

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6492008号
(P6492008)

(45) 発行日 平成31年3月27日(2019.3.27)

(24) 登録日 平成31年3月8日(2019.3.8)

(51) Int.Cl.

F I

G 1 6 H 15/00 (2018.01)

G 1 6 H 15/00

G 1 6 H 70/00 (2018.01)

G 1 6 H 70/00

G 0 6 F 16/00 (2019.01)

G 0 6 F 17/30 2 2 0 C

G 0 6 F 17/30 3 3 0 B

請求項の数 13 (全 25 頁)

(21) 出願番号 特願2015-539584 (P2015-539584)
 (86) (22) 出願日 平成25年8月14日 (2013.8.14)
 (65) 公表番号 特表2015-536498 (P2015-536498A)
 (43) 公表日 平成27年12月21日 (2015.12.21)
 (86) 国際出願番号 PCT/US2013/054835
 (87) 国際公開番号 W02014/070277
 (87) 国際公開日 平成26年5月8日 (2014.5.8)
 審査請求日 平成28年8月1日 (2016.8.1)
 (31) 優先権主張番号 61/720, 611
 (32) 優先日 平成24年10月31日 (2012.10.31)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 13/790, 252
 (32) 優先日 平成25年3月8日 (2013.3.8)
 (33) 優先権主張国 米国 (US)

(73) 特許権者 502303739
 オラクル・インターナショナル・コーポレ
 イション
 アメリカ合衆国カリフォルニア州9406
 5レッドウッド・シティー, オラクル・パ
 ークウェイ500
 (74) 代理人 110001195
 特許業務法人深見特許事務所
 (72) 発明者 トリーベル, ウィリアム・アンドリュース
 アメリカ合衆国、94115 カリフォル
 ニア州、サン・フランシスコ、エディー・
 ストリート、2004

審査官 関 博文

前置審査

最終頁に続く

(54) 【発明の名称】 コホート識別システム

(57) 【特許請求の範囲】

【請求項 1】

1つ以上の有害事象ケースの集合であるコホートを識別するためにコンピュータで実行される方法であって、

1つ以上の有害事象ケースを含むデータソースの1つ以上のデータフィールドについての情報および前記データソースのための構造化照会言語 (SQL) を記述するメタデータを検索することを含み、前記メタデータは、有害事象ケースに対する変更と、クエリーフィールドとを含み、前記1つ以上の有害事象ケースは、各々を識別するためのケース識別子を含み、前記情報は、構造化照会言語 (SQL) クエリーを構築するための情報を含み、前記SQLクエリーを構築するための情報は、データタイプおよびデータフィールドを

10

含み、前記方法はさらに、
 前記検索されたメタデータに基づいて、前記データソースに対してSQLクエリーを作成することと、

前記SQLクエリーを1つ以上のコンパイルルールに基づいてコンパイルすることと、
 前記SQLクエリーを前記データソース上において実行することとを含み、前記SQLクエリーの前記実行はケースシリーズを作成し、前記ケースシリーズは、1つ以上の有害事象ケースのリストであり、かつ、当該ケースシリーズのバージョンを識別するためのタイムスタンプ情報を含む一つ以上のケースリビジョンに関連付けられており、前記方法はさらに、

前記ケースシリーズに基づいてレポートを生成することを含み、前記レポートは前記ケ

20

ースシリーズを視覚化したものを含む、コンピュータで実行される方法。

【請求項 2】

さらに、前記クエリーをクエリーリポジトリに記憶することを含む、請求項 1 に記載のコンピュータで実行される方法。

【請求項 3】

さらに、前記ケースシリーズをケースシリーズリポジトリに記憶することを含む、請求項 1 または 2 に記載のコンピュータで実行される方法。

【請求項 4】

さらに、報告ケースシリーズアプリケーションプログラミングインターフェイスを用いて、前記ケースシリーズを前記ケースシリーズリポジトリから検索することを含み、前記報告ケースシリーズアプリケーションプログラミングインターフェイスは、報告ソフトウェアアプリケーションに対して、前記ケースシリーズのフォーマットを規定する、請求項 3 に記載のコンピュータで実行される方法。

10

【請求項 5】

さらに、前記レポートを前記報告ソフトウェアアプリケーション内において表示することを含む、請求項 4 に記載のコンピュータで実行される方法。

【請求項 6】

さらに、相互運用可能なケースシリーズアプリケーションプログラミングインターフェイスを用いて、前記ケースシリーズを前記ケースシリーズリポジトリから検索することを含み、前記相互運用可能なケースシリーズアプリケーションプログラミングインターフェイスは、パートナーソフトウェアアプリケーションに対して、前記ケースシリーズのフォーマットを規定する、請求項 3 に記載のコンピュータで実行される方法。

20

【請求項 7】

さらに、前記ケースシリーズを前記パートナーソフトウェアアプリケーション内において消費することを含む、請求項 6 に記載のコンピュータで実行される方法。

【請求項 8】

前記ケースシリーズは 1 つ以上の有害事象ケースを含み、各有害事象ケースは有害事象を表現するデータレコードを含む、請求項 1 ～ 7 のいずれかに記載のコンピュータで実行される方法。

【請求項 9】

前記データソースは有害事象レポートデータベースを含む、請求項 1 ～ 8 のいずれかに記載のコンピュータで実行される方法。

30

【請求項 10】

前記ケースシリーズは、1 つ以上の薬剤の安全性に関係付けられる 1 つ以上のレポートまたは患者識別子を含む、請求項 1 ～ 9 のいずれかに記載のコンピュータで実行される方法。

【請求項 11】

複数のケースシリーズをマージするステップをさらに含む、請求項 1 ～ 10 のいずれかに記載のコンピュータで実行される方法。

【請求項 12】

請求項 1 ～ 11 のいずれかに記載の方法をコンピュータに実行させる、プログラム。

40

【請求項 13】

請求項 12 に記載のプログラムを格納したメモリと、
前記メモリに結合され、前記プログラムを実行するためのプロセッサとを備える、システム。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願への相互参照

本願は、2012 年 10 月 31 日に提出された米国仮特許出願連続番号第 61 / 720

50

、611号の優先権を主張するものであり、その主題をここに引用により援用する。

【0002】

分野

1つの実施例は、コンピュータシステムに向けられ、より特定的には、臨床データを管理するコンピュータシステムに向けられる。

【背景技術】

【0003】

背景

薬剤安全性などのような健康科学の領域では、「ケースシリーズ」は有害事象ケースのリストである。「有害事象ケース」または「ケース」は、患者に生じる有害事象の、特定の事態のデータレコードである。各有害事象ケースは一意の識別子を有し得る。一般に健康科学においては、ケースシリーズは、さらに、「患者リスト」または「対象リスト」としても識別され得る。

10

【0004】

多くの薬剤安全性システムはケースシリーズを形成および消費し得、「薬剤安全性システム」は薬剤安全性データを記憶するシステムであり、薬剤安全性データは、1つ以上の有害事象ケースなどのような、1つ以上の薬剤の安全性に関係付けられるデータを含む。典型的には、これらのシステム内では、実行可能なプロセスが、ケースシリーズを形成し、それを1つ以上の実行可能なプロセスに渡し、それらのプロセスはその上で動作する。一般的なシナリオでは、薬剤安全性システムのようなデータソース上においてクエリーを実行する実行可能なプロセスは、ケースシリーズを形成し、それを、レポートを実行する実行可能なプロセスに渡す。このシナリオでは、ケースシリーズは、データソース上においてクエリーを実行する実行可能なプロセスの結果である。換言すれば、ケースシリーズを含むケースのリストは、データソース上において実行可能なプロセスにより実行されるクエリーにおいて指定される条件と一致するケースのリストである。レポートは、ケースシリーズの所望の出力フォーマットであり得、レポートは、実行可能なプロセスによって実行され得る。ケースシリーズは、典型的には、各ケース毎に少なくとも一意の識別子（典型的には「ケース識別子」として識別される）を含み、さらに、ケースシリーズにおいて有害事象ケースを表わすさらに他のケースデータまたはメタデータを含んでもよい。ケースシリーズにおけるデータフィールドは、クエリーまたはレポートにおいて指定されるデータフィールドと必ずしも同じである必要はない。ケースシリーズにおけるデータフィールドは典型的には固定され得るが、レポートデータフィールドは所望の出力フォーマットに依存して変更され得る。

20

30

【0005】

さらに、健康科学の領域では、患者の「コホート」を識別することが必要となり得、患者の「コホート」は同一または同様の特性を有する患者のグループである。薬剤安全性の領域では、患者は有害事象ケースの対象となり得る。より広い健康科学の領域では、患者は臨床レコードの対象となり得る。

【発明の概要】

【課題を解決するための手段】

40

【0006】

概要

1つの実施例は、患者のコホートのような、コホートを識別するコホート識別システムである。コホート識別システムは、データソースの1つ以上のデータフィールドについての情報を含むメタデータを検索し、その情報は、1つ以上のデータフィールドの各データフィールド毎に、データタイプおよび構造化照会言語情報を含む。コホート識別システムは、さらに、検索されたメタデータに基づいて、データソースのためにクエリーを作成する。コホート識別システムは、さらに、1つ以上のコンパイルルールに基づいてクエリーをコンパイルする。コホート識別システムは、さらに、データソース上においてクエリーを実行し、クエリーを実行することは、ケースシリーズを作成する。コホート識別システ

50

ムは、さらに、ケースシリーズに基づいてレポートを生成し、レポートはケースシリーズの視覚化を含む。

【 0 0 0 7 】

図面の簡単な説明

さらに他の実施例、詳細、利点および修正は、添付の図面と関連付けられる、好ましい実施例の以下の詳細な記載から明らかとなる。

【図面の簡単な説明】

【 0 0 0 8 】

【図 1】この発明の実施例を実施し得るシステムのブロック図を示す。

【図 2】この発明の実施例に従う、相互運用可能なケースシリーズシステムのブロック図を示す。

10

【図 3】この発明の実施例に従う、ケースシリーズデータモデルのブロック図を示す。

【図 4】この発明の実施例に従う、相互運用可能なケースシリーズシステムの例示的実現例のブロック図を示す。

【図 5】この発明の実施例に従う、相互運用可能なケースシリーズモジュールの機能のフロー図を示す。

【図 6】この発明の実施例に従う、コホート識別システムのブロック図を示す。

【図 7】この発明の実施例に従う、コホート識別システムの例示的実現例のブロック図を示す。

【図 8】この発明の実施例に従う、コホート識別モジュールの機能のフロー図を示す。

20

【図 9】この発明の実施例に従う、例示的レポートを作成する例示的ケースシリーズを作成する例示的クエリーを示す。

【発明を実施するための形態】

【 0 0 0 9 】

詳細な説明

一実施例においては、データソースおよびそのデータソースのための構造化照会言語（「SQL」）の両方を記述するメタデータを検索し得、およびその検索されたメタデータに基づいてデータソース上で1つ以上のクエリーを作成し実行し得る、コホート識別システムが提供される。データソースは1つ以上の有害事象ケースを含み得、各有害事象ケースは、有害事象を表わすデータレコードを含む。代替的实现例においては、データソースは1つ以上の医療レコードを含み得、各医療レコードは医療データを含むデータレコードである。コホート識別システムは、さらに、1つ以上の実行されたクエリーから1つ以上のケースシリーズを生成し得、その1つ以上のケースシリーズをケースシリーズリポジトリに記憶し得る。「ケースシリーズ」は1つ以上の有害事象ケースの集合である。「患者リスト」は1つ以上の医療レコードの集合である。コホート識別システムは、さらに、1つ以上のケースシリーズに基づいて1つ以上のレポートを生成し得る。生成された1つ以上のレポートを用いることによって、患者のコホートなどのようなコホートを視覚化し得る。「コホート」は、1つ以上の関係付けられる有害事象ケースの集合、または1つ以上の関係付けられる医療レコードの集合である。たとえば、大きな乳癌の発生を伴う地理的領域を判断するために、コホート識別システムは、同様の特性を有する同様の患者のコホートまたはグループを識別するために患者データの集合を解析し得、それは、そのコホートの患者間における共通の因子の解析を可能にし得る。コホート識別システムは検索されたメタデータに基づいてクエリーを作成し実行するので、コホート識別システムは任意のデータモデルとともに用いられ得、1つ以上のソフトウェアアプリケーションと統合され得る。この明細書に記載されるように、「コンピュータアプリケーション」、「ソフトウェアアプリケーション」、または「アプリケーション」は、コンピュータプログラムおよび/またはモジュールの任意の集まりである。たとえば、コホート識別システムは任意の人の健康科学データモデルとともに用いられ得る。コホート識別システムは、さらに、同様のデータモデルを用いる他の領域においても用いられ得る。たとえば、他の実施例では、コホート識別システムは、エンドユーザにとって利用可能な複数のデータソースととも

30

40

50

に用いられ得、複数のデータソースは有害事象データベースと医療履歴データベースとの組合せを含み得る。

【 0 0 1 0 】

ある実施例は、(a) データソースのメタデータ記述およびデータソースの S Q L の両方 (これは参照データに対する拡張可能なアプローチを含む) ; (b) ソフトウェアアプリケーションとの統合のためのアプリケーションプログラミングインターフェイス (「 A P I 」) ; ならびに (c) ケースシリーズおよびメタデータをレポートにおいて用いることを容易にし得る汎用報告ソフトウェアアプリケーションとの統合のための A P I 、を含み得る。データモデルに対するメタデータ駆動型アプローチは、新たな概念を、コホート識別システムにおいて、コホート識別システムを変更することなく実現することを可能にし得る。

10

【 0 0 1 1 】

さらに、ある実施例は、3つのクラスのソリューションを含み得る。第1のクラスのソリューションは、クエリー結果を作成し修正する機能のような、ソフトウェアアプリケーション特化ツールおよびデータモデル特化ツールを含む。アプリケーションによって、クエリー結果は「ケースシリーズ」または「患者リスト」と呼ばれる。第2のクラスのソリューションは、汎用ビジネスインテリジェンス (「 B I 」) クエリーソリューションを含む。第3のクラスのソリューションは、スタンドアロンの構造化照会言語 (「 S Q L 」) クエリービルダを含む。したがって、この発明の実施例は、データモデルから独立しながら (つまりデータソースの特定のデータモデルに依存することなく) 幅広い機能のクラスを提供し得る。

20

【 0 0 1 2 】

図1は、この発明の一実施例を実施し得るシステム10のブロック図である。システム10は、システム10のコンポーネント間において情報を通信するためのバス12または他の通信機構を含む。システム10は、さらに、バス12に作動的に結合され、情報を処理し命令または動作を実行するためのプロセッサ22を含む。プロセッサ22はどのようなタイプの汎用または特定目的のプロセッサであってもよい。システム10は、さらに、情報およびプロセッサ22によって実行されるべき命令を記憶するためのメモリ14を含む。メモリ14は、ランダムアクセスメモリ (「 R A M 」) 、リードオンリメモリ (「 R O M 」) 、磁気ディスクもしくは光ディスクのようなスタティックストレージ、または任意の他のタイプの機械もしくはコンピュータ読取可能媒体の任意の組合せからなり得る。システム10は、さらに、ネットワークインターフェイスカードまたは他の通信インターフェイスなどのような通信装置20を含んで、ネットワークに対するアクセスを与える。この結果、ユーザは、システム10と、直接、またはネットワークもしくは任意の他の方法を介して遠隔でインターフェイスしてもよい。

30

【 0 0 1 3 】

コンピュータ読取可能媒体は、プロセッサ22によってアクセスされ得る任意の利用可能な媒体であってもよい。コンピュータ読取可能媒体は、揮発性媒体および不揮発性媒体、取外し可能な媒体および取外し可能でない媒体、通信媒体、ならびに記憶媒体を含んでもよい。ある通信媒体は、コンピュータ読取可能命令、データ構造、プログラムモジュールまたは他のデータを、搬送波または他の移送機構などのような変調されたデータ信号で含んでもよく、当該技術分野において公知の任意の他の形式の情報配信媒体を含んでもよい。ある記憶媒体は、R A M 、フラッシュメモリ、R O M 、消去可能な P R O M (「 E P R O M 」) 、電氣的に消去可能な P R O M (「 E E P R O M 」) 、レジスタ、ハードディスク、リムーバブルディスク、コンパクトディスクリードオンリメモリ (「 C D - R O M 」) 、または当該技術分野において公知の任意の他の形式の記憶媒体を含んでもよい。

40

【 0 0 1 4 】

プロセッサ22は、さらに、バス12を介して、液晶ディスプレイ (「 L C D 」) などのようなディスプレイ24に作動的に結合され得る。ディスプレイ24はユーザに対して情報を表示し得る。キーボード26およびコンピュータマウスのようなカーソル制御装置

50

28もバスを介して作動的に結合されて、ユーザがシステム10とインターフェイスすることを可能にする。他の実施例では、ユーザは、ヒューマンインターフェイス装置(図1においては図示せず)を用いてシステム10とインターフェイスし得、ヒューマンインターフェイス装置は、ユーザと直接対話し、およびユーザから入力を得るよう構成される装置である。ヒューマンインターフェイス装置の例は、ウェブカム、指紋スキャナおよびヘッドセットを含む。

【0015】

一実施例によると、メモリ14は、プロセッサ22によって実行されると機能を提供してもよいソフトウェアモジュールを記憶し得る。モジュールは、オペレーティングシステム15、コホート識別モジュール16、および他の機能モジュール18を含み得る。オペレーティングシステム15は、システム10のためにオペレーティングシステム機能を提供し得る。コホート識別モジュール16は、以下により詳細に記載されるように、コホートを識別するための機能を提供し得る。ある実施例では、コホート識別モジュール16は複数のモジュールを含み得、各モジュールはコホートを識別するための具体的な個々の機能を提供する。システム10は、さらに、より大きなシステムの一部であり得る。したがって、システム10は、さらに他の機能を含むよう、1つ以上のさらに他の機能モジュール18を含み得る。たとえば、機能モジュール18は、オラクル社(Oracle Corporation)からのOracle Argus Insight製品のモジュールのような、さらに他の機能を提供するモジュールを含んでもよい。

【0016】

プロセッサ22は、さらに、バス12を介してデータベース34に作動的に結合され得る。データベース34は、論理的に関係付けられるレコードまたはファイルの統合された集まりにデータを記憶し得る。データベース34は、オペレーショナルデータベース、解析データベース、データウェアハウス、分散型データベース、エンドユーザデータベース、外部データベース、ナビゲーションデータベース、メモリ内データベース、ドキュメント指向型データベース、リアルタイムデータベース、リレーショナルデータベース、オブジェクト指向型データベース、または技術分野において公知の任意の他のデータベースであり得る。さらに、データベース34は、APIを介してアクセスされ得、照会言語をサポートし得る。

【0017】

図2は、この発明の実施例に従う、相互運用可能なケースシリーズシステム200のブロック図である。相互運用可能なケースシリーズシステム200は、ケースシリーズリポジトリ210と、ケースシリーズデータモデル220と、ケースシリーズAPI230とを含み得る。ある実施例では、相互運用可能なケースシリーズシステム200は、さらに、1つ以上のケースシリーズをブラウジングし操作するための機能を提供するユーザインターフェイスコンポーネント(図2には示されず)を含み得る。

【0018】

この実施例によれば、ケースシリーズリポジトリ210は、1つ以上のケースシリーズのようなデータを記憶し得るリポジトリである。たとえば、ケースシリーズが1つ以上のケース識別子を含み、(各識別子は、ケースシリーズ内に含まれる各有害事象ケースを一意に識別し、)ケースシリーズは、さらに、ケースシリーズ内に含まれる1つ以上の有害事象ケースをともに表現するケースデータおよび/またはケースメタデータを含む実施例においては、ケースシリーズリポジトリ210は、1つ以上のケース識別子ならびに関連付けられるケースデータおよび/またはケースメタデータを記憶し得る。さらに、各ケースシリーズ毎に、ケースシリーズリポジトリ210は、さらに、以下にさらに詳細に記載される、ケースシリーズ履歴情報およびケースリビジョン情報などのような、各ケースシリーズに係付けられる情報を記憶し得る。ケースシリーズリポジトリ210は、データベースまたはファイルなどのような、データを記憶し得る任意のタイプのリポジトリであり得る。

【0019】

さらに、実施例に従うと、ケースシリーズデータモデル 220 は、ケースシリーズの正規表現、およびケースシリーズに関係付けられる情報を含み得るデータモデルである。たとえば、ケースシリーズが、1つ以上のケース識別子を含み、さらに、ケースシリーズ内に含まれる1つ以上の有害事象ケースとともに表現するケースデータおよび/またはケースメタデータを含む実施例においては、ケースシリーズデータモデル 220 は、ケース識別子を表現するデータフィールド、ならびにケースデータおよび/またはケースメタデータを表現する1つ以上のさらに他のデータフィールドを含み得る。

【0020】

さらに、ケースシリーズが、ケースシリーズに関係付けられる情報を含む場合には、ケースシリーズデータモデル 220 は、さらに、ケースシリーズに関係付けられる情報を表現する1つ以上のさらに他のデータフィールドを含み得る。ある実施例では、ケースシリーズに関係付けられる情報は、ケースシリーズ履歴情報を含み得る。ケースシリーズ履歴情報は、ケースシリーズを生成した個人、ケースシリーズを生成した機構、ケースシリーズを修正した1人以上の個人などのような、ケースシリーズの履歴に関する情報を含み得る。ある実施例では、ケースシリーズ履歴情報は、ケースシリーズに関連付けられ得る1つ以上の変更ログのフォーマットに記憶され得る。

【0021】

他の実施例では、ケースシリーズに関係付けられる情報はケースリビジョン情報を含み得る。ケースリビジョン情報は、ケースシリーズの1つ以上のリビジョンに関する情報を含み得る。ある実施例によれば、リビジョンは、ケースシリーズの有害事象ケースに対する任意の変更である。したがって、この実施例によれば、ケースシリーズは1つ以上のケースリビジョンを含み得る。さらに、各ケースリビジョンは、ケースリビジョンを作成したパートナーアプリケーションによって識別され得る。

【0022】

ケースシリーズの1つ以上のリビジョンに関する情報を含むことに加えて、またはその代替として、ケースリビジョン情報は、ケースシリーズの1つ以上のバージョンに関する情報を含み得る。ある実施例によれば、あるバージョンは、ある品質解析サイクルを経たケースシリーズの有害事象ケースに対する変更であり、科学的解析に対する準備ができているものとして識別される。一例においては、ある形成ソフトウェアアプリケーションは、ケースに対するインプロセスリビジョンを含んでもよいが、消費ソフトウェアアプリケーションは、最終バージョンでのみワークをサポートしてもよい。ケースシリーズデータモデル 220 は、ケースシリーズ API 230 との関連において、ソフトウェアアプリケーションがケースリビジョン情報を解釈し、それが消費ソフトウェアアプリケーションにおいて最も正確に用いられ得るようにすることを可能にし得る。したがって、ケースシリーズは、さらに、ケースリビジョンシリーズとしても識別され得る。さらに、ケースリビジョン情報は、同じケースの、複数のリビジョンを含み得る。ある実施例においては、ケースリビジョン情報は、ケースシリーズに関連付けられ得る1つ以上のリビジョンのフォーマットにおいて記憶され得る。ケースシリーズデータモデル 220 の一例を、さらに、図3との関連において記載する。

【0023】

実施例によれば、ケースシリーズ API 230 は、ケースシリーズデータモデル 220 を露出させ得、ケースシリーズデータモデル 220 を、ソフトウェアアプリケーションに対し、そのソフトウェアアプリケーションによって指定されるフォーマットに基づいて表現し得る API である。したがって、ケースシリーズ API 230 は、ソフトウェアアプリケーションがケースシリーズリポジトリ 210 とインターフェイスすることを可能にし得る。ある実施例によれば、ケースシリーズ API 230 は、1つ以上のケースシリーズを形成、消費、検索および/または更新するための機能を与え得る。さらに、一実施例では、第1のソフトウェアアプリケーションは、ケースシリーズ API 230 を用いて、ケースシリーズを形成し、そのケースシリーズをケースシリーズリポジトリ 210 に記憶し得る。実施例によれば、第2のソフトウェアアプリケーションは、ケースシリーズ API

230を用いて、第1のソフトウェアアプリケーションにより形成されケースシリーズリポジトリ210に記憶されたケースシリーズを消費し得る。したがって、第1のソフトウェアアプリケーションにケースシリーズをエクスポートすることを求め、第2のソフトウェアアプリケーションにケースシリーズをインポートすることを求めるのではなく、2つのソフトウェアアプリケーションはケースシリーズAPI230を用いてケースシリーズリポジトリ210とインターフェイスし得る。

【0024】

ある実施例によれば、ケースシリーズリポジトリ210、ケースシリーズデータモデル220、およびケースシリーズAPI230は、4種類のケースシリーズ：(1)名称付きケースシリーズ；(2)アクティブなユーザケースシリーズ；(3)単一使用ケースシリーズ；および(4)ケースヒットリストをサポートする。名称付きケースシリーズは明示的な一意の名称を含むケースシリーズである。名称は、ユーザ、または形成ソフトウェアアプリケーションの機能もしくは実行可能なプロセスによって、名称付きケースシリーズに与えられ得る。さらに、あるソフトウェアアプリケーションは名称付きケースシリーズを名称によって要求し得る。ケースシリーズリポジトリ210は、名称付きケースシリーズ上において検索機能またはブラウジング機能をサポートし得る。アクティブなケースシリーズは、ユーザに関連付けられるケースシリーズである。あるアクティブなケースシリーズは、そのアクティブなケースシリーズが関連付けられるユーザに対して名称付けられ得る。あるアクティブなケースシリーズのコンテンツは、それが上書きされるまで存在し得る。したがって、アクティブなケースシリーズは、複数のソフトウェアアプリケーションに渡り得る個人的なワークスペースの形成を可能にし得る。単一使用ケースシリーズは、単一のレポートを実行する単一の目的に対してケースシリーズリポジトリ210内に存在し得るケースシリーズである。あるトランザクションが完了した後、単一使用ケースシリーズは削除され得る。ケースシリーズリポジトリ210は、ケースシリーズAPI230とは別のAPIを通して、レポートの実行をオーケストレーションし得る。ケースヒットリストは、形成ソフトウェアアプリケーションによって完全に管理され得るケースシリーズである。ケースヒットリストがケースシリーズリポジトリ210に記憶されると、ケースヒットリストは、形成ソフトウェアアプリケーションにのみ知られている識別を与えられ得る。ケースヒットリストは、名称付けられたケースシリーズのリストには現われず、形成ソフトウェアアプリケーションによってその識別情報を渡される消費側に対してアクセス可能であり得る。

【0025】

加えて、一実施例においては、ケースシリーズリポジトリ210、ケースシリーズデータモデル220、およびケースシリーズAPI230は、ケースシリーズに非常に類似している他のシリーズ：(1)事象シリーズ；および(2)製品シリーズをサポートする。薬剤安全性システムなどのようなデータソース上において実行され得るある特定の有害事象に対するクエリーは、ケースシリーズリポジトリ210内において記憶され得るケースシリーズを形成し得る。ケースシリーズにおける各ケースは、クエリーと一致する少なくとも1つの事象を有し得る。しかしながら、各ケースは、さらに、クエリーと一致しないさらに他の事象も含んでもよい。したがって、ケースシリーズは、クエリーと一致するすべてのケース、およびクエリーと一致するか否かに係わらず、それらのケースと関係付けられる事象のすべてを含み得る。対照的に、データソース上において実行されるクエリーから形成される事象シリーズは、そのクエリーと一致するすべてのケースを含み得るが、同様にそのクエリーと一致する、それらのケースに關係付けられる事象しか含み得ない。さらに、ある特定の医療製品または製薬製品に対するクエリーも、ケースシリーズを形成し得る。ケースシリーズにおける各ケースは、そのクエリーと一致する少なくとも1つの製品を有し得る。しかしながら、各ケースは、さらに、そのクエリーと一致しないさらに他の製品も含んでもよい。したがって、ケースシリーズは、クエリーに一致するすべてのケース、およびクエリーに一致するか否かに係わらず、それらのケースに關係付けられる製品のすべてを含み得る。対照的に、データソース上において実行されるクエリーから形

10

20

30

40

50

成される製品シリーズは、そのクエリーに一致するすべてのケースを含み得るが、同様にそのクエリーに一致する、それらのケースに関係付けられる製品しか含み得ない。加えて、名称付き事象シリーズ、名称付き製品シリーズ、アクティブな事象シリーズ、アクティブな製品シリーズ、単一使用事象シリーズ、単一使用製品シリーズ、事象ヒットリストおよび製品ヒットリストは、先に記載された、名称付きケースシリーズ、アクティブなケースシリーズ、単一使用ケースシリーズ、およびケースヒットシリーズに対する有効な変形例であり、それらと同様の態様において働き得る。

【 0 0 2 6 】

ある実施例によれば、ケースシリーズ A P I 2 3 0 は、ケースシリーズ、事象シリーズまたは製品シリーズ上において以下の機能：(a) シリーズを閲覧する；(b) シリーズをセーブする；(c) シリーズをアクティブにする；(d) アクセス権をシリーズに割当てる；(e) あるケースをシリーズに追加する；(f) あるケースをシリーズから削除する；(g) シリーズを削除する；(h) ケースに注釈を付ける；(i) シリーズに注釈を付ける；(j) シリーズをエクスポートする；(k) シリーズを凍結する；および(l) 2つのシリーズをマージする、を実行し得る。前述の機能をさらに詳細に記載する。

【 0 0 2 7 】

ユーザは、1つ以上のケース識別子を含むシリーズに記憶される情報、ケースリビジョン情報、ケースシリーズ履歴情報、シリーズに関係付けられる任意の他の情報、シリーズを作成したクエリー基準、またはシリーズの任意の他のプロパティおよび/もしくはメタデータを閲覧し得る。閲覧は、検索およびソート機能を含み得る。ユーザは、さらに、シリーズをセーブし、それに名称を与え、それを名称付きシリーズにし得る。ユーザはシリーズをアクティブなシリーズにし得る。シリーズを作成したユーザは、読出アクセスおよび/または書込アクセスなど、アクセス権をシリーズに割当て得る。加えて、ユーザはあるケースをあるシリーズに追加し得る。ユーザは、さらに、あるケースをあるシリーズから削除し得る。さらに、ユーザはあるシリーズをケースシリーズリポジトリ 2 1 0 から削除し得る。さらに、ユーザは、テキスト注釈を、あるシリーズにおけるあるケースに対して、そのシリーズのコンテキストにおいて、形成し得る。同じケースが他のシリーズに現われる場合には、さまざまなシリーズにおけるそのケースに対する注釈は別であり得る。ユーザは、さらに、テキスト注釈をシリーズレベルで形成し得る。加えて、ユーザはシリーズをファイルにエクスポートし得る。さらに、ユーザはシリーズを凍結させ得、そのシリーズにおけるケースデータは、それが凍結された日時後は、たとえ、ケースが、薬剤安全性システムなどのような、ケースデータを含む対応のデータソースにおいて更新される場合であっても、変わらない。ユーザは、さらに、シリーズを解凍し得、ケースデータは、再び、利用可能な最も現在のリビジョンを反映し得る。ユーザは、さらに、ユニオンを用いて2つのシリーズをマージし、オペレーションをインタセクトまたはマイナスし、およびしたがって新たなシリーズを作成し得る。

【 0 0 2 8 】

ケースシリーズ A P I 2 3 0 は、さらに、実施例に従って、形成機能、消費機能、検索機能および更新機能を実行する機能を提供し得る。形成機能を実行する際には、ケースシリーズ A P I 2 3 0 は、あるシリーズを受取り、そのシリーズをケースシリーズリポジトリ 2 1 0 内に記憶する。消費機能を実行する際には、ケースシリーズ A P I 2 3 0 は、あるシリーズをケースシリーズリポジトリ 2 1 0 から検索し、そのシリーズをソフトウェアアプリケーション内において実現し得る（たとえばそのシリーズをそのソフトウェアアプリケーションのユーザインターフェイス内において表示するなど）。検索機能を実行する際には、ケースシリーズ A P I 2 3 0 は、ケースシリーズリポジトリ 2 1 0 内に記憶されるシリーズを検索し得る。更新機能を実行する際には、ケースシリーズ A P I 2 3 0 は、ケースシリーズリポジトリ 2 1 0 内に記憶されるシリーズを更新し得る。

【 0 0 2 9 】

ある実施例においては、ある実施例は新たなケースシリーズのケースシリーズリポジトリ 2 1 0 への追加を、薬剤安全性システムなどのようなデータソース上においてクエリー

10

20

30

40

50

を実行することによって行ない得、クエリーによって返される１つ以上のケースはケースシリーズリポジトリ 210 内においてケースシリーズとして記憶され得る。加えて、これらの実施例のうちのいくつかにおいては、ユーザは、新たなシリーズのケースシリーズリポジトリストアに対する追加を、クエリーをデータソース上において実行する以外の態様で行ない得る。より具体的には、ユーザは新たなシリーズの追加を：（a）シリーズを入力すること；または（b）シリーズをインポートすることによって行ない得る。シリーズを入力することによって、ユーザは、手動で１つ以上のケース識別子をケースシリーズリポジトリ 210 内に入力して、１つ以上の新たなシリーズを作成し得る。代替的に、ユーザは、新たなシリーズを、１つ以上のケース識別子を含むファイルから、ケースシリーズリポジトリ 210 にインポートし得る。

10

【0030】

図 3 は、この発明のある実施例に従う、ケースシリーズデータモデル 300 のブロック図を示す。ある実施例では、ケースシリーズデータモデル 300 は、図 2 のケースシリーズデータモデル 220 と同一である。先に論じたように、ケースシリーズデータモデル 300 は、１つ以上のケースシリーズの正規表現、および各ケースシリーズに関係付けられる情報、たとえば各ケースシリーズに関係付けられるケースシリーズ履歴情報、および各ケースシリーズに関係付けられるケースリビジョン情報などを含む。以下により詳細に記載されるように、ケースシリーズデータモデル 300 は複数のデータフィールドを含み得、各データフィールドは（図 2 のケースシリーズリポジトリ 210 のような）ケースシリーズリポジトリのデータフィールドを表現し得、各データフィールドは（図 2 のケースシリーズ API 230 のような）ケースシリーズ API によってそれ自体の一意のフォーマットにおいて表現され得る。

20

【0031】

実施例によれば、ケースシリーズデータモデル 300 はケースシリーズ 310 を含む。ケースシリーズ 310 は、１つ以上のケースシリーズの正規表現である。ある実施例においては、ケースシリーズ 310 は複数のデータフィールドを含み、各データフィールドは、１つ以上のケースシリーズの各ケースシリーズのケースデータまたはメタデータを記憶し得る。これらの実施例のいくつかにおいては、ケースシリーズ 310 は、ケースシリーズのうちのあるケースのあるケース識別子を記憶し得るデータフィールドと、ケースシリーズ内に含まれる１つ以上の有害事象ケースを表現するケースデータまたはメタデータを記憶し得る１つ以上のデータフィールドとを含む。したがって、ケースシリーズ 310 はケースシリーズの各ケース毎にケース識別子を記憶し得、ケースシリーズ 310 はケースシリーズの各ケースに関係付けられるデータおよび／またはメタデータを記憶し得る。したがって、これらの実施例においては、ケースシリーズ 310 は、複数のデータフィールド内に複数の値を記憶することによって、１つ以上のケースシリーズを表現し得る。

30

【0032】

ケースシリーズデータモデル 300 は、さらに、変更ログ 320 を含む。変更ログ 320 は、ケースシリーズに関係付けられるケースシリーズ履歴情報の正規表現である。先に記載されたように、ケースシリーズ履歴情報は、ケースシリーズを生成した個人、ケースシリーズを生成した機構、ケースシリーズを修正した１人以上の個人などのような、ケースシリーズの履歴に関する情報を含み得る。ある実施例では、変更ログ 320 は１つ以上のデータフィールドを含み、各データフィールドは、１つ以上のケースシリーズの各ケースシリーズのケースシリーズ履歴情報を記憶し得る。これらの実施例に従うと、各データフィールドは、ケースシリーズ履歴情報の、異なる構成要素を表現する値を記憶し得る。たとえば、変更ログ 320 の第１のデータフィールドは、ケースシリーズを生成した個人の名前を記憶し得、第２のデータフィールドは、ケースシリーズを生成した機構の名称を記憶し得、第３のデータフィールドは、ケースシリーズを修正した第１の個人の名前を記憶し得、第４のデータフィールドは、ケースシリーズを修正した第２の個人の名前を記憶し得る、などである。ある実施例によれば、ケースシリーズ 310 および変更ログ 320 は１対多数の関係を有し得、１つ以上の変更ログが、あるケースシリーズに関連付けられ

40

50

得る。一実施例では、変更ログ320は、キャラクタ・ラージ・オブジェクト(「CLOB」)として実現され得、変更ログ320の各値はCLOBの終わりに付加され得る。

【0033】

ケースシリーズデータモデル300は、さらに、ケースリビジョン330を含む。ケースリビジョン330は、1つ以上のケースシリーズに関係付けられるケースリビジョン情報の正規表現である。より具体的には、ある実施例においては、ケースシリーズは、1つ以上のケースリビジョンを含むコンテナである。あるシナリオにおいては、ケースシリーズは、同じケースの複数のケースリビジョンを含み得る。他のシナリオでは、ケースシリーズは、そのケースシリーズの各ケース毎に1つのケースリビジョンを含む。先に記載されたように、ケースリビジョン情報は、ケースシリーズの1つ以上のリビジョンおよび/またはケースシリーズの1つ以上のバージョンに関する情報を含み、リビジョンは、ケースシリーズの有害事象ケースに対する任意の変更であり得、バージョンは、規定されたレビュープロセスに従って検証されたケースシリーズの有害ケースに対する変更であり得る。ある実施例では、ケースリビジョン330は1つ以上のデータフィールドを含み、各データフィールドは、1つ以上のケースシリーズの各ケースシリーズのケースリビジョン情報を記憶し得る。これらの実施例によれば、各データフィールドは、ケースリビジョン情報の、異なる構成要素を表現する値を記憶し得る。ある実施例では、ケースシリーズのリビジョンおよび/またはバージョンは、元のケースシリーズのフォーマットと同一のフォーマットで表現され得る。したがって、これらの実施例では、ケースリビジョン330は、ケースシリーズのうちのあるケースのあるケース識別子を記憶し得るデータフィールドを含み、そのケースシリーズのそのケースのケースデータまたはメタデータを記憶し得る1つ以上のデータフィールドを含み、ケースデータまたはメタデータはそのケースシリーズの有害事象ケースに対する変更を含む。他の実施例では、ケースシリーズのリビジョンおよび/またはバージョンは、そのケースシリーズの有害事象ケースに対する変更を含むのみであるフォーマットにおいて表現され得る。したがって、これらの実施例では、ケースリビジョン330はケースシリーズの有害事象ケースに対する変更を記憶する1つ以上のデータフィールドを含む。ある実施例によれば、ケースシリーズ310およびケースリビジョン330は1対多数の関係を有し得、1つ以上のケースリビジョンが、あるケースシリーズと関連付けられ得る。ある実施例では、ケースリビジョン330は、さらに、2つのさらに他のデータフィールド、有効開始日/時データフィールドおよび有効終了日/時データフィールドによって表現され得るタイムスタンプ情報を含む。タイムスタンプ情報は、ケースシリーズのリビジョンまたはバージョンが有効である開始日および/または時間ならびに終了日および/または時間を表現し得る。さらに、有効開始日/時および有効終了日/時は、両方とも、ケースリビジョン情報とともにケースシリーズの形成を可能にし得るソースデータの特徴である。タイムスタンプ情報は、全体的または部分的のいずれでも、ケースシリーズのリビジョンまたはバージョンを識別し得る。

【0034】

ケースシリーズデータモデル300は、さらに、注釈340を含む。注釈340は、ケースシリーズの1つ以上のケースリビジョンに関係付けられる注釈情報の正規表現である。注釈情報は任意のユーザ定義情報を含み得、ユーザ定義情報は、ケースシリーズのケースリビジョンに注釈を付けるよう働き得る。ある実施例においては、注釈340は1つ以上のデータフィールドを含み、各データフィールドはユーザ定義値を記憶し得る。ある実施例によれば、ケースリビジョン330および注釈340は1対多数の関係を有し得、1つ以上の注釈が、あるケースリビジョンに関連付けられ得る。

【0035】

ケースシリーズデータモデル300は、さらに、フォルダ350を含む。フォルダ350は、1つ以上のケースシリーズの論理的グループ化の正規表現である。経時的に、ユーザは、ケースシリーズデータモデル300を用いて、何千ものケースシリーズを生成し得る。入れ子状のフォルダ記憶システムを用いてケースシリーズを編成し得る。したがって、1つ以上のケースシリーズが、あるフォルダに関連付けられ得、1つ以上のフォルダが

10

20

30

40

50

、あるフォルダ内において入れ子にされ得る。したがって、ある実施例によれば、フォルダ 350 およびケースシリーズ 310 は 1 対多数の関係を有し得、1 つ以上のケースシリーズが、あるフォルダに関連付けられ得る。一実施例においては、フォルダ 350 は、ケースシリーズデータモデル 300 の一部ではなく、ケースシリーズデータモデル 300 を用いて生成される 1 つ以上のケースシリーズを 1 つ以上のフォルダに編成するために利用されるドキュメント管理システムのフォルダ機能の表現である。

【0036】

図 4 は、この発明のある実施例に従う、相互運用可能なケースシリーズシステムの例示的実現例のブロック図を示す。より具体的には、図 4 は、図 2 の相互運用可能なケースシリーズシステム 200 などのような、複数のソフトウェアアプリケーションと対話する、相互運用可能なケースシリーズシステムの例を示す。この実現例はケースシリーズリポジトリ 400 を含む。先に記載されたように、ケースシリーズリポジトリ 400 は、1 つ以上のケースシリーズ、および / または 1 つ以上のケースシリーズに関係付けられる情報などのような、データを記憶し得るリポジトリである。ある実施例では、ケースシリーズリポジトリ 400 は、図 2 のケースシリーズリポジトリ 210 と同一である。この実現例は、さらに、ケースシリーズデータモデル 410 を含む。同様に先に記載されたように、ケースシリーズデータモデル 410 は、ケースシリーズの正規表現、およびそのケースシリーズに関係付けられる情報を含み得るデータモデルである。ある実施例においては、ケースシリーズデータモデル 410 は、図 2 のケースシリーズデータモデル 220 および図 3 のケースシリーズデータモデル 300 と同一である。実施例によれば、ケースシリーズデータモデル 410 は、ケースシリーズリポジトリ 400 内に記憶されるデータを表現するデータモデルであり得る。一実施例では、ケースシリーズデータモデル 410 は複数のデータフィールドを含み得、複数のデータフィールドは、ケースシリーズリポジトリ 400 内に含まれる複数のデータフィールドを表現する。

【0037】

この実現例は、さらに、データマイニングアプリケーション 420 およびデータマイニングアプリケーションケースシリーズ API 430 を含む。実施例によれば、データマイニングアプリケーション 420 は、薬剤安全性データ内におけるケースの、1 つ以上の関係付けられるグループを見つけ出すために、データマイニング機能を実行し得る 1 つ以上の実行可能なプロセスを含むソフトウェアアプリケーションである。データマイニングアプリケーション 420 は、1 つ以上のデータマイニングアルゴリズムを用いて、1 つ以上のケースシリーズを形成し得る。データマイニングアプリケーション 420 は、さらに、別のソフトウェアアプリケーションによって形成される 1 つ以上のケースシリーズを消費し得る。一実施例においては、データマイニングアプリケーションはオラクル社 (Oracle Corporation) からの "Empirica Signal" 製品である。

【0038】

さらに、実施例によれば、データマイニングアプリケーションケースシリーズ API 430 は、ケースシリーズデータモデル 410 をデータマイニングアプリケーション 420 に露出し、ケースシリーズデータモデル 410 を、データマイニングアプリケーション 420 に対して、データマイニングアプリケーション 420 によって指定されるフォーマットに基づいて表現し、したがって、データマイニングアプリケーション 420 がケースシリーズリポジトリ 400 とインターフェイスすることを可能にするインターフェイスを提供する。したがって、データマイニングアプリケーション 420 は、データマイニングアプリケーションケースシリーズ API 430 を用いて、1 つ以上のケースシリーズを形成し得、1 つ以上の形成されたケースシリーズをケースシリーズリポジトリ 400 内に記憶し得る。同様に、データマイニングアプリケーション 420 は、データマイニングアプリケーションケースシリーズ API 430 を用いて、1 つ以上のケースシリーズをケースシリーズリポジトリ 400 内から検索し得、その 1 つ以上の検索されたケースシリーズを消費し得る。ある実施例においては、データマイニングアプリケーションケースシリーズ API 430 は、図 2 のケースシリーズ API 230 のコンポーネントを表現する。

【 0 0 3 9 】

この実現例は、さらに、報告アプリケーション 4 4 0 および報告アプリケーションケースシリーズ A P I 4 5 0 を含む。実施例によれば、報告アプリケーション 4 4 0 は、1 つ以上のケースシリーズを視覚化する 1 つ以上のレポートを生成するよう報告機能を実行し得る 1 つ以上の実行可能なプロセスを含むソフトウェアアプリケーションである。報告アプリケーション 4 4 0 は、1 つ以上の報告アルゴリズムを用いて 1 つ以上のケースシリーズを形成し得る。報告アルゴリズム 4 4 0 は、さらに、別のソフトウェアアプリケーションによって形成される 1 つ以上のケースシリーズを消費し得る。一実施例では、報告アプリケーション 4 4 0 はオラクル社 (Oracle Corporation) からの "Oracle Argus Insight" 製品である。

10

【 0 0 4 0 】

さらに、実施例によれば、報告アプリケーションケースシリーズ A P I 4 5 0 は、ケースシリーズデータモデル 4 1 0 を報告アプリケーション 4 4 0 に露出し、ケースシリーズデータモデル 4 1 0 を、報告アプリケーション 4 4 0 に対して、報告アプリケーション 4 4 0 によって指定されるフォーマットに基づいて表現し、およびしたがって、報告アプリケーション 4 4 0 がケースシリーズリポジトリ 4 0 0 とインターフェイスすることを可能にするインターフェイスを提供する。したがって、報告アプリケーション 4 4 0 は、報告アプリケーションケースシリーズ A P I 4 5 0 を用いて、1 つ以上のケースシリーズを形成し得、その 1 つ以上の形成されたケースシリーズをケースシリーズリポジトリ 4 0 0 内に記憶し得る。同様に、報告アプリケーション 4 4 0 は、報告アプリケーションケースシリーズ A P I 4 5 0 を用いて、ケースシリーズリポジトリ 4 0 0 内から 1 つ以上のケースシリーズを検索し得、その 1 つ以上の検索されたケースシリーズを消費し得る。さらに、報告アプリケーションケースシリーズ A P I 4 5 0 は、単に、報告アプリケーション 4 4 0 のレポートにおけるケースシリーズの使用を単純にし得、報告アプリケーション 4 4 0 のレポートがクエリーを実行しその結果のケースシリーズを用いることを可能にし得、所望される場合には、その結果として得られたケースシリーズをさらに他の使用のためにケースシリーズリポジトリ 4 0 0 に記憶し得る。ある実施例では、報告アプリケーションケースシリーズ A P I 4 5 0 は、図 2 のケースシリーズ A P I 2 3 0 のコンポーネントを表現する。

20

【 0 0 4 1 】

したがって、実施例によれば、報告アプリケーション 4 4 0 が、データマイニングアプリケーション 4 2 0 によって形成される 1 つ以上のケースシリーズにアクセスし得、データマイニングアプリケーション 4 2 0 が、報告アプリケーション 4 4 0 によって形成される 1 つ以上のケースシリーズにアクセスし得る中、データマイニングアプリケーション 4 2 0 は報告アプリケーション 4 4 0 と対話し得、およびその逆も可能である。当業者であれば、データマイニングアプリケーション 4 2 0 および報告アプリケーション 4 4 0 は、実施例に従ってケースシリーズを形成および消費するソフトウェアアプリケーションの例であり、代替的实施例においては、データマイニングアプリケーション 4 2 0 および報告アプリケーション 4 4 0 は、代替的機能を含む他のソフトウェアアプリケーションと置換され得ることを容易に理解するであろう。さらに、任意の数のソフトウェアアプリケーションをサポートし、任意の数のソフトウェアアプリケーションがケースシリーズデータモデル 4 1 0 を用いてケースシリーズリポジトリ 4 0 0 にアクセスすることを可能にし得る、任意の数のケースシリーズ A P I が存在し得る。

30

40

【 0 0 4 2 】

図 5 は、この発明のある実施例に従って、相互運用可能なケースシリーズモジュールの機能のフロー図を示す。一実施例においては、図 5 のフロー図の機能 (以下に記載する) は、図 8 のフロー図の機能 (これも以下に記載する) と並んで、各々、メモリまたは何らかの他のコンピュータ読取可能もしくは有形の媒体に記憶され、プロセッサによって実行されるソフトウェアによって実現される。他の実施例では、各機能は、ハードウェアによって (たとえばアプリケーション特化集積回路 (「A S I C」)、プログラマブルゲート

50

アレイ（「PGA」）、フィールドプログラマブルゲートアレイ（「FPGA」）などの使用を通して）またはハードウェアとソフトウェアとの任意の組合せによって実行されてもよい。

【0043】

フローは、開始後、510に進む。510では、ケースシリーズが受取られる。ある実施例では、ケースシリーズはパートナーアプリケーションから受取られる。ケースシリーズは1つ以上の有害事象ケースを含み、各有害事象ケースは有害事象を表現するデータレコードを含む。ケースシリーズは、名称付きケースシリーズ、アクティブなユーザケースシリーズ、単一使用ケースシリーズ、またはケースヒットリストであり得る。ケースシリーズは、さらに、事象シリーズまたは製品シリーズであり得る。ある実施例では、有害事象を表現する各データレコードは薬剤安全性データをさらに含み、薬剤安全性データは、1つ以上の薬剤の安全性に関係付けられる1つ以上のレポートまたは患者識別子を含む。ある実施例では、ケースシリーズは複数のケースリビジョンを含み得る。フローは520に進む。

10

【0044】

520では、ケースシリーズに関係付けられる情報が受取られる。ある実施例ではケースシリーズはパートナーアプリケーションから受取られる。ある実施例ではケースシリーズに関係付けられる情報はケースリビジョン情報を含み、ケースリビジョン情報は、ケースシリーズの有害事象ケースに対する少なくとも1つの変更を含み得る。これらの実施例では、ケースリビジョン情報を含むケースリビジョンが受取られる。ある実施例では、ケースシリーズの有害事象ケースに対する少なくとも1つの変更が、規定されたレビュープロセスに従って検証され、ケースリビジョン情報を含むケースバージョンが受取られる。さらに、ある実施例では、ケースリビジョン情報は、ケースリビジョンを識別するタイムスタンプ情報を含む。タイムスタンプ情報は、有効開始日および/または時間ならびに有効終了日および/または時間を含み得る。

20

【0045】

他の実施例では、ケースシリーズに関係付けられる情報はケースシリーズ履歴情報を含み、ケースシリーズ履歴情報は、ケースシリーズを生成した個人、ケースシリーズを生成した機構、ケースシリーズを修正した1人以上の個人などのような、ケースシリーズの履歴に関する情報を含み得る。これらの実施例においては、ケースシリーズ履歴情報を含む変更ログが作成される。他の実施例においては、ケースシリーズに関係付けられる情報はユーザ定義情報を含む。これらの実施例では、ユーザ定義情報を含む注釈が作成される。他の実施例では、ケースシリーズに関係付けられる情報は、ケースシリーズおよび1つ以上のさらに他のケースシリーズの論理的編成を含む。これらの実施例においては、ケースシリーズおよび1つ以上のさらに他のケースシリーズの論理的編成を含むフォルダが形成される。フローは530に進む。

30

【0046】

530では、ケースシリーズ、およびケースシリーズに関係付けられる情報は、ケースシリーズデータモデルを用いてケースシリーズリポジトリ内に記憶される。ケースシリーズリポジトリは、1つ以上のケースシリーズおよびその1つ以上のケースシリーズに関係付けられる情報などのようなデータを記憶し得るリポジトリである。ケースシリーズデータモデルは、ケースシリーズリポジトリ内におけるケースシリーズおよびそのケースシリーズに関係付けられる情報のフォーマットを規定するケースシリーズの正規表現である。

40

【0047】

ケースシリーズに関係付けられる情報がケースリビジョン情報を含む実施例においては、ケースシリーズデータモデルは、ケースシリーズリポジトリ内においてケースリビジョンのフォーマットを規定する。ケースシリーズに関係付けられる情報がケースシリーズ履歴情報を含む実施例では、ケースシリーズデータモデルは、ケースシリーズリポジトリ内において変更ログのフォーマットを規定する。ケースシリーズに関係付けられる情報がユーザ定義情報を含む実施例では、ケースシリーズデータモデルはケースシリーズリポジトリ

50

リ内において注釈のフォーマットを規定する。ケースシリーズに關係付けられる情報がケースシリーズおよび1つ以上のさらに他のケースシリーズの論理的編成を含む実施例では、ケースシリーズデータモデルはケースシリーズリポジトリ内においてフォルダのフォーマットを規定する。

【0048】

ある実施例では、ケースシリーズデータモデルは、ケースシリーズの1つ以上のケース識別子を表現するデータフィールドと、ケースシリーズの1つ以上の有害事象ケースを表現する1つ以上のさらに他のデータフィールドとを含む。これらの実施例では、各ケース識別子は、1つ以上の有害事象ケースのうちのある有害事象ケースを一意に識別する。さらに、これらの実施例にいくつかにおいては、ケースシリーズデータモデルは、ケースシリーズに關係付けられる情報を表現する1つ以上のさらに他のフィールドを含む。

10

【0049】

ケースシリーズに關係付けられる情報がケースリビジョン情報を含む実施例では、ケースシリーズデータモデルは、ケースリビジョン情報を表現する1つ以上のさらに他のデータフィールドを含む。ケースリビジョン情報がタイムスタンプ情報を含む実施例では、ケースシリーズデータモデルは、タイムスタンプ情報を表現する1つ以上のさらに他のデータフィールドを含む。ケースシリーズに關係付けられる情報がケースシリーズ履歴情報を含む実施例では、ケースシリーズデータモデルは、ケースシリーズ履歴情報を表現する1つ以上のさらに他のデータフィールドを含む。ケースシリーズに關係付けられる情報がユーザ定義情報を含む実施例では、ケースシリーズデータモデルは、ユーザ定義情報を表現する1つ以上のさらに他のデータフィールドを含む。ケースシリーズに關係付けられる情報がケースシリーズと1つ以上のさらに他のケースシリーズとの論理的編成を含む実施例では、ケースシリーズデータモデルは、ケースシリーズおよび1つ以上のさらに他のケースシリーズの論理的編成を表現する1つ以上のさらに他のデータフィールドを含む。フローは540に進む。

20

【0050】

540では、ケースシリーズに關係付けられる情報は、ケースシリーズデータモデルを用いて、ケースシリーズ内に關連付けられる。ケースシリーズに關係付けられる情報がケースリビジョン情報を含む実施例では、ケースリビジョンが、ケースシリーズデータモデルを用いて、ケースシリーズと關連付けられる。ケースシリーズに關係付けられる情報がケースシリーズ履歴情報を含む実施例においては、変更ログが、ケースシリーズデータモデルを用いて、ケースシリーズと關連付けられる。ケースシリーズに關係付けられる情報がユーザ定義情報を含む実施例では、注釈が、ケースシリーズデータモデルを用いて、ケースリビジョンと關連付けられる。ケースシリーズに關係付けられる情報がケースシリーズと1つ以上のさらに他のケースシリーズとの論理的編成を含む実施例では、ケースシリーズはフォルダと關連付けられる。フローは550に進む。

30

【0051】

550では、ケースシリーズ、およびケースシリーズに關係付けられる關連情報が、ケースシリーズAPIを用いて、ケースシリーズリポジトリから検索される。ケースシリーズAPIは、ソフトウェアアプリケーションに対して、そのソフトウェアアプリケーションによって指定されるフォーマットに基づいて、ケースシリーズデータモデルを表現するAPIである。したがって、ケースシリーズAPIは、ソフトウェアアプリケーションのために、ケースシリーズおよびそのケースシリーズに關係付けられる情報のフォーマットを規定し得る。ケースシリーズAPIは、ケースシリーズデータモデルを用いて、ケースシリーズおよび關連の情報をケースシリーズリポジトリから検索し得る。フローは次いで終了する。

40

【0052】

図6は、この発明のある実施例に従って、コホート識別システム600のブロック図を示す。図6において網掛けされる、コホート識別システム600のコンポーネントは、以下により詳細に記載されるように、データソースのデータモデルによって変化し得るコン

50

ポーネントである。コホート識別システム600は、クエリービルダユーザインターフェイス(「UI」)605を含み得る。クエリービルダUI605は、コホート識別システム600のユーザに対して表示され得るユーザインターフェイスであり、クエリービルダUI605はユーザがクエリーを作成することを可能にし得る。他の実施例では、クエリービルダUI605はデータソースの1つ以上のデータフィールドを表示し得、ユーザは、1つ以上のデータフィールドのうちの少なくとも1つを、クエリーの一部であるように選択し得る。これらの実施例では、ユーザは、さらに、クエリーの一部であり得る基準を入力し得る。さらに詳細に記載されるように、ユーザによって作成されたクエリーは、薬剤安全性データなどのようなデータソース内において記憶されるデータを検索するために、薬剤安全性システムなどのようなデータソース上において実行され得る。ある実施例では、ユーザは、クエリービルダUI605内におけるクエリーのためにSQLシンタックスを入力し得る。さらに、ある実施例では、クエリービルダUI605は、クエリーの作者が、パラメータとして識別される、1つ以上のプレースホルダを指定することを可能にし得る。クエリーが実行されると、ユーザは、各パラメータ毎にパラメータ値を入力するよう促され得る。加えて、クエリービルダUI605は、さらに、ユーザがクエリーを実行することを可能にし得る。

10

【0053】

コホート識別システム600は、さらに、メタデータ610を含み得る。実施例によれば、メタデータ610は、薬剤安全性システム内における薬剤安全性データなどのような、データソース内におけるデータを記述する。より具体的には、メタデータ610は、照会され得るデータソースの各データフィールドについて情報を記述する。そのような情報は、データフィールドのデータタイプ、およびデータフィールドを含む構造化照会言語(「SQL」)クエリーを構築するよう必要とされる情報を含み得る。メタデータ610は、さらに、ソースデータまたはソースデータと参照データとの組合せから導き出され得る1つ以上のクエリーフィールドを含み得る。実施例に従うと、クエリービルダUI605は、ユーザがクエリービルダUI605を用いてメタデータ610に基づいてクエリーを作成し得るように、メタデータ610を検索し得る。メタデータ610のため、コホート識別システム600は、ある特定のデータソース、またはあるデータソースのある特定のデータモデルに作動的に結合されるようにのみ限定されない。代わりに、コホート識別システム600はデータモデル非依存型であり、より詳細に記載されるように、幅広いさまざまなデータソースと作動的に結合され得る。メタデータ610は、リポジトリなどのような、コホート識別システム600内に含まれる任意のデータ構造に記憶され得る。

20

30

【0054】

コホート識別システム600は、さらに、クエリーリポジトリ615を含み得る。クエリーリポジトリ615は、1つ以上のクエリーを記憶し得るリポジトリである。実施例によれば、クエリービルダUI605は、クエリービルダUI605内において作成されるクエリーをクエリーリポジトリ615内に記憶し得る。クエリービルダUI605内において作成されたクエリーは、クエリーがその後再使用されそうであり得ると判断されるとき、たとえばクエリーが幅広い範囲のシナリオにおいて用いられそうなデータを検索し得るときに、クエリーリポジトリ615内に記憶され得る。

40

【0055】

コホート識別システム600は、さらに、クエリーコンパイラ620を含み得る。クエリーコンパイラ620は、クエリーリポジトリ615内に記憶されるクエリーを検索し得、記憶されたクエリーをコンパイルし得る。記憶されたクエリーをコンパイルすることによって、クエリーコンパイラ620は、クエリーを実行可能なフォーマットに変換し得、記憶されたクエリーはコンパイルされ得る。いくつかの実施例では、クエリーコンパイラ620は、さらに、一旦クエリーが実行可能なフォーマットに変換されると、クエリーを実行し得る。クエリーを実行する際には、クエリーコンパイラ620は、クエリーをデータソース上で実行し得、クエリーに基づいてデータソースによって返されるデータを検索し記憶し得る。いくつかの実施例では、データソースによって返されるデータは薬剤安全

50

性データを含み、薬剤安全性データは1つ以上の有害事象ケースを含み、各有害事象ケースは有害事象を表現するデータレコードである。さらに、クエリーの実行はケースシリーズを作成し得る。

【0056】

コホート識別システム600は、さらに、ケースシリーズリポジトリ625を含み得る。ケースシリーズリポジトリ625は、1つ以上のケースシリーズなどのようなデータを記憶し得るリポジトリである。たとえば、ケースシリーズが、1つ以上の識別子を含み、さらに、ケースシリーズ内に含まれる1つ以上の有害事象ケースをともになって表現するケースデータおよび/またはケースメタデータを含む実施例では、ケースシリーズリポジトリ625は、1つ以上のケース識別子、ならびに関連付けられるケースデータおよび/またはケースメタデータを記憶し得る。実施例によれば、一旦クエリーコンパイラ620がクエリーを実行し、ケースシリーズを作成すると、クエリーコンパイラは、その作成されたケースシリーズをケースシリーズリポジトリ625内に記憶し得る。ある実施例では、ケースシリーズリポジトリ625は、ケースシリーズの正規表現を含み得る(図6には示されない)関連付けられるケースシリーズデータモデルを含み得る。たとえば、ケースシリーズが、1つ以上の識別子を含み、さらに、ケースシリーズ内に含まれる1つ以上の有害事象ケースをともになって表現するケースデータおよび/またはケースメタデータを含む実施例では、ケースシリーズデータモデルは、ケース識別子を表現するデータフィールド、ならびにケースデータおよび/またはケースメタデータを表現する1つ以上のさらに他のデータフィールドを含み得る。ある実施例では、ケースシリーズリポジトリ625は、図2のケースシリーズリポジトリ210および図4のケースシリーズリポジトリ400と同一である。

【0057】

コホート識別システム600は、さらに、報告ケースシリーズAPI630を含み得る。実施例によれば、報告ケースシリーズAPI630は、ケースシリーズリポジトリ625に関連付けられるケースシリーズデータモデルを露出させ、ケースシリーズリポジトリ625に関連付けられるケースシリーズデータモデルを、報告アプリケーション(たとえば報告アプリケーション640など)に対して、その報告アプリケーションにより指定されるフォーマットに基づいて表現し得るAPIである。したがって、報告ケースシリーズAPI630は、(報告アプリケーション640などのような)報告アプリケーションがケースシリーズリポジトリ625とインターフェイスすることを可能にし得る。換言すれば、報告ケースシリーズAPI630はケースシリーズをケースシリーズリポジトリ625から検索し、そのシリーズを報告アプリケーション(報告アプリケーション640など)内において実現し得る。ある実施例では、報告ケースシリーズAPI630は、図2のケースシリーズAPI230の一部を表現し、図4の報告アプリケーションケースシリーズAPI450と同一である。報告アプリケーション640は、1つ以上のケースシリーズを視覚化する1つ以上のレポートを生成するよう報告機能を実行し得る1つ以上の実行可能なプロセスを含むソフトウェアアプリケーションである。1つ以上のレポートを生成することは、1つ以上のレポートを報告アプリケーション640内において表示することを含み得る。報告アプリケーション640は、さらに、1つ以上の報告アルゴリズムを用いて、1つ以上のケースシリーズを形成し得る。報告アプリケーション640は、さらに、報告ケースシリーズAPI630を用いて、ケースシリーズリポジトリ625内に記憶される、コホート識別システム600によって形成される1つ以上のケースシリーズを消費し得る。ある実施例では、報告アプリケーション640は図4の報告アプリケーション440と同一である。

【0058】

コホート識別システム600は、さらに、相互運用可能なケースシリーズAPI635を含み得る。実施例によれば、相互運用可能なケースシリーズAPI635は、ケースシリーズリポジトリ625に関連付けられるケースシリーズデータモデルを露出し得、ケースシリーズリポジトリ625に関連付けられるケースシリーズデータモデルを、(パート

10

20

30

40

50

ナーアプリケーション 650 などのような) パートナーアプリケーションに対して、そのパートナーアプリケーションによって指定されるフォーマットに基づいて表現し得る API である。したがって、相互運用可能なケースシリーズ API 635 は、(パートナーアプリケーション 650 などのような) パートナーアプリケーションがケースシリーズリポジトリ 625 とインターフェイスすることを可能にし得る。換言すれば、相互運用可能なケースシリーズ API 635 は、ケースシリーズをケースシリーズリポジトリ 625 から検索し、そのシリーズをパートナーアプリケーション (パートナーアプリケーション 650 など) 内において実現し得る。パートナーアプリケーション 650 は、相互運用可能なケースシリーズ API 635 を用いて、ケースシリーズリポジトリ 625 内に記憶される、コホート識別システム 600 によって形成される 1 つ以上のケースシリーズを消費し得るソフトウェアアプリケーションである。パートナーアプリケーション 650 は、さらに、他の機能性、たとえば相互運用可能なケースシリーズ API 635 を用いてケースシリーズリポジトリ 625 内に記憶され得る 1 つ以上のケースシリーズを作成することなどを提供し得る。ある実施例では、相互運用可能なケースシリーズ API 635 は図 2 のケースシリーズ API 230 と同一である。

10

【0059】

コホート識別システム 600 は、さらに、コンパイラルール 645 を含み得る。実施例によれば、コンパイラルール 645 は 1 つ以上のシンタックスルールを含み得、1 つ以上のシンタックスルールは、クエリービルダ UI 605 によって作成されるクエリーが 1 つ以上のシンタックスルールでコンパイルすることを判断するために、クエリーコンパイラ 620 によってそのクエリーに適用され得る。コンパイラルール 645 は、リポジトリなどのような、コホート識別システム 600 内に含まれる任意のデータ構造に記憶され得る。

20

【0060】

コホート識別システム 600 は、さらに、オントロジブラウザ UI 655 を含み得る。データソースが、以下により詳細に記載される参照オントロジデータソースである実施例では、オントロジブラウザ UI 655 は、1 つ以上の参照オントロジを参照オントロジデータソースから検索し、その 1 つ以上の参照オントロジを、コホート識別システム 600 のユーザに対し、ある UI 内において表示し得る。したがって、オントロジからの 1 つ以上の要素が選択され、クエリーにおいて基準として用いられ得る。

30

【0061】

コホート識別システム 600 は、さらに、ケースシリーズエディタおよび管理 UI 665 を含み得る。ケースシリーズエディタおよび管理 UI 665 は、コホート識別システム 600 のユーザが、ケースシリーズリポジトリ 625 内に記憶される 1 つ以上のケースシリーズを編集および管理することを可能にし得る。

【0062】

コホート識別システム 600 は、さらに、ケースシリーズビューア UI 675 を含み得る。ケースシリーズビューア UI 675 は、コホート識別システム 600 のユーザが 1 つ以上のケースシリーズを閲覧することを可能にし得る。1 つ以上のケースシリーズはケースシリーズリポジトリ 625 内に記憶され得る。代替的に、1 つ以上のケースシリーズはデータソース内に記憶され得る。

40

【0063】

さらに、実施例によれば、コホート識別システム 600 は、1 つ以上のデータソースに作動的に結合され得る。先に記載されたように、コホート識別システム 600 のコンポーネント (つまりクエリービルダ UI 605 およびクエリーコンパイラ 620) は、ユーザが、1 つ以上のクエリーを作成し、コホート識別システム 600 に作動的に結合される 1 つ以上のデータソース上において実行することを可能にし得る。ある実施例では、1 つ以上のデータソースは薬剤安全性データを含み得、これらの実施例のうちのいくつかにおいては、薬剤安全性データは、1 つ以上の有害事象ケースなどのような、1 つ以上の薬剤の安全性に関係付けられるデータを含み得る。図 6 に示される実施例においては、1 つ以上

50

のデータソースは、参照オントロジデータソース 660、ならびに有害事象レポートデータベース 670、680 および 690 を含む。参照オントロジデータソース 660 は、参照オントロジに関するデータを含むデータソースである。参照オントロジデータソース 660 の例は、スノメド (「SNOMED」) データソース、国際医薬用語集 (「MedDRA」) データソースまたは世界保健機構 (「WHO」) 薬剤データソースを含む。有害事象レポートデータベース 678、680 および 690 は薬剤安全性データを含むデータソースであり、薬剤安全性データは 1 つ以上の有害事象ケースを含む。しかしながら、これらのデータソースは、示される実施例に従う単なる例示的なデータソースであり、代替的实施例では、コホート識別システム 600 は任意の数のデータソースに作動的に結合され得、各データソースはデータを含む任意のタイプのデータソースであり得る。

10

【0064】

コホート識別システム 600 は、さらに、統合されたクエリー実行エンジン 685 を含み得る。統合されたクエリー実行エンジン 685 は、記憶されたクエリーが複数のデータソースに抗してコンパイルされ実行されることを可能にし得る。統合されたクエリー実行エンジン 685 は、さらに、各データソースから返される 1 つ以上のケースシリーズを単一のケースシリーズにマージし得る。

【0065】

コホート識別システム 600 は、さらに、柔軟な再カテゴリ分類 API 695 を含み得る。柔軟な再カテゴリ分類 API 695 は、インターフェイスを、データソースにおいて用いられる 1 つ以上のコードリストに正規化し得る。大抵の保健関連データベースにおいては、離散値がコードとして記憶される。コードリストを用いて、1 つ以上の自然言語と等価である用語を表示し得る。この特徴は、ユーザが、クエリー基準を自身の言語で指定することを可能にし得る。さらに、「大陸」などのような 1 つ以上のロールアップ用語を用いて、国などのような離散値のグループに言及し得る。柔軟な再カテゴリ分類 API 695 は、さらに、年齢などのような 1 つ以上の範囲の連続する変数が、「成人」または「子供」などのような 1 つ以上の離散的な名称を付されたカテゴリにマッピングされることを可能にし得る。柔軟な再カテゴリ分類 API 695 は、同じコードマッピングを報告の際に用いて、クエリーとレポートとの間における一貫性を確実にすることを可能にし得る。

20

【0066】

図 7 は、この発明のある実施例に従って、コホート識別システムの例示的実現例のブロック図を示す。710 において、クエリーが作成される。クエリーは、薬剤安全性データなどのようなデータソース内に記憶されるデータを検索するために、薬剤安全性システムなどのようなデータソースにおいて実行され得る。実施例によれば、クエリーを作成するために、メタデータが検索され得、そのメタデータはデータソース内のデータを記述する。より具体的には、メタデータは、照会され得るデータソースの各データフィールドについての情報を記述し得る。そのような情報は、データフィールドのデータタイプ、およびデータフィールドを含む SQL クエリーを構築するよう必要とされる情報を含み得る。作成されるクエリーは、さらに、クエリーリポジトリ 720 にて記憶され、クエリーリポジトリ 720 は、1 つ以上のクエリーなどのようなデータを記憶し得るリポジトリである。

30

40

【0067】

730 において、クエリーがクエリーリポジトリ 720 から検索され、そのクエリーは、コンパイルされ、データソースの一例である有害事象レポートデータベース 740 上において実行される。有害事象レポートデータベース 740 は薬剤安全性データを含むデータソースであり、薬剤安全性データは 1 つ以上の有害事象ケースを含む。クエリーを実行する際には、薬剤安全性データなどのようなデータを有害事象レポートデータベース 740 から検索し得る。さらに、クエリーを実行する際には、ケースシリーズを作成し、ケースシリーズリポジトリ 750 内に記憶し得る。

【0068】

50

760において、ケースシリーズはケースシリーズリポジトリ750から検索され、ケースシリーズを視覚化し得るレポート770が生成される。ある実施例によれば、報告ケースシリーズAPIがケースシリーズリポジトリ750とインターフェイスし、ケースシリーズをケースシリーズリポジトリ750から検索し、ケースシリーズを報告アプリケーション内において実現して、ケースシリーズを視覚化し得る1つ以上のレポートを生成し得、報告アプリケーションは生成された1つ以上のレポートを表示し得る。ある実施例では、760において実行されるレポートの生成は、有害事象レポートデータベース740からデータを検索することを含み得る。

【0069】

図8は、この発明のある実施例に従って、コホート識別モジュールの機能のフロー図を示す。このフローは開始後、810に進む。フローはユーザがクエリーを作成したい旨を示したときに開始し得る。810において、メタデータが検索され、このメタデータは、データソースの1つ以上のデータフィールドについての情報を含む。実施例によれば、情報は、1つ以上のデータフィールドの各データフィールド毎に、データタイプおよびSQL情報を含み得る。ある実施例では、データソースは、1つ以上の有害事象ケースを記憶する有害事象レポートデータベースであり得る。フローは820に進む。

【0070】

820において、検索されたメタデータに基づいて、クエリーがデータソースに対して作成される。クエリーは、データソース内において記憶されるデータを検索するために、データソース上において実行されるクエリーであり得る。ある実施例では、検索されたメタデータを用いて、クエリーの一部である、データソースの1つ以上のデータフィールドを判断し得る。さらに、ある実施例においては、検索されたメタデータを用いて、クエリーの一部であるSQLを判断し得る。フローは830に進む。

【0071】

830では、クエリーは、1つ以上のコンパイルルールに基づいてコンパイルされる。実施例によれば、コンパイルルールは、クエリーが1つ以上のシンタックスルールでコンパイルすることを判断するようクエリーに適用される1つ以上のシンタックスルールを含み得る。ある実施例では、クエリーはクエリーリポジトリに記憶され得る。フローは840に進む。

【0072】

840では、クエリーがデータソース上において実行され、クエリーの実行はケースシリーズを作成する。ある実施例では、ケースシリーズは1つ以上の有害事象ケースを含み、各有害事象ケースは、有害事象を表現するデータレコードを含む。これらの実施例のいくつかにおいては、有害事象を表現する各データレコードは、さらに、薬剤安全性データを含み、薬剤安全性データは、1つ以上の薬剤の安全性に関係付けられる1つ以上のレポートまたは患者識別子を含む。ある実施例では、ケースシリーズはケースシリーズリポジトリに記憶される。フローは850に進む。

【0073】

850では、レポートがケースシリーズに基づいて生成され、そのレポートはケースシリーズの視覚化されたものである。ある実施例では、レポートは、ケースシリーズの1つ以上のデータフィールドの視覚的表示を含む。ある実施例によれば、ケースシリーズは、報告ケースシリーズAPIを用いてケースシリーズリポジトリから検索され得、報告ケースシリーズAPIは、報告アプリケーションに対するケースシリーズのフォーマットを規定する。これらの実施例では、レポートは報告アプリケーション内において表示され得る。さらに、ある実施例では、ケースシリーズは、相互運用可能なケースシリーズAPIを用いてケースシリーズリポジトリから検索され得、相互運用可能なケースシリーズAPIはパートナーアプリケーションに対するケースシリーズのフォーマットを規定する。これらの実施例では、ケースシリーズはパートナーアプリケーション内において消費され得る。フローはここで終了する。

【0074】

図 9 は、この発明のある実施例に従って、例示的なレポート 930 を作成する例示的ケースシリーズ 920 を作成する例示的クエリー 910 を示す。ある実施例によれば、ある実行可能なプロセスによって、薬剤安全性システムなどのようなデータソース上においてクエリー 910 が実行され得、クエリー 910 は、すべての致命的な有害事象ケース（つまりデータフィールド「死亡」が 1 の値を有するすべての有害事象ケース）を検索するクエリーである。

【0075】

クエリー 910 の実行は、この実施例に従って、ケースシリーズ 920 を形成し、ケースシリーズ 920 は、クエリー 910 において指定される条件と一致する有害事象ケースのリストを含む。ケースシリーズ 920 は、各有害事象ケース毎に少なくともケース識別子を含み得、さらに、ケースシリーズにおいて有害事象ケースを表現するさらに他のケースデータまたはメタデータを含んでもよい。示される実施例では、ケースシリーズ 920 は複数の有害事象ケースを含み、各有害事象ケースは：（a）ケース識別子データフィールドを含み、各値は、有害事象ケースに対するケース識別子を識別し；さらに、（b）国データフィールドを含み、各値は、有害事象ケースが関連付けられる国を識別する。

【0076】

ある実施例によれば、実行可能なプロセスによって、レポート 930 が、ケースシリーズ 920 に基づいて生成され得る。レポート 930 はケースシリーズ 920 を視覚化したものであり、ケースシリーズ 920 のデータフィールドは、所望のフォーマットによって変更され得る。示される実施例では、レポート 930 は複数の有害事象ケースを含み、各有害事象ケースは：（a）ケース識別子データフィールドを含み、各値は、有害事象ケースに対するケース識別子を識別し；さらに、（b）「深刻な」データフィールドを含み、各値は、有害事象ケースが深刻な有害事象ケースであるかどうかを識別し；さらに、（c）「リスト化された」データフィールドを含み、各値は、有害事象ケースがリスト化された有害事象ケースであるかどうかを識別する。当業者であれば、クエリー 910、ケースシリーズ 920 およびレポート 930 のフォーマットはある例示的实施例に従う例示的なフォーマットであり、クエリー、ケースシリーズおよび/またはレポートは代替的实施例においては他のフォーマットを有し得ることを容易に理解するであろう。

【0077】

したがって、一実施例においては、患者のコホートなどのようなコホートを識別し得るコホート識別システムが提供される。コホート識別システムはメタデータを検索し得、そのメタデータは、データソースおよびそのデータソースに対する SQL の両方を記述し、1 つ以上のクエリーを、そのデータソース上において、その検索されたメタデータに基づいて作成および実行し得る。メタデータはデータソースを正確に記述するため、コホート識別システムは、データソースのデータモデルから独立し得、幅広いさまざまなデータソースに作動的に結合され得、したがって、多数の異なる領域にわたって使用され得る。データソースは、一般的に、その基底にあるデータモデルにおいて異なるに過ぎず、コホート識別システムは、データソースの基底にあるデータモデルに対する依存性を含むことを必要とはされないもので、コホート識別システムは、薬剤安全性の領域に限定されない領域、より大きなコホート識別の領域に参入することができる。したがって、ある実施例によれば、コホート識別を任意の 1 つのパートナーアプリケーションに提供する費用を低減し得る。さらに、ある実施例によれば、より完全なコホート識別の特徴的機能の組をパートナーアプリケーションに提供し得る。

【0078】

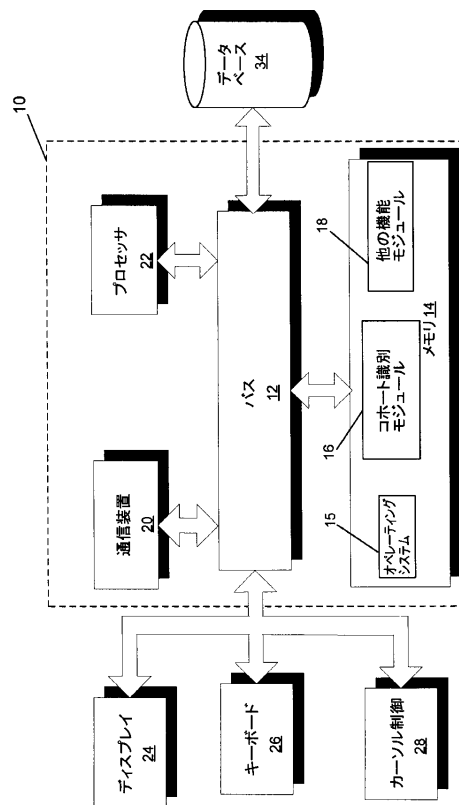
本明細書にわたって記載されるこの発明の特徴、構造または特性は、1 つ以上の実施例において任意の好適な態様で組合されてもよい。たとえば、本明細書にわたる「一実施例」、「いくつかの実施例」、「ある実施例」、または他の同様の文言の使用は、その実施例と関連して記載されるある特定の特徴、構造または特性がこの発明の少なくとも 1 つの実施例に含まれてもよいという事実を指す。したがって、本明細書にわたる「一実施例」、「いくつかの実施例」、「ある実施例」という表現または他の同様の文言の出現は、必

ずしもすべてが同じグループの実施例を指すとは限らず、記載された特徴、構造または特性は、1つ以上の実施例において任意の好適な態様で組合されてもよい。

【0079】

当業者であれば、上に論じられるこの発明は、異なる順序におけるステップで、および／または開示されるものとは異なる構成の要素で実施されてもよいことを容易に理解するであろう。したがって、この発明は、これらの好ましい実施例に基づいて記載されたが、当業者には、この発明の精神および範囲内にありながら、ある修正、変形および代替的構築が明らかであることは明らかであろう。この発明の境界／限界を判断するためには、したがって、特許請求の範囲が参照されるべきである。

【図1】



【図2】

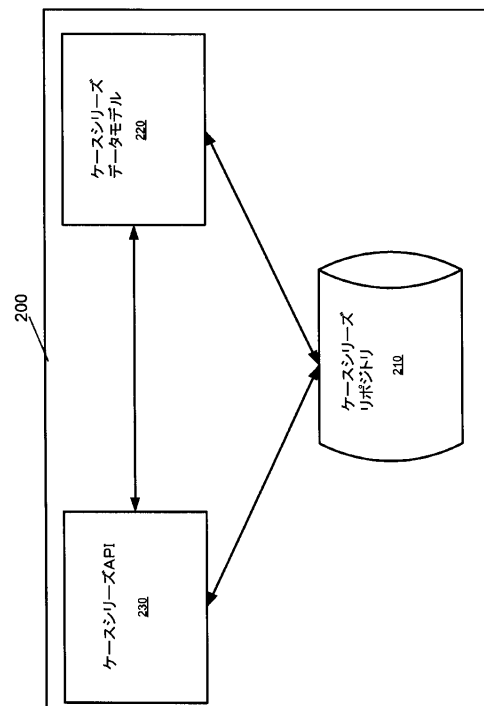


Fig. 1

Fig. 2

【図 3】

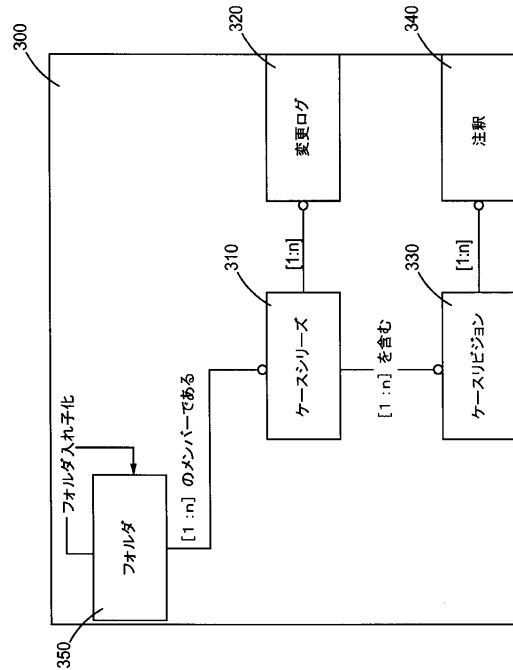


Fig. 3

【図 4】

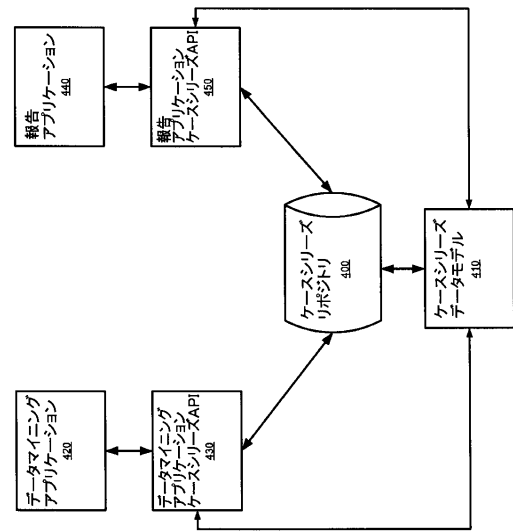


Fig. 4

【図 5】

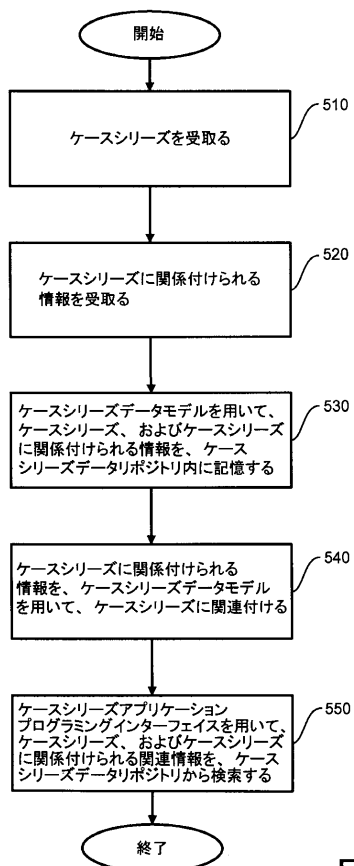


Fig. 5

【図 6】

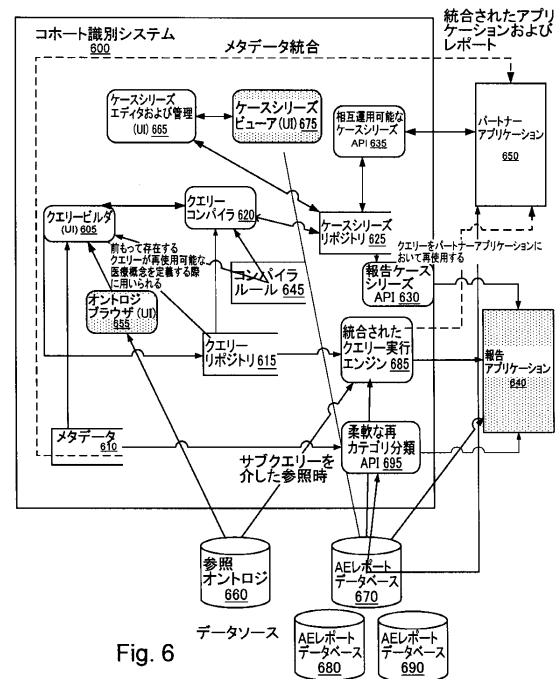
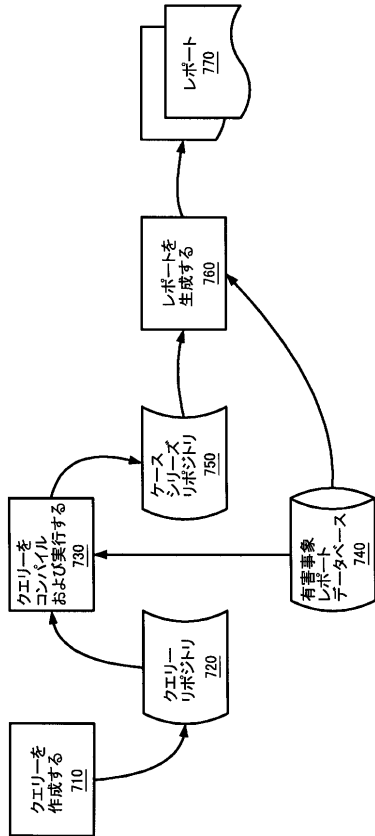


Fig. 6

【図 7】



【図 8】

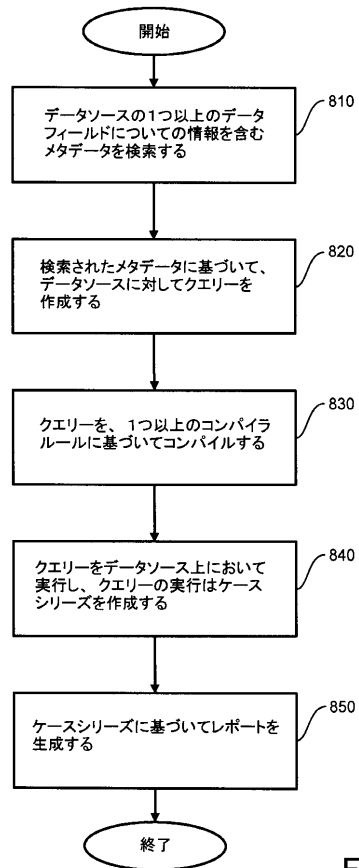


Fig. 7

Fig. 8

【図 9】

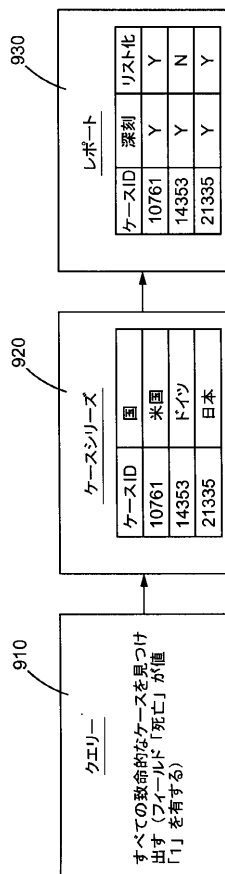


Fig.9

フロントページの続き

- (56)参考文献 特表2011-501845(JP,A)
特開2012-008613(JP,A)
特表2003-515823(JP,A)
米国特許出願公開第2009/0076845(US,A1)
平山 理 , 内部動作を理解しよう 絵で見て学ぶSQL Serverの深層, DB Magazine 第17巻 第11号, 日本, 株式会社翔泳社, 2008年 3月 1日, 第17巻, p.164-172
- (58)調査した分野(Int.Cl., DB名)
G16H 10/00-80/00
G06F 16/00