

(51) International Patent Classification:  
*G06F 16/242* (2019.01)(21) International Application Number:  
PCT/US2020/016719(22) International Filing Date:  
05 February 2020 (05.02.2020)

(25) Filing Language: English

(26) Publication Language: English

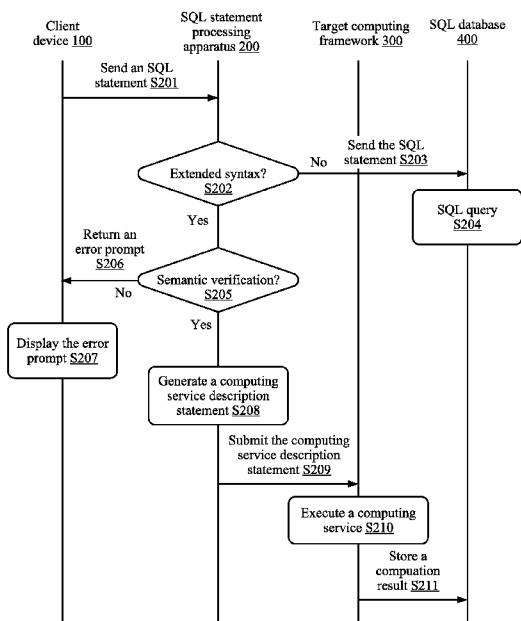
(30) Priority Data:  
201910183056.8 12 March 2019 (12.03.2019) CN(71) Applicant: **ALIBABA GROUP HOLDING LIMITED**  
[—/US]; Fourth Floor, One Capital Place, P.O. Box 847,  
George Town, KY1-1103 (KY).(72) Inventors: **WANG, Yi**; Alibaba Group Legal Department,  
5/f Building 3, No. 969 West Wen Yi Road, Yu Hang Dis-  
trict, Hangzhou, Zhejiang, 311121 (CN). **YANG, Yang**; Al-  
ibaba Group Legal Department, 5/f Building 3, No. 969  
West Wen Yi Road, Yu Hang District, Hangzhou, Zhe-  
jiang, 311121 (CN). **ZHU, Weiguo**; Alibaba Group Legal  
Department, 5/f Building 3, No. 969 West Wen Yi Road,  
Yu Hang District, Hangzhou, Zhejiang, 311121 (CN). **LIU,**  
**Yongfeng**; Alibaba Group Legal Department, 5/f Build-ing 3, No. 969 West Wen Yi Road, Yu Hang District,  
Hangzhou, Zhejiang, 311121 (CN).(74) Agent: **MATTSON, Matthew J.**; Fish & Richardson P.C.,  
P.O. Box 1022, Minneapolis, MN 55440-1022 (US).(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, DJ, DK, DM, DO,  
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,  
HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP,  
KR, KW, 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, ST, SV, SY, TH, TJ, TM, TN, TR,  
TT, TZ, UA, UG, US, UZ, VC, VN, WS, 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).(54) Title: SOLUTION FOR IMPLEMENTING COMPUTING SERVICE BASED ON STRUCTURED QUERY LANGUAGE  
STATEMENT

FIG. 2

(57) Abstract: A computing service based on an SQL statement is pre-  
sented. In an implementation, an existing SQL syntax is extended, so  
that an extended SQL statement includes a computing element needed  
by a computing service, and an SQL statement processing apparatus  
is created between a client and a target computing framework. Syntax  
parsing is performed on a received SQL statement. If parsing indicates  
that the SQL statement is an extended SQL statement, a computing ser-  
vice description statement that can be recognized by the target comput-  
ing framework is generated based on computing element information  
included in the SQL statement, and then the computing service descrip-  
tion statement is submitted to the target computing framework. A com-  
puting service is performed in the target computing framework based on  
the computing service description statement, so that a user's ability to  
write programs needed for target computing framework is not required.

**Published:**

— *with international search report (Art. 21(3))*

# **SOLUTION FOR IMPLEMENTING COMPUTING SERVICE BASED ON STRUCTURED QUERY LANGUAGE STATEMENT**

## **CROSS-REFERENCE TO RELATED APPLICATIONS**

[0001] This application claims priority to Chinese Patent Application No. 201910183056.8, filed on March 12, 2019, which is hereby incorporated by reference in its entirety.

## **TECHNICAL FIELD**

[0002] The present disclosure generally relates to the field of computer technologies, and more specifically, to a method and apparatus for implementing a computing service based on a structured query language statement.

## **BACKGROUND**

[0003] When computation is performed by using structured query language (SQL) statement data, for example, when machine learning model training is performed by using SQL data, a user usually needs to write a machine learning script by using the R or Python programming language, and embeds the written machine learning script in a SQL program statement for execution, so as to implement the machine learning model training. In this case, the user needs to be familiar with the SQL language and other non-SQL languages such as the Python language, and able to perform program writing by using the SQL language and the other non-SQL languages such as the Python language. Consequently, higher demands are imposed on the user.

## **SUMMARY**

[0004] In view of the previous description, the present disclosure provides a method and apparatus for implementing a computing service based on an SQL statement. By using the method and the apparatus, an existing SQL syntax is extended, so that an extended SQL statement includes information about a computing element needed by a computing service, and after an extended SQL statement is received, the extended SQL statement is parsed and a computing service description

statement that can be recognized by a target computing framework is generated based on computing element information. As such, a computing service is performed in the target computing framework based on the generated computing service description statement, so that a user's ability to write programs needed for target computing framework is not required.

**[0005]** According to an aspect of the present disclosure, a method for implementing a computing service based on a structured query language statement is provided, including: performing syntax parsing on a structured query language statement to determine whether an extended syntax identifier exists in the structured query language statement, where the extended syntax identifier indicates a target computing service for the structured query language statement; generating a computing service description statement in a first statement format based on the structured query language statement when the extended syntax identifier exists in the structured query language statement, where the first statement format is a statement format that can be recognized by a target computing framework; and submitting the generated computing service description statement to the target computing framework, to invoke, in the target computing framework based on the computing service description statement, data queried by the structured query language statement to perform target computation, where the structured query language statement includes a computing element needed by the target computing service, when the extended syntax identifier exists in the structured query language statement.

**[0006]** Optionally, in an example of the previous aspect, the target computing service includes a machine learning computing service or a cluster computing service.

**[0007]** Optionally, in an example of the previous aspect, the machine learning computing service includes machine learning model training or machine learning model prediction.

**[0008]** Optionally, in an example of the previous aspect, the generating a computing service description statement in a first statement format based on the structured query language statement includes: generating data query program code in the first statement format based on a data query element in the structured query language statement; and generating machine learning program code in the first statement format based on a computing service element in the structured query language statement.

**[0009]** Optionally, in an example of the previous aspect, the computing element includes a machine learning model name, a machine learning model parameter, a machine learning model feature, a machine learning model label, and a machine learning computation result storage address; and the generating machine learning program code in the first statement format based on a computing service element in the structured query language statement includes: invoking a machine learning program code template corresponding to the machine learning model name; and providing

the machine learning model parameter, the machine learning model feature, and the machine learning model label to the machine learning program code template, to generate the machine learning program code in the first statement format.

**[0010]** Optionally, in an example of the previous aspect, after the target computing framework invokes the data queried by the structured query language statement to perform the target computation based on the machine learning program code, the target computing framework stores a target computation result in the machine learning computation result storage address in a structured query language database.

**[0011]** Optionally, in an example of the previous aspect, before the generating a computing service description statement in a first statement format based on the structured query language statement, the method further includes: performing semantic verification on the structured query language statement.

**[0012]** Optionally, in an example of the previous aspect, the method can further include: obtaining the structured query language statement input from a client input.

**[0013]** According to another aspect of the present disclosure, an apparatus for implementing a computing service based on a structured query language statement is provided, including: a syntax parsing unit, configured to perform syntax parsing on a structured query language statement to determine whether an extended syntax identifier exists in the structured query language statement, where the extended syntax identifier indicates a target computing service for the structured query language statement; a computing service description statement generation unit, configured to generate a computing service description statement in a first statement format based on the structured query language statement when the extended syntax identifier exists in the structured query language statement, where the first statement format is a statement format that can be recognized by a target computing framework; and a description statement submission unit, configured to submit the generated computing service description statement to the target computing framework, to invoke, in the target computing framework based on the computing service description statement, data queried by the structured query language statement to perform target computation, where the structured query language statement includes a computing element needed by the target computing service, when the extended syntax identifier exists in the structured query language statement.

**[0014]** Optionally, in an example of the previous aspect, the target computing service includes a machine learning computing service or a cluster computing service.

[0015] Optionally, in an example of the previous aspect, the machine learning computing service includes machine learning model training or machine learning model prediction; and the computing service description statement generation unit includes: a data query code generation module, configured to generate data query program code in the first statement format based on a data query element in the structured query language statement; and a machine learning code generation module, configured to generate machine learning program code in the first statement format based on a computing service element in the structured query language statement.

[0016] Optionally, in an example of the previous aspect, the computing element includes a machine learning model name, a machine learning model parameter, a machine learning model feature, a machine learning model label, and a machine learning computation result storage address; and the machine learning code generation module is configured to invoke a machine learning program code template corresponding to the machine learning model name, and provide the machine learning model parameter, the machine learning model feature, and the machine learning model label to the machine learning program code template, to generate the machine learning program code in the first statement format.

[0017] Optionally, in an example of the previous aspect, the apparatus can further include: a semantic verification unit, configured to perform semantic verification on the structured query language statement before the computing service description statement in the first statement format is generated based on the structured query language statement.

[0018] Optionally, in an example of the previous aspect, the apparatus can further include: a statement acquisition unit, configured to obtain the structured query language statement from a client input.

[0019] According to another aspect of the present disclosure, a system for implementing a computing service based on a structured query language statement includes: a client device, configured to input a structured query language statement; the previously described apparatus for implementing a computing service based on a structured query language statement; a target computing framework; and a structured query language database.

[0020] According to another aspect of the present disclosure, a computing device is provided, including: at least one processor, and a storage device coupled to the at least one processor, where the storage device stores an instruction, and when the instruction is executed by the at least one processor, the at least one processor is enabled to perform the previously described method for implementing a computing service based on a structured query language statement.

[0021] According to another aspect of the present disclosure, a non-transitory machine-readable storage medium is provided, where the non-transitory machine-readable storage medium stores an executable instruction, and when the instruction is executed, the machine is enabled to perform the previously described method for implementing a computing service based on a structured query language statement.

## BRIEF DESCRIPTION OF DRAWINGS

[0022] Further understanding of the essence and advantages of the content of the present disclosure can be implemented with reference to the following accompanying drawings. In the accompanying drawings, similar components or features can have the same reference numerals.

[0023] FIG. 1 is a block diagram illustrating a system for implementing a computing service based on a structured query language statement, according to an implementation of the present disclosure;

[0024] FIG. 2 is a flowchart illustrating a method for implementing a computing service based on a structured query language statement, according to an implementation of the present disclosure;

[0025] FIG. 3A is a schematic diagram illustrating an example of a conventional SQL statement, according to an implementation of the present disclosure;

[0026] FIG. 3B is a schematic diagram illustrating an example of an SQL statement with an extended syntax identifier, according to an implementation of the present disclosure;

[0027] FIG. 3C is a schematic diagram illustrating another example of an SQL statement with an extended syntax identifier, according to an implementation of the present disclosure;

[0028] FIG. 4 is a flowchart illustrating a process of generating a computing service description statement based on a structured query language statement, according to an implementation of the present disclosure;

[0029] FIG. 5 is a block diagram illustrating an apparatus for implementing a computing service based on a structured query language statement, according to an implementation of the present disclosure;

[0030] FIG. 6 is a block diagram illustrating an implementation example of a computing service description statement generation unit, according to an implementation of the present disclosure; and

[0031] FIG. 7 is a block diagram illustrating a computing device for implementing a computing service based on a structured query language statement, according to an implementation of the present disclosure.

## DESCRIPTION OF IMPLEMENTATIONS

[0032] The subject matter described in the present specification will now be discussed with reference to example implementations. It should be understood that these implementations are merely discussed to enable a person skilled in the art to better understand and implement the subject matter described in the present specification, and are not intended to limit the protection scope, applicability, or examples described in the claims. The functions and arrangements of the discussed elements can be changed without departing from the protection scope of the content of the present disclosure. Various processes or components can be omitted, replaced, or added in examples based on demands. For example, the described method can be performed in an order different from that described here, and steps can be added, omitted, or combined. In addition, features described related to some examples can also be combined in other examples.

[0033] As used in the present specification, the term "include" and its variant represent open terms, meaning "including but not limited to". The term "based" means "at least partially based on". The terms "one implementation" and "an implementation" represent "at least one implementation". The term "another implementation" represents "at least one another implementation". The terms "first", "second", etc. can refer to different objects or the same object. The following can include other definitions, whether explicit or implicit. The definition of a term is consistent throughout the specification, unless otherwise explicitly specified in the context.

[0034] In an implementation of the present disclosure, an existing SQL syntax is extended, so that an extended SQL statement includes information about a computing element needed by a computing service, and an SQL statement processing apparatus is created between an SQL engine (i.e., an SQL client) and a target computing framework. The SQL statement processing apparatus receives an SQL statement submitted by the SQL engine, and performs syntax parsing on the SQL statement submitted by the SQL engine. After it is obtained through parsing that the received SQL statement is an extended SQL statement, a computing service description statement that can be recognized by the target computing framework is generated based on computing element information included in the SQL statement, and then the generated computing service description statement is submitted to the target computing framework. As such, a computing service is performed in the target computing framework based on the generated computing service description statement, so that a user's ability to write programs needed for target computing framework is not required.

[0035] The following describes in detail, with reference to the accompanying drawings, a system and method for implementing a computing service based on an SQL statement according to implementations of the present disclosure.

[0036] FIG. 1 is a block diagram illustrating a system 10 for implementing a computing service based on a structured query language statement (referred to as a "computing service implementation system" for short below), according to an implementation of the present disclosure. As shown in FIG. 1, the computing service implementation system 10 includes a client device 100, an SQL statement processing apparatus 200, a target computing framework 300, and an SQL database 400.

[0037] The client device 100 is configured to write an SQL statement and submit the written SQL statement to the SQL statement processing apparatus 200. For example, a user can directly write an SQL statement in a browser or in a command line, and then the client device 100 sends the SQL statement to the SQL statement processing apparatus 200 by using a streaming RPC message. In the present disclosure, the SQL statement processing apparatus 200 can support a plurality of clients. For example, a client of the browser can be implemented by adding a Python client to the Jupyter Notebook. The client device 100 has an interactive user interface with good experience. After the SQL statement is entered and submitted through the interactive user interface, status information of the SQL statement process can be displayed on the interactive user interface in real time.

[0038] After receiving the SQL statement submitted by the client device 100, the SQL statement processing apparatus 200 performs syntax parsing on the received SQL statement. After a syntax parsing result indicates that the SQL statement is a conventional SQL statement, the SQL statement processing apparatus 200 initiates an SQL query operation to the SQL database 400. After the syntax parsing result indicates that the SQL statement is an extended SQL statement, the SQL statement processing apparatus 200 generates a computing service description statement that can be recognized by the target computing framework 300 based on computing element information included in the SQL statement, and then submits the generated computing service description statement to the target computing framework 300. In the present disclosure, the SQL statement processing apparatus 200 supports an operation performed based on an interactive command line statement, and can return the status information of the SQL statement process to the client device 100 in real time. In the present disclosure, the SQL statement processing apparatus 200 can also be referred to as SQLFlow. Operations and structures of the SQL statement processing apparatus will be described in detail below with reference to the accompanying drawings.

[0039] After receiving the computing service description statement, the target computing framework 300 can invoke, based on the computing service description statement, data queried by

the structured query language statement to perform target computation. After the target computation is completed, the target computing framework 300 stores a computation result in the SQL database, for example, storing the computation result at an SQL database address specified in the computing service description statement.

**[0040]** The SQL database 400 is configured as a data storage engine for storing to-be-queried data. In the present disclosure, the SQL database can include MySQL, SQLite, Hive, ODPS, SparkSQL, etc.

**[0041]** FIG 2 is a flowchart illustrating a method for implementing a computing service based on a structured query language statement, according to an implementation of the present disclosure.

**[0042]** As shown in FIG 2, in block S201, after writing an SQL statement on a client device 100, a user sends the written SQL statement to an SQL statement processing apparatus 200. Message exchange is performed between the client device 100 and the SQL statement processing apparatus 200 by using a streaming RPC message. The sent SQL statement can be a conventional SQL statement, or can be an SQL statement with an extended syntax identifier, that is, an extended SQL statement. An SQL statement syntax extension rule here is specified in advance, and known by both the client device 100 and the SQL statement processing apparatus 200 beforehand.

**[0043]** FIG 3A is a schematic diagram illustrating an example of a conventional SQL statement, according to an implementation of the present disclosure. The SQL statement shown in FIG 3A is a conventional SQL statement. The conventional SQL statement indicates "query data with a target attribute from iris.iris", where the target attribute is represented by "\*" in the SQL statement.

**[0044]** FIG 3B is a schematic diagram illustrating an example of an SQL statement with an extended syntax identifier, according to an implementation of the present disclosure. The SQL statement shown in FIG 3B is an extended SQL statement. The extended SQL statement includes a target computing service used for indicating the structured query language statement. As shown in FIG 3B, the extended SQL statement includes an extended syntax identifier "TRAIN", and the extended syntax identifier indicates that the target computing service is machine learning model training. In addition, the SQL statement shown in FIG 3B further includes a computing element needed by the target computing service. When the target computing service is the machine learning model training, the computing element can include a machine learning model name, a machine learning model parameter, a machine learning model feature, a machine learning model label, and a machine learning computation result storage address (i.e., a machine learning computation result storage address).

**[0045]** Specifically, in the SQL statement shown in FIG 3B, the machine learning model name

is "DNNClassifier", that is, a DNN classifier. The machine learning model parameter is "n\_classes=3, hidden\_units=[10, 20]", to be specific, a quantity of classification categories is 3, and a quantity of hidden layer neurons is [10, 20]. The machine learning model feature is "sepal\_length, sepal\_width, petal\_length, petal\_width". The machine learning model label is "class". The machine learning computation result storage address is "my\_dnn\_model", for example, a table in an SQL database. It is worthwhile to note here that the illustration shown in FIG. 3B is merely an example.

[0046] FIG. 3C is a schematic diagram illustrating another example of an SQL statement with an extended syntax identifier, according to an implementation of the present disclosure. As shown in FIG. 3C, the extended SQL statement includes an extended syntax identifier "PREDICT", and the extended syntax identifier indicates that a target computing service is machine learning model prediction. The extended SQL statement further includes "iris.predict.class" and "USING my\_dnn\_model". Here, "iris.predict.class" is used to predict a target, to be specific, to predict a category of iris, and "USING my\_dnn\_model" is used to indicate a used model storage address, to be specific, to use model information stored in my\_dnn\_model. In addition, the computing element described above with reference to FIG. 3B are recorded in information stored at the used model storage address, to be specific, information stored in "my\_dnn\_model".

[0047] In the present disclosure, in addition to "TRAIN" and "PREDICT", the extended syntax identifier can further include "Cluster". The extended syntax identifier "Cluster" is used to instruct to perform cluster computation, for example, third-party cluster computation. In addition, the extended syntax identifier can further include an extended syntax identifier used to indicate another suitable computing service.

[0048] Returning back to FIG. 2, after receiving the SQL statement from the client device 100, the SQL statement processing apparatus 200 performs syntax parsing on the structured query language statement to determine whether an extended syntax identifier exists in the SQL statement in block S202.

[0049] When no extended syntax identifier exists in the SQL statement, the SQL statement is sent to an SQL database 400 in block S203, and SQL query is performed in the SQL database 400 in block S204. The SQL database 400 returns data obtained through querying to the client device 100 via the SQL statement processing apparatus 200, to display the data on a display interface of the client device 100.

[0050] When the extended syntax identifier exists in the SQL statement, semantic verification is performed on the SQL statement in block S205. Specifically, validity of a data query part in the SQL statement can be verified. For example, the SQL statement processing apparatus 200 can select, based on a data query element in the SQL statement, a small amount of to-be-queried data, to

initiate data query to the SQL database 400. If the SQL database 400 can return a query result, it can be considered that the semantic verification succeeds. If the SQL database 400 cannot return a query result, it can be considered that the semantic verification fails. When the semantic verification fails, an error prompt is returned to the client device 100 in block S206, and the error prompt is displayed on the display interface of the client device 100 to notify the user in block S207.

**[0051]** After the semantic verification succeeds, the SQL statement processing apparatus 200 generates a computing service description statement in a first statement format based on the SQL statement in block S208. Here, the first statement format is a statement format that can be recognized by a target computing framework. In the present disclosure, the target computing framework can include TensorFlow, Pytorch, Xgboost, AutoML, etc. The target computing framework can be described, for example, by a computing element in the SQL statement that uniquely identifies the target computing framework. For example, when a target computing service is the machine learning model training shown in FIG 3B, the target computing framework can be identified by the machine learning model name "DNNClassifier". Here, "DNNClassifier" is, for example, used to indicate that the target computing framework is "TensorFlow". Therefore, the SQL statement processing apparatus 200 generates, based on the SQL statement, a computing service description statement in a statement format that can be recognized by TensorFlow. In addition, a separate identifier can also be set in the computing element to uniquely identify the target computing framework.

**[0052]** In the present disclosure, the target computing service can include a machine learning computing service, a cluster computing service, etc. In addition, the machine learning computing service includes machine learning model training or machine learning model prediction.

**[0053]** When the target computing service is cluster computing, the computing service description statement is a description statement used to execute cluster computation. For example, the description statement can include at least one of the following: a computing task description, training information, loaded data, or model storage information. When the target computing service is the machine learning computing service, the computing service description statement is machine learning program code. A method for generating the machine learning program code is described in detail below with reference to FIG 4.

**[0054]** FIG. 4 is a flowchart illustrating an example process of generating a computing service description statement based on an SQL statement, according to an implementation of the present disclosure. The present example is for a machine learning task.

**[0055]** As shown in FIG. 4, first, in block 410, data query program code in a first statement

format is generated based on a data query element in an SQL statement. For example, when a target computing service is machine learning model training, like the SQL statement shown in FIG. 3B, the data query program code in the first statement format that can be identified by a target computing framework can be generated based on "Select \* From iris.iris" in FIG. 3B. The data query program code can be executed by the target computing framework to obtain data for the machine learning model training from an SQL database 400. When the target computing service is another type of machine learning task, data query program code in a corresponding format can be generated.

[0056] Then, in block 420, machine learning program code in the first statement format is generated based on a computing service element in the SQL statement.

[0057] Specifically, for the machine learning task, the computing element can include a machine learning model name, a machine learning model parameter, a machine learning model feature, a machine learning model label, and a machine learning computation result storage address.

[0058] Correspondingly, that machine learning program code in the first statement format is generated based on a computing service element in the SQL statement can include the following: A machine learning program code template corresponding to the machine learning model name is invoked, and the machine learning model parameter, the machine learning model feature, and the machine learning model label are provided for the machine learning program code template, to generate the machine learning program code in the first statement format. When the machine learning task is the machine learning model training, the computing element explicitly includes the previously described information. When the machine learning task is machine learning prediction, the computing element is recorded in information at a used model storage address.

[0059] Returning back to FIG. 2, after the computing service description statement is generated, the SQL statement processing apparatus 200 submits the generated computing service description statement to the target computing framework 300 in block S209.

[0060] After receiving the computing service description statement, the target computing framework 300 invokes data queried by the SQL statement to perform target computation in block S210. For example, when the extended syntax identifier indicates "machine learning model training", the target computing framework 300 obtains data for the machine learning model training from the SQL database 400 by using data query code, and then performs the machine learning model training based on the machine learning program code by using the obtained data.

[0061] Then, in block S211, after the target computing framework 300 completes the target computation, the target computing framework 300 stores a target computation result at a machine learning computation result storage address in the SQL database for users queries, thereby

implementing a computing service according to an implementation of the present disclosure.

[0062] The method for implementing a computing service based on an SQL statement according to an implementation of the present disclosure is described above with reference to FIG. 1 to FIG. 4. It is worthwhile to note here that the illustration shown in FIG. 2 is merely an implementation of the method for implementing a computing service based on an SQL statement according to an implementation of the present disclosure. In another implementation of the present disclosure, the method for implementing a computing service based on an SQL statement shown in FIG. 2 can be further modified.

[0063] For example, in another implementation of the present disclosure, the operation of block S201 in FIG. 2 may not be included. In this case, the SQL statement can be obtained in advance and stored in the SQL statement processing apparatus 200.

[0064] In addition, in another implementation of the present disclosure, the operation of block S205 in FIG. 2 may not be included. In this case, for an extended SQL statement, it can be considered that there is no semantic problem in the extended SQL statement, so that semantic verification is not required.

[0065] In the present disclosure, an existing SQL syntax is extended, so that an extended SQL statement includes information about a computing element needed by a computing service. After an extended SQL statement is received, the extended SQL statement is parsed and a computing service description statement that can be recognized by a target computing framework is generated based on computing element information. As such, a computing service is performed in the target computing framework based on the generated computing service description statement, so that a user's ability to write programs needed for target computing framework is not required.

[0066] In the present disclosure, syntax parsing is performed on an SQL statement to determine whether the SQL statement is an extended SQL statement. When the SQL statement is the extended SQL statement, a corresponding computation operation is performed. When the SQL statement is not the extended SQL statement, a conventional SQL query operation is performed, so that users with different needs can perform operations on a same user interface.

[0067] In addition, in the present disclosure, before a computing service description statement is generated based on an SQL statement, a semantic query is performed on the received SQL statement, so that a semantic error of the extended SQL statement can be discovered as early as possible, thereby preventing submission of useless machine learning tasks to a machine learning code generation component and saving system resources.

[0068] The method for implementing a computing service based on an SQL statement

according to an implementation of the present disclosure is described above with reference to FIG 1 to FIG 4. The following describes an apparatus for implementing a computing service based on an SQL statement according to an implementation of the present disclosure with reference to FIG 5 and FIG 6.

[0069] FIG 5 is a block diagram illustrating an apparatus 200 for implementing a computing service based on an SQL statement (referred to as an SQL statement processing apparatus below), according to an implementation of the present disclosure. As shown in FIG 5, the SQL statement processing apparatus 200 includes a syntax parsing unit 210, a computing service description statement generation unit 220, and a description statement submission unit 230.

[0070] The syntax parsing unit 210 is configured to perform syntax parsing on an SQL statement to determine whether an extended syntax identifier exists in the SQL statement. The extended syntax identifier indicates a target computing service for the SQL statement. When the extended syntax identifier exists in the structured query language statement, the structured query language statement includes a computing element needed by the target computing service. For an operation of the syntax parsing unit 210, reference can be made to the operation of block S202 described above with reference to FIG 2.

[0071] The computing service description statement generation unit 220 is configured to generate a computing service description statement in a first statement format based on the SQL statement when the extended syntax identifier exists in the SQL statement, where the first statement format is a statement format that can be recognized by a target computing framework. For an operation of the computing service description statement generation unit 220, reference can be made to the operation of block S208 described above with reference to FIG 2.

[0072] FIG 6 is a block diagram illustrating an implementation example of a computing service description statement generation unit 220, according to an implementation of the present disclosure. The example shown in FIG 6 is an example of a machine learning computing service. As shown in FIG 6, the computing service description statement generation unit 220 includes a data query code generation module 221 and a machine learning code generation module 223.

[0073] The data query code generation module 221 is configured to generate data query program code in the first statement format based on a data query element in the SQL statement. For an operation of the data query code generation module 221, reference can be made to the operation of block 410 described above with reference to FIG 4.

[0074] The machine learning code generation module 223 is configured to generate machine

learning program code in the first statement format based on a computing service element in the SQL statement. Specifically, when the target computing service includes the machine learning computing service, the computing element includes a machine learning model name, a machine learning model parameter, a machine learning model feature, a machine learning model label, and a machine learning computation result storage address. Correspondingly, the machine learning code generation module is configured to invoke a machine learning program code template corresponding to the machine learning model name, and provide the machine learning model parameter, the machine learning model feature, and the machine learning model label to the machine learning program code template, to generate the machine learning program code in the first statement format. For an operation of the machine learning code generation module 223, reference can be made to the operation of block 420 described above with reference to FIG. 4.

**[0075]** Returning back to FIG.5, the description statement submission unit 230 is configured to submit the generated computing service description statement to the target computing framework 300, to invoke, in the target computing framework 300 based on the computing service description statement, data queried by the SQL statement to perform target computation.

**[0076]** In addition, in another example of the present disclosure, the SQL statement processing apparatus 200 can further include a semantic verification unit (not shown). The semantic verification unit is configured to perform semantic verification on the SQL statement before the computing service description statement in the first statement format is generated based on the SQL statement. In addition, after the SQL statement semantic verification succeeds, the computing service description statement generation unit 220 performs an operation.

**[0077]** In addition, in another example of the present disclosure, the SQL statement processing apparatus 200 can further include a statement acquisition unit (not shown). The statement acquisition unit is configured to obtain the structured query language statement from a client input.

**[0078]** Implementations of the method and apparatus for implementing a computing service based on an SQL statement according to an implementation of the present disclosure are described above with reference to FIG. 1 to FIG. 6. The SQL statement processing apparatus can be implemented by using hardware, or can be implemented by using software or a combination of hardware and software.

**[0079]** FIG. 7 is a block diagram illustrating a computing device 700 for implementing a computing service based on an SQL statement, according to an implementation of the present disclosure. As shown in FIG. 7, the computing device 700 can include at least one processor 710, a storage device 720, a memory 730, and a communications interface 740, and the at least one processor 710, the storage device 720, the memory 730, and the communications interface 740 are

connected together by using a bus 760. The at least one processor 710 executes at least one computer readable instruction (i.e., the previously described element implemented in a software form) stored or encoded in the storage device.

**[0080]** In an implementation, a computer executable instruction is stored in the storage device, and when the computer executable instruction is executed, the at least one processor 710 is enabled to: perform syntax parsing on an SQL statement to determine whether an extended syntax identifier exists in the SQL statement, where the extended syntax identifier indicates a target computing service for the SQL statement; generate a computing service description statement in a first statement format based on the SQL statement when the extended syntax identifier exists in the SQL statement, where the first statement format is a statement format that can be recognized by a target computing framework; and submit the generated computing service description statement to the target computing framework, to invoke, in the target computing framework based on the computing service description statement, data queried by the SQL statement to perform target computation, where the SQL statement includes a computing element needed by the target computing service, when the extended syntax identifier exists in the SQL statement.

**[0081]** It should be understood that, when the computer executable instruction stored in the storage device is executed, the at least one processor 710 is enabled to perform the operations and functions described with reference to FIG 1 to FIG 6 in the implementations of the present disclosure.

**[0082]** In the present disclosure, the computing device 700 can include but is not limited to a personal computer, a server computer, a workstation, a desktop computer, a laptop computer, a notebook computer, a mobile computing device, a smartphone, a tablet computer, a cellular phone, a personal digital assistant (PDA), a handheld apparatus, a message sending and receiving device, a wearable computing device, a consumer electronics, etc.

**[0083]** According to an implementation, a program product such as a non-transitory machine-readable medium is provided. The non-transitory machine-readable medium can have an instruction (i.e., the previously described element implemented in a software form). When the instruction is executed by a machine, the machine performs the operations and functions described with reference to FIG 1 to FIG 6 in the implementations of the present disclosure. Specifically, a system or apparatus equipped with a readable storage medium can be provided, and software program code for implementing a function of any one of the previous implementations is stored in the readable storage medium, so that a computer or a processor of the system or the apparatus reads out and executes an instruction stored in the readable storage medium.

**[0084]** In this case, program code read from the readable medium can implement a function of

any one of the previous implementations. Therefore, machine-readable code and a readable storage medium that stores the machine-readable code constitute a part of the present invention.

**[0085]** An implementation of the readable storage medium include a floppy disk, a hard disk, a magneto-optical disk, an optical disc (such as a CD-ROM, a CD-R, a CD-RW, a DVD-ROM, a DVD-RAM, a DVD-RW, and a DVD-RW), a magnetic tape, a non-volatile memory card, and a ROM. Optionally, program code can be downloaded from a server computer or a cloud through a communications network.

**[0086]** A person skilled in the art should understand that various variations and modifications can be made to the implementations disclosed above without departing from the essence of the present invention. Therefore, the protection scope of the present invention shall be limited by the appended claims.

**[0087]** It is worthwhile to note that not all steps and units in the previous processes and system structural diagrams are necessary, and some steps or units can be ignored based on an actual demand. An execution sequence of the steps is not fixed, and can be determined based on a demand. The apparatus structure described in the previous implementations can be a physical structure, or can be a logical structure, to be specific, some units may be implemented by a same physical entity, or some units may be implemented by a plurality of physical entities, or can be implemented jointly by some components in a plurality of independent devices.

**[0088]** In the previous implementations, the hardware unit or module can be implemented in a mechanical or electrical method. For example, a hardware unit or module or a processor can include a permanent dedicated circuits or logic (e.g., a dedicated processor, an FPGA, or an ASIC) to complete a corresponding operation. The hardware unit or the processor can further include a programmable logic or circuit (e.g. a general purpose processor or another programmable processor), and can be temporarily set by software to complete a corresponding operation. A specific implementation method (a mechanical manner, a dedicated permanent circuit, or a temporarily set circuit) can be determined based on costs and time.

**[0089]** Exemplary implementations are described in specific implementations that are illustrated above with reference to the accompanying drawings. However, the implementations do not represent all implementations that can be implemented or fall within the protection scope of the claims. The term "example" used throughout the present specification means "used as an example, instance, or illustration" and does not mean "preferred" or "advantageous" over other implementations. For the purpose of providing an understanding for the described technologies, a specific implementation includes specific details. However, these technologies can be implemented without these specific details. In some instances, well-known structures and apparatuses are shown

in block diagrams to avoid making it difficult to understand the concepts of the described implementations.

**[0090]** The previous descriptions of the present disclosure are provided to enable a person of ordinary skill in the art to implement or use the present disclosure. Various modifications made to the present disclosure are obvious to a person of ordinary skill in the art. In addition, the general principle described in the present specification can be applied to another variant without departing from the protection scope of the present disclosure. Therefore, the present disclosure is not limited to the examples and designs described in the present specification, but is consistent with the widest range that conform to principles and novelty features disclosed in the present specification.

## CLAIMS

What is claimed is:

1. A method for implementing a computing service based on a structured query language statement, comprising:

performing syntax parsing on a structured query language statement to determine whether an extended syntax identifier exists in the structured query language statement, wherein the extended syntax identifier indicates a target computing service for the structured query language statement;

generating a computing service description statement in a first statement format based on the structured query language statement when the extended syntax identifier exists in the structured query language statement, wherein the first statement format is a statement format that can be recognized by a target computing framework; and

submitting the generated computing service description statement to the target computing framework, to invoke, in the target computing framework based on the computing service description statement, data queried by the structured query language statement to perform target computation, wherein the structured query language statement comprises a computing element needed by the target computing service, when the extended syntax identifier exists in the structured query language statement.

2. The method according to claim 1, wherein the target computing service comprises a machine learning computing service or a cluster computing service.

3. The method according to claim 2, wherein the machine learning computing service comprises machine learning model training or machine learning model prediction.

4. The method according to claim 3, wherein the generating a computing service description statement in a first statement format based on the structured query language statement comprises:

generating data query program code in the first statement format based on a data query element in the structured query language statement; and

generating machine learning program code in the first statement format based on a computing service element in the structured query language statement.

5. The method according to claim 4, wherein the computing element comprises a machine

learning model name, a machine learning model parameter, a machine learning model feature, a machine learning model label, and a machine learning computation result storage address; and

the generating machine learning program code in the first statement format based on a computing service element in the structured query language statement comprises:

invoking a machine learning program code template corresponding to the machine learning model name; and

providing the machine learning model parameter, the machine learning model feature, and the machine learning model label to the machine learning program code template, to generate the machine learning program code in the first statement format.

6. The method according to claim 5, wherein after the target computing framework invokes the data queried by the structured query language statement to perform the target computation based on the machine learning program code, the target computing framework stores a target computation result in the machine learning computation result storage address in a structured query language database.

7. The method according to claim 1, wherein before the generating a computing service description statement in a first statement format based on the structured query language statement, the method further comprises:

performing semantic verification on the structured query language statement.

8. The method according to any one of claims 1 to 7, further comprising:  
obtaining the structured query language statement from a client input.

9. An apparatus for implementing a computing service based on a structured query language statement, comprising:

a syntax parsing unit, configured to perform syntax parsing on a structured query language statement to determine whether an extended syntax identifier exists in the structured query language statement, wherein the extended syntax identifier indicates a target computing service for the structured query language statement;

a computing service description statement generation unit, configured to generate a computing service description statement in a first statement format based on the structured query language statement when the extended syntax identifier exists in the structured query language statement, wherein the first statement format is a statement format that can be recognized by a target

computing framework; and

a description statement submission unit, configured to submit the generated computing service description statement to the target computing framework, to invoke, in the target computing framework based on the computing service description statement, data queried by the structured query language statement to perform target computation, wherein the structured query language statement comprises a computing element needed by the target computing service, when the extended syntax identifier exists in the structured query language statement.

10. The apparatus according to claim 9, wherein the target computing service comprises a machine learning computing service or a cluster computing service.

11. The apparatus according to claim 10, wherein the machine learning computing service comprises machine learning model training or machine learning model prediction; and

the computing service description statement generation unit comprises:

a data query code generation module, configured to generate data query program code in the first statement format based on a data query element in the structured query language statement; and

a machine learning code generation module, configured to generate machine learning program code in the first statement format based on a computing service element in the structured query language statement.

12. The apparatus according to claim 11, wherein the computing element comprises a machine learning model name, a machine learning model parameter, a machine learning model feature, a machine learning model label, and a machine learning computation result storage address; and

the machine learning code generation module is configured to:

invoke a machine learning program code template corresponding to the machine learning model name; and

provide the machine learning model parameter, the machine learning model feature, and the machine learning model label to the machine learning program code template, to generate the machine learning program code in the first statement format.

13. The apparatus according to claim 9, further comprising:

a semantic verification unit, configured to perform semantic verification on the structured query language statement before the computing service description statement in the first statement format is generated based on the structured query language statement.

14. The apparatus according to any one of claims 9 to 13, further comprising:  
a statement acquisition unit, configured to obtain the structured query language statement from a client input.
15. A system for implementing a computing service based on a structured query language statement, comprising:  
a client device, configured to input a structured query language statement;  
the apparatus for implementing a computing service based on a structured query language statement according to any one of claims 9 to 14;  
a target computing framework; and  
a structured query language database.
16. A computing device, comprising:  
at least one processor; and  
a storage device coupled to the at least one processor, wherein the storage device stores an instruction, and when the instruction is executed by the at least one processor, the at least one processor is enabled to perform the method according to any one of claims 1 to 8.
17. A non-transitory machine-readable storage medium, wherein the non-transitory machine-readable storage medium stores an executable instruction, and when the instruction is executed, the machine is enabled to perform the method according to any one of claims 1 to 8.

1/5

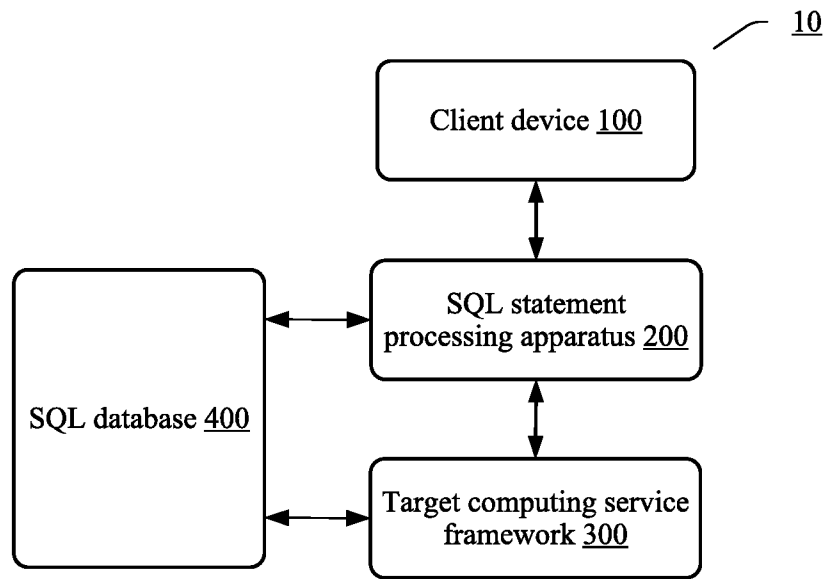


FIG. 1

2/5

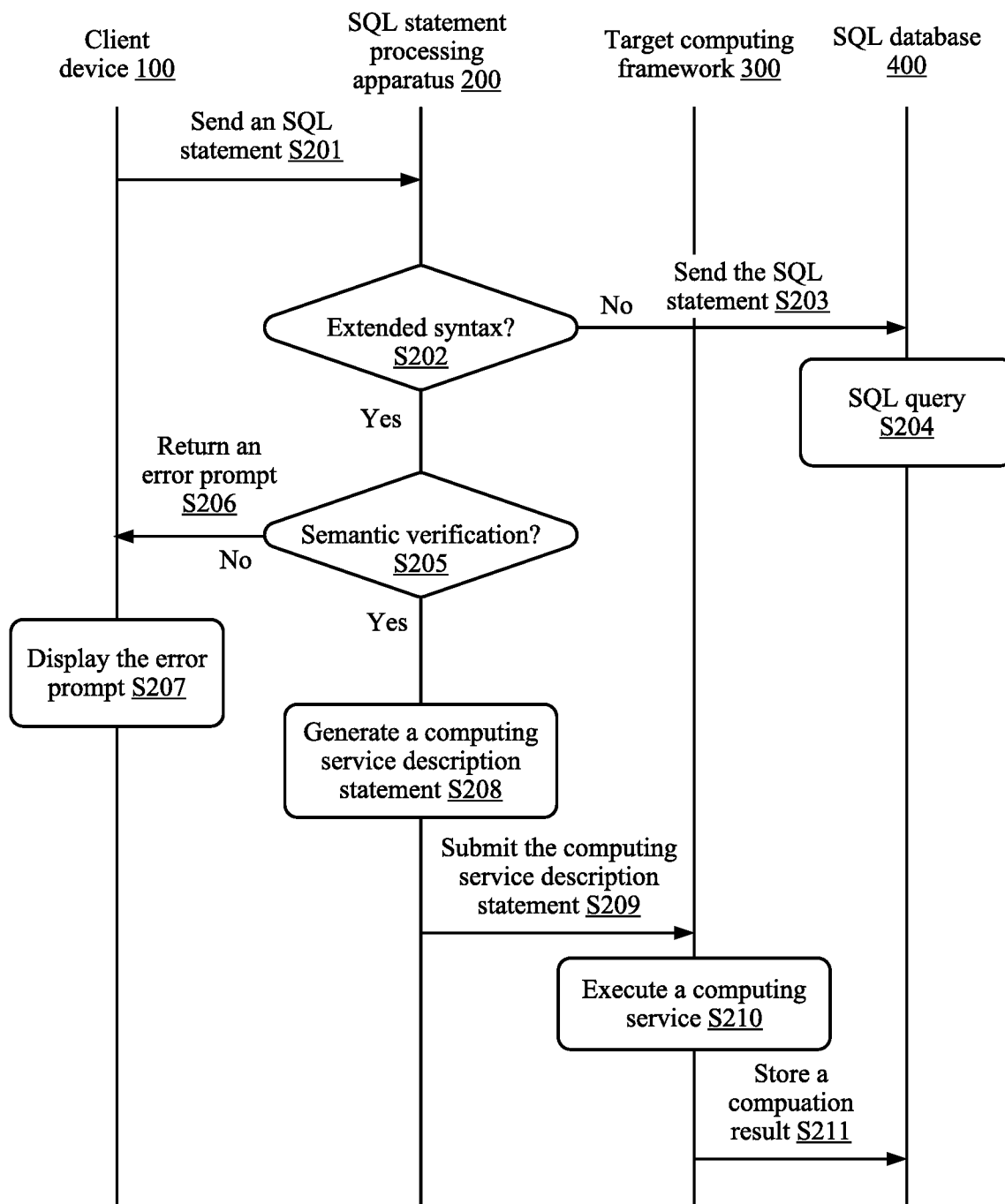


FIG. 2

3/5

|                |              |
|----------------|--------------|
| SELECT *       | Standard SQL |
| FROM iris.iris | Standard SQL |

FIG. 3A

|                                                             |                                                       |
|-------------------------------------------------------------|-------------------------------------------------------|
| SELECT *                                                    | Standard SQL                                          |
| FROM iris.iris                                              | Standard SQL                                          |
| TRAIN DNNClassifier                                         | Extended syntax identifier and machine learning model |
| WITH n_classes = 3, hidden_units = [10, 20]                 | Model parameter                                       |
| COLUMN sepal_length, sepal_width, petal_length, petal_width | Model feature                                         |
| LABEL class                                                 | Model label                                           |
| INTO my_dnn_model                                           | Training result storage                               |

FIG. 3B

|                            |                                                  |
|----------------------------|--------------------------------------------------|
| SELECT *                   | Standard SQL                                     |
| FROM iris.iris             | Standard SQL                                     |
| PREDICT iris.predict.class | Extended syntax identifier and prediction target |
| USING my_dnn_model         | Used model address                               |

FIG. 3C

4/5

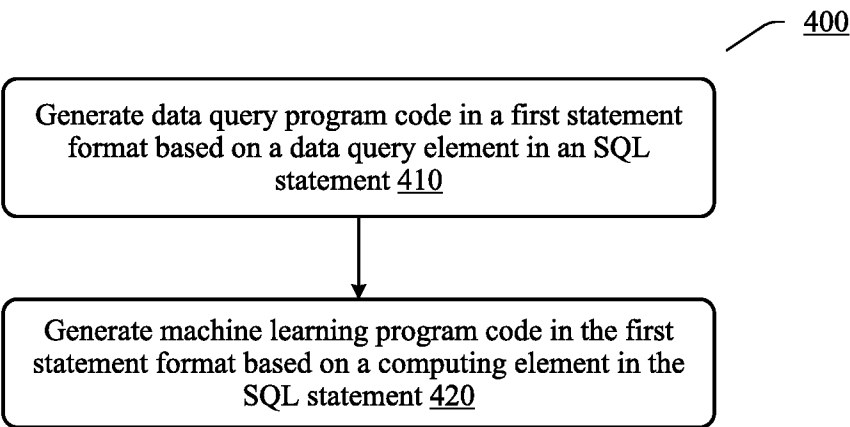


FIG. 4

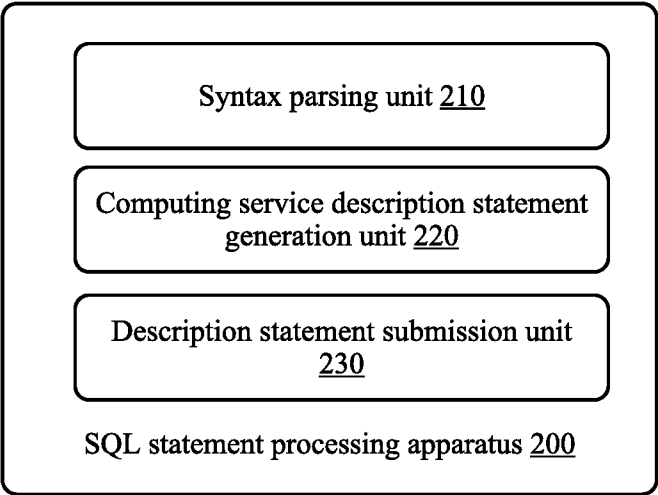


FIG. 5

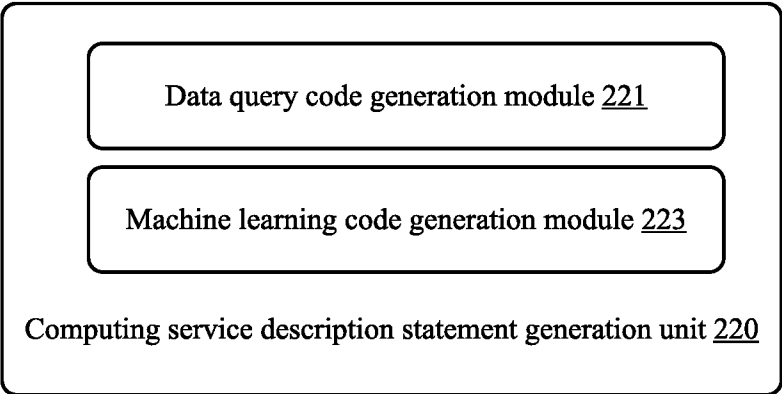


FIG. 6

5/5

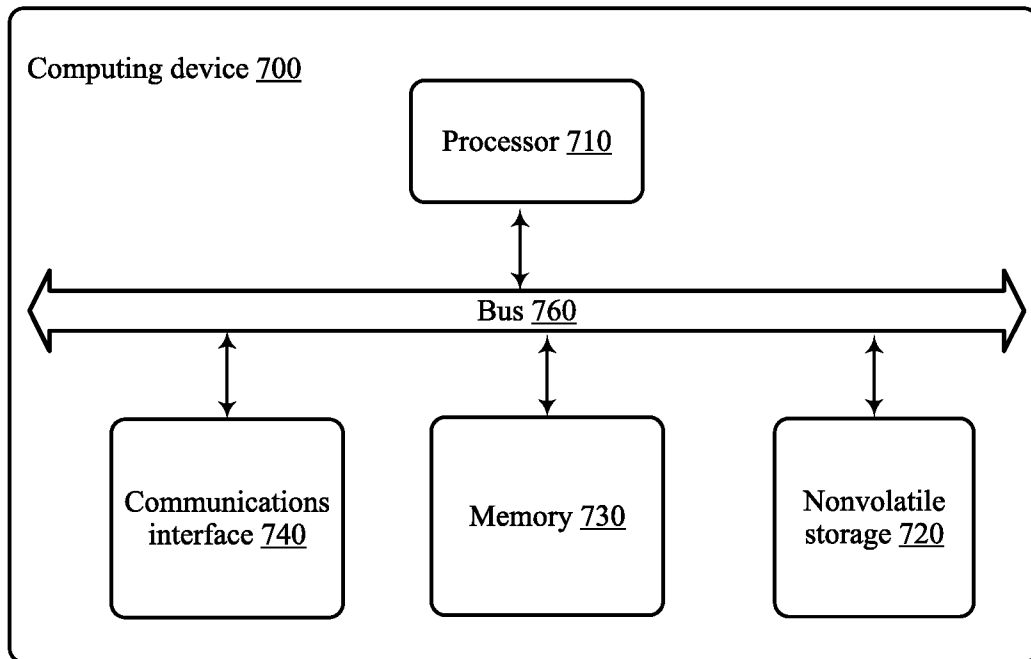


FIG. 7

## INTERNATIONAL SEARCH REPORT

International application No  
PCT/US2020/016719

A. CLASSIFICATION OF SUBJECT MATTER  
INV. G06F16/242  
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)  
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages                      | Relevant to claim No. |
|-----------|---------------------------------------------------------------------------------------------------------|-----------------------|
| X         | US 2017/103106 A1 (KASS ERIC [DE])<br>13 April 2017 (2017-04-13)<br>claim 1; figure 2                   | 1-17                  |
| X         | US 2017/262501 A1 (NAMBOODIRI VIPIN [IN]<br>ET AL) 14 September 2017 (2017-09-14)<br>claim 1; figure 6A | 1-17                  |



Further documents are listed in the continuation of Box C.



See patent family annex.

## \* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

4 June 2020

Date of mailing of the international search report

16/06/2020

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040,  
Fax: (+31-70) 340-3016

Authorized officer

May, Michael

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2020/016719

| Patent document<br>cited in search report | Publication<br>date | Patent family<br>member(s) | Publication<br>date |
|-------------------------------------------|---------------------|----------------------------|---------------------|
| US 2017103106                             | A1                  | 13-04-2017                 | NONE                |
| US 2017262501                             | A1                  | 14-09-2017                 | NONE                |