(12) UK Patent (19) GB (11) 2548021 (13) B

(45) Date of B Publication 31.03.2021

(54) Title of the Invention: Data pipeline for process control system anaytics

(51) INT CL: ***G05B 19/418*** (2006.01) ***G05B 19/042*** (2006.01)

(21) Application No: **1704515.4**

(22) Date of Filing: **05.10.2015**

Date Lodged: **22.03.2017**

(30) Priority Data:
(31) **62060408** (32) **06.10.2014** (33) **US**
(31) **14873721** (32) **02.10.2015** (33) **US**

(60) Parent of Application No(s)
2019649.9, 2019650.7 under section 15(9) of the Patents Act 1977

(86) International Application Data:
PCT/US2015/053931 En 05.10.2015

(87) International Publication Data:
WO2016/057365 En 14.04.2016

(43) Date of Reproduction by UK Office **06.09.2017**

(56) Documents Cited:
US 20140277604 A1 **US 20140250153 A1**
US 20130144405 A1 **US 20130144404 A1**
- KRUMEICH JULIAN ET AL, "Big Data Analytics for Predictive Manufacturing Control - A Case Study from Process Industry", 2014 IEEE INTERNATIONAL CONGRESS ON BIG DATA, IEEE, (20140627), doi:10.1109/BIGDATA.CONGRESS.2014.83, pages 530 - 537, XP032648959

(58) Field of Search:
As for published application 2548021 A viz:
INT CL **G05B, H04L**
Other: **EPO-Internal**
updated as appropriate

Additional Fields
Other: **None**

(72) Inventor(s):
Peter Zornio
Mark J Nixon
Wilhelm K Wojsznis
Michael J Lucas
Eric D Rotvold
Paul R Muston
Terrence L Blevins
Gary K Law

(73) Proprietor(s):
Fisher-Rosemount Systems Inc
1100 W Louis Henna Blvd Bldg 1, Round Rock, Texas 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

GB 2548021 B

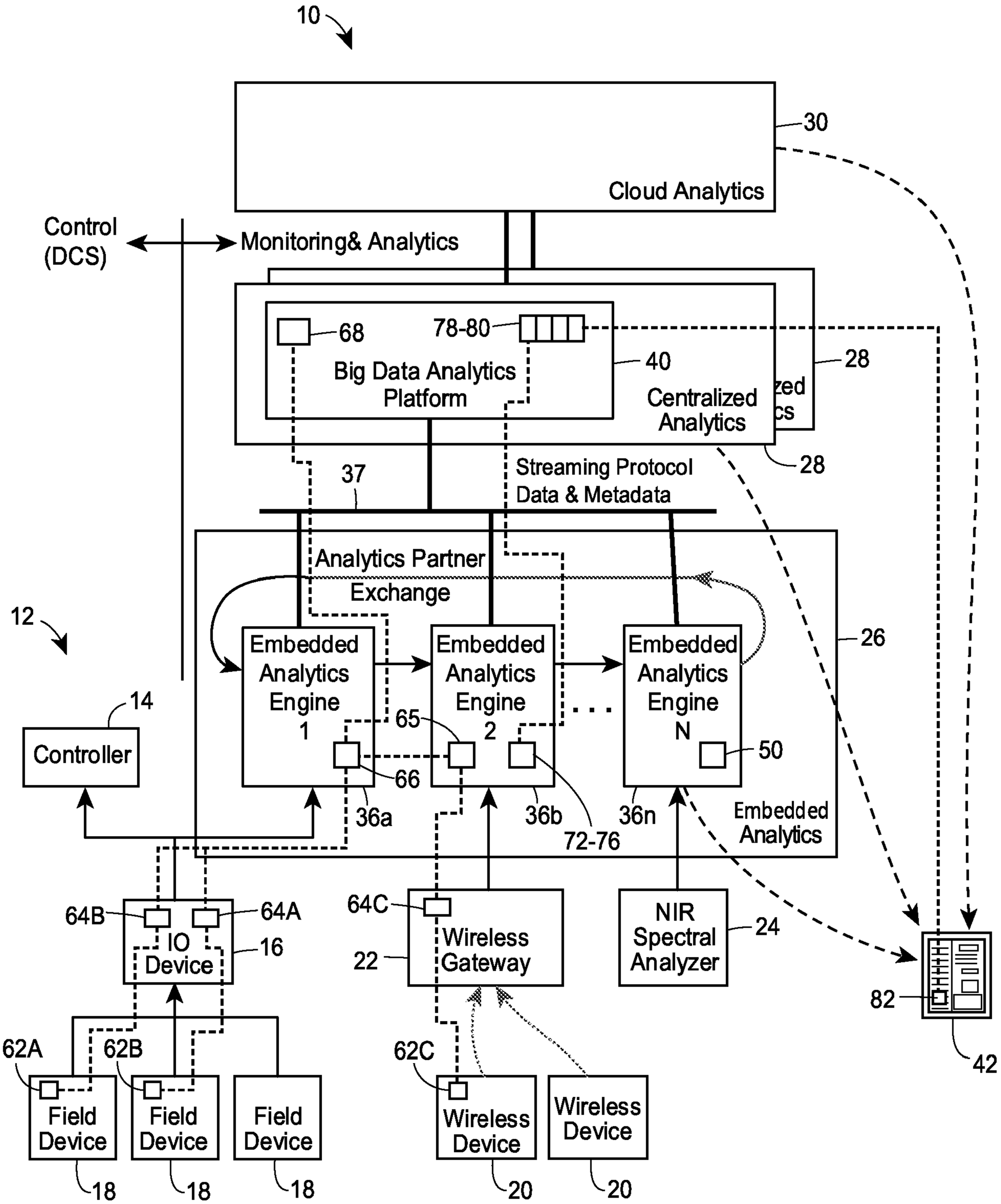

10
30
Cloud Analytics
Control
(DCS)
Monitoring& Analytics
68
78-80
Big Data Analytics
Platform
40
Centralized
Analytics
28
28
37
Streaming Protocol
Data & Metadata
Analytics Partner
Exchange
12
26
14
Controller
Embedded
Analytics
Engine
1
65
Embedded
Analytics
Engine
2
Embedded
Analytics
Engine
N
50
66
36a
36b
36n
72-76
Embedded
Analytics
64B
64A
IO
Device
16
64C
22
Wireless
Gateway
NIR
Spectral
Analyzer
24
82
42
62A
62B
Field
Device
Field
Device
Field
Device
18
18
18
62C
Wireless
Device
Wireless
Device
20
20

FIG. 1

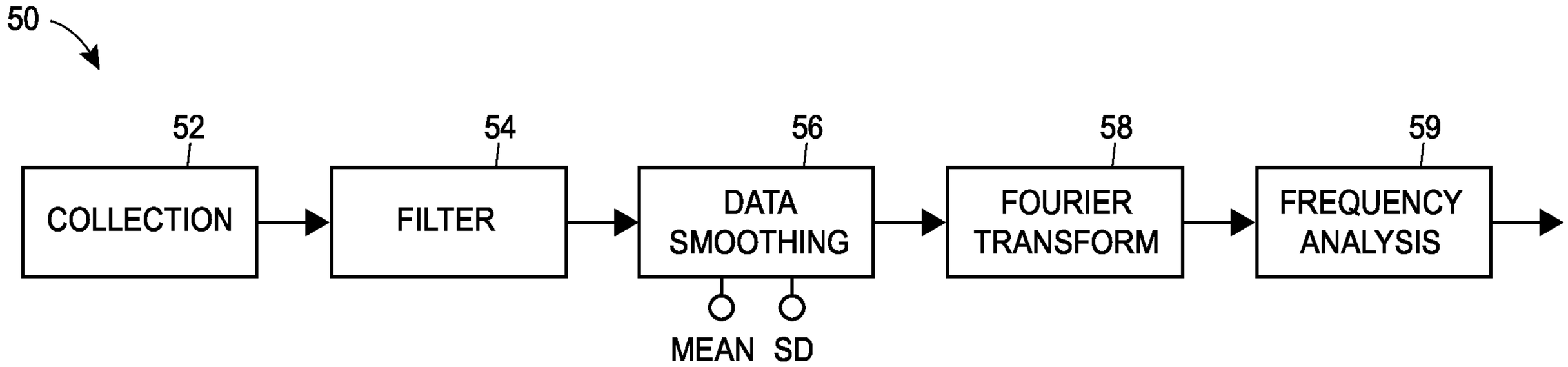

50
52
54
56
58
59
COLLECTION
FILTER
DATA SMOOTHING
FOURIER TRANSFORM
FREQUENCY ANALYSIS
MEAN
SD

FIG. 2A

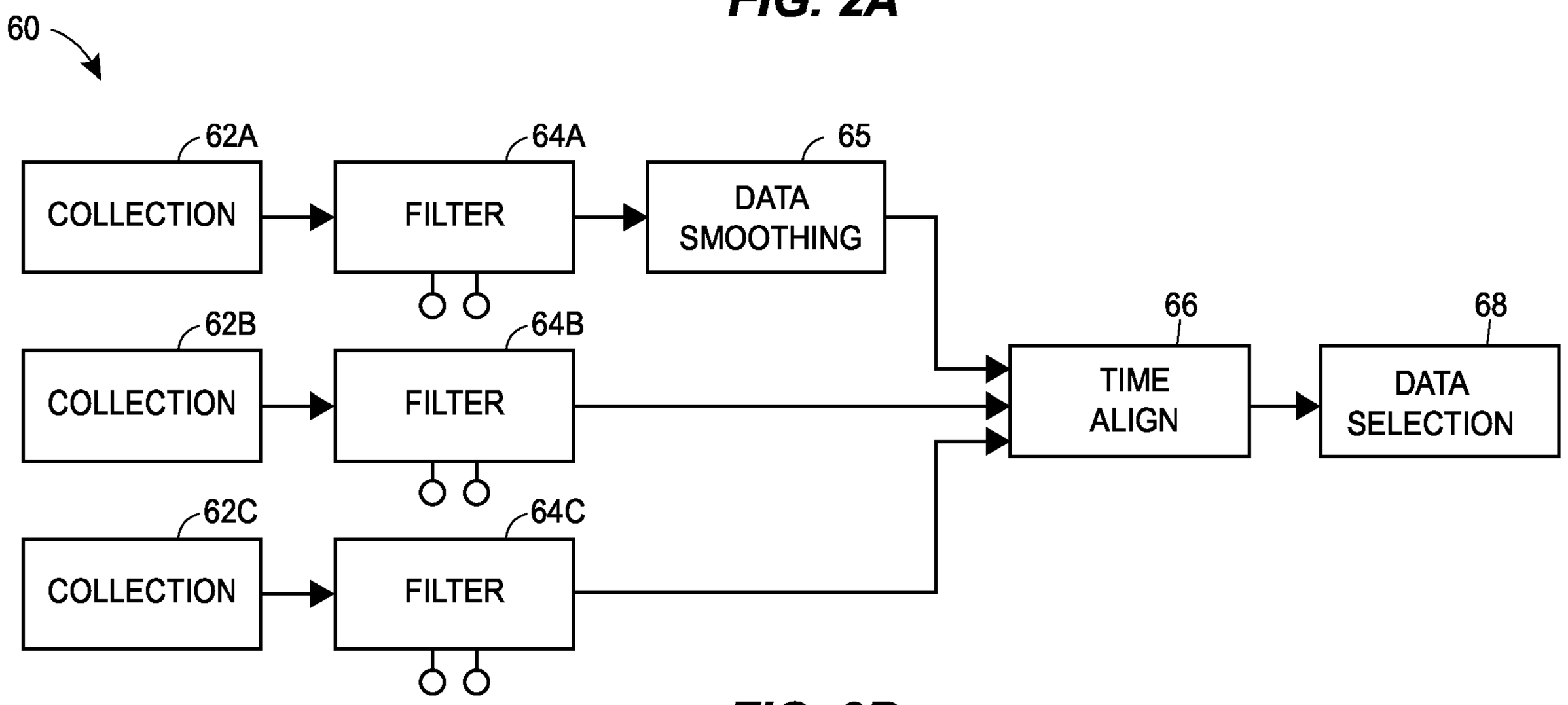

60
62A
64A
65
COLLECTION
FILTER
DATA SMOOTHING
62B
64B
COLLECTION
FILTER
62C
64C
COLLECTION
FILTER
66
68
TIME ALIGN
DATA SELECTION

FIG. 2B

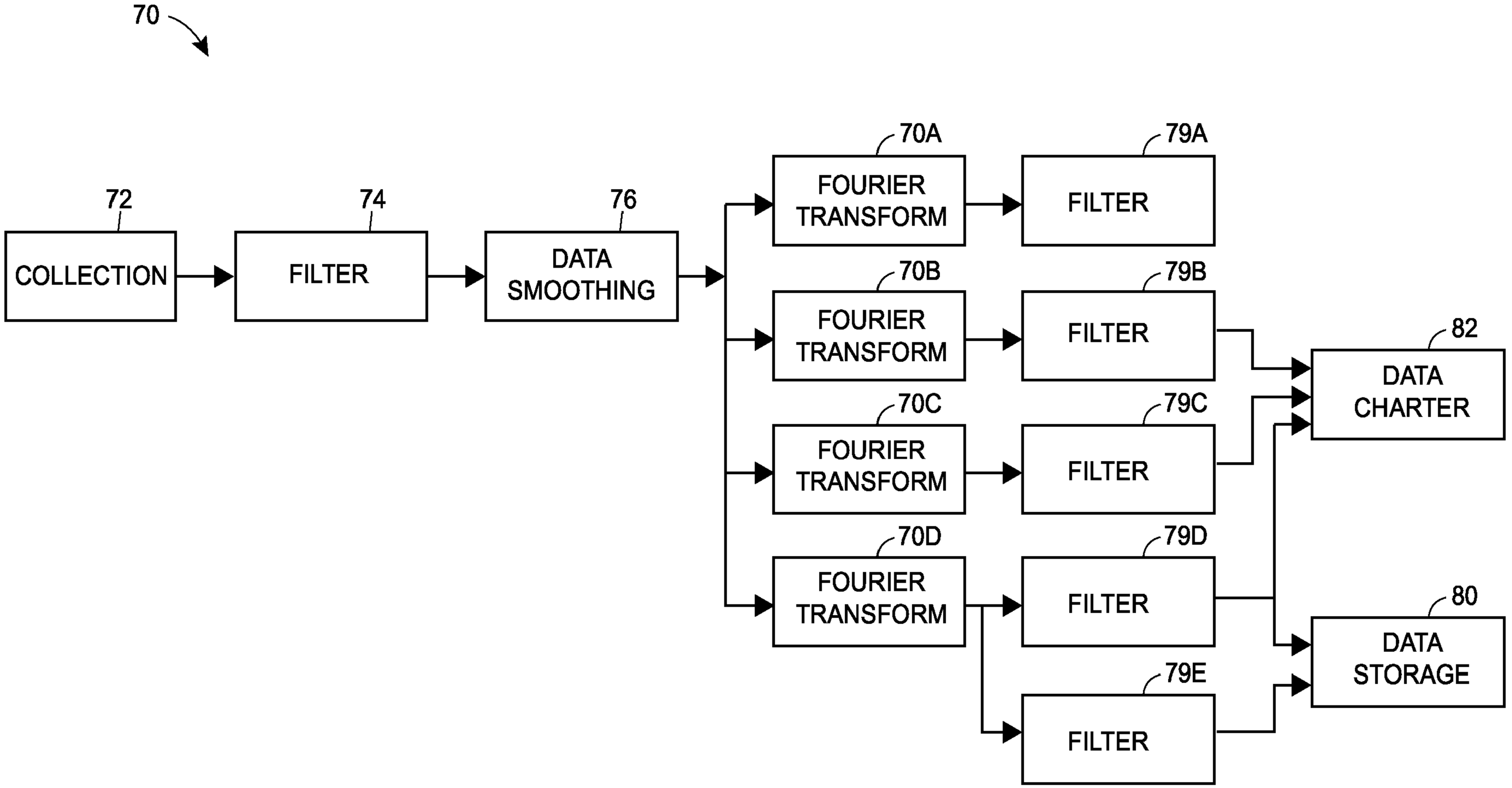

70
72
COLLECTION
74
FILTER
76
DATA SMOOTHING
70A
FOURIER TRANSFORM
79A
FILTER
70B
FOURIER TRANSFORM
79B
FILTER
70C
FOURIER TRANSFORM
79C
FILTER
70D
FOURIER TRANSFORM
79D
FILTER
79E
FILTER
82
DATA CHARTER
80
DATA STORAGE

FIG. 2C

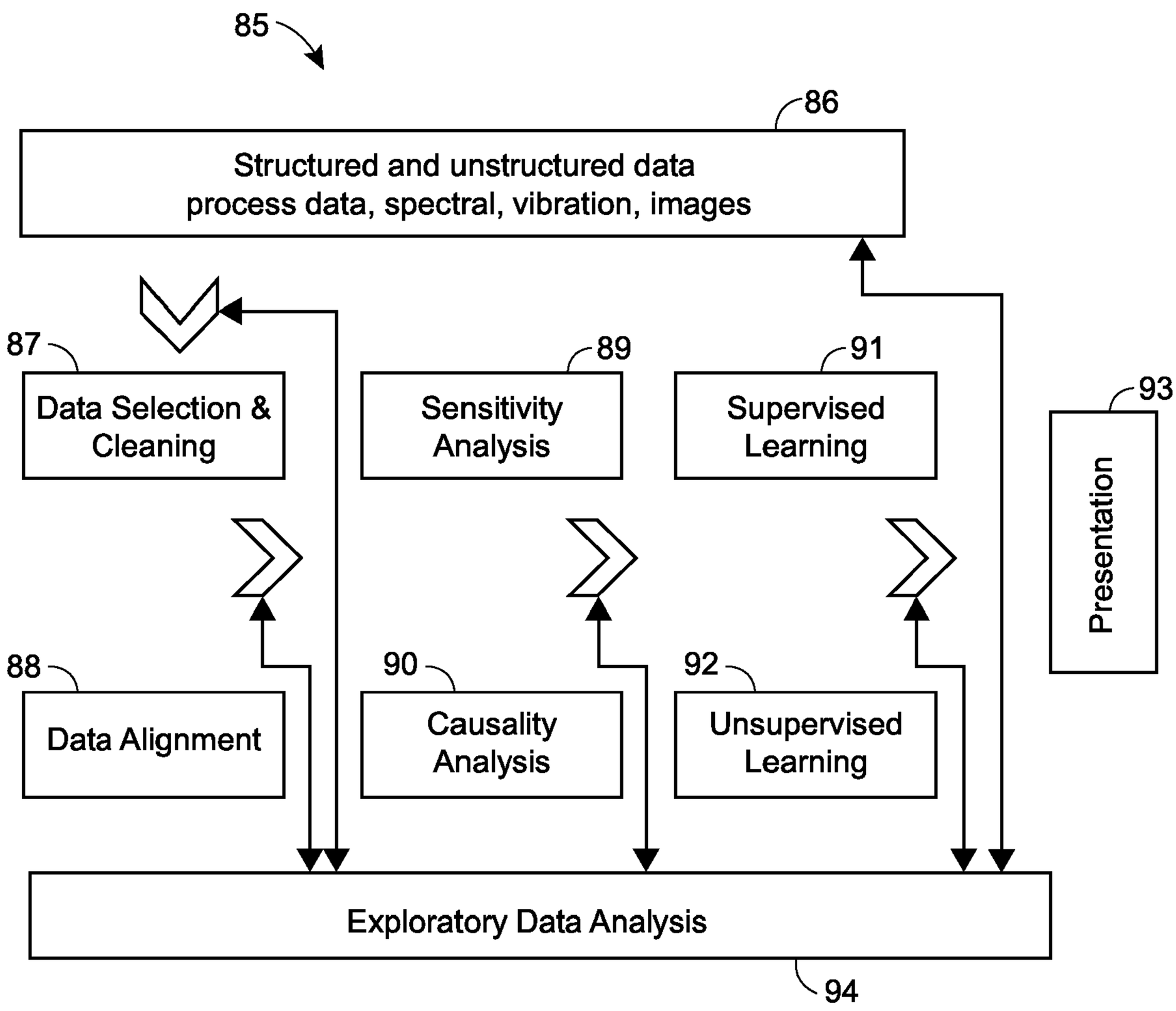

85
86
Structured and unstructured data
process data, spectral, vibration, images
87
Data Selection &
Cleaning
89
Sensitivity
Analysis
91
Supervised
Learning
93
Presentation
88
Data Alignment
90
Causality
Analysis
92
Unsupervised
Learning
Exploratory Data Analysis
94

FIG. 3

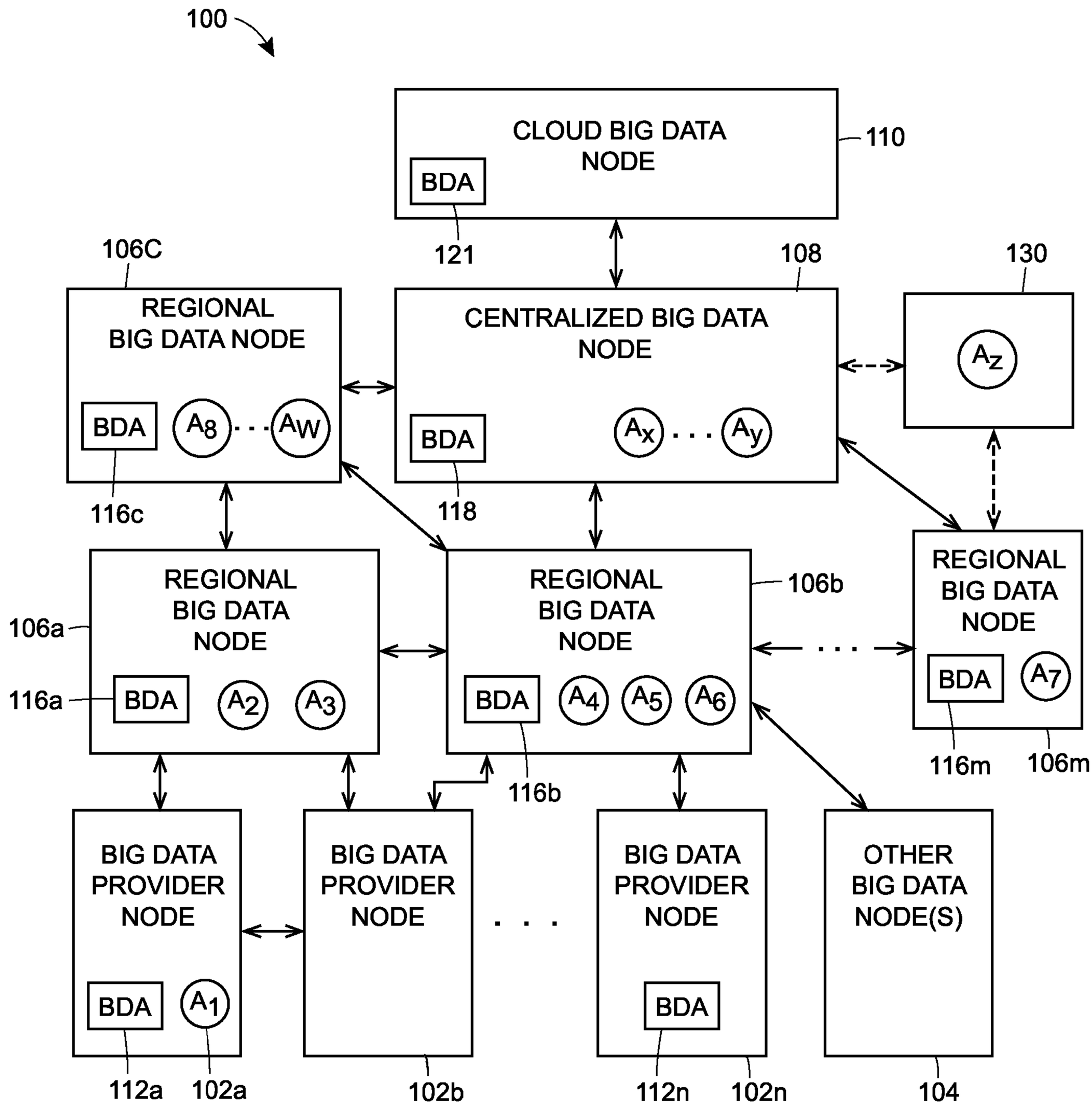

100
CLOUD BIG DATA NODE
110
BDA
121
106C
REGIONAL BIG DATA NODE
BDA
A8
AW
116c
108
CENTRALIZED BIG DATA NODE
BDA
Ax
Ay
118
130
Az
106a
REGIONAL BIG DATA NODE
116a
BDA
A2
A3
REGIONAL BIG DATA NODE
106b
BDA
A4
A5
A6
116b
REGIONAL BIG DATA NODE
BDA
A7
116m
106m
BIG DATA PROVIDER NODE
BDA
A1
112a
102a
BIG DATA PROVIDER NODE
102b
BIG DATA PROVIDER NODE
BDA
112n
102n
OTHER BIG DATA NODE(S)
104

FIG. 4

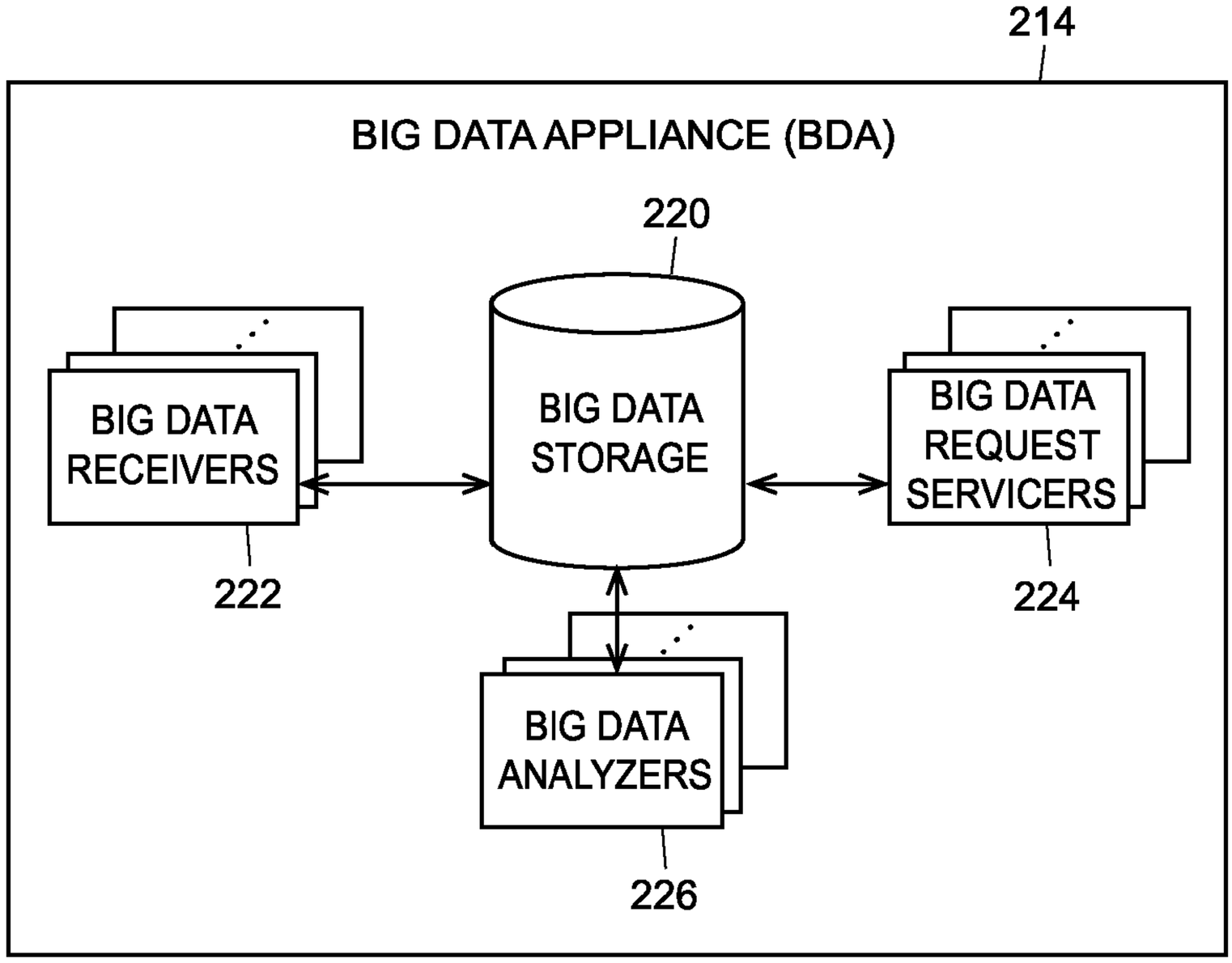

214
BIG DATA APPLIANCE (BDA)
220
BIG DATA RECEIVERS
222
BIG DATA STORAGE
BIG DATA REQUEST SERVICERS
224
BIG DATA ANALYZERS
226

FIG. 5

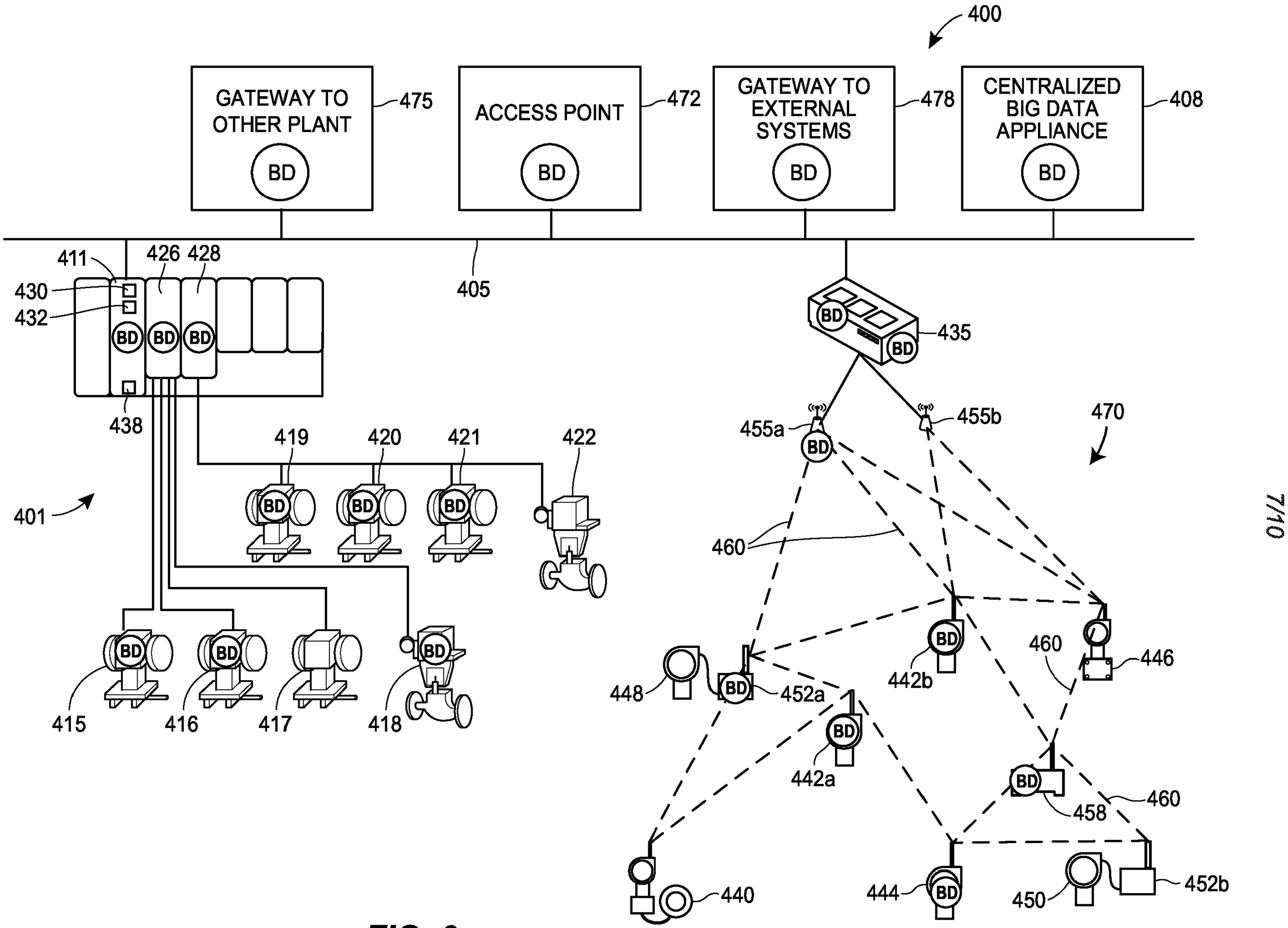

400
GATEWAY TO OTHER PLANT
475
ACCESS POINT
472
GATEWAY TO EXTERNAL SYSTEMS
478
CENTRALIZED BIG DATA APPLIANCE
408
BD
405
411
426
428
430
432
438
435
455a
455b
470
419
420
421
422
401
460
415
416
417
418
448
452a
442b
446
442a
458
440
444
450
452b

FIG. 6

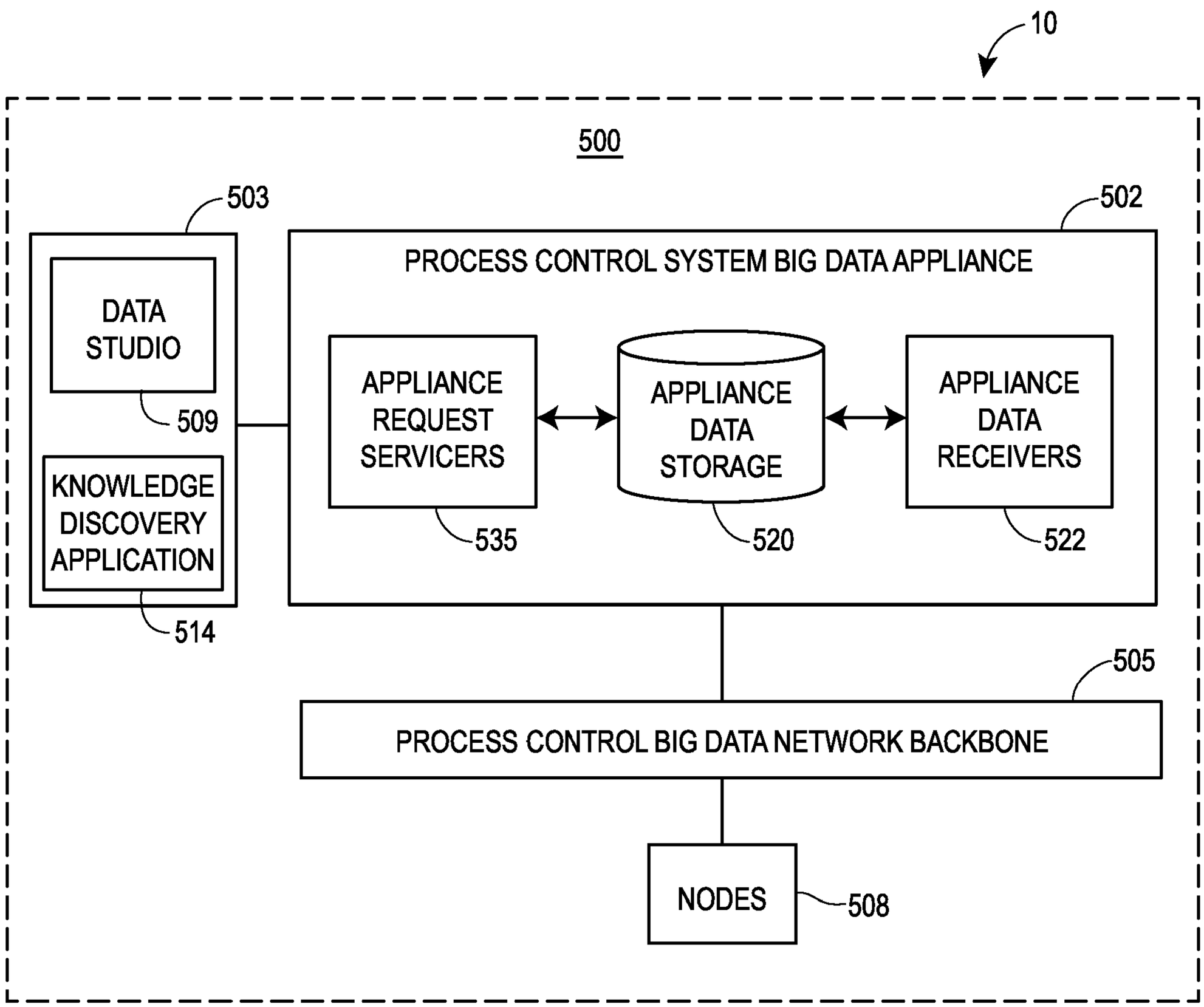

10
500
503
DATA STUDIO
509
KNOWLEDGE DISCOVERY APPLICATION
514
502
PROCESS CONTROL SYSTEM BIG DATA APPLIANCE
APPLIANCE REQUEST SERVICERS
535
APPLIANCE DATA STORAGE
520
APPLIANCE DATA RECEIVERS
522
505
PROCESS CONTROL BIG DATA NETWORK BACKBONE
NODES
508

FIG. 7

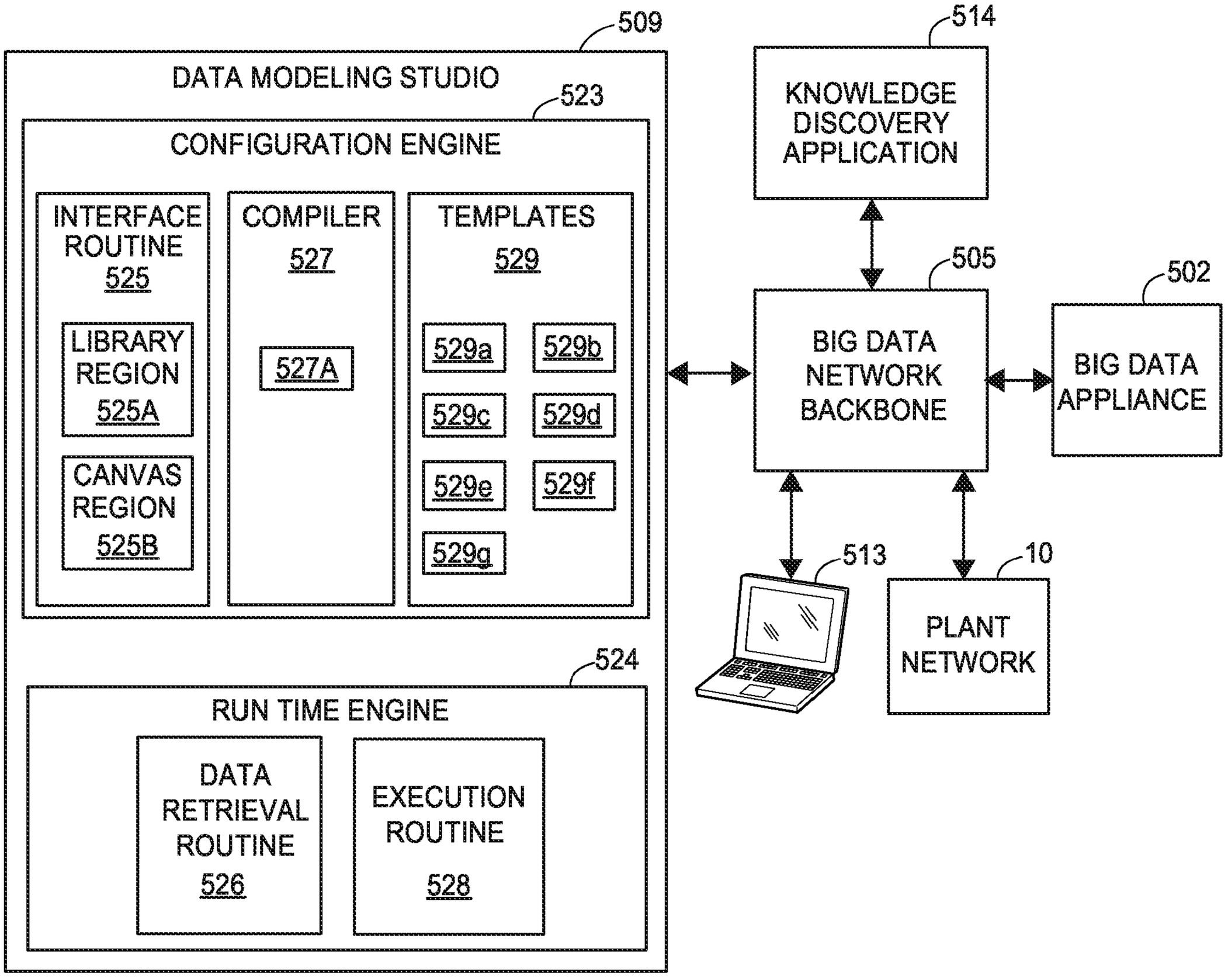

509
DATA MODELING STUDIO
523
CONFIGURATION ENGINE
INTERFACE ROUTINE 525
LIBRARY REGION 525A
CANVAS REGION 525B
COMPILER 527
527A
TEMPLATES 529
529a
529b
529c
529d
529e
529f
529g
524
RUN TIME ENGINE
DATA RETRIEVAL ROUTINE 526
EXECUTION ROUTINE 528
514
KNOWLEDGE DISCOVERY APPLICATION
505
BIG DATA NETWORK BACKBONE
502
BIG DATA APPLIANCE
513
10
PLANT NETWORK

FIG. 8

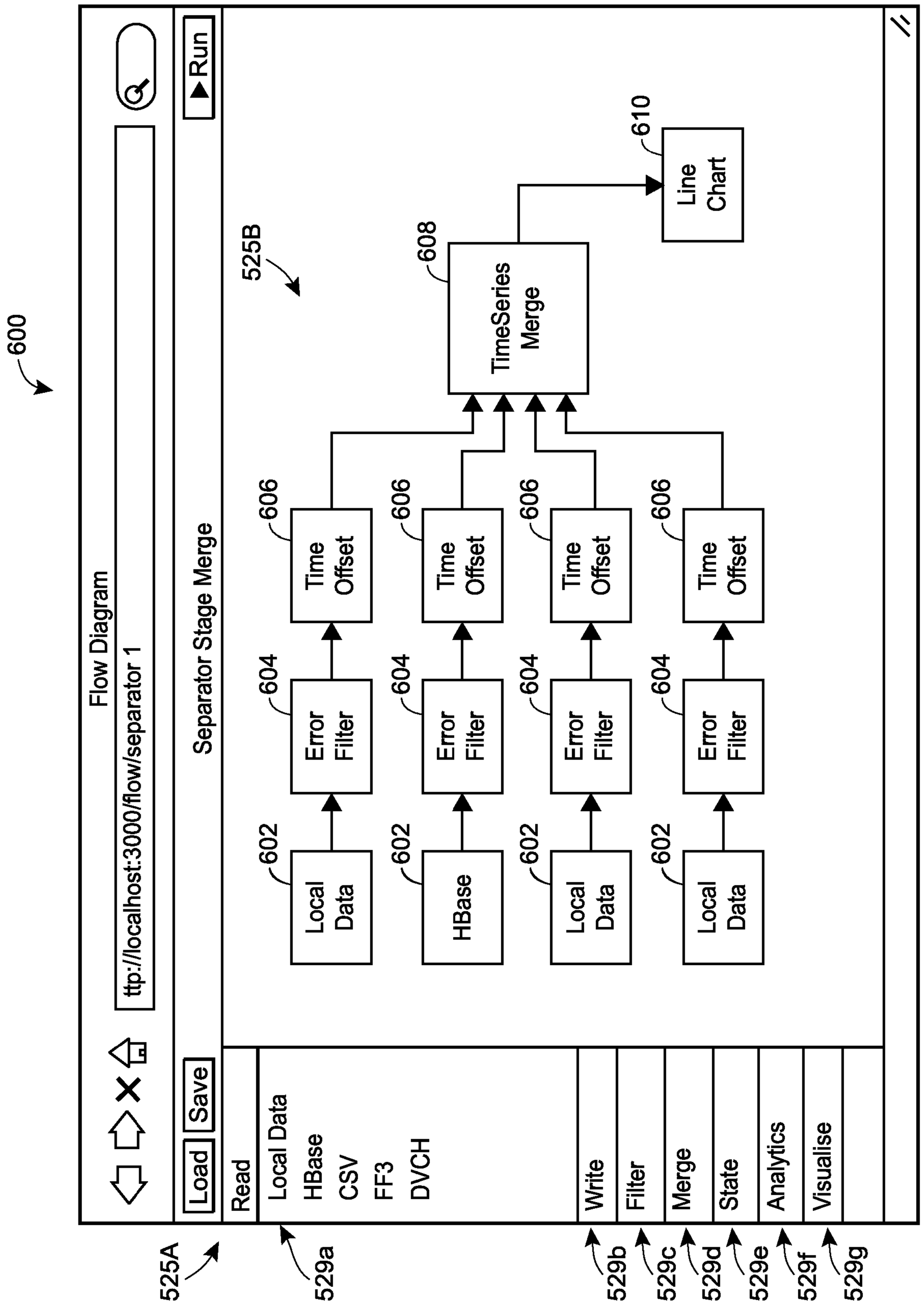

600
Flow Diagram
ttp://localhost:3000/flow/separator 1
Run
Load
Save
Separator Stage Merge
525A
Read
Local Data
HBase
CSV
FF3
DVCH
529a
Write
Filter
Merge
State
Analytics
Visualise
529b
529c
529d
529e
529f
529g
525B
608
TimeSeries Merge
610
Line Chart
606
Time Offset
604
Error Filter
602
Local Data
HBase

FIG. 9

DATA PIPELINE FOR PROCESS CONTROL SYSTEM ANALYTICS

[0001] The present application is also related to US2014/0250153 (U.S. Patent Application No. 13/784,041), entitled "BIG DATA IN PROCESS CONTROL SYSTEMS" and filed March 3, 2013; US2014/0280497 (U.S. Patent 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; US2014/0277656 (U.S. Patent 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; US2014/0277604 (U.S. Patent Application No. 14/212,493), entitled "DISTRIBUTED BIG DATA IN A PROCESS CONTROL SYSTEM" and filed March 14, 2014; US2015/0261215 (U.S. Patent Application No. 14/212,411), entitled "DETERMINING ASSOCIATIONS AND ALIGNMENTS OF PROCESS ELEMENTS AND MEASUREMENTS IN A PROCESS" and filed March 14, 2014; and US2014/0278312 (U.S. Patent Application No. 14/216,823), entitled "DATA MODELING STUIDO" and filed March 17, 2014, the entire disclosures of each of which are hereby expressly incorporated by reference herein.

20 14 12

[0002] Additionally, the present application is related to US 2016/0098021 (U.S. Patent Application No. 14/507,188), entitled "REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS" and filed concurrently herewith; US2016/0098647 (U.S. Patent Application No. 14/507,252), entitled "AUTOMATIC SIGNAL PROCESSING-BASED LEARNING IN A PROCESS PLANT" and filed concurrently herewith; and US2016/0098388 (U.S. Patent Application No.14/506,863), entitled "STREAMING DATA FOR ANALYTICS IN PROCESS CONTROL SYSTEMS" and filed concurrently herewith, the entire disclosures of each of which are hereby expressly incorporated by reference herein.

[0003] This patent relates generally to control and monitoring systems such as those used in process plants, and more particularly, to installing and using a data pipeline to process data in a distributed manner in monitoring and control systems.

[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 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, 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,

20 14 12

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 wired communication paths, wireless communication paths, or a combination of wired and wireless communication paths.

[0006] As an example, the DeltaV™ control system, sold by Emerson Process Management, 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.

20 14 12

[0007] In a process plant or process control system or other monitoring system, when evidence of an abnormal condition or fault occurs (e.g., when an alarm is generated, or when a process measurement or actuator is found to have excessive variation), an operator, instrument technician or process engineer typically uses an analytics tool in combination with his or her knowledge of the process being controlled by the system and its flow path through the system to attempt to determine upstream measurements and process variables that may have contributed to the production of the evidence of the abnormal condition or fault. For example, an operator may feed a historical log of data that has been captured over time from the output of a process control device (e.g., a field device, a controller, etc.) into the DeltaV™ batch analytics product or continuous data analytics tool to attempt to determine the contributions of various process variables and/or measurements to an abnormal or fault condition. Typically, a user decides which historical data logs to feed into the analytics tool and identifies candidate upstream factors (e.g., measurements, process variables, etc.) based on his or her knowledge of the process. Subsequently, these data analytics tools utilize, for example, principal component analysis (PCA) to determine which of the candidate upstream factors impact downstream predicted quality parameters. Thus, the accuracy and effectiveness of the output provided by the analytics tool is based on or limited to the user's knowledge, and as such may not provide complete or correct insight into the sources of the abnormal condition or fault.

220

141

[0008] Additionally, 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.

20 14 12

SUMMARY

[0011] Aspects of the invention are recited by the appended claims.

[0012] A system and method for providing distributed signal processing-based data processing and analysis to determine, for example, potential sources of faults, abnormal operations, and/or variations in the behavior of signals generated by controlling a process in a process plant includes forming and installing one or more data pipelines within various interconnected devices of the process plant or process control system, wherein the data pipelines automatically collect and process data in predetermined manners to perform partial or complete data analysis. The data pipelines may be used to implement signal processing-based learning which may be, for example, a type of big data based learning. Generally speaking, the systems and methods that use the data pipelines may automatically perform data processing in real time or on data being collected real-time in a process plant, or on data that is stored in a database, such as a big data machine. In some cases, some of the signal processing performed using the data pipeline architecture is integrated into the process plant, and is performed close to the source of the one or more signals in real-time as the data is generated by the signal source or sources. The data on which signal processing is performed by the data pipeline architecture may be, for example, indicative of a value over time of an

output signal of a process control device, a process variable, a measurement of a physical parameter such as temperature, pressure, flow rate, etc., a balance of energy, a balance of mass, a performance parameter, an output of an analytics function, and/or any other value that is generated based on the process being controlled in the process plant. Still further, the data pipeline architecture may process other types of data than process control data, such as monitoring data (produced by a monitoring system that does not perform control), device data indicative of device operational parameters, maintenance data used by maintenance personnel to service, repair and replace devices within a network, communication data, etc.

[0013] Moreover, the data pipeline architecture may be used in a system that performs big data based learning in, for example, a process plant network, and the data processing pipeline may include multiple data processing modules communicatively connected together to perform preconfigured processing operations on data in a specific order to produce processed data. The data processing pipeline may perform data processing operations including, for example, data collection, data selection and cleaning, data alignment, sensitivity analysis, causality analysis, supervised learning, and/or unsupervised learning. These separate modules or blocks may be located in the same or in different devices within the plant network and may be communicatively connected together through one or more different communication networks within the plant network.

[0014] The data pipeline or pipelines may be generated in a data modeling studio or configuration system in which a user selects various data processing modules to be applied to particular process data, and connects the modules together to form a data pipeline having multiple processing modules connected together in a particular order. The user may test the data pipeline on stored historical data, such as data stored in a big data appliance, and after testing is complete, may download the data pipeline to various processors, nodes or devices of a plant network to implement the data processing in real time or on-real time data as that data is collected or generated by the plant devices. Moreover, the data modeling studio configuration system may enable the user to specify the node, device or other location of each of the data processing pipeline elements when downloaded to the plant network.

[0015] In one embodiment, a data pipeline runs or extends through various (e.g., two or more) communication networks within or associated with a process plant including a process control network and a further communication network. More particularly, a plant communication system for use within a process plant environment that implements a process includes a process control network including a multiplicity of process control devices

20 14 12

disposed within the process plant to control the process and a process control communication network communicatively coupled to the multiplicity of process control devices, wherein one or more of the multiplicity of process control devices collects or generates process control data. The system also includes a further plant communication network including a plurality of processing devices communicatively coupled together by a further plant communication network link, wherein at least one of the plurality of processing devices on the further plant communication network is communicatively connected to the process control communication network. In this system, a distributed data pipeline is disposed within the plant communication system and the further plant communication network and the distributed data pipeline includes a series of data processing modules communicatively connected together in a particular communicative order to perform processing on process control data in the particular communicative order such that the data pipeline includes one or more upstream data processing modules coupled to a set of downstream data processing modules. Each of the set of downstream data processing modules within the data pipeline processes data received from one or more upstream data processing modules within the data pipeline, and at least one of the one or more upstream data processing modules of the data pipeline is disposed in one of the multiplicity of process control devices connected to the process control communication network and at least one of the downstream data processing modules of the data pipeline is disposed in one of the plurality of processing devices on the further plant communication network.

20 14 12

[0016] The system may further include a database communicatively coupled to one of the downstream data processing modules of the data pipeline for storing the data generated by the one of the downstream data processing modules of the data pipeline and making the stored data available to other applications via the further plant communication network. If desired, one of the upstream data processing modules is disposed in the process control network and collects raw process control data for processing by the data pipeline, and a database is communicatively coupled to the process control communication network for independently storing the collected raw process control data. The database may be disposed in one of the processing devices coupled to the further plant communication network, which may be a big data communication network. The further plant communication network may a plurality of different communication networks coupled together, including a first plant based communication network and a second cloud based communication network. In this case, the data pipeline may include at least one different data processing module stored in a processing

device within each of the plurality of different communication networks. The different communication networks may include a wireless and/or a wired communication network, and the process control communication network may include a plurality of different control communication networks communicatively coupled together.

[0017] If desired, the data processing modules may be or include any of a data collection module that obtains data from a data source within the process control network, a data cleaning module that cleans data obtained from the data source within the process control network, a data alignment module that time aligns data from multiple data sources within the process control network, a sensitivity data analysis module that determines the sensitivity of data to a particular event within the plant, a causality analysis module that determines a causal relationship between data from a data source and an event within the plant, a learning module that performs supervised or unsupervised learning on process data, or a data presentation module that presents data generated within the data pipeline to a user.

[0018] Still further, the plant communication system may be constructed such that the data pipeline includes one or more data collection modules that collect process data from the process control network, one or more intermediate data processing modules that process the data collected by one or more of the data collection modules or another intermediate data processing module, and one or more end-use data processing modules that store output data generated by the one or more intermediate data processing modules or the one or more end-use data processing modules in a manner that can be accessed by one or more other applications. Still further, one or more of the data processing modules of the data pipeline may be intermittently connected within the data pipeline, either to other data processing modules within the data pipeline or to applications subscribing to data from the data pipeline. For example, one or more intermittently connected data processing modules may be data display modules that display data produced within the data pipeline to a user, these modules may be disposed in a handheld device that is intermittently connected to the further plant communication network, etc.

[0019] Additionally, the output data of a data pipeline may be made to be simultaneously accessible to two or more other applications. Still further, the data pipeline modules may execute at any desired execution rate, which rate may be different for different modules within the data pipeline. Likewise, any of the data pipeline modules may execute at a rate faster than, slower than or equal to the execution rate of a process controller that is controlling the process from which process data is being collected and analyzed within the

20 14 12

data pipeline. Here, one of the multiplicity of process control devices is a process controller that executes one or more control routines at a controller execution rate, and the process control data collected by the one or more upstream data processing modules of the data pipeline disposed in one of the multiplicity of process control devices may be collected at a rate greater than the controller execution rate. Additionally, any number (e.g., two or more) of the data processing modules within the data pipeline may process collected data at a rate greater than the controller execution rate.

[0020] In another embodiment, a data pipeline is disposed in one or more communication networks of a process plant that processes data at a rate faster than the controller execution rate of the process control system of the process plant. In particular, a plant control and communication system for controlling a process within a plant includes a multiplicity of process control devices disposed within the plant to control the process including a process controller that executes to control the process at an execution rate. The plant control and communication system also includes a plurality of data processing devices disposed within the plant and on or more communication networks that communicatively couple the multiplicity of process control devices and the plurality of data processing devices. Here, one or more data sources is disposed within the plant and these data sources are distributed among the multiplicity of process control devices to collect raw data from the process control devices within the plant at a collection data rate greater than the execution rate of the controller. Still further, a database is communicatively coupled to the one or more communication networks for storing the collected raw data from the one or more data sources at the collection data rate and a data pipeline is disposed in the one or more of the plurality of processing devices within the plant, the data pipeline including a series of data processing modules communicatively connected together in a particular communication order, wherein at least one of the data processing modules of the data pipeline performs a data processing operation on raw data from one of the one or more data sources, and at least another one of the data processing modules performs data processing on data received from one of the data processing modules within the data pipeline. Here, at least two of the data processing modules operate at a rate faster than the execution rate of the process controller.

[0021] In this system, the at least two of the data processing modules may operate at a rate faster than the execution rate of the process controller by communicating data between the two data processing modules at a rate faster than the execution rate of the process controller

20 14 12

and/or by performing processing on data at a rate faster than the execution rate of the process controller.

[0022] In another embodiment, multiple distributed data pipelines are disposed in a process plant to process data within the plant and to feed processed data to higher level analytic applications. In one example, a process plant communication system for use within a process plant environment that implements a process includes a process control network having a multiplicity of process control devices disposed within the plant to control the process and a process control communication network communicatively coupled between the multiplicity of process control devices, wherein one or more of the multiplicity of process control devices collects or generates process control data. Moreover, a further plant communication network is disposed in the plant and includes a plurality of processing devices communicatively coupled together by a further plant communication network link, wherein at least one of the plurality of processing devices on the further plant communication network is communicatively connected to the process control communication network. A plurality of distributed data pipelines is disposed in the system, each data pipeline including a series of different data processing modules communicatively connected together to perform processing on process control data in a particular sequence, wherein each data pipeline includes one or more data collection modules that collect process data from the process control network, one or more intermediate data processing modules that process the data collected by one or more of the data collection modules or another intermediate data processing module, and one or more ending data processing modules that produce data pipeline output data using data generated by the one or more intermediate data processing modules. Here, at least two data processing modules of each data pipeline are located in different devices within the process control communication network and the further plant communication network. The system further includes a set of higher level analytic applications that is communicatively connected within the further plant communication network, wherein each of the set of higher level analytic applications subscribes to data generated by the ending data processing module of one or more of the distributed data pipelines.

[0023] If desired, one or more of the set of higher level analytic applications may be intermittently connected to one or more of the distributed data pipelines and the data processing modules of one of the distributed data pipelines may process data and communicate data within the one of the distributed data pipelines during on-line operation of the process. If desired, one or more of the higher level analytic applications may subscribe to

20 14 12

data (output data or end-use data) from two or more of the distributed data pipelines, and one or more of the higher level analytic applications may be disposed in a handheld device that is connected to the further plant network via a wireless communication connection.

[0024] In another embodiment, a method of processing data in a plant may be used in a plant having a process control network including a multiplicity of process control devices disposed within the plant to control a process and a process control communication network communicatively coupled between the multiplicity of process control devices, wherein one or more of the multiplicity of process control devices collects or generates process control data, and the plant control network includes a further plant network including a plurality of processing devices communicatively coupled together by a further plant communication network link. The method may include storing a distributed data pipeline within the process control network and in the further plant network, wherein storing the distributed data pipeline includes storing a series of data processing modules that are configured to be communicatively connected together in a particular order to perform processing on process control data in various different ones of the process control devices and processing devices. The method may also include collecting data at one or more of the data processing modules of the data pipeline during on-line operation of the process control network and processing the collected process data in the data processing modules of the distributed data pipeline during on-line operation of the process control network. Still further, the method may include storing the processed data developed by the data processing modules of the distributed data pipeline in a computer memory device and using the data developed by the data processing modules of the distributed data pipeline in a further application executed on a further processing device.

[0025] If desired, storing the processed data developed by the data processing modules of the distributed data pipeline in a computer memory device may include storing the processed data in a big data machine. Moreover, storing the distributed data pipeline within the process control network and in the further plant network may include storing at least one of the data processing modules within one of the multiplicity of process control devices and storing another one of the data processing modules in one of the processing devices of the further plant network. In addition, storing the distributed data pipeline within the process control network and in the further plant network may include storing at least one of the data process modules in a device within a first plant sub-network of the further plant network and storing another one of the data process modules in a second plant sub-network of the further plant

20 14 12

network. Here, the first plant sub-network may be, for example, a big data network in the plant and the second plant sub-network may be, for example, a cloud based network.

[0026] Again, if desired, collecting data at one or more of the data processing modules of the data pipeline during on-line operation of the process control network may include collecting data at a rate faster than an exaction rate of a process controller used within the process control network to perform on-line process control.

[0027] If desired, the data pipeline may include one or more data collection modules that collect data from the process control network, one or more intermediate data processing modules that process the data collected by one or more of the data collection modules or another intermediate data processing module, and one or more end-use data processing modules that store output data generated by the one or more intermediate data processing modules or the end-use data processing modules in a manner that can be accessed by one or more other applications. For example, one of the data processing modules of the data pipeline may be intermittently connected within the data pipeline and/or the intermittently connected data processing module may be disposed in a handheld device that is intermittently connected to the further plant communication network.

20 14 12

[0028] In another embodiment, a method of configuring a data processing pipeline is used within a process plant having a process control network with a multiplicity of process control devices that operate to control the process and having a process communication network having a multiplicity of processing devices communicatively coupled together. The configuration method includes storing a set of data processing modules within different ones of the multiplicity of process control devices and the processing devices, communicatively interconnecting the data processing modules in a fixed manner to create a set of three or more data processing modules communicatively connected in series, and configuring the set of three or more data processing modules connected in series to perform specific types of data processing on data collected in the plant. The step of configuring may include configuring one or more of the series of data processing modules within process control devices within the process control network as data collection modules to collect raw process data, configuring one or more of the series of data processing modules as intermediate process modules to communicate with one or more data collection modules and to process the raw process data collected by the data collection modules to create intermediate process data, and configuring one or more of the series of data processing modules as end-use process modules to communicate with one or more of the intermediate data processing modules to process the

intermediate process data to generate end-use process data. Additionally, the configuration method may include configuring one or more of the end-use data process modules to provide the end-use process data to one or more subscribing applications communicatively connected to the process communication network.

[0029] If desired, the method of configuring a data processing pipeline may include configuring the one or more of the series of data processing modules stored within process control devices within the process control network as data collection modules by configuring the one or more of the series of data collection modules to collect the raw process data at a rate faster than a controller execution rate of a process controller that performs process control within the process control network. Still further, configuring one or more of the series of data processing modules as intermediate data process modules may include configuring the one or more intermediate data process modules to process the raw process data at a rate faster than the execution rate of the process controller. Also, configuring one or more of the series of data processing modules as end-use process modules may include configuring the one or more of the end-use process modules to process the intermediate process data at a rate faster than the execution rate of the process controller.

20 14 12

[0030] The method of configuring a data processing pipeline within a process plant may include storing a set of data processing modules within different ones of the multiplicity of process control devices and the processing devices by storing some of the data processing modules in the process control network devices and others of the data processing modules in the process communication network devices. The method of configuring a data processing pipeline within a process plant may also include storing a set of data processing modules within different ones of the multiplicity of process control devices and the processing devices by storing different data processing modules to perform processing on process data in a particular sequence, wherein the stored set of data processing modules includes one or more upstream data collection modules that collect process data from the process control network, one or more intermediate process modules communicatively connected downstream from the one or more data collection modules or one or more other intermediate process modules that process the data collected by one or more of the data collection modules or another intermediate process module, and one or more end-use process modules communicatively coupled downstream from the one or more intermediate process modules that produce the end-use data using the intermediate data from the one or more intermediate process modules.

[0031] Likewise, configuring one or more of the series of data processing modules to process the process data to generate intermediate process data may include configuring the one or more intermediate data processing modules to receive data from two or more upstream data processing modules and to combine the data from the two or more upstream data processing modules to create the intermediate process data and may include configuring the one or more intermediate process modules to provide the intermediate process data to two or more downstream data processing modules. Configuring one or more of the end-use process modules to process the intermediate process data to generate end-use process data may include configuring the one or more end-use process modules to provide a stream of end-use data to multiple different applications and/or to provide end-use data to one or more intermittently connected applications. Configuring the set of three or more data processing modules connected in series to perform specific types of data processing on data collected in the plant may also include configuring two of the data processing modules to be intermittently communicatively connected together.

[0032] As an example, the method of configuring a data processing pipeline within a process plant may include storing the set of data processing modules within different ones of the multiplicity of process control devices and the processing devices by storing different ones of the communicatively coupled data processing modules in different communication networks including in the process control network and the process communication network. This storing step may also include any or all of storing one of the data processing modules of the data pipeline in a process control field device, storing one of the data processing modules in a process controller, storing one of the data processing modules in a regional big data analytics machine, storing one of the data processing modules in a cloud based data analytical machine, and/or storing one of the data processing modules in a handheld device.

BRIEF DESCRIPTION OF THE DRAWINGS

[0033] Fig. 1 is a block diagram of a plant network including process control devices, monitoring devices and big data appliance devices, in which data pipelines may be created and installed.

[0034] Figs 2A-2C are block diagrams of various different example data pipelines which may be created and installed in the plant network of Fig 1.

[0035] Fig. 3 is a block diagram illustrating a set of data processing functions that can be combined in various manners to create one or more data pipelines in a plant network.

[0036] Fig. 4 is a block diagram of an example big data network for a process plant or process control and monitoring system which may support automatic or autonomous signal processing-based learning using data pipelines.

[0037] Fig. 5 is a block diagram of an example big data appliance of Fig. 4.

[0038] Fig. 6 illustrates a further block diagram of an example process plant or process control system in which automatic or autonomous signal processing-based learning based on data pipelines may be performed.

[0039] Fig. 7 illustrates a block diagram of an example of a process plant or process control system that includes a big data machine that collects and stores data from a plant, and a data studio that may be used to create and execute data pipelines that implement one or more data models that use or analyze the data collected from the plant.

[0040] Fig. 8 is a block diagram of an example embodiment of the data modeling studio of Fig. 6 as connected within a generic plant environment.

[0041] Fig. 9 is a screen presentation of a graphical user interface of the data modeling studio of Figs. 7 and 8 that may be used to create one or more data pipelines for testing and implementation in a real-time plant network.

20 14 12

DETAILED DESCRIPTION

[0042] Techniques, systems, and methods for automatically or autonomously performing signal processing-based learning in a process plant or process control system include the creation and use of one or more data pipelines as foundational structures upon which the signal processing-based learning is performed in real time or in an on-line plant network. Generally, these techniques, systems and methods enable signal processing to be automatically or autonomously performed on or applied to data (e.g., signals) that are generated as a result of the operation of a control system or a monitoring system within a process plant, as well as being based on maintenance activities or maintenance data, monitoring data, device data, business or planning data, etc. Typically, the data or the signals on which the data pipelines operate are indicative of a parameter value that varies over time based on the control, maintenance, monitoring, etc. of the process. Thus, as used herein, the term "signal" generally refers to a time series of a parameter value, e.g., a time series of an output of a controller, a time series of a disturbance variable, a time series process variable measurement, a time series of a monitored or measured variable, etc. In contrast with known process control signal processing techniques in which a user obtains a historical signal feed

from a process plant *a posteriori* and provides the historical signal feed into an off-line, separate signal processor, the techniques, systems, and methods described herein used data pipelines to enable one or more signal processing functions to be integrated into the process plant, maintenance system, monitoring, system or process control system so that signal processing is performed autonomously (e.g., as a background process) in real-time as the signal(s) is/are generated by one or more signal sources in real-time. Thus, using the techniques, systems, and methods disclosed herein, signal processing is incorporated into 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 monitoring system and enabling the results of these signal processing activities to be used by other analytic modules or higher analytic applications within the plant. Indeed, in some embodiments, one or more signal processing functions may be integral or integrated with a signal source while, in other cases, the signal processing functions may be located in more centralized devices. In any event, the integrated or incorporated signal processing functions may be configured to autonomously and automatically execute on the real-time signals generated by signal sources as a result of the real-time operations of the on-line process plant or monitoring system.

220

141

[0043] It is noted that the techniques, systems and methods discussed herein may be applied to single signal, to multiple signals respectively, and/or to multiple signals as a group or as a whole. However, for ease of discussion herein and not for limitation purposes, the singular term "signal" is utilized.

[0044] The use of one or more data pipeline structures to perform one or more signal processing functions (e.g., the application of the one or more signal processing functions to a real-time signal) may provide results that indicate one or more characteristics of a signal, that indicate other information about a signal or a group of signals, etc. In still another case, the use of data pipelines may be implemented to perform data mining and data learning in addition to data processing or signal conditioning. For example, one or more dominant frequencies or harmonics that are present in the signal may be determined by the application of the one or more signal processing functions within a data pipeline. In some process plants or process control systems, the characteristic(s) of the signal as determined by data processing performed by a data pipeline are provided to one or more analytics functions that also operate automatically or autonomously during the real-time operations of the process plant. The analytics function(s) may determine one or more sources of the characteristic(s) of

the signal, e.g., based on an element alignment map or similar resource that indicates or identifies elements (e.g., process elements) that are upstream, in the process, of the signal source. In these or other cases, data learning may be performed using a data pipeline to identify other types of knowledge about the process plant, the control system, the monitoring or monitored system, etc.

[0045] Generally speaking, data pipelines are used as elemental structures in a distributed processing system implemented within a plant to clean, filter, process, and perform learning and analytics on data from various sources in a plant or other monitoring system. In particular, one or more data pipelines may be established to define specific processing to be performed on one or more streams of data received from various sources, such as from control system resources, maintenance system resources, field devices, controllers, monitoring systems, lab analyses, etc. Each data pipeline is made up of multiple different processing blocks or data processing modules which operate on certain signals or data (e.g., a stream of data) provided thereto and produce one or more data streams that are then provided to a downstream block or module within the data pipeline. Generally speaking, the initial inputs to the data pipeline will be one or more sets of raw data, such as control signals, process variable measurement signals (e.g., pressure, temperature, flow, etc. data), monitoring signals, device data, maintenance data, etc. Moreover, each block within a data pipeline will process the data input thereto (e.g., received from one or more other data processing modules in the data pipeline) in some manner to, for example, clean the data, combine the data, fill in missing data, perform filtering on the data, perform learning or analytics on the data, etc. As the data flows through the various different data processing modules of the data pipeline, the data pipeline creates new data (which may be intermediate data or end-use data) which may be exposed and be made available to other data consumers, including other data processing modules within a data pipeline, other data pipelines, other analytic modules such as higher level analytic applications. Moreover, the outputs of a data pipeline (referred to herein as end-use data) may represent or provide analytic results that may be used to, for example, set alarms, perform better maintenance or control, provide graphs and charts to a user to illustrate some operation of the plant or monitoring system, or any other knowledge gleaned from the data, as processed within the data pipeline. Still further, various different modules or blocks of a data pipeline may be implemented in different devices within a plant network or other network and data may flow between the different blocks of a data pipeline in any desired manner and at any desired speed or rate.

20 14 12

Generally speaking, data processing modules within a data pipeline may take the form of (1) data collection modules that collect data of some sort from the plant or process, such as raw process data, controller data, I/O data, maintenance data, etc., (2) intermediate process modules which are communicatively connected to (that is, downstream of) one or more data collection modules or other intermediate process modules, and which operate to analyze (e.g., combine, process, filter, etc.) collected data or other data provided thereto from other intermediate process modules to produce intermediate data, and (3) end-use process modules which may be communicatively connected to (i.e., downstream of) one or more data collection modules or intermediate process modules and which use the data from these other modules to create processed data (end-use data) that is provided to one or more data consumers, such as user interfaces, higher level analytic applications, databases, etc. If desired, the data pipelines may operate to generate end-use data in real time during operation of the process and may provide end-use data to data consumers constantly, periodically, or intermittently. That is, end-use applications may connect to and remain connected to a data pipeline to receive end-use data from a data pipeline in a consistent or constant basis, or may periodically or intermittently connect to a data pipeline to receive end-use data when these applications or users need that data. Still further, data within the data pipeline may be collected and processed at any desired rate, such as faster than the execution rate of one or more process controllers that are controlling the process. In this case, the data pipelines within the process plant operate independently of process controllers (even though data pipelines may use or collect data generated by the process controllers).

[0046] Thus, as will be understood, data processing and knowledge learning techniques may be automatically or autonomously implemented within a plant network with the use of one or more data pipelines which, through their operation, implement specific actions on data that is collected, generated or measured within the plant to thereby enable perform specific types of higher level data processing and learning. Each data pipeline defines a series of operations or functions to be performed on a particular set of data or sets of data in a particular (e.g., fixed) order to thereby perform some overall processing that leads to knowledge about the plant or learning within the plant. In other words, each data pipeline implements a series of predetermined data processing functions in a particular, predefined order, to perform or implement one or more data models and/or to implement data learning within the plant. Typically, although not necessarily, the process plants in which at least portions of the data pipelines are utilized include a process control big data network and

20 14 12

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. Application No. 13/784,041, in aforementioned U.S. Patent Application No. 14/174,413, in aforementioned U.S. Application No. 14/212,493, and/or in aforementioned U.S. Application No. 14/507,188 , entitled "REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS." As such, a process plant that performs automatic or autonomous signal processing-based learning using data pipelines 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.

[0047] In one example system, individual data pipelines may be wrapped in analytics modules which may be tagged, distributed, and used for generating predictions, specialized visualizations, alarms and alerts within a plant control or monitoring setting. That is, analytics modules may combine or use one or more data pipelines to perform analytics on plant data using the outputs of or the exposed data produced by the data pipelines. The outputs of the analytics blocks may then be used to perform predictions, monitor for faults, and provide recommendations on-line in real-time (for example, to provide operators and maintenance workers with recommendations throughout their shifts as things evolve within the plant). As will be understood, the data pipelines may be used to collect, store, process and combine time-series, asset, machine health data, monitored data, etc. in any desired manner to produce any desired or useful output.

[0048] As an example, a data pipeline architecture may be used in a plant network such as the plant network 10 illustrated in Fig. 1. The plant network 10 of Fig. 1 is illustrated as including a control system 12 having a controller 14 coupled via an input/output device 16 to various field devices 18, which may be valves, sensors, transmitters, or any other types of devices that measure process, device or plant parameters and/or that perform some physical control action within a plant. While the field devices 18 are illustrated in Fig. 1 as being wired field devices, they can be wireless devices as well or instead and may be configured in any desired manner to perform control of plant assets, systems, fluid flows, etc. Still further, the plant network 10 includes monitoring devices or other field devices or measurement devices 20 which are illustrated as being wireless devices connected within the plant network 10 via a wireless gateway device 22. The devices 20 can be any type of devices, such as maintenance devices, monitoring devices, sensors, etc. Moreover, the devices 20 may be

20 14 12

wireless or wired devices and may communicate within the plant network 10 using any desired communication protocol and/or communications network. Still further, Fig. 1 illustrates a data processing device in the form of a spectral analyzer 24 coupled to the plant network 10. Here, the spectral analyzer 24 may periodically perform analysis on samples of the plant, or other data provided thereto, to provide or generate information or data about some aspect of the plant operation. Of course, the spectral analyzer 24 is but an example of a testing or analysis device that can be used in the plant network 10 to provide information or data about the operation of the plant, control system, monitored system, etc. Moreover, it will be understood that any other types of data providers can be coupled to the plant network 10, including devices that provide off-line lab measurements or analyses, other types of analyzers, such as signal analyzers of any type.

[0049] As illustrated in Fig. 1, the plant network 10 also includes various levels of analytic structure including one or more embedded analytics layers 26, also referred to herein as regional big data nodes, centralized analytic layers 28, and cloud based analytic layers 30. Each of the regional, centralized and cloud based analytic layers 26, 28, 30, include one or more analytic engines or signal processing engines that may be executed on one or more processors, computers, servers, etc. at these layers in the plant network 10. As illustrated in Fig. 1, the regional embedded analytics layers or blocks 26 may include one or more analytic engines 36a-36n which may be embedded within devices in the plant 10 closer (in a logical or communicatively connected manner) to the control system 12 and monitoring system devices (e.g., 20, 22, 24, etc.) While the analytic engines 36a-36n may be implemented in controllers, field devices, gateway devices, etc., these engines may also or instead be implemented in stand-alone processing devices that, for example, implement big data machines. The analytic engines 36a-36n may be connected together via a streaming protocol bus 37 and may be connected to various ones of the control devices 14, 16, 18, of the control system 12, to the monitoring devices 20, 24 or to other devices within the system, such as the spectrum analyzer 24 via various different buses or communication networks which may use the same or different communication protocols. In some cases, the embedded analytic engines 36a-36n may be embedded within the control and monitoring devices or other devices, or may be stand-alone devices or chips within devices that perform data analytics.

[0050] As illustrated in Fig. 1, the embedded analytics engines 36a-36n are connected together and are connected to analytics platforms 40 in one or more of the centralized analytics layers or blocks 28 via an analytics bus 37. The analytics platforms 40 in the

20 14 12

centralized layers or centralized big data nodes 28 may be a big data platform having a big data appliance as will described in more detail herein. The analytics engines 36a-36n and the big data platforms 40 may store the data from the devices in the control, monitoring and analytics machines at the lower levels of the plant network 10 of Fig. 1. Still further, the big data appliances of the centralized layers 28 may be connected to data analytics within the cloud based analytic block 30. While not illustrated in Fig. 1, the cloud based analytic block 30 may include any number of analytic devices and big data devices that perform data analytics on the data provided thereto or available thereto via the other levels or blocks of Fig. 1.

[0051] Still further, as illustrated in Fig. 1, one or more handheld devices 42 may connect wirelessly to any of the analytic machines, for example, any of the devices in the analytic blocks 26, 28 and 30, to perform data analytics. Thus, for example, one or more handheld devices 42 may interface with the analytic engines 36a-36n, with the big data appliances 40 and/or with servers or other machines in the cloud analytics block 30 to perform data analytics using data pipelines. While not explicitly illustrated in Fig. 1, the handheld devices 42 may connect to other devices to acquire data, such as to the field devices 18, monitoring devices 20, controllers 14, I/O devices 16, gateways 22, data processing devices 24, etc.

[0052] As will be understood, the lower tiers of Fig. 1 illustrate that data is collected from several data sources (I/O devices, I/O buses, wireless and wired field devices, near-infrared spectroscopy devices, and equipment monitoring systems, for example). Although a portion of the data may be sent to traditional controllers 14, the bulk of the data is streamed to the embedded analytics engines 36a-36n in the regional big data nodes (i.e., the embedded analytics layers 26) of the network 10. The embedded analytics engines 36a-36n cache the data and begin processing locally in real-time, and the embedded analytics engines 36a-36n may include both learning and runtime execution capability. Because there are many types of analytics, fault detection, and recommendation systems possible, the analytics engines 36a-36n may be capable of executing a wide range of algorithms. As part of this capability, the analytics engines 36a-36n may include specialized signal processing and learning chips (TI signal processing and IBM neuro-learning chips, for example). Such a combination of hardware and software may be utilized to perform predictions, fault detection, and provide recommendations. As part of this system, the embedded analytics could begin identifying leading indicators for the equipment they are monitoring and then work with other embedded analytics engines to identify the parameters that have the greatest impact on those leading

20 14 12

indicators. Once trained, the embedded analytics modules 36a-36n could then be downloaded where they would then be used to monitor the process, assets, and equipment. A unique feature is that the analytics engines 36a-36n can continuously learn in a manner defined by a data pipeline, and such learning could be cached for consideration at a later date at any of the big data devices or nodes.

[0053] Still further, as illustrated in Fig. 1, the data analytics can be executed or run in the handheld devices 42. In fact, with the development of new chipsets with built-in analytics capabilities, the handheld units 42 may be one of the best places to actually run analytics. With this factor in mind, the handheld devices 42 may pull data from the lower lever devices such as controllers 14, field devices 18, monitoring devices 20, and from higher level devices, such as embedded analytics engines 36a-36n, centralized big data servers 40, and the cloud layer 30, to perform specialized processing. In such a case, specialized applets could be developed for specific equipment, devices, processes, companies, etc. and these applets could be executed on the handheld devices 42.

20 14 12

[0054] As noted above, the strategies sent to the embedded analytics engines 36a-36n, for example, may include one or more analytics blocks chained together into a data pipeline. For example a first block in the data pipeline may be used to filter certain data (e.g., controller data, field device data, measurement data, monitoring data, device data, etc.), a second block in the data pipeline may be used to address missing data within the filtered set of data (e.g., find the missing data, fill in the data based on interpolation or other techniques, etc.), a third block in the data pipeline may be used to time align the data from the second block, a fourth block in the data pipeline may be used to select parameters from an array or matrix of the aligned data, a fifth block in the data pipeline may be used to perform fault detection from the selected parameters, and a final block in the data pipeline may be used to perform predictions on the fault detection data. Each data pipeline may be hosted by a module that includes alarms/alerts and that may manually adjustable alarm limits, for example.

[0055] Of course, each data pipeline could include various different or various different types of data processing blocks that may be used to perform processing on data collected within the plant 10. For example a block or module within a data pipeline could include or perform 1) data cleaning (to remove noise and inconsistent data), 2) data integration (where multiple data sources may be combined), 3) data selection (where data relevant to an analysis task are retrieved from the database), 4) data transformation (where data are transformed and consolidated into forms usable by downstream algorithms or blocks), 5) data mining (an

essential process where intelligent methods are applied to extract patterns within data), 6) pattern evaluation (to identify the truly interesting patterns representing knowledge based on interestingness measures), etc. Additionally, knowledge presentation and predictions may be generated by one or more data pipelines or blocks within data pipelines. In some cases, for example data pipelines may include visualization techniques to present findings and may include on-line models that may be used to provide predictions in control or monitoring. Moreover, in some cases the predictions may also be tied back into control strategies, thereby closing the control loop based on data pipeline analyses.

[0056] As an example only, Figs. 2A-2C illustrate various different data pipelines that could be implemented in a plant network 10 such as that of Fig. 1. In particular, Fig. 2A illustrates a data pipeline 50 having five data analytics blocks communicatively coupled together (as illustrated by the arrows) to perform different data processing tasks on a single stream of data. In the example data pipeline 50 of Fig 2A, a first data processing block 52 collects process data of a particular type (e.g., monitoring or valve data), a second block 54 filters the data collected by the block 52, a third data processing block 56 performs data smoothing by filling in missing data or processing the filtered data to eliminate outliers, etc. The block 56 is illustrated as providing or making certain calculated data visible or exposed for use by other blocks (as part of different data pipelines or analytic modules which may connect to or obtain the exposed data). In the case of the block 56, the average (mean) and standard deviation (SD) of the filtered and smoothed data is provided as exposed parameters. In any event, a fourth block 58 of the data pipeline 50 performs a Fourier transform on the filtered and smoothed data from the block 56 and may provide a frequency representation of the data at the output thereof. A block 59 in the data pipeline 50 then performs an analysis of the frequency data at the output of the block 58 to determine one or more dominant frequencies of the signal. The dominant frequency data may be the ultimate output of the data pipeline 50 and may be stored in a database or exposed for other analytic blocks to use.

[0057] In a similar manner, Fig. 2B illustrates a data pipeline 60 having various data processing blocks that converge data streams within the data pipeline 60. In this example, the data pipeline 60 has three data collection blocks 62A-62C which are connected to receive different data streams from various different devices or data sources, filter blocks 64A-64C which filter the data streams from the blocks 62A-62C and a data processing block 65 which may process data from one of the filter blocks 64A to remove outliers. A block 66 then time aligns the data from the blocks 64B and 64C and 65 to produce a matrix of time aligned data

20 14 12

while a block 68 selects data within the matrix of data produced by the block 66 to perform further processing. The filter blocks 64A, 64B and 64C are illustrated as exposing parameters about the filtered data, such as any statistical measure of the filtered or incoming data, the filter coefficients, etc. Of course, other data could be calculated and exposed by these or any others of the blocks in the data pipeline 60 of Fig. 2B.

[0058] Still further, Fig. 2C illustrates a data pipeline 70 that includes blocks that operate on diverging data streams. In this case, a first data processing block 72 (e.g., a data collection module) collects process data of a particular type (e.g., monitoring or valve data), a second block 74 (e.g., an intermediate process module) filters the data collected by the block 72, and a third data processing block 76 (e.g., another intermediate process module) smooths the filtered data by filling in missing data or processing he filtered data to eliminate outliers, etc. Thereafter, blocks 78A-78D (also intermediate process modules) perform Fourier transforms on the filtered and smoothed data from the block 76 at different sampling rates and blocks 79A-79E filter the data from the Fourier transform blocks 78A-78D. In this case, the blocks 79D and 79E may filter the same data from the block 78D using different filtering coefficients (e.g., pass ranges), to produce different sets of filtered data. In any event, the data from the blocks 79A-79E may be stored in a database by a block 80 (which may be an end-use process module) for use in other analytics, may be presented to a user as a graph or chart by a block 82 (which may be an end-use process module), etc.

[0059] As will be understood, each of the blocks in the various data pipelines of Figs. 2A-2C may be stored and executed in different or in the same devices, processors, etc. within the plant network 10 of Fig. 1. As an example, the data pipeline 60 of Fig. 2B is illustrated in Fig. 1 as having the various data processing blocks thereof disposed in different devices within the plant network 10. In this case, the data collection blocks 62A-62C are disposed in various ones of the field devices 18 and 20 of Fig. 1, with the filtering blocks 64A-64C being stored and executed in various ones of the I/O devices 16 and gateway devices 22. The smoother block 65 is stored and executed in one of the analytic engines 36b while the time aligning block 66 is stored and executed in another one of the analytic engines 36a. The selection block 68 is stored and executed in the big data analytics platform 40. Of course, each of the separate blocks of the data pipeline 60 are communicatively connected over the different communication networks coupled between the devices in which these blocks are executed, and the blocks of the data pipeline 60 are configured to communicate with one another (as defined by the data pipeline 60) using appropriate communications for that

20 14 12

network or via inter-network communications. Thus, a particular data pipeline may be distributed out among devices that use different communication networks and protocols and still operate seamlessly. Of course, the data exposed or output by any of the blocks of the data pipeline 60 may be provided to or accessed by any of the other devices or applications in the plant network 10, including those in the handheld devices 42, the cloud 30, etc.

[0060] As a further example, the entire data pipeline 50 of Fig. 2A is illustrated as being disposed in and executed in the analytics engine 36n. In another example, the first three blocks of the data pipeline 70 of Fig. 2C are illustrated as being executed in the analytics engine 36b while most of the rest of the data processing blocks of the data pipeline 70 of Fig. 2C are illustrated as being executed in the big data platform 40 with the charting block 80 being illustrated as being run in a handheld device 42.

[0061] Of course, the configuration and locations of the various blocks of the data pipelines of Fig. 2C are exemplary only and any particular block or module of a data pipeline may be stored and executed in any suitable device within the plant network 10 based on, for example, the processing power of the device, the data storage capability of the device, the proximity of the device to a data source, etc. Moreover, different data processing modules of the same data pipeline may be stored in different networks, such as in a process control network (e.g., any of those associated with the field devices 18 and/or 20 and the controller 14 of Fig. 1) and/or a plant communication network, such as any of the communication networks 26, 28, 30 or Fig. 1. The point is that, once created and installed in the various devices in the plant network 10, a data pipeline operates on the data provided thereto to produce new data which can be stored or provided to any of various analytic modules routines, other data pipelines, etc. which may use this data to run models, perform higher level analytics, present data to a user, etc. In some cases, the data pipelines may themselves be analytic modules that are generally distributed in the plant 10 to process data streams in known or predetermined manners to make new data or new types of data available for use or processing by other analytic modules or for consumption by a user. Of course, if desired, data pipelines may be used to implement an entire analytic module (from start to finish) or may be incorporated into or used as a portion of an analytic module. As a result, a particular data pipeline, once installed in a plant, may be used by more than one analytic module to perform different types of analytics on the same data or using the same data.

[0062] Moreover, it will be understood that the data pipelines of Figs. 2A-2C are merely examples and that any other types of data pipelines defining any set of processing blocks

20 14 12

communicatively coupled together in any particular order may be designed and used. Still further, data pipelines may have converging or diverging streams or both, and may be much more complicated than the simple data pipelines illustrated in Figs. 2A-2C, with any number of diverging and/or converging branches, any number of modules connected in any manner, etc.

[0063] As will be appreciated, the use of data pipelines as fundamental data processing architectural concept in monitoring and analytics systems enables the efficient processing of data in real-time or in a time frame used by analytic modules. Such a data pipeline includes a set of algorithms that have been arranged in a specific order and configured in order to develop models that may be used by other analytic modules. The resultant models may be downloaded and used in a runtime environment to perform predictions, fault detection, and recommendations. Modules that may be included in the data pipeline include, for example, data selection and/or data cleaning modules, data alignment modules, sensitivity analysis modules, causality analysis modules, supervised learning modules and unsupervised learning modules.

[0064] Moreover, data pipelines or any of the processing modules within any of the data pipelines can execute data collection and processing at any desired communication and data processing rates. As an example, data pipelines or the processing modules thereof may collect raw process data at a data rate faster than, the same as, or slower than the execution rate of a process controller disposed within a process control network in which the raw process data is generated. Generally speaking, it is typically desirable to collect raw process data at a data rate faster than the execution of a process controller that may use such data to perform process control activities within a plant. Moreover, to process such data in real-time, it is typically desirable to configure at least some of the data processing modules in a data pipeline to operate at an execution rate faster than the execution rate of the process controller and possibly to send data between different processing modules of a data pipeline at rates faster than the execution rate of the process controller. This higher data rate enables a data pipeline to process and analyze raw data at a data rate faster than the controller execution rate, which typically enables better or more complete process analytics.

[0065] In a more general sense, Fig. 3 depicts a block diagram 85 of various types of data processing blocks or functions that can be connected together to form a data pipeline within a plant network. In particular, a block 86 represents one or more sources of data within the plant including structured and unstructured data, such as process data, spectral data, vibration

20 14 12

data, image data, etc. The data sources represented by the block 86 can be any data source, such as a source of measured or raw data, processed data, control data, monitoring data, maintenance data, device data, data produced by other data pipelines or other analytics modules, etc. This data can be available from an on-line source, from a database or historian or from an external source (e.g., the internet, a source external to the plant network, etc.)

[0066] The diagram 85 also includes blocks representing various types of processing that can be performed by one or more data processing blocks in a data pipeline, including a data selection and cleaning block 87, a data alignment block 88, a sensitivity analysis block 89, a causality analysis block 90, a supervised learning block 91, and an unsupervised learning block 92. The blocks 87-92 include the main types of data processing functions that can or will be performed in a data pipeline, and each data pipeline will typically include one or more processing blocks of the types represented by the blocks 87-92 communicatively connected together in some manner. As illustrated by the arrows in Fig. 3, the blocks 87-92 in a data pipeline will obtain data from the block 86 (e.g., some specified source of data) and may be connected together in any order or manner to perform various data processing functions.

[0067] Still further, as illustrated by a presentation block 93 and an exploratory analysis block 94 in Fig. 3, the data pipelines created from a combination of one or more of the blocks 87-92 can provide outputs or processed data to various users in the plant network. The presentation block 93, for example, provides some type of presentation of the data produced by the data pipeline. Such a presentation may be, for example, a graph, a chart, a view box, an animation, etc., that provides some representation, illustration, depiction or other viewing of the data produced by one or more blocks of the data pipeline (e.g., from a causality block 90, a sensitivity block 89, a supervised or unsupervised learning block 91, 92, etc.) Likewise, the exploratory analysis block 94 can provide a user with an interface into a data pipeline to make changes, add new or change data sources, debug the data pipeline, etc., all in an effort to create new or more useful data pipelines, to try new data processing or new sources of data in the data pipeline, etc.

[0068] Of course, there are many types of algorithms or techniques that may be used to perform data processing in any of the blocks 87-92, and only a few examples of these techniques will be discussed below. In particular, the data selection and cleaning functions of the block 87 may be implemented using various techniques. For example, these blocks may select or specify the parameters or data to clean and the time period of data to clean or use using any desired techniques. These selections may be absolute or fixed as specified by the

20 14 12

block creator or may be determined based on some ongoing criterion or calculation. For example, the data selection may start automatically when a specified parameter shows high variability (either on-line or in the historical data stored in a database). A data selection block 87 may explore the period of time when variability is high for a parameter of concern and may extend this period, based on the process response time, for example, to the past and to the future if such data is available. In another case, a data selection block 87 may review variability of downstream and upstream parameters within a defined period of time, and may select parameters which exceed minimal variation (assuming upstream/downstream parameters are predefined at the configuration or otherwise). If an upstream/downstream path within a plant is not defined, the data selection block 87 may select parameters for the process unit, develop a principle component analysis (PCA) with the data over the selected period of time, may calculate an oscillation significance index (OSI) for all parameters, and may then select parameters based on this index value. Of course, many other manners of selecting data may be used.

20 14 12

[0069] In another case, a data selection and cleaning block 87 may detect missing or bad data and perform data correction or exclusion on this data. For example, a block 87 may give a bad status data value the last known good value for that variable (e.g., on-line). In another case, a block 87 may fill in missing data using interpolation techniques. For example, if the slope of the data trend of the last good portion of data (before the missing data) is about the same as the slope of the portion of data after the period of missing data, the missing parameter values may be filled in according the slope. This rule may be applied up to a length of data equal to the process response time or if process response time is unknown up to a predetermined number of samples (e.g. 10). If the slopes of data prior to and after the missing data are not the same, the variable mean value of the data may be applied to generated the missing data values. Likewise, if a significant number of values of data are missing, e.g., two times more than the available data, or if several variables are bad in the same time period, the period of time when data is bad can be cut from the data set. Any known procedure for excluding or separating data for multi-state models or for performing data slicing may be used. In this case, the data slicing may be performed after defining parameter delays relative to the leading variable. For unsupervised learning, a simple data cut may be applied. Still further, if more than 20 percent of the values over a selected period are bad, the variable may be excluded from the modeling/learning altogether, unless this variable is a key variable which definitely needs to be in the analysis.

[0070] In still another case, the block 87 may perform outlier removal by applying, for example, a 3σ (standard deviation rule), or applying a Hammel analysis, (i.e., a median absolute deviation rule), or applying a model based rule (e.g., AR/ARIMA) or applying Kalman based outlier detection rules, to name but a few.

[0071] Still further, a data selection and cleaning block 87 may perform data filtering. For example, a block 87 may apply filtering only when a specific application requires the filtering or may provide a selection of the any desired or known filtering techniques, such as Savitzky-Golay filtering, exponential filtering, averaging filtering, optional wavelet filtering, etc.

[0072] In a similar manner, a block 88 may perform data alignment using any desired techniques. For example, data alignment may be performed by selecting a downstream quality parameter for a supervised learning or a process downstream leading parameter with a significant variation for unsupervised learning. The data alignment may apply a cross-correlation algorithm for each upstream parameter one by one to define a time delay for upstream parameters or may apply PCA techniques for each upstream parameter for defining time delay. The block 88 may develop a new file of aligned data based on the time delays and, when the quality parameter is determined in a lab, may use only the first value after a change.

[0073] In still a further case, a block 89 may perform sensitivity analysis using, for example, a time aligned file (such as produced a block 88) and develop a partial least squares (PLS) model. The block 89 may define gain for every model input experimentally using step changes on every input sequentially (to determine sensitivities of variables to changes in other variables). The block 89 may exclude inputs with smaller than minimal gain from the model configuration and may optionally apply PCA and develop OSIs (oscillation significance indexes) for input parameters.

[0074] Likewise, a block 90 may perform causality analysis using non-aligned and/or aligned files and may develop PCA models from these files. The block 90 may also define OSI indexes for input parameters (significance indexes), and may exclude inputs with smaller than required minimal index value from the model configuration.

[0075] Still further, a block 91 may perform supervised learning using any desired non-linear models, such as neural network models, or linear models such as PLS or MLR models. A block 92 may perform unsupervised learning by applying analytic modeling, such as PCA and SVD modeling, by performing data clustering, K-means, k-NN, decision trees, bagging,

20 14 12

boosting, random forest, conditional Bayesian probability analyses, etc. Of course, these are just a few examples of the types of processing that can occur or that can be implemented in each of the blocks 87-92 in a data pipeline and many other types of data processing can be applied as well or instead.

[0076] As a further example of the use of data pipelines, Fig. 4 depicts a simplified block diagram of an example big data network 100 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 operates using one or more data pipelines to implement one or more instances of the automatic or autonomous signal processing-based learning. The process control big data network 100 includes one or more process control big data nodes 102-110, 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 some cases, 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 100.

20 14 12

[0077] The example process control big data network 100 includes one or more different types of process control big data nodes or devices 102-110, 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 102-110 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. Accordingly, the process control big data network 100 comprises the process control system big data network backbone and the process control big data nodes 102-110 that are communicatively connected thereto. In an example, the process control big data network 100 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 100 via the backbone. As will be described in more detail, one or more data pipelines may be implemented within the various devices, nodes , etc. of the big data network 100 to perform data processing and supervised or unsupervised learning on data generated within, collected within or otherwise available at the various nodes 102-110 of the big data network 100.

[0078] The process control big data network backbone may include any number of wired communication links and any number of wireless communication links that support one or more suitable routing protocols, e.g., protocols included in the Internet Protocol (IP) suite (e.g., UDP (User Datagram Protocol), TCP (Transmission Control Protocol), Ethernet, etc.), or other suitable routing protocols. The backbone may support 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. For example, aforementioned U.S. Application No. 14/506,863 entitled "STREAMING DATA FOR ANALYTICS IN PROCESS CONTROL SYSTEMS" describes examples of streaming protocols and techniques for process control big data, any one or more of which may be utilized by the process control big data network backbone in the network 100. Typically, each node included in the process data big data network 100 may support at least an application layer (and, for some nodes, additional layers) of the routing protocol(s) supported by the backbone. In some cases, each process control big data node 102-110 is uniquely identified within the process control system big data network 100, e.g., by a unique network address.

[0079] At least a portion of the process control system big data network 100 may be an ad-hoc network. As such, at least some of the nodes 102-110 (and/or one or more other nodes, such as a user interface device 130) may connect to the network backbone (or to another node of the network 100) in an ad-hoc manner.

[0080] As Fig. 4 is a simplified diagram that depicts communicative connections between various big data nodes 102-110 in the process control big data network 100, the process control network backbone is not explicitly illustrated in Fig. 4. 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 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 Application No. 13/784,041, but may be utilized with any suitable communication network backbone.

[0081] Turning now to the different types of process control big data nodes or devices 102-110, generally, a process control big data node of the network 100 may be a "big data provider" and/or may include a "big data appliance," as is discussed below.

[0082] 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 100. The process control big data that is generated, collected, observed, and/or forwarded by provider nodes may include data that has been directly utilized in or generated from controlling or monitoring a process within the plant, e.g., first-order real-time and configuration data that is generated or used by process control devices such as controllers, input/output (I/O) devices, and field devices such as transmitters, sensors, etc. 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 100 and/or of other communication networks in the plant, data indicative of bandwidth, network access attempts, diagnostic data, monitoring 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 100 by analyzing process control data, monitoring data, maintenance data, etc. 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 using one or more data pipelines as described herein.

[0083] In some cases, a big data provider node may include 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 100. 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, monitoring data, maintenance data (such as that collected by or generated by maintenance devices, algorithms, procedures, etc.) 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 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. Application Nos. 13/784,041, 14/174,413, and 14/212,493.

[0084] 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 Application No. 14/212,493 and/or in U.S. Patent Application No. 14/507,188 , entitled "REGIONAL BIG DATA IN PROCESS CONTROL SYSTEMS", for example, although other suitable embedded big data appliances may be 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.

[0085] In some cases, a big data appliance may be a stand-alone big data node of the process control big data network 100. 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.

20 14 12

[0086] Fig. 5 depicts a simplified block diagram of an example big data appliance 214, instances of which may be included in any of the nodes 102-110 of the process control big data network 100 of Fig. 4 or similar nodes of Fig. 1. 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. For example, a data point may be tagged and stored 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.

[0087] 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. The big data analyzers 226 may individually and/or collectively perform large scale data analysis on the stored data (e.g., data mining, data discovery, etc.) using one or more data pipelines 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 some cases, 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 Application No. 14/507,188 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.

[0088] 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 cases, 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 224 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 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.

20 14 12

[0089] Returning again to Fig. 4, the process control big data network 100 may include process control big data provider nodes 102-110 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 102a-102n 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" 102a-102n typically are nodes and/or devices that generate, route, and/or receive primary process control data or monitoring data (from a process monitoring system) to enable the one or more processes to be controlled or monitored in real-time in the process plant. Examples of local big data provider nodes 102a-102n 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.).

[0090] Other examples of local big data provider nodes 102a-102n 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 100 and/or other communication networks). Examples of such types of local big data provider nodes 102a-102n 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 102a-102n 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.

[0091] In some cases, a local big data provider node includes a respective local big data appliance, as illustrated in Fig. 4 by the nodes 102a, 102n that respectively include the embedded big data appliances 112a, 112n. Each local, embedded big data appliance 112a, 112n receives and stores respective local big data provided by its respective provider 102a, . . ., 102n. Further, in some local big data provider nodes, such as in the node 102a, one or more analytics functions, routines, operations, or processes (represented by the encircled $\mathbf{A_1}$) may be performed on at least some of the local big data stored in the appliance 112a. The analytics $\mathbf{A_1}$ may be performed by one or more of the big data analyzers 226 of Fig. 5 and the analytics $\mathbf{A_1}$ may be performed or implemented with one or more data pipelines. The learned information, learnings, and/or the results of the one or more analytics $\mathbf{A_1}$ may also be stored in the local big data appliance 112a, and at least some of the learned information or results may be provided to another big data node 106a.

[0092] Some local provider nodes, e.g., as illustrated by the node 102n, include a respective local, embedded big data appliance 112n for local big data collection and historization, however, the resident appliance 112n performs minimal or no analytics. As

20 14 12

such, the node 102n merely streams (or otherwise transmits, e.g., upon request or at suitable times) locally stored big data to another node 106b, e.g. for analytical processing or for further forwarding. Some local big data nodes, e.g., the node 102b, do not include any big data appliance at all. Such nodes 102b may stream, in real-time or with the aid of a cache, locally observed big data to one or more other big data nodes 102a, 106b.

[0093] 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 102a-102n. 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.

[0094] 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 cached, collected, stored, transmitted, and/or streamed by the big data provider nodes 102a-102n 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.

[0095] 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 102a-102n as big data. Data pipelines as disclosed herein may operate on or use any of this data.

[0096] In some embodiments, dynamic measurement and control data may be automatically cached, collected, stored, transmitted, and/or streamed by the big data provider

20 14 12

nodes 102a-102n 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 100. Of course, data pipelines as disclosed herein may operate on or use any of this data.

[0097] 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 102a-102n 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 102a-102n. 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 102a-102n. Data pipelines as disclosed herein may operate on or use any of this data.

[0098] In some situations, added parameters created by end users are automatically captured in the big data provider nodes 102a-102n. 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.

[0099] Referring again to the network 100 of Fig. 4, at one or more levels or tiers above the local big data nodes 102a-102n, the process control big data network 100 may include one or more regional big data nodes 106a-106m. 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,

20 14 12

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 106a, 106b, 106m 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 102 and, in some cases, by other big data provider nodes 104. Other big data provider nodes 104 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 130, 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.). As will be understood, the regional big data nodes 106a-106m may correspond to any of the regional layers or blocks 26 of Fig. 1.

[00100] As illustrated in Fig. 4, a respective grouping or region serviced by a regional big data node 106a-106m may comprise one or more big data provider nodes 102a-102n and/or other big data nodes 104 that are related according to some geographical, physical, functional, logical, or other desired manner. For example, the regional big data node 106a services a region including the local big data provider nodes 102a and 102b, and the regional big data node 106b services a region including the local big data nodes 102b and 102n, as well as another big data node 104. The particular nodes 102, 104 included in a particular region may stream or deliver data to their respective regional big data node 106 for purposes of regional big data storage, access, and/or analysis. Further, any of the big data provider nodes 102a-102n and/or the other big data nodes 104 may communicate with a particular regional big data node 106a-106m 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 106a-106m or not. Moreover, each data pipeline as described herein may be implemented within a specific regional big data node 106a-106m or across various regional big data nodes 106a-106m.

[00101] Accordingly, each regional big data node 106a-106m includes a respective regional big data appliance 116a-116m via which big data is received, stored as regional big data, and accessed or requested. Further, each regional big data node 106a-106m typically includes one or more analytics functions, routines, operations, or processes (e.g., $\mathbf{A_2}$ – $\mathbf{A}_w$) that may individually and/or cooperatively operate on at least some of the regional big data. For example, the regional big data appliance 116b may receive local big data from local

20 14 12

provider nodes 102b, 102n that are configured to cooperatively control the flow of a liquid through a portion or region of the process plant, and the node 106b may perform an analytics process $\mathbf{A_4}$ on at least some of the received data to determine an average transport delay of the liquid within the portion or region of the process plant. The analytics process $\mathbf{A_4}$ may, for example, tie into and use data produced or developed by one or more data pipelines or may instead be implemented partially or whole by one or more data pipelines. The results of the analytics $\mathbf{A_4}$ may then be stored or historized as additional regional big data within the regional big data appliance 116b. In an embodiment, each of the analytics $\mathbf{A_2} - \mathbf{A_w}$ are performed by one or more big data analyzers 226 of Fig. 5 that are resident on their respective big data node and these analytics may be implemented using one or more data pipelines downloaded into and implemented or executed at the various devices of the bid data nodes 106a-106m.

[00102] In some situations, the regional big data nodes or devices 106a-106m communicate received or generated big data, learned knowledge or information, and/or analytics results with another regional big data node 106a-106m, e.g., as peers. In some cases, data pipelines may be implemented across multiple peer nodes. To illustrate by continuing with the above example, the regional big data node 116a receives learned information that has been generated by the analytics analysis $\mathbf{A_4}$ performed by the regional big data node 106b. Subsequently, the regional big data node 106a may then perform one or more respective regional analytics $\mathbf{A_2}$, $\mathbf{A_3}$ on at least part of the peer-provided learned information from the node 106b in conjunction with the local big data received from the local big data nodes 102a, 102b within its own region. These analytics may be implemented using the same or different data pipelines as each other and/or as the analytics $\mathbf{A_4}$. The analytics $\mathbf{A_2}$, $\mathbf{A_3}$ may, in turn, using still further data pipelines, generate additional regional big data for historization at the regional big data appliance 116a and/or for provision to other big data nodes 106b, 106c, 108. As such, as a regional big data node 106a-106m may originate regional big data in some scenarios (e.g., based on the results or learning of any resident analytics that were performed thereby), a regional big data node 106a-106m may also operate as a regional big data provider node.

[00103] Grouping of the big data provider nodes 102a-102n under respective regional big data nodes may be carried out according to any desired manner such as geographical, physical, functional, logical, etc. For example, in an illustrative but non-limiting scenario, a process in the process plant may produce a particular product based on two intermediate

20 14 12

products. As such, the local big data provider node 102a may represent a first control loop that produces the first intermediate product, and the local big data provider node 102b may represent a second control loop that produces the second intermediate product. Thus, all process control data that is generated, collected, received or otherwise observed by the two control loops 102a, 102b may be transmitted to the regional big data node 106a for historization, storage and analysis.

[00104] In a similar manner, the regional big data node 106b may receive and analyze data from its respective group of big data provider nodes. For example, the regional big data node 106b may be responsible for analyzing the big data corresponding to the production of another product that is based on intermediate products from each of the big data provider nodes 102b, 102n in conjunction with big data provided by other sources 104.

[00105] At the regional big data node 106a, the received big data may be analyzed (e.g., by using one or more analytics functions or processes $\mathbf{A_2}$, $\mathbf{A_3}$) to create or generate learned knowledge that describe meaningful relationships, patterns, correlations, trends, etc., across time and/or across at least some of the various data sets. For example, a certain combination of events in the two control loops 102a, 102b may lead to poor product quality when the particular product is eventually produced. To determine the root causes of the poor product quality, the regional big data node 106a analyzes data generated by the combination of events at or shortly after their occurrence (e.g., when the data corresponding to the events' occurrences is received at the regional big data node 106a) and may do so using one or more data pipelines created for this purpose and executed in the various devices in the node 106a. The regional big data node 106a may generate learned knowledge that predicts the poor product quality based on the occurrence of these events, and/or may automatically adjust or change one or more parameters in real-time to mitigate the effects of the combination of events if and when they occur in the future. For instance, the regional big data node 106a may determine a revised setpoint or revised parameter values to better regulate and manage the two control loops 102a, 102b.

[00106] Generally, each regional big data node 106a-106m (or its respective big data appliance 116a-116m) analyzes data from its respective group or region of big data provider nodes to determine meaningful patterns, correlations, trends, etc. and may use one or more data pipelines to perform these analytics. The learned patterns, correlations, trends, etc. are then stored in the respective regional big data appliances 116a-116m as learned knowledge. As used herein, the term "learned knowledge" or "learnings" generally refers to data,

20 14 12

services, functions, routines, and/or applications that are generated as a result of one or more analyses being performed on big data. Further, each regional big data node 106a-106m (or its respective big data appliance 116a-116m) may determine or define a new service, function, routine, or application (and/or modify an existing service, function, routine, or application) based on the initially learned knowledge, which, in turn, is stored as further learned knowledge.

[00107] Regional big data nodes 106a-106m may be utilized for layered or leveled learning. For example, one or more regional big data nodes may transmit their learned knowledge and/or stored data to an upstream big data node that oversees multiple regions. As illustrated in Fig. 4, a regional big data node 106c receives learned knowledge and/or data from the regional big data nodes 106a and 106b, and the node 106c historizes the received big data in its respective embedded appliance 116c. The regional big data node 106c may perform further analysis or learning on at least some of the received learned knowledge and/or data (e.g., by using one or more of the analytics functions $\mathbf{A_8} - \mathbf{A}_w$) to generate additional learned knowledge (e.g., data patterns, trends, correlations, etc., services, functions, routines, and/or applications), which may be, in turn, stored as additional regional big data within the embedded appliance 116c and/or provided to other big data nodes 106a, 106b, 108. Of course, data pipelines may be used to implement any of the analytic functions $\mathbf{A_8}$-$\mathbf{A_w}$, or any combination of analytic functions $\mathbf{A_1}$-$\mathbf{A_w}$.

[00108] In some cases, layered or leveled learning is carried out on a bottom-up basis using data pipelines. In an illustrative but non-limiting example, a regional big data node 106a analyzes data received from its respective group of local big data provider nodes 102a, 102b to determine if its "region" is operating correctly. Knowledge that the regional big data node 106a learns from its analysis may result in the regional big data node 106a generating a new diagnostic parameter (e.g., a learned parameter). The regional big data node 106a may transmit the generated diagnostic parameter as defined by a data pipeline to an upstream big data node 106c for storage, usage, and/or access. The regional big data node 106a may independently initiate the sharing of the new diagnostic parameter with the upstream regional big data node 106c (e.g., automatically as generated or on a periodic basis) using the same or a different data pipeline, or the regional big data node 106a may cause the new diagnostic parameter to be transmitted when the upstream regional big data node 106c requests the regional big data node 106a to share one or more types of new learned knowledge.

20 14 12

[00109] Of course, layered or leveled learning using data pipelines may be carried out on a top-down basis. To illustrate by continuing with the above example, the upstream regional big data node 106c may analyze the received diagnostic routine and determine that the diagnostic routine is useful or applicable to other regional big data nodes (e.g., the regional big data node 106b). Accordingly, the upstream regional big data node 106c may distribute the diagnostic parameter to the regional big data node 106b using one or more other data pipelines so that the regional big data node 106b and/or any of the local provider nodes 102a, 102n, 104 included in its region is able to utilize the diagnostic parameter for its respective diagnostic purposes. The upstream regional big data node 106c may independently initiate the sharing of the new diagnostic parameter with the regional big data node 106b, or the upstream regional big data node 106c may cause the new diagnostic parameter to be transmitted upon a request made by the regional big data node 106b. Alternatively or additionally, the upstream regional big data node 106c may generate a general diagnostic parameter by aggregating and analyzing learned knowledge received from all regional big data nodes that it is overseeing or connected to. In this scenario, the upstream regional big data node 106c distributes the general diagnostic parameter to any or all of its regional big data nodes, e.g., automatically as generated or on a periodic basis, upon request of a particular regional big data node, when the upstream regional big data node 106c receives data from a regional big data node that indicates the general diagnostic may be of use, or for some other reason (such as defined by a different data pipeline). Subsequently, and in a similar manner, each regional big data node downstream of the regional big data node 106c may distribute the general diagnostic routine to any number of the local big data providers in its respective region.

[00110] In still other cases, regional big data nodes, e.g., the nodes 106a and 106b, may share learned knowledge with each other, e.g., in a peer-to-peer manner, using one or more data pipelines. For example, the regional big data node 106a transmits a new or learned analytics routine or parameter directly to the regional big data node 106b so that the regional big data node 106b may utilize the new analytics routine for its own purposes.

[00111] It is noted that in Fig. 4, only one upstream regional big data node 106c is depicted. However, the techniques and concepts discussed with respect to Fig. 4 may be applied to any number of upstream regional big data nodes supporting multiple layers or levels of big data historization, storage and learning.

[00112] Further, as both regional big data appliances and localized big data appliances service different respective big data nodes and/or different respective groups or regions of the process plant, but do not service the entire process plant or more than one region thereof, both regional big data appliances and localized big data appliances are generally and categorically referred to herein as "distributed big data appliances." Generally, distributed big data appliances communicate big data with multiple other big data appliances. For example, a local big data appliance included in a particular big data provider node may communicate learned knowledge and/or big data to other localized big data appliances included in other big data provider nodes, to one or more regional big data appliances, and/or to a centralized big data appliance (which is described in more detail below). Similarly, a regional big data appliance may receive big data from one or more localized big data appliances and/or big data provider nodes. The regional big data appliance may communicate learned knowledge and/or big data to other regional big data appliances, and/or to a centralized big data appliance.

[00113] As mentioned above, in some configurations of the process control big data network 100, at least some of the regional big data nodes or devices 106a-106m, local big data nodes or devices 102a-102n, and/or other big data nodes or devices 104 communicate respective big data, analytics results, and/or learned information to a centralized big data node 108, and again may do so according to one or more predefined data pipelines. A "centralized big data node" (which corresponds to the layer 28 of Fig. 1), typically services multiple regions of the process plant, and in some situations, services a majority or an entirety of the process plant. As such, the centralized big data node 108 includes one or more centralized, embedded big data appliances 118 to receive, store, and provide access to process plant big data. For example, the centralized big data appliance 118 may provide comprehensive, long-term historization of most or all of the big data generated by the process plant, and/or the centralized big data appliance 118 may publish big data for process plant-wide availability to other big data nodes, or even to computing devices within or external to the process plant that are not process control big data nodes.

[00114] In some configurations, a single centralized big data node 108 or appliance 118 may not service an entire process control system or plant, but may service more than one region of the process control system or plant. For example, different centralized big data nodes 108 or appliances 118 may be used within a single plant or system to partition different

types or areas of big data for security and access purposes. In some configurations, a single centralized big data node 108 or appliance 118 services the entire process plant.

[00115] In the process plant, one or more of the regional big data nodes 106a-106m may cause some or all of its generated or received learned knowledge and/or data to be streamed or otherwise delivered to the centralized big data node 108. For example, one or more of the regional big data nodes 106a-106m transmits at least some of its respectively stored learned knowledge and/or data to the centralized big data node 108. In some embodiments, one or more of the regional big data nodes 106a-106m pushes at least some of its respectively stored learned knowledge and/or data to the centralized big data node 108 at periodic intervals in a manner defined by one or more appropriate data pipelines. In some embodiments, one or more of the regional big data nodes 106a-106m provides at least a portion of its respectively stored learned knowledge and/or data in response to a request from the centralized big data node 108. In this case, data pipeline blocks may be set up to automatically publish or send data to the next (upstream) block in the pipeline or to request data from a previous (downstream) block in the data pipeline.

[00116] The centralized big data node 108 and/or its embedded appliance 118 may be configured to further analyze any or all of the received learned knowledge and/or data received from the regional big data nodes 106a-106m, e.g., by utilizing one or more analytics functions $\mathbf{A}_x - \mathbf{A}_y$. In an embodiment, each of the analytics $\mathbf{A}_x - \mathbf{A}_y$ are performed by one or more big data analyzers 226 of Fig. 5 that are resident on the respective big data node. The one or more analytics functions $\mathbf{A}_x - \mathbf{A}_y$ may operate as defined by one or more data pipelines on the received learned knowledge and/or data to generate additional knowledge and determine relationships between various entities and providers internal and external to the process plant. The additional knowledge and determined relationships may be stored and otherwise utilized as additional centralized big data at the embedded appliance 118, for example. In some cases, the centralized big data node 108 or appliance 118 utilizes the generated knowledge and relationships to control one or more processes of the plant accordingly.

[00117] Indeed, any node 102-106 of the big data network 100 may stream or otherwise provide big data to a centralized big data appliance 118, e.g., for historization or long-term storage. For example, a local big data provider node 102 may stream its big data directly to the centralized big data node 108. Similarly, any node 102-106 of the big data network may request services provided by the embedded centralized appliance 118, and/or may request

20 14 12

access to data and metadata stored therein. Further, in embodiments in which multiple centralized big data nodes 108 or appliances 118 service a single process plant, the multiple centralized big data nodes 108 or appliances 118 may communicate in a peer-to-peer manner, similar to that described for the regional big data nodes 106a-106m.

[00118] Also similar to the regional big data node 106a-106m, a centralized big data node 108 may itself be a producer or provider of big data in some situations, such as when analytics performed by the centralized big data node 108 (e.g., one or more the analytics functions $\mathbf{A}_x - \mathbf{A}_y$) result in additional discovered or learned information that is stored at the centralized big data appliance 118 and made accessible to other big data nodes 102-106. However, typically, the majority of the volume of big data handled and serviced by a centralized big data appliance 118 is received from other big data nodes 102-106. A description of an example centralized big data node 108 and an example centralized big data appliance 118 which may be utilized with any or all of the techniques described herein may be found in aforementioned U.S. Patent Application No. 13/784,041. However, it is understood that any or all of the techniques described herein may be alternatively or additionally utilized with centralized big data appliances other than those described in U.S. Patent Application No. 13/784,041.

20 14 12

[00119] In some configurations, a centralized big data node 108 communicates data and learned information to a remote big data node (e.g., a big data node that is remote with respect to the process plant) for big data historization, storage, access, and/or analysis. Such a big data node, referred to herein as a "cloud big data node 110," may provide services to multiple different process plants or process control systems 10. For example, a company that operates multiple different oil refineries may provide a cloud big data node 110 and a cloud big data appliance 121 to service big data related to all of its oil refineries. Data pipelines as described herein may be implemented partially or completely in the cloud big data node 110. For instance, via the cloud big data node 110 and the resident cloud big data appliance 121, a centralized big data node of a particular refinery may obtain published big data generated by the process control big data network 100 of the process plant, and may utilize the obtained, published big data for operations at the particular refinery using the same data pipeline. In some embodiments, any of the big data nodes 102-106 may directly stream or provide data to the cloud big data node 110 according to one or more data pipelines. Similarly, any of the big data nodes 102-106 may request services provided by the embedded appliance 121 of the cloud big data node 110, and/or access data and metadata stored therein according to one or

more data pipelines. Although not illustrated in Fig. 4, a cloud big data node 110 may include one or more respective analytics routines, functions, or processes therein, e.g., as may be provided by big data analyzers 226 of Fig. 5 and that may be implemented according to one or more data pipelines as described herein.

[00120] Further, it is noted that not all types of big data nodes are included in all process plants. For example, a highest level of big data processing at a particular process plant may be at the regional level, and as such the particular process plant may not include any centralized big data nodes 108, and may not be connected to a cloud big data node 110. Generally, though, to facilitate or support process control big data, a process plant includes at least one local big data provider node 102 and at least one big data appliance 112, 116, 118.

[00121] Additionally, 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 100. Data pipelines may pass through these nodes, however, to perform communications between the various data processing blocks of the data pipeline.

[00122] Moreover, typically, big data nodes or devices 102-110 do not have an integral user interface, although some of the big data nodes or devices 102-110 may have the capability to be in communicative connection with one or more user interface devices 130, e.g., by communicating over a wired or wireless communication link, or by plugging a user interface device 130 into a port of the big data nodes or devices 102-110. In Fig. 4, the user interface device 130 is depicted a big data node that is wirelessly connected to the process control big data network 100.

[00123] The user interface device 130 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,

20 14 12

cameras, and/or any other user interface technology. The user interface devices 130 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. The user interface devices 130 may be implemented as wireless handheld devices such as the devices 42 of Fig. 1 and may be connected to one or more data pipelines to receive end-use data from the one or more data pipelines. Moreover, these devices or end-use devices may be intermittently connected to a data pipeline or may remain connected to a data pipeline for long periods of time.

[00124] In some embodiments, a user interface device 130 may have one or more built-in analytic capabilities (denoted in Fig. 4 by the encircled $\mathbf{A}_z$). In other words, a user interface device 130 may communicate with any number of big data nodes and/or big data appliances to download and/or receive data and perform local analysis $\mathbf{A}_z$ on the downloaded/received data to discover or learn knowledge and may do so a part of a data pipeline or by connecting to a data pipeline. Indeed, in some configurations, a user interface device 130 may itself be a big data provider node, and may itself provide at least some of the results of its analysis $\mathbf{A}_z$ as big data to one or more other local, regional, centralized, or cloud big data nodes 102-110. Examples of the usage of user interface devices in process control big data networks (which may be utilized with any or all of the techniques described herein) may be found, for example, in aforementioned U.S. Patent Application No. 14/028,785. Of course, however, other techniques of utilizing user interface devices with process control big data network 100 may be alternatively or additionally employed in conjunction with any or all of the techniques described herein.

[00125] Fig. 6 depicts a block diagram of another example plant network 400 including various devices associated with a process plant or process control system 401 in which the automatic signal processing-based learning techniques, methods, systems and apparatuses using one or more data pipelines as disclosed herein may be implemented and included. For example, at least a portion of the automatic signal processing-based learning system may be implemented in the process plant network 400 using one or more data pipelines. At least a portion of the process plant network 400 may be supported by a process control big data network, such as the process control big data network 100 of Fig. 4. However, while the process plant network 400 is described below with respect to features illustrated in Figs. 1, 4 and 5, for ease of discussion and not for limitation purposes, it is understood that the process plant network 400 of Fig. 6 may utilize a process control big data network other than that

20 14 12

described in Figs. 1, 4 and 5 or the process plant network 400 may omit any process control big data network, nodes, and/or devices.

[00126] In Fig. 6, 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. 6 may be referring to Fig. 4, a local big data provider node and/or appliance 102a-102n, 112a-112n, a regional big data provider node and/or appliance 106a-106m, 116a-116m, a centralized big data provider node 108 and/or appliance 118, or another type of big data node 104, or may refer to similar nodes in Fig 1.

[00127] In Fig. 6, the process control big data nodes BD are nodes of the process control big data network 400. If desired, the process control big data network 400 may be the process control big data network 100 of Fig. 4, and the nodes BD may be the nodes 102-108 of the network 100. In Fig. 6, 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 100.

[00128] As illustrated in Fig. 6, 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, by using a public or internet based communication network, using a local area network, etc. In Fig. 6, the controller 411 is a big

20 14 12

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.

[00129] The controller 411, which may be, by way of example, the DeltaV™ controller sold by Emerson Process Management, 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 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. 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. 6, the controller 411, the field devices 415-422 and the I/O cards 426, 428 are illustrated as being wired devices, and the field devices 440-446 are illustrated as being 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.

20 14 12

[00130] 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 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.

[00131] 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 401. 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 401. 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® 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.

20 14 12

[00132] The wired devices 411-422 illustrated in Fig. 6 include big data wired process control devices 415, 416, and 418-421 and the I/O cards 426, 428. Fig. 6 also illustrates 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. 6, the field devices 415-418 are standard 4-20 mA devices or HART® 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 418-421and/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.

[00133] The network 400 is also illustrated as including a monitoring system (which may or may not be part of the process control system 401). The monitoring system includes wireless field devices 440-446 illustrated in Fig. 6 and these also include examples of wireless big data nodes or devices BD (e.g., devices 442a, 442b, 444). Fig. 6 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® 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.

[00134] The wireless gateway 435, as illustrated in Fig. 6, 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. 6). 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.

[00135] 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

20 14 12

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.

[00136] Similar to the wired field devices 415-422, the wireless devices 440-446 of the wireless network 470 may perform monitoring function and/or 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.

[00137] 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. 6 may be a legacy 4-20 mA device and a field device 450 may be a traditional wired HART® 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. 6, 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, 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. 6, 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.

[00138] Accordingly, the system of Fig. 6 includes several examples of big data devices of nodes BD which primarily serve to provide network routing functionality and administration

20 14 12

to various networks of the process control system and in which or through which data pipelines may be implemented. 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 and theses nodes or devices may implement such communications as part of implementing one or more data pipelines. 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® and which may also be used for data pipeline communications. As illustrated in Fig. 6, 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.

[00139] 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. 6, one or more wireless access points 472 are big data devices BD that may implement data pipelines or portions of data pipelines utilizing other wireless protocols, such as Wi-Fi® or other IEEE 802.11 compliant wireless local area network protocols, mobile communication protocols such as WiMAX (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.

[00140] Additionally in the system of Fig. 6, one or more gateways 475, 478 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 which may be used to implement data pipelines or analytic modules that use data pipelines. Typically, such systems are customers or

220
141

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 110 and/or a cloud big data appliance 121.

[00141] A plant gateway node 475 illustrated in Fig. 6 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, etc.) as part of a data pipeline. 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.

[00142] Although Fig. 6 only illustrates a single controller 411 with a finite number of field devices 415-422 and possibly 440-446, this example 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, the system of Fig. 6 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. 6).

20 14 12

[00143] Further, the combination of aspects, devices, and components included in the example process plant 40 as illustrated by Fig. 6 is exemplary only. The data pipeline techniques, systems, methods, and apparatuses disclosed herein may be utilized in process plants with zero or more any of the aspects illustrated in Fig. 6. For example, the data pipeline 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.

[00144] Referring now simultaneously to Figs. 1-6, any number of any the nodes 102, 106, 108, 110 of the process control big data network, the big data appliances 112, 116, 118, 121, 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 include an instance of or a part of a data pipeline. That is, an instance of an analytic module implemented as a data pipeline may be integral with any node or nodes or device or devices included in or associated with a process plant or process control system that generates a signal whose value varies over time, such as a process control device, a big data node, an analyzer, etc. At least a portion of the signal processing module implemented as a data pipeline is implemented as software or computer-executable instructions stored on a memory of any one or more of the nodes or devices illustrated in Figs. 1-6, and that are executable by a processor that is co-resident thereon.

[00145] In some cases, at least a portion of the signal processing module implemented as a data pipeline is implemented as a chip, an integrated circuit, a semiconductor circuit, or other suitable hardware that is included in any one of the nodes or devices illustrated in Figs. 1 and 4-6. For example, at least a portion of a signal processing module may be implemented on one or more integrated digital signal processing (DSP) chips such as, e.g., the Texas Instruments KeyStone multicore DSP and ARM™ processors, or other suitable DSP chips. The one or more integrated DSP chips may be embedded into or integral with any node or device included in or associated with a process plant or process control system that generates a signal whose value varies over time, such as a process control device, a big data node, an analyzer, etc. For example, one or more of the integrated DSP chips may be included in a valve, vibration measurement device, transmitter, etc. The number of chips may be increased as desired to speed up, enhance, and/or add signal processing functions/capabilities.

20 14 12

[00146] In some cases, the signal processing modules implemented as a data pipeline are not integral with a node or device (of the process plant or process control system) that is generating the signal whose value varies over time, but nonetheless the signal processing modules are communicatively coupled to the node or device. For example, the signal processing module implemented in the form of a data pipeline may be physically attached to the node or device generating the signal whose value varies over time so that the output of the signal source traverses the various blocks or components of the signal processing module prior to being transmitted over a process control network, or the signal processing module may monitor the network link over which the output of the node or device is transmitted.

[00147] Creating or configuring a data pipeline may be performed through a data studio configuration application that generally provides a configuration environment that a user can use to easily define the blocks and the interconnections between blocks in the data pipeline, as well as the location at which each block will be downloaded and run or executed within the plant network. The data studio may generally include a configuration regions that stores templates of data processing blocks that may be used to create a data pipeline and a configuration region that enables a user to specify which processing blocks to use in data pipeline and the communication interconnections between these blocks. Once created in the configuration region, a data pipeline may be tested using historical data within a big data machine. After testing is complete, the created data pipeline may be downloaded to the various devices in the plant network and may run automatically on data as the data is collected within a plant to continuously provided new learning within the plant.

[00148] Generally speaking, the set of blocks in a data pipeline may be used to read data, filter, shape, perform analytics, visualize results and perform other functions on data. The blocks are used to build a 'data pipeline' combining and shaping data sources into a form that may be processed and tested by the analytics blocks described with respect to Figs. 1-6.

[00149] In a general sense, the data studio may be viewed at two levels. First, the data studio provides users with a structured environment to create processing and calculation entities that perform data processing to, in effect, generate predictions, perform fault detections, and generate predictions. The data studio is used to shape data, generate models, test models, compare models, and perform the functions normally executed by a data scientist. Once models have been developed, the same set of models may be downloaded into the calculation engines within the plant, for example, the embedded analytics engines or

big data appliances of Figs. 1-6. These models are also referred to herein as analytics modules.

[00150] Fig. 7 is a block diagram of an example process control or monitoring system of a process plant (such as that of Figs. 1, 4 or 6) that includes a big data network 500 which, in turn, collects data within a process plant. The process control system of Fig. 7 additionally includes a user interface station 503 provided within, for example, the process plant environment, that executes a data studio 509 to enable a user to create and run various data models on the data collected by the big data network. The user interface station 503 also executes a data knowledge application 514 (which may be associated with or implemented as function within the block 94 of Fig. 3), that executes models performing systematic or comprehensive diagnosis, prognosis, analysis, identifying relationships, etc., to discover and evaluate data retrieved from a process plant. Although Fig. 7 depicts that only a single user interface station 503 hosts the data studio 509 and data knowledge application 514, in some embodiments, the data studio 509 and data knowledge application 514 may be hosted on different user interface stations 503, on multiple user interface stations 503 or may be distributed among one or more devices in the process plant 509. More specifically, the data studio 509 as implemented within the plant or process control system provides a user or a model developer with a tool to assist in systematically and efficiently defining or creating data pipelines that run within the plant environment on historical data previously collected from the plant, real-time data currently collected within the plant, or both. The data studio 509 and the data knowledge application 514 are able to take advantage of the fact that the big data network 500 collects, stores and provides efficient access to all or most of the data generated within or collected within the plant or process control system. Moreover, the data modeling studio 509 uses this data to provide the model developer with a more efficient manner of designing, testing and implementing models within the plant or process control system. The data knowledge application 514 provides an additional environment for implanting models with the plant or process control system, as well as viewing and exploring relationships between the data as well as the results of the executed model. As illustrated in Fig. 7, the data studio 509 may be a set of instructions executed by one or more processors of one or more devices, such as on the user interface station 503. Of course, the data studio 509 may be executed on any of a variety of computing devices, such tablets, mobile phones, laptop computers, desktop computers, servers, etc. and execute within one device or may be distributed among any number of such devices. Similarly, the data knowledge application

20 14 12

514 may be a set of instructions executed by one or more processors of one or more devices and executed on any of a variety of computing devices.

[00151] In a general sense, the example process control system big data network 500 includes a process control system big data apparatus or appliance 502, a process control system big data network backbone 505, and a plurality of nodes 508 that are communicatively connected to the backbone 505. The nodes 508 are additionally connected to process plant equipment like controllers, field devices, plant equipment, monitoring equipment, etc. Process-related data, plant-related data, communication related data and other types of data may be collected and cached at the plurality of nodes 508, and that data may be delivered, via the network backbone 505, to the process control system big data apparatus or appliance 502 for long-term storage (e.g., "historization") and processing. At least some of the data may be delivered between nodes 508 of the network 500, e.g., to control a process in real-time.

[00152] Any type of data related to the process plant system may be collected and stored at the process control system big data appliance 502. For example, real-time process data such as continuous, batch, measurement and event data that is generated while a process is being controlled in the process or plant network (and, in some cases, that is indicative of an effect of a real-time execution of the process) may be collected and stored. Moreover, process definition, arrangement or set-up data, such as process plant configuration data and/or batch recipe data may be collected and stored. Additionally, data corresponding to the configuration, execution and results of process diagnostics may be collected and stored. Of course, other types of process data, such as alarms or alerts, may also be collected and stored. In a similar manner, data highway traffic and network management data of the backbone 505 and of various other communication networks of the process plant 10 may be collected and stored. Likewise, user-related data such as data related to user traffic, login attempts, queries and instructions may be collected and stored. Still further, 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 collected and stored.

[00153] The process plant big data network backbone 505 may include a plurality of networked computing devices, such as the user interface device 503 (hosting the data studio 509 and the knowledge discovery application 514), or switches that are configured to route packets to/from various nodes 508 of the process control system big data network 500 and to/from the process control big data appliance 502 (which is itself a node of the process

20 14 12

control system big data network 500). The plurality of networked computing devices of the backbone 505 may be interconnected by any number of wireless and/or wired links. Moreover, the plant network big data network backbone 505 may include one or more firewall devices.

[00154] By way of example, the big data network backbone 505 may support one or more suitable routing protocols, e.g., protocols included in the Internet Protocol (IP) suite (e.g., UPD (User Datagram Protocol), TCP (Transmission Control Protocol), Ethernet, etc.), or other suitable routing protocols. If desired, at least some of the nodes 508 may utilize a streaming protocol such as the Stream Control Transmission Protocol (SCTP) to stream cached data from the nodes 508 to the process control big data appliance 502 via the network backbone 505. Typically, each node 508 included in the process data big data network 500 may support at least an application layer (and, for some nodes, additional layers) of the routing protocol(s) supported by the backbone 505. Each node 508 may be uniquely identified within the process control system big data network 500, e.g., by a unique network address. Additionally, at least a portion of the process control system big data network 500 may be an ad-hoc network. As such, at least some of the nodes 508 may connect to the network backbone 505 (or to another node of the network 500) in an ad-hoc manner.

[00155] Moreover, data that is related to devices within the process plant network (e.g., to physical equipment included in the process plant such as machines and devices) but that may not be generated by applications that directly configure, control, or diagnose a process may also be collected and stored in the big data machine 502. For example, vibration data and steam trap data, plant safety data and data indicative of a value of a parameter corresponding to plant safety (e.g., corrosion data, gas detection data, etc.) may be stored. Likewise, data indicative of an event corresponding to plant safety may be collected and stored in the big data machine 502. Likewise, data corresponding to the health of machines, plant equipment and/or devices, equipment data (e.g., pump health data determined based on vibration data and other data), data corresponding to the configuration, execution and results of equipment, machine, and/or device diagnostics may be collected and stored.

[00156] In some cases, data generated by or transmitted to entities external to the process or plant may be collected and stored in the big data machine 502, such as data related to costs of raw materials, expected arrival times of parts or equipment, weather data, and other external data. Of course, all data that is generated, received, or observed by any of the nodes 508 that are communicatively connected to the network backbone 505 may be collected and

20 14 12

caused to be stored at the process control system big data appliance 502. Of course, any or all of these types of data could be collected by one or more data collection modules of one or more data pipelines or may be used by any of the processing modules of any data pipeline.

[00157] As illustrated in Fig. 7, the data studio 509 may be configured to provide a primary interface into the process control system big data network 500 for configuration and data exploration, e.g., on a user interface or other interface device for use by a user or other applications. The data studio 509 may be connected to the big data appliance 502 via the process control system big data network backbone 505, or may be directly connected to the process control system big data appliance 502 or may be coupled to the big data appliance 502 in any other manner.

[00158] In the network of Fig. 7, the process control big data apparatus or appliance 502 is centralized within the network 500, and is configured to receive data (e.g., via streaming and/or via some other protocol) from the nodes 508 of the network 500 and to store the received data. As such, the process control big data apparatus or appliance 502 may include a data storage area 520 for historizing or storing the data that is received from the nodes 508, a plurality of appliance data receivers 522 that may collect data for storage in the big data machine 502, and a plurality of appliance request servicers 535 that request and access data from the big data storage 520. In any event, it is useful if all or most of the data collected in the process control network and stored in the data storage area 520 is available to a user of the data modeling studio 509 and the knowledge discovery application 514.

[00159] The process control system big data storage area 520 may include multiple physical data drives or storage entities, such as RAID (Redundant Array of Independent Disks) storage, cloud storage, or any other suitable data storage technology that is suitable for data bank or data center storage. However, to the nodes 508 of the network 500, the data storage area 520 has the appearance of a single or unitary logical data storage area or entity. As such, the data storage 520 may be viewed as a centralized big data storage area 520 for the process control big data network 500 or for the process plant. If desired, a single logical centralized data storage area 520 may service multiple process plants (e.g., the process plant and another process plant). For example, a centralized data storage area 520 may service several refineries of an energy company. In addition, the centralized data storage area 520 may be directly connected to the backbone 505 via, for example, at least one high-bandwidth communication link. Moreover, the centralized data storage area 520 may include an integral firewall.

20 14 12

[00160] The structure of the unitary, logical data storage area 520 may support the storage of all or most process control system related data. For example, each entry, data point, or observation of the data storage entity may include an indication of the identity of the data (e.g., source, device, tag, location, etc.), a content of the data (e.g., measurement, value, etc.), and a time stamp indicating a time at which the data was collected, generated, received or observed. As such, these entries, data points, or observations are referred to herein as "time-series data." The data may be stored in the data storage area 520 using a common format including a schema that supports scalable storage, streamed data, and low-latency queries, for example.

[00161] In one case, the schema may include storing multiple observations in each row, and using a row-key with a custom hash to filter the data in the row. The hash may be based on the time stamp and a tag. For example, the hash may be a rounded value of the time stamp, and the tag may correspond to an event or an entity of or related to the process control system. Metadata corresponding to each row or to a group of rows may also be stored in the data storage area 520, either integrally with the time-series data or separately from the time-series data. For example, the metadata may be stored in a schema-less manner separately from the time-series data.

[00162] If desired, the schema used for storing data at the appliance data storage 520 may also be utilized for storing data in the cache of at least one of the nodes 508. Accordingly, in this case, the schema is maintained when data is transmitted from the local storage areas of the nodes 508 across the backbone 505 to the process control system big data appliance data storage 520.

[00163] In addition to the data storage 520, the process control system big data appliance 502 may further include one or more appliance data receivers 522, each of which is configured to receive data packets from the backbone 505, process the data packets to retrieve the substantive data and timestamp carried therein, and store the substantive data and timestamp in the data storage area 520. The appliance data receivers 522 may reside on a plurality of computing devices or switches, for example. Multiple appliance data receivers 522 (and/or multiple instances of at least one data receiver 522) may operate in parallel on multiple data packets.

[00164] In cases in which the received data packets include the schema utilized by the process control big data appliance data storage area 520, the appliance data receivers 522

merely populate additional entries or observations of the data storage area 520 with the schematic information (and, may optionally store corresponding metadata, if desired). In cases in which the received data packets do not include the schema utilized by the process control big data appliance data storage area 520, the appliance data receivers 522 may decode the packets and populate time-series data observations or data points of the process control big data appliance data storage area 520 (and, optionally corresponding metadata) accordingly.

[00165] Additionally, the process control system big data appliance 502 may include one or more appliance request servicers 535, each of which is configured to access time-series data and/or metadata stored in the process control system big data appliance storage 520, e.g., per the request of a requesting entity or application. The appliance request servicers 535 may reside on a plurality of computing devices or switches, for example. At least some of the appliance request servicers 535 and the appliance data receivers 522 may reside on the same computing device or devices (e.g., on an integral device), or may be included in an integral application.

[00166] In some cases, multiple appliance request servicers 535 (and/or multiple instances of at least one appliance request servicer 535) may operate in parallel on multiple requests from multiple requesting entities or applications. Of course, a single appliance request servicer 535 may service multiple requests, such as multiple requests from a single entity or application, or multiple requests from different instances of an application. In any event, the data modeling studio 509 and the knowledge discovery application 514 of Fig. 7 may access the data stored in the data storage area 520 via one or more request servicers 535.

[00167] Fig. 8 is a block diagram of an example embodiment of the data studio 509 as coupled to the big data appliance 502 and the process plant network via the process control big data backbone 505 of Fig. 7. As further illustrated in Fig. 8, one more additional user interface devices 513 may access the data studio 509 via the process control big data backbone 505. Additionally, the knowledge discovery application 514 is coupled to the process control big data backbone 505. Generally speaking, the data studio 509 provides a structured environment for graphically creating and programming data pipelines configured to perform operations on data collected from the big data machine or other data sources. In this context, a data pipeline is a series of interconnected mathematical or analytical routines performed on data to produce an output that gives some knowledge concerning the operation of a process plant (such as the process plant). The data pipeline is constructed of one or more

20 14 12

data inputs, one or more functions that are performed on the data inputs and one or more outputs that are a result of the functions.

[00168] As illustrated in Fig. 8, the data studio 509 includes a configuration engine 523 for generating a structured environment to facilitate graphical data pipeline creation and a run-time engine 524 for downloading or executing created pipelines. More specifically, the configuration engine 523 includes an interface routine 525 to generate elements of a graphic user interface for the structured environment, a plurality of templates 529 that serve as the building blocks of the data pipelines and a compiler 527 that converts the model into a data format executable by the run-time engine 524. The run time engine 524 includes a data retrieval routine 526 for retrieving data used in executing the data pipelines, and an execution routine 528 for running the executable data pipeline.

[00169] The interface routine 525 includes a set of instructions stored in memory that when, executed by a processor, generates a set of user interface elements of a drag and drop graphical user interface to facilitate creation of the data pipeline including a library region 525a that displays the templates 529 and a canvas region 525b that serves as the main presentation window for creating models. The templates 529 serve as the building blocks of the data pipelines and can be selected and moved to the canvas region 525b to create a data pipeline. The templates 529 included in the configuration engine 523 of the data studio 509 include data source or read templates 529a (that specify sources of data used in the pipeline), data write templates 529b that specify where to write data or output data or otherwise send data from the pipeline, filter templates 529c that specify filtering routines to be run on data streams in a data pipeline, merge templates 529d that specify data merging operations to be performed on one or more streams of data, state templates 529e that specify state detection and related processing activities, data analytic templates 529f that specify analytics to be performed on data, and visualization templates 529g that specify various graphing or other visualization functions to be performed on data to display the data to a user, for example. Of course, the templates illustrated in Fig. 8 are exemplary only and other types of templates can be created and stored as well.

[00170] In some cases, the data source or read templates 529a indicate data sources that collect various data types (structured and/or unstructured), contexts and/or boundary conditions of data that is communicated, generated, received and/or observed within the process plant. The data source or read templates 529a may pertain to database data, streamed data, transactional data, and/or any other type of data that is communicated via the process

20 14 12

control system big data network and is stored or historized in process control system big data storage 520. The data source or read templates 529a may also include data sources stored in the big data appliance 502 that collects and stores all (or almost all) process data and plant data collected as part of the big data architecture of the process plant. The data sources 529a may define comprehensive, high fidelity data stored in the big data appliance 502, defining any or all of the various parameters of the process plant implementing a big data architecture.

[00171] Example data source templates 529a include pressure readings, valve readings, temperature readings, vibration readings. The user may set parameters for the historical or real time data so that only a subset for the data is selected. For example, the data source templates 529a may indicate a data source that collects temperature data in degrees Celsius that are typically expected to be in the range of Temperature A to Temperature B. The data source templates 529a may also represent data sources external from the process plant, such as an external plant site or an external database, such as a weather or environment database. In a similar manner, data write templates 529b may specify any database, machine or other data pipeline or analytic module at which to write data or make data from the pipeline available.

20 14 12

[00172] The other templates 529c-529g can implement or define any basic or unitary functionality to be performed on data. For example, the analytic templates 529f may include mathematical functions such as neural network routine, a correlation routine, a threshold routine or a statistical processes such as an averaging routine, a maximum routine, a minimum routine, a filter routine, etc. The templates 529e may include classification techniques such as random forest algorithms or partial least square regressions. Still further, the templates 529c-f may include data processing techniques for preparing data for analysis, such as a filtering routine that removes data higher than a certain threshold level or a clipping routine to remove outlier data. The user may also be able to alter one or more parameters of the templates 529. For example, the user may select a filter routine and define the characteristics of a low point filters (a particular slope), a clipping filter (a threshold level), etc.

[00173] The visualization templates 529g define how to interpret and/or present the result of the functionality performed on the data by the data pipeline. For example, if the output of a pipeline is to be graphically presented to a user, a template 529g may represent a graph, a chart or a display interface. The output template 529g may further define one or more data massaging techniques to prepare the data for the graphical presentation specified by the

output template. For example, a bar graph may require a numerical value rounded to two significant figures and the output template may also include a rounding routine to prepare the data. Output or visualization templates 529g may also indicate a particular location where the output of a data pipeline should be sent. For example, a template 529g may indicate that a set of values produced by the data pipeline are to be sent to one or more controllers in a process plant (such as the process plant). Example values include a set point value, a control routine, etc. However, an output can be any value such as a binding representing a relationship, a specific value, a binary value indicating whether or not a data output meets a threshold, a percentage corresponding to a potency, etc.

[00174] However, the user is not limited to using pre-defined templates included in the data studio 509. The user may edit existing templates, add new templates, and/or save the edited/new template to the data studio 509 for future use. For example, if a pre-existing averaging routine accepts two data source inputs, a user may edit the averaging routine template to accept three inputs. The user may also create new templates from scratch to add additional functionality to the data studio 509. The user may also save data pipelines or portions of data pipelines created with the data studio 509 as templates for future use.

[00175] The compiler 527 may include a compiler routine 527a stored in memory that executes on a processor to transform the graphical data pipeline created using the user interface into a compiled data pipeline in an executable data format supported by the run time engine 524.

[00176] The run time engine 524, which executes the compiled data pipeline to produce a desired output (as defined by the data pipeline), includes a data retrieval routine 526 for retrieving data from the memory 520 and/or from the plant devices, and an execution routine 528 for running and/or executing the executable models. More specifically, a processor may execute the retrieval routine 526 to retrieve data from data sources and functions, corresponding to the graphical templates 529 used in the data pipeline, from memory. In some embodiments, the retrieval routine 526 may retrieve data from a big data appliance 520 via the big data network backbone 505 and/or from the plant itself. The execution routine 528 may include a set of instructions stored in memory that execute a compiled data pipeline on a processor to produce one or more outputs.

[00177] The execution routine 528 may run a data pipeline in an off-line environment that is isolated from the operation of the plant where the data pipeline may be run on historical

20 14 12

data stored in memory or may run a data pipeline in an online environment, where the retrieval routine 526 may retrieve real time or near real time data streams from one or more devices in the plant or big data appliance and produce an output, that may be provided to a user and/or that may be used to affect the operation of the plant.

[00178] As a more specific example, a data source template 529a (defining the data to be input into a data pipeline) may be connected to a first filter template 529c defining filtering functionality (e.g., a low pass, a high pass, a smoothing filter, etc.) The filtering function template 529c can be connected to a second filter template 529c defining a clipping routine, which can be connected to an analytics template 529f defining a neural network technique which can then be connected to a filter template 529c defining another filtering technique which can be connected to a template 529f defining an averaging technique which can be connected to a visualize template 529g which defines a manner of displaying the output.

[00179] Fig. 9 illustrates an example graphical user interface 600 of the data studio 509 for facilitating graphical creation of a data pipeline. As discussed above, the interface routine 525 of the configuration engine 523 generates a drag and drop graphical user interface 500 including a library region 525a that displays the templates 529 and a canvas region 525b that serves as the main presentation window for creating data pipelines. As illustrated in Fig. 9, the library region 525a displays a plurality of templates 529 that serve as the graphical building blocks of the data pipeline. The templates 529 may include the various data templates 529 including read templates 529a, write templates 529b, filter templates 529c, merge templates 529d, state templates 529e, analytics templates 529f and visualization templates 529g.

[00180] A user may construct the data pipeline selecting different templates 529 from the library region 525a and dragging them to the canvas region 525b. The user may arrange the templates 529 in the canvas region 526b and use the communication interconnections to define flow between the templates in order to define the data flow within the data pipeline. Although Fig. 9 illustrates a drag and drop graphical user interface, other embodiments of the data studio 509 may include other graphical or non-graphical interfaces.

[00181] The configuration engine further 523 enables a user to modify the data pipeline templates 529 depicted in the canvas region 525b by defining specific parameters to be used in the templates 529 and the interconnections between data templates, to create interconnected blocks forming a data pipeline, to define communication connections,

20 14 12

execution rates, etc. The data pipeline may include, for example, an input block defining a specific data source of data to be retrieved for the data pipeline, one or more functional blocks defining data processing procedures to be performed on the data within the pipeline and one or more output blocks defining an operation associated with the output of the one or more data outputs.

[00182] In the example of Fig. 9, a data pipeline is created that includes four read blocks 602 (defining different data sources from which to obtain data for the pipeline), a filter block 604 connected to filter each of the acquired data streams, and a time offset block 606 connected to each of the filter blocks 604 which may operate to offset or time align the data from each of the various sources to a common or aligned time. Likewise, the data produced by each of the blocks 606 is provided to a block 608 which merges the time series data and provides merged or a matrix of data to a visualization block 610 in the form of a line chart. The visualization block 610 may produce a line chart shown the merged time series data in a desirable format.

20 14 12

[00183] Of course, when creating the data pipeline of Fig. 9, the user may configure each block with various parameters (e.g., filter coefficients, sampling rates, time offsets, visualization parameters, etc.). Moreover, the user may specify the location or position of each of the blocks in the data pipeline and the execution rates and communication rates of the blocks within the data pipeline when the data pipeline is downloaded to and executed in the plant as a data pipeline. The user may, for example, specify the device, node, big data machine, etc. at which each block is to be downloaded and run during plant operation or run-time.

[00184] Still further, the user may test the created data pipeline on historical data as stored in a database, and when satisfied with the operation of the data pipeline, may download the data pipeline into the plant so that the various blocks of the data pipeline are downloaded to and executed in the specified nodes, machines or devices. Moreover, at this time, the configuration engine 523 or the compiler 527 thereof may configure the blocks of the data pipeline to communicate with one another over the communication networks installed between the various devices in which the blocks are stored and executed. In some cases, different blocks of the same data pipeline may communicate with one another using different communication networks and even different communication protocols.

[00185] Once downloaded, the data pipelines run or execute in the plant on a continuous or on-going basis during run-time of the plant to provide data processing on the data as specified by the data pipeline. This data is then available for viewing and use by any user, any machine or any analytic module within the plant. Moreover, analytical modules may use or may be made up of one or more data pipelines and a single data pipeline may be used in multiple other analytic modules.

[00186] As will be understood, the monitoring and analytics engines, algorithms, and data pipelines described herein can be run external to the distributed control systems (DCSs), asset management systems, and existing machine health systems of a typical process control plant or network. Each of these systems is considered to critical to their respective organizations (operations, maintenance, and reliability engineering) and so running the analytic modules and data pipelines on which these modules operate on different machines prevents stressing each of these systems, the operating companies trying to support them, and the various organizations and entities trying to support them.

[00187] Moreover, a difference between DCS systems and monitoring and analytics system described herein is that the monitoring and analytics system is not consuming the data in real-time. The applications are designed to receive data, perform analysis, and make predictions, perform fault detections, generate recommendations (using data pipelines) that are available to humans and other systems within the time constants of the systems. For example, in batch operation, the updated predictions can be in terms of minutes. The same is true for most continuous operations. This operation is very different from control operations where control actions have direct impacts on the operation of the process. Streaming data only needs to be available when applications actually need the data and so the data pipelines described herein (and the analytic modules which operate on top of the data pipelines) can operate at a different rate.

[00188] Moreover, as will be understood, the data pipeline structure described herein is able to be used to identify leading indicators and then use those leading indicators to identify parameters that have the most impact on the leading indicators. To get the best value out of this analysis, information about the relationship of each parameter to each other parameter (e.g. upstream, downstream) may be used as well as part of the data provided to or used by a data pipeline. Several control and device hierarchies are available in a typical plant that provide this information. For example, the equipment hierarchy available from systems such as DeltaV and AMS and the PI&Ds of the plant may be used to define the physical

20 14 12

relationship of equipment (e.g., that a crude tank is upstream of desalter which is upstream of a heater which is upstream of an atmospheric tower).

[00189] Analytics modules may use any of this data along with the processing performed by data pipelines to perform on-going or on-demand analytics. The analytics modules described herein may thus provide the pipelines defined with the data studio with an executable context. The analytic modules may have a tag, a description, an execution rate, and may support data pipeline blocks and parameters. The analytic modules may be downloaded into any computer or execution engine in the system and may automatically resolve references to data that may be sent via the monitoring and analytics bus. Analytic modules may also be moved between computing elements and the configuration system may automatically rebind data links within this happens to assure the data pipelines remain intact.

[00190] Still further, once analytic modules have been debugged, they may be stored as templates that can be re-used for similar applications. In this manner, libraries of analytics templates may be created and reused many times. If needed these modules may also be licensed and resold.

20 14 12

[00191] Moreover, if desired, the analytic modules, and/or the data pipelines used within or to support analytic modules may be debugged using, for example, the data studio 509. If the analytics predictions or data processing are off, the user may use the data studio 509 to operate or run the module (or data pipeline) on-line and to view data values at each block in the data pipeline or module. The user can then use the data studio 509 to see data values and correct blocks, algorithms within the blocks, etc. leading to the error.

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 or monitor a process in real-time. Moreover, as used herein, a "piece of equipment," "equipment piece," or "equipment," 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.

[00192] When implemented in software, any of the applications, blocks, 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.

[00193] 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 scope of the invention.

20 14 12

CLAIMS

1. A plant communication system for use within a process plant environment that implements a process, comprising:

a process control network including a multiplicity of process control devices disposed within the process plant to control the process and a process control communication network communicatively coupled to the multiplicity of process control devices, wherein one or more of the multiplicity of process control devices collects or generates process control data;

a further plant communication network including a plurality of processing devices communicatively coupled together by a further plant communication network link, wherein at least one of the plurality of processing devices on the further plant communication network is communicatively connected to the process control communication network; and

a distributed data pipeline that is defined, in a configuration environment, to perform, on process control data, a particular processing operation by using a series of data processing modules communicatively connected together in a particular communicative order across a plurality of different devices of the process plant environment, and different portions of the distributed data pipeline are downloaded from the configuration environment into the plurality of different devices of the process plant environment for runtime execution of the particular processing operation, the distributed data pipeline including one or more upstream data processing modules coupled to a set of downstream data processing modules, wherein:

each data processing module included in the set of downstream data processing modules within the distributed data pipeline processes data received from a respective one or more upstream data processing modules within the distributed data pipeline,

at least one of the one or more upstream data processing modules of the distributed data pipeline is disposed in one of the multiplicity of process control devices connected to the process control communication network, and

at least one of the downstream data processing modules of the distributed data pipeline is disposed in one of the plurality of processing devices on the further plant communication network.

2. The plant communication system of claim 1, further including a database communicatively coupled to one of the downstream data processing modules of the data

220

141

pipeline for storing the data generated by the one of the downstream data processing modules of the distributed data pipeline and making the stored data available to other applications via the further plant communication network, the other applications including at least one analytics module or application.

3. The plant communication system of claim 1, wherein one of the upstream data processing modules is disposed in the process control network and collects raw process control data for processing by the distributed data pipeline, and further including a database communicatively coupled to the process control communication network for independently storing the collected raw process control data.

4. The plant communication system of claim 3, wherein the database is disposed in one of the processing devices coupled to the further plant communication network.

5. The plant communication system of claim 1, wherein the further plant communication network is a big data communication network.

6. The plant communication system of claim 1, wherein the further plant communication network comprises a plurality of different communication networks coupled together.

7. The plant communication system of claim 6, wherein the plurality of different communication networks includes a first plant based communication network and a second cloud based communication network.

8. The plant communication system of claim 6, wherein the distributed data pipeline includes at least one different data processing module stored in a respective processing device within each of the plurality of different communication networks.

20 14 12

9. The plant communication system of claim 6, wherein at least one of the plurality of different communication networks is a wireless communication network.

10. The plant communication system of claim 1, wherein the process control communication network comprises a plurality of different control communication networks communicatively coupled together.

11. The plant communication system of claim 1, wherein one of the data processing modules is a data collection module that obtains data from a data source within the process control network.

12. The plant communication system of claim 11, wherein one of the data processing modules is a data cleaning module that cleans data obtained from the data source within the process control network.

13. The plant communication system of claim 11, wherein one of the data processing modules is a data alignment module that time aligns data from multiple data sources within the process control network.

14. The plant communication system of claim 11, wherein one of the data processing modules is a sensitivity data analysis module that determines the sensitivity of data to a particular event within the plant.

15. The plant communication system of claim 11, wherein one of the data processing modules is a causality analysis module that determines a causal relationship between data from a data source and an event within the plant.

16. The plant communication system of claim 11, wherein one of the data processing modules is a learning module that performs supervised learning on the data.

20 14 12

17. The plant communication system of claim 11, wherein one of the data processing modules is a learning module that performs unsupervised learning on the data.

18. The plant communication system of claim 11, wherein one of the data processing modules is a data presentation module that presents data generated within the distributed data pipeline to a user.

19. The plant communication system of claim 1, wherein the distributed data pipeline includes one or more data collection modules that collect process data from the process control network, one or more intermediate data processing modules that process the data collected by one or more of the data collection modules or another intermediate data processing module, and one or more end-use data processing modules that store output data generated by the one or more intermediate data processing modules or the one or more end-use data processing modules in a manner that can be accessed by one or more other applications.

20. The plant communication system of claim 19, wherein one or more of the data processing modules of the distributed data pipeline is intermittently connected within the distributed data pipeline.

21. The plant communication system of claim 20, wherein one of the one or more intermittently connected data processing modules is a data display module that displays data produced within the distributed data pipeline to a user.

22. The plant communication system of claim 20, wherein one of the one or more intermittently connected data processing modules is disposed in a handheld device that is intermittently connected to the further plant communication network.

20 14 12

23. The plant communication system of claim 19, wherein the output data is simultaneously accessible by two or more other applications.

24. The plant communications system of claim 1, wherein one of the multiplicity of process control devices is a process controller that executes one or more control routines at a controller execution rate, and wherein the process control data collected by the one or more upstream data processing modules of the distributed data pipeline disposed in one of the multiplicity of process control devices collects process control data at a rate greater than the controller execution rate.

25. The plant communications system of claim 24, wherein two or more of the data processing modules within the distributed data pipeline process collected data at a rate greater than the controller execution rate.

20

14 12

26. A method of processing data in a plant having a process control network including a multiplicity of process control devices disposed within the plant to control a process and a process control communication network communicatively coupled between the multiplicity of process control devices, wherein one or more of the multiplicity of process control devices collects or generates process control data, and including a further plant network including a plurality of processing devices communicatively coupled together by a further plant communication network link, the method comprising:

storing a distributed data pipeline within the process control network and in the further plant network, the distributed data pipeline defined, in a configuration environment, as a series of different data processing modules that are configured to be communicatively connected together in a particular order across a plurality of different devices of the process plant environment, and different portions of the distributed data pipeline are downloaded from the configuration environment into the plurality of different devices to thereby perform, during run-time, a processing operation on process control data, the plurality of different devices including various different ones of the process control devices and processing devices;

collecting data at one or more of the data processing modules of the distributed data pipeline during on-line operation of the process control network;

processing the collected process data in the data processing modules of the distributed data pipeline during on-line operation of the process control network;

storing the processed data developed by the data processing modules of the distributed data pipeline in a computer memory device; and

using the data developed by the data processing modules of the distributed data pipeline in a further application executed on a further processing device.

27. The method of processing data of claim 26, wherein storing the processed data developed by the data processing modules of the distributed data pipeline in a computer memory device includes storing the processed data in a big data machine.

28. The method of processing data of claim 26, wherein storing the distributed data pipeline within the process control network and in the further plant network includes storing at least one of the data processing modules within one of the multiplicity of process control devices and includes storing another one of the data processing modules in one of the processing devices of the further plant network.

29. The method of processing data of claim 26, wherein storing the distributed data pipeline within the process control network and in the further plant network includes storing at least one of the data processing modules in a device within a first plant sub-network of the further plant network and storing another one of the data process modules in a second plant sub-network of the further plant network.

30. The method of processing data of claim 29, wherein the first plant sub-network is a big data network in the plant and the second plant sub-network is a cloud based network.

31. The method of processing data of claim 26, wherein collecting data at one or more of the data processing modules of the data pipeline during on-line operation of the process control network includes collecting data at a rate faster than an execution rate of a controller used within the process control network to perform on-line process control.

32. The method of processing data of claim 26, wherein at least one of the process control network or the further plant network includes a wireless communication network.

33. The method of processing data of claim 26, wherein one of the data processing modules is a data cleaning module that executes to clean data obtained from a data source within the process control network.

34. The method of processing data of claim 26, wherein one of the data processing modules is a data alignment module that executes to time align data from multiple data sources within the process control network.

35. The method of processing data of claim 26, wherein one of the data processing modules is a sensitivity data analysis module that executes to determine the sensitivity of data to a particular event within the plant.

36. The method of processing data of claim 26, wherein one of the data processing modules is a causality analysis module that executes to determine a causal relationship between data from a data source and an event within the plant.

37. The method of processing data of claim 26, wherein one of the data processing modules is a learning module that executes to perform supervised or unsupervised data learning.

20 14 12

38. The method of processing data of claim 26, wherein one of the data processing modules is a data presentation module that executes to present data generated within the distributed data pipeline to a user.

39. The method of processing data of claim 26, wherein the distributed data pipeline includes one or more data collection modules that collect process data from the process control network, one or more intermediate data processing modules that process the data collected by one or more of the data collection modules or another intermediate data processing module, and one or more end-use data processing modules that store output data generated by the one or more intermediate data processing modules or the end-use data processing modules in a manner that can be accessed by one or more other applications.

40. The method of processing data of claim 39, wherein one of the data processing modules of the distributed data pipeline is intermittently connected within the distributed data pipeline.

41. The method of processing data of claim 40, wherein the intermittently connected data processing module is disposed in a handheld device that is intermittently connected to the further plant communication network.

20 14 12