

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6110950号
(P6110950)

(45) 発行日 平成29年4月5日 (2017.4.5)

(24) 登録日 平成29年3月17日 (2017.3.17)

(51) Int. Cl.

F I

G 0 6 F 13/00 (2006.01)

G 0 6 F 13/00 5 4 0 A

請求項の数 20 (全 21 頁)

(21) 出願番号 特願2015-537747 (P2015-537747)
 (86) (22) 出願日 平成25年10月11日 (2013.10.11)
 (65) 公表番号 特表2015-535376 (P2015-535376A)
 (43) 公表日 平成27年12月10日 (2015.12.10)
 (86) 国際出願番号 PCT/US2013/064485
 (87) 国際公開番号 W02014/062490
 (87) 国際公開日 平成26年4月24日 (2014.4.24)
 審査請求日 平成27年5月15日 (2015.5.15)
 (31) 優先権主張番号 201210395535.4
 (32) 優先日 平成24年10月17日 (2012.10.17)
 (33) 優先権主張国 中国 (CN)
 (31) 優先権主張番号 14/050,841
 (32) 優先日 平成25年10月10日 (2013.10.10)
 (33) 優先権主張国 米国 (US)

(73) 特許権者 510330264
 アリババ・グループ・ホールディング・リ
 ミテッド
 ALIBABA GROUP HOLDI
 NG LIMITED
 英国領、ケイマン諸島、グランド・ケイマ
 ン、ジョージ・タウン、ワン・キャピタル
 ・プレイス、フォース・フロア、ビー・オ
 ー、ボックス 847
 (74) 代理人 110000028
 特許業務法人明成国際特許事務所

最終頁に続く

(54) 【発明の名称】 オンボードハードウェア関数を備えたウェブページ

(57) 【特許請求の範囲】

【請求項 1】

クライアントデバイス上で動作するクライアントアプリケーションにウェブページデー
 タを送信するためのシステムであって、

1 または複数のプロセッサであって、

前記クライアントアプリケーションによって送信されたウェブページ要求を受信し、

前記ウェブページ要求内のマーキングパラメータを検出し、前記マーキングパラメータは、前記ウェブページ要求を送信した前記クライアントアプリケーションのクライアントタイプを表し、

前記マーキングパラメータに少なくとも部分的に基づいて、前記クライアントアプリ
 ケーションの前記クライアントタイプを判定し、

前記クライアントタイプがブラウザクライアントアプリケーションであると判定され
 た場合に、ウェブページ情報を含む第1のウェブページデータを前記クライアントアプリ
 ケーションに送信し、

前記クライアントタイプが非ブラウザクライアントアプリケーションであると判定され
 た場合に、ウェブページ情報と、前記クライアントデバイス上でオンボードハードウェ
 ア関数を実行するためのコードと、を含む第2のウェブページデータを前記クライアント
 アプリケーションに送信するよう構成された、1 または複数のプロセッサと、

前記1 または複数のプロセッサに接続され、前記1 または複数のプロセッサに命令を提
 供するよう構成された1 または複数のメモリと、

10

20

を備える、システム。

【請求項 2】

請求項 1 に記載のシステムであって、前記クライアントタイプが非ブラウザクライアントアプリケーションであると判定された場合に、前記 1 または複数のプロセッサは、さらに、前記ウェブページ要求に含まれるプラットフォーム情報を用いて、前記クライアントアプリケーションが動作するプラットフォームを識別するよう構成されている、システム。

【請求項 3】

請求項 2 に記載のシステムであって、前記 1 または複数のプロセッサは、さらに、前記プラットフォーム上で前記オンボードハードウェア関数を実行するために必要なサポート情報を送信するよう構成されている、システム。

10

【請求項 4】

請求項 3 に記載のシステムであって、前記サポート情報はプラットフォーム固有である、システム。

【請求項 5】

請求項 1 に記載のシステムであって、前記オンボードハードウェア関数は、カメラ関数、マイク関数、または、それら両方を含む、システム。

【請求項 6】

請求項 3 に記載のシステムであって、前記サポート情報は、前記オンボードハードウェア関数に関する API 情報を含む、システム。

20

【請求項 7】

クライアントアプリケーションにウェブページデータを送信するための方法であって、前記クライアントアプリケーションによって送信されたウェブページ要求を受信する工程と、

前記ウェブページ要求内のマーキングパラメータを検出する工程であって、前記マーキングパラメータは、前記ウェブページ要求を送信した前記クライアントアプリケーションのクライアントタイプを表す、工程と、

前記マーキングパラメータに少なくとも部分的に基づいて、前記クライアントアプリケーションの前記クライアントタイプを判定する工程と、

前記クライアントタイプがブラウザクライアントアプリケーションであると判定された場合に、ウェブページ情報を含む第 1 のウェブページデータを前記クライアントアプリケーションに送信する工程と、

30

前記クライアントタイプが非ブラウザクライアントアプリケーションであると判定された場合に、ウェブページ情報と、オンボードハードウェア関数を実行するためのコードと、を含む第 2 のウェブページデータを前記クライアントアプリケーションに送信する工程と、

を備える、方法。

【請求項 8】

表示するためのウェブページを生成するためのシステムであって、

1 または複数のプロセッサであって、

40

前記システム上で動作するクライアントアプリケーションのクライアントタイプを識別するマーキングパラメータを含むウェブページ要求を生成し、

前記クライアントアプリケーションの前記クライアントタイプは非ブラウザクライアントアプリケーションであり、前記マーキングパラメータは、前記ウェブページ要求を送信した前記クライアントアプリケーションの前記クライアントタイプを表し、

前記クライアントアプリケーションを用いて、前記ウェブページ要求を送信し、

前記ウェブページ要求に応答して送信されたウェブページデータを受信し、

前記ウェブページデータは、ウェブページ情報と、前記システム上で指定のオンボードハードウェア関数を実行するために用いられるコードとを含み、

前記ウェブページデータを用いて、前記オンボードハードウェア関数を提供すると共

50

に表示される前記ウェブページを生成するよう構成された、１または複数のプロセッサと、

前記１または複数のプロセッサに接続され、前記１または複数のプロセッサに命令を提供するよう構成された１または複数のメモリと、
を備える、システム。

【請求項 9】

請求項 8 に記載のシステムであって、前記１または複数のプロセッサは、さらに、

前記クライアントアプリケーションが動作するプラットフォームを識別するプラットフォーム情報を取得し、

前記プラットフォーム情報に少なくとも部分的に基づいて、前記システム上で前記指定のオンボードハードウェア関数を実行するために用いられる前記コードによって必要とされるサポート情報を取得するよう構成されている、システム。

10

【請求項 10】

請求項 8 に記載のシステムであって、前記ウェブページデータを用いて、前記ウェブページを生成することは、

ページレンダリングエンジン呼び出して、前記ウェブページ情報をレンダリングし、

前記オンボードハードウェア関数を実行するために用いられる前記コードを、前記レンダリングされたウェブページ情報に追加することを含む、システム。

【請求項 11】

請求項 10 に記載のシステムであって、前記ウェブページデータを用いて、前記ウェブページを生成することは、

前記オンボードハードウェア関数を実行するためのサポート情報を、前記レンダリングされたウェブページ情報に追加することを含む、システム。

20

【請求項 12】

請求項 8 に記載のシステムであって、前記オンボードハードウェア関数は、カメラ関数、マイク関数、または、それら両方を含む、システム。

【請求項 13】

請求項 8 に記載のシステムであって、前記オンボードハードウェア関数は、前記ウェブページに関連付けられたボタン、リンク、またはイベントトリガを通して呼び出される、システム。

30

【請求項 14】

請求項 9 に記載のシステムであって、前記サポート情報は、前記オンボードハードウェア関数に関する A P I 情報を含む、システム。

【請求項 15】

表示するためのウェブページを生成するための方法であって、

クライアントデバイス上で動作するクライアントアプリケーションのクライアントタイプを識別するマーキングパラメータを含むウェブページ要求を生成する工程であって、前記クライアントアプリケーションの前記クライアントタイプは非ブラウザクライアントアプリケーションであり、前記マーキングパラメータは、前記ウェブページ要求を送信した前記クライアントアプリケーションの前記クライアントタイプを表す、工程と、

40

前記クライアントアプリケーションを用いて、前記ウェブページ要求を送信する工程と、

前記ウェブページ要求にตอบสนองして送信されたウェブページデータを受信する工程であって、前記ウェブページデータは、ウェブページ情報と、指定のオンボードハードウェア関数を実行するために用いられるコードとを含む、工程と、

前記ウェブページデータを用いて、前記クライアントデバイス上で前記オンボードハードウェア関数を提供すると共に表示される前記ウェブページを生成する工程と、
を備える、方法。

【請求項 16】

請求項 1 に記載のシステムであって、前記マーキングパラメータは、前記ウェブページ

50

要求に関連するユニバーサルリソースロケータ（URL）に追加される列に対応する、システム。

【請求項 17】

請求項 1 に記載のシステムであって、前記マーキングパラメータに少なくとも部分的に基づいて、前記クライアントアプリケーションの前記クライアントタイプを判定することは、前記クライアントデバイス上で動作する前記クライアントアプリケーションがブラウザアプリケーションに対応するか非ブラウザアプリケーションに対応するかを判定することを含む、システム。

【請求項 18】

請求項 1 に記載のシステムであって、前記ウェブページ要求は、前記マーキングパラメータおよびプラットフォーム情報を含み、前記プラットフォーム情報は、前記クライアントデバイスのオペレーティングシステムを識別する、システム。

10

【請求項 19】

請求項 8 に記載のシステムであって、前記マーキングパラメータは、前記システム上で動作する前記クライアントアプリケーションがブラウザアプリケーションに対応するか非ブラウザアプリケーションに対応するかを識別する、システム。

【請求項 20】

請求項 8 に記載のシステムであって、前記ウェブページ要求は、さらに、前記システムのエベレーティングシステムを識別するプラットフォーム情報を含む、システム。

【発明の詳細な説明】

20

【技術分野】

【0001】

他の出願の相互参照

本願は、2012年10月17日出願の中国特許出願第201210395535.4号「METHOD FOR SENDING BACK PAGES; METHOD AND DEVICE FOR GENERATING PAGES」の優先権を主張する。当該出願は、すべての目的のために参照により本明細書に組み込まれる。

【0002】

本願は、ウェブページ技術の分野に関し、特に、異なるタイプのクライアントでウェブページを生成および表示するための技術に関する。

30

【背景技術】

【0003】

スマートフォンの発展で、ますます多くのウェブサイトが携帯電話クライアントを導入している。ユーザが携帯電話でウェブサイトログオンする時、ユーザは、ブラウザで、または、ウェブサイトによって提供された特別なクライアント（インスタントメッセージング・クライアントなど）でログオンしうる。

【0004】

ユーザ操作を容易にするために、カメラ、マイク、または、その他のハードウェア装置のための関数が、しばしば、電話クライアントに追加される。例えば、携帯電話マイクロブログ・クライアント・アプリケーションが、リアルタイム写真撮影およびアップロード関数を有し、また、電話のインスタントメッセージング・クライアントが、音声チャット関数を有しうる。既存のブラウザアプリケーションは、通例、データ入力、ファイルアップロードなどの関数のみを許容して、ブラウザアプリケーションを動作させるデバイス上の特定のハードウェアのアクセスを許容しないので、かかる関数は、通例、ブラウザに表示されたウェブページからは省かれる。

40

【0005】

通例、ユーザがブラウザおよびクライアントを用いてウェブサイトを訪問すると、サーバは、同じウェブページデータを送り返す。ブラウザは、ウェブページデータを受信した後、ウェブページデータ内のウェブページ情報を用いて、表示すべきウェブページを生成することができる。クライアントがウェブページデータを受信した後、クライアントプロ

50

グラムが、ウェブページデータ内のウェブページ情報から生成されたウェブページにハードウェア関数を追加する。すなわち、クライアントプログラムは、オンボードハードウェア情報を呼び出して、ハードウェア関数を生成してウェブページに追加し、最終的に、ハードウェア関数を含むウェブページを表示する。したがって、ブラウザおよびクライアントの両方がログインのために利用されうるが、それらの特定の関数は、ウェブページ的主要コンテンツが基本的に同じであっても、ウェブサイトサーバによって送り返されたウェブページデータに関して異なっている。

【 0 0 0 6 】

上述の方法のように、ウェブページに表示される内容を変更する場合、通常は、サーバ側で修正を行う必要がある。さらに、ハードウェア関連関数を変更する場合には、クライアント側で修正を行った後に、クライアントをアップグレードする必要がある。サーバおよびクライアントは、別個に修正される必要があり、修正された関数は、クライアントがアップグレードされるまでは体験できない。また、クライアントは、アップグレード中には操作できない。アップグレード処理は、開発者およびユーザの双方にとって複雑で時間がかかる処理である。

【図面の簡単な説明】

【 0 0 0 7 】

以下の詳細な説明と添付の図面において、本発明の様々な実施形態を開示する。

【 0 0 0 8 】

【図 1 A】オンボードハードウェア関数を備えたウェブページをサポートする環境の一実施形態を示すブロック図。

【 0 0 0 9 】

【図 1 B】ウェブページ情報を送信するための処理の一実施形態を示すフローチャート。

【 0 0 1 0 】

【図 2】ウェブページを生成して表示するための処理の一実施形態を示すフローチャート。

【 0 0 1 1 】

【図 3】ウェブページ情報を送信するためのデバイスの一実施形態を示す構造図。

【 0 0 1 2 】

【図 4】本願の一実施形態に記載のウェブページを生成するためのデバイスの一実施形態を示す構造図。

【 0 0 1 3 】

【図 5】ブラウザアプリケーションおよび非ブラウザクライアントアプリケーションによって表示されるウェブページの例を示すユーザインターフェースの図。

【発明を実施するための形態】

【 0 0 1 4 】

本発明は、処理、装置、システム、物質の組成、コンピュータ読み取り可能な格納媒体上に具現化されたコンピュータプログラム製品、および/または、プロセッサ（プロセッサに接続されたメモリに格納および/またはそのメモリによって提供される命令を実行するよう構成されたプロセッサ）を含め、様々な形態で実装されうる。本明細書では、これらの実装または本発明が取りうる任意の他の形態を、技術と呼ぶ。一般に、開示された処理の工程の順序は、本発明の範囲内で変更されてもよい。特に言及しない限り、タスクを実行するよう構成されるものとして記載されたプロセッサまたはメモリなどの構成要素は、ある時間にタスクを実行するよう一時的に構成された一般的な構成要素として、または、タスクを実行するよう製造された特定の構成要素として実装されてよい。本明細書では、「プロセッサ」という用語は、1または複数のデバイス、回路、および/または、コンピュータプログラム命令などのデータを処理するよう構成された処理コアを指すものとする。

【 0 0 1 5 】

以下では、本発明の原理を示す図面を参照しつつ、本発明の1または複数の実施形態の

10

20

30

40

50

詳細な説明を行う。本発明は、かかる実施形態に関連して説明されているが、どの実施形態にも限定されない。本発明の範囲は、特許請求の範囲によってのみ限定されるものであり、多くの代替物、変形物、および、等価物を含む。以下の説明では、本発明の完全な理解を提供するために、多くの具体的な詳細事項が記載されている。これらの詳細事項は、例示を目的としたものであり、本発明は、これらの具体的な詳細事項の一部または全てがなくとも特許請求の範囲に従って実施可能である。簡単のために、本発明に関連する技術分野で周知の技術事項については、本発明が必要以上にわかりにくくならないように、詳細には説明していない。

【0016】

本願は、サーバがブラウザおよび非ブラウザクライアントアプリケーションに対して異なるウェブページデータを送り返すように、クライアントタイプごとのウェブページを生成するための技術を提供する。例えば、非ブラウザクライアントアプリケーションのためのウェブページデータは、写真のアップロード、マイクを介したチャットなど、オンボードハードウェアを必要とする特定の関数を含みうるが、ブラウザクライアントのためのウェブページデータは、かかる関数を除外しうるか、または、異なる関数を含みうる。非ブラウザクライアントアプリケーションは、ウェブページデータのためのコンテナと見なされてもよい。サーバは、表示されるウェブページおよびコンテンツを決定する。したがって、ウェブページ表示コンテンツの変更およびハードウェア関数の変更は共に、サーバ側で実行されうる。クライアントは、アップグレードされる必要がない。したがって、アップグレード処理は、大幅に簡略化される。ユーザは、ウェブページまたは非ブラウザクライアントアプリケーションにおける関数の変更を考慮することなしにウェブページにアクセスできる。操作は、実行が容易である。

【0017】

図1Aは、オンボードハードウェア関数を備えたウェブページをサポートする環境の一実施形態を示すブロック図である。クライアントデバイス101は、パーソナルコンピュータ、携帯端末（スマートフォンまたはタブレットなど）、もしくは、任意のその他の適切なデバイスであってよい。クライアントデバイスは、ネットワーク（例えば、インターネット）を介してサーバ103と通信する。

【0018】

ユーザは、スマートフォンまたはタブレットデバイスなどのクライアントデバイスを用いてウェブサイトへアクセスする時、ブラウザ（例えば、Safari（登録商標）、Chrome（登録商標）、Internet Explorer（登録商標）など）またはそのウェブサイト専用のクライアント（非ブラウザアプリケーションとも呼ばれる）などのクライアントアプリケーションで、ウェブサイトへのログオンまたはその他のアクセスを行いうる。クライアントアプリケーションは、ウェブサイトサーバとデータ交換する。いくつかの実施形態において、クライアントデバイスは、非ブラウザアプリケーションによってアクセスされうる特定のオンボードハードウェア（カメラまたはマイクなど）を備える。したがって、ユーザがウェブサイトと相互作用するさらなる方法を提供するために、特定のハードウェア関数が、非ブラウザアプリケーションに表示されるウェブページに追加される。

【0019】

図1Bは、ウェブページ情報を送信するための処理の一実施形態を示すフローチャートである。処理100は、サーバシステム（図1Aの103など）上で実行されてよい。

【0020】

工程102で、ウェブページ要求が受信される。

【0021】

ユーザは、ユーザのクライアント装置上で動作するブラウザまたは非ブラウザクライアントアプリケーションを用いて、ウェブサイトへのログインまたはその他のアクセスを最初に行う時、まず、ウェブページ要求を送信し、それは、サーバによって受信される。ウェブページ要求は、要求されたウェブページのアドレス、すなわち、要求されたウェブペ

ージのユニバーサルリソースロケータ (URL) を含む。後に詳述するように、ウェブページ要求は、クライアントタイプを示すマーキングを含む。

【0022】

工程104で、ウェブページ要求を送信したクライアントのクライアントタイプを識別するために用いられるマーキングパラメータが検出される。マーキングパラメータは、ウェブページ要求を送信したクライアントのタイプを評価するために用いられる。

【0023】

本明細書で用いられているように、クライアントタイプは、ブラウザアプリケーション (例えば、Safari (登録商標)、Google Chrome (登録商標)、Internet Explorer (登録商標) など) および非ブラウザクライアントアプリケーション (例えば、iOS (登録商標) 用のTaobaoショッピングアプリケーション、Android (登録商標) 用のTaobaoショッピングアプリケーションなど) を含む。いくつかの実施形態において、非ブラウザクライアントアプリケーションは、ハイブリッドアプリケーションであってもよい。いくつかの実施形態において、異なるクライアントタイプフィールドが、マーキングパラメータを用いて、異なるウェブページ要求内に設定される。いくつかの実施形態では、ブラウザ由来のウェブページ要求については、ウェブページ要求内にマーキングパラメータを設定する必要がない。非ブラウザクライアントアプリケーション由来のウェブページ要求については、ウェブページ要求のソースを識別するために、マーキングパラメータがウェブページ要求内に設定される。例えば、ブラウザを用いるユーザが、所望のURLとしてhttp://www.taobao.comを入力してよく、ブラウザによって送信される対応するウェブページ要求は、http://www.taobao.comというURLを含む。非ブラウザクライアントアプリケーションは、さらなるパラメータを含むことを除けば同一であるURLにアクセスするよう構成されている。非ブラウザクライアントアプリケーションによって送信されるウェブページ要求は、http://www.taobao.com/version=1&type=1というURLを含んでよく、ここで、「type」というパラメータおよびそれに対応する値の意味は、クライアントプログラムのプログラムによって規定される (例えば、「type=1」は、クライアントプログラムが非ブラウザプログラムであることを示す)。その他のパラメータおよび値が用いられてもよい。

【0024】

サーバは、ウェブページ要求を受信した後、ウェブページ要求のソース (すなわち、要求を行うクライアントアプリケーションのタイプ) を決定し、異なる要求ソースに対して異なるウェブページデータを送り返す。

【0025】

工程106で、サーバは、まず、ウェブページ要求がマーキングパラメータを含むか否かを検出する。ウェブページ要求がマーキングパラメータを含まない場合、そのウェブページ要求はブラウザから来たと判定され、制御は工程107に進む。ウェブページ要求がマーキングパラメータを含む場合、制御は工程108に進む。

【0026】

工程107で、ウェブページ要求はブラウザによって送信されたと見なされ；したがって、ブラウザベースのウェブページデータがクライアントに送信される。いくつかの実施形態において、ブラウザベースのウェブページデータは、ウェブページ情報 (すなわち、ブラウザに送り返されたウェブページによって表示されるコンテンツの関連データ) を含む。

【0027】

工程108で、ウェブページ要求は非ブラウザクライアントアプリケーションによって送信されたと見なされ；したがって、特定の非ブラウザクライアントアプリケーションに基づいた非ブラウザウェブページデータがクライアントに送信される。いくつかの実施形態において、非ブラウザウェブページデータは、ウェブページ情報と、1または複数のオンボードハードウェア関数を実行するためのハードウェア関数情報とを含む。

【 0 0 2 8 】

ウェブページ情報は、ブラウザまたは非ブラウザクライアントアプリケーションがインターネットウェブページをダウンロードして形成するためのウェブページデータ（HTMLコード、JavaScript（登録商標）コード、および/または、ウェブページのページレイアウト、フォント、色、テキスト情報、グラフィック情報などを指定するCSSコードなど）を含む。ハードウェア関数情報は、クライアントによる実行のために提供されたコードを含み、オンボードカメラを用いて写真または動画を撮る、オンボードマイクを用いて音声ベースのコメントを記録するなどのためのハードウェアAPIコールを呼び出すJavaScript（登録商標）コードなど、オンボードハードウェア関数を呼び出すために用いられる。したがって、特定のオンボードハードウェアを備えたクライアントデバイス上で動作する非ブラウザクライアントアプリケーションによって表示されるウェブページは、オンボードハードウェアを利用するさらなる関数を含むことができる。

10

【 0 0 2 9 】

図5は、ブラウザアプリケーションおよび非ブラウザクライアントアプリケーションによって表示されるウェブページの例を示すユーザインターフェースの図である。ユーザインターフェースの図500および502の両方は、異なるクライアントアプリケーションに表示された同じウェブサイトの同じ商品の商品フィードバックページを示す。500は、ブラウザによって表示されたウェブページの例である。502は、非ブラウザクライアントアプリケーションによって表示されたウェブページの例である。ページは、処理100などの処理を用いてそれぞれのクライアントアプリケーションにダウンロードされる。図に示すように、502は、オンボードハードウェアに依存する追加の関数、特に、それぞれ、ボタン504および506をユーザがクリックすることによって呼び出される「写真撮影」関数および「コメント記録」関数、を備える。

20

【 0 0 3 0 】

上記を要約すると、本願のサーバは、ウェブページ要求を受信した後、ウェブページ要求を送信したクライアントのタイプを特定するためのマーキングパラメータをウェブページ要求内で検出し、ウェブページ要求を送信したクライアントのタイプを評価する。クライアントタイプがブラウザであると判定された場合、サーバは、ウェブページ情報を含む第1のウェブページデータを送り返す。ウェブページ要求のクライアントタイプが非ブラウザクライアントアプリケーションであると判定された場合、サーバは、第2のウェブページデータを送り返す。第2のウェブページデータは、ウェブページ情報と、1または複数の指定されたオンボードハードウェア関数を実行するためのコード（例えば、ハードウェアAPIコールを呼び出すJavaScript（登録商標）コード）とを含む。いくつかの実施形態において、ウェブページ要求のクライアントタイプは、サーバ側から評価される。異なるクライアントタイプに対しては異なるデータ（ウェブページ情報およびハードウェア関数を含む）が、サーバによって送信される。したがって、非ブラウザクライアントアプリケーションは、サーバ送信データのコンテナと見なされる。したがって、ウェブページ表示コンテンツの変更およびハードウェア関数の変更は共に、サーバ側で実行されうる。非ブラウジングクライアントプログラムは、アップグレードされる必要がない。ユーザは、サーバへのログインまたはその他のアクセスを行うだけでよく、クライアントプログラムにさらなる変更を実行する必要はない。アップグレード処理が簡素化され、実行が容易になる。

30

40

【 0 0 3 1 】

いくつかの実施形態において、ウェブページ要求は、任意選択的にプラットフォーム情報を含む。様々な携帯電話が、様々なオペレーティングシステムを有する。例としては、Android（登録商標）、iOS（登録商標）、および、Windows Phone 7（WP7）（登録商標）が挙げられる。したがって、クライアントアプリケーションは、異なるオペレーティングシステムにインストールされうる。異なるオペレーティングシステムのメカニズムは異なっているので、それらがハードウェア関数を起動する方法も異なる。したがって、異なるオペレーティングシステムにインストールされた非ブラウ

50

ザクライアントアプリケーションは、ハードウェア関数をロードする時に、異なるデータを必要とする。例えば、異なるプラットフォーム上のハードウェア関数をサポートするには、異なるアプリケーションプログラミングインターフェース（API）情報が必要とされる。したがって、サーバは、非ブラウザクライアントアプリケーションがインストールされたオペレーティングシステムの種類を識別する必要がある。いくつかの実施形態において、識別処理を容易にするために、ウェブページ要求は、さらに、非ブラウザクライアントアプリケーションがインストールされたオペレーティングシステムを識別するために用いられるプラットフォーム情報を含む。

【0032】

いくつかの実施形態において、要求が非ブラウザクライアント由来であると判定された後、処理は、さらに、クライアントアプリケーションが動作するプラットフォーム環境（すなわち、クライアントプログラムの動作をサポートするオペレーティングシステム）を識別するためにプラットフォーム情報を検出する工程を備える。いくつかの実施形態において、さらなるパラメータが、クライアントアプリケーションによって構成され（例えば、ハードコードされる、構成ファイルまたはその他の場所から読み出される、など）、ウェブページ要求に含められる。例えば、iOSデバイスによって送信される要求は、`http://www.taobao.com/version=1&platform=iOS&type=1`というURLを含んでよく、Androidベースのクライアントプログラムによって送信される要求は、`http://www.taobao.com/version=1&platform=Android&type=2`というURLを含んでよい。

【0033】

いくつかの実施形態において、ウェブページ要求を送信したクライアントのタイプを識別するウェブページ要求内のマーキングパラメータは、ウェブページ要求のクライアントタイプがブラウザアプリケーションであるか非ブラウザクライアントアプリケーションであるかを判定するために用いられる。次に、サーバは、クライアントプログラムがインストールされたオペレーティングシステムのタイプを決定する。いくつかの実施形態において、上記の例に示したプラットフォーム情報パラメータは、ウェブページ要求のクライアントソースを区別するために用いられる。すなわち、プラットフォーム情報は、クライアントアプリケーションが動作するプラットフォーム（すなわち、オペレーティングシステム）を評価するために用いられ、オンボードハードウェア関数を実行するためにコードによって必要とされるプラットフォーム固有のサポート情報が、クライアントアプリケーションに送信される。

【0034】

様々な実施形態において、非ブラウザクライアントアプリケーションは、Android（登録商標）、iOS（登録商標）、WP7（登録商標）など、複数の異なるプラットフォームで実行される。

【0035】

プラットフォーム上で動作するクライアントがハードウェア関数をロードする時、そのプラットフォームのサポート情報（そのプラットフォームのAPI情報、および、写真の最大サイズ、記録の最大長、ハードウェアによってキャプチャされたデータをサーバにアップロードできるか否かなどのハードウェアの制限など）をサーバから送信される必要がある。サポート情報は、プラットフォームに固有であり、換言すると、異なるサポート情報が、異なるプラットフォームによって必要とされる。クライアントが呼び出すハードウェア関数をサポートしないプラットフォーム上でクライアントが動作している場合、さらなるサポート情報が送信される必要はない。

【0036】

任意選択的に、ウェブページ要求が、第1のプラットフォーム上で動作するクライアントプログラムに由来する場合、第2のウェブページデータは、さらに、ハードウェア関数をロードするために第1のプラットフォームによって必要とされるサポート情報を含む。

10

20

30

40

50

サポート情報は、ハードウェア関数をロードするためのサポートを提供するためのものである。具体的なサポート情報は、異なるプラットフォーム（すなわち、オペレーティングシステム）に従って様々である。

【0037】

いくつかの実施形態において、ハードウェア関数をサポートするプラットフォームは、プラットフォームデータを必要とする。例えば、非ブラウザクライアントアプリケーションがPhoneGapツールによって開発された場合、非ブラウザクライアントアプリケーションをAndroidプラットフォーム上で実行するために、PhoneGap JavaScriptライブラリが、ハードウェア関数のロード中にロードされる。Androidプラットフォームのためのandroid.jsまたはiOSプラットフォームのためのios.jsなど、ローカルプラットフォーム上でJavaScript（登録商標）をサポートするためのその他のJavaScript（登録商標）コードが、いくつかの実施形態において、必要とされダウンロードされてもよい。

10

【0038】

いくつかの実施形態において、サポート情報は、関数コール、パラメータ、戻り値など、ハードウェア関数に関するAPI情報を含む。したがって、クライアントプラットフォームは、サーバがサポート情報を送り返すべきか否か/サーバが送り返すべきサポート情報はどのタイプかという要求の仕様、および、クライアントによるオペレーティングシステムの実際の仕様によって区別できる。多くの異なるプラットフォームが可能であり、特定のプラットフォームを区別するための任意の適切な技術を利用できる。

20

【0039】

上記を要約すると、本願に記載のウェブページ要求は、さらに、サーバがクライアントプログラムの位置するプラットフォームを区別できるように、クライアントプログラムが動作するプラットフォームを識別する情報を含む。オンボードハードウェア関数を呼び出すコードによって必要とされるプラットフォーム固有のサポート情報が決定され、プラットフォームに送信される。したがって、サーバによって送り返されたウェブページデータは、様々な種類のオペレーティングシステムのニーズを満たすことができる。

【0040】

説明のために、以下の記載では、オンボードハードウェア関数の例としてカメラ関数および/またはマイク関数を用いる。ハードウェア関数情報のタイプは、クライアントデバイスで利用できる具体的なハードウェアに依存する。加速度計関数、バイオセンサ関数など、その他のハードウェア関数も利用できる。

30

【0041】

図2は、ウェブページを生成して表示するための処理の一実施形態を示すフローチャートである。処理200は、クライアントデバイス上で実行されてよい。

【0042】

工程202で、クライアントは、必要に応じてウェブページ要求を送信したクライアントのタイプを識別するマーキングパラメータをウェブページ要求に追加し、ウェブページ要求を送信する。

【0043】

ユーザがクライアントプログラムを用いてクライアントウェブサイトにログオンすると、クライアントは、ウェブページのロードを要求するためにウェブページ要求を送信する。サーバがウェブページ要求の送信元のクライアントのタイプを識別できるように、マーキングパラメータが、必要に応じてウェブページ要求に追加される。マーキングパラメータが存在すれば、クライアントは非ブラウザクライアントアプリケーションとして識別される。例えば、「platform」というマーキングパラメータが、ウェブページ要求に含まれ、クライアントプログラムの実装に応じて、マーキングパラメータの値は、「iOS」、「Android」、「WP7」と設定されうる。

40

【0044】

工程204で、ウェブページデータが受信される。ウェブページデータは、ウェブペー

50

ジ要求に応答して送り返される。

【0045】

クライアントは、ウェブページ要求を送信したクライアントプログラムを識別した後にサーバによって送り返されたクライアントプログラム固有のウェブページデータを受信する。クライアントタイプが非ブラウザクライアントアプリケーションであった場合、ウェブページデータは、ハードウェア関数情報およびウェブページ情報（すなわち、クライアントでウェブページによって表示されるコンテンツ）を含む。

【0046】

ウェブページ情報は、ウェブページをロードして形成するためにクライアントに供給されるウェブページデータ（例えば、ウェブページをフォーマットするHTML文書情報）を含む。ハードウェア関数情報は、クライアントによる実行のために提供されるコードを含んでおり、オンボードハードウェア関数（例えば、`hardware.useCamera()`）のようなJavaScript（登録商標）APIコールを呼び出すために用いられる。

【0047】

工程206で、1または複数のウェブページが、ウェブページデータに従って生成され、表示される。

【0048】

クライアントタイプが非ブラウザクライアントアプリケーションであった場合、クライアントプログラムは、ウェブページを表示するためのコンテナと見なされる。コンテナによって運ばれるデータは、サーバによって送信されたデータである。したがって、クライアントは、ウェブページデータを受信した後、ウェブページデータに従ってハードウェア関数を含むウェブページを生成できる。例えば、クライアントプラットフォームがAndroid（登録商標）であることをサーバが検出した場合、返されるウェブページデータは、返されるウェブページデータに以下のスクリプトを含めることによってAndroidのオンボードハードウェア関数をサポートするJavaScript（登録商標）ファイルを特に呼び出すデータを含む：`<script type="text/javascript" src="http://cdn.taobao.com/android/hardware.js">`。さらなるサポートデータ（写真の最大ピクセルサイズ、記録されるメッセージの最大長など）が、サーバから取得される。次いで、クライアントアプリケーションは、既存の技術を用いてウェブページを表示する。

【0049】

従来技術では、非ブラウザクライアント内のハードウェア関数はクライアントコードとして実装されるので、クライアントウェブページ内のハードウェア関数の変更は、最初にクライアントを更新することなしには実行できない。しかしながら、本願では、ハードウェア関数を実装するための関連データは、サーバによって提供される。したがって、ハードウェア関数の変更は、サーバ側で完了できる。アップグレードは、クライアントのアップグレードを必要としないので、アップグレード処理は、ユーザにとってトランスペアレントになる。

【0050】

上記を要約すると、本願において、クライアントは、ウェブページを要求する時に、ウェブページ要求のクライアントタイプを示すためにマーキングパラメータをウェブページ要求に追加する。クライアントタイプが非ブラウザクライアントアプリケーションであった場合、クライアントアプリケーションは、サーバによって送り返されたウェブページデータ（ハードウェア関数情報を含む）を受信し、オンボードハードウェア関数を含むウェブページを生成して表示する。本願では、クライアントプログラムは、ウェブページをロードしている時に、クライアントデバイスのプラットフォームによって提供された既存のAPIを呼び出すことにより、サーバによって送り返されたウェブページデータだけに基づいてハードウェア関数のロードを実行できる。クライアントプログラムは、ウェブページを表示するためのコンテナと見なされてよく、クライアントプログラムを動作させるこ

10

20

30

40

50

とが非常に容易になる。ウェブページデータおよびハードウェア関数の変更には、クライアントアプリケーションのアップグレードが必要ない。

【 0 0 5 1 】

いくつかの実施形態において、ウェブページ要求を送信する前に、クライアントプログラムは、さらに、クライアントプログラムが動作するプラットフォームに関する情報を取得する。ここで、プラットフォーム情報は、クライアントプログラムが動作するプラットフォームを評価するために用いられ、プラットフォームは、プラットフォーム上のクライアントプログラムがオンボードハードウェア関数を実行するためのコードによって必要とされるサポート情報を取得するための基礎となる。

【 0 0 5 2 】

さらに、プラットフォーム情報は、ウェブページ要求に追加される。プラットフォーム情報は、クライアントプログラムが動作するオペレーティングシステムに関する情報である。

【 0 0 5 3 】

クライアントプログラムは様々なオペレーティングシステムにインストール可能であり、異なるオペレーティングシステムはハードウェア関数をロードする時に異なるデータを必要とするので、クライアントプログラムは、送信するウェブページ要求に対応するプラットフォーム情報を取得する必要もある。次いで、プラットフォーム情報をウェブページ要求に追加する。プラットフォーム情報は、クライアントプログラムがロードされたオペレーティングシステムを識別するよう機能する。

【 0 0 5 4 】

プラットフォーム情報のクライアントソースがオンボードハードウェアの呼び出しをサポートするプラットフォームであった場合、ウェブページデータは、さらに、プラットフォームサポート情報を含む。例えば、Androidシステムがハードウェア関数をロードする間に、JSライブラリがロードされる必要があり、JSライブラリのロードは、関連コードのサポートを必要とする。したがって、ウェブページデータは、さらに、クライアントプラットフォームのサポート情報を含む。サポート情報は、ハードウェア関数のロードをサポートするためのものであり、サポート情報の具体的なタイプは、異なるプラットフォーム（すなわち、オペレーティングシステム）に従って様々である。

【 0 0 5 5 】

いくつかの実施形態において、オンボードハードウェア関数を含むウェブページを生成するための基礎としてウェブページデータを用いることは、ページレンダリングエンジン（Trident、Genko、Webkit、Prestoなど）を呼び出してウェブページ情報をレンダリングし、オンボードハードウェア関数を実行するためのコードをレンダリング処理に追加することを含む。

【 0 0 5 6 】

ウェブページを生成する処理は、ウェブページレンダリングの処理でもある。例えば、ウェブページ情報は、ウェブページHTMLコードであってよく、その場合、ウェブページレンダリングは、上から下までのHTMLコードの実行であってよい。画像、FLASH、および、その他のかかるデータは、ウェブページに必要な時にサーバから取得される。レンダリング処理では、コード実行が、ハードウェアを必要とする関数に到達すると、オンボードハードウェア関数を実行するためのコードが追加され、次いで、そのコードが実行される。例えば、カメラ関数を追加するために、hardware.useCamera()という関数コールが追加される。次いで、HTMLコードの実行が継続してHTMLコードの実行が終了し、その時点で、オンボードハードウェア関数を含むウェブページが生成される。

【 0 0 5 7 】

上記の処理において、追加されるハードウェア関数の実行は、以下の2つのタイプを含む：1) 即時実行 - - 例えば、連絡先情報のロードでは、ハードウェア関数情報（ハードウェア関数コードなど）を直接ロードできる。2) 非即時実行 - - 例えば、ハードウェア

10

20

30

40

50

関数がボタンによってトリガされる。この場合、ハードウェアインターフェースが、ハードウェア関数情報に接続するようにウェブページ内に構成されうる。ウェブページが表示され、ユーザがボタンをクリックすると、インターフェースをトリガすることができ、そして、ハードウェア関数情報が実行される。

【0058】

上記の処理において、本願のクライアントは、PhoneGapを用いて開発できる。HTML、CSS、および、JavaScript（登録商標）で構築されたPhoneGapは、モバイルクロスプラットフォームのモバイルアプリケーションを迅速に開発するためのプラットフォームである。それにより、開発者が、iPhone、Android、Palm、Symbian、WP7、Bada、および、Blackberryなどのスマートフォンの主要機能（地理的測位、加速度計、連絡先、音声、および、振動など）を利用することが可能になる。さらに、PhoneGapは、豊富なプラグインを有しているため、機能数の拡大が可能である。

10

【0059】

PhoneGapは、或る種のクライアント開発ミドルウェアを提供し、ミドルウェアは、フロントエンド開発技術を用いてオンボードハードウェア関数を呼び出すインターフェースを提供する。PhoneGap内のSDK（ソフトウェア開発キット：Software Development Kit）によって提供されたWebViewコントロールは、WebKit（オープンソースブラウザエンジン）に基づいており；コアはウェブページのレンダリングおよび表示を実現できる。

20

【0060】

いくつかの実施形態において、ウェブページデータがさらにサポート情報を含む場合、ウェブページデータに従って生成されたハードウェア関数を含むウェブページは、ウェブページ情報をレンダリングするためのページレンダリングエンジン呼び出し、ロケーションハードウェア関数を実行するためのコードおよびサポート情報をレンダリングに追加して、オンボードハードウェア関数を含むウェブページを生成することを含む。

【0061】

ウェブページを生成する処理は、ウェブページレンダリングの処理でもある。例えば、ウェブページ情報は、ウェブページHTMLコードであってよく、その場合、ウェブページレンダリングは、上から下までのHTMLコードの実行であってよい。画像、FLASH、および、その他のかかるデータは、ウェブページに必要な時にサーバから取得される。レンダリング処理では、コード実行が、ハードウェアを必要とする関数に到達すると、ハードウェア関数コードは、ハードウェア関数ロードのために実行中にサポート情報を必要とする。例えば、Androidオペレーティングシステムでは、JSライブラリおよびその他のかかる情報が、ハードウェアロード中に呼び出される必要がある。したがって、サーバによって送り返されたサポート情報が必要である。適切なJSライブラリが、ハードウェア関数情報のロードを完了させるために、ハードウェア関数ロード中に呼び出される。次いで、HTMLコードの実行が継続してHTMLコードの実行が終了し、その時点でオンボードハードウェア関数を含むウェブページが生成されうる。

30

【0062】

実際の処理では、PhoneGapツールは、異なるオペレーティングシステムに対して異なるクライアントプログラムを構成する。異なるクライアントプログラムがハードウェア関数を呼び出す場合、利用する技術は様々である。したがって、必要とされるデータも様々である。例えば、異なるサポート情報および異なるインターフェース構成が必要とされる。したがって、サーバ側が、マーキングパラメータおよびプラットフォーム情報を用いて検出を実行した後に、検出結果に従って異なるウェブページデータを送り返すことが好ましい。

40

【0063】

カメラ関数情報は、ウェブページ内の写真のロード、スキャン、カメラフラッシュ、および、その他のかかるハードウェア関数などのカメラ関連関数を生成できる。マイク関数

50

情報は、記録およびその他のかかる音声収集ハードウェア関数などのマイク関連関数を生成できる。

【 0 0 6 4 】

オンボードハードウェア関数を含むウェブページ内のハードウェア関数は、ボタン、リンク、または、イベントトリガをサポートするその他のインターフェースデバイスを通して実現されることが好ましい。多くのその他の実装も可能である。

【 0 0 6 5 】

いくつかの実施形態において、ウェブページ内のオンボードハードウェア関数は、ボタンを用いて実行される。すなわち、ユーザがウェブページ上のボタンをクリックすると、そのイベントが適切なハードウェアの起動をトリガする。一例として図 5 を参照すると、ボタン 5 0 4 (「写真撮影」) がクリックされると、カメラが起動され、カメラを用いて写真を撮影するための A P I が呼び出される。

10

【 0 0 6 6 】

いくつかの実施形態において、オンボードハードウェア関数は、リンクを用いて実行される。ボタンと同様に、リンクアドレスがユーザによって選択されると、リンクを含むウェブページが、適切なハードウェアを起動し、適切なハードウェア関数を呼び出す。

【 0 0 6 7 】

いくつかの実施形態において、オンボードハードウェア関数は、イベントトリガを用いて実施される。例えば、ウェブページ内のアプリケーションが、音声によって実行される。このアプリケーションが選択されると、マイクがオンになるようにトリガされる。

20

【 0 0 6 8 】

図 3 は、ウェブページ情報を送信するためのデバイスの一実施形態を示す構造図である。デバイス 3 0 0 は、サーバを用いて実装されてよく、受信モジュール 1 1、検出モジュール 1 2、第 1 の返送モジュール 1 3、および、第 2 の返送モジュール 1 4 を備える。受信モジュール 1 1 は、ウェブページ要求を受信するよう構成されている。検出モジュール 1 2 は、ウェブページ要求を送信したクライアントのタイプを識別するマーキングパラメータをウェブページ要求内で検出し、ウェブページ要求を送信したクライアントのタイプを評価するよう構成されている。第 1 の返送モジュール 1 3 は、ウェブページ要求のクライアントタイプがブラウザであるとの判定に基づいて、ブラウザに基づく第 1 のウェブページデータを送り返すよう構成されており、第 1 のウェブページデータは、ウェブページ情報を含む。第 2 の返送モジュール 1 4 は、ウェブページ要求のクライアントタイプが非ブラウザクライアントアプリケーションであるとの判定に基づいて、クライアントプログラムに基づく第 2 のページデータを送り返すよう構成されており、第 2 のページデータは、指定されたオンボードハードウェア関数を実行するためのコードおよびウェブページ情報を含む。

30

【 0 0 6 9 】

本願では、ウェブページ要求を送信したクライアントのタイプが非ブラウザクライアントアプリケーションであった場合、ウェブページ要求は、さらに、クライアントプログラムが動作するプラットフォームを識別する情報を含む。第 1 の返送モジュール 1 3 は、さらに、クライアントプログラムが動作するプラットフォームを評価し、プラットフォーム上のクライアントプログラムがオンボードハードウェア関数を実行するためのコードによって必要とされるサポート情報を送り返すための基礎として、プラットフォーム情報を用いるよう構成される。

40

【 0 0 7 0 】

図 4 は、本願の一実施形態に記載のウェブページを生成するためのデバイスの一実施形態を示す構造図である。

【 0 0 7 1 】

デバイス 4 0 0 は、非ブラウザクライアントアプリケーションを実行するクライアントデバイスとして実装されてよい。デバイス 4 0 0 は、処理 2 0 0 を実施できる。デバイス 4 0 0 は、送信モジュール 2 1、受信モジュール 2 2、および、生成・表示モジュール 2

50

3を備える。送信モジュール21は、ウェブページ要求を送信するクライアントのタイプを識別するマーキングパラメータをウェブページ要求に追加してウェブページ要求を送信するよう構成されており、クライアントのタイプは非ブラウザクライアントアプリケーションである。受信モジュール22は、ウェブページ要求およびマーキングパラメータに従って送り返されたウェブページデータを受信するよう構成されており、ウェブページデータは、指定されたオンボードハードウェア関数を実行するためのコードおよびウェブページ情報を含む。生成・表示モジュール23は、オンボードハードウェア関数を含むウェブページを生成してウェブページを表示するための基礎としてウェブページデータを用いるよう構成されている。

【0072】

10

図の例において、送信モジュール21は、さらに、クライアントプログラムが動作するプラットフォームに関する情報を取得するよう構成されており、プラットフォーム情報は、クライアントプログラムが動作するプラットフォームを評価するため、および、プラットフォーム上のクライアントプログラムがオンボードハードウェア関数を実行するためのコードによって必要とされるサポート情報を取得するために利用できる。

【0073】

図の例において、生成・表示モジュール23は、ページレンダリングエンジン呼び出して、ウェブページ情報をレンダリングし、ハードウェア関数を含むウェブページを生成するためにオンボードハードウェア関数情報を実行するためのコードをレンダリング処理に追加するよう構成されている。

20

【0074】

図の例において、オンボードハードウェア関数は、カメラ関数、マイク関数、および/または、任意のその他の適切なハードウェア関数を含む。

【0075】

図の例において、オンボードハードウェア関数を含むウェブページデータ内のハードウェア関数は、ボタン、リンク、または、イベントトリガを通して実現される。

【0076】

本記載に含まれる実施形態の各々は漸進的に記載されており、各実施形態の説明は他の実施形態とは異なる領域に焦点を当てており、実施形態の記載は各実施形態の同一または類似の部分について相互に参照できる。

30

【0077】

上述のモジュールは、1または複数のプロセッサ上で実行されるソフトウェアコンポーネントとして、特定の機能を実行するよう設計されたプログラム可能論理デバイスおよび/または特定用途向け集積回路などのハードウェアとして、もしくは、それらの組み合わせとして実装することができる。いくつかの実施形態において、モジュールは、コンピュータデバイス（パーソナルコンピュータ、サーバ、ネットワーク装置など）に本願の実施形態に記載された方法を実行させるための複数の命令など、不揮発性記憶媒体（光学ディスク、フラッシュ記憶装置、携帯用ハードディスクなど）に格納することができるソフトウェア製品の形態で具現化されてよい。モジュールは、単一のデバイス上に実装されてもよいし、複数のデバイスにわたって分散されてもよい。モジュールの機能は、互いに統合されてもよいし、複数のサブモジュールにさらに分割されてもよい。

40

【0078】

本発明は、本発明の方法、装置（システム）、および、コンピュータプログラム製品に基づいたフローチャートおよび/またはブロック図を参照して記載されている。フローチャートおよび/またはブロック図内の各処理および/またはブロック、ならびに、フローチャートおよび/またはブロック図内の処理および/またはブロックの組み合わせは、コンピュータプログラム命令で実現されうることを理解されたい。これらのコンピュータ命令は、マシンを生成するために、汎用コンピュータ、専用コンピュータ、組み込みプロセッサ、または、その他のプログラム可能なデータ処理装置のプロセッサに供給されてよく、その結果、コンピュータまたはその他のプログラム可能なデータ処理装置のプロセッサ

50

によって命令が実行されることで、フローチャートの１または複数の処理および／またはブロック図の１または複数のブロックに記載された機能を実現するために用いられるデバイスが生成される。

【 0 0 7 9 】

これらのコンピュータプログラム命令は、コンピュータまたはその他のプログラム可能なデータ処理装置を導くことができる専用のコンピュータ読み取り可能な記憶デバイス上に格納されてもよく、その結果、これらのコンピュータ読み取り可能なデバイス上に命令が格納されることで、命令デバイスを備える製品が生成される。これらの命令デバイスは、フローチャートの１または複数の処理および／またはブロック図の１または複数のブロックに記載された機能を実現する。

10

【 0 0 8 0 】

これらのコンピュータプログラム命令は、コンピュータまたはその他のプログラム可能なデータ処理装置上にロードされてもよく、その結果、一連の動作工程が、コンピュータまたはその他のプログラム可能な装置上で実行され、コンピュータ処理が行われる。このように、コンピュータまたはその他のプログラム可能な装置上で実行された命令は、フローチャートの１または複数の処理および／またはブロック図の１または複数のブロックに記載された機能を実現するための工程を提供する。

【 0 0 8 1 】

本発明によって提供される、ウェブページを送り返すための方法およびデバイスならびにウェブページを生成するための方法およびデバイスについて詳述した。本明細書は、具体的な実施形態を用いて、本発明の実施例の原理および形態を説明している。上記の実施形態の説明は、単に、本発明の方法およびその中心概念の理解を助けるよう意図されたものである。さらに、当業者は、本発明の概念に基づいて、具体的な実装の形態および応用範囲に対して修正を加えることができる。要するに、本記載の内容は、本発明を限定するものとして理解されるべきではない。

20

【 0 0 8 2 】

上述の実施形態は、理解しやすいようにいくぶん詳しく説明されているが、本発明は、提供された詳細事項に限定されるものではない。本発明を実施する多くの代替方法が存在する。開示された実施形態は、例示であり、限定を意図するものではない。

30

本発明は、たとえば、以下のような態様で実現することもできる。

適用例 1 :

クライアントデバイス上で動作するクライアントアプリケーションにウェブページデータを送信するためのシステムであって、

１または複数のプロセッサであって、

前記クライアントアプリケーションによって送信されたウェブページ要求を受信し、

前記ウェブページ要求内のマーキングパラメータを検出し、

前記マーキングパラメータに少なくとも部分的に基づいて、前記クライアントアプリケーションのクライアントタイプを判定し、

40

前記クライアントタイプがブラウザクライアントアプリケーションであると判定された場合に、ウェブページ情報を含む第１のウェブページデータを前記クライアントアプリケーションに送信し、

前記クライアントタイプが非ブラウザクライアントアプリケーションであると判定された場合に、ウェブページ情報と、前記クライアントデバイス上でオンボードハードウェア関数を実行するためのコードと、を含む第２のウェブページデータを前記クライアントアプリケーションに送信するよう構成された、１または複数のプロセッサと、

前記１または複数のプロセッサに接続され、前記１または複数のプロセッサに命令を提供するよう構成された１または複数のメモリと、
を備える、システム。

50

適用例 2 :

適用例 1 のシステムであって、前記クライアントタイプが非ブラウザクライアントアプリケーションであると判定された場合に、前記 1 または複数のプロセッサは、さらに、前記ウェブページ要求に含まれるプラットフォーム情報を用いて、前記クライアントアプリケーションが動作するプラットフォームを識別するよう構成されている、システム。

適用例 3 :

適用例 2 のシステムであって、前記 1 または複数のプロセッサは、さらに、前記プラットフォーム上で前記オンボードハードウェア関数を実行するために必要なサポート情報を送信するよう構成されている、システム。

10

適用例 4 :

適用例 3 のシステムであって、前記サポート情報はプラットフォーム固有である、システム。

適用例 5 :

適用例 1 のシステムであって、前記オンボードハードウェア関数は、カメラ関数、マイク関数、または、それら両方を含む、システム。

20

適用例 6 :

適用例 3 のシステムであって、前記サポート情報は、前記オンボードハードウェア関数に関する A P I 情報を含む、システム。

適用例 7 :

クライアントアプリケーションにウェブページデータを送信するための方法であって、前記クライアントアプリケーションによって送信されたウェブページ要求を受信する工程と、

前記ウェブページ要求内のマーキングパラメータを検出する工程と、

前記マーキングパラメータに少なくとも部分的に基づいて、前記クライアントアプリケーションのクライアントタイプを判定する工程と、

30

前記クライアントタイプがブラウザクライアントアプリケーションであると判定された場合に、ウェブページ情報を含む第 1 のウェブページデータを前記クライアントアプリケーションに送信する工程と、

前記クライアントタイプが非ブラウザクライアントアプリケーションであると判定された場合に、ウェブページ情報と、オンボードハードウェア関数を実行するためのコードと、を含む第 2 のウェブページデータを前記クライアントアプリケーションに送信する工程と、

を備える、方法。

40

適用例 8 :

表示するためのウェブページを生成するためのシステムであって、

1 または複数のプロセッサであって、

前記システム上で動作するクライアントアプリケーションのクライアントタイプを識別するマーキングパラメータを含むウェブページ要求を生成し、

前記クライアントアプリケーションの前記クライアントタイプは非ブラウザクライアントアプリケーションであり、

前記クライアントアプリケーションを用いて、前記ウェブページ要求を送信し、

前記ウェブページ要求に応答して送信されたウェブページデータを受信し、

前記ウェブページデータは、ウェブページ情報と、前記システム上で指定のオンボ

50

ードハードウェア関数を実行するために用いられるコードとを含み、

前記ウェブページデータを用いて、前記オンボードハードウェア関数を提供すると共に表示される前記ウェブページを生成するよう構成された、1または複数のプロセッサと、

前記1または複数のプロセッサに接続され、前記1または複数のプロセッサに命令を提供するよう構成された1または複数のメモリと、
を備える、システム。

適用例9：

適用例8のシステムであって、前記1または複数のプロセッサは、さらに、
前記クライアントアプリケーションが動作するプラットフォームを識別するプラットフォーム情報を取得し、

前記プラットフォーム情報に少なくとも部分的に基づいて、前記システム上で前記指定のオンボードハードウェア関数を実行するために用いられる前記コードによって必要とされるサポート情報を取得するよう構成されている、システム。

適用例10：

適用例8のシステムであって、前記ウェブページデータを用いて、前記ウェブページを生成することは、

ページレンダリングエンジン呼び出して、前記ウェブページ情報をレンダリングし、
前記オンボードハードウェア関数を実行するために用いられる前記コードを、前記レンダリングされたウェブページ情報に追加することを含む、システム。

適用例11：

適用例10のシステムであって、前記ウェブページデータを用いて、前記ウェブページを生成することは、

前記オンボードハードウェア関数を実行するためのサポート情報を、前記レンダリングされたウェブページ情報に追加することを含む、システム。

適用例12：

適用例8のシステムであって、前記オンボードハードウェア関数は、カメラ関数、マイク関数、または、それら両方を含む、システム。

適用例13：

適用例8のシステムであって、前記オンボードハードウェア関数は、前記ウェブページに関連付けられたボタン、リンク、またはイベントトリガを通して呼び出される、システム。

適用例14：

適用例8のシステムであって、前記サポート情報は、前記オンボードハードウェア関数に関するAPI情報を含む、システム。

適用例15：

表示するためのウェブページを生成するための方法であって、
クライアントデバイス上で動作するクライアントアプリケーションのクライアントタイプを識別するマーキングパラメータを含むウェブページ要求を生成する工程であって、前記クライアントアプリケーションの前記クライアントタイプは非ブラウザクライアントアプリケーションである、工程と、

前記クライアントアプリケーションを用いて、前記ウェブページ要求を送信する工程と

、

10

20

30

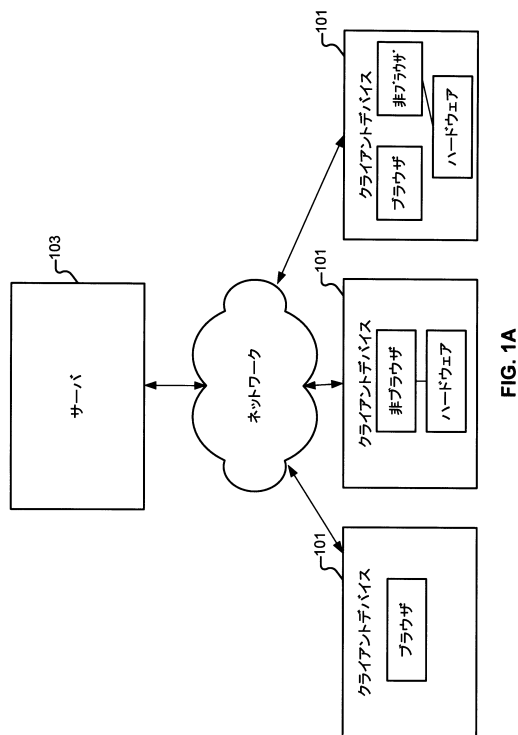
40

50

前記ウェブページ要求に応答して送信されたウェブページデータを受信する工程であって、前記ウェブページデータは、ウェブページ情報と、指定のオンボードハードウェア関数を実行するために用いられるコードとを含む、工程と、

前記ウェブページデータを用いて、前記クライアントデバイス上で前記オンボードハードウェア関数を提供すると共に表示される前記ウェブページを生成する工程と、を備える、方法。

【図 1 A】



【図 1 B】

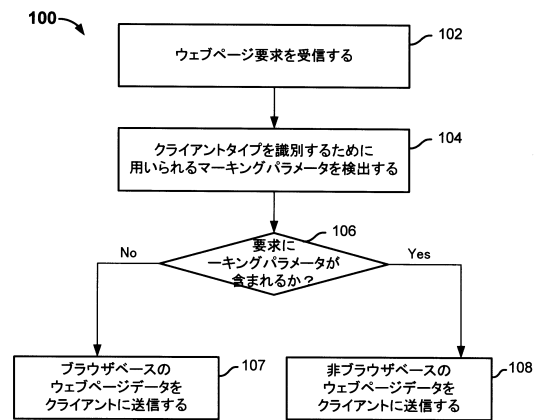


FIG. 1B

【図 2】

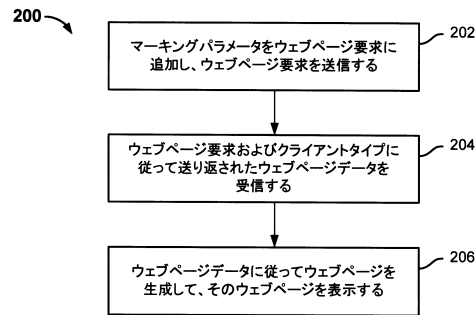


FIG. 2

【図 4】

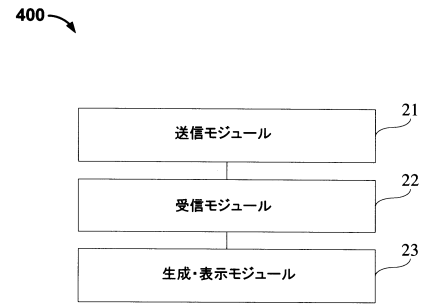


FIG. 4

【図 3】

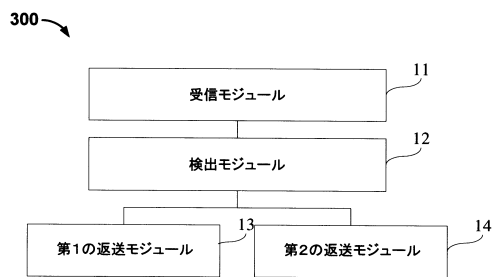


FIG. 3

【図 5】

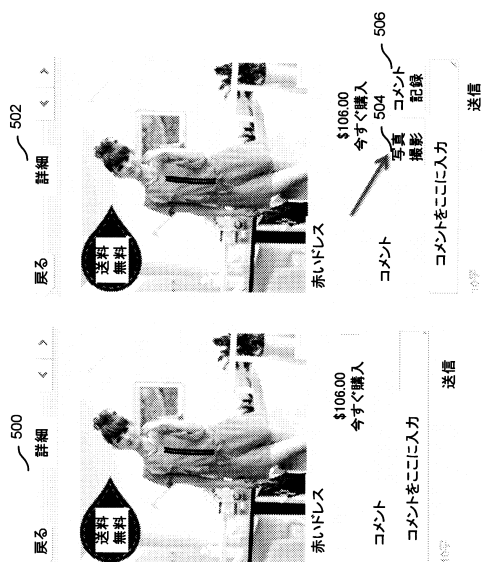


FIG. 5

フロントページの続き

(72)発明者 チィー・ズーシェン

中華人民共和国 ハンチョウ 3 1 1 1 2 1 , ユ・ハン・ディストリクト , ウェスト・ウェン・イー
・ロード , ビルディング 3 , 5 階 , ナンバー 9 6 9 , アリババ・グループ・リーガル・デパート
メント内

審査官 小林 義晴

(56)参考文献 米国特許出願公開第 2 0 1 2 / 0 2 2 0 2 6 3 (U S , A 1)

(58)調査した分野(Int.Cl. , D B 名)

G 0 6 F 1 3 / 0 0