

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6213247号
(P6213247)

(45) 発行日 平成29年10月18日 (2017.10.18)

(24) 登録日 平成29年9月29日 (2017.9.29)

(51) Int.Cl.	F I
G06F 17/30 (2006.01)	G06F 17/30 330B
G06F 12/00 (2006.01)	G06F 17/30 180D
	G06F 17/30 419B
	G06F 12/00 513J

請求項の数 15 (全 21 頁)

(21) 出願番号	特願2014-4205 (P2014-4205)	(73) 特許権者	000005223
(22) 出願日	平成26年1月14日 (2014.1.14)		富士通株式会社
(65) 公開番号	特開2014-137820 (P2014-137820A)		神奈川県川崎市中原区上小田中4丁目1番1号
(43) 公開日	平成26年7月28日 (2014.7.28)	(74) 代理人	100107766
審査請求日	平成28年9月5日 (2016.9.5)		弁理士 伊東 忠重
(31) 優先権主張番号	13151328.5	(74) 代理人	100070150
(32) 優先日	平成25年1月15日 (2013.1.15)		弁理士 伊東 忠彦
(33) 優先権主張国	欧州特許庁 (EP)	(74) 代理人	100146776
			弁理士 山口 昭則
		(72) 発明者	リー・ヴィヴィアン
			イギリス国、アールジー42 2キュービー、ブラックネル パークシャー、ハウエル クローズ、4番

最終頁に続く

(54) 【発明の名称】 データ記憶システム、データ記憶システムで実行されるプログラム及び方法

(57) 【特許請求の範囲】

【請求項 1】

複数の見出し付き列の各々にエントリを有する関係型データの行を有する関係型データベースを格納するよう構成される関係型データベース記憶装置と、

前記関係型データに関連するグラフデータを有するグラフデータベースを格納するよう構成されるグラフデータベース記憶装置であって、前記グラフデータベースはトリプルとしてエンコードされる、グラフデータベース記憶装置と、

関係型データの1又は複数の行のサブセットを定める条件を指定する関係型データクエリを受信するよう構成されるクエリハンドラであって、前記関係型データの1又は複数の行からのエントリが検索結果に含まれる、クエリハンドラと、

を有し、前記クエリハンドラは、関係型データベースの1又は複数の行の前記サブセットに関連するグラフデータについて前記グラフデータベースを検索するためにグラフデータクエリを生成するよう構成される、データ記憶システム。

【請求項 2】

前記関係型データベース内の各エントリは、前記グラフデータベース内のノードに対応し、前記関係型データベース内のエントリに関連するデータは、前記グラフデータベースにおいて、前記エントリに対応するノードにリンク付けされたノードで表される、請求項1に記載のデータ記憶システム。

【請求項 3】

前記関係型データベース内の関係型データの行は、前記見出し付き列のうち主要列にあ

る前記行の個々のエントリの値に従ってインデックス付けされ、

前記グラフデータベース内のグラフデータは、オントロジに従って構造化され、該オントロジでは、主要列リソースは前記主要列に対応し、前記主要列リソースの述語は他の見出し付き列の各々に対応する更なる列リソースへのリンクを示す、

請求項 1 又は 2 に記載のデータ記憶システム。

【請求項 4】

前記主要列リソース及び前記更なる列リソースの各々は、対応する見出し付き列の値を表す目的語へのリンクを示す値の述語を有する、請求項 3 に記載のデータ記憶システム。

【請求項 5】

前記主要列リソースのインスタンス及び前記更なる列リソースのインスタンスは、テーブルエントリに対応し、前記テーブルエントリの値は、個々の値の述語によりリンク付けされた目的語に格納される、請求項 4 に記載のデータ記憶システム。

10

【請求項 6】

前記主要列の名称を前記主要列として識別される形式で、前記主要列の名称と関連付けて、及び前記更なる見出し付き列の各々について、前記更なる見出し付き列の名称に対応する列リソースの名称と関連付けて、格納するよう構成されるマッピングユニット、

を更に有し、

前記クエリハンドラは、前記マッピングユニットを参照して前記グラフデータクエリを生成するよう構成される、請求項 3 乃至 5 のいずれか一項に記載のデータ記憶システム。

【請求項 7】

20

前記関係型データクエリは S Q L ステートメントであり、前記グラフデータクエリは S P A R Q L クエリである、請求項 1 乃至 6 のいずれか一項に記載のデータ記憶システム。

【請求項 8】

前記関係型データベースのデータベースドライバを更に有し、

前記クエリハンドラは、前記データベースドライバのコンポーネントとして設けられる、請求項 1 乃至 7 のいずれか一項に記載のデータ記憶システム。

【請求項 9】

前記関係型データクエリは、前記見出し付き列のうちのどれからエントリが前記検索結果に含まれるべきかを指定する、請求項 1 乃至 8 のいずれか一項に記載のデータ記憶システム。

30

【請求項 10】

前記クエリハンドラは、前記関係型データクエリを受信すると、前記マッピングユニットを参照して、前記関係型データベースの前記主要列の識別子を得て、前記関係型データクエリの前記条件を満たす前記関係型データベースの行の主要列エントリの値を得るよう構成され、

前記クエリハンドラは、前記主要列エントリの得られた値と一致する値を有する前記主要列リソースのインスタンスのサブセットにリンク付けられたノードの値を要求するグラフデータクエリを生成するよう構成される、請求項 6 に記載のデータ記憶システム。

【請求項 11】

前記クエリハンドラは、前記マッピングユニットを参照して、前記関係型データクエリで指定された前見出し付き列のエントリに対応する前記更なる列リソースを識別し、

40

前記クエリハンドラは、前記グラフデータクエリに、前記主要列リソースのインスタンスのサブセットにリンク付けられた前記の識別された列リソースのインスタンスであるノードの値についての要求を含め、前記グラフデータクエリに、前記ノードを主語として有する全てのトリプルについての要求を含めるよう構成される、

請求項 10 に記載のデータ記憶システム。

【請求項 12】

前記クエリハンドラは、前記主要列のエントリを要求する更なる関係型データクエリを生成し、前記関係型データクエリと同一の条件を指定し、前記更なる関係型データクエリを前記関係型データベースに発行し、結果を受信することにより、前記関係型データクエ

50

りの前記条件を満たす前記関係型データベースの行の主要列のエントリの値を得るよう構成される、

請求項 10 又は 11 に記載のデータ記憶システム。

【請求項 13】

データ記憶システムで実行される方法であって、前記データ記憶システムは、複数の見出し付き列の各々にエントリを有する関係型データの行を有する関係型データベースを格納するよう構成される関係型データベース記憶装置と、

前記関係型データに関連するグラフデータを有するグラフデータベースを格納するよう構成されるグラフデータベース記憶装置であって、前記グラフデータベースはトリプルとしてエンコードされる、グラフデータベース記憶装置と、を有し、前記方法は、クエリハンドラで、関係型データの 1 又は複数の行のサブセットを定める条件を指定する関係型データクエリを受信するステップであって、前記関係型データの 1 又は複数の行からのエントリが検索結果に含まれる、ステップと、関係型データベースの 1 又は複数の行の前記サブセットに関連するグラフデータについて前記グラフデータベースを検索するためにグラフデータクエリを生成するステップと、を有する方法。

【請求項 14】

コンピューティング装置により実行されると、前記コンピューティング装置に請求項 13 に記載の方法を実行させる、コンピュータプログラム。

【請求項 15】

分散型コンピューティング環境にあるコンピューティング装置により実行されると、前記コンピューティング装置を、請求項 1 乃至 12 のいずれか一項に記載のデータ記憶システムとして動作させる、コンピュータプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、データ記憶システム及びクエリ技術の分野に関する。特に、本発明は、データクエリを 1 より多いクエリフォーマットに変換し、したがって異なる基本データフォーマットを有するデータベースが単一クエリにより問い合わせ可能な分野に関する。

【背景技術】

【0002】

データベースユーザ又は管理者は、新しい欄を追加することにより又は特定の他の方法でスキーマを変更することにより、既存の関係型データベースに格納されるデータの範囲を拡張したいと望み得る。しかしながら、これを行うことは、広範囲に渡りスキーマのルールに柔軟性がないために、実際には不可能か又は非常に困難である。したがって、RDF トリプルストアのようなグラフデータベースが、関係型データベースに格納されるデータを拡張するための手段として採用されてきた。グラフデータベースは、関係型データベースの基本スキーマを変更することなく、関係型データベースに格納されるデータに関連するデータを格納することを可能にする。

【0003】

特に、RDF トリプルとしてエンコードされたデータを有するグラフデータベースは、既存の関係型データベースの拡張としてデータベース管理者により用いられ得る。RDF (Resource Description Framework) のデータモデルの仕様及び XML シンタックスは、1999 年の W3C により発行された。依然として比較的新しい概念だが、主語 (subject) - 述語 (predicate) - 目的語 (object) を有するデータエントリを格納する RDF トリプルストアは、柔軟なデータ構造のために近年次第に普及してきた。良い適用例は、大部分のデータが IT システム、例えばアカウントティング、CRM、HR に格納される企業環境におけるデータ統合である。これらのシステムは、通常、RDBMS (Relational Database Management System) で構築される。しかしながら、機能的 IT システムではなく内部ネットワークにあるデータ、例えば MS Word、MS Excel、HTML 又は PDF フォーマットの文書、例えば「検索できない」データ又は公共ライブラリ

10

20

30

40

50

に格納されたデータも存在する。これらのデータは、機能的ＩＴシステムに格納されたデータに関連するが、異なるフォーマット及び／又は場所にある。

【０００４】

ＲＤＦトリプルストアは、既存のＩＴシステム内のデータと密接な関係を有するこれらの検索できないデータのために最適な解決策を提供する。

【発明の概要】

【発明が解決しようとする課題】

【０００５】

しかしながらグラフデータベースに格納されたデータは、関係型データベースのためのクエリフォーマットによっては検索できない。

【課題を解決するための手段】

【０００６】

本発明の実施形態は、複数の見出し付き列の各々にエントリを有する関係型データの行を有する関係型データベースを格納するよう構成される関係型データベース記憶装置と、前記関係型データに関連するグラフデータを有するグラフデータベースを格納するよう構成されるグラフデータベース記憶装置であって、前記グラフデータベースはトリプルとしてエンコードされる、グラフデータベース記憶装置と、関係型データの１又は複数の行のサブセットを定める条件を指定する関係型データクエリを受信するよう構成されるクエリハンドラであって、前記関係型データの１又は複数の行からのエントリが検索結果に含まれる、クエリハンドラと、を有し、前記クエリハンドラは、関係型データベースの１又は複数の行の前記サブセットに関連するグラフデータについて前記グラフデータベースを検索するためにグラフデータクエリを生成するよう構成される、データ記憶システムを含む。

【０００７】

有利なことに、本発明の実施形態は、受信したデータベースクエリと互換性のないデータフォーマットを有するデータベースに格納されたデータを検索するのに適するフォーマットで追加データクエリを生成することにより、第１のデータベースを検索する受信したデータベースクエリの範囲を自動的に拡張するメカニズムを提供する。したがって、追加データはクエリで検索でき、検索結果は、第１のデータベースを単独で検索する受信したデータベースクエリにより達成されるものよりも包括的である。したがって、検索結果に依存するアプリケーション又は意志決定プロセスは、より情報に基づく意志決定を行うことができる。

【０００８】

特に、本発明の実施形態は、関係型データベースから情報を得るために構造化されたクエリを、グラフデータベースからの関係型データベースの結果に関連する情報を得るために拡張できるようにする。これは、関係型データベースの基本データベーススキーマを変更することなく、関係型データベースに格納されたデータに関連するデータを、グラフデータベースに格納できるという利点を有する。これは、性能の利益をもたらす。さらに、関係型データベース宛てのクエリに応答して、関連データが自動的に検索されるという利点を有する。

【０００９】

まとめると、本発明の実施形態は、関係型データベースを検索することにより、本発明の実施形態がなければ検索不可能なデータを検索するメカニズムを提供する。ここで、データは、関係型データベース内のデータに関連付けられる。このメカニズムは、関係型データベースに適するフォーマットのクエリを調べ、対応するクエリを生成し、対応するクエリをグラフデータベース（つまり、基本トリプルストア）へ実行のために転送する。関係型データベースクエリとグラフデータベースクエリの両者の結果は、クエリ発信元へ返される。したがって、本発明がなければ検索不可能な、より豊富な且つより詳細な情報にアクセスできる。

【００１０】

任意的に、本発明の実施形態では、前記関係型データベース内の各エントリは、前記グラフデータベース内のノードに対応し、前記関係型データベース内のエントリに関連するデータは、前記グラフデータベースにおいて、前記エントリに対応するノードにリンク付けされたノードで表される。

【0011】

有利なことに、請求項2に定めるグラフデータベースオントロジは、グラフデータベースに格納されるが関係型データベースに格納されない追加情報を作成する基礎を提供する。つまり、グラフデータのトリプルとしてのエンコーディングは、クエリハンドラが、グラフデータベースから関連情報を見付けるために関係型データベースエントリに対応するノードからの述語リンクに従うグラフデータクエリを構成できることを意味する。グラフデータベースにおいてノードで表される関連データは、情報種類を定めるノード及び情報種類の値を格納するリンク付けられたノードを含む。トリプルの観点から、クエリハンドラは、関係型データベース内のエントリとグラフデータベース内のノードとの間の対応を確立するための十分な情報を与えられた場合、主語としてのエントリ及びその値に対応するノードを有するグラフデータベース内のトリプルを単に検索できる。

10

【0012】

更なる任意的な特徴として、前記関係型データベース内の関係型データの行は、前記見出し付き列のうち主要列にある前記行の個々のエントリの値に従ってインデックス付けされ、前記グラフデータベース内のグラフデータは、オントロジに従って構造化され、該オントロジでは、主要列リソースは前記主要列に対応し、前記主要列リソースの述語は他の見出し付き列の各々に対応する更なる列リソースへのリンクを示す。

20

【0013】

有利なことに、主要列は、関係型データベース内の各行にユニークな識別子を設ける。したがって、主要列に対応するリソースが全ての関連情報の階層構造のルートであるオントロジを実装することにより、クエリハンドラは、関係型データベース内の行の選択されたサブセットに関連するデータを返すグラフデータクエリを生成できる。

【0014】

さらに、前記主要列リソース及び前記更なる列リソースの各々は、対応する見出し付き列の値を表す目的語へのリンクを示す値の述語を有しても良い。

【0015】

30

さらに、このような実施形態では、前記主要列リソースのインスタンス及び前記更なる列リソースのインスタンスは、テーブルエントリに対応し、前記テーブルエントリの値は、個々の値の述語によりリンク付けされた目的語に格納される。

【0016】

有利なことに、テーブルエントリに対応するノードを、値の述語を有するリソースのインスタンスとして有することにより、更なる関連情報片が、ノードにリンクする述語 - 目的語ペアとして格納される。単にテーブルエントリを見出し付き列に対応する述語の目的語として有することは、グラフデータベースに格納されるべき更なる関連情報を提供しない。

【0017】

40

本発明の実施形態は、前記主要列の名称を前記主要列として識別される形式で、前記主要列の名称と関連付けて、及び前記更なる見出し付き列の各々について、前記更なる見出し付き列の名称に対応する列リソースの名称と関連付けて、格納するよう構成されるマッピングユニット、を更に有し、前記クエリハンドラは、前記マッピングユニットを参照して前記グラフデータクエリを生成するよう構成されても良い。

【0018】

有利なことに、マッピングユニットは、特定の実装では、クエリハンドラがグラフデータクエリを構成することを可能にする情報を格納する。特定の実施形態では、マッピングユニットは、関係型データベースの列見出しとグラフデータベースのリソース名との間の関連を格納する。したがって、関係型データクエリ内で指定された関係型データベースの

50

列見出しは、グラフデータベース内のリソース名に対応する。これらの関係型データベースの列見出しのエントリに関連するデータは、対応するリソース名のインスタンスを主語として有するトリプルに格納される。したがって、クエリハンドラに关系型データクエリ内で指定された列見出しに対応するリソース名を知らせることにより、クエリハンドラは、グラフデータベース内の関連データを検索するためのグラフデータクエリを構成できる。マッピングユニットは、グラフデータクエリをアドレスすべき場所をクエリハンドラに知らせるために、関係型データベースに関連するグラフデータベースの場所を表す情報も格納しても良い。

【0019】

マッピングユニットは、マッピングファイルを格納するメモリ内の記憶場所、又はマッピング情報の他の格納された形式であっても良い。マッピングユニットは、クエリハンドラ自体の一部として、関係型データベース又はグラフデータベースのための管理システム/サービスの一部として、又は外部コンポーネントとして設けられても良い。

【0020】

関係型データクエリは、RDBMSを介して関係型データベースからデータを検索し及び読み出すのに適するフォーマットであり、データが返される行を定める条件と、検索結果で返されるべきエントリの値を有する見出し付き列の列見出しの指標との一方及び両方を有しても良い。このようなクエリの例示的なフォーマットは、SQLステートメントである。グラフデータクエリは、グラフデータベースからデータを検索し及び読み出すのに適する任意のフォーマットであり得る。さらに、追加データはグラフデータクエリを介して検索されるので、指定された主語に関連するプロパティ又は他のデータを格納する任意のトリプルが検索されるある種のワイルドカード、また指定された主語に関連するトリプルに関連するトリプルを有することは、クエリにとって有利である。このような例示的なフォーマットのクエリは、1又は複数のSPARQLクエリ、場合によっては入れ子状のSPARQLクエリである。つまり、本発明の実施形態では、関係型データクエリはSQLステートメントであり、グラフデータクエリはSPARQLクエリである。さらに、関係型データクエリは、どの見出し付き列からのエントリが検索結果に含まれるべきかを指定しても良い。

【0021】

クエリハンドラは、単独のプログラム又は装置であっても良い。或いは、クエリハンドラは、データベースドライバのコンポーネントとして設けられても良い。有利なことに、関係型データベースのデータベースドライバ内にクエリハンドラを例えばプラグインプログラムとして埋め込むことは、クエリハンドラにより実行される手順が可能な限りクエリ発信元に透過であることを保証する。データベースドライバは、例えばJDBC又はODBCであっても良い。グラフデータクエリを生成し実行する処理は、データベースドライバで関係型データクエリが受信されるのに応答して自動的に実行され、検索結果はデータベースドライバで結合される。したがって、クエリ発信元が実行されている処理を知る唯一の方法は、追加検索結果の受信による。したがって、本発明は、関係型データベース管理システム (relational databasemanagement system: RDBMS) ドライバにインストールされるプラグインプログラムとして具現化されても良い。プラグインプログラムは、関係型データクエリ (例えば、SQLステートメント) がRDBMSに発行されるときを検出し、(必要な場合) マッピングユニットから情報を読み出し、グラフデータクエリを生成し、グラフデータクエリをグラフデータベースに発行する。

【0022】

本発明の実施形態では、関係型データクエリは、どの見出し付き列からのエントリが検索結果に含まれるべきかを指定しても良い。有利なことに、このような実施形態は、クエリを特定の列に集中させ、他の行ではなく1つの行の中の幾つかのエントリの値を返すことを可能にする。

【0023】

任意で、前記クエリハンドラは、前記関係型データクエリを受信すると、前記マッピン

10

20

30

40

50

グユニットを参照して、前記関係型データベースの前記主要列の識別子を得て、前記関係型データクエリの前記条件を満たす前記関係型データベースの行の主要列エントリの値を得よう構成され、前記クエリハンドラは、前記主要列エントリの得られた値と一致する値を有する前記主要列リソースのインスタンスのサブセットにリンク付けられたノードの値を要求するグラフデータクエリを生成するよう構成される。

【0024】

本発明の実施形態では、主要列は、グラフデータベースにおいて、データベースの行及びその関連データに対応する階層構造の最上位として表される。したがって、主要列の中のエントリの値が分かっている限り、データベース内の特定の行に関連するデータは容易に検索でき、関連データの値が返される。

10

【0025】

実際に、データトラフィックを削減するために、グラフデータクエリは、関係型データベース内の特定の列に対応するグラフデータ及びそれに関連するデータに集中できる。つまり、前記クエリハンドラは、前記マッピングユニットを参照して、前記関係型データクエリで指定された前見出し付き列のエントリに対応する前記更なる列リソースを識別し、前記クエリハンドラは、前記グラフデータクエリに、前記主要列リソースのインスタンスのサブセットにリンク付けられた前記の識別された列リソースのインスタンスであるノードの値についての要求を含め、前記グラフデータクエリに、前記ノードを主語として有する全てのトリプルについての要求を含めるよう構成されても良い。

【0026】

20

関連データがグラフデータベース内で検索されるデータ行の主要列にあるエントリの値を得るための特定のメカニズムとして、クエリハンドラは、主要列エントリを要求し関係型データクエリと同じ条件を指定する更なる関係型データクエリを生成し、更なる関係型データクエリを関係型データベースに発行し、結果を受信することにより、関係型データクエリの条件を満たす関係型データベースの行の主要列エントリの値を得よう構成されても良い。

【0027】

本発明の別の態様の実施形態では、データ記憶システムで実行される方法を提供する。前記データ記憶システムは、複数の見出し付き列の各々にエントリを有する関係型データの行を有する関係型データベースを格納するよう構成される関係型データベース記憶装置と、前記関係型データに関連するグラフデータを有するグラフデータベースを格納するよう構成されるグラフデータベース記憶装置と、を有し、前記グラフデータベースはトリプルとしてエンコードされる。前記方法は、クエリハンドラで、関係型データの1又は複数の行のサブセットを定める条件を指定する関係型データクエリを受信するステップであって、前記1又は複数の行からのエントリが前記検索結果に含まれる、ステップと、関係型データの1又は複数の行の前記サブセットに関連するグラフデータについて前記グラフデータベースを検査するためのグラフデータクエリを生成するステップと、を有する。

30

【0028】

前記方法は、1又は複数の以下のステップを更に有しても良い。

【0029】

40

(前記クエリハンドラで)前記関係型データベースに前記関係型データクエリを発行し、結果を受信するステップ。

【0030】

(前記クエリハンドラで)前記グラフデータベースに前記グラフデータクエリを発行し、結果を受信するステップ。

【0031】

(前記クエリハンドラで)前記グラフデータクエリ及び前記関係型データクエリからの結果を収集するステップ。

【0032】

(前記クエリハンドラで)前記の受信した関係型データクエリに対して、前記の収集し

50

た結果で応答するステップ。

【0033】

本発明の別の態様の実施形態は、コンピューティング装置により実行されると、前記コンピューティング装置に本発明を実施する方法を実行させる、コンピュータプログラムを有する。

【0034】

本発明の別の態様の実施形態は、分散型コンピューティング環境内のコンピューティング装置により実行されると、前記コンピューティング装置を本発明を実施するデータ記憶システムとして機能させる、コンピュータプログラムスーツを有する。

【0035】

本発明の別の態様の実施形態は、コンピュータ又は分散型コンピュータネットワークにより実行されると、前記コンピュータ又は前記分散型コンピュータネットワークを本発明を実施するデータ記憶システムにする（又は該システムとして機能させる）ソフトウェアを有する。分散型コンピュータネットワークは、分散型であっても良い1又は複数の記憶ユニットを有しても良い。ソフトウェアは、非一時的記憶媒体で提供され得るコンピュータプログラム又はコンピュータプログラムスーツであっても良い。

【0036】

態様（ソフトウェア／方法／装置）が別個に議論されたが、1つの態様に関連して議論されたその特徴及び影響は、他の態様にも等しく適用できる。したがって、方法の特徴が議論される場合、装置の実施形態はその特徴を実行する又は適切な機能を提供するよう構成されるユニット又は装置を有すること、及びプログラムは該プログラムが実行されるコンピューティング装置に前記方法の特徴を実行させるものと解釈される。

【0037】

上述の態様のいずれにおいても、種々の特徴は、ハードウェアで、又は1又は複数のプロセッサで動作するソフトウェアモジュールとして実装されても良い。一態様の特徴は、他の態様のいずれにも適用できる。

【0038】

本発明は、上述の任意の方法を実行するコンピュータプログラム又はコンピュータプログラムプロダクト、及び上述の任意の方法を実行するプログラムを格納しているコンピュータ可読媒体も提供する。

【0039】

本発明を実施するコンピュータ・プログラムは、コンピュータ可読媒体に格納されてもよい。或いは、例えば、インターネット・ウェブサイトから提供されるダウンロード可能なデータ信号のような信号形式又は任意の他の形式であってもよい。

【0040】

本発明の好適な特徴は、単なる例として添付の図面を参照して以下に説明される。

【図面の簡単な説明】

【0041】

【図1】本発明を具現化するシステムの概略図である。

【図2】本発明の実施形態の例示的なコンポーネントアーキテクチャである。

【図3】どのようにマッピングファイルがコンパイルされるか、及びマッピングファイルに格納される情報を示す。

【図4】本発明の実施形態における関係型データベースを示す。

【図5】本発明の実施形態におけるRDFデータセットを示す。

【図6】本発明を実施するプラグインプログラムにより実行される処理の例を示す。

【図7】A乃至Cは、本発明を実施するクエリハンドラにより生成される例示的なSPARQLステートメントを示す。

【図8】本発明を具現化するシステムのシステムアーキテクチャを示す。

【発明を実施するための形態】

【0042】

図 1 は、本発明を具現化するシステムの概略図である。クライアントアプリケーション 20 は、データコネクションによりクエリハンドラ 10 に接続される。クエリは、クライアントアプリケーション 20 からクエリハンドラ 10 へ提出され、検索結果は、クエリに対する応答により返される。クエリハンドラ 10 は、関係型データベース記憶装置 12 及びグラフデータベース記憶装置 14 に、データ通信コネクションにより接続される。関係型データベース記憶装置 12 に格納された関係型データベースを検索するのに適する第 1 のフォーマットのクエリは、クエリハンドラ 10 から関係型データベース記憶装置 12 へ提出され、クエリに応答して、関係型データベース記憶装置 12 から（又は関係型データベースを制御するデータベースマネージャのような機能ユニットから）検索結果が返される。グラフデータベース記憶装置 14 に格納されたグラフデータベースを検索するのに適する第 2 のフォーマットのクエリは、クエリハンドラ 10 からグラフデータベース記憶装置 14 へ提出され、クエリに応答して、グラフデータベース記憶ユニット 14 から（又はグラフデータベースを制御するデータベースマネージャのような機能ユニットから）検索結果が返される。クライアントアプリケーション 20 によりクエリハンドラ 10 へ提出されるクエリは、関係型データベースを検索するのに適したフォーマットである。

10

【0043】

クライアントアプリケーションは、関係型データクエリのソースの一例である。この状況では、関係型データクエリのソースは、ソースの形式に関係なく、「クライアントアプリケーション」であると考えられる。

【0044】

20

クエリハンドラ 10 は、ソフトウェア又はハードウェアの形式で提供され得る機能ユニットである。クエリハンドラ 10 は、データベース管理システムの一部として設けられても良い。例えば、クエリハンドラ 10 は、関係型データベース記憶装置 12 に格納される関係型データベースの管理システムの一部として設けられても良い。したがって、クエリハンドラ 10 は、関係型データベースを格納するサーバ又は相互接続されたサーバ群にインストールされる。或いは、クエリハンドラ 10 は、関係型データベースのデータベースドライバのコンポーネントであっても良い。このドライバは、関係型データベースの管理システムと相互作用するよう構成される。代替で、クエリハンドラ 10 は、関係型データベースのデータベースドライバと別個に設けられ、データベースドライバへ向けられる関係型データベース記憶装置 12 に格納された関係型データベースを検索するための関係型データクエリを傍受するよう構成されても良い。クエリハンドラは、関係型データベースを検索するのに適するフォーマットの関係型データクエリを受信し、関係型データクエリを関係型データベースを検索するのに適するフォーマットの 1 又は複数の関係型データクエリ及びグラフデータベースを検索するのに適するフォーマットの 1 又は複数のグラフデータクエリに変換するよう構成される。例えば、グラフデータは、RDF トリプルのようなトリプルとしてエンコードされ、1 又は複数のグラフデータクエリが RDF トリプルストアを検索するのに適するフォーマットになるようにする。

30

【0045】

クエリハンドラ 10 は、関係型データクエリ及びグラフデータクエリからの検索結果を受信し統合する機能も提供しても良い。

40

【0046】

関係型データベース記憶装置 12 は、複数の見出し付き列の各々の中にエントリを有する何行もの関係型データを有する関係型データベースを格納するよう構成される装置である。関係型データフォーマットは、例えば SQL (Structured Query Language) ステートメントにより検索できる確立されたデータ記憶の形式である。関係型データベース記憶装置 12 は、サーバ又は相互接続されたサーバ群であっても良い。関係型データベース記憶装置 12 は、関係型データベースに格納された関係型データへのアクセス、変更及び記憶を制御するよう構成される関係型データベース管理システム (relational database management system: RDBMS) を格納するか又はそれをインストールしている。関係型データベース記憶装置 12 は、場合によっては RDBMS を介して、関係型データクエリを

50

受信し、それらに検索結果で応答するよう構成されても良い。例えば、関係型データクエリは、条件を指定しても良い。この条件は、関係型データベース内の行の中のデータにより満たされる場合、その行からのデータが検索結果で返されるべきであることを示す。さらに、関係型データクエリは、見出し付き列を指定しても良い。この見出し付き列から、条件を満たす行の中のエントリの値が検索結果に含まれる。

【0047】

グラフデータベース記憶装置14は、トリプル、例えばR D Fトリプルとしてエンコードされたグラフデータを有するグラフデータベースを格納するよう構成される装置である。グラフデータフォーマットは、例えばS P A R Q L (SPARQL Protocol and RDF Query Language) ステートメントにより検索できる確立されたデータ記憶の形式である。グラフデータベース記憶装置14は、サーバ又は相互接続されたサーバ群であっても良い。グラフデータベース記憶装置14は、グラフデータをエンコードするトリプルへのアクセス、変更及び記憶を制御するよう構成されるグラフデータベース管理システム (graph database management system: GDBMS) を格納するか又はそれをインストールしている。グラフデータベース記憶装置14は、場合によってはG D B M Sを介して、グラフデータクエリを受信し、それらに検索結果で応答するよう構成されても良い。例えば、グラフデータクエリは、検索結果として返されるべきR D Fトリプルの範囲を指定しても良い。例えば、グラフデータクエリは特定の主語を指定しても良く、指定された主語を主語として有するトリプルが検索結果に含まれる。

【0048】

データグラフは、グラフデータベース又はグラフデータセットとも表され、意味的知識モデルの表現を提供する。本発明の実施形態のデータ記憶は、データベース、例えばグラフデータベースであっても良い。

【0049】

グラフデータベースは、データをノード及びアークの形式で格納することにより、関係型データベースの重要な拡張を表す。ここで、ノードはエンティティ又はインスタンスを表し、アークは任意の2個のノード間の特定種類の関係を表す。無向グラフでは、ノードAからノードBへのアークは、ノードBからノードAへのアークと同じであると考えられる。有向グラフでは、2つの方向は別のアークとして扱われる。

【0050】

グラフデータベースは、概して2つの主な種類に分類できる広範な種類の異なるアプリケーションで用いられる。第1の種類は、知的意思決定支援及び自己学習のようなクラス記述子の大規模な集合体 (「知識ベースアプリケーション」と称される) を有する複雑な知識ベースシステムを有する。第2の種類は、社会的データ及びビジネスインテリジェンスのようなトランザクションデータに対するグラフ検索の実行を含むアプリケーション (「トランザクションデータアプリケーション」と称される) を有する。多くのアプリケーションは、両方の種類を表し得る。しかしながら、大部分のアプリケーションは、主に知識ベース又はトランザクションデータアプリケーションのいずれかで特徴付けられ得る。グラフデータベースは、種々の分野の膨大な構造化又は非構造化データを格納できる大規模な「意味ネットワーク」を維持するために用いることができる。意味ネットワークは、知識表現の形式として用いられ、コンセプトを表すノード及びコンセプト間の意味関係を表すアークを有する有向グラフである。

【0051】

記憶のためにグラフデータベースをエンコードする幾つかのアプローチがある。グラフデータは、多次元アレイとして又は他のシンボルにリンク付けされたシンボルとしてメモリに格納されても良い。別の形式のエンコーディングは、各々指定された種類のオブジェクトの有限シーケンス又は順序付きリストである「タプル」の使用である。n個のオブジェクトを含むタプルは、「nタプル」として知られる。ここで、nは零より大きい任意の非負整数である。長さ2のタプル (2タプル) は、通常、ペアと呼ばれる。3タプルはトリプルと呼ばれ、4タプルはクワドラプルと呼ばれ、以降同様である。

【 0 0 5 2 】

記載されるエンティティは、トリプルの主語として表されても良い。識別されたプロパティの範囲は、目的語として表されても良い。また、範囲とエンティティとの間の関係は、述語として表されても良い。トリプルは、グラフデータを複数の主語 - 述語 - 目的語表現として特徴付けることにより、グラフデータのエンコーディングを提供する（ここで、グラフデータは、データ記憶に格納されるデータの例である）。この文脈では、主語及び目的語は、グラフデータのグラフノードであり、オブジェクト、インスタンス又はコンセプト（集合的に「グラフリソース」）のようなエンティティであり、述語は、主語と目的語の間の関係の表現である。述語は、目的語への特定の種類のリンクを提供することにより、主語に関する何かを断言する。例えば、主語は、（例えば、URIを介して）ウェブリソースを示しても良く、述語はリソースの個々の特性、特徴又は状況を示し、目的語は、該特性、特徴又は状況のインスタンス、範囲、又は例を示す。言い換えると、トリプルステートメントの集合は、元来、方向性グラフデータを表す。RDF標準は、このようなトリプルの形式化された構造を定める。

10

【 0 0 5 3 】

グラフデータは、階層的方法で解釈され得る。例えば、主語ノードは、階層構造において目的語ノードより上にあると考えられる。

【 0 0 5 4 】

関係型データベースは、データを行と列で格納する。行及び列は、データを格納する前に定める必要のあるテーブルを構成する。テーブルの定義及びこれらのテーブルに含まれるデータ間の関係は、スキーマと称される。関係型データベースは、固定スキーマを用いる。

20

【 0 0 5 5 】

クライアントアプリケーション20は、関係型データベース記憶装置12に格納される関係型データベースにクエリを発行する。クエリは、関係型データを検索するのに適するフォーマット、例えばSQLステートメントであり、したがって関係型データクエリと称される。クエリハンドラ10は、関係型データベースのためのデータベースドライバ内のプラグインの形式で提供されても良く、クライアントアプリケーション20から関係型データクエリを受信する。関係型データクエリは、少なくとも条件を指定する。この条件は、関係型データベース内のデータの行により満たされる場合、その行からのデータが検索結果に含まれるべきであることを示す。つまり、条件を満たす行内の見出し付き列内のエントリの値が、関係型データベースから読み出され、クエリハンドラ10に検索結果として提供される。関係型データクエリは、どの見出し付き列からエントリの値が読み出され検索結果に含まれるかも指定しても良い。クエリハンドラ10は、関係型データクエリを関係型データベース記憶装置12に、より具体的には装置に格納された関係型データベースのようなクエリハンドラ10のコンポーネントにより、場合によっては関係型データベース管理システムを介して、発行する。結果は、関係型データベース記憶装置12により又はそのコンポーネントにより、クエリハンドラ10に返される。

30

【 0 0 5 6 】

さらに、クエリハンドラは、関係型データクエリの結果に含まれるデータに関連するデータについて、グラフデータベースを検索するためのクエリを生成するよう構成される。このように生成されたクエリは、グラフデータを検索するのに適する、例えばRDFトリプルストアを検索するのに適するフォーマットなので、グラフデータクエリと称される。例えば、関係型データベース内の主要列が存在しても良い。したがって、グラフデータベース記憶装置に格納されるグラフデータベースのオントロジは、リソースが主要列に対応し、主要列リソースのインスタンスが関係型データベース内のエントリに対応し、関係型データベース内の対応するエントリの対応する値と同じ値にリンクする述語の値を有するようになる。したがって、データが関係型データクエリの検索結果に含まれる主要列内のエントリの値は、関係型データについてグラフデータベースを検索する基礎として用いることができる。つまり、グラフデータベースは、関係型データベース内のエントリに関連

40

50

する全てのデータが、関係型データベース内の対応するエントリと同じ値を有する主要列リソースのインスタンスにリンク付けされるように構造化され得る。したがって、主要列は、関係型データベース内のどの行がグラフデータベース内のデータに関連するかを識別するユニークな識別子として動作する。グラフデータは、グラフデータベース記憶装置に又はそれに格納されたグラフデータベース若しくはその管理システムに発行され、検索結果はクエリハンドラ 10 へ返される。

【0057】

クエリハンドラ 10 は、関係型データクエリ及びグラフデータクエリからの結果を照合し集約し、照合し集約した結果をクライアントアプリケーション 20 に返すよう構成される。

10

【0058】

図 2 は、本発明の実施形態の例示的なコンポーネントアーキテクチャである。図 2 のコンポーネントアーキテクチャは、以下のコンポーネントを有する。

【0059】

RDBMS 120。本願明細書の他の場所で言及される関係型データベース記憶装置の例である。特に、関係型データベース（又は 1 より多い関係型データベース）を格納し管理する。

【0060】

RDFトリプルストア 140。本願明細書の他の場所で言及されるグラフデータベース記憶装置の例である。RDFトリプルストアは、RDBMS 120 を介してアクセス可能な関係型データベース内に格納されたデータに関連するグラフデータをエンコードする。

20

【0061】

RDBMSスキーマ及びRDFオントロジマッピング 110。本願明細書の他の場所で言及されるマッピングユニットの例である。RDBMSスキーマ/RDFオントロジマッピング 110 は、プラグインプログラム 100 の外部に示されるが、プラグインプログラム 100 の一部として含まれても良い。RDBMSスキーマ/RDFオントロジマッピング 110 は、ファイル又は格納された情報の特定の他のレポジトリであっても良く、クエリハンドラ 10 によりアクセス可能であり読み取り可能である。

【0062】

JDBC/ODBCドライバ 130 は、通常、RDBMS 120 に格納された関係型データベースに提出されたクエリを処理するドライバソフトウェアプログラムである。しかしながら、図 2 の特定の構成では、データベースドライバは、本願明細書の他の場所で言及されるクエリハンドラ 10 の例であるプラグインプログラム 100 を有する。プラグインプログラム 100 は、実際に、クエリを傍受し、JDBC/ODBCドライバ 130 の代わりに検索結果を得るために処理を実行する。

30

【0063】

プラグインプログラム 100 で受信された関係型データクエリは、SQLステートメントの形式であり、したがって RDBMS 120 に格納された関係型データベースを検索するのに適する形式である。プラグインプログラム 100 は、関係型データクエリ（SQLステートメント）の検索結果に関連する RDFトリプルストア 140 内のデータを見つけるために、関係型データクエリに基づき SPARQLクエリを生成するよう構成される。プラグインプログラム 100 は、RDBMSドライバの内部にインストールされ、いかなる形式、例えば JDBC 又は ODBC であっても良く、SQLステートメントがドライバ 130 で受信されると SQLステートメントを検出する。プラグインプログラム 100 は、次に、RDFトリプルストア 140 内で検索するために及び関係型データ検索結果内の検索された行の主要データ列の値を識別するためにリソース名を識別できる順序で、必要な情報を RDBMSスキーマ/RDFオントロジマッピング 110 から検索するよう構成される。プラグインプログラム 100 は、次に、関係型データ検索結果内の検索された行内の主要データの値と一致する値を有する主要データ列に対応するリソースのインスタンス（のプロパティ）と関連するデータを検索する SPARQLクエリを生成するよう構成

40

50

される。プラグインプログラム 100 は、次に、生成した SPARQL クエリを RDF トリプルストア 140 に転送するよう構成される。

【0064】

RDBMS スキーマ / RDF オントロジマッピング 110 は、RDF オントロジ語彙を有する RDBMS スキーマをマッピングするマッピングファイルを格納する（又は単にマッピングファイルであっても良い）。単純な例として、RDBMS スキーマフィールドは、RDF クラス定義又はステートメントにリンク付けられる。したがって、SQL ステートメントがクライアントアプリケーションから RDBMS 120 へ JDBC / ODBC ドライバ 130 を介して発行されると、プラグインプログラム 100 は、SQL ステートメントの結果に関連する RDF トリプルストア 140 内の RDF トリプルステートメントを探す SPARQL クエリを生成するために必要な情報を得るために、RDBMS スキーマ / RDF オントロジマッピング 110 を参照できる。RDF トリプルストア 140 の場所（分散型コンピューティング環境又は更に広範な環境内のアドレス）もマッピングファイルに格納されても良いので、プラグインプログラム 100 は、SPARQL クエリがどこに発行されるべきかを識別する情報にアクセスできる。

10

【0065】

図 3 は、どのようにマッピングファイルがコンパイルされるか、及びマッピングファイルに格納される情報を示す。RDBMS スキーマ 111 及び RDF オントロジ 112 は、マッピングファイルに格納されてもされなくても良いが、図 3 ではマッピングファイルの構成コンポーネントを示すために図示される。

20

【0066】

データは、RDF トリプルストア 140 に、RDF オントロジ 112 で定義される RDF クラスのインスタンス及びそのプロパティとして格納される。RDBMS スキーマ 111 は、関係型データベースの見出し付き列の名称を与える。関係型データベース内の行は見出し付き列の各々の中にエントリを有し、データは、関係型データベース内にそれらのエントリの値として格納される。

【0067】

マッピングファイルは、RDBMS スキーマ / RDF オントロジマッピング 110 として表現される。RDBMS スキーマ / RDF オントロジマッピング 110 は、RDBMS スキーマ 111 内の見出し付き列と RDF オントロジ 112 内のリソース名との間の対応を記録する。関係型データベース内の関係型データに関連するグラフデータは、RDF トリプル内に、関係型データが属する見出し付き列のインスタンスのプロパティとして、主要列として指定される見出し付き列内の関係型データが属する行のエントリに対応する値を有するリソースのインスタンスのプロパティとして（つまり、インデックス付けされた関係型データの行に従って）、格納される。

30

【0068】

マッピング方法を更に説明するために、RDBMS テーブルを RDF オントロジにマッピングし関係するデータを RDF トリプルストアに格納できるようにする例示的なルールセットを以下に説明する。

【0069】

図 4 の例示的な関係型データベースでは、簡単のため 1 行のデータのみを示す。5 個の見出し付き列、つまり Ref_No（参照番号）、Date（日付）、Trading（取引）、Travel（旅行）、Total（合計）がある。図示のデータの行は、各見出し付き列にその行のエントリの値を有する。Ref_No は主要列である。これは、各行にユニークな値を有する見出し付き列であり、このユニークな値により行がインデックス付けされる。

40

【0070】

RDF オントロジでは、関係型データベースからの各見出し付き列は、対応するリソースを有し、命名の取り決めに従って命名される。この特定の例では、リソースは、Expense_Ref_No、Expense_Date、Expense_Trading、Expense_Travel と命名される。これらは、RDF オントロジのリソースであり、したがって、これらは、RDF オントロジにおいて

50

r d f 種類又はクラスとして以下のように定義される。

【数 1】

```
Expense_Ref_No rdf:type rdfs:Class
Expense_Date rdf:type rdfs:Class
Expense_Trading rdf:type rdfs:Class
Expense_Travel rdf:type rdfs:Class
```

【0071】

10

本例で用いられた命名の取り決めは、クラスの名称が、アンダースコアにより分離されたテーブルの名称と列の名称との組み合わせに従うことである。これは、異なるテーブルに重複する列の名称が存在する場合に混乱を回避する。

【0072】

さらに、主要列に対応するクラス、本例ではExpense_Ref_Noは、他の列見出しに対応するクラスへのリンクを含む述語を有する。したがって、主要列に対応するクラスは、関係型データベース内の行を表す階層構造のルートになる。

【0073】

この特定の例では、Expense_Ref_No rdf:type rdfs:Classは、少なくとも以下の述語を有する。

20

【数 2】

```
has_date
has_trading
has_travel
has_total.
```

【0074】

最後に、各列リソースは
(外 1)

30

```
has_value
```

述語を有し、リソースの各インスタンスは関係型データベース内の対応する見出し付き列内のエントリに対応し、

(外 2)

```
has_value
```

述語によりリンク付けされた目的語は、対応するエントリの値を反映する値を有する目的語である。例えば、travel_1 1 1 はExpense_Travelクラスのインスタンスであり、値「3 0 0 2」を有する目的語にリンクする述語has_valueを有する。インスタンスは、対応するExpense_Ref_Noインスタンスを介して参照可能である。対応するExpense_Ref_Noインスタンスにtravel_1 1 1 がリンク付けされるので、travel_1 1 1 及び値「3 0 0 2」は、Ref_Noの対応する値を有するExpense_Ref_Noクラスのインスタンスのプロパティである。

40

【0075】

図 5 は、上述の例示的なルールセットに従う R D F データセットを示す。図 4 に示したデータの行に関連するデータは R D F データセットに格納されることが分かる。例えば、「Trading」と見出しの付いた列にあるエントリの値を計算するために用いられた為替レ

50

ートは、Trading列にあるエントリに対応するリソースのインスタンスのプロパティとして与えられる。さらに、Travel列にあるエントリに関連する送り状番号は、Travel列にあるエントリに対応するリソースのプロパティとして与えられる。

【 0 0 7 6 】

図 6 は、本発明を実施するプラグインプログラムにより実行される処理の例を示す。コンポーネント間の情報転送を説明するために、処理は、図 2 に対応するコンポーネントアーキテクチャに重ね合わせて図示される。

【 0 0 7 7 】

図 2 のコンポーネントアーキテクチャに加えて、図 6 の処理が重ね合わされたコンポーネントアーキテクチャは、プラグインプログラム 1 0 0 のコンポーネントとしてローカルレジストリ 1 0 1 を有する。ローカルレジストリ 1 0 1 は、プラグインプログラム 1 0 0 にアクセス可能な記憶場所であり、プラグインプログラム 1 0 0 により実行中のプロセスに関連するデータは、一時的又はもっと恒久的な基盤に格納され得る。

【 0 0 7 8 】

処理は、クライアントアプリケーション 2 0 からの S Q L クエリ A の受信により開始される。S Q L ステートメント A は、R D B M S 1 2 0 に格納された関係型データベースを検索するのに適するフォーマットである。

【 0 0 7 9 】

ステップ S 1 で、プラグインプログラム 1 0 0 は、S Q L クエリ A を受信し（傍受し）、ローカルレジストリ 1 0 1 に記録する。

【 0 0 8 0 】

図 3 ～ 5 に図示した例示的なデータ及びそれらの関連する説明では、S Q L クエリ A の一例は、以下の通りである（これは、この議論の目的で「説明のためのデータ例」として参照されるべきである）。

【 数 3 】

```
SELECT e.Trading, e.Travel FROM Expense as e
WHERE e.Date="May_2012";
```

【 0 0 8 1 】

ステップ S 2 で、プラグインプログラム 1 0 0 は、関係型データベース又は S Q L クエリ A により検索されている関係型データベーステーブル内のどの見出し付き列が主要列であるかを識別するために、マッピングユニット 1 1 0 へメッセージを送信する。マッピングユニット 1 1 0 は、関係型データベース又はデータベーステーブル内の主要列の名称を識別する情報で応答する。

【 0 0 8 2 】

図示のデータ例では、ステップ S 2 で返される情報は、以下と類似する何からの形式を取っても良い。

【 数 4 】

```
Primary Key P = Ref_No.
```

【 0 0 8 3 】

S Q L クエリ A は、どの行からのデータが検索結果に含まれるかを決定する条件を含む。ステップ S 3 で、プラグインプログラム 1 0 0 は、S Q L ステートメント / クエリの形式であっても良い更なる関係型データクエリ B を生成する。更なる関係型データクエリ B は、S Q L クエリ A と同じ条件を含むが、特に、主要列にあるエントリの値を要求する。

【 0 0 8 4 】

図示のデータ例では、更なる関係型データクエリ B は、以下の形式を取る。

10

20

30

40

【数 5】

```
SELECT e.Ref_No FROM Expense as e
WHERE e.Date = "May_2012";
```

【0085】

ステップS4で、SQLクエリA及び更なる関係型データクエリBは、RDBMS120に発行される。ステップS5で、SQLクエリAの結果R_Aは、JDBC/ODBCドライバ130又はプラグインプログラム100へ返される。ステップS6で、更なる関係型データクエリBの結果R_Bは、プラグインプログラム100へ返され、ローカルレジス

10

【0086】

図示のデータ例では、結果R_Aは以下の通りであっても良い。

【数 6】

```
RA = Trading value: 10900, Travel value: 3002;
and the results RB:
RB = "10009".
```

【0087】

SQLクエリAは、条件を満たす行にあるテーブルエントリからのどの見出し付き列からデータが読み出されるべきかを指定する選択句を含む。ステップS7で、プラグインプログラム100は、マッピングユニット110へメッセージを送信して、SQLクエリAで指定された見出し付き列に対応するRDFトリプルストア140内のデータを規定するRDFオントロジにおけるクラスの名称を要求する。簡単のため、メッセージに回答してマッピングユニットにより返される1又は複数の名称のセットは、Mと表す。

20

【0088】

図示のデータ例では、マッピングユニット110により提供される情報は以下の通りであっても良い。

【数 7】

```
Expense.Trading -> Expense_Trading
Expense.Travel -> Expense_Travel.
```

30

【0089】

ステップS8で、プラグインプログラム100は、Sと表されるSPARQLクエリのようなグラフデータクエリを生成する。Sは、選択句にMを、及び条件フィルタとしてR_B内の値と一致する主要列に対応するクラスの値を含む。クエリSは、クエリ内で指定されたクラスインスタンスの全てのプロパティを要求しても良い。

【0090】

図示のデータ例では、SPARQLクエリSは、単一の子SPARQLクエリとして生成できる。しかしながら、説明を簡単にするため、ここでは3個の別個のクエリの観点から説明する。

40

【0091】

まず、R_B（これは、acct:ref_no_122インスタンスを返すべきである）を含むExpense_Ref_Noのインスタンスを見付けるために、図7AのSPARQLクエリが用いられる。

【0092】

次に、図7BのSPARQLクエリは、RDFストアからExpense_Travel及びExpense_Tradingインスタンス全てを見付けるために用いられる。ここでExpense_Ref_No=acct:ref_no_122である。このSPARQLクエリの戻り結果は、acct:trading_111及びacc

50

t:travel_1 1 1である。

【0093】

さらに、図7CのSPARQLクエリは、主語がacct:trading_1 1 1及びacct:travel_1 1 1である場合の述語及び目的語全てを見付けるために用いられる。

【0094】

ステップS9で、SPARQLクエリSは、場合によってはSQLクエリAがアドレスされた関係型データベース120に対応するRDFトリプルストア140を識別する場所情報を用いて、RDFトリプルストア140に発行される。

【0095】

ステップS10で、SPARQLクエリSの結果R_Cは、プラグインプログラム100に返される。ステップS11で、結果R_A及びR_Cは、プラグインプログラムにより結合され、クライアントアプリケーション20に返される。

【0096】

図示のデータ例では、SQLクエリの実行全体の結合結果は、以下を含む。

【数8】

R_A = Trading value: 10900, Travel value: 3002

R_C = Trading has_exchange_rate: 1.212; Travel has_invoice: 3011

【0097】

注：2つの結果セットを個々のデータベースと別個に返すこともできる。

【0098】

したがって、本発明を具現化するメカニズムは、従来のSQL検索結果セットの一部として利用できない追加情報へのアクセスを可能にする。例えば、

RDBMSからの戻り結果セットは次の通りである。

【数9】

R_A = Trading value: 10900, Travel value: 3002

【0099】

一方、RDFトリプルストアから、クエリに応答して以下の追加情報が返される。

【数10】

Trading where has_value 10009, also has_exchange_rate: 1.212

Travel where has_value 3002, also has_invoice: 3011

【0100】

図8は、本発明を具現化するシステムのシステムアーキテクチャを示す。図8のシステムのシステムアーキテクチャは、前述の例と類似している。同様の参照符号が同様のコンポーネントに対して用いられ、これらのコンポーネントの説明は省略する。

【0101】

図8のシステムアーキテクチャでは、1より多い関係型データベース(120__1、120__2、120__n)は、同じJDBC/ODBCドライバ130を介してアクセスできる。さらに、各関係型データベース(120__1、120__2、120__n)は、対応するグラフデータベース(140__1、140__2、140__n)を有し、対応する関係型データベース内のデータに関連するデータをRDFトリプルとして格納する。したがって、プラグインプログラム100は、関係型データベース(120__1、120__2、120__n)のいずれかに対するSQLクエリを受信し、対応するグラフデータベース(140__1、140__2、140__n)内の関連データを見付けるためにSPARQLクエリを生成するよう構成される。さらに、マッピングユニット110は、1より多いマッピ

10

20

30

40

50

ングファイルを有する。マッピングファイルは、各対応するデータベース対に対して1つある。

【符号の説明】

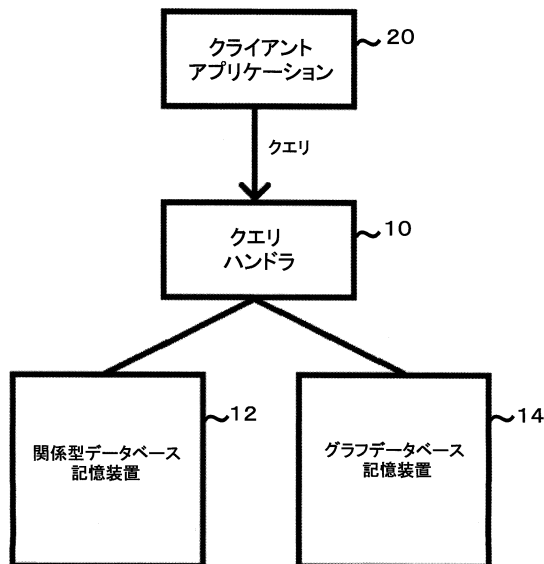
【0102】

- 10 クエリハンドラ
- 12 関係型データベース記憶装置
- 14 グラフデータベース記憶装置
- 20 クライアントアプリケーション
- 100 プラグインプログラム
- 101 ローカルレジストリ
- 110 RDBMSスキーマ/RDFオントロジマッピング
- 111 RDBMSスキーマ
- 112 RDFオントロジ
- 130 JDBC/ODBCドライバ
- 140 RDFトリプルストア

10

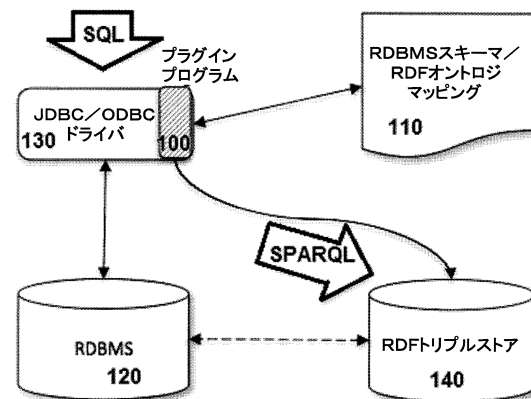
【図1】

本発明を具現化するシステムの概略図



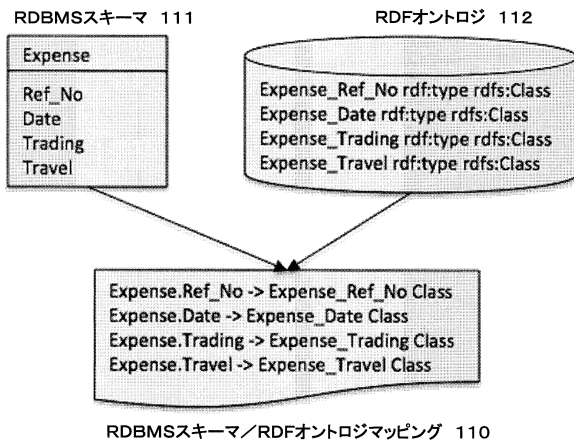
【図2】

本発明の実施形態の例示的なコンポーネントアーキテクチャ



【図 3】

どのようにマッピングファイルがコンパイルされるか、及びマッピングファイルに格納される情報を示す図



【図 5】

本発明の実施形態におけるRDFデータセットを示す図

RDFデータセット

```

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix acct: <http://fujitsu.com/accounting/ontology/2012/>.

<http://fujitsu.com/acct/data/2012/ref_no_122>
rdf:type acct:Expense_Ref_No; acct:has_id "10009"; acct:has_date date_may;
acct:has_trading acct:trading_111; acct:has_travel acct:travel_111.

<http://fujitsu.com/acct/data/2012/date_may>
rdf:type acct:Expense_Date; acct:has_value "14/04/2011".

<http://fujitsu.com/acct/data/2012/trading_111>
rdf:type acct:Expense_Trading; acct:has_value "10900";
acct:has_exchange_rate "1.212".

<http://fujitsu.com/acct/data/2012/travel_111>
rdf:type acct:Expense_Travel; acct:has_value "3002"; acct:has_invoice "3011".
  
```

【図 4】

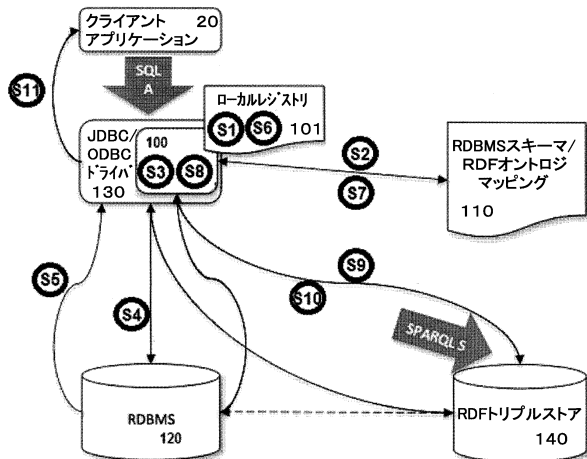
本発明の実施形態における関係型データベースを示す図

Expense

Ref_No	Date	Trading	Travel	Total
...	...			
10009	14/04/2011	10900	3002	13902
...	...			

【図 6】

本発明を実施するプラグインプログラムにより実行される処理の例を示す図



【図 7】

本発明を実施するクエリハンドラにより生成される例示的な SPARQL ステートメントを示す図

```

PREFIX acct: <http://fujitsu.com/accounting/ontology/2012/>.

SELECT ?Expense_Ref_No
WHERE {
  ?Expense_Ref_No acct:has_id R8
}
  
```

(A)

```

PREFIX acct: <http://fujitsu.com/accounting/ontology/2012/>.

SELECT ?Expense_Trading, ?Expense_Travel
WHERE {
  acct:ref_no_122 acct:has_trading ?Expense_Trading.
  acct:ref_no_122 acct:has_travel ?Expense_Travel
}
  
```

(B)

```

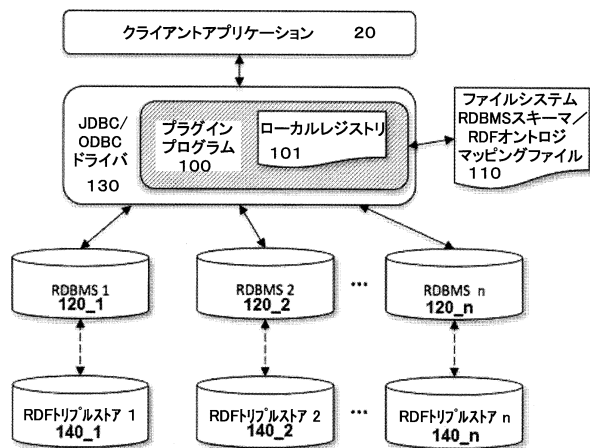
PREFIX acct: <http://fujitsu.com/accounting/ontology/2012/>.

SELECT ?predicates, ?objects
WHERE {
  acct:trading_111 ?predicates ?objects .
  acct:travel_111 ?predicates ?objects
}
  
```

(C)

【図 8】

本発明を具現化するシステムのシステムアーキテクチャを示す図



フロントページの続き

(72)発明者 カルヴァーリョ・ヌノ
イギリス国、ダブリュ5 4ディーダブリュ、ロンドン、ウィンミル ロード、ピカリング ハウス 8番

(72)発明者 松塚 貴英
イギリス国、ダブリュ3 0ジェイエヌ、ロンドン、チャーチル ガーデنز 33番

審査官 樋口 龍弥

(56)参考文献 米国特許第09098566(US, B1)
米国特許出願公開第2006/0047636(US, A1)
米国特許出願公開第2008/0172353(US, A1)
特開平10-222539(JP, A)

(58)調査した分野(Int.Cl., DB名)
G06F 17/30
G06F 12/00