

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6416770号
(P6416770)

(45) 発行日 平成30年10月31日(2018.10.31)

(24) 登録日 平成30年10月12日(2018.10.12)

(51) Int.Cl.

F 1

G 16 H 10/00 (2018.01)
G 06 F 17/30 (2006.01)G 06 Q 50/24
G 06 F 17/30 230 Z

請求項の数 14 (全 25 頁)

(21) 出願番号	特願2015-539585 (P2015-539585)
(86) (22) 出願日	平成25年8月14日 (2013.8.14)
(65) 公表番号	特表2015-536499 (P2015-536499A)
(43) 公表日	平成27年12月21日 (2015.12.21)
(86) 國際出願番号	PCT/US2013/054836
(87) 國際公開番号	W02014/070278
(87) 國際公開日	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	13/790,379
(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・コーポレーション
(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 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
(72) 発明者	ブラウンボーグス, マイケル・ローレンス ドイツ、80939 ミュンヘン、トゥーヤベーク、13

前置審査

最終頁に続く

(54) 【発明の名称】相互運用可能なケースシリーズシステム

(57) 【特許請求の範囲】

【請求項 1】

ケースシリーズリポジトリを複数のソフトウェアアプリケーションと統合するためにコンピュータによって実行される方法であって、

第1のソフトウェアアプリケーションにより形成されたケースシリーズを受取ることを含み、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み、前記方法はさらに、

ケースリビジョンを受取ることを含み、前記ケースリビジョンはケースリビジョン情報を含み、前記ケースリビジョン情報は、前記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、前記方法はさらに、

ケースシリーズデータモデルを用いて、前記ケースシリーズおよび前記ケースリビジョンをケースシリーズリポジトリ内に記憶することを含み、前記ケースシリーズデータモデルは、前記ケースシリーズリポジトリ内における前記ケースシリーズおよび前記ケースリビジョンの第1のフォーマットを規定し、前記方法はさらに、

前記ケースシリーズデータモデルを用いて、前記ケースリビジョンを前記ケースシリーズと関連付けること、

第2の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および前記関連付けられるケースリビジョンを検索することを含み、前記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、前記ケースシリーズデータモデルを公開して、前記第2のソフ

10

20

トウェアアプリケーションによって指定される第2のフォーマットに基づいて前記ケースシリーズデータモデルを前記第2のソフトウェアアプリケーションに対して表現することにより、前記第2のソフトウェアアプリケーションが前記ケースシリーズリポジトリとインターフェイスすることを可能にする、コンピュータで実行される方法。

【請求項2】

前記ケースシリーズデータモデルは、前記ケースシリーズの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、前記ケースシリーズの前記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み、各ケース識別子は、前記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、請求項1に記載のコンピュータで実行される方法。 10

【請求項3】

前記ケースシリーズデータモデルは、前記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、請求項2に記載のコンピュータで実行される方法。

【請求項4】

前記ケースリビジョン情報は、前記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、請求項1～3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5】

前記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、

前記ケースシリーズデータモデルは、前記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、請求項4に記載のコンピュータで実行される方法。 20

【請求項6】

変更ログを作成することをさらに含み、前記変更ログはケースシリーズ履歴情報を含み、前記ケースシリーズ履歴情報は、前記ケースシリーズの履歴に関する情報を含み、前記方法はさらに、

前記ケースシリーズデータモデルを用いて前記変更ログを前記ケースシリーズリポジトリ内に記憶することを含み、前記ケースシリーズデータモデルは、前記ケースシリーズリポジトリ内における前記変更ログのフォーマットを規定し、前記方法はさらに、

前記ケースシリーズデータモデルを用いて前記変更ログを前記ケースシリーズに関連付けることを含む、請求項1～5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7】

注釈を作成することをさらに含み、前記注釈はユーザ定義情報を含み、前記方法はさらに、

前記ケースシリーズデータモデルを用いて前記注釈を前記ケースシリーズリポジトリ内に記憶することと、

前記ケースシリーズデータモデルを用いて前記注釈を前記ケースリビジョンに関連付けることを含む、請求項1～6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8】

フォルダを作成することをさらに含み、前記フォルダは、前記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み、前記方法はさらに、 40

前記ケースシリーズデータモデルを用いて前記フォルダを前記ケースシリーズリポジトリ内に記憶することと、

前記ケースシリーズデータモデルを用いて前記ケースシリーズを前記フォルダと関連付けることを含む、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のうちの1つを含む、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。

【請求項10】

前記ケースシリーズは、事象シリーズまたは製品シリーズの1つを含む、請求項1～9 50

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 1 1】

前記ケースシリーズは、1つ以上の薬剤の安全性に関する1つ以上のレポートまたは患者識別子を含む、請求項1～10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 1 2】

前記ケースシリーズは複数のケースリビジョンを含む、請求項1～11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 1 3】

請求項1～12のいずれかに記載の方法をコンピュータに実行させるためのプログラム
10
。

【請求項 1 4】

請求項13に記載のプログラムが格納されたメモリと、

前記メモリに接続され、前記プログラムを実行する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61/720,611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。

20

【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。

30

【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するデータを含む。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

40

50

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。

【0005】

多数の薬剤安全性システムは、同じ薬剤安全性データ上で動作し得る複数のソフトウェアアプリケーションまたはシステムを有する。さらに、異なる目的に対して編成された同じケースからなる複数のデータベースを有することは一般的であり得る。たとえば、米国食品医薬品庁（「U S F D A」または「F D A」）は、少なくとも4つの薬剤安全性データベースのネットワークを有し、そのデータにアクセスし得る少なくとも5つの主な薬剤安全性ソフトウェアアプリケーション／システムを有する。

【発明の概要】

10

【課題を解決するための手段】

【0006】

概要

一実施例は、ケースシリーズリポジトリを1つ以上のソフトウェアアプリケーションと統合する相互運用可能なケースシリーズシステムである。相互運用可能なケースシリーズシステムは、ケースシリーズを取り、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。相互運用可能なケースシリーズシステムはさらに、ケースリビジョンを取り、ケースリビジョン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つの変更を含む。相互運用可能なケースシリーズシステムはさらに、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ケースリビジョンをケースシリーズリポジトリ内に記憶し、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ケースリビジョンのフォーマットを規定する。相互運用可能なケースシリーズシステムはさらに、ケースシリーズデータモデルを用いて、ケースリビジョンをケースシリーズと関連付ける。相互運用可能なケースシリーズシステムはさらに、ケースシリーズプログラミングインターフェイスを用いて、ケースシリーズおよび関連付けられるケースリビジョンを検索し、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、ソフトウェアアプリケーションに対して、ケースシリーズおよび関連付けられるケースシリーズリビジョンのフォーマットを規定する。

20

【0007】

30

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】

【0008】

【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。

【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。

40

【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。

【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】

50

【0009】**詳細な説明**

一実施例においては、ケースシリーズリポジトリを、ケースシリーズを形成および消費する1つ以上のソフトウェアアプリケーションと統合し得る、相互運用可能なケースシリーズシステムが提供される。本明細書にて記載されるとおりでは、「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。ソフトウェアアプリケーションは、相互運用可能なケースシリーズシステムの一部であってもよく、相互運用可能なケースシリーズシステムを含む、より大きなシステムの一部であってもよく、別のシステムの一部であってもよく、または全くどのようなシステムの一部でなくてもよい。「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。「有害事象ケース」または「ケース」は、有害事象を表現するデータレコードを含む。「患者リスト」は1つ以上の医療レコードの集合である。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。相互運用可能なケースシリーズシステムは、1つ以上のケースシリーズを、それら1つ以上の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、に加えて記憶し得る中央ケースシリーズリポジトリを含み得る。相互運用可能なケースシリーズシステムは、さらに、1つ以上の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を用いて1つ以上のケースシリーズを形成および消費し得るように1つ以上のソフトウェアアプリケーションがインターフェイスし得るアプリケーションプログラミングインターフェイス(「API」)を含み得る。相互運用可能なケースシリーズは、さらに、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み得るケースシリーズデータモデルを含み得る。

【0010】

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に作動的に接合され、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(「RAM」)、リードオンリメモリ(「ROM」)、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。

【0011】

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、RAM、フラッシュメモリ、ROM、消去可能なPROM(「EPROM」)、電気的に消去可能なPROM(「EEPROM」)、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ドライブオンリメモリ(「CD-ROM」)、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。

【 0 0 1 2 】

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（「LCD」）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 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。

10

【 0 0 1 3 】

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、相互運用可能なケースシリーズモジュール 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。相互運用可能なケースシリーズモジュール 16 は、以下により詳細に記載されるように、ケースシリーズレポジトリを、ケースシリーズを形成および消費するアプリケーションと統合するための機能を提供し得る。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、ケースシリーズレポジトリを、ケースシリーズを形成および消費するアプリケーションと統合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。

20

【 0 0 1 4 】

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、オペレーションナルデータベース、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、リレーションナルデータベース、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。

30

【 0 0 1 5 】

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 API 230 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。

40

【 0 0 1 6 】

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

50

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関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。

【0017】

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。

10

【0018】

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。

20

【0019】

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。

【0020】

30

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るものとして識別される。換言すると、実施例によれば、ケースシリーズのすべてのバージョンは、ケースシリーズのリビジョンでもあるが、ケースシリーズのすべてのリビジョンがケースシリーズのバージョンでもあるというわけではない。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。

40

【0021】

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

50

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 A P I である。したがって、ケースシリーズ A P I 2 3 0 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 2 1 0 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 A P I 2 3 0 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 A P I 2 3 0 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 2 1 0 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 A P I 2 3 0 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 2 1 0 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 A P I 2 3 0 を用いてケースシリーズリポジトリ 2 1 0 とインターフェイスし得る。
10

【 0 0 2 2 】

ある実施例によれば、ケースシリーズリポジトリ 2 1 0 、ケースシリーズデータモデル 2 2 0 、およびケースシリーズ A P I 2 3 0 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 2 1 0 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 2 1 0 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 2 1 0 は、ケースシリーズ A P I 2 3 0 とは別の A P I 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 2 1 0 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。
20
30

【 0 0 2 3 】

加えて、一実施例においては、ケースシリーズリポジトリ 2 1 0 、ケースシリーズデータモデル 2 2 0 、およびケースシリーズ A P I 2 3 0 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 2 1 0 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
40
50

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、それらのケースに関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を含んでもよい。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、それらのケースに関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。

【0024】

ある実施例によれば、ケースシリーズAPI230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(1) 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。

【0025】

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、読み出アクセスおよび／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210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の後は、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。

【0026】

ケースシリーズAPI230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API230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210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API230は、あるシリーズをケースシリーズリポジトリ210から検索し、そのシリーズをソフトウェア

10

20

30

40

50

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 7 】

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データソース上においてクエリー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 2 1 0 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 2 1 0 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 2 1 0 にインポートし得る。

【 0 0 2 8 】

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 3 0 0 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 3 0 0 は、図 2 のケースシリーズデータモデル 2 2 0 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 3 0 0 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 3 0 0 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 2 1 0 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 A P I 2 3 0 のような)ケースシリーズ A P I 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。

【 0 0 2 9 】

実施例によれば、ケースシリーズデータモデル 3 0 0 はケースシリーズ 3 1 0 を含む。ケースシリーズ 3 1 0 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 3 1 0 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 3 1 0 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 3 1 0 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 3 1 0 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 3 1 0 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。

【 0 0 3 0 】

ケースシリーズデータモデル 3 0 0 は、さらに、変更ログ 3 2 0 を含む。変更ログ 3 2 0 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関、ケースシリーズを修正した 1 人以上の個人などのよう、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 3 2 0 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

10

20

30

40

50

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。

10

【0031】

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事象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。

20

30

40

【0032】

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

50

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。

【0033】

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、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。10

【0034】

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。2030

【0035】

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。40

【0036】

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50

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API430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API430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API430は、図2のケースシリーズAPI230のコンポーネントを表現する。

【0037】

この実現例は、さらに、報告アプリケーション440および報告アプリケーションケースシリーズAPI450を含む。実施例によれば、報告アプリケーション440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440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440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440はオラクル社(Oracle Corporation)からの"Oracle Argus Insight"製品である。

【0038】

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API450は、ケースシリーズデータモデル410を報告アプリケーション440に露出し、ケースシリーズデータモデル410を、報告アプリケーション440に対して、報告アプリケーション440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440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、報告アプリケーション440は、報告アプリケーションケースシリーズAPI450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400内に記憶し得る。同様に、報告アプリケーション440は、報告アプリケーションケースシリーズAPI450を用いて、ケースシリーズリポジトリ400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API450は、単に、報告アプリケーション440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440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400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API450は、図2のケースシリーズAPI230のコンポーネントを表現する。

【0039】

したがって、実施例によれば、報告アプリケーション440が、データマイニングアプリケーション420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420が、報告アプリケーション440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420は報告アプリケーション440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420および報告アプリケーション440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、代替的実施例においては、データマイニングアプリケーション420および報告アプリケーション440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410を用いてケースシリーズリポジトリ400にアクセスすることを可能にし得る

10

20

30

40

50

、任意の数のケースシリーズ A P I が存在し得る。

【 0 0 4 0 】

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アレイ（「P G A」）、フィールドプログラマブルゲートアレイ（「F P G A」）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 10 もよい。

【 0 0 4 1 】

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 20 に進む。

【 0 0 4 2 】

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に関係付けられる情報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。 30

【 0 0 4 3 】

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。 40

【 0 0 4 4 】

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 50

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。

【0045】

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。ケースシリーズに関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。

【0046】

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、ケースシリーズに関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。

【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、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。ケースシリーズに関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。

【0048】

540では、ケースシリーズに関係付けられる情報は、ケースシリーズデータモデルを用いて、ケースシリーズ内に関連付けられる。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、ケースシリーズと関連付けられる。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、ケースシリーズと関連付けられる。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、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関連付けられる。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関連付けられる。フローは550に進む。

【0049】

550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関連情報が、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関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 A P I 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関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。

【 0 0 5 0 】

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 6 0 0 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 6 0 0 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ポーネントである。コホート識別システム 6 0 0 は、クエリービルダユーザインターフェイス（「 U I 」） 6 0 5 を含み得る。クエリービルダ U I 6 0 5 は、コホート識別システム 6 0 0 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 U I 6 0 5 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 U I 6 0 5 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を選択し得る。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 U I 6 0 5 内におけるクエリーのために S Q L 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 U I 6 0 5 は、クエリーの作者が、パラメータとして識別される、1 つ以上のプレイ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 U I 6 0 5 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

【 0 0 5 1 】

コホート識別システム 6 0 0 は、さらに、メタデータ 6 1 0 を含み得る。実施例によれば、メタデータ 6 1 0 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 6 1 0 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 S Q L 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 6 1 0 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 U I 6 0 5 は、ユーザがクエリービルダ U I 6 0 5 を用いてメタデータ 6 1 0 に基づいてクエリーを作成し得るように、メタデータ 6 1 0 を検索し得る。メタデータ 6 1 0 のため、コホート識別システム 6 0 0 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 6 0 0 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 6 1 0 は、リポジトリなどのような、コホート識別システム 6 0 0 内に含まれる任意のデータ構造に記憶され得る。30

【 0 0 5 2 】

コホート識別システム 6 0 0 は、さらに、クエリーリポジトリ 6 1 5 を含み得る。クエリーリポジトリ 6 1 5 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 U I 6 0 5 は、クエリービルダ U I 6 0 5 内において作成されるクエリーをクエリーリポジトリ 6 1 5 内に記憶し得る。クエリービルダ U I 6 0 5 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 6 1 5 内に記憶され得る。40

【 0 0 5 3 】

コホート識別システム 6 0 0 は、さらに、クエリーコンパイラ 6 2 0 を含み得る。クエ50

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。

10

【0054】

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。

20

【0055】

コホート識別システム 600 は、さらに、報告ケースシリーズ API 630 を含み得る。実施例によれば、報告ケースシリーズ API 630 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 API 630 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 API 630 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 API 630 は、図 2 のケースシリーズ API 230 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 API 450 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 API 630 を用いて、ケースシリーズリポジトリ 625 内に記憶

30

40

50

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。

【 0056 】

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 API 635 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 API 635 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

【 0057 】

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

【 0058 】

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

【 0059 】

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。40

【 0060 】

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。

【 0061 】

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 U I 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、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、参考オントロジに関するデータを含むデータソースである。参考オントロジデータソース 660 の例は、スノメド（「S N O M E D」）データソース、国際医薬用語集（「M e d D R A」）データソースまたは世界保健機構（「W H O」）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、代替的実施例では、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

【 0 0 6 2 】

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。20

【 0 0 6 3 】

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 A P I 695 を含み得る。柔軟な再カテゴリ分類 A P I 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、国などの離散値のグループに言及し得る。柔軟な再カテゴリ分類 A P I 695 は、さらに、年齢などの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 A P I 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における一貫性を確実にすることを可能にし得る。30

【 0 0 6 4 】

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プロック図を示す。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 S Q L クエリーを構築する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データを記憶し得るリポジトリである。40

【 0 0 6 5 】

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。

【0066】

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。10

【0067】

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。20

【0068】

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。30

【0069】

830では、クエリーは、1つ以上のコンパイラルールに基づいてコンパイルされる。実施例によれば、コンパイラルールは、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。

【0070】

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。40

【0071】

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50

スシリーズ A P I 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 A P I 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 A P I 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。

【 0 0 7 2 】

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 9 3 0 を作成する例示的ケースシリーズ 9 2 0 を作成する例示的クエリー 9 1 0 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 9 1 0 が実行され得、クエリー 9 1 0 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。

【 0 0 7 3 】

クエリー 9 1 0 の実行は、この実施例に従って、ケースシリーズ 9 2 0 を形成し、ケースシリーズ 9 2 0 は、クエリー 9 1 0 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 9 2 0 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 9 2 0 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。

【 0 0 7 4 】

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 9 3 0 が、ケースシリーズ 9 2 0 に基づいて生成され得る。レポート 9 3 0 はケースシリーズ 9 2 0 を視覚化したものであり、ケースシリーズ 9 2 0 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 9 3 0 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 9 1 0 、ケースシリーズ 9 2 0 およびレポート 9 3 0 のフォーマットはある例示的実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実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。

【 0 0 7 5 】

したがって、一実施例においては、複数のソフトウェアアプリケーション間におけるケースシリーズの転送を容易にし得る、相互運用可能なケースシリーズシステムが提供される。実施例によれば、相互運用可能なケースシリーズシステムは、はるかに高い度合の相互アプリケーション自動化を可能にし得る。さらに、相互運用可能なケースシリーズは、データを用いて達する結論の科学的整合性および検証性を増大させ、個人が自分の作業を検査し、改善することをより容易にし得る。相互運用可能なケースシリーズシステムは、薬剤安全性領域における任意の製品と関係があり得、なぜならば、相互運用可能なケースシリーズシステムを用いて、ケースシリーズを共有することを求められる 2 つ以上のアプリケーションをよりよく統合し得る、または単一のアプリケーションのトランザクションおよび報告モジュールをよりよく統合し得るからである。

【 0 0 7 6 】

薬剤安全性を超えて、相互運用可能なケースシリーズシステムは、疫学および臨床への橋渡しとなる医療調査などの他の健康科学領域において用いられ得る。たとえば、

10

20

30

40

50

ある健康科学アプリケーションを用いて、患者のコホートを識別し、それを患者リストとしてセーブし得る。相互運用可能なケースシリーズシステムは、コホート識別 / クエリーアプリケーションとビジネスインテリジェンス可視化アプリケーションとの間においてインターフェイスとして用いられ得る。これらの2つのアプリケーションが患者リストを互いに転送する必要があるときはいつでも、相互運用可能なケースシリーズシステムはその患者リストの転送を容易にし得るであろう。さらに、多数の健康管理機関は、同じ患者に対するデータを含むアプリケーションおよびデータベースのネットワークを有する。ある相互運用可能なケースシリーズシステムは、ある実施例によれば、これらのアプリケーション間における患者リストの転送を容易にし得るであろう。

【0077】

10

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。

【0078】

20

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ろう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】

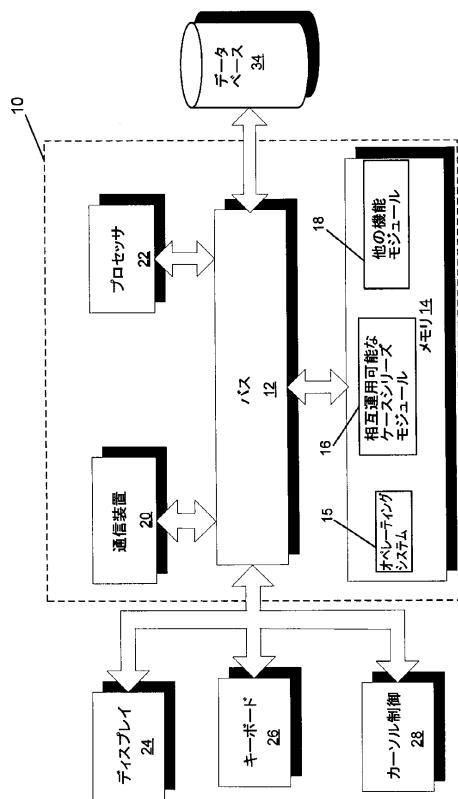


Fig. 1

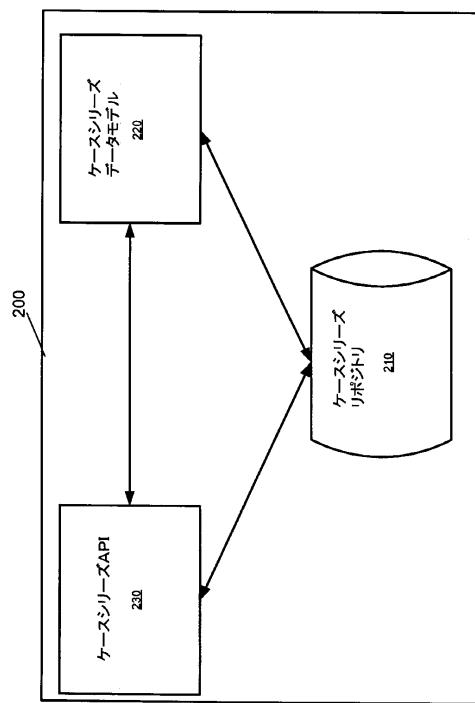


Fig. 2

【図3】

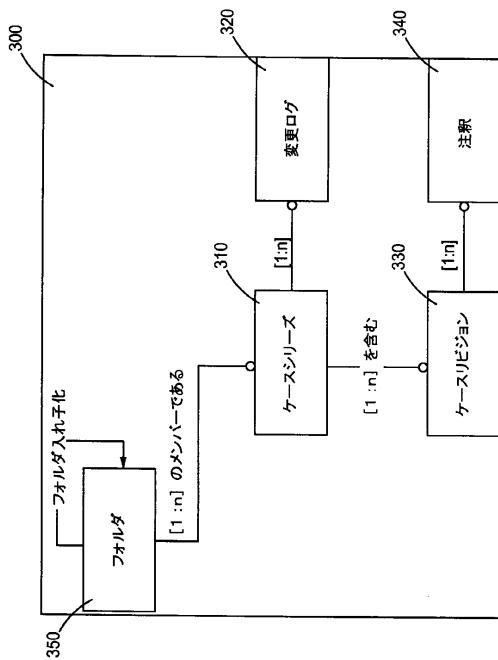


Fig.3

〔 四 4 〕

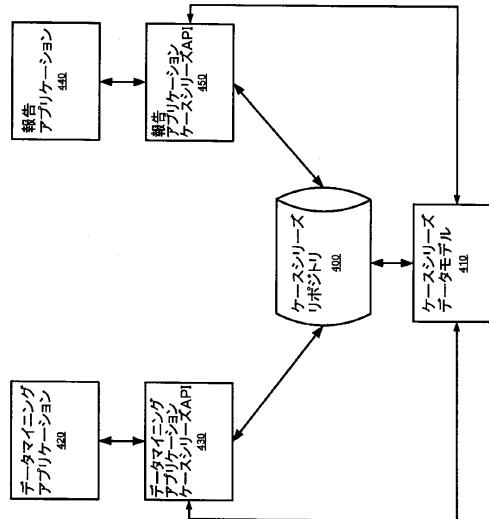


Fig. 4

【図5】

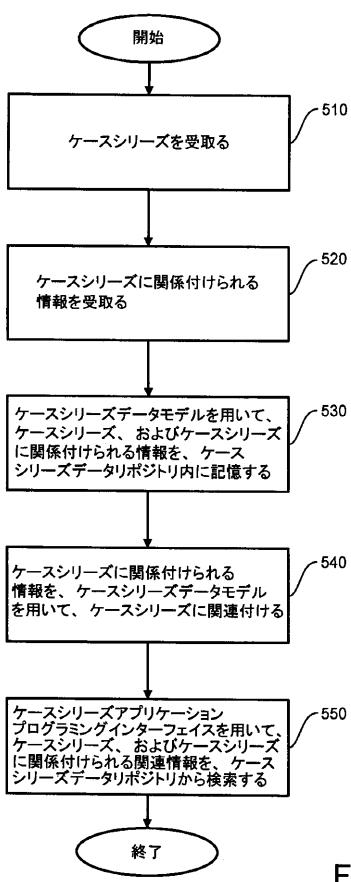


Fig. 5

【 义 6 】

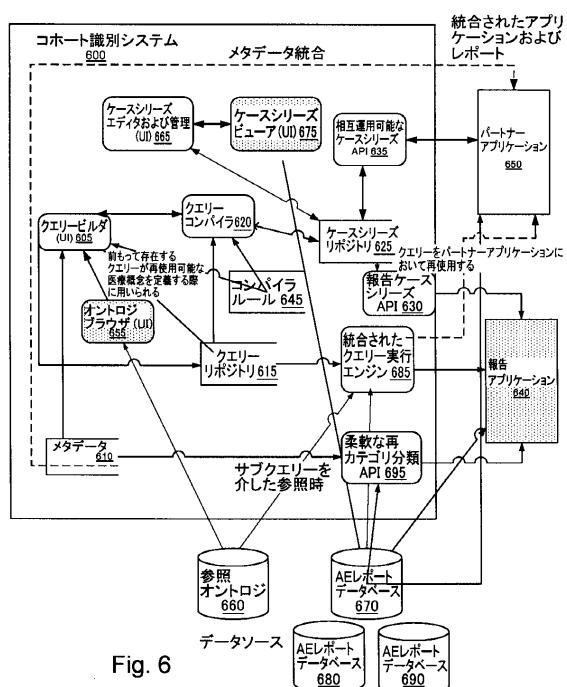


Fig. 6

【図7】

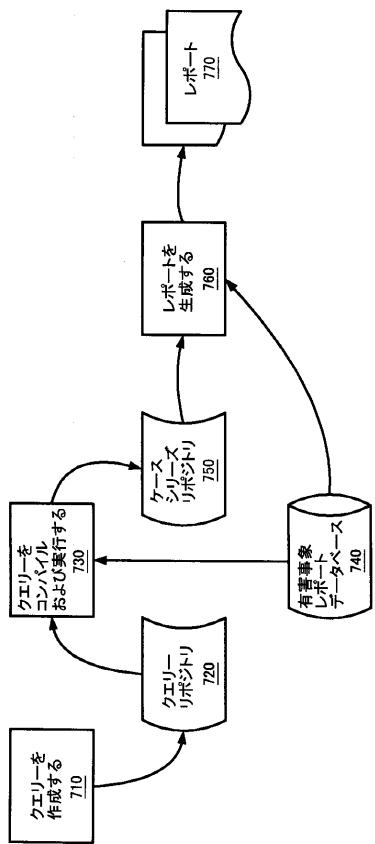


Fig. 7

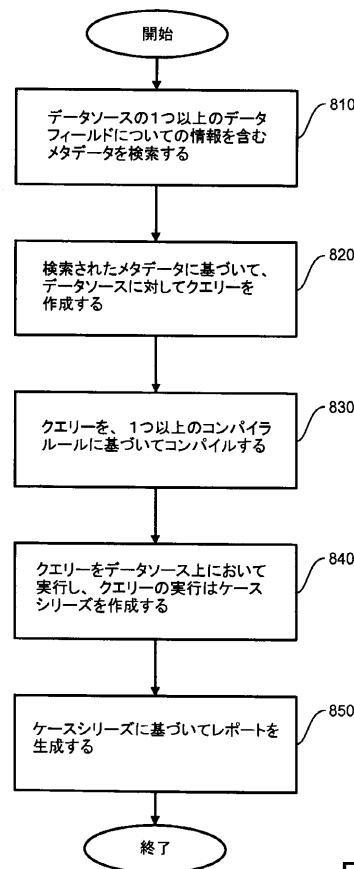


Fig. 8

【図9】

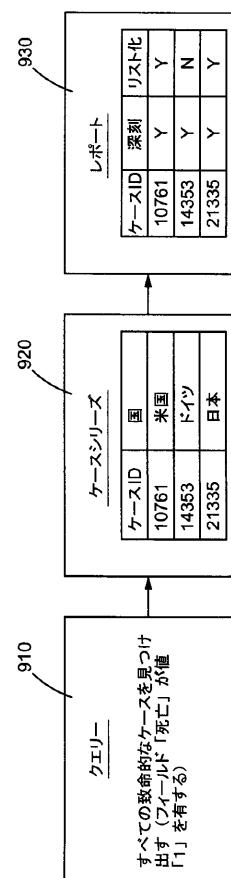


Fig. 9

フロントページの続き

審査官 松野 広一

(56)参考文献 特開2007-133704(JP,A)

特開平10-143539(JP,A)

特表2002-529970(JP,A)

特開2006-178951(JP,A)

特開2010-055634(JP,A)

特開2008-188329(JP,A)

特開2006-285973(JP,A)

米国特許出願公開第2010/0070304(US,A1)

米国特許出願公開第2009/0076845(US,A1)

米国特許出願公開第2009/0055421(US,A1)

米国特許出願公開第2008/0319958(US,A1)

(58)調査した分野(Int.Cl., DB名)

G06Q 10/00-99/00

G16H 10/00-80/00

G06F 17/30