communications network 470, and/or via the big data network backbone 405, if the wireless devices are distributed and/or centralized big data devices.

[00120] Accordingly, FIG. 4 includes several examples of big data devices of nodes BD which primarily serve to provide network routing functionality and administration to various networks of the process control system. For example, the wireless gateway 435, the access point 455a, and the router 458 each include functionality to route wireless packets in the wireless communications network 470. The wireless gateway 435 performs traffic management and administrative functions for the wireless network 470, as well as routes traffic to and from wired networks that are in communicative connection with the wireless network 470. The wireless network 470 may utilize a wireless process control protocol that specifically supports process control messages and functions, such as WirelessHART(RTM). As shown in FIG. 4, the devices 435, 455a, 452a, 442a, 442b and 458 of the wireless network 470 support big data in the process control plant 40, however, any number of any types of nodes of the wireless network 470 may support distributed big data in the process plant 40.

[00121] Other devices that communicate using other wireless protocols may be big data nodes or devices BD of the process control big data network 400. In FIG. 4, one or more wireless access points 472 are big data devices BD that utilize other wireless protocols, such as Wi-Fi(RTM) or other IEEE 802.11 compliant wireless local area network protocols, mobile communication protocols such as WiMAX(RTM) (Worldwide Interoperability for Microwave Access), LTE (Long Term Evolution) or other ITU-R (International Telecommunication Union Radiocommunication Sector) compatible protocols, short-wavelength radio communications such as near field communications (NFC) and Bluetooth, or other wireless communication protocols. Typically, such wireless access points 472 allow handheld or other portable computing devices (e.g., user interface devices) to communicate over a respective wireless network that is different from the wireless network 470 and that supports a different wireless protocol than the wireless network 470. In some scenarios, in addition to portable computing devices, one or more process control devices (e.g., controller 411, field devices 415-422, or wireless devices 435, 440-458) may also communicate using the wireless protocol supported by the access points 472.

[00122] Additionally in FIG. 4, one or more gateways 475, 748 to systems that are external to the immediate process control system 40 are big data nodes or devices BD of the process control big data

network 400. Typically, such systems are customers or suppliers of information generated or operated on by the process control system 40. For example, a plant gateway node 475 may communicatively connect the immediate process plant 40 (having its own respective process control big data network backbone 405) with another process plant having its own respective process control big data network backbone. In another example, a single process control big data network backbone 405 may service multiple process plants or process control environments. In still another example, the plant gateway node 475 communicatively connects the immediate process plant 40 with a cloud big data node 210 and/or a cloud big data appliance 221.

[00123] In FIG. 4, a plant gateway node 475 communicatively connects the immediate process plant 40 to a legacy or prior art process plant that does not include a process control big data network 400 or backbone 405. In this example, the plant gateway node 475 may convert or translate messages between a protocol utilized by the process control big data backbone 405 of the plant 40 and a different protocol utilized by the legacy system (e.g., Ethernet, Profibus, Fieldbus, DeviceNet(RTM), etc.). The one or more external system gateway nodes 478 communicatively connect the process control big data network 400 with the network of an external public or private system, such as a laboratory system (e.g., Laboratory Information Management System or LIMS), an operator rounds database, a materials handling system, a maintenance management system, a product inventory control system, a production scheduling system, a weather data system, a shipping and handling system, a packaging system, the Internet, another provider's process control system, or other external systems.

[0100] Although FIG. 4 only illustrates a single controller 411 with a finite number of field devices 415-22 and 440-446, this is only an illustrative and non-limiting embodiment. Any number of controllers 411 may support big data, and any of the controllers 411 may communicate with any number of wired or wireless field devices 415-422, 440-446 to control a process in the plant 40. Furthermore, the process plant 40 may also include any number of wireless gateways 435, routers 458, access points 455, wireless process control communications networks 470, access points 472, and/or gateways 475, 478. Still further, FIG. 4 may include any number of centralized big data appliances 408, which may receive and store collected data and/or generated learned data or knowledge from any or all of the devices in the process plant 40. In some embodiments, the process control big data network 400 may include any number of regional big data appliances and nodes (not shown in FIG. 4).

[0101] Further, the combination of aspects, devices, and components included in the example process plant 40 as illustrated by FIG. 4 is exemplary only. The techniques, systems, methods, and apparatuses disclosed herein may be utilized in process plants with zero or more any of the aspects illustrated in FIG. 4. For example, the techniques, systems, methods, and apparatuses disclosed herein may be utilized in a process plant without a centralized big data appliance 408, or in a process plant with one or more regional big data appliances and/or nodes. In another example, the techniques, systems, methods, and apparatuses disclosed herein may be utilized in a process plant with only legacy devices.

[0102] Referring now simultaneously to FIGS. 1-4, any number of any the nodes 202, 206, 208, 210 of the process control big data network, the big data appliances 212, 216, 218, 221, the big data nodes 411, 415, 416, 418, 419, 420, 421, 426, 428, 435, 442a, 442b, 444, 452a, 455a, 458, 472, 475, 478, and/or the legacy devices 417, 422, 440, 446, 448, 450, 452b, 455b may be configured to and operable to stream data. For

example, with respect to legacy devices, function blocks may, in some embodiments, be instantiated within one or more of the legacy devices to implement the data streaming described herein.

[0103] Specifically, data streaming as described herein below may occur in various contexts between various devices and device types. With reference still to FIG. 4, legacy devices (e.g., device 417) and/or field devices having embedded big data nodes (e.g., 415, 416), may stream data to I/O devices (e.g., I/O devices 426, 428), legacy devices (e.g., device 417) and/or field devices having embedded big data nodes (e.g., 415, 416), may stream data directly to a controller (e.g., 411) and/or to any big data node (e.g., 411, 415, 416, 418, 419, 420, 421, 426, 428, 435, 442a, 442b, 444, 452a, 455a, 458, 472, 475, 478) or big data appliance (e.g., 408), controllers (e.g., 411) may stream data to big data nodes in I/O devices (e.g., 426, 428), field devices (e.g., 419), other controllers, big data appliances (e.g., 408), servers/gateways (e.g., 478), and external devices (not shown). Big data nodes (e.g., 419) and big data appliances (e.g., 408) may stream data to any other big data node or big data appliance or to any requesting device, including, by way of example and not limitation, any other big nodes or big data appliances, external devices, servers, mobile devices, and workstations.

[0104] An advantage of the presently described data streaming techniques, methods, and systems includes the availability of data and metadata at the time it is needed and/or requested. The available data include continuous IO data, data from function blocks, events, batch data, lab entry data, spectral data, alarm data, analysis data, etc. For the data to be useful, for example, in embodiments where the consumer of the data is a function block or a controller, the data values must be collected at the rate at which they are accessed or changed within the control system, and streamed to any receiving devices at a rate equal to or greater than the rate at which they are access or changed within the control system. In previous systems, collection data at the same or greater rate as the data is generated is prevented by several system limitations, including the design of the controller, the communications architecture of the system, and the capabilities and design of the data historians. Each of these limitations will be explained below, and the improvements to each described relative to the presently described techniques.

[0105] Controllers designed for previous generations of systems were not designed to collect and/or buffer data as the data are being generated in or received at the controller. Typically, controllers capture periodic snapshots of the system parameter values, and communicate the snapshot of values to a downstream node for collection and storage in a historian. By capturing only periodic snapshots, changes, short-term trends, and instability in data or events may be missed or, if captured, may be delayed before being communicated to a downstream node for analysis or storage.

[0106] A controller in accordance with the presently described techniques, methods, and systems for data streaming is redesigned to facilitate the buffering that is required to support big data collection and storage. Among the improvements to controller design are the inclusion of additional memory. A double buffer mechanism is operable to capture block parameters after module execution as fast as the modules can execute (e.g., 100 msec). As a first buffer collects data, a second buffer streams the data to one or more receiving devices (e.g., to a big data node and/or a big data appliance).

[0107] At the same time, the communications infrastructures of previous systems were designed to send individual parameter values from device to device. As described herein, the present system is designed to capture and send large numbers of parameters and data, and can send the data as a stream, allowing more

of the available data in the system to be collected, utilized, analyzed, stored, and processed. Additionally, the present system minimizes network overhead by sending data and metadata separately, allowing metadata to be transmitted over the network only when it is needed.

[0108] Historians were likewise ill-equipped to support the collection of all data in a process. Instead, historians were designed to support collection of continuous process data, but event data such as process alarms, batch data, or hardware alarms were stored in a different database, making it difficult to analyze in real-time (or non-real time, for that matter) with the continuous process data. Other types of data (e.g., hardware or communications diagnostic data) were not stored at all in order to reduce disk access time and storage space. Further, historians generally required data to be stored in a compressed format, which can cause the loss of valuable information through so-called “compression loss.” The improved techniques, methods, and systems, including the methods by which data are transmitted between devices separately from corresponding metadata facilitate the collection, storage, and organization of the higher volumes of data while simultaneously minimizing the processing required to efficiently process the incoming data.

[0109] The improved data transmission techniques (i.e., data streaming) are described below with respect to communication between a controller and a big data node or big data appliance. However, the same principles should be understood to apply to communications to and from other devices, in various combinations as described above.

[0110] In the described data streaming techniques, devices and, in particular, devices receiving a data stream maintain information, in the form of metadata, about the content and structure of the data stream. As a result, the network bandwidth required to communicate a given set of data is reduced. The “resident” metadata – metadata that are stored on the device – falls into three general categories: standard data descriptions, custom data descriptions, and stream format metadata. Standard data descriptions define parameters that are streamed by default, and will be defined for function blocks, units, and devices that access and/or act upon process inputs and outputs by default. Similarly, when a user assigns additional parameters to be collected, stored, and/or analyzed as part of the big data, custom data descriptions define the parameter to be communicated, including the data type, name, and associated module or block. The data descriptions facilitate the processing of the standard and additional parameters by the controller and/or the big data appliance. The resident metadata also include stream format data that define the process data, alarm data, event data, and device diagnostic data that may be included in the data stream. For each data type, and for each source within the data type, the stream format metadata include an ID assigned by the controller to each instance of a function block, parameter, unit, device, alarm, and/or event, and a link to the associated metadata for that ID.

[0111] FIG. 5 depicts a block diagram of an example controller 500. The example controller 500 includes a buffer 502, which may be one of two or more buffers that reside in one or more memory devices of the controller 500. The buffer 502 depicted in FIG. 5 represents a set of data to be streamed from the controller to another device, such as a workstation or a big data node/appliance. The controller 500 also includes a control module 504 with an analog input block 506, a digital input block 508, and a parameter 510, each of which generates data and sends the data to the buffer 502. A unit 512 also provides data to the buffer 502. In embodiments, data may be added to the buffer from the control module 504, input blocks 506 and 508, parameter 510, and the unit 512, according to a variety of functions such as one or more of the functions

described with reference to FIG. 1. For example, a collection block 104 may collect data from the module 504, the input blocks 506 and 508, the parameter 510, and the unit 512, to be placed in the buffer 502.

[0112] In embodiments, the data in the buffer 502 do not correspond with a particular execution of a unit or module, but instead span multiple executions of the unit or module. That is, the buffer 502 may be large enough (i.e., have sufficient memory allocated to it) that it can collect data over multiple executions of the module 504 before streaming the collected data, without overflowing. The time between the start of buffering and the streaming of the buffered data is dependent on how often the data are needed and/or requested, the size of the buffers 502, the type and quantity of data being collected and stored in the buffer 502, the number of modules 504 assigned to the buffer 502 and/or to the controller 500, etc. Various tradeoffs between transmitting data often (to achieve close-to-real-time availability) and buffering large quantities of data (to decrease traffic frequency on the network, for example) will be apparent to those of ordinary skill, and can be configured according to the requirements of a particular system.

[0113] In any event, the buffer 502 may collect and temporarily store both process data 514 and alarm/event data 516. Each set of streamed data has a full timestamp 518. The timestamp 518 may correspond, for example, to the time that the buffer 502 commenced collecting data (which would typically correspond to the time that another buffer stopped collecting data and commenced streaming the data it had collected). Additionally, each module or unit included in the data stream (and, correspondingly, in the data buffer 502) will be indicated with an identifier (ID) and will have a corresponding incremental timestamp, indicating the offset from the timestamp 518. For example, in FIG. 5, the buffer 502 includes a module ID 520 corresponding to the module 504 and an incremental timestamp 521 corresponding to the execution time of the module 504. For a particular module the buffer further includes identifiers for each block and/or parameter in the module. Data in the buffer 502 for the module 506, for example, includes a block ID 522 corresponding to the analog input block 506, a block ID 524 corresponding to the digital input block 524, and a parameter ID 526 corresponding to the parameter 510. A unit ID 528 corresponds to the unit 512, and has an associated incremental timestamp 530 corresponding to the execution time of the unit 512. Each of the IDs is followed (after any timestamps) by corresponding data as depicted in FIG. 5. In some embodiments, a size field may be included in the data to be streamed (see, e.g., FIG. 7). The size field may be reserved until the buffer is full or otherwise ready to be streamed to a receiving device, and a value input into the field upon completion of the data collection, once the data stream is formatted to be transmitted.

[0114] The buffer 502 also includes the alarm and event data 516. Like the process data 514, the alarm and event data 516 includes an ID and an incremental timestamp for each event and/or alarm, in addition to the data for the alarm or the event. For example, FIG. 5 depicts an alarm ID 532 and an incremental timestamp 534 before the data for the alarm corresponding to the alarm ID 532, and an event ID 536 and an incremental timestamp 538 before the data for the event corresponding to the event ID 536. Of course, alarms and events may occur multiple times in the course of execution of a particular module, or during the period that the buffer is collecting data. Accordingly, the buffer may include multiple instances of a particular event and/or alarm ID.

[0115] FIG. 6 illustrates one arrangement for configuring a controller 600 and a big data appliance 602 for data streaming. In the illustrated arrangement, the controller 600 is pre-loaded with a set of standard data descriptions (metadata) 604. The standard data descriptions 604 define the data that are streamed by default. If the user assigns additional block, module, unit, and/or device parameters to be streamed, then

identifiers (IDs) are automatically assigned to the additional parameters and uploaded to the controller 600 as custom data descriptions (metadata) 606. A stream format description (metadata) 608 defines the format of the data stream including, what data are included in the stream, and the ID assigned by the controller to each instance of a function block, parameter, unit, device, alarm, and event, and provides a link to the associated description metadata 604 and/or 606. Though FIG. 6 depicts the custom data descriptions 606 and the stream format description 608 being uploaded to the controller 600, in some embodiments, the controller may be capable of generating the custom data descriptions 606 and/or the stream format description 608 within the controller 600 when the controller is configured to collect data and/or to utilize data to control the process plant.

[0116] In any event, the controller 600 communicates all of the metadata to the big data appliance 602. In this manner, the big data appliance 602 is automatically configured to receive the data stream, before any data is actually transmitted to the big data appliance 602. The data stream thereafter communicated from the controller 600 to the big data appliance 602 need only include the identifiers assigned by the controller, the timestamps, and the data. No additional metadata are required to be streamed with the data, thereby reducing bandwidth overhead and decreasing the time required to stream a given quantity of data. While the big data appliance 602 is depicted in FIG. 7 as a stand-alone device, the big data appliance 602 could, instead, be any big data node as described with reference to FIG. 2, for example. Additionally, the sending device, while depicted and described as a controller 600 need not be a controller at all. Instead, the arrangement above may be used between any two sending and receiving devices including, by way of example, between a first big data node and a second big data node, between a big data node and a mobile device (e.g., a smart phone, tablet computer, laptop computer, etc.), between a big data node and a cloud server, between a big data node and an analytics module, etc.

[0117] In embodiments, the techniques, methods, and systems for data streaming include mechanisms that facilitate automatic updating of the streaming metadata when the receiving device encounters an unknown element in the data stream. FIG. 7 illustrates one such implementation in the form of a block diagram. In FIG. 7, a portion 700 of a process control system includes a controller 702, a big data appliance 704, a configuration database 706, a big data network backbone 708 communicatively coupling the controller 702, the big data appliance 704 and the configuration database 706. In the example portion 700 of a process control system depicted in FIG. 7, the controller 702 and the big data appliance 704 are configured as described above with respect to FIG. 6. As depicted in FIG. 7, however, as data (line "A", in FIG. 7) are streamed from the controller 702 to the big data appliance 704, it is possible that the data stream will include a new ID in the data stream (e.g., because a user added a parameter to the controller 702, and the parameter was added to the data stream). When the big data appliance 704 encounters the new ID in the data stream, the big data appliance 704 will not recognize the ID and will send a request (line "B", in FIG. 7) to the controller 702 to send updated metadata. The controller 702 will send updated metadata (line "C", in FIG. 7), which may include one or more of the standard data descriptions, the custom data descriptions, and the stream format description. In an embodiment, the controller 702 sends the custom data descriptions and the stream format description.

[0118] FIGS. 8A and 8B are flow charts depicting the method described above. In particular, FIG. 8A depicts a method 800 for streaming data to another device, while FIG. 8B depicts a method 801 for receiving a data stream. Collectively, the FIGS. 8A and 8B depict the flow of data (broken lines) between the two

devices and the operation of the methods 800 and 801 with respect to one another. In some embodiments, the transmitting device (the device performing the streaming of the data) is a controller streaming data to receiving device that comprises big data appliance (e.g., for historization), while in other embodiments, the transmitting device is a big data node (e.g., a node comprising a big data appliance storing historized data) to a receiving device that is a workstation computer, a laptop computer, a tablet computer, a smart phone, etc. However, the methods 800 and 801 are not limited, respectively, to controllers and big data nodes. Instead, the methods 800 and 801 may apply, respectively, to any device transmitting a data stream and any device receiving a data stream. In the method 800, the transmitting device (i.e., the device streaming data) stores metadata (block 802). The stored metadata provide information about the data being streamed, information about the structure of the data stream, or both. In some embodiments, the stored metadata are used by the transmitting device to collect the data to be streamed (e.g., to know which data to collect) and/or to determine which data to include in the data stream. In any event, the metadata are transmitted to the receiving device (block 806). The metadata may be transmitted to the receiving device any time after they are stored. In embodiments, the metadata are transmitted to the receiving device immediately upon becoming available in the transmitting device, while in other embodiments, the metadata are transmitted to the receiving device immediately prior to the first data stream transmission. The receiving device receives the metadata (block 803) and stores the metadata locally.

[0119] The data for the data stream are collected and buffered/stored locally prior to transmission (block 804). In embodiments, a trigger event causes the active butter to start streaming data and, presumably, buffering of incoming data is handled by a previously non-active buffer. In any event, the collected/buffered data are streamed to a receiving device (block 808). The receiving device receives the data stream (block 805) and parses the data stream according to the received metadata (block 807). If, while parsing the data stream, the receiving device does not encounter any new, unrecognized IDs among the data in the data stream (block 809), then the receiving device continues to receive and parse the data stream (blocks 805, 807). If, on the other hand, the receiving device encounters one or more new, unrecognized IDs among the data in the data stream (block 809), then the receiving device caches the data in the data stream (block 811) and sends a request to the transmitting device, requesting updated metadata (block 813).

[0120] At the same time, the transmitting device continues to collect and buffer data (block 810), and to stream the data to the receiving device (block 812). If the transmitting device does not receive a request for updated metadata (block 814), then the transmitting device continues to collect and buffer data, and to stream the data to the receiving device. However, if the transmitting device receives a request for updated metadata (block 814), then the transmitting device generates or receives updated metadata (block 816), and transmits the updated metadata to the receiving device (block 818), before continuing to collect, buffer, and transmit the data as a stream.

[0121] When the receiving device receives the updated metadata (block 815), the receiving device parses and processes (e.g., historizing it, if the receiving device is a big data appliance) the cached data stream (block 817), before proceeding to continue to receive the data stream from the transmitting device (block 805).

[0122] In an alternative embodiment, the transmitting device updates the metadata for the data stream as soon as a configuration change occurs. That is, instead of waiting for a request to send updated data to the receiving device, the transmitting device automatically generates updated metadata upon any configuration

change, and transmits the updated metadata to the receiving device before commencing the transmission of a data stream having any new format or any newly added IDs. In this manner, the data stream need not be cached by the receiving device, as the receiving device will always have the metadata required to parse and/or process the data stream. This may have the effect of decreasing cache/memory requirements for the receiving device, and decreasing latency that may be caused by the request for the new metadata and the later parsing/processing of the cached data stream.

[0123] FIGS. 9A and 9B are flow charts depicting the method described above, which is a variation of the method of FIGS. 8A and 8B. In particular, FIG. 9A depicts a method 900 for streaming data to another device, while FIG. 9B depicts a method 901 for receiving a data stream. Collectively, the FIGS. 9A and 9B depict the flow of data (broken lines) between the two devices and the operation of the methods 900 and 901 with respect to one another. As with the methods 800 and 801, the methods 900 and 901 may be implemented, respectively, by any device transmitting a data stream and any device receiving a data stream.

[0124] In the method 900, the transmitting device (i.e., the device streaming data) stores metadata (block 902). The stored metadata provide information about the data being streamed, information about the structure of the data stream, or both. In some embodiments, the stored metadata are used by the transmitting device to collect the data to be streamed (e.g., to know which data to collect) and/or to determine which data to include in the data stream. In any event, the metadata are transmitted to the receiving device (block 906). The metadata may be transmitted to the receiving device any time after they are stored. In embodiments, the metadata are transmitted to the receiving device immediately upon becoming available in the transmitting device, while in other embodiments, the metadata are transmitted to the receiving device immediately prior to the first data stream transmission. The receiving device receives the metadata (block 903) and stores the metadata locally.

[0125] The data for the data stream are collected and buffered/stored locally prior to transmission (block 904). In embodiments, a trigger event causes the active buffer to start streaming data and, presumably, buffering of incoming data is handled by a previously non-active buffer. In any event, the collected/buffered data are streamed to a receiving device (block 908). The receiving device receives the data stream (block 905) and parses the data stream according to the received metadata (block 907). At the same time, the transmitting device continues to collect and buffer data (block 910), and to stream the data to the receiving device (block 912). If the configuration of the transmitting device is not changed or updated (block 914), then the transmitting device continues to collect and buffer data (block 910), and to stream the data to the receiving device (block 912). However, if the configuration of the transmitting device is changed (block 914), then the transmitting device generates or receives updated metadata (block 916), and transmits the updated metadata to the receiving device (block 918), and transmits the updated data stream (block 908) before continuing to collect, buffer, and transmit the data as a stream (blocks 910, 912).

[0126] If the receiving device does not receive new metadata (block 909), then the receiving device continues to receive the data stream (block 905) and parse the data stream according to the stored metadata (block 907). However, if the receiving device receives updated metadata (block 909), the receiving device stores the updated metadata (block 911) and then continues to receive the data stream (block 905) and parses and processes (e.g., historizing it, if the receiving device is a big data appliance) the data stream according to the (newly stored) updated metadata (block 907).

[0127] Embodiments of the techniques described in the present disclosure may include any number of the following aspects, either alone or combination:

[0128] 1. A method for communicating data in a process control system, the method comprising: storing a first set of metadata in a device operable to transmit a data stream; storing, in the device operable to transmit the data stream, a set of data for transmitting as the data stream; transmitting the first set of metadata to a device operable to receive the data stream; receiving the first set of metadata at the device operable to receive the data stream; transmitting the data stream to the device operable to receive the data stream; receiving the data stream at the device operable to receive the data stream, after receiving the first set of metadata; and parsing the received data stream, at the device operable to receive the data stream, according to the first set of metadata.

[0129] 2. The method according to aspect 1, wherein the first set of metadata includes stream format description metadata describing the structure of the data stream.

[0130] 3. The method according to either aspect 1 or aspect 2, wherein the first set of metadata includes data description metadata describing the content of the data stream.

[0131] 4. The method according to aspect 3, wherein the data description metadata includes standard data description metadata and custom data description metadata.

[0132] 5. The method according to any one of the preceding aspects, wherein the set of data includes parameter data, module and block data, event data, and alarm data.

[0133] 6. The method according to any one of the preceding aspects, wherein the device operable to transmit the data stream is a controller, and wherein the device operable to receive the data stream comprises a big data appliance.

[0134] 7. The method according to aspect 6, wherein the controller is pre-configured with a set of standard data descriptors.

[0135] 8. The method according to either aspect 6 or aspect 7, wherein the controller automatically generates a set of custom data descriptors when a new parameter is added to the parameters assigned to the controller.

[0136] 9. The method according to any one of the preceding aspects, wherein device operable to transmit the data stream automatically updates the metadata if a configuration of the device operable to transmit the data stream is changed.

[0137] 10. The method according to any one of aspects 1 to 5, wherein the device operable to transmit the data stream comprises a big data appliance, and wherein the device operable to receive the data stream is a device selected from the group consisting of: a workstation computer, a laptop computer, a smart phone, and a tablet computer.

[0138] 11. The method according to any one of aspects 1 to 5, wherein the device operable to transmit the data stream comprises a big data appliance, and wherein the device operable to receive the data stream is a cloud computing device.

[0139] 12. The method according to any one of the preceding aspects, further comprising: identifying in the data stream, at the device operable to receive the data stream, an identifier that is not in the first

metadata; caching the data stream at the device operable to receive the data stream; sending a request, from the device operable to receive the data stream to the device operable to send the data stream, to provide an updated set of metadata; receiving the request to provide the updated set of metadata at the device operable to send the data stream; sending the updated set of metadata from the device operable to send the data stream to the device operable to receive the data stream; receiving, at the device operable to receive the data stream, the updated set of metadata; and parsing the cached data stream according to the updated set of metadata.

[0140] 13. The method according to any one of the preceding aspects, further comprising: recognizing, in the device operable to send the data stream, a configuration change; updating the metadata according to the recognized configuration change; sending the updated metadata from the device operable to send the data stream to the device operable to receive the data stream; receiving, at the device operable to receive the data stream, the updated set of metadata; and sending, from the device operable to send the data stream to the device operable to receive the data stream, the data stream, wherein the data stream is structured according to the updated set of metadata; and parsing the data stream, in the device operable to receive the data stream, according to the updated set of metadata.

[0141] 14. The method of any one of the preceding aspects, wherein storing, in the device operable to transmit the data stream, the set of data for transmitting as the data stream comprises: collecting data from a plurality of process control devices; and queuing the collected data to until a trigger event.

[0142] 15. The method according to aspect 14, wherein the trigger event is one of: a queue in which the collected data are queued is full, a time-to-send timer expires, or a client requests data in the queue.

[0143] 16. The method according to any one of the preceding aspects, wherein the first set of metadata comprises metadata describing a relationship of a first subset of data in the set of data to a second subset of data in the set of data.

[0144] 17. The method according to aspect 16, wherein the relationship is an upstream relationship.

[0145] 18. The method according to aspect 16, wherein the relationship is a downstream relationship.

[0146] 19. The method according to any one of the preceding aspects, further comprising performing an analysis on the set of data to generate analysis data.

[0147] 20. The method according to aspect 19, wherein the analysis is performed at the device operable to transmit the data stream, and wherein the analysis data are added to the data stream.

[0148] 21. The method according to any one of the preceding aspects, wherein the data stream comprises a timestamp.

[0149] 22. The method according to aspect 21, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the set of data.

[0150] 23. The method according to any one of the preceding aspects, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.

[0151] 24. A method of streaming data in a process control system, the method comprising: storing a first set of metadata in a device operable to transmit a data stream; collecting a set of data for transmitting as

the data stream; buffering the set of collected data; transmitting the first set of metadata to a device operable to receive the data stream; and transmitting the data stream to the device operable to receive the data stream, wherein no further metadata are transmitted unless a trigger event causes the device operable to transmit the data stream to transmit additional metadata.

[0152] 25. The method according to aspect 24 further comprising: receiving a request for updated metadata; generating or downloading updated metadata; and sending updated metadata in response to the request for updated metadata.

[0153] 26. The method according to aspect 24, further comprising: recognizing a changed configuration parameter; generating or downloading updated metadata; and sending the updated metadata to the device operable to receive the data stream before sending a data stream according to the updated metadata.

[0154] 27. The method according to any one of aspects 24 to 26, wherein the first set of metadata includes stream format description metadata describing the structure of the data stream.

[0155] 28. The method according to any one of aspects 25 to 27, wherein the first set of metadata includes data description metadata describing the content of the data stream.

[0156] 29. The method according to aspect 28, wherein the data description metadata includes standard data description metadata and custom data description metadata.

[0157] 30. The method according to any one of aspects 24 to 29, wherein the set of data includes parameter data, module and block data, event data, and alarm data.

[0158] 31. The method according to any one of aspects 24 to 30, wherein the device operable to transmit the data stream is a controller, and wherein the device operable to receive the data stream comprises a big data appliance.

[0159] 32. The method according to aspect 31, wherein the controller is pre-configured with a set of standard data descriptors.

[0160] 33. The method according to either aspect 31 or aspect 32, further comprising automatically generating a set of custom data descriptors when a new parameter is added to the parameters assigned to the controller.

[0161] 34. The method according to any one of aspects 24 to 30, wherein the device operable to transmit the data stream comprises a big data appliance, and wherein the device operable to receive the data stream is a device selected from the group consisting of: a workstation computer, a laptop computer, a smart phone, and a tablet computer.

[0162] 35. The method according to any one of aspects 24 to 30, wherein the device operable to transmit the data stream comprises a big data appliance, and wherein the device operable to receive the data stream is a cloud computing device.

[0163] 36. The method according to any one of aspects 24 to 35, wherein the first set of metadata comprises metadata describing a relationship of a first subset of data in the set of data to a second subset of data in the set of data.

[0164] 37. The method according to aspect 36, wherein the relationship is an upstream relationship.

- [0165]** 38. The method according to aspect 36, wherein the relationship is a downstream relationship.
- [0166]** 39. The method according to any one of aspects 24 to 38, further comprising performing an analysis on the set of data to generate analysis data.
- [0167]** 40. The method according to aspect 39, further comprising adding the analysis data to the data stream.
- [0168]** 41. The method according to any one of aspects 24 to 40, wherein the data stream comprises a timestamp.
- [0169]** 42. The method according to aspect 41, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the set of data.
- [0170]** 43. The method according to any one of aspects 24 to 42, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.
- [0171]** 44. A method for receiving a data stream in a process control system, the method comprising: receiving a first set of metadata from a device operable to transmit the data stream; receiving the data stream after receiving the first set of metadata; parsing the received data stream according to the first set of metadata; and continuing to receive streamed data as long as the data stream can be parsed according to the first set of metadata.
- [0172]** 45. The method according to aspect 44, further comprising: identifying in the data stream an ID that cannot be parsed according to the first set of metadata; caching the data stream; sending a request to the device operable to send the data stream to provide updated metadata; receiving the updated metadata; parsing the cached data stream according to the updated metadata; continuing to receive the data stream; and parsing the data stream according to the updated metadata.
- [0173]** 46. The method according to aspect 44, further comprising: receiving updated metadata; continuing to receive the data stream; and parsing the data stream according to the updated metadata, wherein the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the first set of metadata.
- [0174]** 47. The method according to any one of aspects 44 to 46, wherein the first set of metadata includes stream format description metadata describing the structure of the data stream.
- [0175]** 48. The method according to any one of aspects 44 to 47, wherein the first set of metadata includes data description metadata describing the content of the data stream.
- [0176]** 49. The method according to aspect 48, wherein the data description metadata includes standard data description metadata and custom data description metadata.
- [0177]** 50. The method according to any one of aspects 44 to 49, wherein the set of data includes parameter data, module and block data, event data, and alarm data.
- [0178]** 51. The method according to any one of aspects 44 to 50, wherein the device operable to transmit the data stream is a controller, and wherein the method is performed by a device comprising a big data appliance.

- [0179]** 52. The method according to aspect 51, wherein the controller is pre-configured with a set of standard data descriptors.
- [0180]** 53. The method according to either aspect 51 or aspect 52, wherein the controller automatically generates a set of custom data descriptors when a new parameter is added to the parameters assigned to the controller.
- [0181]** 54. The method according to any one of aspects 44 to 50, wherein the device operable to transmit the data stream comprises a big data appliance, and wherein the device performing the method is a device selected from the group consisting of: a workstation computer, a laptop computer, a smart phone, and a tablet computer.
- [0182]** 55. The method according to any one of aspects 44 to 50, wherein the device operable to transmit the data stream comprises a big data appliance, and wherein the device performing the method is a cloud computing device.
- [0183]** 56. The method according to any one of aspects 44 to 55, wherein the first set of metadata comprises metadata describing a relationship of a first subset of data in the set of data to a second subset of data in the set of data.
- [0184]** 57. The method according to aspect 56, wherein the relationship is an upstream relationship.
- [0185]** 58. The method according to aspect 56, wherein the relationship is a downstream relationship.
- [0186]** 59. The method according to any one of aspects 44 to 58, further comprising performing an analysis on the set of data to generate analysis data.
- [0187]** 60. The method according to aspect 59, wherein the analysis data are stored in a big data appliance.
- [0188]** 61. The method according to any one of aspects 44 to 60, wherein the data stream comprises a timestamp.
- [0189]** 62. The method according to aspect 61, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the set of data.
- [0190]** 63. The method according to any one of aspects 44 to 62, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.
- [0191]** 64. A controller for a process control system, the controller comprising: a processor; a memory communicatively coupled to the processor and storing a set of metadata; an input receiving data from one or more process control devices; one or more routines executing on the processor, according at least in part to the received data, to control the process; a collection routine executing on the processor and collecting data received from the one or more process control devices; a queuing routine executing on the processor to store the collected data in a buffer; a data streaming routine executing on the processor and cooperating with a communication device to: transmit the stored set of metadata to a receiving device; assemble the buffered data into a data stream according to the stored set of metadata; and transmit the data stream to the receiving device.

- [0192]** 65. The controller according to aspect 64, wherein the data streaming routine executing on the processor is further operable to: receive a request for updated metadata; generate or download updated metadata; and send the updated metadata in response to the request for updated metadata.
- [0193]** 66. The controller according to aspect 64, further comprising a metadata updating routine operable to receive or generate updated metadata, and wherein the data streaming routine executing on the processor is further operable to: send updated metadata in response to a detected change in a configuration of the controller or in response to the generation or reception of updated metadata; and continue sending the data stream, wherein the data stream sent after the updated metadata are sent is structured according to the updated metadata.
- [0194]** 67. The controller according to any one of aspects 64 to 66, wherein the stored set of metadata includes stream format description metadata describing the structure of the data stream.
- [0195]** 68. The controller according to any one of aspects 64 to 67, wherein the stored set of metadata includes data description metadata describing the content of the data stream.
- [0196]** 69. The controller according to aspect 68, wherein the data description metadata includes standard data description metadata and custom data description metadata.
- [0197]** 70. The controller according to any one of aspects 64 to 69, wherein the collected data includes parameter data, module and block data, event data, and alarm data.
- [0198]** 71. The controller according to any one of aspects 64 to 70, wherein the receiving device comprises a big data appliance.
- [0199]** 72. The controller according to aspect 71, further comprising a routine operative to automatically generate a set of custom data descriptors when a new parameter is added to the parameters assigned to the controller.
- [0200]** 73. The controller according to any one of aspects 64 to 70, wherein the receiving device is a device selected from the group consisting of: a workstation computer, a laptop computer, a smart phone, and a tablet computer.
- [0201]** 74. The controller according to any one of aspects 64 to 70, wherein the receiving device is a cloud computing device.
- [0202]** 75. The controller according to any one of aspects 64 to 74, wherein the stored set of metadata comprises metadata describing a relationship of a first subset of data in the collected data to a second subset of data in the collected data.
- [0203]** 76. The controller according to any one of aspects 64 to 75, further comprising an analysis routine executing on the processor to analyze the collected data to generate analysis data.
- [0204]** 77. The controller according to aspect 76, wherein the data streaming routine executing on the processor is further operable to add the analysis data to the data stream.
- [0205]** 78. The controller according to any one of aspects 64 to 77, wherein the data stream comprises a timestamp.

- [0206]** 79. The controller according to aspect 78, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the collected data.
- [0207]** 80. The controller according to any one of aspects 64 to 79, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.
- [0208]** 81. A big data appliance for storing big data in a process control system, the big data appliance comprising: a processor; a memory communicatively coupled to the processor; a high fidelity data storage device for storing big data; a receiver operable to: receive a set of metadata; receive a data stream; parse the data stream according to the set of metadata; store data in the high fidelity data storage device according to the parsing; continue to receive and store data as long as the data stream can be parsed according to the first set of metadata.
- [0209]** 82. The big data appliance according to aspect 81, wherein the receiver is further operable to: identify in the data stream an ID that cannot be parsed according to the set of metadata; cache the data stream; send a request to provide updated metadata; receive the updated metadata; parse the cached data stream according to the updated metadata; store the data in the high fidelity data storage device according to the parsing; continue to receive the data stream; parse the continued data stream according to the updated metadata; and store the data from the continued data stream in the high fidelity data storage device.
- [0210]** 83. The big data appliance according to aspect 81, wherein the receiver is further operable to: receive updated metadata; continue to receive the data stream; and parse the data stream according to the updated metadata, wherein the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the set of metadata.
- [0211]** 84. The big data appliance according to any one of aspects 81 to 83, wherein the set of metadata includes stream format description metadata describing the structure of the data stream.
- [0212]** 85. The big data appliance according to any one of aspects 81 to 84, wherein the set of metadata includes data description metadata describing the content of the data stream.
- [0213]** 86. The big data appliance according to aspect 85, wherein the data description metadata includes standard data description metadata and custom data description metadata.
- [0214]** 87. The big data appliance according to any one of aspects 81 to 86, wherein the data stream includes parameter data, module and block data, event data, and alarm data.
- [0215]** 88. The big data appliance according to any one of aspects 81 to 87, wherein the big data appliance is operable to stream big data to a downstream device.
- [0216]** 89. The big data appliance according to any one of aspects 81 to 88, wherein the set of metadata comprises metadata describing a relationship of a first subset of data in the data stream to a second subset of data in the data stream.
- [0217]** 90. The big data appliance according to any one of aspects 81 to 89, further comprising an analysis module operable to perform an analysis on the data in the data stream to generate analysis data.
- [0218]** 91. The big data appliance according to aspect 90, wherein the analysis data are stored in the high fidelity data storage device.

- [0219]** 92. The big data appliance according to any one of aspects 81 to 91, wherein the data stream comprises a timestamp.
- [0220]** 93. The big data appliance according to aspect 92, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the data stream.
- [0221]** 94. The big data appliance according to any one of aspects 81 to 93, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.
- [0222]** 95. An apparatus in a process control system, the apparatus comprising: a processor; a memory communicatively coupled to the processor and storing a set of metadata; a data source providing data to the apparatus; a queuing routine executing on the processor to buffer data received from the data source; a data streaming routine executing on the processor and cooperating with a communication device to: transmit the stored set of metadata to a receiving device; assemble the buffered data into a data stream according to the stored set of metadata; and transmit the data stream to the receiving device.
- [0223]** 96. The apparatus according to aspect 95, wherein the apparatus is a controller and the data source is an input receiving data from one or more process control devices, and further comprising: one or more routines executing on the processor to control the process according at least in part to the received data; and a collection routine executing on the processor and collecting data received from the one or more process control devices.
- [0224]** 97. The apparatus according to either aspect 95 or aspect 96, wherein the data streaming routine executing on the processor is further operable to: receive a request for updated metadata; generate or download updated metadata; and send the updated metadata in response to the request for updated metadata.
- [0225]** 98. The apparatus according to either aspect 95 or aspect 96, further comprising a metadata updating routine operable to receive or generate updated metadata, and wherein the data streaming routine executing on the processor is further operable to: send updated metadata in response to a detected change in a configuration of the apparatus or in response to the generation or reception of updated metadata; and continue sending the data stream, wherein the data stream sent after the updated metadata are sent is structured according to the updated metadata.
- [0226]** 99. The apparatus according to any one of aspects 95 to 98, wherein the stored set of metadata includes stream format description metadata describing the structure of the data stream.
- [0227]** 100. The apparatus according to any one of aspects 95 to 99, wherein the stored set of metadata includes data description metadata describing the content of the data stream.
- [0228]** 101. The apparatus according to any one of aspects 95 to 100, wherein the receiving device comprises a big data appliance.
- [0229]** 102. The apparatus according to any one of aspects 95 to 101, further comprising an analysis routine executing on the processor to analyze the data to generate analysis data.

- [0230]** 103. The apparatus according to aspect 102, wherein the data streaming routine executing on the processor is further operable to add the analysis data to the data stream.
- [0231]** 104. The apparatus according to any one of aspects 95 to 103, wherein the data stream comprises a timestamp.
- [0232]** 105. The apparatus according to aspect 104, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the data.
- [0233]** 106. The apparatus according to any one of aspects 95 to 105, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.
- [0234]** 107. An apparatus for receiving a stream of big data in a process control system, the apparatus comprising: a processor; a memory communicatively coupled to the processor; a non-transitory memory device; a receiver operable to: receive a set of metadata; receive a data stream; parse the data stream according to the set of metadata; process data according to the parsing; continue to receive and process data as long as the data stream can be parsed according to the first set of metadata.
- [0235]** 108. The apparatus according to aspect 107, wherein the apparatus is a big data appliance for storing big data, wherein the non-transitory memory device comprises a high fidelity data storage device, and wherein processing data comprises storing the data in the high fidelity data storage device.
- [0236]** 109. The apparatus according to either aspect 107 or aspect 108, wherein the receiver is further operable to: identify in the data stream an ID that cannot be parsed according to the set of metadata; cache the data stream; send a request to provide updated metadata; receive the updated metadata; parse the cached data stream according to the updated metadata; process the data according to the parsing; continue to receive the data stream; parse the continued data stream according to the updated metadata; and process the data from the continued data stream.
- [0237]** 110. The apparatus according to either aspect 107 or aspect 108, wherein the receiver is further operable to: receive updated metadata; continue to receive the data stream; and parse the data stream according to the updated metadata, wherein the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the set of metadata.
- [0238]** 111. The apparatus according to any one of aspects 107 to 110, wherein the set of metadata includes stream format description metadata describing the structure of the data stream.
- [0239]** 112. The apparatus according to any one of aspects 107 to 111, wherein the set of metadata includes data description metadata describing the content of the data stream.
- [0240]** 113. The apparatus according to any one of aspects 107 to 112, wherein the apparatus is operable to stream big data to a downstream device.
- [0241]** 114. The apparatus according to any one of aspects 107 to 113, further comprising an analysis module operable to perform an analysis on the data in the data stream to generate analysis data.
- [0242]** 115. The apparatus according to aspect 114, wherein the analysis data are stored in the non-transitory memory device.

- [0243]** 116. The apparatus according to any one of aspects 107 to 115, wherein the data stream comprises a timestamp.
- [0244]** 117. The apparatus according to aspect 116, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the data stream.
- [0245]** 118. The apparatus according to any one of aspects 107 to 117, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.
- [0246]** 119. A method according to any of the preceding method aspects, in which data are streamed and/or processed and/or stored at a second rate equal to or greater than a first rate at which the data are collected.
- [0247]** 120. A controller according to any of the aspects above, wherein the controller streams data at a second rate equal to or greater than a first rate at which the controller collects the data.
- [0248]** 121. A big data appliance according to any of the aspects above, wherein the big data appliance receives and/or processes and/or stores data in a data stream at a second rate equal to or greater than a first rate at which the data are generated in the process.
- [0249]** 122. Any one of the preceding aspects in combination with any one or more other preceding aspects.
- [0250]** When implemented in software, any of the applications, services, and engines described herein may be stored in any tangible, non-transitory computer readable memory such as on a magnetic disk, a laser disk, solid state memory device, molecular memory storage device, or other storage medium, in a RAM or ROM of a computer or processor, etc. Although the example systems disclosed herein are disclosed as including, among other components, software and/or firmware executed on hardware, it should be noted that such systems are merely illustrative and should not be considered as limiting. For example, it is contemplated that any or all of these hardware, software, and firmware components could be embodied exclusively in hardware, exclusively in software, or in any combination of hardware and software. Accordingly, while the example systems described herein are described as being implemented in software executed on a processor of one or more computer devices, persons of ordinary skill in the art will readily appreciate that the examples provided are not the only way to implement such systems.
- [0251]** Thus, while the present invention has been described with reference to specific examples, which are intended to be illustrative only and not to be limiting of the invention, it will be apparent to those of ordinary skill in the art that changes, additions or deletions may be made to the disclosed embodiments without departing from the spirit and scope of the invention.
- [0252]** When used in this specification and claims, the terms "comprises" and "comprising" and variations thereof mean that the specified features, steps or integers are included. The terms are not to be interpreted to exclude the presence of other features, steps or components.
- [0253]** The features disclosed in the foregoing description, or the following claims, or the accompanying drawings, expressed in their specific forms or in terms of a means for performing the disclosed function, or a

method or process for attaining the disclosed result, as appropriate, may, separately, or in any combination of such features, be utilised for realising the invention in diverse forms thereof.

16 08 21

CLAIMS

1. A method for receiving a data stream in a process control system, the method comprising:
 - receiving a set of metadata from a device operable to transmit the data stream, the device being a controller communicatively connected to one or more process control devices and executing one or more control routines to control a process according at least in part to data received from the one or more process control devices;
 - receiving, from the device, the data stream after receiving the set of metadata, the data stream including process control data comprising data received by the controller from the one or more process control devices and data indicative of a control signal generated by the one or more control routines;
 - parsing the received data stream according to the set of metadata; and
 - continuing to receive the data stream as long as the continued data stream can be parsed according to the set of metadata.

2. The method according to claim 1, wherein the set of metadata is a first set of metadata, and the method further comprises:
 - identifying in the data stream an ID that cannot be parsed according to the first set of metadata;
 - caching the data stream;
 - sending a request to the device operable to send the data stream to provide updated metadata;
 - receiving the updated metadata;
 - parsing the cached data stream according to the updated metadata;
 - continuing to receive the data stream; and
 - parsing the continued data stream according to the updated metadata.

3. The method according to claim 1 or 2, wherein the set of metadata is a first set of metadata, and the method further comprises:
 - receiving updated metadata;
 - continuing to receive the data stream; and
 - parsing the continued data stream according to the updated metadata,wherein the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the first set of metadata.

4. The method according to any of claims 1 to 3, wherein the set of metadata includes stream format description metadata describing the structure of the data stream.

5. The method according to any of claims 1 to 4, wherein the set of metadata includes data description metadata describing the content of the data stream.

6. The method according to any of claims 1 to 5, wherein the method is performed by a device comprising a big data appliance.

16 08 21

7. The method according to any of claims 1 to 6, further comprising performing an analysis on the set of data to generate analysis data.
8. The method according to claim 7, wherein the analysis data are stored in a big data appliance.
9. The method according to any of claims 1 to 8, wherein the data stream comprises a timestamp.
10. The method according to claim 9, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the set of data.
11. The method according to any of claims 1 to 10, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.
12. An apparatus for receiving a stream of big data in a process control system, the apparatus comprising:
 - a processor;
 - a memory communicatively coupled to the processor;
 - a non-transitory memory device;
 - a receiver operable to:
 - receive a set of metadata from a controller communicatively connected to one or more process control devices and executing one or more control routines to control a process according at least in part to data received from the one or more process control devices;
 - receive a data stream from the controller after receiving the set of metadata, the data stream including process control data comprising data received by the controller from the one or more process control devices and data indicative of a control signal generated by the one or more control routines;
 - parse the data stream according to the set of metadata;
 - process data according to the parsing;
 - continue to receive the data stream and parse the continued data stream according to the set of metadata as long as the continued data stream can be parsed according to the set of metadata.
13. The apparatus according to claim 12, wherein the apparatus is a big data appliance for storing big data, wherein the non-transitory memory device comprises a high fidelity data storage device, and wherein the receiver is further operable to store the processed data in the high fidelity data storage device.
14. The apparatus according to claim 12 or 13, wherein the set of metadata is a first set of metadata, and the receiver is further operable to:
 - identify in the data stream an ID that cannot be parsed according to the first set of metadata;
 - cache the data stream;
 - send a request to provide updated metadata;
 - receive the updated metadata;

parse the cached data stream according to the updated metadata;
process the data according to the parsing;
continue to receive the data stream;
parse the continued data stream according to the updated metadata; and
process the data from the continued data stream.

15. The apparatus according to any of claims 12 to 14, wherein the set of metadata is a first set of metadata, and the receiver is further operable to:

receive updated metadata;
continue to receive the data stream; and
parse the data stream according to the updated metadata,
wherein the data stream received after the updated metadata can be parsed according to the updated metadata, and could not be parsed according to the first set of metadata.

16. The apparatus according to any of claims 12 to 15, wherein the set of metadata includes stream format description metadata describing the structure of the data stream.

17. The apparatus according to any of claims 12 to 16, wherein the set of metadata includes data description metadata describing the content of the data stream.

18. The apparatus according to any of claims 12 to 17, wherein the apparatus is operable to stream big data to a downstream device.

19. The apparatus according to any of claims 12 to 18, further comprising an analysis module operable to perform an analysis on the data in the data stream to generate analysis data.

20. The apparatus according to claim 19, wherein the analysis data are stored in the non-transitory memory device.

21. The apparatus according to any of claims 12 to 20, wherein the data stream comprises a timestamp.

22. The apparatus according to claim 21, wherein the data stream comprises a plurality of incremental timestamps, each incremental timestamp associated with a sub-set of data in the data stream.

23. The apparatus according to any of claims 12 to 22, wherein the data stream comprises a plurality of identifiers, each identifier identifying one of a block, a module, a parameter, an event, or an alarm.

16 08 21