(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 January 2017 (05.01.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/002021 A2

(51) International Patent Classification: *G06F 17/30* (2006.01)

(21) International Application Number: PCT/IB2016/053864

(22) International Filing Date: 2 July 2016 (02.07.2016)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
3298/CHE/2015 29 June 2015 (29.06.2015) IN

(71) Applicant: **ABB SCHWEIZ AG** [CH/CH]; Brown Boveri Strasse 6, 5400 Baden (CH).

(72) Inventors: **JETLEY, Raoul**; Flat H-1706, Brigade Metropolis, Mahadevpura, Bangalore 560048 (IN). **VAIDYANATHAN, Aparajithan**; Flat L-706, Brigade Metropolis, Mahadevpura, Bangalore 560048 (IN). **GUGALIYA, Jinendra**; C-305, Surabhi Apartment, Bilekahalli, Bangalore 560076 (IN). **SUDARSAN, Sithu D**; Flat J-703, Brigade Metropolis, Mahadevpura, Bangalore 560048 (IN).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

— *with information concerning request for restoration of the right of priority in respect of one or more priority claims (Rules 26bis.3 and 48.2(b)(vii))*

(54) **Title**: A SYSTEM AND METHOD FOR MULTI-LANGUAGE LOOKUP AND TRANSLATION OF ENGINEERING ARTIFACTS

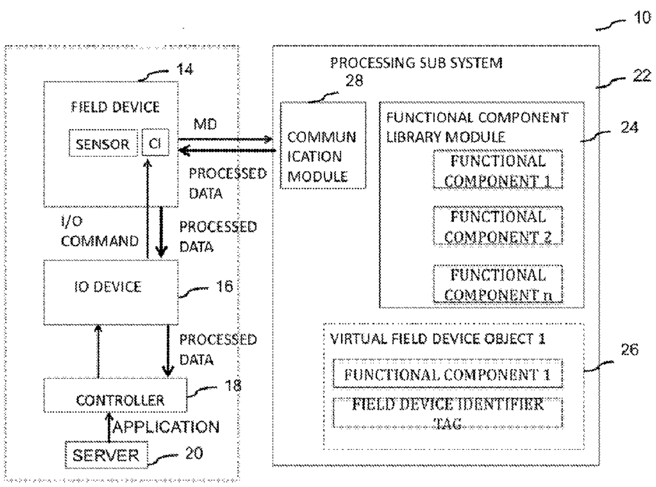


FIG. 1

(57) **Abstract**: A system and method for performing multi-language lookup and translation of a plurality of engineering artifacts have been disclosed. The lookup and translation is performed for retrieving results in multiple languages when the system receives a query from an end-user through a user interface. The received query is converted to a machine-understandable query and is searched across multiple indexed records in one or more languages. The indexed records are generated by identifying the narrative text in the engineering artifact, parsing the narrative text to generate terms in one or more languages, and add the record as an index corresponding to each of the engineering artifact. Further, the results are retrieved and arranged in decreasing order of the relevance and is presented to the end-user in the one or more languages. The displayed results include the original source of the document and a translated document, along with corresponding links.

WO 2017/002021 A2

A SYSTEM AND METHOD FOR MULTI-LANGUAGE LOOKUP AND TRANSLATION OF ENGINEERING ARTIFACTS

TECHNICAL FIELD

[0001] The present disclosure relates to a translation tool. More particularly, the present disclosure relates to a translation tool used for translating engineering artifacts across a plurality of databases in an engineering automation environment.

BACKGROUND

[0002] Typically, industrial automation deals primarily with automation of manufacturing, quality control and material handling process. Industrial automation eliminates or at least eases the decision making of humans and manual command-response activities with the use of mechanized equipment and logical programming commands.

[0003] The tools used in industrial projects support multiple languages customized as per the requirement of the project and the client. The language support is enabled particularly for three categories namely engineering data, engineering libraries, and human-computer interface. These three categories together form a base for multi-language support within the context of industrial automation engineering projects and are generally generated and consumed in different languages. The functionality of multi-language support within the engineering systems are dispersed within these three categories in a disconnected manner, which poses difficulties for end users while trying to retrieve relevant information.

[0004] This disconnected approach is cumbersome and restricts end users to efficiently create, locate, and configure engineering entities. Further, conventional systems fail to retrieve relevant results that are available in another language forcing the end user to switch to non-native language in order to leverage the convenience of using the features provided by the industrial tool.

[0005] In order to overcome the drawbacks associated with the existing industrial automation translation tools, there was felt a need for a holistic, unobtrusive, language-agnostic searching and translating tool. Further, there was felt a need for the translation tool for searching and translating information available in a plurality of databases and libraries containing text in multiple languages.

OBJECTS

[0006] An object of the present disclosure is to provide a system and method which ensures searching and translation of the information from a plurality of databases and libraries in multiple languages.

[0007] One more object of the present disclosure is to provide a decentralized, easy to use and robust architecture for searching and translating the texts and diagrams available in the plurality of engineering databases.

[0008] Another object of the present disclosure is to provide a system and method for creating engineering data and engineering libraries in a plurality of languages and provide the flexibility to consume the information in multiple languages for the end-user.

[0009] Yet another object of the present disclosure is to provide a basis for allowing the end-user to look-up and consume the information across different parts of the automation system in languages that are different from the original language used to create the information.

[0010] One more object of the present disclosure is to provide a system and method that minimizes/eliminates the complexities associated with the industrial automation translation tools for the end-user.

SUMMARY

[0011] The present disclosure envisages a system for providing multi-language lookup and translation of engineering artifacts in industrial automation projects. The system includes a database, an indexer, a user interface, a query generator, a search module, and a lookup and translation system. In accordance with the present disclosure, the database extracts a plurality of the engineering artifacts from a plurality of engineering systems using an application program interface provided by the engineering systems. The extracted engineering artifacts are indexed by the indexer by generating indexed records and sub-indexed records for each of the engineering artifacts. The indexed terms are stored as a reverse lookup table to increase the efficiency of the search. Further, the indexed records and sub-indexed records are generated in each of the supported languages for each of the engineering artifacts. The indexed records and the sub-indexed records are stored in the indexer along with a plurality of attributes associated with each of the indexed record and the sub-indexed record with predetermined language related constraints.

[0012] Typically, the user interface is configured to receive a plurality of queries from an end user in one or more supported languages. The received query is then transmitted to the query generator, which translates the query generated by the end-user to a machine-understandable query. The lookup and translation module translates the machine-understandable queries into a plurality of supported languages using a statistical language detection algorithm and a natural language parser. Further, the lookup and translation module provides lookup and translation for each of the query term originally generated by the end-user. The search module utilizes the results provided by the lookup and translation module and searches the indexed records in the plurality of the supported languages simultaneously. Further, the relevant results are aggregated, retrieved and arranged in the decreasing order of the relevance. Additionally, the lookup and translation module is also configured to translate the engineering artifact in the plurality of languages using standard language translation techniques. It is, therefore, possible to retrieve and translate the textual data available in the diagrams and present the translated textual data in the diagram to the end-user. Once the results are retrieved, the retrieved results, both in the default language and in one or more supported languages are displayed to the user along with the original links to the engineering artifacts.

[0013] The present disclosure envisages a method for providing multi-language lookup and translation of engineering artifacts in industrial automation projects. The method includes receiving a query regarding the engineering artifact from the end-user through the user interface. The end-user provides the query in one of the supported languages and selects an option to retrieve the results in one or more supported languages. Once the query is received from the end-user, the query is expanded to generate a machine-understandable query, and the machine understandable query is translated to a plurality of languages. The machine understandable query is searched against a plurality of indexes holding indexed and sub-indexed records in a plurality of languages as a reverse lookup table. The retrieved searches in a plurality of languages are aggregated and organized in the decreasing order of the relevance. The relevance is determined using a plurality of parameters such as term frequency, scoring of the term, and the like. The retrieved results are a plurality of languages and is displayed to the end-user through the user interface along with the relevant links to the original engineering artifacts available in the database. In accordance with the present disclosure, the diagrams containing textual data are translated to a scalable vector graphics (SVG) representation, and the translation is performed, and then the results are displayed in an SVG browser or as a standalone system.

BRIEF DESCRIPTION OF ACCOMPANYING DRAWINGS

[0014] These and the other features, aspects, and advantages envisaged by the present disclosure will become apparent when the following detailed description is analyzed in conjunction with the accompanying drawings in which like characters represent like parts throughout, and wherein:

[0015] FIG.1 is a block diagram illustrating the functional components of a multi-language lookup and translation system of engineering artifacts, in accordance with the present disclosure;

[0016] FIG.2 is a flowchart illustrating the steps involved in a method for generating index for a plurality of engineering artifacts, in accordance with the present disclosure; and

[0017] FIG.3 is a flowchart illustrating the steps involved in a method for a multi-language lookup and translation of engineering artifacts into a plurality of languages, in accordance with the present disclosure.

DETAILED DESCRIPTION

[0018] The present disclosure envisages a system that provides a multi-language lookup and translation of engineering artifacts into a plurality of languages.

[0019] In accordance with the present disclosure, the term ‘engineering data’ includes artifacts such as a signal list, an input-output list, a plurality of piping and instrument diagrams, a plurality of control narratives, cause-and-effect diagrams, and the like. Further, the term engineering data also signifies the artifacts created during the process of engineering that includes text in different languages. Examples of the artifacts include but are not limited to processed or semi-processed signal list with attributes, a plurality of system diagrams, descriptions of a plurality of devices, signal mapping configurations, a plurality of text documents, a plurality of control logic diagrams, and hardware configurations.

[0020] In accordance with the present disclosure, the term 'engineering libraries' refers to system-wide entities, including but not limited to a plurality of control logic templates, a plurality of function blocks, input-output mapping entities, a plurality of system topology components, macros, and the like.

[0021] The term 'human-computer interface' refers to a collection of a plurality of engineering tool interfaces that allow the end-user to create and consume the engineering data and the engineering libraries in a cohesive manner.

[0022] FIG.1 is a block diagram 100 illustrating the functional components of a multi-language lookup and translation system of engineering artifacts, in accordance with the present disclosure. The functional components include a user interface 102, a database 106, an indexer 108, a query generator 110, a search module 112, and a lookup and translator module 114. Typically, the user interface 102 forms a front-end system, and the other components form a backend system 104 for the multi-language lookup and translation system. The front-end system receives one or more queries from the end-user and displays relevant results, while the back-end system 104 processes the queries to perform multi-language lookup and translation of the plurality of the engineering artifacts.

[0023] In accordance with the present disclosure, the end-user provides the query to the user interface 102 in one or more languages. The user interface 102 receives the query and transmits the same to the query generator 108. The query generator 108 generates a machine-understandable query by parsing, tokenizing, searching, and matching the received query against a plurality of index records available in the database 104. The indexer 108 indexes each of the engineering artifacts available in the database 104 as an indexed record holding a plurality of attributes related to the artifact. Further, the search module 112 searches the indexer 110 with the machine-understandable query in one or more languages. The lookup and translation module 114 performs a multi-language lookup and translates the machine-understandable queries in one or more languages dynamically. The search module 112 further aggregates a plurality of relevant results retrieved in multiple languages, preferably in a decreasing order of relevance and transmits the retrieved results to the user interface 102.

[0024] Referring to FIG. 1, the user interface 102 is presented on a client device such as a laptop, desktop, smartphone and the like. The user interface 102 receives a plurality of queries from the end-user in one or more languages and transmit the same to the back-end system 104. The end-user has an option to select one or more languages in which the queries are submitted to the user interface 102. In a preferred implementation, the end-user selects one of the default languages and one or more supported languages of the multi-language lookup and translation system.

[0025] Typically, the user interface 102 receives the queries from the end-user in a plurality of structures. Examples of the query structures include, but are not limited to natural language words, phrases in one or more languages, a single character, multiple character wildcard phrases, fuzzy phrases for approximate matches, a plurality of Boolean operators, a plurality of logic operators, a plurality of proximity operators, and the like.

[0026] Once the response to the queries demanded by the end-user is retrieved from the backend system 104, the user interface 102 displays the relevant results in one or more languages. The one or more languages displayed on the user interface 102 include the default language of the system and the language selected by the end-user. The default language of the system 100 is set by the administrator of the system. In an alternative implementation, the default language of the system 100 is set by the end-user of the system.

[0027] The database 106 is a repository configured for storing the engineering artifacts extracted from a plurality of sources. The engineering artifacts are extracted from a plurality of third-party systems such as centralized engineering system repositories. The engineering artifacts are generally extracted using an application program interface (API) provided by the engineering system repositories. In accordance with the present disclosure, the database 104 further stores miscellaneous information (corresponding to the engineering artifacts) such as system metadata, mandatory files and folders, guidance for operating the multi-language lookup and translation system, and the like. The information stored in the database 106 is refreshed and updated periodically. Further, an end-user with administrative access to the database 106 has the ability to add, delete and modify the information stored in the database 106.

[0028] In accordance with the present disclosure, the engineering artifacts stored in the database 106 include distinct types of data including, but not limited to documents containing texts, a plurality of control logic encoded in textual languages, and diagrams and HMI components containing textual data. The textual data stored in the database 106 is extracted using the application program interfaces provided by the engineering systems regardless of the languages used therein.

[0029] The examples of the documents containing texts include, but are not limited a plurality of signal lists, control narratives, tag lists, requirements of the project, design documents, and the like. The documents containing both narrative texts and images are treated similar to the plain text documents, and a plurality of unparsed-able images are ignored by the database 106. Further, the examples of the control logic include but are not limited to standard tests, interface languages, and the like. The examples of the diagrams and the HMI components include but are not limited to piping and instrumentation diagrams (P&ID), user interface components, system topology diagrams, control logic encoded in free body diagrams (FBDs), sequential function charts (SFCs), abstract control diagrams, and the like.

[0030] The indexer 108 is configured to fetch the data stored in the database 106 and index each of the engineering artifacts as an indexed record. The artifact data is stored in the indexed record so as to facilitate efficient storage and a quick look-up. The indexer 108 maintains the index as a reverse lookup table containing a list of terms used in the engineering artifact and a corresponding frequency of a particular term occurring therein. In accordance with the present disclosure, each artifact in the index is represented as an indexed record, with constituent fields of the index record containing text corresponding to a plurality attributes of the engineering artifact. Thus, the index generated by the indexer 108 is a collection of indexed records, where each indexed record corresponds to the data and engineering details relating to the particular engineering artifact.

[0031] As discussed above, the engineering artifacts are generally of distinct types, and each of the data types is handled differently by the indexer 108. In accordance with the present disclosure, the text-based documents are indexed by parsing and tokenizing the text, with individual words in the text maintained as the indexed record. A plurality of additional attributes such as file name, document type, related project, and the like are maintained as metadata in the additional fields of the indexed record. In the preferred implementation, the text-based codes are parsed using a parser (not shown in figures), which is configured to identify a plurality of individual statements within the code blocks. Further, the parsed statements are tokenized to retrieve individual statements within the code blocks. Each term is stored along with a corresponding line number or statement, offset data, corresponding code block and application information, type information of the term (i.e., local variable, global variable, constant, reserved word, and the like). In an alternative implementation, the indexer 108 performs the parsing function without a specialized parser.

[0032] Further, the indexer 108 indexes the textual data in the diagrams by converting the textual data to a scalable vector graphics (SVG) based representation. The SVG representation is portable and allows the end-user to view (the SVG representation) in a browser or as a stand-alone client. SVG is a representation of the graphical information incorporating a plurality of elements, and positioning of the elements and the textual data, including associated positioning information of the textual data. The indexer 108 utilizes a plurality of APIs for converting the diagrams to a portable file representation. The portable file representation thus obtained is parsed to obtain the textual components within the diagram. Further, the obtained textual components are tokenized and stored in the index as an indexed record. The index maintains terms for the text in the diagrams along with position information and additional attributes such as the font, the typeface and the like for the text. The index also stores each token corresponding to a word (or phrase) in the text in the index, mapping the text to the original diagram, and the text component within the diagram and a location and position of the text within the diagram. In addition, the indexer 108 also maintains a link to the original diagram of the engineering artifact from the database 106.

[0033] In accordance with the present disclosure, based on the language used in the engineering artifact, the textual components are stored in one of multiple (sub) indexes corresponding to the language used in the engineering artifact. Each (sub) index corresponds to a particular language and includes terms specific to that language. The (sub) indexes are stored in the index as a sub-index record which is a subset of the indexed record relating to the particular engineering artifact. The indexed records and the sub-indexed records are searched when the query is received from the end-user and converted to the machine-understandable query through the search module 112.

[0034] In accordance with the present disclosure, the indexed records, and the sub-indexed records are stored as reverse lookup tables that map each index term to a source record, maintaining an offset within the record where the term occurs. The storage of terms as reverse lookup table enables the end-user to quickly search for specific terms, while preserving the original context of the record. Further, the indexer 108 also provides stemming to maintain a plurality of root words used in the indexed records. Maintaining the stemming of the root words improves accuracy and precision of the search. Thus, every record in the database 106 is tokenized and added to the index.

[0035] The query generator 110 is a module configured for translate (into a machine understandable format) the query received from the user interface 102 and generate a (machine understandable) query by parsing, tokenizing, searching, and matching the end-user generated a query against the indexed records and the sub-indexed records available in the indexer 108. Further, the query generator 110 translates the query generated by the end-user into a machine-understandable query in one or more languages in parallel. In a preferred implementation, the query generator 110 and the indexer 108 works in conjunction with each other for parsing, tokenizing, searching, and matching the machine-understandable queries in stored as the index record and the sub-index record for each of the engineering artifact. The query generator 110, at times, generates the machine-understandable query based on one or more pre-defined constraints such as a language constraint, depth of the query, and the like. The constraints are generally defined by the end-user, with or without the help of machine guidance.

[0036] In another implementation, the query generator 110 generates machine understandable queries from the queries generated by the end-user and further parses, tokenizes, and matches the (machine understandable) queries against the stored index records and sub-index records. In such a case, the indexer 108 generates only index records and sub-index records of the particular engineering artifact in one or more languages. The translation to the machine-understandable query is based on the conventional query generating techniques known in the state-of-art.

[0037] The system 100 for providing lookup and translation of engineering artifacts in the industrial automation environment further includes the search module 112. Typically, the search module 112 is configured for searching the indexed records and the sub-indexed records stored in the indexer 108. Further, the search module 112 searches the indexer 108 in one or more languages, in parallel, and in near real-time. The search module 112 evaluates the (user generated) query in each of the supported languages against the corresponding index. For example, in an *n*-word query, when *n*-tokens are generated with one token representing one word, multiple searches in multiple supported languages are executed in parallel.

[0038] Further, the search module 112 aggregates the results based on the (user-specified) query constraints such as mandatory or optional textual data. In accordance with the present disclosure, each document is listed only once. The list of the relevant documents matching the machine-understandable query is ranked based on pre-determined factors. One example of the predetermined factor is a term frequency, which ranks the documents that include the search terms occurring more often higher over the ones that have lesser number of occurrences. Another example of the predetermined factor is an inverse document frequency, which considers the number of documents containing a specific term, the fewer documents having a term the higher the rank for documents containing the particular search term. The other predetermined factors include, but are not limited to boosting at index time and search time, a combination of one or more predetermined factors, and the like.

[0039] The system 100 further includes a lookup and translation module 114 configured to perform lookup and translation of the textual data available in the engineering artifact to the one or more required languages. Typically, the lookup and translation module 114 utilize statistical algorithms for detecting the language used within the engineering artifacts. In accordance with the present disclosure, in the statistical detection methods, one or more databases for commonly used words in one or more languages are maintained. The individual words in the engineering artifacts are matched against the commonly used words in the database 106 when a narrative text from the particular engineering artifact is parsed. The matches are ranked in order of relevance of the results with respect to the machine understandable queries generated. Further, the match for the words are executed in one or more languages. A plurality of customized heuristics are used to determine the number of tokens to match and identify the language. Further, when multiple languages are used in the engineering artifact, the matched words are stored in multiple indexes as indexed records, corresponding to each language used in the record.

[0040] Once the source language for a record is identified, the record is indexed by the indexer 108 using a natural language parser (NLP) specific to the language. The NLP parses the input text to generate terms. In accordance with the present disclosure, the terms correspond to a plurality of words, a sequence of words, a plurality of grouped words, a plurality of phrases in the record, and the like.

[0041] In accordance with the present disclosure, the lookup and translation module 114 further configure the NLP to provide additional pre-processing for the indexed record. The examples of the pre-processing of the indexed record include, but are not limited to removal of stop words (like the articles and commonly used words), augmentation of the acronyms (for example, using 'BFP' and 'boiler feed pump' interchangeably), commonly used phrases (for example, 'the big apple' is augmented to match 'New York City'), and the like. In addition, the lookup and translation module 114 also configure the NLP to create a synonym list that helps the end-user to look-up words that represent the same object or has a same/similar meaning within the context of the automation engineering system. For example, an abbreviation such as CLD is synonymized as 'control logic diagram'.

[0042] Referring to FIG. 2, there is shown a flowchart 200 illustrating the steps involved in a method for generating an index of a plurality of engineering artifacts. The flowchart 200 in accordance with the present disclosure includes a plurality of steps for generating indexed and sub-indexed records from the data fetched from the database containing the information of the plurality of engineering artifacts. The indexes are generated for searching the relevant results for the queries generated by the end-user in one or more languages.

[0043] The index generation is a one-time activity, which is carried out to list the terms available in the engineering artifact and associate them with the corresponding documents, in one or more languages. The process of index generation is initiated by the indexer (Step 202).

[0044] In accordance with the present disclosure, to initiate the process of index generation, the data available in the plurality of the engineering artifacts has to be extracted and read (Step 204). The information regarding the plurality of the engineering artifacts is extracted from a database such as an engineering system database 206. In the preferred implementation, the information is extracted from a single engineering system database. In an alternative implementation, the information is extracted from multiple engineering system databases. The extracted information from the engineering system 206 includes the records from multiple languages. Further, the extracted information includes a plurality of formats, including but not limited to textual data, control flow diagrams, images, and the like.

[0045] Once the information from the engineering artifacts is extracted, the narrative text language in each of the records is detected (Step 208). The language in each of the records is detected based on a plurality of techniques, including but not limited to statistical language algorithm, natural language parsing and processing and the like. In accordance with the present disclosure, when the statistical detection methods are used, a database of commonly used words database 210 in a plurality of languages is maintained. The matched words are ranked in order of the relevance. During the language detection and matching process using the commonly used words database 210, when the machine-generated queries are matched for one or more languages, a primary language, and a secondary language are identified and scored accordingly. A plurality of customized heuristics are used to determine the number of tokens required to match the natural language (used in the narrative text). When multiple languages are used in the extracted engineering artifact, the indexes are stored as multiple indexes, corresponding to each language used in the engineering artifact.

[0046] Further, after the detection of the language of the narrative text in the engineering artifacts, the narrative text is pre-processed using the natural language preprocessor (Step 212).The natural language preprocessor utilizes one or more standard language pre-processing techniques for preprocessing the identified narrative text. The pre-processing of the narrative text include removal of stop words such as articles and commonly used words, augmentation of acronyms, matching of commonly used phrases, and the like. Further, the pre-processing includes generating a synonym list that enables the user to look-up words that represent the same object or have the same meaning within the context of the automation engineering system 206.

[0047] Once the source language of the narrative text of the engineering artifact has been identified and pre-processed, the engineering artifact is indexed using a natural language parser (NLP) specific to that language to generate terms (Step 214). The terms correspond to words, a sequence of words, phrases in the record, and the like. In accordance with the present disclosure, the terms are stored as in index (as a reverse lookup table), which maps each index term in the source record, and maintaining the offset within the record where the term occurs. Additionally, during the parsing of the narrative languages of the engineering artifact, an option is provided to use word-stemming. The use of word-stemming enables the indexer to maintain the root words used in the record, which improves the accuracy of the search.

[0048] Further, the generated terms are stored as in indexed record corresponding to each of the engineering artifacts for the identified languages (Step 216). At the end of the index generation phase, a plurality of indexes are generated, and the lookup and translation system supports one index corresponding to one language. The indexes are then used for searching and information retrieval.

[0049] In accordance with the present disclosure, after the generation of the index for each of the engineering artifact, the system checks for more engineering artifacts to be indexed (Step 218). The index generation process is stopped (temporarily) when there are no more engineering artifacts to be indexed (Step 220). The index generation process is initiated again (Step 204) when there are a number of other engineering artifacts to be indexed.

[0050] In accordance with the present disclosure, the generated indexes are stored and maintained offline, and hence are to be periodically refreshed to synchronize with the data in a plurality of automation engineering repositories. The period for refreshing the index is preferably configured by the end-user, according to the needs of the project. Refreshing the index changes specific record or field that is updated and the refresh is processed in an incremental manner.

[0051] Referring to FIG. 3, there is shown a flowchart 300 illustrating the steps involved in a method for a multi-language lookup and translation of engineering artifacts. The process of lookup and multi-language translation of the plurality of engineering artifacts is initiated (step 302). The information retrieved by the lookup and the multi-language translation assists the end-user to gather unknown knowledge about a system and perform detailed analysis including, but not limited to sentiment analysis, predictive analytics to gain deeper insights into the data available in the plurality of engineering artifacts.

[0052] Further, the query from the end-user is received through the user interface in one of the supported languages (Step 304). The received query provided in the user interface includes distinct structures, including but not limited to natural language words or phrases, single words, multiple words, wildcard searches, Boolean operators, a combination of logic operators, proximity searches, and a combination thereof. Typically, the end-user is presented with a list of languages to submit the query to the user interface. The default language of the system 100 is selected, when the end-user does not select any of the supported languages.

[0053] The received query is then expanded to generate a machine-understandable query from the user-generated a query (Step 306). In accordance with the present invention, the query generator generates the machine-understandable query from the user-generated a query. A plurality of the standard techniques available in the state-of-art is utilized for expanding the user-generated a query.

[0054] Further, the expanded query is translated into a plurality of languages (Step 308). Typically, the machine-understandable queries are translated to all the supported languages used in the indexed records using the lookup and translation module. Further, a set of language specific dictionaries are used to map individual terms and phrases between the languages. In accordance with the present disclosure, a dictionary lookup is used to convert the original text available in the source language to the destination language which is stored as an index. In order to convert text from one language to another, the dictionaries are configured to provide a mapping between the words and phrases in each language. The mapping is effectively provided by using a plurality of open source dictionaries, third-party vendors, and the like. Additionally, the mappings are optionally maintained in a separate index in a traditional database.

[0055] Once the machine-understandable queries are translated to the plurality of supported languages, the search module performs different searches in parallel, by evaluating the machine-understandable query in each language against the corresponding index (Step 310). The searches are performed based on the pre-determined constraints imposed by the lookup and translation system, and by the end-user.

[0056] Further, the results of the search are displayed in the user interface in the decreasing order of the relevance (Step 312). The displayed results also include a relevance score and the identity of the document. Typically, the links are also provided to the original record and the textual data translated to the requested language, when the user generated query language is different from the original language in the retrieved document. In accordance with the present disclosure, when the matching result is found in a figure or a diagram, the link to the original engineering artifact is provided to the end-user via the user interface. In addition, the translated version the diagram is also displayed. In the translated version, the text stored in the SVG file corresponding to the diagram is translated into the target language. The resulting SVG file is then rendered using an SVG browser (or a renderer). Further, the link to the translated SVG file is also provided to the end-user via the user interface. The process is temporarily stopped when a lookup and translation is completed for a query generated by the end-user (Step 314).

[0057] The present disclosure envisages a system and method for performing lookup and translation of engineering artifacts in an industrial automation system. In comparison to a traditional lookup and translation system, the system envisaged by the present disclosure provides a holistic, unobtrusive, language-agnostic searching mechanism to look for information from a plurality of repositories and libraries containing text in multiple languages. The system envisaged provides end-user in the automation engineering system with the ability to lookup text contained in the plurality of engineering artifacts, regardless of the storage location and the format. Further, the system proposed in the present disclosure eliminates the need for conducting searches in multiple languages for the same/similar content. Further, the proposed system envisages the end-user to create data and libraries in the language of choice and provide flexibility to consume the engineering artifacts in multiple languages. The search technique included in the system provides the basis for lookup and consumption of various parts of the automation system in languages, which are different from the original language used to create the information.

[0058] Furthermore, the system envisages provides an advantage of storing engineering artifacts in the index, by making the system highly scalable and ensures fast search and retrieval of the records even when the index size grows sublinearly and reaches a near plateau for large data sets. Further, since only the index is needed for the search, less memory is required as opposed to a traditional structured query language (SQL) based search and retrieval. Additionally, storing multiple indexes corresponding to individual language ensures that the searches are performed in parallel, further improving the performance of the system. Further, storing the text in native language indexes also ensures that the context and semantics of the text in the original language is preserved, which may otherwise be lost in the translation.

METHOD AND SYSTEM TO INCREASE PROCESSING CAPABILITY OF FIELD DEVICES IN AN INDUSTRIAL CONTROL SYSTEM

FIELD OF THE INVENTION AND USE OF INVENTION

[0001] The invention generally relates to the field of control systems and more specifically to field devices, and provides a method and system to increase the processing capability of the field devices deployed in an industrial plant.

PRIOR ART AND PROBLEM TO BE SOLVED

[0002] Industrial Control System is typically used in process industries such as refinery, oil and gas, paper and pulp, manufacturing and the like. It is a specially designed control system to control industrial process in industrial plants including complex, large and geographically industrial processes. Typically, in the Control System, field devices such as sensors and actuators measure process parameters like temperature and pressure, and controls the valve by changing its position, are connected to input and output devices, which in turn communicate with controller modules through a communication bus for data monitoring, data logging, alarming and controlling purpose.

[0003] The communication protocols used in the Control System are of different types such as foundation field bus, HART® (highway Addressable Remote Transducer Protocol), PROFIBUS® (Process fieldbus), Modbus (serial communication protocol) and use a native network of the Control System.

[0004] The hardware resources for the field devices broadly include, communication, processing/analysis/control, and sensing/actuating hardware. Communication related hardware handles the protocol specific communication. Processing/analysis/control related hardware handles the raw sensing data processing tasks. And, the sensor related hardware performs the signal sensing task.

[0005] Present day field devices are built with more intelligence for data processing and analysis by addition of hardware resources that perform sensor data processing, analysis and enable the protocol specific communication. However, while the capabilities of the field devices have been enhanced, addition of these hardware resources has also increased manufacturing cost, as well as life-cycle maintenance cost of these field devices.

[0006] Further, to handle new complex tasks, the existing field devices need to be upgraded often. Still further, if such a field device is damaged, the replacement of damaged device with new device has to be done... Replacement involves various steps to resume functionality of old field device by new field device. Such shortcomings, in turn lead to additional operational costs of the plants and control and monitoring operations.

OBJECTS OF THE INVENTION

[0007] There is a need to find an alternate technical solution that substitutes the complexity of adding of processing hardware on the field devices.

[0008] It is an object of the invention to address the above need by providing a method and system for increasing the processing capability of the field devices without adding additional hardware resources. This is achieved by using a processing sub-system hosted on a network transparent to the Control System. This allows the processing and analysis part of field data to be performed on a separate entity. For example, the separate entity can reside on a remote service server, cloud, or web server. This not only reduces the hardware burden on the field devices, it further allows for sharing of processing and analysis functionalities among the field devices.

SUMMARY OF THE INVENTION

[0009] In one aspect, an industrial control system for control and monitoring of one or more process conditions in an industrial plant is provided. The industrial control system includes one or more field devices, one or more input-output devices, at least one controller, and at least one server. A processing sub-system is provided for scaling processing capability of a field device in the industrial control system, wherein the processing sub-system includes a functional component library module having a plurality of functional components. A virtual field device object is configured using one or more functional components from the plurality of functional components, and a field device identifier tag of a field device in the industrial control system. The processing sub-system includes a communication module for receiving measurement data from the field device. The functional components are configured for processing the measurement data to generate processed field device data, and the communication module is configured for communicating the processed field device data.

The control system includes one or more pre-configured communication interfaces to communicate the processed field device data to the industrial control system.

[0010] In another aspect, a method for scaling processing capability of a field device in an industrial control system using a processing sub-system is disclosed. The industrial control system includes one or more field devices, one or more input-output devices, at least one controller, and at least one server. The method includes a step for configuring a virtual field device object in the processing sub-system using one or more functional components from a plurality of functional components, and a field device identifier tag of a field device in the industrial control system. The method further includes receiving measurement data from the field device in the processing sub-system; and processing the measurement data in the processing sub-system by the virtual field device object corresponding to the field device, to generate processed field device data. The method then includes a step for communicating the processed field device data using one or more pre-configured communication interfaces to the industrial control system.

DRAWINGS

[0011] These and other features, aspects, and advantages of the present invention will become better understood when the following detailed description is read with reference to the accompanying drawings in which like reference numerals represent corresponding parts throughout the drawings, wherein:

[0012] FIG. 1 is a diagrammatic representation of an exemplary control system for control and monitoring of one or more process conditions;

[0013] FIG. 2 is a diagrammatic representation of one embodiment of system of FIG. 1 with a coupler to receive measurement data from the field devices and communicate the measurement data to distinct virtual field device objects in the processing sub-system;

[0014] FIG. 3 is a diagrammatic representation of another embodiment of system of FIG. 1 with a coupler to receive measurement data from the field devices and communicate the measurement data to one virtual field device object in the processing sub-system;

[0015] FIG. 4 is a diagrammatic representation of an exemplary implementation showing the communication interface between the field device and the processing sub-system;

[0016] FIG. 5 is a diagrammatic representation of another exemplary implementation showing the communication interface between the field device and the processing sub-system;

[0017] FIG. 6 is a diagrammatic representation of a prior art intelligent field device with the different data layers;

[0018] FIG. 7 is a diagrammatic representation of field device data layer with using the processing sub-system of the invention;

[0019] FIG. 8 is a block for a prior art field device with the hardware modules for advance processing and analytics;

[0020] FIG. 9 is a block diagram for an exemplary field device that is used with the processing sub-system, according to an embodiment of the invention; and

[0021] FIG. 10 is a flowchart showing exemplary steps for the method for scaling processing capability of a field device according to an aspect of the invention.

DETAILED DESCRIPTION OF THE INVENTION

[0022] As used herein and in the claims, the singular forms "a," "an," and "the" include the plural reference unless the context clearly indicates otherwise.

[0023] FIG. 1 is a diagrammatic representation of an exemplary industrial control system 10 for control and monitoring of a process, according to one aspect of the invention. The industrial control system 10 includes field devices 14, input-output (IO) devices 16, at least one controller 18, and at least one server 20. The control system 10 further includes a server or processing sub-system 22 for scaling processing capability of a field device 14 in the industrial control system 10. The field devices referred herein are physical on- field hardware devices which are located on in the actual process plant. The processing sub – system advantageously allows for creating off-field or virtual field devices that use measurement data from on-field devices and do desired processing separately thus removing the processing burden and also rigid electronic configurations otherwise needed from the on-field devices. The processing sub-system is explained in more detail below.

[0024] The processing sub-system 22 includes a functional component library module 24 having different functional components, represented as Functional Component 1, 2...n in FIG. 1. Each functional component represents rules and logic with respect to the functionality of the functional component. The functional component library module may be in-built in the processing sub-system or may be provided externally on a separate device that is accessible by the processing sub-system. For example, one functional component would contain the logic needed for sensor linearization.

[0025] In the processing sub-system, a virtual field device object 26 is configured using required functional components (Functional Component 1 in FIG. 1), and a field device identifier tag of a field device 14 in the industrial control system 10. This ensures process control integrity, as the field device measurement data is always identifiable, and further processing includes the identity of the field device and therefore the identity of the particular process condition being controller and monitored. As mentioned herein above, the functional components in the virtual field device object, process the measurement data as per the functionality, rules, and logic embedded in the functional component, to generate processed field device data. Thus when a deployed field device does not have a particular processing capability in-built through the hardware resources on the field device, the processing can be accomplished using the virtual field device object. This enhances the capability of existing device without having to upgrade the hardware of the existing device.

[0026] The processing sub-system 22 includes a communication module 28for receiving measurement data (shown as MD in FIG. 1) from the field device. The communication module may be integrated with the virtual field device object, or may be provided as a separate module or may be integrated with the processing sub-system. The communication module 28 is also used for communicating the processed field device data after the processing via the virtual field device object.

[0027] In one example, the field device 14 is equipped with a communication interface represented as “CI”, it could be for example, a wireless interface, for example a low power wifi, Bluetooth interface etc. In one exemplary implementation “CI” transfers measurement data directly to the virtual field device object 26 with an identity (i.e. a unique identity ‘ID’) for the field device via the communication module 28. In this case, the measurement data that is initially received as an analog signal by the field device is converted to digital data in the field device itself.

[0028] In one exemplary embodiment, the communication module is provided as a separate module, it may be provided as a coupler 30 as shown in FIG. 2, the coupler 30 is configured as a hardware device or a software module to receive measurement data from one or more field devices (shown as Field Device 1, 2, 3 in FIG. 2) in separate communication channels, and transmit the measurement data (shown as MD1, MD2, MD3) by tagging it with the field device identity. In one exemplary implementation tagging is done using a unique field device identifier tag.

[0029] In an alternate embodiment, as shown in FIG. 3, the coupler 30 can be configured as a data aggregator device, and multiple field devices are connected to the data aggregator (DA) device. The DA device converts the field device analog output to digital data using ADC (analog to digital converter). Then, it tags the data with their field device unique ID. Further, it aggregates all the sensor data and sends it to the processing sub-system (virtual field device object, indicated by reference numeral 26*) via wired or wireless medium.

[0030] It would be appreciated by those skilled in the art that the aggregator device of FIG. 3 can communicate the field device data from different field devices in several ways. For example, the aggregator device can communicate field device data from each of the field device with their respective unique IDs. In another example, the aggregator includes an intermediate processor to perform specific processing and send the resultant of the processing to the processing sub-system. For e.g. one of the field device data may be sending measurements related to “volume” of fluid, and the other field device data may be sending the measurements related to “velocity” of the fluid, and these measurements may be time synchronized so that they relate to same measurement instance. The aggregator may include an intermediate processor to calculate the “flow rate” based on the “volume” and velocity” measurements, and communicate the flow rate measurement data to the processing sub-system along with the unique IDs of both the field devices. There may be other combinations to aggregate the field device data, based on the real world requirement. However, it may be noted here, that even when the aggregator does not have any intermediate processor, a desired functional component can be created in the processing subsystem itself.

[0031] It would be understood by those skilled in the art that the control system includes one or more pre-configured communication interfaces to communicate the processed field device data to the industrial control system. These pre-configured communication interfaces link a native network of the industrial control system with a second network of the processing sub-system, wherein the second network is different from the native network. The pre-configured communication interfaces are provided via the communication module and communicate the processed field device data to at least one of the field device, an input-output device coupled to the field device, the at least one controller and the server

[0032] The FIG. 4 shows the one of the deploying option where field device has two interfaces. One interface for communicating with the processing sub-system and another interface for communicating with typical IO (Input-Output) device. Interface 1 as shown in FIG. 4 is used for traditional IO communication for communicating sensing and processing task requests. Task requests would be understood by those skilled in the art as instructions from IO devices to receive specific information or to provide specific information. The IO interface to enable communication can implement known DCS protocols such as HART® protocol, PROFIBUS®, or any other DCS native network communication protocol. Interface 2 is used for enhancing field device capability by enabling the physical field device to connect to the processing sub-system. The connection between Interface 2 and processing sub-system can be wired or wireless. As explained herein above, based on the processing task request, the field device processing power can be enhanced by adding functional components in the virtual field device object. For upper layer entities such as IO device, controller and server, receiving the processed field device data will be seamless via the field device.

[0033] Another scenario of deployment is shown in FIG 5, where the field device has only one interface, interface 3 to communicate the measurement data to processing sub-system. The interface 3 could be wired or wireless. The processing sub-system accesses measurement data from the field device via interface 3 and performs the complex processing required by the application as provided by the customer to generate processed field data. Communication of processed field data from the processing sub-system, is enabled as follows in this exemplary embodiment. A coupler is used to receive the processed field data via interface 4 and converts the processed field device data received from the processing sub-system into protocol specific format like HART®, PROFIBUS® etc., and communicates this processed field device data via interface 5 to IO device. One coupler can be used to represent multiple field devices using IO channels as mentioned before. Each of the IO channels of the coupler can be configured for HART®/ PROFIBUS®/Foundation Fieldbus® etc. protocols. For any of the upper layer entities such as IO devices, controller and server the communicating with the coupler is same as communicating with the field device.

[0034] FIG. 6 and FIG. 7 illustrate the impact of external processing sub-system on the field device data protocols. FIG. 6 illustrates a prior art intelligent field device with the different data layers 40 for different processing tasks. FIG. 7 shows that through the present invention, the field device data 42 is considerably reduced as the processing layers are shifted to the processing sub-system, and are no longer required to be present on the field device. Similarly, FIG. 8 illustrates a block diagram 44for a prior art field device with all the hardware modules for advance processing and analytics, and FIG. 9 illustrates a block diagram 46for an exemplary field device that is used with the processing sub-system, according to an embodiment of the invention.

[0035] Some exemplary use cases are described herein below that further illustrate the different embodiments of the invention:

[0036] Case-1: Full fledge field device with IO and Virtual field device Object (VF) connectivity:

Full fledge field device refers to a field device which can be operated alone. The prior art field devices can be considered as full fledge device. These have sensing, processing and communication capabilities built into the field device. Adding the processing sub-system of the invention to full fledge device enhances the device capabilities, while ensuring basic processing within the field device itself. Field device can have inbuilt VF interface or it can use external VF adaptor/coupler component to connect with VF. In this scenario, field device process basic requests from IO device by its own and it forwards complex processing requests to VF.

[0037] Case 2: Dumb field device with IO and VF connectivity:

Dumb devices are those field devices which do not have processing capability built into the field device. For example, 4-20mA temperature device, which only measures the temperature and generates the output in the form of 4-20mA current. It does not have capability to say whether measured value in Centigrade or Fahrenheit. The complete processing activity is carried out in the processing sub-system of the invention. All the requests (or commands) which come to dumb field device from IO devices or any higher entity like controllers, are forwarded to the processing sub-system for processing and response from processing sub-system are received by the dumb field device and sent back to IO devices or controllers.

[0038] Case-3: Dumb field device only VF connectivity:

In this scenario, the dumb field device has only interface with the processing sub-system. Dumb field device sends measurement data to the processing sub-system, and rest of the processing logics are executed in the processing sub-system. IO device, controller or server can access the processed field device data directly from processing sub-system. This scenario is suited for remote monitoring kind of application.

[0039] Case 4: Field device and Processing sub-system communication:

This example shows the traditional communication between Controller/Server via an Application (coded instructions for particular process tasks), IO device, field device and processing sub-system. Assume that the Application want to read minimum temperature value from last six month. For this, the Application sends a request to IO device and IO device sends the request to a field device that is monitoring temperature. This particular field device's memory is not enough to store six months data so it is incapable to process this request, and the field device therefore forwards this request to the processing sub-system.

The field device, in this implementation, is in communication with the processing sub-system through a communication interface. The virtual field device object in the processing sub-system is reading and storing (the functional component in this case is "Read and Store") field device measurement data periodically and it has field device historian data. The processing sub-system receives the field device request to report minimum temperature from last six months the application request and return the response to the field device. After receiving the response from VF, the field device responds back to IO and then to Application at controller or server level. In this scenario, the interaction between physical field device and VF in the processing sub-system is completely isolated from the Application. It would be understood by those skilled in the art that the communication between VF and field device can be asynchronous or synchronous communication.

[0040] Case 5: Controller/Server direct communication with the Processing sub-system:

In another exemplary implement, since the processing sub-system has all field device measurement data, the Application at Controller/Server can directly interact with the processing sub-system to get the processed field device data. There is no need of IO device and associated hierarchical devices to read field device data. The processing sub-system in this case asynchronously reads the field device measurement data and performs the processing in background. When the processing sub-system gets the request from Application, it responds to the Application with the processed field device data that is sent directly to controller/server. A virtual field bus may be used in an exemplary embodiment for connecting and sharing information between field device objects, and external controller/server.

[0041] The processing sub-system as described herein is implemented in an exemplary embodiment as a software application residing to the external device, such as a cloud, remote or web server, and is connected with field device via wireless or wired medium. The processing sub-system continuously accesses the field device measurement data based using the virtual field device object. Based on the demand set by the application, processing sub-system performs complex processing, analysis and control task and returns back the result as processed field device data to the same field device, or elsewhere as per the communication interface and communication protocol as explained in several use case herein above. The execution of processing tasks separately and external to the field device as enabled by the invention, gives flexibility to execute any kind of complex processing since virtually unlimited resources are available.

[0042] The technique mentioned herein allows customized processing of field data by enabling or disabling or selecting functional components on demand, re-use of functional components, thus adding processing capabilities and flexibility. On top of basic libraries set for the functional components, new libraries can be added for providing new functional components in the processing sub-system, for example for dynamic process trend diagnostics and such other analytics demand.

[0043] Another aspect the invention described herein, is a method for scaling processing capability of a field device in an industrial control system. The method is illustrated in flowchart 50 of FIG. 10. This is achieved by using the processing sub-system external to the field device hardware, as explained in reference to the FIG. 1-5 above. The method includes a step 52 for configuring a virtual field device object in the processing sub-system using necessary functional components, and a field device identifier tag of a particular field device in the industrial control system. The functional components are provided through a library of pre-configured functional components in the processing sub-system, or may be provided in dynamic manner, on-the-fly based i.e. need based on the requirement received from an operator station. The ability to select functional components external to the field device, allows great flexibility and scaling of processing capability of any given field device.

[0044] For example, functional components of ‘pressure linearization’, and ‘historian’, may be selected in one embodiment. In another embodiment a functional component of ‘asset management’ may be added. In yet another embodiment, a functional component “Alarm and Event” may be added. Other functional components include but are not limited to “Security”, “Calibration”, “PROFIBUS® Communication”, “Foundational Fieldbus Communication” etc. Each functional component has rules and logic to obtain the processed output, which is referred herein as processed field device data relating to that functionality.

[0045] For generating processed field device data, the method includes a step 54 for receiving measurement data from the field device in the processing sub-system, processing the measurement data in the processing sub-system as shown at step 56 by the virtual field device object corresponding to the field device. The method then includes a step 58 for communicating the processed field device data using one or more pre-configured communication interfaces to the industrial control system for use in control and monitoring function.

[0046] It would be appreciated by those skilled in the art, that not only the processing capability of field device is easily scalable using the method described herein, different communication interfaces can be provided to communicate the processed field device data to suit the plant and customer requirement.

[0047] Based on the communication interface, a suitable communication protocol is selected to communicate the processed field device data. The format for processed field device data is flexible as explained earlier. It can be converted into any protocol application layer data format such as, HART®, PROFIBUS® or Profinet packet suited to consumer.

[0048] The different configurations for communication interfaces implemented by the method of the invention, include a communication interface for communicating the processed field device data to the same field device from which the measurement data was received. In another embodiment, the communication interface is provided to communicate the processed field device data to an input-output device coupled to the field device from which the measurement data was received. In another embodiment, the communication interface is provided to communicate the processed field device data to the at least one controller. In another embodiment, the communication interface is provided to communicate the processed field device data to the server. These configurations have been explained in relation to the control system of the invention.

[0049] These pre-configured communication interfaces are configured to receive at least one of request commands, input-output commands, and control commands from at least one the field device, an input-output device coupled to the field device, the at least one controller, the server, and the operator station. The one or more pre-configured communication interfaces use at least one of wired or wireless communication means.

[0050] The consumers of processed field device data include controller, IO device, asset monitoring application and other customer requirement based applications operated at the controller or server level. Since the processed field device data now resides on an external network such as the cloud network (public/private), the processed field device data can be access processed directly from the cloud by providing the necessary communication configuration.

[0051] The proposed solution provides great level of flexibility, to offer customer specific products. With same dump physical device, various customized virtual field device objects can be offered to customer. Based on application demand, customer can avail the enhanced features of device.

[0052] Since the processing/analysis specific hardware resource is now de-coupled from the field device, the manufacturing cost decreases significantly. In addition, for the future product the existing manufacturing line can be reused.

[0053] Enabling virtual field device object using the processing sub-system provides unlimited processing capabilities to the physical field device. Now the processing sub-system is capable for performing data analytics, data processing, and complex control activity specific to customer needs, and on demand, device capability can be enhanced. Depending on the need, compute/process and amount of data processing could vary dynamically and transparently. Data can be stored for longer period and made available even if there is any issue with the DCS system. Unlike remote service, this solution provides two way communication with DCS.

[0054] Since the processing power any field device can be increased/decreased dynamically on the fly it gives opportunity to device vendor to offer various kind of service options to the customer. Further, without changing/upgrading of physical device new capabilities to device can be added. The processing sub-system can be customized based on application requirements provided by the customer and offered as service subscription to the customer.

[0055] The described embodiments may be implemented using standard programming and engineering techniques related to software, firmware, hardware, or any combination thereof. The described operations may be implemented as code maintained in a non-transitory "computer readable medium", where a processor may read and execute the code from the computer readable medium. The code implementing the described operations may further be implemented in hardware logic (e.g. an integrated circuit chip, Programmable Gate Array (PGA), Application Specific Integrated Circuit (ASIC), etc.). Still further, the code implementing the described operations may be implemented in "transmission signals" transmission signal may be decoded and stored in non transitory hardware or a computer readable medium, where transmission signals may propagate through space or through a transmission media, such as an optical fiber, copper wire, etc. An "article of manufacture" comprises non-transitory computer readable medium, hardware logic, or transmission signals in which code may be implemented. A device or server in which the code implementing the described embodiments of operations is encoded may comprise a non-transitory computer readable medium or hardware logic. Of course, those skilled in the art will recognize that many modifications may be made to this configuration without departing from the scope of the present invention, and that the article of manufacture may comprise suitable information bearing medium known in the art.

[0056] The processing sub-system, modules and devices referred herein may use a non-transitory data storage unit or data storage device. A computer network may be used for allowing interaction between two or more electronic devices or modules, and includes any form of inter/intra enterprise environment such as the world wide web, Local Area Network (LAN), Wide Area Network (WAN), Storage Area Network (SAN) or any form of Intranet or other automation and industrial communication environment relevant to the industrial plant.

[0057] While only certain features of the invention have been illustrated and described herein, many modifications and changes will occur to those skilled in the art. It is, therefore, to be understood that the appended claims are intended to cover all such modifications and changes as fall within the true spirit of the invention.

CLAIMS:

I/We claim,

1. A system (100) for multi-language lookup and translation of a plurality of engineering artifacts in industrial automation projects, said system comprising:

a database (106) configured to store a plurality of engineering artifacts extracted from a plurality of engineering system repositories, wherein the plurality of engineering artifacts are extracted using an application program interface provided by the plurality of engineering system repositories;

an indexer (108) configured to index each of the engineering artifacts as an indexed record in a plurality of supported languages as a reverse lookup table, wherein the indexer fetches the engineering artifact from the database (106), and wherein the indexed record of the engineering artifacts includes a list of terms used in each of the engineering artifacts and a plurality of attributes relating to the engineering artifact;

a user interface (102) configured to receive a plurality of queries generated by an end-user in the plurality of supported languages, transmit the received queries to the indexer (108), and display a plurality of relevant results in the plurality of supported languages;

a query generator (110) configured to generate machine-understandable queries by parsing, tokenizing, searching and matching the queries generated by the end-user, and translate the generated queries into the plurality of supported languages in parallel;

a search module (112) configured to search the indexer (108) with the machine-understandable queries generated by the query generator (110) against the indexed records, and aggregating a plurality of retrieved results in a decreasing order of relevance, wherein the search module (112) performs search based on a plurality of pre-determined constraints; and

a lookup and translation module (114) configured to perform lookup and translation of a textual data available in the plurality of the engineering artifacts in a plurality of the supported languages, wherein the lookup and translation module utilizes statistical methods and natural language parsing and processing for detecting the language used within the engineering artifacts.

2. The system as claimed in claim 1, wherein the indexer (108) is further configured to generate a sub-indexed record for the textual data with each of the indexed record corresponding to the language used.

3. The system as claimed in claim 1, wherein the indexer (108) indexes the textual data by parsing and tokenizing the textual data in the engineering artifact with a plurality of individual words and adding attributes to the recorded index, wherein the attributes relating to the engineering artifacts include a file name, a document type, a related project, type information of the term.

4. The system as claimed in claim 1, wherein the indexer (108) indexes the textual data in diagrams present in the engineering artifacts as an indexed diagram by converting to a scalable vector graphics (SVG) based representation, wherein the indexed diagram includes textual data, position of textual data, and graphical information of the diagrams.

5. The system as claimed in claim 1, wherein the reverse lookup table in the indexer includes a list of terms used in the artifacts and the corresponding frequency of the term occurring therein.

6. The system as claimed in claim 1, wherein the lookup and translation module (114) utilizes a plurality of language-specific dictionaries for mapping individual and phrases between the supported languages used in the plurality of engineering artifacts.

7. The system as claimed in claim 1, wherein the engineering artifacts are selected from a group consisting of documents containing text, control logic encoded in textual languages, diagrams and Human Machine Interface (HMI) components containing textual data.

8. The system as claimed in claim 1, wherein the search module (112) retrieves a plurality of relevant links to the corresponding the engineering artifact and the translated textual data via the user interface (102) when the query language is different from the default language.

9. A method for multi-language lookup and translation of a plurality of engineering artifacts in industrial automation projects, said method comprising:

receiving a query from an end-user through the user interface, wherein the end-user inputs the query in a plurality of languages (304), and wherein the query is in a form a plurality of structures;

expanding the query received by the end-user through a query generator (306), wherein expanding query includes generating a machine-understandable query by detecting the language of the received query, and parsing and tokenizing the query in the plurality of the languages;

translating the expanded query into the plurality of languages (306) using the lookup and translation module for generating a plurality of indexed records, wherein the translation is executed by parsing the detected text to generate a plurality of terms in the plurality of languages, and wherein the translation is executed in parallel in the plurality of the languages;

searching the databases in the plurality of languages against the indexed records in the plurality of languages in parallel (308), and aggregating a plurality of results in the decreasing order of the relevance, and wherein the searching is performed based on a plurality of pre-determined constraints; and

displaying a plurality of relevant results to the end-user in the plurality of languages in the user interface, wherein the relevant results are displayed in the decreasing order of the relevance, and wherein the translated results are provided with a relevant link to the original engineering artifact.

10. The method as claimed in claim 9, wherein the textual data is indexed by parsing and tokenizing the textual data in the engineering artifacts with the plurality of individual words and adding pre-determined attributes to the recorded index.

We Claim:

1. A method for scaling processing capability of a field device in an industrial control system using a processing sub-system, wherein the industrial control system comprises one or more field devices, one or more input-output devices, at least one controller, and at least one server, the method comprising:

configuring a virtual field device object in the processing sub-system using one or more functional components from a plurality of functional components, and a field device identifier tag of a field device in the industrial control system;

receiving measurement data from the field device in the processing sub-system;

processing the measurement data in the processing sub-system by the virtual field device object corresponding to the field device, to generate processed field device data; and

communicating the processed field device data using one or more pre-configured communication interfaces to the industrial control system.

2. The method of claim 1 wherein at least one of the one or more pre-configured communication interfaces is configured for communicating the processed field device data to at least one the field device, an input-output device coupled to the field device, the at least one controller, and the server.

3. The method of claim 1 wherein the one or more pre-configured communication interfaces are configured to receive at least one of request commands, input-output commands, and control commands from at least one the field device, an input-output device coupled to the field device, the at least one controller, and the server.

4. The method of claim 1 wherein configuring the virtual field device object triggers receiving of the measurement data from the field device.

5. The method of claim 1 wherein the one or more pre-configured communication interfaces use at least one of wired or wireless communication means.

6. The method of claim 1 wherein receiving measurement data is done using a coupler, wherein the coupler is configured to receive measurement data from the one or more field devices and transmit measurement data identified for each of the one or more field devices to the processing sub-system using a unique field device identifier tag for each of the one or more field devices.

7. A control system for control and monitoring of one or more process conditions in an industrial plant, the control system comprising:

an industrial control system having one or more field devices, one or more input-output devices, at least one controller, and at least one server;

a processing sub-system for scaling processing capability of a field device in the industrial control system, wherein the processing sub-system comprises:

a functional component library module having a plurality of functional components;

a virtual field device object configured using one or more functional components from the plurality of functional components, and a field device identifier tag of a field device in the industrial control system;

a communication module for receiving measurement data from the field device,

wherein the functional components are configured for processing the measurement data to generate processed field device data, and wherein the communication module is configured for communicating the processed field device data; and

one or more pre-configured communication interfaces to communicate the processed field device data to the industrial control system.

8. The control system of claim 7 wherein the one or more pre-configured communication interfaces link a native network of the industrial control system with a second network of the processing sub-system, wherein the second network is different from the native network.

9. The control system of claim 7 further comprising a coupler configured to receive measurement data from the one or more field devices and transmit measurement data identified for each of the one or more field devices to the processing sub-system using a unique field device identifier tag for each of the one or more field devices.

10. The control system of claim 7 wherein the one or more pre-configured communication interfaces communicate the processed field device data to at least one of the field device, an input-output device coupled to the field device, the at least one controller and the server.

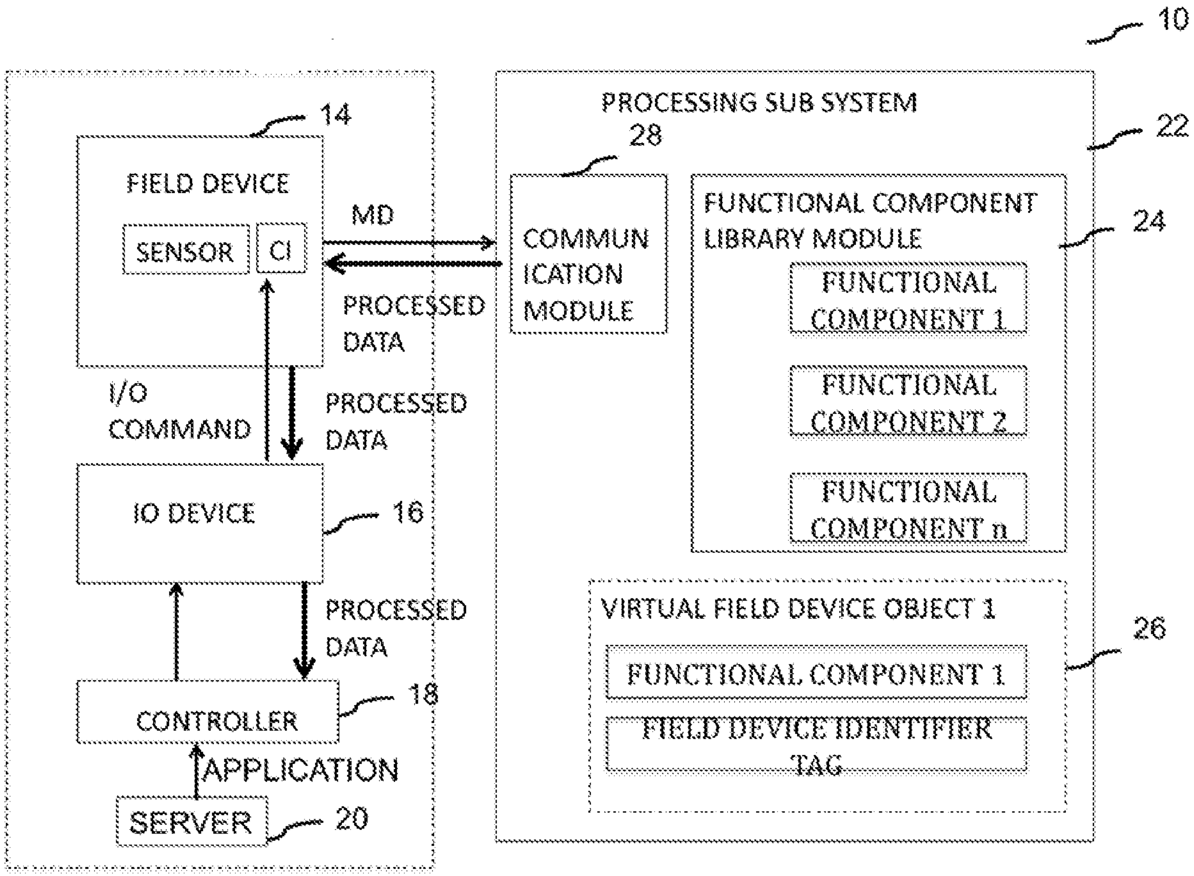

10
PROCESSING SUB SYSTEM
22
28
14
FIELD DEVICE
SENSOR
CI
MD
COMMUN ICATION MODULE
FUNCTIONAL COMPONENT LIBRARY MODULE
24
FUNCTIONAL COMPONENT 1
FUNCTIONAL COMPONENT 2
FUNCTIONAL COMPONENT n
PROCESSED DATA
I/O COMMAND
PROCESSED DATA
IO DEVICE
16
PROCESSED DATA
VIRTUAL FIELD DEVICE OBJECT 1
26
FUNCTIONAL COMPONENT 1
FIELD DEVICE IDENTIFIER TAG
CONTROLLER
18
APPLICATION
SERVER
20

FIG. 1

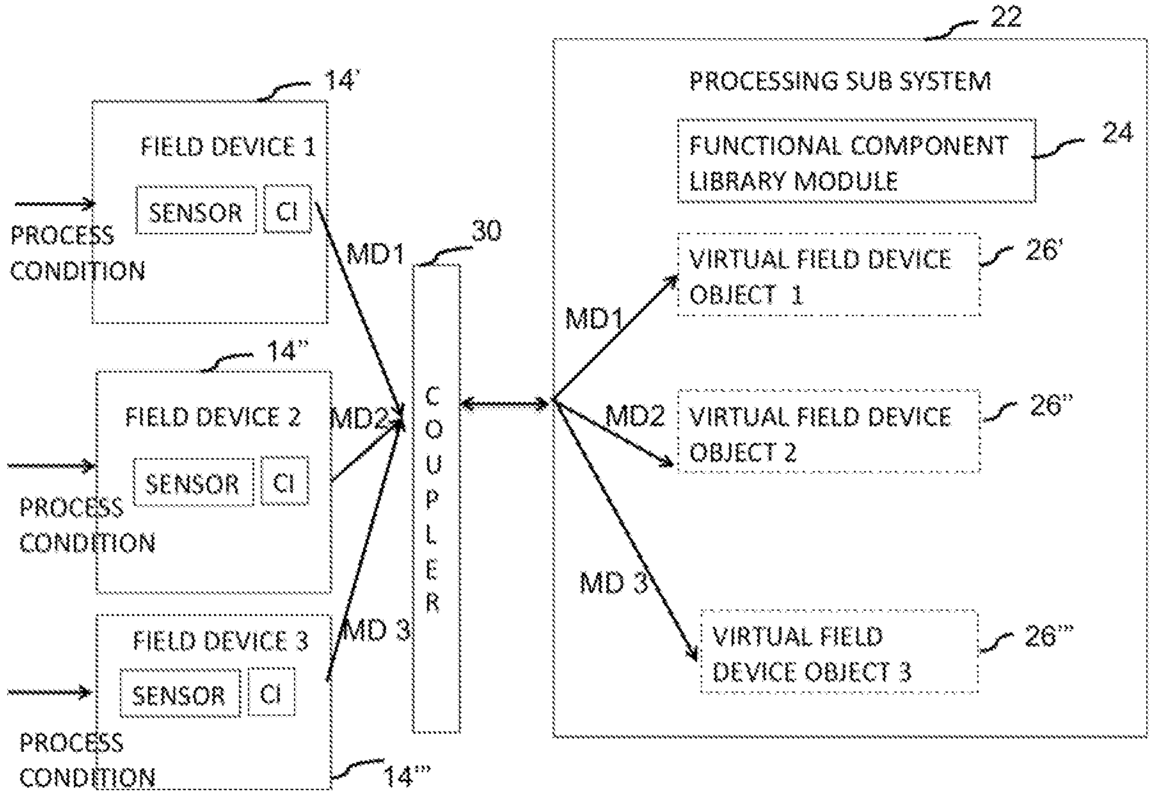

22
PROCESSING SUB SYSTEM
FUNCTIONAL COMPONENT LIBRARY MODULE
24
14'
FIELD DEVICE 1
SENSOR
CI
PROCESS CONDITION
MD1
30
VIRTUAL FIELD DEVICE OBJECT 1
26'
MD1
14''
FIELD DEVICE 2
MD2
SENSOR
CI
PROCESS CONDITION
C O U P L E R
MD2
VIRTUAL FIELD DEVICE OBJECT 2
26''
MD 3
FIELD DEVICE 3
MD 3
SENSOR
CI
PROCESS CONDITION
VIRTUAL FIELD DEVICE OBJECT 3
26'''
14'''

FIG. 2

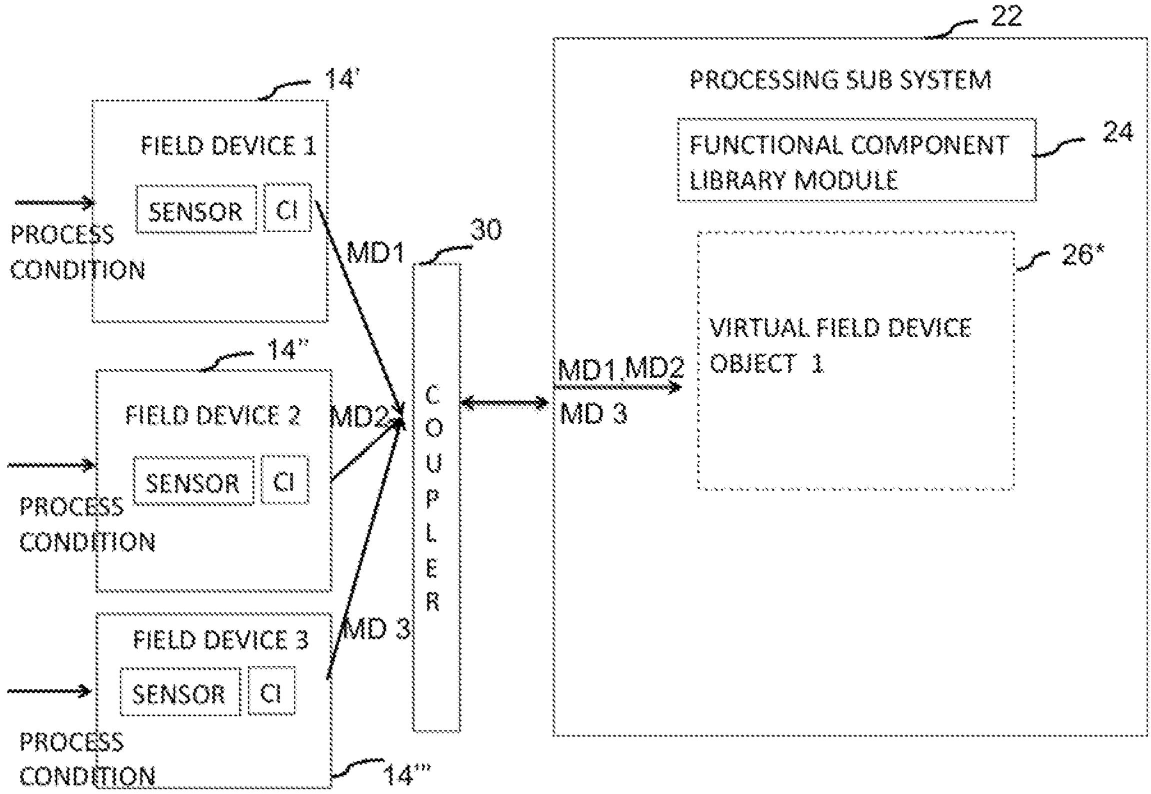

22
PROCESSING SUB SYSTEM
FUNCTIONAL COMPONENT LIBRARY MODULE
24
14'
FIELD DEVICE 1
SENSOR
CI
PROCESS CONDITION
MD1
30
26*
VIRTUAL FIELD DEVICE OBJECT 1
14''
FIELD DEVICE 2
MD2
SENSOR
CI
PROCESS CONDITION
COUPLER
MD1,MD2
MD 3
FIELD DEVICE 3
MD 3
SENSOR
CI
PROCESS CONDITION
14'''

FIG. 3

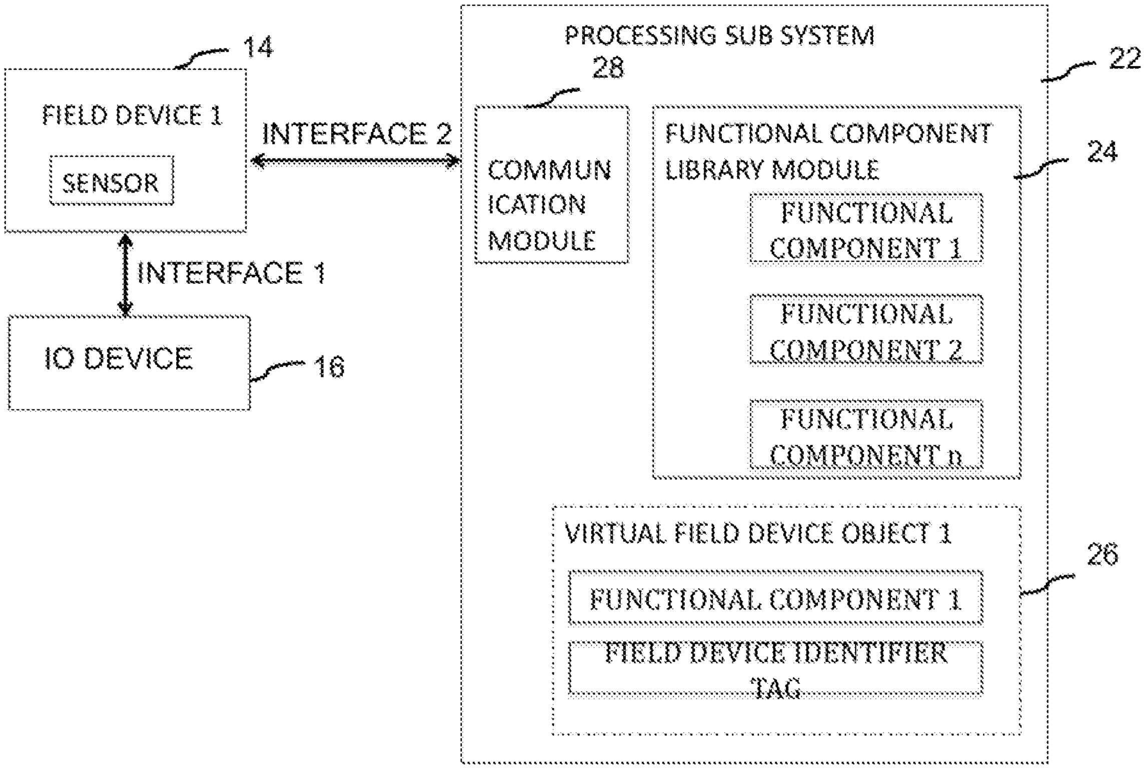

PROCESSING SUB SYSTEM
22
14
FIELD DEVICE 1
SENSOR
INTERFACE 2
28
COMMUN ICATION MODULE
FUNCTIONAL COMPONENT LIBRARY MODULE
24
FUNCTIONAL COMPONENT 1
FUNCTIONAL COMPONENT 2
FUNCTIONAL COMPONENT n
INTERFACE 1
IO DEVICE
16
VIRTUAL FIELD DEVICE OBJECT 1
26
FUNCTIONAL COMPONENT 1
FIELD DEVICE IDENTIFIER TAG

FIG. 4

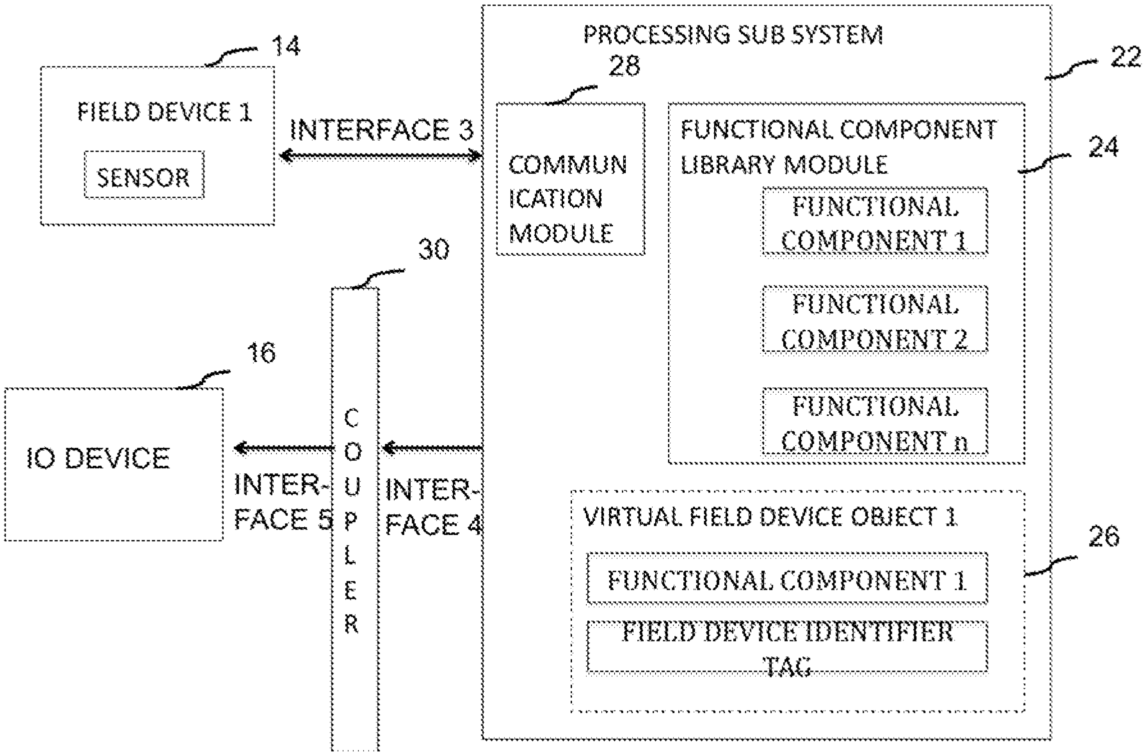

PROCESSING SUB SYSTEM
22
28
14
FIELD DEVICE 1
SENSOR
INTERFACE 3
COMMUN
ICATION
MODULE
FUNCTIONAL COMPONENT
LIBRARY MODULE
24
FUNCTIONAL
COMPONENT 1
FUNCTIONAL
COMPONENT 2
FUNCTIONAL
COMPONENT n
30
16
IO DEVICE
COUPLER
INTER-
FACE 5
INTER-
FACE 4
VIRTUAL FIELD DEVICE OBJECT 1
26
FUNCTIONAL COMPONENT 1
FIELD DEVICE IDENTIFIER
TAG

FIG. 5

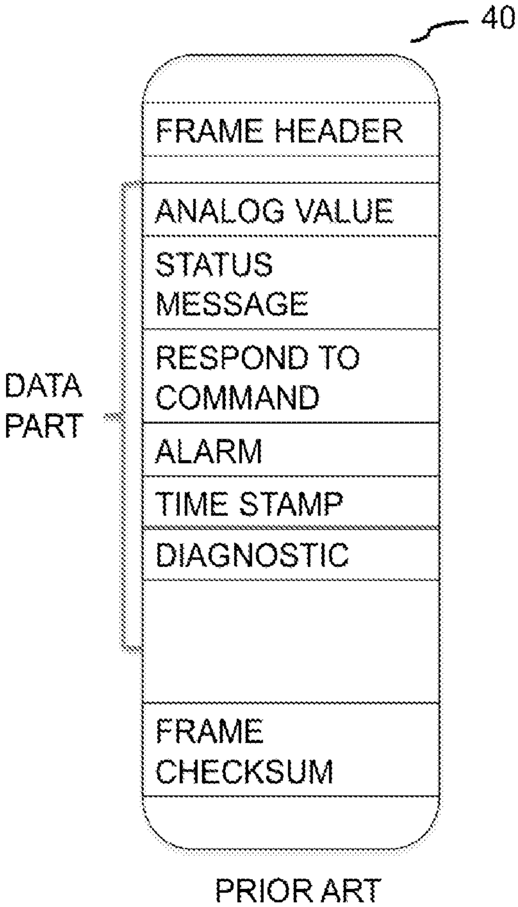

40
FRAME HEADER
ANALOG VALUE
STATUS MESSAGE
RESPOND TO COMMAND
ALARM
TIME STAMP
DIAGNOSTIC
FRAME CHECKSUM
DATA PART
PRIOR ART

FIG. 6

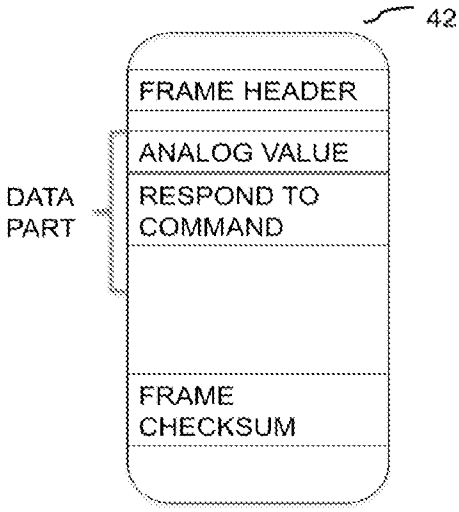
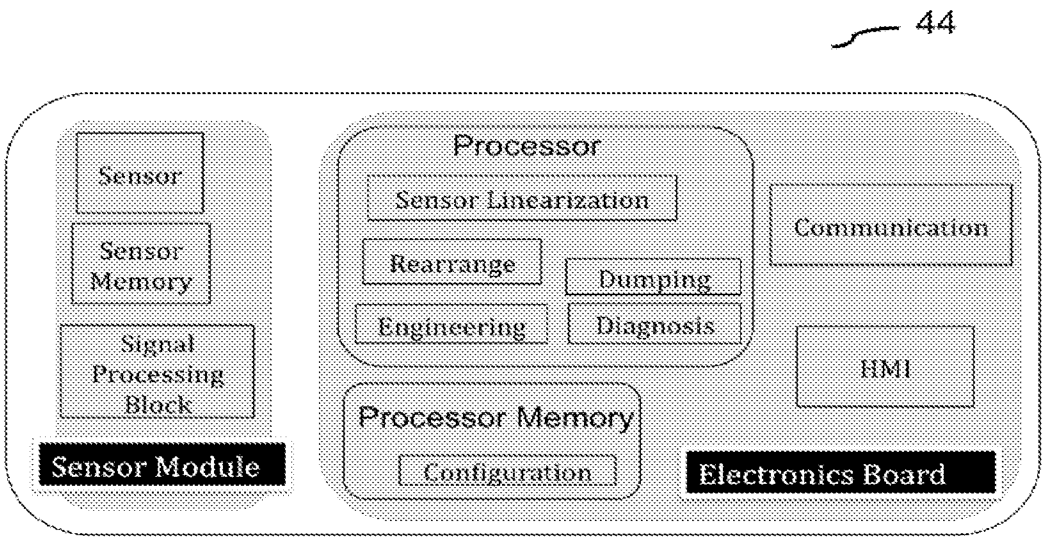

42
FRAME HEADER
ANALOG VALUE
RESPOND TO COMMAND
FRAME CHECKSUM
DATA PART

FIG. 7



PRIOR ART

FIG. 8

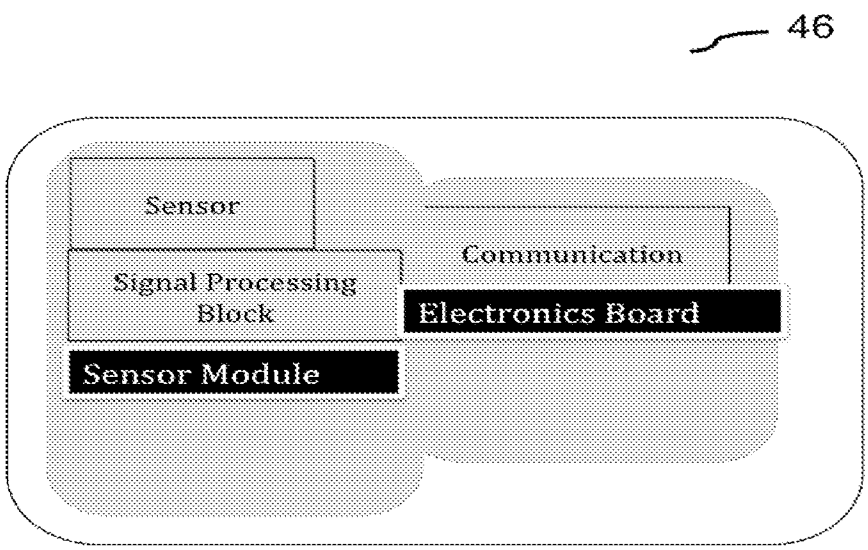

46
Sensor
Communication
Signal Processing Block
Electronics Board
Sensor Module

FIG. 9

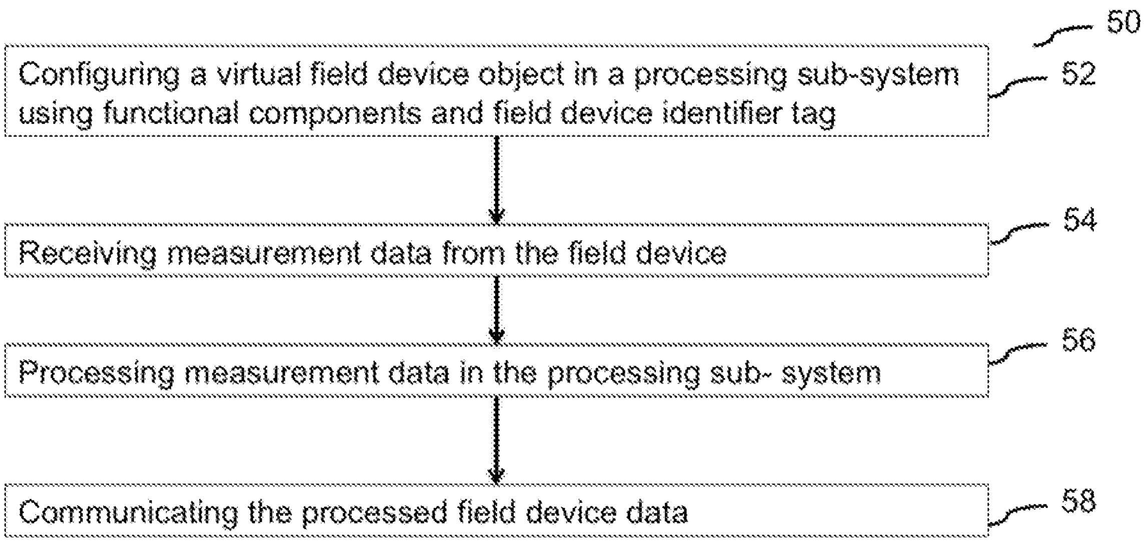

50
Configuring a virtual field device object in a processing sub-system using functional components and field device identifier tag
52
Receiving measurement data from the field device
54
Processing measurement data in the processing sub- system
56
Communicating the processed field device data
58

FIG. 10

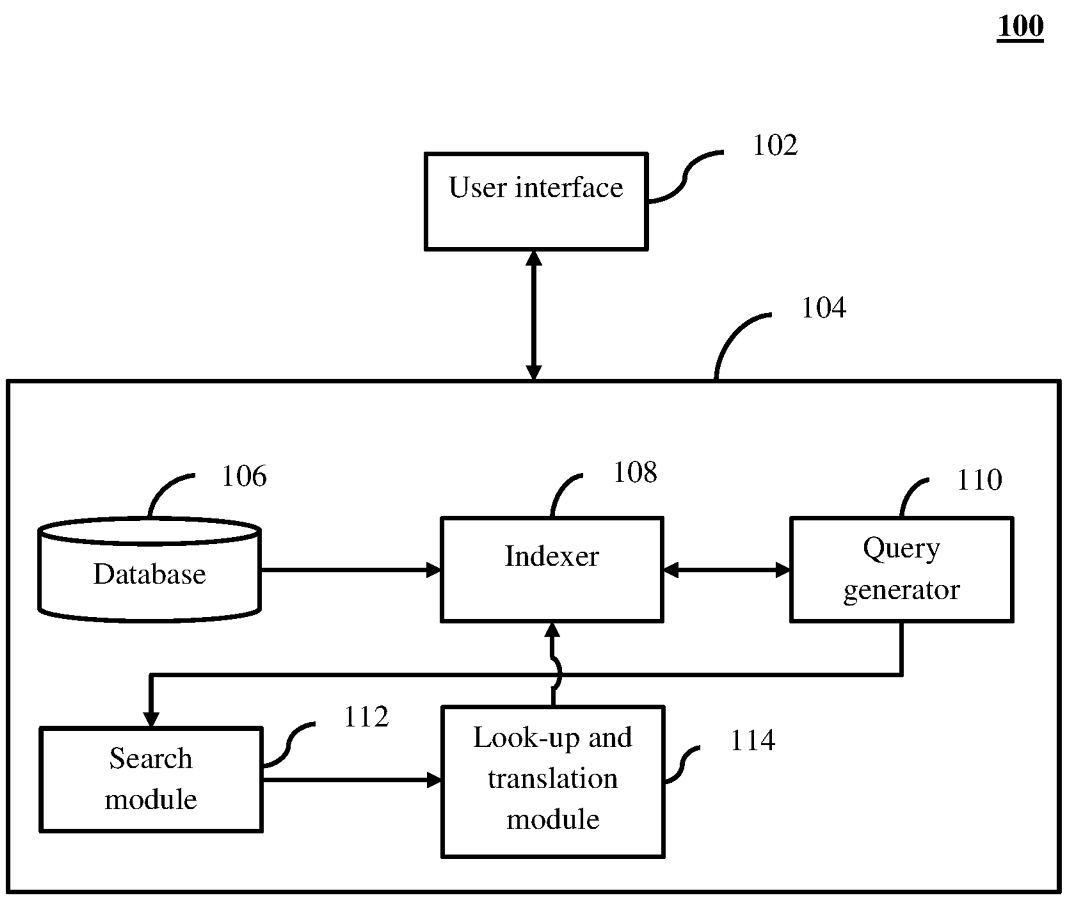

100
102
User interface
104
106
Database
108
Indexer
110
Query
generator
112
Search
module
Look-up and
translation
module
114

FIG. 1

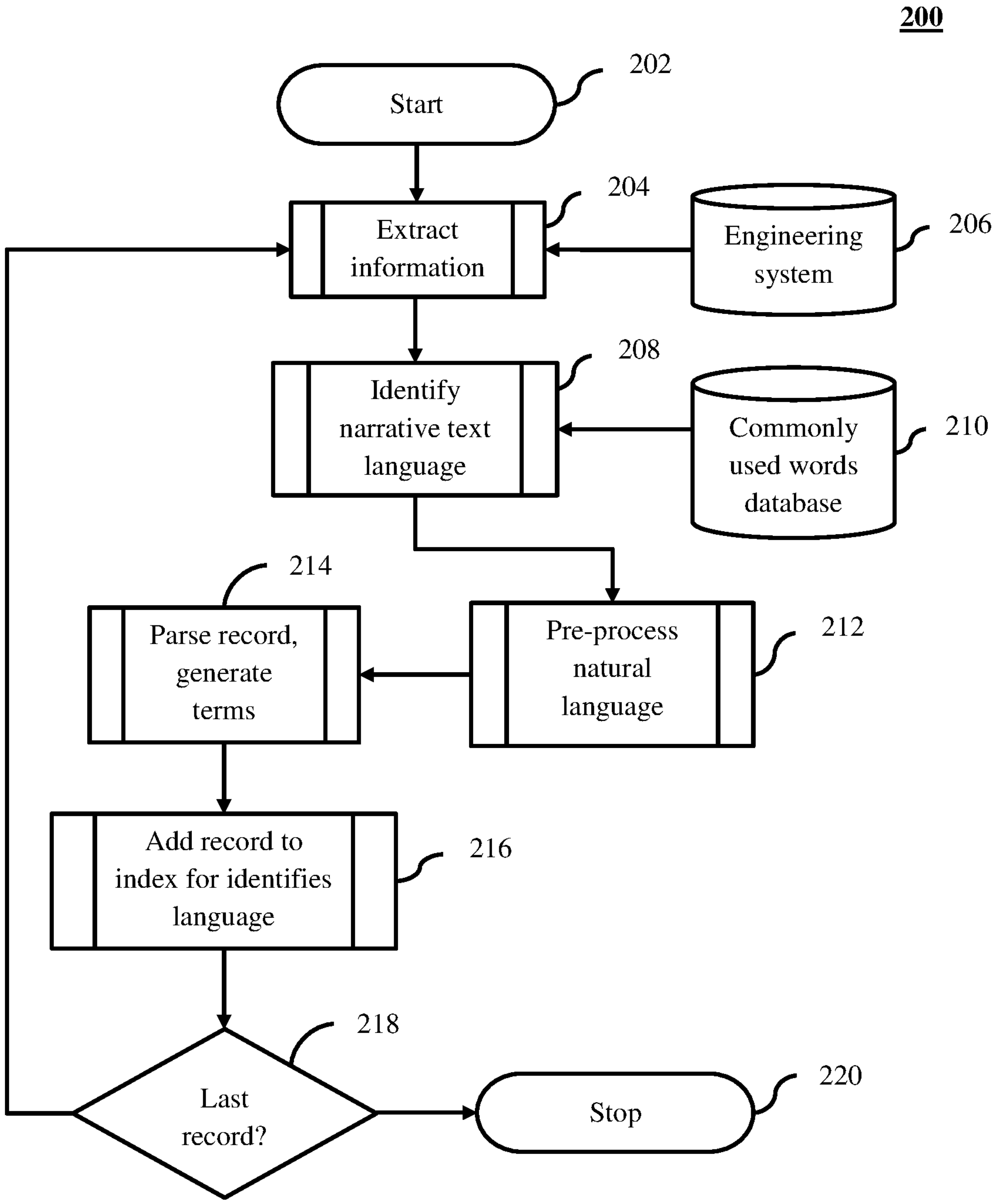

200
Start
202
Extract information
204
Engineering system
206
Identify narrative text language
208
Commonly used words database
210
Parse record, generate terms
214
Pre-process natural language
212
Add record to index for identifies language
216
Last record?
218
Stop
220

FIG. 2

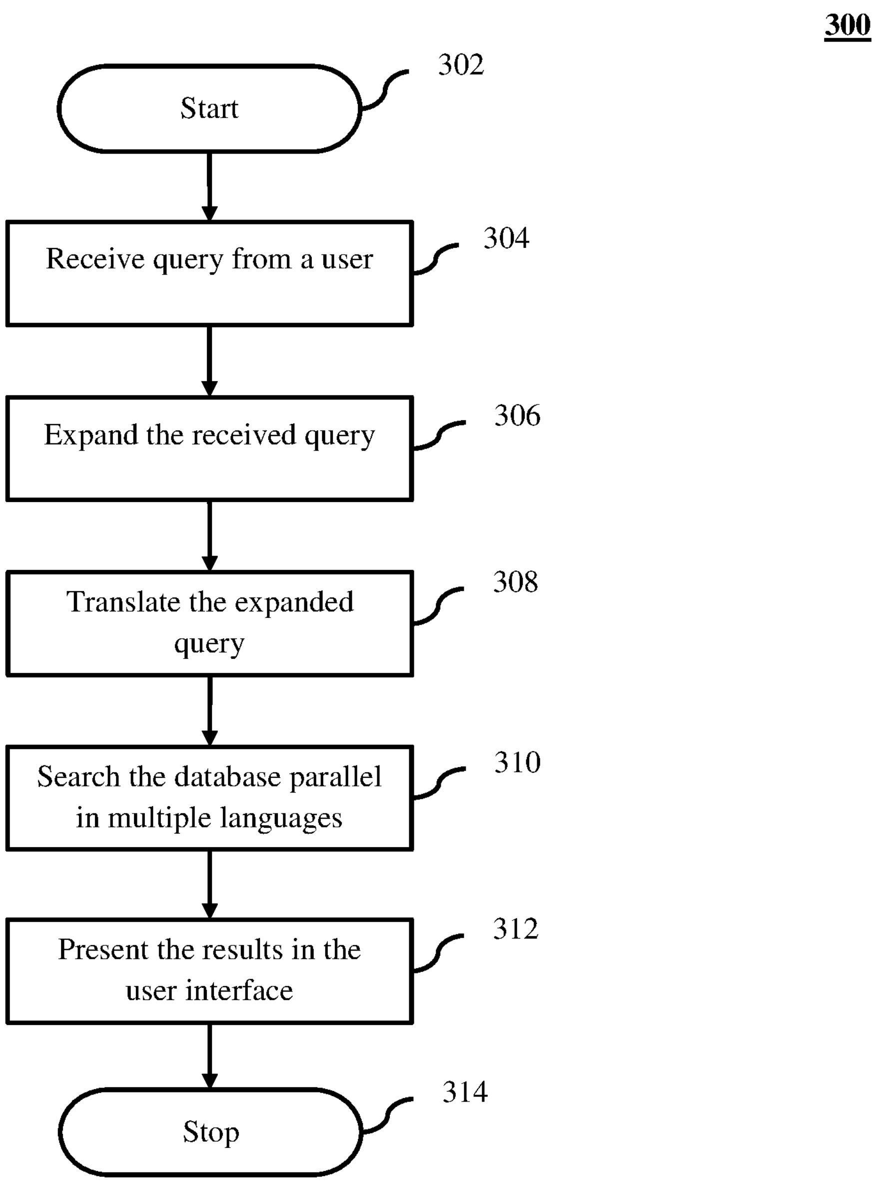

300
302
Start
304
Receive query from a user
306
Expand the received query
308
Translate the expanded query
310
Search the database parallel in multiple languages
312
Present the results in the user interface
314
Stop

FIG. 3