Title: DYNAMIC SCHEMA FOR UNIFIED PLANT MODEL

TECHNICAL FIELD

[0001] The present invention relates generally to industrial system automation, and more particularly to automatically configuring an industrial system based at least in part upon metadata associated with the system.

BACKGROUND OF THE INVENTION

[0001] Advancements in computer technology and software have enabled communication to occur almost instantaneously. For example, through email an individual in New York can receive a document, image, message, or the like from an individual in California in a matter of seconds. Such advances in technology are not limited solely to communications, but directly or indirectly affect every business and individual. For instance, industrial control systems have enabled modern factories to become partially or completely automated in many circumstances, thereby reducing costs associated with the factory and consequently passing on such cost reduction to a consumer.

[0002] To further reduce costs of production, industrial systems have become increasingly automated. For example, programmable logic controllers (PLCs) can be coupled to various machinery and control a process that requires precise timing between actions of such machinery. Moreover, as communications technology has continued to advance, intercommunication across geographic boundaries has enabled growth of distributed manufacturing environments. For instance, a first plant in a first geographic location can be utilized to manufacture a window frame, while a second plant in a second geographic location can be employed to manufacture a corresponding window. Obviously, inefficiencies in manufacturing would result if a far greater number of windows were created than doorframes, as an excess of windows would exist. For example, storage for maintaining the excess windows may be required, dispensing with or recycling the windows may be required, and/or altering the generated windows to meet design specifications for a disparate window frame may be required. These and other costs are often associated with

manufacturing system that do not have an efficient communication/automation system associated therewith.

[0003] Systems and methodologies to facilitate automation across geographic/industrial system boundaries have been developed and implemented in various factory settings. Such systems and methodologies, however, are typically rules-based and are not flexible with altering requirements of a manufacturing environment. For example, a manufacturing device that replaces an outdated manufacturing device within a system may not be compatible with other devices in such system given a desired process. Accordingly, an expert in computer programming and one familiar with the system/device is required to render the device compatible with other devices within the system as well as the computer platform that enables automation. Requiring this extra implementation of a device into an existing system results in downtime of the system, lost man-hours, and additional cost of the programmers that enable implementation. Similarly, implementing a new process while utilizing existing equipment can require a substantial amount of time.

[0004] Further, communications/computerized components employed in connection with industrial automation systems may not be properly configured to properly deliver databases and/or database entries to a user. For example, an industrial automation system may be configured to generate a database according to a proprietary internal format. Such database, however, may be desirably displayed in a format that can be viewed from remote locations, wherein such remote locations have not been updated according to the proprietary format. In conventional industrial automation systems, this alteration of database formats requires a computer programmer to generate code specifically for the transformation between formats. Such programming is rule-based, and does not adapt well to altering conditions of an industrial system. Further, if the databases are dynamically created (*e.g.*, the databases receive a constant flow of data from sensors, machinery, input from operators, ...), a schema mapping from a first format to a second format must be robust and adaptive, and again requires programming by an expert.

[0005] Accordingly, there exists a need in the art for a system and/or methodology for automatically configuring an industrial automation system while mitigating needs for expert programming services.

SUMMARY OF THE INVENTION

[0006] The following presents a simplified summary of the invention in order to provide a basic understanding of some aspects of the invention. This summary is not an extensive overview of the invention. It is intended to neither identify key or critical elements of the invention nor delineate the scope of the invention. Its sole purpose is to present some concepts of the invention in a simplified form as a prelude to the more detailed description that is presented later.

[0006a] According to a first aspect of the present invention, there is provided a platform embodied on a computer-readable storage medium that facilitates automatic configuration of at least a portion of an industrial system, comprising: a data model representative of the industrial system, wherein the data model accords to an XML schema and comprises metadata that describes relationships between physical entities represented within the data model, data objects that represent the physical entities within the industrial system, and data objects that represent processes undertaken by the physical entities represented within the data model; one or more sensors that detect an addition, modification, or removal of a first physical entity within the industrial system; and a configuration component that automatically configures one or more of the physical entities represented within the data model by generating code for utilization by the one or more physical entities, the configuration based at least in part upon the data model and metadata therein; wherein the data model is modified based on the detection of the addition, modification or removal of the first physical entity; and a system controller represented within the data model is automatically reprogrammed based on the modified data model.

[0006b] According to a second aspect of the present invention, there is provided an XML database comprising the platform as defined in the first aspect.

[0006c] According to a third aspect of the present invention, there is provided a SQL database comprising the platform as defined in the first aspect.

[0006d] According to a fourth aspect of the present invention, there is provided a system comprising executable components embodied on a computer-readable storage medium that when executed on a processor facilitates industrial automation, the system comprising: a component that retrieves metadata associated with an industrial environment and generates a metadata based model of the industrial environment; a sensor component that detects an addition, modification, or deletion of a first industrial device; and a configuration component that updates the metadata based model in

response to the detected addition, modification or deletion and automatically configures one or more industrial devices based at least in part upon the updated metadata based model.

[0006e] According to a fifth aspect of the present invention, there is provided a method for automatically configuring at least a portion of an industrial system, comprising: receiving a data model that is representative of the industrial system, wherein the data model conforms to an XML schema and comprises metadata that describes relationships between physical entities represented within the data model, data objects that represent the physical entities within the industrial system, and data objects that represent processes undertaken by the physical entities represented within the data model; configuring a machine within the industrial system by automatically generating code and applying the code to the machine based at least in part upon a user request and the data model, sensing whether an alteration to a first physical entity within the industrial system occurs; modifying the data model based at least in part upon the sensed alteration by at least one of adding, removing, and altering a data object that represents the first physical entity; and automatically generating software to re-configure a second physical entity within the industrial system based on the modified data model.

[0007] Embodiments of the present invention provide a system and/or methodology for automatically configuring an industrial system. Benefits according to embodiments of the present invention include configuration of industrial device(s)/process(es) without requiring manually programming interfacing therebetween. Further, embodiments of the present invention provide for automatic generation and structuring of databases to be employed in connection with an industrial environment. For example, it may be desirable to store data relating to an industrial environment within a database that can be viewed at disparate locations. Accordingly, an XML database or the like may be desirable to allow for viewing with a conventional browser. Moreover, it maybe desirable to store an industrial automation platform in a data store of a particular format.

[0008] The above and other benefits can be accomplished by utilizing metadata in connection with intelligent components that structure and analyze such metadata. The metadata can be utilized to define/describe relationships and/or resources between physical device(s) within an industrial environment, process(es) within an industrial environment, resources between software objects and/or

applications, associations between software objects and/or applications, and the like. Further, the metadata cannot only describe relationships between objects that do not inherit from one another, but can farther describe inheritance relationships between various objects. Thus, metadata can exist that robustly describes the industrial environment, and enables generation of a robust representation of the industrial environment.

[0009] Such a robust metadata-driven representation enables automatic configuration of software applications as well as configuration of physical device(s) and/or process(es) within the industrial system. For instance, a user can request that a

particular process be configured and implemented. An analysis component can analyze the metadata existent within the system and automatically configure physical device(s) required to implement the requested process. Further, an analysis of the metadata enables generation of software objects required for the desirable implementation, and database generation and structuring that is utilized to store/analyze data relating to the requested process. Such automatic configuration is a substantial benefit over conventional industrial automation systems, which require manual programming of software required for implementation.

[0010] In accordance with embodiments of the present invention, metadata can be collected according to particular users. For example, user identity, user access privileges, *etc,* can be defined/described within the metadata. Thereafter, when the user desires to view data within the automation platform according to embodiments of the present invention, filters can be dynamically created that provide the user with only pertinent data. Such filters are beneficiary due to robustness of the industrial system representation. Specifically, if the user were provided with all data relating to the entire industrial system he would be overwhelmed due to massive quantities of such data. Further, these filters can be built directly into the metadata, enabling them to be adaptive over time with respect to alterations in the industrial system and alterations with respect to users.

[0011] Moreover, a controller that is employed to control one or more device(s)/process(es) within an industrial automation environment can utilize metadata associated with embodiments of the present invention. For example, a controller can be provided with metadata that describes/defines the industrial system, and thereafter utilize such metadata to control industrial machine(s)/process(es). As more metadata is generated and accessible to the controller, more intelligent control decisions can be made. The controller can quickly locate desirable metadata by analyzing associations defined in the metadata. For example, a links table can be provided to enable the controller to quickly locate and utilize metadata in connection with controlling an industrial machine/process.

[0012] To the accomplishment of the foregoing and related ends, embodiments of the invention then, comprise the features hereinafter fully described and particularly pointed out in the claims. The following description and the annexed drawings set forth in detail

certain illustrative aspects of the invention. These aspects are indicative, however, of but a few of the various ways in which the principles of the invention may be employed and the present invention is intended to include all such aspects and their equivalents. Other objects, advantages and novel features of the invention will become apparent from the following detailed description of the invention when considered in conjunction with the drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] Fig. 1 is a high-level block diagram of a system that facilitates automatic configuration of at least a portion of an industrial system in accordance with an aspect of the present invention.

[0014] Fig. 2 is a block diagram of a system that facilitates automatic configuration of at least a portion of an industrial system in accordance with an aspect of the present invention.

[0015] Fig. 3 is a block diagram of a system that facilitates automatic configuration of at least a portion of an industrial system in accordance with an aspect of the present invention.

[0016] Fig. 4 is a block diagram of a system that facilitates automatic configuration of at least a portion of an industrial system in accordance with an aspect of the present invention.

[0017] Fig. 5 is a flow diagram illustrating a methodology for automatically configuring at least a portion of an industrial system in accordance with an aspect of the present invention.

[0018] Fig. 6 is a flow diagram illustrating a methodology for filtering data within a metadata driven automation platform in accordance with an aspect of the present invention.

[0019] Fig. 7 is a flow diagram illustrating a methodology for automatically generating a database in connection with an industrial environment in accordance with an aspect of the present invention.

[0020] Fig. 8 is a flow diagram illustrating a methodology for automatically configuring at least a portion of an industrial environment in accordance with an aspect of the present invention.

[0021] Fig. 9 is a flow diagram illustrating a methodology for declaring an application and automatically configuring an automation platform based at least in part upon the declaration.

[0022] Fig. 10 is an exemplary transformation of an automation platform from a first format to a second format in accordance with an aspect of the present invention.

[0023] Fig. 11 is an exemplary filtering embodiment in accordance with an aspect of the present invention.

[0024] Fig. 12 is an exemplary automation platform that automatically updates association/resources between objects in accordance with an aspect of the present invention.

[0025] Fig. 13 is an exemplary embodiment illustrating control of a plurality of industrial devices in accordance with an aspect of the present invention.

[0026] Fig. 14 is an exemplary configuration of a plurality of devices in accordance with an aspect of the present invention.

[0027] Fig. 15 illustrates an exemplary manner in which metadata can be employed in connection with a controller in accordance with an aspect of the present invention.

[0028] Fig. 16 illustrates data collection in connection with a controller in accordance with an aspect of the present invention.

[0029] Fig. 17 illustrates a distributed industrial system utilizing a metadata store to control device(s)/process(es) within the distributed industrial system in accordance with an aspect of the present invention.

[0030] Fig. 18 is an exemplary operating environment in which the present invention can function.

[0031] Fig. 19 is an exemplary computing environment that can be employed in connection with the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0032] The present invention is now described with reference to the drawings, wherein like reference numerals are used to refer to like elements throughout. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It may be

evident, however, that the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to facilitate describing the present invention.

[0033] As used in this application, the terms "component," "handler," "model," "system," and the like are intended to refer to a computer-related entity, either hardware, a combination of hardware and software, software, or software in execution. For example, a component may be, but is not limited to being, a process running on a processor, a processor, an object, an executable, a thread of execution, a program, and/or a computer. By way of illustration, both an application running on a server and the server can be a component. One or more components may reside within a process and/or thread of execution and a component may be localized on one computer and/or distributed between two or more computers. Also, these components can execute from various computer readable media having various data structures stored thereon. The components may communicate *via* local and/or remote processes such as in accordance with a signal having one or more data packets (*e.g.*, data from one component interacting with another component in a local system, distributed system, and/or across a network such as the Internet with other systems *via* the signal).

[0034] Referring now to the drawings, Fig. 1 illustrates a high-level system overview in connection with one particular aspect of the subject invention. The present invention relates to a novel system 100 that facilitates automation of one or more of a manufacturing process and/or entity. Factory automation is often accomplished at least in part by a software vendor providing a framework for a manufacturing system, and a customer requesting that such framework be adapted/completed according to their particular manufacturing needs. Accordingly, alterations in a manufacturing system/process require the software vendor to modify the software to accommodate the aforementioned alterations. Further, proprietary machinery often needs specialized configuration to enable it to effectively communicate with a controller as well as other machinery within the manufacturing system. The system 100 is employed to provide a common model for manufacturing factories (plants) from a system perspective shared across software vendor and

customer applications. Further, the common model that is facilitated by the present invention can span all industry segments.

[0035] The system 100 includes an industrial system 102 and an automation platform 104 that comprises a representation 106 of the industrial system. The industrial system 102 can include various industrial machinery (*e.g.*, pumps, presses, conveyors,...), programmable logic controllers (PLCs), switches, sensors, servers, databases, or any other suitable software/device that can be used in connection with an industrial system. The representation 106 effectively represents entities 108 within the industrial system 102 by employing metadata 110 associated with the industrial system 102 as well as software objects that are employed to represent such industrial system 102. Metadata is data that describes data. Conventionally, metadata has been employed in connection with the World Wide Web due to a need to locate particular information amongst a massive amount of information. The system representation 106 is more intelligent and adaptive than conventional database representations of industrial systems, as the metadata 110 can expand and change with system alterations without requiring an expert programmer. The metadata 110 includes data that defines associations between the system entities 108 within the automation platform 104 (and within the industrial system 102). The metadata 110 can further include definitions of industrial processes that occur within the industrial system 102, and the system entities 104 can include representations of such processes. More particularly, the system entities 108 within the system representation 106 can include objects that represent disparate industrial processes, industrial machinery, servers, *etc.*, and the metadata 110 can further define such objects. Furthermore, a collection of objects can be employed to represent disparate processes/machinery within the industrial system 102. Thus, it is understood that the system representation 106 is not limited to a specific manner of representing the industrial system 102, and that any suitable manner of representation is contemplated by the present invention.

[0036] A configuration component 112 analyzes the system representation 106 and configures the industrial system 102 in accordance with such analysis. The configuration component 112 is an intelligent component that configures the industrial system 102 and/or the system representation 106 based at least in part upon the metadata 110. In accordance with one aspect of the present invention, sensor(s)

within the industrial system can determine existence of a new or altered piece of machinery, and inform the automation platform 104 of such addition. The configuration component 112 can thereafter analyze the system representation 106 to automatically configure the machinery in accordance with the metadata 110 that describes the industrial system 102. For example, the new/altered machinery can itself include metadata that is delivered to the automation platform 104 *via* a suitable network connection, and the metadata relating to the machinery can be added to the existent metadata 110. In accordance with another aspect of the present invention, metadata associated with machinery can be manually entered in by an operator/user of the automation platform 104. For instance, existence of a new piece of machinery can be detected, and the configuration component 112 can prompt a user to enter metadata relating to the piece of machinery. Particularly, the configuration component 112 can determine metadata required to configure the machinery, and provide a template or other suitable data entry mechanism to a user. Upon receiving the requested metadata, the configuration component 112 can configure the industrial system 102 *via* the system representation 106.

[0037] Furthermore, the configuration component 112 can automatically generate user interfaces in accordance with the metadata 110 as well as interfaces that facilitate data exchange between the industrial system 102 and the system representation 106. For instance, the metadata 110 can include information relating to a particular user of the automation platform 104, such as user identity, user privileges, user activities, and other suitable information that can be utilized in connection with creating an appropriate user interface. Therefore, a customized user interface can be automatically generated by the configuration component 112.

[0038] Moreover, the configuration component 112 can be employed to automatically structure databases for use when collecting data relating to the industrial system 102. For example, the industrial system 102 can include a plurality of sensors and can generate a substantial amount of data relating to manufacturing system(s)/process(es). Conventionally, collecting such data required a substantial amount of manual programming to generate a database structure suitable for the particular data collected, resulting in wasted man-hours. The system 100 enables a user to request data collected from a particular machine, set of machines, and/or

process, and the configuration component 112 automatically generates a database structure that collects data in accordance with the request. The configuration component 112 can utilize the system representation 106 to determine machinery, sensor(s), and the like within the industrial system 102 that should be monitored to acquire the desired data. For example, the configuration component 112 can receive a request from a user to configure the physical system 102 for a specific process, and structure a database that collects data relating to that process. Based upon the system representation 106 in general, and the metadata 110 in particular, the industrial system 102 can be automatically configured for a desired process. Further, databases that are employed in connection with monitoring the process and controlling the process can be automatically structured to collect/organize data associated with the process. The generated databases can be in any suitable format, including SQL, XML, and any other suitable database format/language. In accordance with one aspect of the present invention, the system 100 can facilitate transformation of a database from a first structure/language (SQL) to a disparate structure/language (XML) to enable viewing of the data on a browser.

[0039] In accordance with yet another aspect of the present invention, the automation platform 104 is an extensible automation platform, wherein the metadata 110 expands as the industrial system 102 is utilized. This enables the automation platform 104 to more intelligently and efficiently configure the industrial system 102 over time as more metadata is generated/added to the metadata 110 within the system representation 106. Further, controllers utilized to control machinery/process(es) within the industrial system 102 can utilize the metadata 110 in connection with system control, thereby rendering such controllers adaptive to changing conditions within the industrial system 102.

[0040] In accordance with one aspect of the present invention, the automation platform 104 can be developed in a form of an XML schema (rather than a database), and from such schema the industrial system 102 can be configured and database generation associated with the industrial system 102 can be accomplished. Further, a standard that facilitates data exchange and semantics associated with data exchange can be employed in connection with the XLM schema. For example, ISA 95, ISA 88, and any other suitable standard employed in connection with manufacturing can be

utilized to facilitate creation of the XML schema (and consequently the automation platform 104). The standards can be employed in connection with data exchange between machinery and the automation platform 104 as well as between customer and vendor applications.

[0041] Now turning to Fig. 2, a system 200 that facilitates automatic configuration of an industrial system 202 is illustrated. The industrial system 202 includes physical device(s) 204 such as machinery, PLCs, racks, cables and various other entities associated with manufacturing systems. The physical device(s) 204 can also include computer-related items, such as hubs, a network connection, servers, data stores, and other suitable computer-related items. Thus, the physical device(s) 204 include any physical entity that can be resident within the industrial system 202. The industrial system 202 further includes process(es) 206 that are undertaken within the industrial system 202. For example, a plurality of the physical device(s) 204 can operate in conjunction to complete one of the process(es) 206 within the industrial system 202.

[0042] The industrial system 202 (*e.g.*, the physical device(s) 204 and/or the process(es) 206) is automatically configured *via* an automation platform 208. The automation platform includes a system representation 210, wherein system entities 212 are represented by data structures (*e.g.*, one or more objects). Particularly, the system entities 212 include physical device(s) 214 and process(es) 216 that represent the physical device(s) 204 and the process(es) 206, respectively, within the industrial system 202. The system representation 210 further includes metadata 218 that describes data/objects employed to represent the system entities 212. Particularly, the metadata 218 includes associations 220 that define relationships amongst the physical device(s) 214, relationships amongst the process(es) 216, and relationships between the physical device(s) 214 and the process(es) 216. Furthermore, the associations 220 can define relationships between software object(s)/entities that represent the physical device(s) 214 and the process(es) 216, as well as relationships amongst data that is employed to generate interfaces and/or exchange data. Thus, the associations 220 within the metadata 218 can define/describe any suitable association existent within the industrial system 202 and/or the system representation 210.

[0043] The metadata 218 further includes resources 222 that define/describe resources available/utilized within the industrial system 202 and/or the system representation 210. For example, the resources 222 can describe available bandwidth in a network connection, machine resources within the industrial system 202, available switches that facilitate communication between machinery, and/or any other suitable resource that is found within the industrial system 202. Moreover, the resources 222 can include computer-related resources, such as processing resources, data storage resources, resources associated with particular objects, *etc.*

[0044] The automation platform 208 further includes a configuration component 224 that intelligently configures the industrial system 202 based at least in part upon the system representation 210. More particularly, the configuration component 224 analyzes the system entities 212 and the associations 220 and the resources 222 associated with the system entities 212 and configures the corresponding industrial system 202 in accordance with the analysis. In accordance with one aspect of the present invention, the configuration component 224 can configure the industrial system 202 based at least in part upon a user request. For example, an operator may desirably configure a subset of the industrial system 202 to effectuate a particular process. The automation platform 208 can generate a graphical user interface that enables a user to adequately describe/define the desired process. The configuration component 224 can analyze the physical device(s) 214 and current process(es) 216 together with the associations 220 and the resources 222, and thereafter configure the industrial system 202 in accordance with the analysis. Furthermore, the automation platform 208 can retain metadata associated with the requested process, thereby enabling the automation platform 208 to be an extensible automation platform. Thus, for example, a process need only be configured once, and can later be re-used if desired. Moreover, database(s) relating to the configured industrial system 202 can be automatically structured and generated for utilization and analysis. For instance, databases that are accessed by a controller to facilitate optimal control of the physical device(s) 204 and/or process(es) 206 can be automatically generated and maintained by the automation platform 208.

[0045] Now referring to Fig. 3, a system 300 that facilitates automatic configuration of an industrial system 302 is illustrated. The system 300 further

provides a mechanism for exchanging data between customer and vendor applications. The system 300 includes an automation platform 304 that is employed to configure machinery/process(es) within the industrial system 302. The automation platform 304 can further be employed to automatically generate database structures that are utilized in connection with collecting and/or analyzing data output from the industrial system 302. The automation platform includes a system representation 306 that is employed to represent the industrial system 302. For example, the system representation can include system entities 308, wherein such system entities 308 can comprise representations of machinery within the industrial system 302, representation of process(es) that are undertaken within the industrial system 302, or any other suitable entity within the industrial system 302 that can be represented. The system representation 306 further includes metadata 310, wherein the metadata at least describes/defines associations 312 amongst and/or between the system entities 308 as well as resources 314 within the industrial system 302 collectively and/or system entities 308 individually.

[0046] The automation platform 304 further includes a configuration component 316 that automatically configures the industrial system 302 based at least in part upon the system representation 306 in general, and the metadata 310 that describes the industrial system 302 in particular. For example, a user could define a particular process that is desirably completed within the industrial system 302 by inputting such process into the automation platform 304. A representation of the process is generated based upon the declared desired process, and the configuration component 316 updates the system representation 306 in accordance with the declared process and the metadata 310. The configuration component 316 can thereafter automatically generate code that enables the industrial system 302 to configure itself for completion of the process. Moreover, the configuration component 316 can automatically generate databases or other suitable storage structures and store such data along with generated code in a storage unit 318. For example, the storage unit 318 can store AJAX Command Definition (ACD) file(s), XML schemas/files, can store the automation platform 304, or any other suitable data. Further, the storage unit 314 can include an external data store that is accessible over a suitable network. Thus, the present invention can employ data from disparate geographic locations as

well as configure industrial systems that are remotely located from the storage unit 318.

[0047] The automation platform 304 can communicate with the industrial system 302 *via* one or more manufacturing interface(s) 320. For example, the manufacturing interface(s) 320 can include a reporting framework 322 that facilitates reporting on occurrences within the industrial system, a collaboration framework 324 that enables machinery within the industrial system to collaborate to complete a process, and a portal framework 326 that facilitates communication between the industrial system 302 and the automation platform 304. For example, the collaboration framework 302 can include one or more adaptors, and the portal framework 326 can include one or more adaptors and/or one or more portlets. The reporting framework 322, the collaboration framework 324, and the portal framework 326 can be developed and applied independently in a free-forming environment. The present invention provides such manufacturing interface(s) with a schema 328 that enables automation of various parts of the manufacturing interface(s) as well as process(es) within the industrial system. In accordance with one aspect of the present invention, a known standard can be employed in connection with the schema 328 to provide common semantics across the system 300. Particularly, ISA 95 and/or ISA 88 can be employed in connection with the present invention.

[0048] Turning now to Fig. 4, a system 400 that facilitates automatic configuration and operation of an industrial system 402 is illustrated. The system 400 includes an automation platform 404 that comprises a representation 406 of the industrial system 402. The system representation 406 can include system entities 408 (*e.g.*, machinery and/or process(es)) and metadata 410 that further describes the industrial system 402 (and consequently the system entities 408). Particularly, the metadata 410 defines/describes associations 412 amongst and/or between the system entities 408 as well as resources 414 of the industrial system 402 in general and/or the system entities 408 individually. Utilizing the metadata 410 provides a more complete and intelligent representation of the industrial system 402 when compared with conventional database models.

[0049] The automation platform 404 further includes a configuration component 416 that automatically configures the industrial system 402. The

configuration is based at least in part upon the associations 412 and the resources 414 within the metadata 410. For example, the industrial system 402 can be automatically configured for a process, together with database generation to store data relating to the process. Moreover, the configuration component can be associated with a schema that facilitates transformation of databases and contents thereof into a desirable format (*e.g.*, XML to enable browser viewing of such data).

[0050] The system 400 further includes an interface generator 418 that automatically generates a customized interface according to user-related metadata 420. For example, the user-related metadata 420 can include user identity, user privileges, geographic location of a user, or any other suitable information that relates to a user. The interface generator 418 can access such metadata 420 and filter portions of the system representation 406 that are not required by a user 424 associated with the metadata 420. A substantial amount of metadata is utilized to represent the industrial system 402 – thus, displaying this metadata to the user 424 would be overwhelming to such user 424 and unproductive. Thus, the interface generator 418 reduces data delivered to the user 424 *via* the filter to provide such user 424 with a manageable amount of data. Consequently, applications should appear as simple to a user as if the system representation 406 was a conventional database representation. Further, the applications utilized in connection with the automation platform 404 will be more manageable because they can take advantage of the metadata 410 and 420 to automate tasks and make intelligent decisions. Further, it is understood that while the interface generator 418, filter 422, and user-related metadata 420 are shown separate from the automation platform 404, the automation platform (schema) can include within it the interface generator 418, filter 422, and/or user-related metadata 420. Thus, filters can be declared directly in the metadata. Accordingly, the filter 422 is metadata-driven and not structure-driven, thereby rendering filters adaptive to changing users and/or industrial system alterations.

[0051] Referring now to Fig. 5, a methodology 500 for utilizing metadata to automatically configure an industrial system is illustrated. While, for purposes of simplicity of explanation, the methodology 500 is shown and described as a series of acts, it is to be understood and appreciated that the present invention is not limited by the order of acts, as some acts may, in accordance with the present invention, occur in

different orders and/or concurrently with other acts from that shown and described herein. For example, those skilled in the art will understand and appreciate that a methodology could alternatively be represented as a series of interrelated states or events, such as in a state diagram. Moreover, not all illustrated acts may be required to implement a methodology in accordance with the present invention.

[0052] At 502, an industrial system is modeled. For example, systems and/or processes can be modeled by a plurality of objects with disparate types. While object-oriented-programming can be utilized in accordance with one aspect of the present invention, it is to be understood that any suitable technique for modeling an industrial system can be employed. Further, the model can include software applications that are employed in connection with an industrial system. For instance, particular manufacturers may have specific applications that are employed in connection with a manufacturing process. These applications can be represented within the industrial system model.

[0053] At 504, metadata defining associations and resources within the industrial model is received. The received metadata can include a links table that defines associations between particular objects/types within the model. Further, the metadata can define/describe physical machinery within the industrial system and associations between such physical machinery as well as associations between process(es), process(es) and machines, or any other suitable relationships within an industrial system. Moreover, the received metadata can define/describe relationships between software applications utilized in connection with the industrial system, databases within the industrial system, computer resources within the industrial system, *etc.* Similarly, the metadata can describe computing resources and/or industrial resources within the industrial system. Thus, a robust description of an industrial system can be generated based upon the metadata.

[0054] At 506, the model is altered. For example, a particular piece of industrial machinery can be modified, added, updated, replaced, removed, or any other suitable operation. In accordance with another aspect of the present invention, the model can be altered according to a desired change within the industrial system and/or to reflect updates in hardware/software. At 508, metadata associated with the alteration is received. For example, if machinery is removed, sensors can notice such

removal and generate metadata relating to the removal (*e.g.*, updated associations, updated resources, ...). In another example, a software application can be updated, and metadata relating to the update can be extracted and received.

[0055] At 510, the collection of metadata is analyzed with respect to the alteration in the model. For example, an intelligent configuration component can review the metadata with respect to the altered model, and at 512 automatically configure the industrial system in accordance with the analysis. For instance, upon review and analysis of the metadata, the configuration component can automatically generate software to be employed in a PLC that effectively controls one or more machines for a particular process. Further, the configuration component can automatically create user interfaces, databases, and any other suitable software that requires generation to effectuate configuration of the system as well as automation thereof.

[0056] Turning now to Fig. 6, a methodology 600 for filtering data within a robust system representation and providing such filtered data to a user is illustrated. At 602, a model of an industrial system is received. For example, the model can utilize various objects of disparate classes and types to represent the industrial system. For instance, one or more objects can represent one or more machines and/or processes within the industrial system. At 604, metadata is associated with the model. Particularly, the model includes metadata that defines and/or describes associations between machinery within the industrial system. For example, the metadata can describe processes undertaken by machinery and how such machinery interacts to facilitate automation of the processes. Moreover, the model includes metadata that defines and/or describes resources existent within the industrial system (*e.g.*, available machinery, capacity of machinery, PLC processing capabilities, ...). In accordance with another aspect of the present invention, metadata within the model describes data utilized to generate the model. For instance, various objects can be employed to facilitate representation of the industrial system. The metadata can define and/or describe relationship(s) between such objects as well as resources associated with the objects.

[0057] At 606, metadata is associated with one or more user(s). For example, metadata describing a user's identity, user privileges, user location, or any other

suitable information relating to a user can be collected and employed within the methodology 600. The metadata can be manually generated through a template or a series of automatically generated interrogatories that request particular information from the user. Further, this metadata can be generated automatically by watching a user over time and providing a component that makes inferences relating to the user. As used herein, the term "inference" refers generally to the process of reasoning about or inferring states of the system, environment, and/or user from a set of observations as captured *via* events and/or data. Inference can be employed to identify a specific context or action, or can generate a probability distribution over states, for example. The inference can be probabilistic – that is, the computation of a probability distribution over states of interest based on a consideration of data and events. Inference can also refer to techniques employed for composing higher-level events from a set of events and/or data. Such inference results in the construction of new events or actions from a set of observed events and/or stored event data, whether or not the events are correlated in close temporal proximity, and whether the events and data come from one or several event and data sources. Various classification schemes and/or systems (*e.g.*, support vector machines, neural networks, expert systems, Bayesian belief networks, fuzzy logic, data fusion engines...) can be employed in connection with performing automatic and/or inferred action in connection with the subject invention. Thus, for example, a component can infer that a user is associated with particular access privileges by watching which applications the user typically accesses.

[0058] At 608, data is provided to the user based at least in part upon the metadata associated with the model and the metadata associated with the user. This data is less than the entire industrial system model, as delivering such data to a user would be overwhelming due to the substantial amount of metadata that describes the industrial system and/or platform. The data can be delivered to the user *via* a graphical user interface (GUI). For instance, a hierarchical representation of a subset of the industrial system can be provided to the user, wherein the subset is pertinent to the user's activities within the industrial system. Moreover, organization by process, location, machinery, or any other suitable organization can be projected to the user in accordance with the aforementioned metadata. In accordance with one aspect of the

present invention, the metadata itself can declare filters for particular users (*e.g.*, the filters can be declared directly within the metadata). Such filtering is an advancement over conventional filters, as conventional filters required manual entering of disparate views associated with disparate users. The metadata in connection with the model enables automatic generation of views associated with users, as given common semantics amongst the metadata databases utilized for delivering data to particular users can be automatically generated.

[0059] Now referring to Fig. 7, a methodology 700 for automatically generating and configuring a database relating to an industrial system is illustrated. At 702, a model of an industrial system is provided. At 704, metadata is associated with the model. In particular, the metadata defines/describes associations and resources relating to the industrial system, as well as associations and resources amongst software objects that are employed within the model. At 706, a data exchange schema is associated with the model. For example, ISA 95 can be employed in connection with the model to facilitate communications between the model and the industrial system represented by the model. Particularly, the data exchange schema can include transforming components that transform code within the model to code readable by entities within the industrial system (*e.g.*, PLCs).

[0060] At 708, data is exchanged between the model and the industrial system based upon a particular process implemented within the industrial system and modeled within the model. For a simplistic example, a counter can be employed by a piece of machinery, and such numbers can be delivered to the model for storage and/or analysis. At 710, a database is automatically structured and data is entered into such structured database. In conventional data acquisition systems/methodologies within industrial environments, acquiring data requires manually creating a database structure prior to implementation of a new process. The methodology 700 can be employed to automatically generate a database structure that receives data relating to a new process. For instance, the model and metadata associated therewith can be employed to automatically configure an industrial system for a particular process. Further, databases that store information relating to the process can be automatically generated based at least in part upon the aforementioned metadata, thereby increasing manufacturing efficiency. Upon structuring of the database, data acquired from the

process (*e.g., via* sensors) is directed to the database for appropriate storage. Further, the database can be structured according to user desires. For instance, the database can be automatically configures as an XML database.

[0061] Now turning to Fig. 8, a methodology 800 that facilitates configuration of a portion of an industrial system is illustrated. At 802, a model of an industrial system is provided. For example, the model can include representations of process(es) and/or machinery within the industrial system. In accordance with one aspect of the present invention, object-oriented programming can be utilized in connection with creating the model. At 804, metadata is associated with the model. The metadata defines and/or describes associations and resources related to the industrial system as well as associations and resources related to the industrial model. At 806, the model can receive a request for a particular task. For example, the model can receive a request for automatically configuring a plurality of machines to implement a desirable process. For another example, the model can receive a request to structure an XML database relating to a particular process.

[0062] At 808, data related to the requested task is automatically collected and analyzed. For example, a data collection component can be declared, wherein such component reviews the industrial system the model and obtains data relating to the requested task. For instance, metadata within the industrial system model can be collected and analyzed to determine steps necessary to complete the task. More particularly, if the task is to generate an XML database, metadata relating to storage capacity can be collected and analyzed together with components that are employed to structure XML databases. At 810, a determination is made regarding whether there is sufficient data (*e.g.,* metadata, resources, ...) within the model to complete the requested task. If sufficient data exists, then the model and/or the industrial system are automatically configured to complete the task at 812. For example, if the requested task is to complete an XML database, such database can be automatically configured based upon the request and the collected data.

[0063] If there is insufficient data to complete the task, at 814 a user is provided with a set of alternatives. For example, if the request was to generate an XML database but the resources available only allow for creation of a SQL database, the user can be automatically asked if generation of the SQL database is an acceptable

alternative. A graphical user interface can be employed to provide the user with a list of alternative(s). In another example, a request can be to complete a manufacturing process at a first location within a plant, wherein there exists insufficient machinery at such first location. Thereafter the user can be provided with one or more alternative(s), such as a second location wherein the requested manufacturing process can be completed.

[0064] At 816 the user decides whether one of the alternatives provided at 814 is acceptable. If one of the provided alternatives is acceptable, then the user can select such alternative and the system is automatically configured at 812 according to the selected alternative. Otherwise, the user is provided with instructions that enable completion of the requested task at 818. The collection and analysis of data at 808 enables the methodology 800 to provide the user with instructions and/or resources required to complete a task. For instance, if there does not exist sufficient defined linkages to generate an XML database, then the user can be asked to provide such linkages to enable creation of the XML database.

[0065] Now referring to Fig. 9, a methodology 900 for utilizing an automation platform as a declarative model is illustrated. At 902, a desirable application is declared within an automation platform that includes a model of an industrial system. For example, the desirable application can be a manufacturing task completed within the industrial system and/or a computer-based task, such as generating a database structure for analysis of at least a portion of the industrial system. In accordance with one aspect of the present invention, this application can be declared *via* a request by a user. At 904, a data collection object can be declared to enable collection of data relating to the declared application. Such data collection object can be declared automatically within the automation platform and or manually by a user. For example, the user can request that data be collected relating to a particular process, machine, PLC, *etc.*

[0066] At 906, the data collection object searches over the automation platform as well as over the industrial system to collect data pertinent to the declared application (*e.g.*, data relating to industrial system entities and relationships is obtained, as well as data relating to the automation platform). For instance, if the declared application relates to a manufacturing task that is to be implemented by

various interconnected and communicating machinery, the declared data collection object can determine what machinery exists, type of connection between machinery, review control algorithms associated with the machinery, or any other suitable related data that can be employed in connection with obtaining data related to the task. At 908, the industrial system and/or the manufacturing platform are automatically configured for the desired application. For example, machinery within the industrial system can be automatically configured and/or databases relating to a machine and/or the application can be automatically generated.

[0067] Turning now to Fig. 10, an exemplary transformation 1000 of an industrial system model from a first data structure to a second data structure is illustrated. An SQL database 1002 includes an industrial system model 1004 that is utilized in connection with configuring an industrial system as described above. The industrial system model 1004 includes metadata 1006 that describes/defines associations of machinery within the industrial system as well as associations between objects/applications existent within the industrial system model 1004. For example, the industrial system model 1004 can include various software applications (*e.g.*, spreadsheet application, word processing application, ...), and the metadata 1006 can describe/define associations between such applications. Furthermore, the metadata 1006 can describe/define resources associated with the industrial system and/or resources associated with the industrial system model 1004. For instance, software applications within the industrial system model 1004 can be associated with a processor that has finite resources.

[0068] In accordance with one aspect of the present invention, a desired functionality 1010 is determined. In the displayed exemplary transformation 1000, the desired functionality 1010 is a transformation of the SQL database 1002 into an XML database 1012. While the illustrated transformation involves transforming an SQL database into an XML database, it is to be understood that any suitable transformation between database structures can be accomplished through utilization of metadata in accordance with the present invention. The desired functionality 1010 can be entered by a user after booting of a machine, or can be automatically bootstrapped to a service call (*e.g.*, a hard disk boot). Therefore, whenever the service call is received, the desired functionality 1010 is performed. A data collector 1014 is

employed to collect data from the SQL database 1002 that enables transformation of the SQL database to the XML database 1012. For example, the data collector 1014 can be declared as such within a declarative industrial automation platform. Further, if insufficient data exists to complete the desired transformation, a user can be provided with alternative(s) and/or instruction(s) to facilitate such transformation as illustrated with respect to methodology 800 (Fig. 8).

[0069] Upon collection of requisite data by the data collector 1014, a schema 1016 is provided to transform the SQL database 1002 into the XML database 1012. In accordance with one aspect of the present invention, the schema 1016 employs a standard (*e.g.*, ISA 95, ISA 88, ...) to enable common semantics to be employed between the SQL database 1002 and the XML database 1012. Further, the schema 1016 includes intelligent components that can automatically generate software portions associated with the industrial system model 1004 to facilitate desirable transformation to an XML-based industrial system model 1018. For example, the schema 1016 can automatically generate a database instance, C sharp assemblies, set up icon depositories for icons employed for visualization purposes, *etc.* The resultant database 1012 ill include the industrial model 1018 and metadata 1020 associated with the industrial model 1018 in a desired format. For example, the XML database may be desirable due to an ability to view contents of such database on a conventional browser at disparate geographic locations. Thus, the entire industrial system model 1004 can be automatically transformed to a desirable format *via* analyzing the metadata 1006 without requiring manual structuring of a new database in the desirable format.

[0070] Now referring to Fig. 11, an exemplary embodiment 1100 illustrating filtering technologies of the present invention is shown. An automation platform 1102 is accessible by a plurality of users, both for software development and industrial applications. Particularly, in the exemplary embodiment 1100 an enterprise user 1104, a process operator 1106, a production management user 1108, and a control system developer 1110 are all concurrently accessing the automation platform 1102. The users 1102 – 1110 can be all located at a single manufacturing plant, or distributed geographically at disparate manufacturing plants, wherein the automation platform 1102 facilitates configuration/automation of a distributed manufacturing

environment. The users 1102 – 1110 all desire to access/modify disparate portions of the automation platform 1102, which includes metadata (not shown) describing/defining the automation platform 1102 as previously disclosed. Thus, providing each of the users 1102 – 1110 with an entirety of the automation platform 1102 would be overwhelming due to completeness of a system representation (not shown) within the automation platform 1102.

[0071] Conventionally, filters could be manually built for particular users, thereby providing a user only with data that is required for task(s) that such user generally undertakes. In accordance with the present invention, filters for the respective users 1102 - 1110 can be automatically generated based upon metadata associated with the users 1102 – 1110, the industrial system represented within the automation platform 1102, and the automation platform 1102 itself. Thus, the enterprise user 1104 is only provided with information pertinent to such user 1104, the process operator 1106 is only provided with information pertinent to the process operator 1106, *etc.* Generating filters based upon metadata allows for a more intelligent representation of the automation platform 1102 to be delivered to each user when compared to conventional filters, which are inflexible and not adaptive to changes associated with an industrial system, automation platform, and/or user alterations.

[0072] Now referring to Fig. 12, an exemplary industrial system representation 1200 in accordance with the present invention is illustrated. The industrial system representation 1200 includes a plurality of software objects 1202 – 1216 of multiple types. Such objects 1202 - 1216 are described/defined within the objects as well as within metadata 1218 related to the objects 1202 – 1216. Particularly, the metadata 1218 can include data relating to a plurality of associations 1220, where the associations 1220 describe relatedness between the objects 1202 – 1216. In accordance with one aspect of the present invention, the associations 1220 can be a links table. Thus, if a new object 1222 is added to the system representation 1200, the associations 1220 (*e.g.*, the links table) can be updated accordingly.

[0073] Further, the system representation 1200 provides inheritance capabilities between the objects, and the metadata can define inheritance between objects and the associations 1220 can include links between inherited objects. This

provides a much more powerful metadata model than conventional metadata-related systems/methodologies. Particularly, data relating to inheritance within the metadata 1218 makes building applications easier, and gives another dimension in which to manipulate data. Further, automatic ordering of objects is more efficient than manually programming such hierarchy in a data structure. The system representation 1200 illustrates the objects 1202 – 1216 linked together according to a type of association. Such links can be horizontal links (*e.g.*, an association not involving inheritance), vertical links (*e.g.*, an association involving inheritance), or any other suitable relation that can exist between objects.

[0074] Turning now to Fig. 13, an exemplary industrial environment 1300 utilized in connection with an automation platform of the present invention is illustrated. An automation platform is stored within a computing environment 1302, which facilitates automatically configuring and/or automating one or more industrial machine(s)/process(es). The automation platform includes a data collection component that obtains data relating to the industrial environment. Particularly, in the industrial environment 1300 there exists PLCs 1304 and 1306 and a plurality of industrial devices controlled thereby. Particularly, PLC 1304 is connected to and facilitates control of a first industrial device 1308 through an Nth industrial device 1310, where N is an integer greater than 1. PLC 1306 is connected to and facilitates control of a disparate industrial device 1312 through an Mth industrial device 1314, where M is an integer greater than 1. The data collection component can obtain information relating to physical connection of the PLCs 1304 and 1306 with the industrial devices 1308 – 1314. Further, the data collection component can obtain information relating to applications resident upon the industrial devices 1308 – 1314, communication protocols employed in connection with the industrial devices 1308 – 1314, and any other suitable information relating to the industrial devices 1308 – 1314.

[0075] The automation platform can thereafter configure the PLCs 1304 and 1306 to provide the industrial devices 1308 – 1314 with control software and generate communication interfaces between the PLCs 1304 and 1306 and the industrial devices 1308 – 1314. Moreover, user interfaces enabling optimal operation of the industrial devices 1308 – 1314 can be provided to a user. In accordance with one aspect of the

present invention, the PLC 1304 and the PLC 1306 can be located at disparate geographical locations, but be configured through a central automation platform. This machinery in disparate locations can be configured to communicate with one another and complete process(es) *via* the automation platform.

[0076] Referring now to Fig. 14, an exemplary configuration 1400 between devices in an industrial environment is illustrated. A first device 1402 and a second device 1404 are configured to interact with one another. For example, communications are enabled between the first device 1402 and the second device 1404. Similarly, the first device 1402 and the second device 1404 can act conjunctively to complete a process. Thereafter, a third device 1406 can be configured to communicate with and/or operate in conjunction with the second device 1404. Thereafter it may be desirable to configure the first device 1402 with respect to the third device 1406. A data collection component (not shown) can be employed to obtain data relating to the three devices 1402 – 1406, and configuration amongst such devices can occur automatically. In accordance with another aspect of the present invention, data relating to the three devices 1402 – 1406 can already have been obtained by a data collection component. Thereafter, a user can declare that it would be desirable to configure the first device 1402 with the second device 1404 (*e.g.*, rendering an automation platform as a declarative model). At a later time, as all the data is collected, the third device 1406 can be configured with the second device 1404 and/or the first device 1402 *via* a request or declaration. Such an aspect of the present invention provides for an extensible declarative automation platform utilizable in connection with an industrial environment.

[0077] Turning to Fig. 15, an exemplary system 1500 that facilitates control of one or more machine(s) to effectuate an industrial process is illustrated. The system 1500 includes a plurality of devices 1502 – 1506. Specifically, the system 1500 comprises a first device 1502 through an Nth device 1506, where N is an integer greater than 2. The devices 1502 – 1506 are controlled by a controller 1508 that utilizes metadata 1510 in connection with controlling the devices 1502 – 1506. Further, additional metadata can be accessed by the controller 1508 from a metadata store (not shown). In such a metadata rich environment as described above, the controller 1508 can make more intelligent decisions when compared to conventional

controllers. Further, metadata created by the devices 1502 – 1506 can be directed to the controller 1508 to provide additional metadata to the controller 1508 for use in controlling such devices 1502 – 1506. As metadata is continuously added and accessed by the controller 1508, the controller 1508 can make increasingly intelligent decisions and adapt to alterations relating to the devices 1502 – 1506.

[0078] Now turning to Fig. 16, a system 1600 that facilitates metadata collection and/or generation within an industrial system automation platform is illustrated. The system 1600 is an extensible system, wherein the more metadata that is collected/generated, the more structured and more intelligent the system 1600 and parts thereof become. The system 1600 includes a data collector 1602 that facilitates collection and organization of metadata. Particularly, the data collector includes a registration component 1604 and a placement component 1606, wherein the registration component registers incoming metadata from an industrial system 1608 (*e.g.*, configures associations between the received metadata and existent metadata). For example, the data collector 1602 can receive metadata from sensor(s) 1610 from within the industrial system 1608. The received metadata can describe/define process(es) undertaken within the industrial system 1608, machinery within the industrial system 1608, computer-related entities within the industrial system 1608, *etc.* Further, while not shown, an automation platform can automatically generate metadata according to application configuration and/or interfacing between the industrial system 1608 and the automation platform. Thus, data can be collected over an entire set of activities (*e.g.*, physical system, controller, software application, ...). The data collector 1602 collects and registers the received metadata, and places it at an appropriate location within a metadata store 1612. This metadata store 1612 is accessible by a controller 1614 that facilitates control of at least a portion of the industrial system 1608. The controller 1614 further can receive data directly from the sensor(s) 1610 that is not metadata. Further, through utilization of collected metadata within the metadata store 1612 plug-and-play features are available. Specifically, through metadata a device could be "plugged in" to the industrial system 1608, and applications, databases, configurations between machines, and any other suitable steps necessary to configure the device with the industrial system 1608 can be automatically generated *via* metadata.

[0079] Now referring to Fig. 17, an exemplary embodiment 1700 in accordance with an aspect of the present invention is illustrated. A metadata store 1702 is accessed by two disparate industrial systems 1704 and 1706, which are located at different geographical locations. For instance, the first industrial system 1704 can be located within a factory in New York while the second industrial system 1706 can be located within a factory in Nevada. Due to advancement in communication technology, both industrial systems 1704 and 1706 can utilize the single metadata store 1702 to configure and/or automate process(es) and/or machinery within such industrial systems 1704 and 1706. In accordance with another aspect of the present invention, the metadata store 1702 can be a distributed store across various databases. The industrial systems 1704 and 1706 access the metadata store 1702 through any suitable network connection 1708. For example, the network connection can be a T1 connection, Ethernet connection, or any other suitable wireless and/or wireline connection. The industrial systems 1704 and 1706 include controllers 1710 and 1712, respectively, and access the metadata store 1702 *via* the network connection to facilitate control of process(es) and/or machinery within the industrial systems 1704 and 1706. Furthermore, the industrial systems 1704 and 1706 (and controllers 1710 and 1712) can be employed to add metadata to the metadata store 1702. As amount of metadata increases, the controllers 1710 and 1712 will be able to make more intelligent decisions in connection with controlling portions of the industrial systems 1704 and 1706.

[0080] With reference to Fig.18, an exemplary environment 1810 for implementing various aspects of the invention includes a computer 1812. The computer 1812 includes a processing unit 1814, a system memory 1816, and a system bus 1818. The system bus 1818 couples system components including, but not limited to, the system memory 1816 to the processing unit 1814. The processing unit 1814 can be any of various available processors. Dual microprocessors and other multiprocessor architectures also can be employed as the processing unit 1814.

[0081] The system bus 1818 can be any of several types of bus structure(s) including the memory bus or memory controller, a peripheral bus or external bus, and/or a local bus using any variety of available bus architectures including, but not limited to, 11-bit bus, Industrial Standard Architecture (ISA), Micro-Channel

Architecture (MSA), Extended ISA (EISA), Intelligent Drive Electronics (IDE), VESA Local Bus (VLB), Peripheral Component Interconnect (PCI), Universal Serial Bus (USB), Advanced Graphics Port (AGP), Personal Computer Memory Card International Association bus (PCMCIA), and Small Computer Systems Interface (SCSI).

[0082] The system memory 1816 includes volatile memory 1820 and nonvolatile memory 1822. The basic input/output system (BIOS), containing the basic routines to transfer information between elements within the computer 1812, such as during start-up, is stored in nonvolatile memory 1822. By way of illustration, and not limitation, nonvolatile memory 1822 can include read only memory (ROM), programmable ROM (PROM), electrically programmable ROM (EPROM), electrically erasable ROM (EEPROM), or flash memory. Volatile memory 1820 includes random access memory (RAM), which acts as external cache memory. By way of illustration and not limitation, RAM is available in many forms such as synchronous RAM (SRAM), dynamic RAM (DRAM); synchronous DRAM (SDRAM), double data rate SDRAM (DDR SDRAM), enhanced SDRAM (ESDRAM), Synchlink DRAM (SLDRAM), and direct Rambus RAM (DRRAM).

[0083] Computer 1812 also includes removable/non-removable, volatile/non-volatile computer storage media. Fig. 18 illustrates, for example a disk storage 1824. Disk storage 1824 includes, but is not limited to, devices like a magnetic disk drive, floppy disk drive, tape drive, Jaz drive, Zip drive, LS-100 drive, flash memory card, or memory stick. In addition, disk storage 1824 can include storage media separately or in combination with other storage media including, but not limited to, an optical disk drive such as a compact disk ROM device (CD-ROM), CD recordable drive (CD-R Drive), CD rewritable drive (CD-RW Drive) or a digital versatile disk ROM drive (DVD-ROM). To facilitate connection of the disk storage devices 1824 to the system bus 1818, a removable or non-removable interface is typically used such as interface 1826.

[0084] It is to be appreciated that Fig. 18 describes software that acts as an intermediary between users and the basic computer resources described in suitable operating environment 1810. Such software includes an operating system 1828. Operating system 1828, which can be stored on disk storage 1824, acts to control and

allocate resources of the computer system 1812. System applications 1830 take advantage of the management of resources by operating system 1828 through program modules 1832 and program data 1834 stored either in system memory 1816 or on disk storage 1824. It is to be appreciated that the present invention can be implemented with various operating systems or combinations of operating systems.

[0085] A user enters commands or information into the computer 1812 through input device(s) 1836. Input devices 1836 include, but are not limited to, a pointing device such as a mouse, trackball, stylus, touch pad, keyboard, microphone, joystick, game pad, satellite dish, scanner, TV tuner card, digital camera, digital video camera, web camera, and the like. These and other input devices connect to the processing unit 1814 through the system bus 1818 *via* interface port(s) 1838. Interface port(s) 1838 include, for example, a serial port, a parallel port, a game port, and a universal serial bus (USB). Output device(s) 1840 use some of the same type of ports as input device(s) 1836. Thus, for example, a USB port may be used to provide input to computer 1812, and to output information from computer 1812 to an output device 1840. Output adapter 1842 is provided to illustrate that there are some output devices 1840 like monitors, speakers, and printers, among other output devices 1840, which require special adapters. The output adapters 1842 include, by way of illustration and not limitation, video and sound cards that provide a means of connection between the output device 1840 and the system bus 1818. It should be noted that other devices and/or systems of devices provide both input and output capabilities such as remote computer(s) 1844.

[0086] Computer 1812 can operate in a networked environment using logical connections to one or more remote computers, such as remote computer(s) 1844. The remote computer(s) 1844 can be a personal computer, a server, a router, a network PC, a workstation, a microprocessor based appliance, a peer device or other common network node and the like, and typically includes many or all of the elements described relative to computer 1812. For purposes of brevity, only a memory storage device 1846 is illustrated with remote computer(s) 1844. Remote computer(s) 1844 is logically connected to computer 1812 through a network interface 1848 and then physically connected *via* communication connection 1850. Network interface 1848 encompasses communication networks such as local-area networks (LAN) and wide-

area networks (WAN). LAN technologies include Fiber Distributed Data Interface (FDDI), Copper Distributed Data Interface (CDDI), Ethernet/IEEE 1102.3, Token Ring/IEEE 1102.5 and the like. WAN technologies include, but are not limited to, point-to-point links, circuit switching networks like Integrated Services Digital Networks (ISDN) and variations thereon, packet switching networks, and Digital Subscriber Lines (DSL).

[0087] Communication connection(s) 1850 refers to the hardware/software employed to connect the network interface 1848 to the bus 1818. While communication connection 1850 is shown for illustrative clarity inside computer 1812, it can also be external to computer 1812. The hardware/software necessary for connection to the network interface 1848 includes, for exemplary purposes only, internal and external technologies such as, modems including regular telephone grade modems, cable modems and DSL modems, ISDN adapters, and Ethernet cards.

[0088] Fig. 19 is a schematic block diagram of a sample-computing environment 1900 with which the present invention can interact. The system 1900 includes one or more client(s) 1910. The client(s) 1910 can be hardware and/or software (*e.g.*, threads, processes, computing devices). The system 1900 also includes one or more server(s) 1930. The server(s) 1930 can also be hardware and/or software (*e.g.*, threads, processes, computing devices). The servers 1930 can house threads to perform transformations by employing the present invention, for example. One possible communication between a client 1910 and a server 1930 can be in the form of a data packet adapted to be transmitted between two or more computer processes. The system 1900 includes a communication framework 1950 that can be employed to facilitate communications between the client(s) 1910 and the server(s) 1930. The client(s) 1910 are operably connected to one or more client data store(s) 1960 that can be employed to store information local to the client(s) 1910. Similarly, the server(s) 1930 are operably connected to one or more server data store(s) 1940 that can be employed to store information local to the servers 1930.

[0089] What has been described above includes examples of the present invention. It is, of course, not possible to describe every conceivable combination of components or methodologies for purposes of describing the present invention, but one of ordinary skill in the art may recognize that many further combinations and

permutations of the present invention are possible. Accordingly, the present invention is intended to embrace all such alterations, modifications and variations that fall within the spirit and scope of the appended claims. Furthermore, to the extent that the term "includes" is used in either the detailed description or the claims, such term is intended to be inclusive in a manner similar to the term "comprising" as "comprising" is interpreted when employed as a transitional word in a claim.

CLAIMS

1. A platform embodied on a computer-readable storage medium that facilitates automatic configuration of at least a portion of an industrial system, comprising:

a data model representative of the industrial system, wherein the data model accords to an XML schema and comprises metadata that describes relationships between physical entities represented within the data model, data objects that represent the physical entities within the industrial system, and data objects that represent processes undertaken by the physical entities represented within the data model;

one or more sensors that detect an addition, modification, or removal of a first physical entity within the industrial system; and

a configuration component that automatically configures one or more of the physical entities represented within the data model by generating code for utilization by the one or more physical entities, the configuration based at least in part upon the data model and metadata therein; wherein

the data model is modified based on the detection of the addition, modification or removal of the first physical entity; and

a system controller represented within the data model is automatically reprogrammed based on the modified data model.

2. The platform of claim 1, further comprising a reporting framework that reports events that occur within the industrial system, and wherein the configuration component employs the reported events in connection with configuring the one or more physical entities.

3. The platform of claims 1 or 2, further comprising a database automatically configured based at least in part upon the data model.

4. The platform of any one of the previous claims, further comprising an industrial process implemented based at least in part upon the data model.

5. The platform of claim 4, wherein the industrial process is implemented upon a request by a user.

6. The platform of claim 4, wherein the industrial process is implemented *via* a declaration by a user.

7. The platform of any one of the previous claims, further comprising a graphical user interface generated based at least in part upon the data model.

8. The platform of any one of the previous claims, wherein the system controller employs the metadata to control the one or more physical objects within the industrial system.

9. The platform of any one of the previous claims, wherein object oriented programming is utilized in connection with the data model.

10. The platform of any one of the previous claims, further comprising a filtering component that filters information delivered to a user.

11. The platform of claim 10, wherein the filtering component is generated based at least in part upon the metadata.

12. The platform of claim 11, wherein the filtering component is declared within the metadata.

13. The platform of claim 10, wherein the filtering component is generated based at least in part upon metadata relating to the user.

14. The platform of any one of the previous claims, wherein the metadata describes associations between objects utilized in connection with the data model and at least a portion of the industrial system is configured based at least in part upon the described associations.

15. The platform of claim 14, further comprising one or more links tables utilized to describe the associations between the objects.

16. The platform of claim 15, wherein the one or more links tables account for inheritance between objects.

17. The platform of any one of the previous claims, wherein the metadata describes associations between one or more of industrial device(s), industrial process(es), and computer-related devices within the industrial system, and at least a portion of the industrial system is configured based at least in part upon the described associations.

18. The platform of any one of the claims 1-16, wherein the metadata describes resources allocated amongst objects utilized in connection with the data model and at least a portion of the industrial system is configured based at least in part upon the described resources.

19. The platform of any one of claims 1-16, wherein the metadata describes resources associated with one or more of industrial devices, industrial process(es), and computer-related devices within the industrial system, and at least a portion of the industrial system is configured based at least in part upon the described resources.

20. The platform of any one of the previous claims, wherein the data model is based at least in part upon one or more of an ISA 95 and an ISA 88.

21. An XML database comprising the platform of claim 1.

22. A SQL database comprising the platform of claim 1.

23. The platform of claim 1 bootstrapped to a service call.

24. The platform of claim 23, wherein the service call is a hard disk boot.

25. The platform of any one of the previous claims, further comprising a manufacturing framework that facilitates communication between the industrial system and the platform.

26. The platform of claim 25, wherein the manufacturing framework comprises one or more of a reporting framework, a collaboration framework, and a portal framework.

27. The platform of any one of the previous claims, wherein the metadata is obtained from an external data store.

28. The platform of any one of the previous claims, further comprising a transformation component that transforms data within the data model from a first format to a second format.

29. The platform of claim 28, wherein the first format being SQL and the second formal being XML.

30. The platform of claim 1, wherein said platform facilitates automation of industrial systems at disparate geographic locations.

31. A system comprising executable components embodied on a computer-readable storage medium that when executed on a processor facilitates industrial automation, the system comprising:

a component that retrieves metadata associated with an industrial environment and generates a metadata based model of the industrial environment;

a sensor component that detects an addition, modification, or deletion of a first industrial device; and

a configuration component that updates the metadata based model in response to the detected addition, modification or deletion and automatically configures one or more industrial devices based at least in part upon the updated metadata based model.

32. A method for automatically configuring at least a portion of an industrial system, comprising:

receiving a data model that is representative of the industrial system, wherein the data model conforms to an XML schema and comprises metadata that describes relationships between physical entities represented within the data model, data objects that represent the physical entities within the industrial system, and data objects that represent processes undertaken by the physical entities represented within the data model;

configuring a machine within the industrial system by automatically generating code and applying the code to the machine based at least in part upon a user request and the data model,

sensing whether an alteration to a first physical entity within the industrial system occurs;

modifying the data model based at least in part upon the sensed alteration by at least one of adding, removing, and altering a data object that represents the first physical entity; and

automatically generating software to re-configure a second physical entity within the industrial system based on the modified data model.

33. The method of claim 32, wherein the metadata further describes resources associated with the physical entities.

34. The method of claims 32 or 33, further comprising:

receiving metadata associated with the sensed alteration, the metadata describing relationships between the altered physical entity and other physical entities within the industrial system,

wherein reconfiguration of the second physical entity within the industrial system is based at least in part upon the sensed alteration and the received metadata.

Dynamic Schema For Unified Plant Model

ABSTRACT

An industrial system configuration platform facilitates automation of an industrial system. The industrial system configuration platform comprises a representation of the industrial system and a configuration component that automatically configures at least a portion of the industrial system based at least in part upon metadata that describes the representation. A physical device can be automatically configured based at least in part upon the metadata, and a database can be generated and structured based at least in part upon the metadata. The industrial system configuration platform further enables an industrial process to be implemented based at least in part upon the metadata.

Fig. 1