

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6137505号
(P6137505)

(45) 発行日 平成29年5月31日(2017.5.31)

(24) 登録日 平成29年5月12日(2017.5.12)

(51) Int.Cl.

G06F 9/44 (2006.01)

F 1

G06F 9/06 620A

請求項の数 18 (全 18 頁)

(21) 出願番号 特願2014-546194 (P2014-546194)
 (86) (22) 出願日 平成24年12月12日 (2012.12.12)
 (65) 公表番号 特表2015-501993 (P2015-501993A)
 (43) 公表日 平成27年1月19日 (2015.1.19)
 (86) 國際出願番号 PCT/US2012/069318
 (87) 國際公開番号 WO2013/090476
 (87) 國際公開日 平成25年6月20日 (2013.6.20)
 審査請求日 平成27年11月24日 (2015.11.24)
 (31) 優先権主張番号 13/323,198
 (32) 優先日 平成23年12月12日 (2011.12.12)
 (33) 優先権主張国 米国(US)

(73) 特許権者 314015767
 マイクロソフト テクノロジー ライセンシング, エルエルシー
 アメリカ合衆国 ワシントン州 98052 レッドモンド ワン マイクロソフト ウェイ
 (74) 代理人 100107766
 弁理士 伊東 忠重
 (74) 代理人 100070150
 弁理士 伊東 忠彦
 (74) 代理人 100091214
 弁理士 大貫 進介

最終頁に続く

(54) 【発明の名称】ウェブアプリケーションのための軽量フレームワーク

(57) 【特許請求の範囲】

【請求項 1】

ウェブアプリケーションを開発するための方法であって、
スクリプト言語ベースのフレームワークのためのインターフェースを介して、前記スクリプト言語ベースのフレームワークに含まれるコントロールを管理する呼び出しを受けるステップであり、前記呼び出しは前記スクリプト言語ベースのフレームワークを参照するアプリケーションにより提供される、ステップと、

前記呼び出しにより管理された前記コントロールに基づいて、ユーザインターフェースを生成するステップと、

前記呼び出しにより管理された前記コントロールに基づいて、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスにアクセスするステップと、

前記呼び出しにより管理された前記コントロールに基づいて、前記スクリプト言語ベースのフレームワークに直接的に組み込まれた第1のサードパーティサービスを用いるステップと、

を含む方法。

【請求項 2】

前記呼び出しは、前記アプリケーションに含まれるスクリプト言語コードから生成される、請求項1に記載の方法。

【請求項 3】

10

20

前記コントロールを管理する前記呼び出しは、前記コントロールを有効にし、あるいは前記コントロールを無効にするのうち1つをなす、請求項1に記載の方法。

【請求項4】

前記コントロールは、前記クラウドプラットフォーム上の前記分散型データストアにより提供される前記バックエンドのクラウドベースのサービスにアクセスするように前記スクリプト言語ベースのフレームワーク内に予めプログラムされる、請求項1に記載の方法。

【請求項5】

前記呼び出しに基づいて、前記コントロールの属性又は前記コントロールの効果のうち少なくとも1つの値を設定するステップ、をさらに含む請求項1に記載の方法。

【請求項6】

前記呼び出しにより管理された前記コントロールに基づいて、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスにアクセスするステップは、

前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアにデータを書き込むステップ、

前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアからデータを読み出すステップ、

前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアの中のデータを検索するステップ、又は

前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアの中のデータをインデックス化するステップ

のうち少なくとも1つをさらに含む、請求項1に記載の方法。

【請求項7】

前記呼び出しにより管理された前記コントロールに基づいて、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスにアクセスするステップは、前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアに組み込まれた第2のサードパーティサービスにアクセスするステップをさらに含む、請求項1に記載の方法。

【請求項8】

前記クラウドプラットフォーム上の前記分散型データストアに組み込まれた前記第2のサードパーティサービスは、ウェブ検索サービス、画像検索サービス、ビデオ検索サービス、ソーシャルネットワーキングサービス、マイクロブログサービス、地図作成サービス、半構造化データベース検索サービス、認証サービス又はクラウドプラットフォームサービスのうち1又は複数を含む、請求項7に記載の方法。

【請求項9】

前記アプリケーションは、前記スクリプト言語ベースのフレームワークに含まれる複数のコントロールを管理する複数の呼び出しを提供するスクリプト言語コードを含む、請求項1に記載の方法。

【請求項10】

前記アプリケーションにより提供される前記複数の呼び出しにより管理された前記複数のコントロール間で情報を交換するステップ、をさらに含む請求項9に記載の方法。

【請求項11】

前記複数のコントロールのうち第1のコントロールを、第1のデータソースからのオブジェクトにバインドするステップと、

前記複数のコントロールのうち第2のコントロールを、第2のデータソースからのオブジェクトにバインドするステップと、

をさらに含む請求項9に記載の方法。

【請求項12】

前記スクリプト言語ベースのフレームワークは、クライアント側のフレームワークであ

10

20

30

40

50

る、請求項 1 に記載の方法。

【請求項 1 3】

プロセッサと、

前記プロセッサに結合されたメモリと、

を含む装置であって、

前記メモリは、スクリプト言語ベースのフレームワークを実施するように構成された命令を記憶し、前記スクリプト言語ベースのフレームワークは、

有効にされたときにバックエンドのクラウドベースのサービスにアクセスし、かつユーザインタフェースを生成するように予めプログラムされた一式のコントロールと、

前記一式のうちのコントロールを管理する呼び出しを受けるインタフェースコンポーネントであり、前記呼び出しは前記スクリプト言語ベースのフレームワークを参照するアプリケーションにより提供される、インタフェースコンポーネントと、

前記呼び出しにより管理された前記コントロールに基づいてユーザインタフェースを生成するユーザインタフェース（UI）作成コンポーネントと、

前記呼び出しにより管理された前記コントロールに基づいて、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスにアクセスし、前記呼び出しにより管理された前記コントロールに基づいて、前記スクリプト言語ベースのフレームワークに直接的に組み込まれた第 1 のサードパーティサービスを使用するデータアクセスコンポーネントと、

を含む、装置。

10

【請求項 1 4】

前記データアクセスコンポーネントは、前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアに組み込まれた第 2 のサードパーティサービスにアクセスする、請求項 1_3 に記載の装置。

【請求項 1 5】

前記一式の中の複数のコントロール間で情報を交換する通知コンポーネント、をさらに含む請求項 1_3 に記載の装置。

【請求項 1 6】

前記アプリケーションを実行しているクライアント装置の能力を検出し、かつ前記コントロールに基づいて生成される前記ユーザインタフェースを前記能力に応じて調整する調整コンポーネント、をさらに含む請求項 1_3 に記載の装置。

30

【請求項 1 7】

前記データアクセスコンポーネントは、前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアにデータを書き込み、前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアからデータを読み出し、前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアの中のデータを検索し、あるいは前記呼び出しにより管理された前記コントロールに基づいて、前記クラウドプラットフォーム上の前記分散型データストアの中のデータをインデックス化するのうち少なくとも 1 つをなす、請求項 1_3 に記載の装置。

40

【請求項 1 8】

有効にされたときにバックエンドのクラウドベースのサービスにアクセスし、かつユーザインタフェースを生成するように予めプログラムされた一式のコントロールを含むスクリプト言語ベースのフレームワークのためのインタフェースを介して、前記スクリプト言語ベースのフレームワークに含まれる前記一式のコントロールのうちのコントロールを有効にする呼び出しを受けるステップであり、前記呼び出しは前記スクリプト言語ベースのフレームワークを参照するアプリケーションにより提供される、ステップと、

前記呼び出しにより有効にされた前記コントロールに基づいて、ユーザインタフェースを生成するステップと、

前記呼び出しにより有効にされた前記コントロールに基づいて、クラウドプラットフォ

50

ーム上の分散型データストアにより提供される前記バックエンドのクラウドベースのサービスにアクセスするステップと、

前記呼び出しにより有効にされた前記コントロールに基づいて、前記スクリプト言語ベースのフレームワークに直接的に組み込まれたサードパーティサービスを用いるステップと、

をプロセッサに実行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】

本明細書において説明される種々の技術は、スクリプト言語ベースのフレームワークを用いてウェブアプリケーションを開発することに属する。 10

【背景技術】

【0002】

ウェブアプリケーションは、インターネットなどのネットワークを介してアクセスすることができるアプリケーションである。ウェブアプリケーションは、クライアント側のウェブブラウザ又はクライアント装置上の別のアプリケーション（例えば、モバイルアプリケーション、デスクトップアプリケーションなど）を通じてアクセス及び利用することができる、ブラウザによりサポートされる言語（例えば、マークアップ言語と組み合わせられたスクリプト言語）で、コード化することができる。ウェブアプリケーションは、ウェブブラウザが至る所に存在することとウェブブラウザをクライアントとして利用することの便利さとに起因して、ますます一般的になってきている。さらに、ウェブアプリケーションの需要は、集中型のアプリケーション市場の成長につながっている。 20

【発明の概要】

【発明が解決しようとする課題】

【0003】

近年、クラウドコンピューティングが、より普及してきている。それに応じて、ウェブアプリケーションは、クラウドプラットフォーム上に構築することができ、クラウドプラットフォームにおいて、ウェブアプリケーションは、クラウドの中で動作することができ、かつ／あるいはクラウドにより提供されるバックエンドサービスを使用することができる。クラウドプラットフォームに構築されるアプリケーションの従来型の開発は、一般に、種々のクラウドプラットフォーム固有情報（例えば、プロトコル、データ構造など）の理解を有するウェブプログラマを参加させて、クラウドからデータを読み出す、クラウドにデータを書き込む、他のバックエンドサービスを利用する、などを行う。クラウドプラットフォーム上にアプリケーションを開発することに関連するこうした複雑さに加えて、従来型の開発は、しばしば、時間がかかる場合がある。集中型のアプリケーション市場の成長とともに、ウェブプログラマは、モバイル装置及びウェブブラウザで動作する強力なアプリケーションをより少ない時間で書く方法をますます期待している。 30

【課題を解決するための手段】

【0004】

スクリプト言語ベースのフレームワークを用いてウェブアプリケーションを開発することに属する種々の技術が、本明細書において説明される。スクリプト言語ベースのフレームワークのインターフェースを、公開する（exposed）ことができる。さらに、呼び出しを、スクリプト言語ベースのフレームワークのインターフェースを介して受けることができる。上記呼び出しは、スクリプト言語ベースのフレームワークに含まれるコントロールを管理することができる。さらに、上記呼び出しは、スクリプト言語ベースのフレームワークを参照するアプリケーションにより提供されることができる。その上、ユーザインターフェースを、上記呼び出しにより管理されたコントロールに基づいて生成することができ、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスに、上記呼び出しにより管理されたコントロールに基づいてアクセスすることができる。 40

【0005】

スクリプト言語ベースのフレームワークは、一式のコントロールを含むことができ、上記一式のコントロールは、バックエンドのクラウドベースのサービスにアクセスし、かつ、有効にされたときにユーザインターフェースを生成するように、予めプログラムされる。一例に従い、スクリプト言語ベースのフレームワークに含まれるコントロールのうち1又は複数に対して有効にする、無効にする、属性の値を設定する、などするための呼び出しを提供するアプリケーションを、開発することができる。例えば、上記呼び出しは、アプリケーションに含まれるスクリプト言語コードから生成することができる。したがって、こうした機能性を実行するためにアプリケーションにカスタムコードを含むのではなく、スクリプト言語ベースのフレームワークに含まれる機能性を用いる（例えば、上記一式のコントロールを管理することによる）アプリケーションを、開発することができる。10

【0006】

種々の実施形態において、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスには、分散型データストアにデータを書き込むこと、分散型データストアからデータを読み出すこと、分散型データストアに保有されているデータを検索すること、分散型データストアの中のデータをインデックス化すること(indexing)などのうち1又は複数を含むことができる。さらに、又は別法として、1又は複数のサードパーティサービスに、クラウドプラットフォーム上の分散型データストアを介して、及び/又はスクリプト言語ベースのフレームワークから直接的に、アクセス可能とすることができる。20

【0007】

上記の要約は、本明細書において論じられるシステム及び/又は方法のいくつかの態様の基本的理解を与えるための簡素化された要約を提示する。この要約は、本明細書において論じられるシステム及び/又は方法の広範囲に及ぶ概観ではない。上記要約は、重要な/必須の要素を識別すること、あるいはこうしたシステム及び/又は方法の範囲を詳述することを、意図するものではない。上記要約の唯一の目的は、後から提示されるより詳細な説明に対する前置きとして簡素化された形式でいくつかの概念を提示することである。

【図面の簡単な説明】

【0008】

【図1】ウェブアプリケーションの開発を容易にする例示的なシステムの機能的プロック図を示す。30

【図2】サードパーティサービスを用いるウェブアプリケーションの開発を容易にする例示的なシステムの機能的プロック図を示す。

【図3】スクリプト言語ベースのフレームワークを用いてアプリケーションを開発するための例示的なシステムの機能的プロック図を示す。

【図4】ウェブアプリケーションを開発するための例示的な方法論を示すフロー図である。

【図5】ウェブアプリケーションを開発するための別の例示的な方法論を示すフロー図である。

【図6】例示的なコンピューティング装置を示す。

【発明を実施するための形態】

【0009】

次に、図面を参照して、軽量の、クライアント側のフレームワークを用いるウェブアプリケーションを開発することに属する種々の技術を説明する。図面において、全体にわたって、同様の参照番号を用いて同様の要素を参照する。下記記載において、説明の目的で、多くの特定の詳細が、1又は複数の態様への十分な理解を与えるために説明される。しかしながら、こうした態様(群)は上記の特定の詳細を用いずに実施され得ることが、明らかであろう。他の例において、周知の構造及び装置が、1又は複数の態様を説明することを容易にするために、プロック図形式で示される。さらに、ある特定のシステムコンポーネントにより実行されるものとして説明される機能性は、複数コンポーネントにより実50

行することができる事を、理解されたい。同様にして、例えば、あるコンポーネントは、複数コンポーネントにより実行されるものとして説明される機能性を実行するように構成することができる。

【0010】

さらに、「又は」という用語は、排他的な「又は」ではなくて包括的な「又は」を意味するものである。すなわち、他に指定のない、又は文脈から明らかでない限り、「XはA又はBを用いる」は、自然な包含的順列のいずれかを意味するものである。すなわち、「XはA又はBを用いる」という表現は、次の例、すなわち、XがAを用いる、XがBを用いる、あるいはXがA及びBの双方を用いるのうち、いずれの下でも満たされる。加えて、本願及び添付の請求項において使用される冠詞「1つの（“a”、“an”）」は、概して、他に指定のない限り又は単数形を対象とすることが文脈から明らかでない限り、「1又は複数の」を意味するものとして解釈されるべきである。

10

【0011】

本明細書において説明されるとおり、軽量フレームワークは、ウェブプログラマによる迅速なウェブアプリケーション開発を容易にすることができる。本明細書において説明される軽量フレームワークは、スクリプト言語ベースのフレームワークとすることができる。その上、軽量フレームワークは、装置互換のユーザインタフェース作成を可能にし、かつ、クラウドプラットフォーム上の分散型データストアに保有されているデータ、及び／又はクラウドプラットフォーム上の分散型データストアにより提供される若しくは上記分散型データストアを介して利用可能なバックエンドのクラウドベースのサービスに、アクセスすることができる。対称的に、ウェブアプリケーション開発のために使用される従来型のフレームワークは、しばしば、データアクセスをサポートすることなくユーザインタフェース作成をサポートする。それに応じて、従来型のウェブアプリケーション開発において、ウェブプログラマは、しばしば、クラウドからデータを読み出し、クラウドにデータを書き込み、又は他のバックエンドのクラウドベースのサービスを用いるウェブアプリケーションを開発するために、クラウド技術、データ構造、プロトコルなどの知識を有することを必要とする。

20

【0012】

次に、図面を参照し、図1が、ウェブアプリケーションを開発することを容易にするシステム100を示す。システム100は、スクリプト言語ベースのフレームワーク102を含み、スクリプト言語ベースのフレームワーク102は、例示的な実施形態において、クライアント側のフレームワークである。例えば、スクリプト言語ベースのフレームワーク102は、機能豊富な、軽量フレームワークとすることができる。その上、スクリプト言語ベースのフレームワーク102は、アプリケーション開発中に行われる一般的なアクティビティに関連するオーバーヘッドを軽減することによって、ウェブアプリケーションの開発をサポートすることができる。したがって、上記のアクティビティに関連する機能性はスクリプト言語ベースのフレームワーク102に包含することができるため、ウェブプログラマは、アプリケーション開発中に上記のような共通のアクティビティを行う必要がなく、代わって、ウェブプログラマは、スクリプト言語ベースのフレームワーク102により供給される機能性を使用するアプリケーションを作成することができる。

30

【0013】

ウェブアプリケーションを開発することには、通常は、一式のユーザインタフェースを構築すること、及びバックエンドシステムへのデータアクセスを構築すること（及び／又は、バックエンドシステムを構築すること）を含む。ユーザインタフェースは、種々の形式の入力をユーザから収集することができ、バックエンドシステムの仕様に合うように上記入力を処理することができる。その上、ユーザインタフェースは、ユーザに出力を提示することができる。スクリプト言語ベースのフレームワーク102は、ウェブプログラマが基本的なスクリプト言語コード及び／又はマークアップ言語コードを用いてバックエンドシステムへのデータアクセスとユーザインタフェースの作成との双方を管理することを可能にすることによって、ウェブアプリケーションの開発を簡素化する。例えば、スクリ

40

50

プト言語は、JavaScrip^t（登録商標）とすることができます。マークアップ言語の一例を、HTMLとすることができます。しかしながら、特許請求される対象事項は前述の例に限定されないことを、十分理解されたい。

【0014】

スクリプト言語ベースのフレームワーク102は、データアクセスコンポーネント104とユーザインタフェース（UI）作成コンポーネント106とを含む。データアクセスコンポーネント104は、クラウドベースのバックエンドシステムを組み込む。具体的には、データアクセスコンポーネント104は、クラウドプラットフォーム110上の分散型データストア108により提供されるバックエンドのクラウドベースのサービスにアクセスすることができる。例えば、データアクセスコンポーネント104は、分散型データストア108からデータを読み出すこと、分散型データストア108にデータを書き込むこと、分散型データストア108により抽象化されたサードパーティのバックエンドのクラウドベースのサービス（群）を用いることなどができる。データアクセスコンポーネント104は、分散型データストア108を自動的に使用するように設計することができる。したがって、ウェブプログラマにより開発されるウェブアプリケーションは、ウェブプログラマが分散型データストア108により提供されるバックエンドのクラウドベースのサービスへのアクセスを組み込む必要なく、分散型データストア108を自動的に使用することができる。

【0015】

その上、UI作成コンポーネント106は、一式のユーザインタフェースを生成することができる。UI作成コンポーネント106は、種々の形式の入力をユーザから収集することができ、かつバックエンドシステムの仕様に合うように上記入力を処理することができる、ユーザインタフェースを生成することができる。さらに、UI作成コンポーネント106は、ユーザに出力を提示することができるユーザインタフェースを生成することができる。

【0016】

さらに、スクリプト言語ベースのフレームワーク102は、一式のコントロール112を含む。一例に従い、スクリプト言語ベースのフレームワーク102は、コントロール112を含む1又は複数のスクリプト言語ライブラリを含むことができる。コントロール112は、ユーザインタフェースを作成するためのUI作成コンポーネント106と、クラウドプラットフォーム110上の分散型データストア108により提供されるバックエンドのクラウドベースのサービスにアクセスするためのデータアクセスコンポーネント104とによって、活用することができる。1つの例解に従い、コントロール112の中から所与のコントロールが有効にされたとき、UI作成コンポーネント106は上記所与のコントロールに基づいてユーザインタフェースを生成することができ、データアクセスコンポーネントは上記所与のコントロールに基づいてクラウドプラットフォーム110上の分散型データストア108により提供されるバックエンドのクラウドベースのサービスにアクセスすることができる。したがって、ウェブプログラマが所与のコントロールの機能性を提供するためにカスタムコードを書くこととは対照的に、上記所与のコントロールをウェブプログラマが活用することができる。

【0017】

スクリプト言語ベースのフレームワーク102は、さらに、インターフェースコンポーネント114を含む。インターフェースコンポーネント114は、アプリケーション116（例えば、ウェブアプリケーション）のためのスクリプト言語インターフェースとすることができます、ここで、アプリケーション116は、スクリプト言語コード及び/又はマークアップ言語コードを含むことができる。アプリケーション116は、インターフェースコンポーネント114を介してスクリプト言語ベースのフレームワーク102を参照して、スクリプト言語ベースのフレームワーク102に含まれた機能性を用いることができる。その上、アプリケーション116は、インターフェースコンポーネント114を介して、スクリプト言語ベースのフレームワーク102に呼び出し（群）を提供することができる。例え

10

20

30

40

50

ば、呼び出し(群)は、コントロール 112 のうち 1 又は複数を管理することができる(例えば、ある呼び出しが、コントロール 112 の中からあるコントロールを管理することができる)。一例に従い、呼び出しは、アプリケーション 116 に含まれたスクリプト言語コードから生成することができ、したがって、アプリケーション 116 は、コントロール 112 のうち 1 又は複数を管理する 1 又は複数の呼び出しを生成することができるスクリプト言語コードを含むことができる。ここで、上記 1 又は複数の呼び出しは、インタフェースコンポーネント 114 を介してスクリプト言語ベースのフレームワーク 102 に提供されることができる。

【0018】

スクリプト言語ベースのフレームワーク 102 の中のコントロール 112 は、アプリケーション 116 に含まれるスクリプト言語コードに基づいて(例えば、インタフェースコンポーネント 114 を介してアプリケーション 116 により提供される呼び出し(群)に応答して)、有効又は無効にすることができる。その上、コントロール 112 は、(例えば、インタフェースコンポーネント 114 を介してアプリケーション 116 により提供される呼び出し(群)に応答して)アプリケーション 116 の中のスクリプト言語コードにより管理されることができる属性及び効果を有することができる。一例に従い、コントロール 112 のうち 1 又は複数は、アプリケーション 116 の中のスクリプト言語コードに基づいて、動的に様式化することができる。別の例として、コントロール 112 のうち 1 又は複数は、アプリケーション 116 の中のスクリプト言語コードに応答して設定されることができる広範な属性を有することができる。

10

20

【0019】

スクリプト言語ベースのフレームワーク 102 は、ウェブプログラマが、アプリケーション 116 を開発するときに使用することができる。スクリプト言語ベースのフレームワーク 102 は、データアクセス(例えば、データアクセスコンポーネント 104 を介して)とユーザインターフェース生成(例えば、UI 作成コンポーネント 106 を介して)に関する機能性を提供する。それに応じて、スクリプト言語ベースのフレームワーク 102 の使用は、ウェブプログラマがアプリケーション 116 を開発するときに費やす時間の量を、削減することができる。したがって、スクリプト言語ベースのフレームワーク 102 は、アプリケーション 116 のウェブプログラマの、スクリプト言語コード及び/又はマークアップ言語コードを用いてデータアクセスとユーザインターフェース作成とを管理する能力を、改善することができる。

30

【0020】

データアクセスコンポーネント 104 は、クラウドプラットフォーム 110 上の分散型データストア 108 に直接的にアクセスすることができる。それに応じて、ウェブプログラマは、シンプルなスクリプト言語コード及び/又はマークアップ言語コードを用いて、分散型データストア 108 に対して読み出し及び/又は書き込みをするアプリケーション 116 を生成することができる。ウェブプログラマは、クラウド技術、データ構造、プロトコル、及び類似のものの知識を有することなく、アプリケーション 116 を作成することができる。むしろ、アプリケーション 116 は、コントロール 112 のうち 1 又は複数を管理する(例えば、有効にする、無効にする、属性又は効果の値を設定する、様式化するなど)ことができ、アプリケーション 116 により管理されたコントロール 112 のうち 1 又は複数に基づいて、データアクセスコンポーネント 104 は、クラウドプラットフォーム 110 上の分散型データストア 108 に直接的にアクセスすることができる。

40

【0021】

分散型データストア 108 は、クラウドプラットフォーム 110 上に構築される。分散型データストア 108 は、クラウドプラットフォーム 110 上の、十分に拡大縮小可能な、複製されたデータストアとすることができる。例示的な実施形態において、分散型データストア 108 は、NoSQL ストアとすることができる。さらに、分散型データストア 108 は、検索コンポーネント 118 及びインデックス化コンポーネント 120 の上に構築することができる。検索コンポーネント 118 は、フルのテキスト検索及び/又は画像

50

検索を提供する検索エンジンとすることができる。クラウドプラットフォーム110は、さらに、検索コンポーネント118とインデックス化コンポーネント120に対する追加又は代替として、他のバックエンドサービスを提供することができる。したがって、一例に従い、データアクセスコンポーネント104は、検索コンポーネント118を用いて分散型データストア108の中のデータを検索すること、インデックス化コンポーネント120を用いて分散型データストア108の中のデータをインデックス化することなどができる。

【0022】

クラウドプラットフォーム110上の分散型データストア108は、一式のアプリケーション・プログラミング・インターフェース(APIs)122を通じて公開される。API122は、RESTfulな(表現的状態転送(representational state transfer))APIとすることができます。さらに、API122は、スクリプト言語ベースのフレームワーク102に包含することができます。例えば、データアクセスコンポーネント104は、API122を用いて分散型データストア108にアクセスすることができます。それに応じて、スクリプト言語ベースのフレームワーク102は、ウェブプログラマがAPI122をどのように活用すべきかを学習する必要がないように、API122を抽象化することができます。むしろ、インターフェースコンポーネント114は、アプリケーション116を開発するためにウェブプログラマが用いることができる簡素なスクリプト言語インターフェースを公開することができます。例として、スクリプト言語ベースのフレームワーク102において利用可能なコントロール112は、インターフェースコンポーネント114を介して公開することができます。

【0023】

その上、一式のサードパーティサービスを、分散型データストア108に包含することができます。上記一式のサードパーティサービスは、すなわち、サードパーティサービス1の(参照番号)124からサードパーティサービスNの(参照番号)126までであり、ここで、Nは、実質的に任意の整数とすることができます(本明細書において集合的にサードパーティサービス124~126と呼ぶ)。それに応じて、分散型データストア108は、サードパーティサービス124~126のAPIへのアクセスを提供することができます。したがって、データアクセスコンポーネント104は、クラウドプラットフォーム110上の分散型データストア108を介してサードパーティサービス124~126(例えば、サードパーティのバックエンドのクラウドベースのサービス)にアクセスすることができます。サードパーティサービス124~126の例には、ウェブ検索サービス、画像検索サービス、ビデオ検索サービス、ソーシャルネットワーキングサービス、マイクロブログ(microblogging)サービス、地図作成(mapping)サービス、半構造化(semi-structured)データベース検索サービス、認証サービス、クラウドプラットフォームサービスなどを含むことができる。しかしながら、特許請求される対象事項は前述のサードパーティサービス124~126の例に限定されないと、むしろ、いかなるサードパーティサービスも本願に添付の請求項の範囲に入ることが意図されることとを、十分理解されたい。

【0024】

さらに、アプリケーション116は、開発を完了すると、ウェブホスティングサービス(図示せず)にアップロードすることができます。アプリケーション116は、スクリプト言語コード及び/又はマークアップ言語コードで書くことができるので、アプリケーション116は、例えば、実質的に任意のウェブサイトに埋め込むこと又はアップロードすることができます。アプリケーション116は、利用可能になると、実行のためにクライアント装置(図示せず)のウェブブラウザに読み出すこと又はロードすることができる。さらに、アプリケーション116は、現代のデスクトップ、ラップトップ及びモバイルのブラウザを介して動作することができるように、クライアント装置に依存しない(agnostic)ものとすることができます。

【0025】

10

20

30

40

50

下記例解は、クラウドベースのデータストアの中にアドレス帳を維持するアプリケーションを開発する従来型の手法を論証するものである。ウェブプログラマは、最初に、データオブジェクトとデータオブジェクト間の関係と（例えば、人々、人々が属する組織など）を定義することができる。その後、ウェブプログラマは、一式のユーザインタフェースを構築することができる。一例に従い、ウェブプログラマは、連絡先を作成するためのユーザインタフェース、上記連絡先を他の連絡先と関連付けるためのユーザインタフェース、連絡先を更新するためのユーザインタフェース、連絡先を削除するためのユーザインタフェース、連絡先にわたって検索するためのユーザインタフェースなどを構築することができる。ユーザインタフェースを開発し、データオブジェクトを定義すると、ウェブプログラマは、一式のバックエンドサービス（例えば、バックエンドのクラウドベースのサービス）を構築すること、及び／又は一式のバックエンドサービスにアクセスするシステムを構築することができる。バックエンドサービスの例には、クラウドベースのデータストアにデータを書き込むサービス、クラウドベースのデータストアからデータを読み出すサービス、クラウドベースのデータストアに保有されているデータにわたって検索するサービス、クラウドベースのデータストアの中のデータを編集するサービス、クラウドベースのデータストアからデータを削除するサービスなどを含むことができる。前述の手法は、ウェブプログラマにとって時間のかかるものとなるおそれがある。その上、こうした開発は、種々のプロトコル、データ構造などについてのかなりの理解を伴う場合がある。

【0026】

対照的に、スクリプト言語ベースのフレームワーク102は、クラウドプラットフォーム110上の分散型データストア108に組み込まれる。その上、スクリプト言語ベースのフレームワーク102は、アプリケーション116に含まれるスクリプト言語コードの機能として管理されることができる一式のコントロール112を含む。さらに、コントロール112は、クラウドベースのバックエンドシステム（例えば、クラウドプラットフォーム110上の分散型データストア108）にアクセスするように、スクリプト言語ベースのフレームワーク102の内部に予めプログラムされることができる。したがって、ウェブプログラマが広範なバックエンドプログラミングをユーザインタフェース開発と同様に行わなければならないことに代わって、ウェブプログラマは、スクリプト言語コード及び／又はマークアップ言語コードを書いてスクリプト言語ベースのフレームワーク102のコントロール112を管理することができ、コントロール112は、クラウドプラットフォーム110上の分散型データストア108の豊富なバックエンドに組み込まれる。

【0027】

次に、図2を参照し、サードパーティサービスを用いるウェブアプリケーションを開発することを容易にするシステム200を示す。システム200は、スクリプト言語ベースのフレームワーク102を含み、システム200はさらに、データアクセスコンポーネント104、UI作成コンポーネント106、一式のコントロール112、及びインタフェースコンポーネント114を備えることができる。上記で説明されたとおり、データアクセスコンポーネント104は、API122を介してクラウドプラットフォーム110上の分散型データストア108に直接的にアクセスすることができる。

【0028】

その上、1又は複数のサードパーティサービスを、スクリプト言語ベースのフレームワーク102に直接的に組み込むことができる。したがって、サードパーティサービス1の（参照番号）202からサードパーティサービスMの（参照番号）204まで（本明細書において集合的にサードパーティサービス202～204と呼ぶ）のAPIを、スクリプト言語ベースのフレームワーク102のデータアクセスコンポーネント104に対して公開することができる。ここで、Mは、実質的に任意の整数とすることができます。スクリプト言語ベースのフレームワーク102は、サードパーティサービス202～204のAPIを抽象化することができる。それに応じて、アプリケーション116は、コントロール112のうち1又は複数を管理するスクリプト言語コードを含むことができ、ここで、コントロール112は、アプリケーション116のウェブプログラマが1又は複数のサード

10

20

30

40

50

パーティサービス 202～204 の API についてのデータ構造、プロトコルなどの知識を有する必要なく、データアクセスコンポーネント 104 にサードパーティサービス 202～204 のうち 1 又は複数を用いさせることができる。したがって、上記の組み込みがスクリプト言語ベースのフレームワーク 102 に包含されているため、ウェブプログラマは、アプリケーション 116 を開発するときに 1 又は複数のサードパーティサービス 202～204 にアクセスするコードを書く必要がなくなる。

【0029】

サードパーティサービス 202～204 の例には、ウェブ検索サービス、画像検索サービス、ビデオ検索サービス、ソーシャルネットワーキングサービス、マイクロブログサービス、地図作成サービス、半構造化データベース検索サービス、認証サービス、クラウドプラットフォームサービスなどを含むことができる。しかしながら、特許請求される対象事項は前述のサードパーティサービス 202～204 の例に限定されない。一例に従い、サードパーティサービスは、分散型データストア 108 とデータアクセスコンポーネント 104 との双方に組み込まれることができる（例えば、サードパーティサービスは、サードパーティサービス 124～126 のうちの 1 つとサードパーティサービス 202～204 のうちの 1 つとすることができます）。さらに、又は別法として、サードパーティサービスは、分散型データストア 108 又はデータアクセスコンポーネント 104 のいずれかに組み込まれることができます（例えば、サードパーティサービスは、サードパーティサービス 124～126 のうちの 1 つ、又はサードパーティサービス 202～204 のうちの 1 つとすることができます）。

10

【0030】

例として、クライアント装置（例えば、アプリケーション 116 を実行している装置）のインターネットプロトコル（IP）アドレスが、（例えば、上記 IP アドレスに関連するコンテンツを返すために）サードパーティサービスに提供されることになるとき、コントロール（例えば、コントロール 112 の中の）は、データアクセスコンポーネント 104 に、IP アドレスをサードパーティサービスに間接的に渡すのではなく、IP アドレスをサードパーティサービスに直接的に供給させることができます。したがって、データアクセスコンポーネント 104 は、クラウドプラットフォーム 110 上の分散型データストア 108 を介してアクセス可能なサードパーティ 124～126 のうち 1 又は複数に IP アドレスを送信するのではなく、データアクセスコンポーネント 104 に直接的に組み込まれたサードパーティ 202～204 のうち 1 又は複数に IP アドレスを送信することができます。

20

【0031】

次に、図 3 を参照し、スクリプト言語ベースのフレームワーク 102 を用いてアプリケーション 116 を開発するためのシステム 300 を示す。スクリプト言語ベースのフレームワーク 102 は、クラウドベースのバックエンドシステム（群）（例えば、図 1 のクラウドプラットフォーム 110 上の分散型データストア 108、図 2 のサードパーティサービス 202～204 など）にアクセスすることができるデータアクセスコンポーネント 104 と、ユーザインターフェースを生成する UI 作成コンポーネント 106 と、アプリケーション 116 とインターフェースをとることができるインターフェースコンポーネント 114 とを含む。

30

【0032】

その上、スクリプト言語ベースのフレームワーク 102 は、コントロール 112 を含む。コントロール 112 は、例えば、モデル・ビュー・コントローラ（MVC）モデルに基づくことができる。コントロール 112 は、アプリケーション 116 に含まれたスクリプト言語コードの機能として、有効にすること又は無効にすることができる。例えば、コントロール 112 のうち 1 又は複数がアプリケーション 116 において有効にされるとき、有効にされたコントロール（群） 112 は、UI 作成コンポーネント 106 にユーザインターフェース（群）を生成させることができる。さらに、コントロール 112 は、コントロール 112 に適用することができる広範な属性及び効果を有することができる。1 つの例

40

50

解に従い、コントロール 112 のうち 1 又は複数は、アプリケーション 116 に含まれたスクリプト言語コードに基づいて様式化することができる。例えば、コントロール 112 のうち 1 又は複数をアプリケーション 116 が管理して、UI 作成コンポーネント 106 に、クライアント装置のディスプレイにレンダリングされるユーザインタフェース（群）をアニメ化させることができる。見込まれるアニメーションを、スクリプト言語ベースのフレームワーク 102 に含まれるコントロール 112 に組み込むことができる。さらなる例に従い、スクリプト言語ベースのフレームワーク 102 に含まれるコントロール 112 は、互いに依存関係を有することができる。この例に従い、コントロール 112 間の上記依存関係は、UI 作成コンポーネント 106 が、生成されたユーザインタフェース（群）に豊富なアニメーションを提供することを、可能にすることができます。別の例として、コントロール 112 は、例えば、氏名、敬称などの複数の属性を有することができる。

【0033】

一例に従い、コントロール 112 のうち 1 又は複数は、アプリケーション 116 のための認証を管理することができる。例えば、アプリケーション 116 は、一式の利用可能な認証スキーム（例えば、種々のサードパーティサービスにより提供される）から認証スキームを選択するスクリプト言語コードを含むことができる。その後、選択された認証スキームを、バックエンド（例えば、図 1 のクラウドプラットフォーム 110 上の分散型データストア 108）からフロントエンドまでシステムにわたって用いることができる。

【0034】

さらに、UI 作成コンポーネント 106 は、アプリケーション 116 を実行しているクライアント装置の能力を検出することができ、かつ、検出された能力の機能のとおりに、コントロール 112 のうち 1 又は複数に基づいて生成されるユーザインタフェースを調整することができる、調整コンポーネント 302 を含むことができる。例えば、アプリケーション 116 がスマートフォンにロードされた場合、調整コンポーネント 302 は、クライアント装置がカメラ及びマイクロフォンを含むことを、検出することができる。この例に従い、調整コンポーネント 302 は、コントロール 112 のうち 1 又は複数に基づいて、UI 作成コンポーネント 106 により生成されるユーザインタフェースを変更して、スマートフォンのカメラ及びマイクロフォンに関連する特性を含ませることができる。一方で、調整コンポーネント 302 が、クライアント装置はカメラ及びマイクロフォンを欠いていることを検出したとき、上記の特性は、UI 作成コンポーネント 106 により生成されるユーザインタフェースにおいて省略することができる。調整コンポーネント 302 を含むことによって、スクリプト言語ベースのフレームワーク 102 は、アプリケーション 116 が、アプリケーション 116 を実行しているクライアント装置の能力を用いることを、可能にすることができます。その上、アプリケーション 116 は、ウェブプログラマによる様々な種類の装置をサポートするためのコーディングなく、スクリプト言語ベースのフレームワーク 102 に包含された調整コンポーネント 302 を活用することによって、モバイル装置、デスクトップ、ラップトップなどにわたって動作することができる。

【0035】

UI 作成コンポーネント 106 は、さらに、段階的にアプリケーション 116 の特性をロールアウトすることができるフライティング（flying）コンポーネント 304 を含むことができる。例えば、ユーザのサブセットがアプリケーション 116 のある特性へのアクセスを有することを、ウェブプログラマが所望する場合がある。したがって、フライティングコンポーネント 304 は、上記ユーザのうち上記サブセットに対して生成されるユーザインタフェース（群）においてアプリケーション 116 の上記特性を提供することができると同時に、アプリケーション 116 の上記ユーザのうち残りに対して生成されるユーザインタフェース（群）において上記特性を隠すことができる。スクリプト言語ベースのフレームワーク 102 にフライティングコンポーネント 304 を含むことによって、アプリケーション 116 の特性の交互の（staggered）ロールアウトを可能にするカスタムコードを、ウェブプログラマによって書く必要がなくなる。

【0036】

10

20

30

40

50

その上、スクリプト言語ベースのフレームワーク 102 は、コントロール 112 から通知を受信し、かつコントロール 112 に通知を送信する、通知コンポーネント 306 を含むことができる。したがって、スクリプト言語ベースのフレームワーク 102 において、通知コンポーネント 306 は、コントロール 112 間の情報を交換することができ、これにより、コントロール 112 間で依存関係を有することを可能にすることができる。1つの例解に従い、アプリケーション 116 は、スクリプト言語シェルとすることができる、上記スクリプト言語シェルは、複数のコントロール 112 に対する呼び出しを生成するスクリプト言語コードを含むことができる。この例解に従い、アプリケーション 116 の内部で呼び出された上記複数のコントロール 112 は、シェル全体の (shell-wide) 通知を通知コンポーネント 306 から受信することができ、これにより上記複数のコントロール 112 の間で相互作用を容易にすることができます。
10

【0037】

例えは、2つのコントロール（例えは、コントロール 112 からの）を、ウェブページに含むことができ、例えは、一方のコントロールは氏名のリストビューとすることができます、他方のコントロールは詳細ビューとすることができます。この例に従い、上記2つのコントロールは、通知コンポーネント 306 を介して情報を交換することができる。したがって、例えは、ある氏名がリストビューの中で選択された（例えは、クリックされた）場合、詳細ビューは、上記リストビューの中で選択された氏名に対応する詳細を示すことができる。その上、通知コンポーネント 306 は、コントロール 112 により使用される情報を交換してイベント連鎖（chaining）を生じさせることができる。それに応じて、コントロール 112 の振る舞いは、イベント（例えは、データをロードすること、トランザクションを処理すること、アニメーションの完了、受信されたユーザ入力など）に基づいて連鎖されることができる。
20

【0038】

さらに、データアクセスコンポーネント 104 は、コントロール 112 をデータソース（例えは、クラウドベースのバックエンドシステム（群））にバインドすることができるソースバインドコンポーネント 308 を含むことができる。一例に従い、ソースバインドコンポーネント 308 は、コントロール 112 のうち2つを、種々のソースからのオブジェクトにバインドすることができる。この例に従い、ソースバインドコンポーネント 308 は、コントロール 112 のうち一方を第1のソーシャルネットワーキングサービスからのオブジェクトに、コントロール 112 のうち他方を第2のソーシャルネットワークサービスからのオブジェクトに、バインドすることができる。しかしながら、特許請求される対象事項は前述の例に限定されないことを、十分理解されたい。ソースバインドコンポーネント 308 は、コントロール 112 を種々のデータソースにバインドするため、データアクセスコンポーネント 104 は、種々のデータソースからの情報のための標準化スキーマを用いることができる。スキーマのこうした標準化は、コントロール 112 間の情報の柔軟な交換を可能にすることができる。種々の実施形態において、ソースバインドコンポーネント 308 は、コントロール 112 のうち1又は複数を、オープンデータプロトコル（O D d a t a ）ソースにバインドすることができるが、特許請求される対象事項はそのように限定されないことを、十分理解されたい。
30
40

【0039】

図4～図5は、スクリプト言語ベースのフレームワークを用いてウェブアプリケーションを開発することに関連する例示的な方法論を示す。上記方法論は、ある並びで実行される一連の動作であるとして図示及び説明されるが、上記方法論は上記並びの順序によって限定されないことを、把握及び十分理解されたい。例えは、いくつかの動作が、本明細書において説明される順序とは異なる順序で発生してよい。加えて、ある動作が、別の動作と同時に発生してよい。さらに、いくつかの例において、すべての動作が本明細書において説明される方法論を実装するために必要とされるわけではない場合がある。

【0040】

その上、本明細書において説明される動作は、1若しくは複数のプロセッサにより実施

すること及び／又は1若しくは複数のコンピュータ読取可能媒体に記憶することができる、コンピュータ実行可能命令とすることができる。コンピュータ実行可能命令は、ルーチン、サブルーチン、プログラム、実行のスレッド、及び／又は類似のものを含むことができる。またさらに、上記方法論の動作の結果を、コンピュータ読取可能媒体に記憶すること、ディスプレイ装置に表示すること、及び／又は類似のことができる。

【0041】

図4は、ウェブアプリケーションを開発するための方法論400を示す。402において、スクリプト言語ベースのフレームワークのためのインターフェースを、公開することができる。404において、呼び出しを、上記インターフェースを介して受けることができる。上記インターフェースを介して受けた呼び出しは、スクリプト言語ベースのフレームワークに含まれるコントロールを管理することができる。さらに、上記呼び出しは、スクリプト言語ベースのフレームワークを参照するアプリケーションにより提供されることができる。例えば、上記呼び出しは、コントロールを有効にすること、コントロールを無効にすること、コントロールの属性又は効果を設定することなどができる。406において、ユーザインターフェースを、上記呼び出しにより管理されたコントロールに基づいて生成することができる。408において、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスに、上記呼び出しにより管理されたコントロールに基づいてアクセスすることができる。

【0042】

次に、図5を参照し、ウェブアプリケーションを開発するための方法論500を示す。502において、一式のコントロールを含むスクリプト言語ベースのフレームワークのためのインターフェースを、公開することができ、上記一式のコントロールは、バックエンドのクラウドベースのサービスにアクセスし、かつ、有効にされたときにユーザインターフェースを生成するように、予めプログラムされる。504において、スクリプト言語ベースのフレームワークに含まれる一式のコントロールの中のコントロールを有効にすること、呼び出しを、上記インターフェースを介して受けることができる。上記呼び出しは、スクリプト言語ベースのフレームワークを参照するアプリケーションにより提供されることができる。506において、ユーザインターフェースを、上記呼び出しにより有効にされたコントロールに基づいて生成することができる。508において、クラウドプラットフォーム上の分散型データストアにより提供されるバックエンドのクラウドベースのサービスに、上記呼び出しにより有効にされたコントロールに基づいてアクセスすることができる。

【0043】

次に、図6を参照し、本明細書において開示されるシステム及び方法論に従って使用することができる例示的なコンピューティング装置600の高水準の例解を示す。例えば、コンピューティング装置600は、スクリプト言語ベースのフレームワークを用いてウェブアプリケーションを開発するシステムにおいて使用されることができる。別の例として、コンピューティング装置600は、スクリプト言語ベースのフレームワーク上に構築されたウェブアプリケーションを実行するシステムにおいて使用されることができる。コンピューティング装置600は、少なくとも1つのプロセッサ602を含み、上記少なくとも1つのプロセッサ602は、メモリ604に記憶された命令を実行する。上記命令は、例えば、上記で論じられた1若しくは複数のコンポーネントにより実行されるとして説明された機能性を実施する命令、又は上記で説明された方法のうち1若しくは複数を実施する命令とすることができます。プロセッサ602は、システムバス606を通じてメモリ604にアクセスすることができる。実行可能命令を記憶することに加えて、メモリ604は、アプリケーション、スクリプト言語ベースのフレームワークなどを記憶することもできる。

【0044】

コンピューティング装置600は、さらに、データストア608を含み、データストア608は、システムバス606を通じてプロセッサ602によりアクセスすることができる。データストア608は、実行可能命令、アプリケーション、スクリプト言語ベースの

10

20

30

40

50

フレームワークなどを含むことができる。コンピューティング装置 600 はさらに、入力インタフェース 610 を含み、入力インタフェース 610 は、外部装置がコンピューティング装置 600 とやりとりすることを可能にする。例えば、入力インタフェース 610 を用いて、外部のコンピュータ装置、ユーザなどから指示を受け取ることができる。コンピューティング装置 600 はさらに、出力インタフェース 612 を含み、出力インタフェース 612 は、コンピューティング装置 600 を 1 又は複数の外部装置とインターフェースでつなぐ。例えば、コンピューティング装置 600 は、出力インタフェース 612 を通じて、テキスト、画像などを表示することができる。

【 0045 】

さらに、単一のシステムとして図示されているが、コンピューティング装置 600 は分散型システムとすることを可能であることを、理解されたい。したがって、例えば、いくつかの装置が、ネットワーク接続を経由して通信することができ、コンピューティング装置 600 により実行されるとして説明されたタスクを集合的に実行することができる。

【 0046 】

本明細書において、「コンポーネント」及び「システム」という用語は、コンピュータ実行可能命令を用いて構成されたコンピュータ読取可能データ記憶装置を包含することができる。上記コンピュータ実行可能命令は、プロセッサにより実行されると、特定の機能性を実行させる。コンピュータ実行可能命令は、ルーチン、関数、又は類似のものを含むことができる。コンポーネント又はシステムは、単一の装置上に局在させてよく、あるいはいくつかの装置にわたって分散させてよい。

【 0047 】

さらに、本明細書において、「例示的」という用語は、「何かの例解又は例としての役割を果たすこと」を意味することができる。

【 0048 】

本明細書において説明される種々の機能は、ハードウェア、ソフトウェア、又はそれらの組み合わせにおいて実装することができる。ソフトウェアにおいて実装される場合、上記機能は、コンピュータ読取可能媒体上の 1 又は複数の命令又はコードとして、記憶すること又は伝達することができる。コンピュータ読取可能媒体には、コンピュータ読取可能記憶媒体を含む。コンピュータ読取可能記憶媒体は、コンピュータがアクセスすることができる任意の利用可能な記憶媒体とることができる。限定ではなく例として、こうしたコンピュータ読取可能記憶媒体には、RAM、ROM、EEPROM、CD-ROM 若しくは他の光学ディスク記憶装置、磁気ディスク記憶装置若しくは他の磁気記憶装置、又は、命令若しくはデータ構造の形式で所望のプログラムコードを担体若しくは記憶するために使用することができ、かつコンピュータによりアクセスすることができる任意の他の媒体を、含むことができる。本明細書において、磁気ディスク (Disk) 及びディスク (disc) には、コンパクトディスク (CD)、レーザーディスク (登録商標)、光ディスク、デジタル多用途ディスク (DVD)、フロッピー (登録商標) 磁気ディスク、及びブルーレイディスク (BD) を含み、ここで、磁気ディスクは一般に磁気的にデータを再現し、ディスクは一般にレーザーを用いて光学的にデータを再現する。さらに、伝播される信号は、コンピュータ読取可能記憶媒体の範囲の中に含まれない。コンピュータ読取可能媒体は、さらに、1 つの場所から別の場所へのコンピュータプログラムの転送を容易にする任意の媒体を含む通信媒体を含む。接続が、例えば、通信媒体であり得る。例えば、ソフトウェアが、同軸ケーブル、光ファイバケーブル、ツイストペア、デジタル加入者回線 (DSL)、又は赤外線、無線及びマイクロ波などの無線技術を用いて、ウェブサイト、サーバ、又は他の離れた情報源から送信される場合、上記の同軸ケーブル、光ファイバケーブル、ツイストペア、DSL、又は赤外線、無線及びマイクロ波などの無線技術は、通信媒体の定義に含まれる。上記の組み合わせもまた、コンピュータ読取可能媒体の範囲の中に含まれるべきである。

【 0049 】

上記で説明されたものには、1 又は複数の実施形態の例を含む。前述された態様を説明

10

20

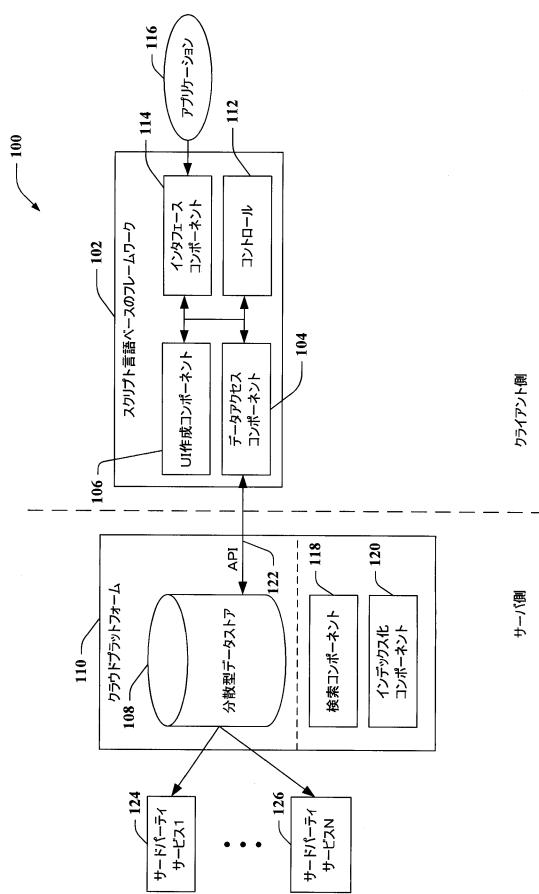
30

40

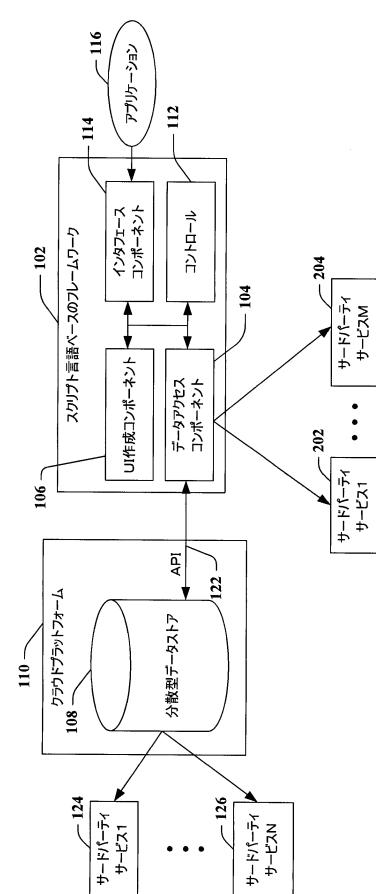
50

する目的で上記の装置又は方法論のあらゆる考えられる変更及び調整を説明することは、もちろん不可能であり、しかしながら当業者は、種々の態様の多くのさらなる変更及び置換があり得ることを認識し得る。それに応じて、上記の説明された態様は、添付の請求項の主旨及び範囲に入るすべての上記のような調整、変更及び変形を包含することが意図される。その上、「含む」という用語が本願の詳細説明又は請求項のいずれかにおいて使用される限りにおいて、上記の用語は、「備える」という用語に類似の、「備える」が請求項において移行語として用いられるときに解釈されるような方式で、包括的であることが意図される。

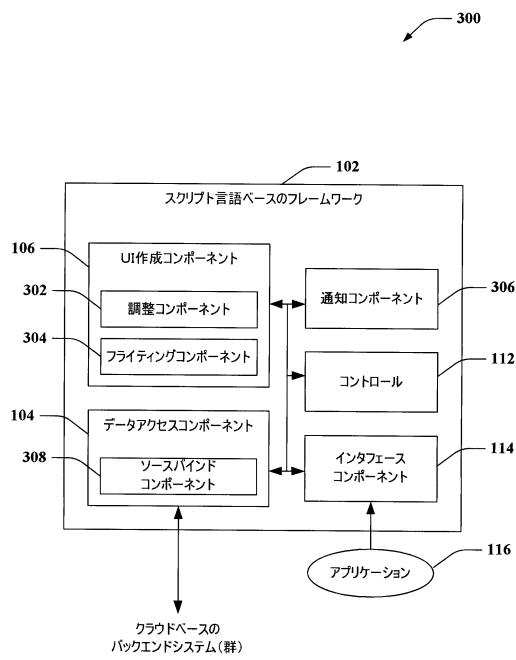
【 义 1 】



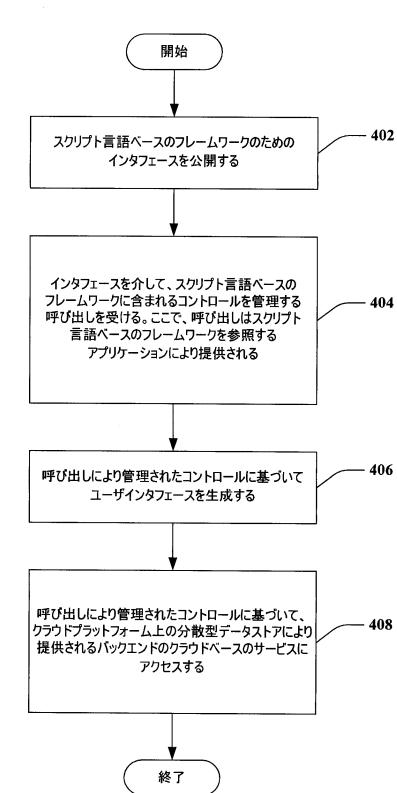
【 図 2 】



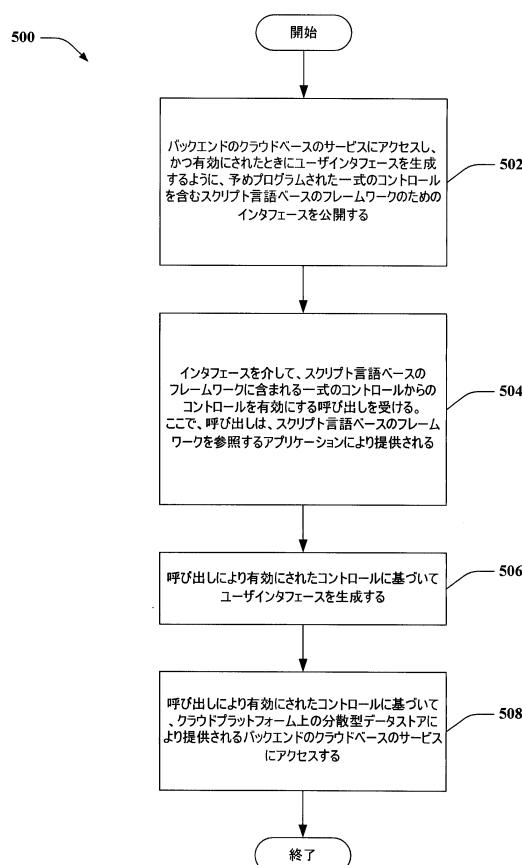
【図3】



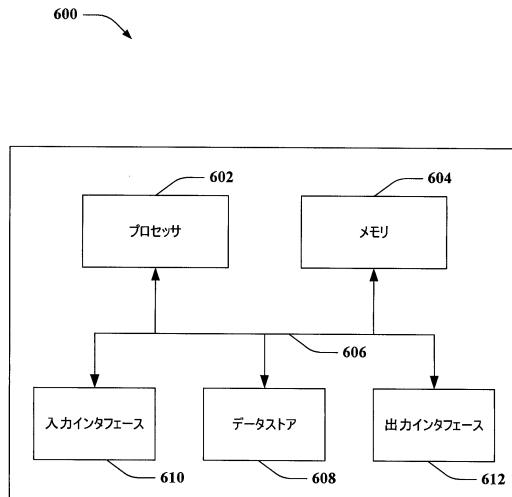
【図4】



【図5】



【図6】



フロントページの続き

(72)発明者 イックマン, スティーヴン

アメリカ合衆国 98052-6399 ワシントン州 レッドモンド ワン マイクロソフト
ウェイ マイクロソフト コーポレーション エルシーエー - インターナショナル パテンツ 内

(72)発明者 ラスキノ, デイヴィッド

アメリカ合衆国 98052-6399 ワシントン州 レッドモンド ワン マイクロソフト
ウェイ マイクロソフト コーポレーション エルシーエー - インターナショナル パテンツ 内

審査官 長谷川 篤男

(56)参考文献 特表2006-512694 (JP, A)

特開2010-117793 (JP, A)

特開2009-086759 (JP, A)

川田 徹, Visual Basic 入門講座 [3], I/O 1999年6月号, 株式会社工学
社, 1999年 6月 1日, 第24巻 第6号, pp102-105

(58)調査した分野(Int.Cl., DB名)

G 06 F 9 / 44