

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5830088号
(P5830088)

(45) 発行日 平成27年12月9日(2015. 12. 9)

(24) 登録日 平成27年10月30日(2015. 10. 30)

(51) Int.Cl.	F I
G 0 6 F 13/00 (2006.01)	G 0 6 F 13/00 5 1 0 A
G 0 6 F 15/00 (2006.01)	G 0 6 F 15/00 4 1 0 Z

請求項の数 15 (全 63 頁)

(21) 出願番号	特願2013-514213 (P2013-514213)	(73) 特許権者	314015767
(86) (22) 出願日	平成23年5月31日(2011. 5. 31)		マイクロソフト テクノロジー ライセン
(65) 公表番号	特表2013-536482 (P2013-536482A)		シング, エルエルシー
(43) 公表日	平成25年9月19日(2013. 9. 19)		アメリカ合衆国 ワシントン州 9805
(86) 国際出願番号	PCT/US2011/038511		2 レッドモンド ワン マイクロソフト
(87) 国際公開番号	W02011/156172		ウェイ
(87) 国際公開日	平成23年12月15日(2011. 12. 15)	(74) 代理人	100140109
審査請求日	平成26年4月25日(2014. 4. 25)		弁理士 小野 新次郎
(31) 優先権主張番号	12/814, 359	(74) 代理人	100075270
(32) 優先日	平成22年6月11日(2010. 6. 11)		弁理士 小林 泰
(33) 優先権主張国	米国 (US)	(74) 代理人	100101373
			弁理士 竹内 茂雄
		(74) 代理人	100118902
			弁理士 山本 修

最終頁に続く

(54) 【発明の名称】 ウェブアプリケーションホームボタン

(57) 【特許請求の範囲】

【請求項 1】

クライアントデバイス上にインストールされるウェブアプリケーションに関連付けられたウェブサイトのスタートURLへウェブアプリケーションモードブラウザをナビゲートするステップと、

前記スタートURLに関連付けられたウェブページ以外のウェブページへ前記ウェブアプリケーションモードブラウザをナビゲートするステップと、

ウェブアプリケーションホームボタンの選択を受け取るステップと、

前記ウェブアプリケーションホームボタンの選択を受け取ったことに応答して、前記ウェブアプリケーションモードブラウザを前記ウェブサイトのスタートURLへナビゲートするステップを含み、

前記ウェブアプリケーションと前記ウェブアプリケーションホームボタンは、前記クライアントデバイスのデスクトップ又はタスクバーにファビコンをドラッグアンドドロップすることによってインストールされたものであり、

前記ウェブサイトのスタートURLは、ドラッグアンドドロップされるファビコンの元となるページから得られること、

を含む、コンピューターによって実施される方法。

【請求項 2】

前記ウェブアプリケーションホームボタンが、ブランド設定を可能にするように構成される、請求項 1 に記載の方法。

【請求項 3】

前記ウェブアプリケーションホームボタンが、原色を使用したブランド設定を可能にするように構成される、請求項 1 に記載の方法。

【請求項 4】

前記ウェブアプリケーションホームボタンが、前記ウェブアプリケーションモードブラウザに関連付けられたナビゲーションボタンの隣に表示されるように構成される、請求項 1 に記載の方法。

【請求項 5】

前記ウェブアプリケーションホームボタンが、前記ウェブアプリケーションモードブラウザに関連付けられたナビゲーションボタンの隣に表示されるように構成され、前記ナビゲーションボタンが、前記ウェブサイトに関連付けられた原色を利用する、請求項 1 に記載の方法。

10

【請求項 6】

前記ウェブアプリケーションホームボタンが、前記ウェブアプリケーションモードブラウザに関連付けられた戻るおよび進むナビゲーションボタンの隣に表示されるように構成される、請求項 1 に記載の方法。

【請求項 7】

前記ウェブサイトのスタート URL が、HTML から確認される、請求項 1 に記載の方法。

【請求項 8】

20

コンピューター実行可能命令を格納するコンピューティングデバイスであって、前記コンピューター実行可能命令が、実行されたときに、

クライアントデバイス上にインストールされるウェブアプリケーションに関連付けられたウェブサイトのスタート URL へウェブアプリケーションモードブラウザをナビゲートするステップと、

前記スタート URL に関連付けられたウェブページ以外のウェブページへ前記ウェブアプリケーションモードブラウザをナビゲートするステップと、

ウェブアプリケーションホームボタンの選択を受け取るステップと、

前記ウェブアプリケーションホームボタンの選択を受け取ったことに応答して、前記ウェブアプリケーションモードブラウザを前記ウェブサイトのスタート URL へナビゲートするステップを含み、

30

前記ウェブアプリケーションと前記ウェブアプリケーションホームボタンは、前記クライアントデバイスのデスクトップ又はタスクバーにファビコンをドラッグアンドドロップすることによってインストールされたものであり、

前記ウェブサイトのスタート URL は、ドラッグアンドドロップされるファビコンの元となるページから得られること、

を含む動作を実行する、コンピューティングデバイス。

【請求項 9】

前記ウェブアプリケーションホームボタンが、ブランド設定を可能にするように構成される、請求項 8 に記載のコンピューティングデバイス。

40

【請求項 10】

前記ウェブアプリケーションホームボタンが、原色を使用したブランド設定を可能にするように構成される、請求項 8 に記載のコンピューティングデバイス。

【請求項 11】

前記ウェブアプリケーションホームボタンが、前記ウェブアプリケーションモードブラウザに関連付けられたナビゲーションボタンの隣に表示されるように構成される、請求項 8 に記載のコンピューティングデバイス。

【請求項 12】

前記ウェブアプリケーションホームボタンが、前記ウェブアプリケーションモードブラウザに関連付けられたナビゲーションボタンの隣に表示されるように構成され、前記ナ

50

ビゲーションボタンが、前記ウェブサイトに関連付けられた原色を利用する、請求項 8 に記載のコンピューティングデバイス。

【請求項 1 3】

前記ウェブアプリケーションホームボタンが、前記ウェブアプリケーションモードブラウザに関連付けられた戻るおよび進むナビゲーションボタンの隣に表示されるように構成される、請求項 8 に記載のコンピューティングデバイス。

【請求項 1 4】

前記ウェブアプリケーションホームボタンが、前記ウェブアプリケーションモードブラウザに関連付けられた戻るおよび進むナビゲーションボタンの隣に表示されるように構成され、前記戻るおよび進むナビゲーションボタンが、前記ウェブサイトに関連付けられた原色を利用する、請求項 8 に記載のコンピューティングデバイス。

10

【請求項 1 5】

コンピューター実行可能命令を具体化する 1 つまたは複数のコンピューター可読記憶メモリーデバイスであって、前記コンピューター実行可能命令が、実行されたときに、

1 つまたは複数のナビゲーションボタンと、

クライアントによってインストール可能なウェブアプリケーションに関連付けられるウェブアプリケーションホームボタンであって、前記ウェブアプリケーションに関連付けられたウェブサイトに関連付けられたスタート URL へのナビゲーションを可能にする、ウェブアプリケーションホームボタンと

を含むウェブアプリケーションモードブラウザを実施することを含み、

20

前記ウェブアプリケーションと前記ウェブアプリケーションホームボタンは、前記クライアントデバイスのデスクトップ又はタスクバーにファビコンをドラッグアンドドロップすることによってインストールされたものであり、

前記ウェブサイトのスタート URL は、ドラッグアンドドロップされるファビコンの元となるページから得られる、

1 つまたは複数のコンピューター可読記憶メモリーデバイス。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ウェブアプリケーションホームボタンに関する。

30

【背景技術】

【0002】

[0001]より最近では、業界は、ウェブアプリケーションまたはウェブサイトをユーザーのコンピューターデスクトップ環境すなわち「デスクトップ」と統合するという概念に焦点を合わせ始めている。しかし、そうすることに関連した難題がある。たとえば、今日では、ウェブサイトがデスクトップ統合の目的で自らのサイトの境界を定義することは困難である。したがってエンドユーザーは、自分たち自身が開発するクライアント側スクリプトを通じて境界を定義することを委ねられている。これは、問題となる可能性がある。なぜなら、エンドユーザーは、ある特定のウェブサイトがどのように構築されているかを必ずしも知っているとは限らない可能性があるためである。たとえばエンドユーザーは、リンク、ウェブプロパティどうしの間における関係、または機能をユーザーに提供するためにウェブサイトによって採用されているその他のニュアンスのすべてを必ずしも知っているとは限らない可能性がある。したがってエンドユーザーのスクリプトは、これらのリンクまたはニュアンスを理解しない可能性があり、それゆえに、望ましくないまたは破綻したユーザー経験につながる可能性がある。

40

【0003】

[0002]さらに、ユーザーは今日、デュアルブート問題として知られている問題に直面している。具体的には、ユーザーは、自分たちのパーソナルコンピューターをブートし、自らのブラウザを始動し、最後に、自分たちが内部で作業したいと望む特定のウェブアプリケーションを起動することを余儀なくされている。この問題は、ブラウザのクロム内

50

に現れて、ウェブアプリケーションに関連した手元の特定のタスクにユーザーが単に集中することを不可能にする、気を散らすものなど、ユーザーにとってあまりにも多くの気を散らすものをブラウザが提供することがあるという事実によって、複雑にされている。

【発明の概要】

【発明が解決しようとする課題】

【0004】

本発明は、ウェブアプリケーションホームボタンを提供する。

【課題を解決するための手段】

【0005】

[0003]この「発明の概要」は、以降の「発明を実施するための形態」においてさらに説明されるコンセプトから抜粋したものを、簡略化された形式で紹介するために提供される。この「発明の概要」は、特許請求される主題の鍵となる特徴または必要不可欠な特徴を特定することを意図されておらず、特許請求される主題の範囲を限定するために使用されることも意図されていない。

【0006】

[0004]さまざまな実施形態が、エンドユーザーがウェブアプリケーションおよびウェブサイトをクライアントデバイスデスクトップなどのクライアントデバイス上にインストールできるようにするメカニズムを提供する。1つまたは複数の実施形態によれば、ウェブサイトに関連付けられた開発者が、ユーザー対話に関連付けられた境界を定義すること、およびそれらの境界をランタイムエンジンによって施行させることを可能にするために、クライアント側コードが利用されることが可能である。少なくともいくつかの実施形態においては、開発者は、JavaScript（登録商標）コードを通じて、スタートメニューショートカットの作成、ナビゲーション、およびいわゆるジャンプリスト統合、ならびに複数のその他の機能のためのさまざまな構成を提供することができる。

【0007】

[0005]同様の機能を指すために、複数の図面を通じて同じ番号が使用される。

【図面の簡単な説明】

【0008】

【図1】[0006]本明細書に記載のさまざまな原理が1つまたは複数の実施形態に従って採用されることが可能である動作環境を示す図である。

【図2】[0007]1つまたは複数の実施形態によるウェブアプリケーションウィンドウを示す図である。

【図3】[0008]1つまたは複数の実施形態によるJavaScript APIを示す図である。

【図4】[0009]1つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。

【図5】[0010]1つまたは複数の実施形態によるウェブアプリケーション対話方法におけるステップを示す流れ図である。

【図6】[0011]1つまたは複数の実施形態によるクライアントデスクトップの一部を示す図である。

【図7】[0012]1つまたは複数の実施形態によるJavaScript APIを示す図である。

【図8】[0013]1つまたは複数の実施形態による、ウェブサイトとカスタムジャンプリストの間における動的な対話を示す図である。

【図9】[0014]1つまたは複数の実施形態によるクライアントデスクトップの一部を示す図である。

【図10】[0015]1つまたは複数の実施形態によるクライアントデスクトップの一部を示す図である。

【図11】[0016]1つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。

10

20

30

40

50

【図 1 2】 [0017] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 1 3】 [0018] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 1 4】 [0019] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

【図 1 5】 [0020] 1 つまたは複数の実施形態によるドラッグアンドドロップオペレーションを図式的に示す図である。

【図 1 6】 [0021] 1 つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。

10

【図 1 7】 [0022] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

【図 1 8】 [0023] 1 つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。

【図 1 9】 [0024] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

【図 2 0】 [0025] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 2 1】 [0026] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

20

【図 2 2】 [0027] 1 つまたは複数の実施形態による、ウェブサイトを表示するブラウザと、証明書ストアと、関連付けられたウェブアプリケーションと、ウェブアプリケーション証明書ストアとの間における関係を示す図である。

【図 2 3】 [0028] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 2 4】 [0029] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 2 5】 [0030] 1 つまたは複数の実施形態による複数のウェブアプリケーションインスタンスの一例を示す図である。

【図 2 6】 [0031] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

30

【図 2 7】 [0032] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 2 8】 [0033] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 2 9】 [0034] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

【図 3 0】 [0035] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

【図 3 1】 [0036] 1 つまたは複数の実施形態によるクライアントデスクトップを示す図である。

40

【図 3 2】 [0037] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 3 3】 [0038] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 3 4】 [0039] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 3 5】 [0040] 1 つまたは複数の実施形態による、ウェブアプリケーションとブラウザの間における関係を示す図である。

【図 3 6】 [0041] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図で

50

ある。

【図 3 7】 [0042] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 3 8】 [0043] 1 つまたは複数の実施形態による一時的なウェブアプリケーションの一例を示す図である。

【図 3 9】 [0044] 1 つまたは複数の実施形態によるサイトモードブラウザ (site mode browser) を示す図である。

【図 4 0】 [0045] 1 つまたは複数の実施形態による方法におけるステップを示す流れ図である。

【図 4 1】 [0046] 1 つまたは複数の実施形態を実装するために利用されることが可能である例示的なシステムを示す図である。

【発明を実施するための形態】

【0009】

概観

[0047] さまざまな実施形態が、エンドユーザーがウェブアプリケーションおよびウェブサイトをクライアントデバイスデスクトップなどのクライアントデバイス上にインストールできるようにするメカニズムを提供する。1 つまたは複数の実施形態によれば、ウェブサイトに関連付けられた開発者が、ユーザー対話に関連付けられた境界を定義すること、およびそれらの境界をランタイムエンジンによって施行させることを可能にするために、クライアント側コードが利用されることが可能である。少なくともいくつかの実施形態においては、開発者は、JavaScript (登録商標) コードを通じて、スタートメニューショートカットの作成、ナビゲーション、およびいわゆるジャンプリスト統合、ならびに複数のその他の機能のためのさまざまな構成を提供することができる。

【0010】

[0048] 境界は、どのようにして機能がエンドユーザーに表示されるかに関する、開発者によって定義された経験と考えられることが可能である。境界は、別々のアプリケーションに関連付けられる可能性があるトップレベルドメインおよびサブドメインなどのウェブサイトドメイン、またはドメイン上にホストされるウェブサイトのサブセットに関連付けられる。したがって境界は、特定のウェブアプリケーションを機能させるために利用される一式のドメイン、サブドメイン、フォルダ、サブサイト、プロトコル、ホスト、パスなどによって定義されることが可能である。

【0011】

[0049] 1 つまたは複数の実施形態においては、ウェブサイトは、上述および後述の機能を選択することができる。この場合には、開発者は、自分たちのウェブサイトとユーザーの経験との境界を定義するコード (場合によっては、JavaScript (登録商標) で表される) を提供することができる。代替として、または追加として、上述および後述の機能を選択しないウェブサイトは、デフォルトの経験がユーザーに提供されるようにすることができる。

【0012】

[0050] 以降の考察においては、「動作環境」と題されたセクションが提供され、このセクションは、1 つまたは複数の実施形態が採用されることが可能である 1 つの環境を説明する。これに続いて、「統合インフラストラクチャー」と題されたセクションが、1 つまたは複数の実施形態に従ってウェブアプリケーションがクライアントデバイス上に統合されることを可能にするインフラストラクチャーを説明する。次いで、「ジャンプリスト統合」と題されたセクションが、いわゆるジャンプリストが 1 つまたは複数の実施形態に従ってどのようにして統合されることが可能であるかを説明する。次いで、「タスクバーの固定」と題されたセクションが、ウェブアプリケーションが 1 つまたは複数の実施形態に従ってどのようにしてタスクバーに固定されることが可能であるかを説明する。これに続いて、「証明書とログインセッションの関連付け」と題されたセクションが、証明書とログインセッションが 1 つまたは複数の実施形態に従ってどのようにして関連付けられるこ

10

20

30

40

50

とが可能であるかを説明する。次いで、「関連付けられた証明書を伴うウェブアプリケーションの作成および起動」と題されたセクションが、ウェブアプリケーションが1つまたは複数の実施形態に従ってどのようにして作成され起動されることが可能であるかを説明する。これに続いて、「ウェブアプリケーションタスクセッション」と題されたセクションが、1つまたは複数の実施形態によるタスクセッションの概念を説明する。次いで、「ウェブアプリケーションとブラウザの間における遷移」と題されたセクションが、ウェブアプリケーションとブラウザの間における遷移が1つまたは複数の実施形態に従ってどのようにして生じることが可能であるかを説明する。これに続いて、「ブラウザからの一時的なウェブアプリケーションの作成」と題されたセクションが、一時的なウェブアプリケーションが1つまたは複数の実施形態に従ってどのようにしてブラウザから作成されることが可能であるかを説明する。次いで、「インストールされたウェブアプリケーションへの一時的なウェブアプリケーションの変換」と題されたセクションが、一時的なウェブアプリケーションが1つまたは複数の実施形態に従ってどのようにしてインストールされたウェブアプリケーションに変換されることが可能であるかを説明する。これに続いて、「ウェブアプリケーションスーパーホームボタン」と題されたセクションが、1つまたは複数の実施形態によるウェブアプリケーションに関連付けられたホームボタンを説明する。最後に、「例示的なシステム」と題されたセクションが、1つまたは複数の実施形態を実装するために利用されることが可能である例示的なシステムを説明する。

10

【0013】

[0051]次いで、1つまたは複数の実施形態が実装されることが可能である例示的な動作環境について考察されたい。

20

動作環境

[0052]図1は、1つまたは複数の実施形態による動作環境を全体として100で示す。環境100は、1つまたは複数のプロセッサ104と、1つまたは複数のコンピューター可読記憶媒体106と、コンピューター可読記憶媒体上に存在して（1つまたは複数の）プロセッサによって実行可能である1つまたは複数のアプリケーション108とを有するコンピューティングデバイス102を含む。コンピューター可読記憶媒体は、典型的にはコンピューティングデバイスに関連付けられるすべての形態の揮発性および不揮発性のメモリーおよび／または記憶媒体を、限定ではなく例として含むことができる。そのような媒体は、ROM、RAM、フラッシュメモリー、ハードディスク、取り外し可能な媒体などを含むことができる。コンピューティングデバイスの具体的な一例が、以降で図41において示され、説明される。

30

【0014】

[0053]さらにコンピューティングデバイス102は、ウェブブラウザ110の形態のソフトウェアアプリケーションを含む。任意の適切なウェブブラウザが使用されることが可能であり、その例は、本明細書の譲受人およびその他から入手可能である。さらにコンピューター可読記憶媒体106は、上述および後述のように機能するウェブアプリケーションモードブラウザ111を含むことができる。ウェブアプリケーションモードブラウザ111は、ウェブサイトからのAPIコールを受け取り、ウェブサイトへのAPIコールを行い、ウェブアプリケーションインストールプロセスを監視し、境界を施行し、上述および後述の機能を可能にするランタイムエンジンとして機能する。オペレーションにおいて、ウェブアプリケーションモードブラウザは、完全なブラウザの軽量化バージョンであり、通常のブラウザ機能の多くがオフにされている。いくつかの場合においては、ウェブアプリケーションモードブラウザは、通常のブラウザコントロールの多くを含まない「クロムレス(chrome-less)」ブラウザと考えられることが可能である。しかし、いくつかのコマンドは、ミニチュアコントロールバーを通じて表示されることが可能である。したがってウェブアプリケーションモードブラウザは、ユーザーにとって気を散らすものの多くを取り除き、ウェブサイトが自分たちのウェブアプリケーションとユーザーがどのようにして対話するかをコントロールすることができる、方向付けられた、ウェブサイトによって定義されたユーザー経験を可能にする。

40

50

【0015】

[0054]オペレーションにおいて、ウェブアプリケーションモードブラウザは、ウェブサイトと、クライアントデバイスのオペレーティングシステムとの間に論理的に存在すると考えられることが可能である。したがってウェブアプリケーションモードブラウザは、ウェブサイトからのコールを受け取り、それに応答して、本明細書に記載の機能に影響を与えるためにオペレーティングシステムコールを行うことができる。同様に、ウェブアプリケーションモードブラウザは、ウェブサイトの機能に影響を与えることになるコールをオペレーティングシステムから受け取ることができる。たとえばオペレーティングシステムは、デスクトップのタスクバーとの対話を可能にするAPIを表示する。ウェブアプリケーションモードブラウザは、ウェブサイトからのコールを受け取ることができ、それに応答して、以降で明らかになるようなタスクバー機能を可能にするAPIコールを行うことができる。

10

【0016】

[0055]ウェブアプリケーションモードブラウザ111は、スタンドアロンのコンポーネントとして実装されることが可能である。代替として、または追加として、ウェブアプリケーションモードブラウザ111は、アプリケーション108および／またはブラウザ110の一部として実装されることが可能である。

【0017】

[0056]さらに環境100は、インターネットなどのネットワーク112と、1つまたは複数のウェブサイト114とを含み、それらのウェブサイト114との間で、コンテンツが送受信されることが可能である。そのようなコンテンツは、上述および後述のように、クライアントデスクトップ上に統合される、またはクライアントデバイスを通じてその他の形で使用可能である、ウェブアプリケーションなどのコンテンツを含むことができる。

20

【0018】

[0057]コンピューティングデバイス102は、限定ではなく例として、デスクトップコンピューター、ポータブルコンピューター、携帯情報端末(PDA)などのハンドヘルドコンピューター、携帯電話など、任意の適切なコンピューティングデバイスとして具体化されることが可能である。

【0019】

[0058]例示的な動作環境について説明してきたが、次いで、ウェブアプリケーションがクライアントデバイス上に統合されることを可能にするインフラストラクチャーに関する論考について考察されたい。

30

【0020】

統合インフラストラクチャー

[0059]1つまたは複数の実施形態によれば、ウェブサイトは、より一般的な「サイトモード(site mode)」の経験の一部として提供されるドメインナビゲーションを選択することができる。ドメインナビゲーションは、ウェブサイトが、ユーザーが特定のドメインの内側および外側のリンクにアクセスする際の自分たちの現行のウェブサイトの動作をカスタマイズできるようにする。開発者によって指定された境界の内側のリンクにアクセスする際には、コンテンツは、関連付けられたウェブアプリケーションの一部としてウェブアプリケーションモードブラウザによってレンダリングされるウェブアプリケーションウィンドウ内でレンダリングされ消費されることが可能である。開発者によって指定された境界の外側のリンクにアクセスする際には、関連付けられたコンテンツは、ウェブアプリケーションモードブラウザの外側にあるデフォルトブラウザの内側でレンダリングされ消費されることが可能である。これは、ウェブサイトが、どのドメインがそのウェブサイトの拡張として扱われるべきか、およびどれがそのウェブサイトの拡張として扱われるべきでないかを定義できるようにする。

40

【0021】

[0060]1つまたは複数の実施形態においては、ナビゲーションドメインは、ウェブ開発者によって定義されることが可能であり、統合されたウェブサイトの一部としてウェブア

50

アプリケーションモードブラウザーによって表示されるか、またはウェブアプリケーションモードブラウザーの外側でデフォルトブラウザー内に表示されるコンテンツのリンクを識別することができる。さらに、ウェブアプリケーションページの集合を共に関連付けるために使用されるデフォルトドメインパラメータが定義されることが可能である。

【0022】

[0061]一例として、下記のインラインドメインページ定義 (in-line domain page definition) を考察されたい。

```
*contoso.crm.dynamics.com\*;*.microsoft.com\*;
```

[0062]このドメインページ定義は、そのすぐ下のフォームのURIが同じデスクトップウェブアプリケーションウィンドウ内に表示されることを可能にする。

```
sales.contoso.crm.dynamics.com\*
```

```
hr.contoso.crm.dynamics\*
```

```
*.microsoft.com\crm\
```

[0063]同様に、このドメインページ定義は、その他のURIが、たとえリンク参照がデスクトップウェブアプリケーションウィンドウ内のページの内側にあっても、デスクトップウェブアプリケーションウィンドウの外側に表示されるようにする。

```
www.bing.com
```

```
home.live.com
```

[0064]上記のドメインページ定義においては、ウェブアプリケーションインストールAPIの内側でワイルドカードが利用される。このAPIは、典型的には、ウェブサイトによって提供されるウェブサイト統合リンクをユーザーが選択すると、ウェブサイトによってコールされる。このAPIは、デスクトップ、タスクバー、またはスタートメニュー、またはウェブサイトショートカットを起動するために使用されることになるその他の任意の適切な場所における情報およびコンテンツをウェブアプリケーションファイルすなわち「.webapp」ファイルに投入することができる。ウェブアプリケーションファイルを示すために任意の適切なファイル拡張子が使用されることが可能であるということを認識および理解されたい。ナビゲーションドメインおよびその他の境界の情報は、.webappファイル内に格納される。

【0023】

[0065].webappファイルが起動されると、その内部のナビゲーションドメインが、ウェブアプリケーションモードブラウザー111によって施行される。ユーザーによって選択されたリンク、またはウェブサイトによってアクセスされたリンクは、ワイルドカードドメインとマッチする限り、ウェブアプリケーションウィンドウの内側で実行し続ける。しかし、定義されたナビゲーションドメインの外側にあるウェブサイトが検知されると、デフォルトブラウザーが、インスタンス化されるかまたはその他の形で使用され、そのウェブサイトに関連付けられたコンテンツが、ウェブアプリケーションウィンドウの外側に、かつデフォルトブラウザーの内側に表示される。

【0024】

[0066]一例として、図2を考察されたい。図2は、a.com(202)、b.com(204)、およびd.com(206)のナビゲーションドメインセットを有するウェブアプリケーションウィンドウ200を示し、これは、これらのドメインからのすべてのページが、ウェブアプリケーションウィンドウ200の内側に表示されることを意味する。c.com(208)またはe.com(210)からのページは、ウェブアプリケーションウィンドウ200内からアクセスされると、デフォルトブラウザーウィンドウに表示され、ウェブアプリケーションウィンドウ200には表示されない。

【0025】

[0067]図3は、1つまたは複数の実施形態によるJavaScript APIを300で示す。示されたJavaScript APIは、ウェブサイトがウェブアプリケーションをクライアントデスクトップと統合できるようにする。このAPIは、ウェブアプリケーションまたはランタイムエンジンによって施行されることになるナビゲーションド

10

20

30

40

50

メインを定義する。この例においては、ナビゲーションドメインは、上述のようなワイルドカード表現を使用して記述される。このAPIは、クライアントデバイス上のコンテンツおよび情報を、webappアプリケーションファイル302に投入すること、またはそれらのコンテンツおよび情報で、webappアプリケーションファイル302を更新することを可能にし、ナビゲーションドメインおよびその他の情報を、webappアプリケーションファイル302に格納する。これらのナビゲーションドメインは、ウェブアプリケーションが起動されるときに施行される。

【0026】

[0068]図示され説明された実施形態においては、webappアプリケーションファイル302は、ウェブサイトが自分のサイトモード構成に関して定義した情報を含む。この情報は、ウェブアプリケーションモードブラウザによって表示される最初のページであるスタートURLと、ウェブサイトが指定したナビゲーションドメインのすべてと、ウェブアプリケーションタイトルと、いわゆるファビコンとを含む。以降で説明されるように、その他の情報が含まれることも可能である。

10

【0027】

[0069]次いで、ウェブアプリケーションがクライアント側で起動されると、ウェブアプリケーションモードブラウザは、ウェブアプリケーションファイルを読み取り、その中で定義された境界を施行する。上述のように、ウェブアプリケーション経験は、特定のウェブサイトおよびそのニュアンスに精通した開発者によって定義されるため、完全な統合されたユーザー経験が提供されることが可能である。

20

【0028】

[0070]図4は、1つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアントデバイス上で実行するソフトウェアによって実行される。

【0029】

[0071]ステップ400は、ウェブサイト統合機能を選択する。このステップは、任意の適切な方法で実行されることが可能である。典型的には、このステップは、ユーザーが、ウェブアプリケーションインストールプロセスを開始するために手段を選択するか、またはその他の形でアクションを起こすと、実行される。たとえばユーザーは、ウェブアプリケーションを統合できるようにするリンクを選択することができる。これがどのようにして行われることが可能であるかについての具体的な例が、以降で提供される。

30

【0030】

[0072]ステップ402は、クライアントデバイス上で、ウェブアプリケーションファイル（ここでは、「webapp」ファイルと示される）を作成する。このファイルは、はじめに作成されたときには、ウェブサイトから、またはウェブサイトのために受け取られるコンテンツおよび情報が後に投入されることが可能であるアーチファクト（artifact）またはシェル（shell）を構成することができる。ステップ404は、ウェブアプリケーションコンテンツをウェブアプリケーションファイルに投入する。このステップは、任意の適切な方法で実行されることが可能である。たとえば、このステップは、JavaScript APIの使用を通じて実行されることが可能であり、そのJavaScript APIの例は、上記および下記で提供されている。代替として、または追加として、このステップは、HTMLなどのマークアップの使用を通じて実行されることが可能である。

40

【0031】

[0073]クライアント上にウェブアプリケーションファイルを作成し、そのウェブアプリケーションファイルにコンテンツを投入すると、ウェブアプリケーションは、起動され対話されることが可能になる。

【0032】

50

[0074]図5は、1つまたは複数の実施形態によるウェブアプリケーション対話方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアント上で実行するソフトウェア、およびウェブサイトをサポートするサーバにおいて実行するソフトウェアによって実行される。したがって、この図の1つの列は、クライアントによって、またはクライアントにおいて、たとえばウェブアプリケーションモードブラウザによって実行されるステップを示すために「クライアント」と表示され、1つの列は、ウェブサイトによって、またはウェブサイトのために実行されるステップを示すために「ウェブサイト」と表示される。

10

【0033】

[0075]ステップ500は、サイトモードのユーザー選択を受け取る。このステップは、任意の適切な方法で実行されることが可能である。たとえば、サイトモード選択を受け取るために、クライアントデスクトップ上にインストールされたショートカットが利用されることが可能である。サイトモード選択を受け取ったことに応答して、ステップ502は、スタートURLを要求する。スタートURLは、上述のウェブアプリケーションファイル内で見つけれることが可能である。

【0034】

[0076]ステップ504は、スタートURL要求を受け取る。ステップ506は、スタートURLに関連付けられたコンテンツを含む、関連付けられたウェブリソースをクライアントに返す。

20

【0035】

[0077]ステップ508は、関連付けられたウェブリソースを受け取り、ステップ510は、それらのウェブリソースをウェブアプリケーションウィンドウ内にレンダリングする。上述のように、ウェブアプリケーションウィンドウは、ウェブアプリケーションモードブラウザによってレンダリングされる。ステップ512は、ウェブアプリケーションウィンドウ内にレンダリングされたリソースに関するユーザー対話を受け取る。ユーザー対話は、任意の適切なタイプのユーザー対話を含むことができる。たとえばユーザー対話は、ウェブアプリケーションウィンドウ内から生じるナビゲーションアクティビティを含むことができる。ステップ514は、ユーザー対話が、ウェブアプリケーションファイルによって定義された境界内にあるかどうかを確かめる。ユーザー対話が、ウェブアプリケーションファイルによって定義された境界内にある場合には、ステップ516は、ユーザー対話に関連付けられたコンテンツをウェブアプリケーションウィンドウ内にレンダリングする。その一方で、ユーザー対話が、ウェブアプリケーションファイルによって定義された境界内にない場合には、ステップ518は、ユーザー対話に関連付けられたコンテンツをデフォルトウェブブラウザ内にレンダリングする。

30

【0036】

[0078]このようにして、ウェブサイト開発者によって所望されるとおりにユーザー経験が保持されることを確実にするために、特定のウェブサイトに関してそれらの開発者によって定義された境界が施行されることが可能である。たとえば、ウェブサイトによって定義されたスタートURLはホームページであり、ウェブアプリケーションモードにおけるそのホームページへの以降のアクセスは、ユーザーの独自に定義したブラウザホームページではなく、スタートURLへとナビゲートする。これは、何らかの無関係なホームページではなく、ウェブアプリケーション固有のページへの迅速なアクセスを可能にする。これは、エンドユーザーが、正確に機能することもあれば正確に機能しないこともある自分たち自身のサイト固有の経験を定義しなければならない状況を緩和する。したがって、完全な、統合された、インテリジェントに管理された経験が、エンドユーザーに提供されることが可能である。

40

【0037】

[0079]例示的なインストールおよび対話の経験について説明してきたが、次いで、ジャ

50

ンプリスト統合の概念について考察されたい。

ジャンプリスト統合

[0080] 1つまたは複数の実施形態においては、開発者は、ウェブサイトと対話するために使用されることが可能であるデスクトップ統合中の一連のいわゆるジャンプリストタスクをウェブサイトが定義できるようにすることができる。さらに、少なくともいくつかの実施形態においては、ウェブサイトは、カスタムジャンプリストを作成および更新することができる。

【0038】

[0081] ジャンプリストは、ユーザーに提示される関連するタスクまたはコンテンツのセットを構成するリストと考えられることが可能である。ジャンプリストを通じて、ウェブサイトは、ユーザーに対する関連する有用な情報の抽出を促進することができる。ジャンプリストは、特定のウェブアプリケーションの性能または機能に関連する。たとえば、電子メールアプリケーション用のジャンプリストは、連絡先を開く能力、新たな電子メールメッセージを作成する能力などをユーザーに与えるタスクを含むことができる。いくつかの場合においては、ジャンプリストは、関連する最も頻繁に使用されるコマンドおよびデータのリストを含むことができる。

【0039】

[0082] 1つまたは複数の実施形態においては、ジャンプリスト機能は、静的な要素と動的な要素の両方を含むように実装されることが可能である。

[0083] 開発者は、上述のようにウェブアプリケーションファイルを投入するウェブアプリケーションインストールプロセスの間に静的な要素を定義することができる。静的な要素に関連付けられた設定は、ウェブアプリケーションファイル内に格納されることが可能である。1つまたは複数の実施形態においては、それらの設定は、リスト名および関連付けられたタスクを含むことができる。少なくともいくつかの場合においては、静的な要素は、一般に使用される機能を表す要素を構成することができる。

【0040】

[0084] 動的な要素に関連付けられた設定は、ウェブアプリケーションウィンドウ内で動作するウェブサイトページによって駆動されることが可能である。これらの設定は、ユーザーにとって識別可能な表示（しるし、indicia）を動的に表示する設定を含む。たとえば、ある設定は、アイテムをカスタムジャンプリストに追加することができ、ある設定は、オーバーレイアイコンを表示することができ、オーバーレイアイコンの例は、以降で提供される。少なくともいくつかの実施形態においては、動的な設定は、ウェブアプリケーションが起動されるたびにクリアされることが可能であり、ウェブアプリケーションスクリプトコードによって構成されることが可能である。

【0041】

[0085] 一実施形態によるカスタムジャンプリストの一例として、図6を考察されたい。図6は、クライアントデスクトップの一部を全体として600で示す。カスタムジャンプリスト602が、1つまたは複数の実施形態に従って表示される。ここでは、静的な要素が604で示され、動的な要素が606で示される。この例においては、静的な要素のリスト名は、「Tasks（タスク）」であり、タスクすなわち静的な要素は、「New Email Message（新たな電子メールメッセージ）」、「New Appointment（新たなアポイントメント）」、「New Contact（新たな連絡先）」、および「New Task（新たなタスク）」を含む。動的な要素のリスト名は、「Today（今日）」であり、タスクすなわち動的な要素は、この例においては、ユーザーのカレンダーから生成されるリマインダーを含む。動的な要素は、関連付けられたウェブサイトによって動的に投入される。したがって、この例においては、動的な要素またはコンテンツは、典型的にはユーザーのアクションとは無関係に、ユーザーに通知を提供することに関連付けられる。

【0042】

[0086] 上述のように、ジャンプリストは、デスクトップ統合プロセスの間に定義される

ことが可能である。タスクおよびジャンプリスト定義は、ウェブアプリケーションファイル内に格納されることが可能である。一例として、図7を考察されたい。図7は、1つまたは複数の実施形態によるJavaScript APIを700で示す。示されたJavaScript APIは、ウェブサイトがクライアントデスクトップと統合すること、およびジャンプリストを定義することを可能にする。このJavaScript APIは、「custom task (カスタムタスク)」および「customJumpList (カスタムジャンプリスト)」の存在を除いて、図3に関して説明されたJavaScript APIと同じかまたは同様のものとするができる。簡潔にするために、図3の内容のうちのいくつかが省略されている。少なくともいくつかの実施形態においては、ジャンプリストタスクの静的なリストの最初の作成は、マークアップによって、たとえばHTMLドキュメント内で定義されるHTMLタグを使用して定義されることが可能である。

10

【0043】

[0087]たとえば、静的なリストの機能を記述するためにメタタグを使用するすぐ下の例を考察されたい。

```
<META name="msapplication-task"
content="name=Task;uri=http://marap/test3.html;icon-
uri=http://marap/icon3.ico"/>
```

[0088]タスクが実行される際、1つまたは複数の実施形態においては、数個のオプションがある。たとえば、タスクに関連付けられたURLは、現在閲覧されているウェブページを含む同じウェブアプリケーション/ブラウザウィンドウ内で開かれることが可能である。代替として、または追加として、新たなページが起動されることも可能である。代替として、または追加として、新たなポップアップウィンドウが表示されることも可能である。

20

【0044】

[0089]これらのパラメータは、定義されてシステムによって読み取られると、ユーザーが、ウェブサイトスタートメニューに追加すること、または以降で説明されるようにウェブサイトをタスクバーに固定することなどによって、ウェブサイトを自分のデスクトップ上にインストールする際に、使用されることが可能である。同時に、少なくともいくつかの実施形態においては、ジャンプリスト上のウェブサイトクライアントコードによって操作されることが可能である動的な値のセットがある。

30

【0045】

[0090]「カスタムタスク」は、上述のように静的なタスクを定義するためにウェブサイトによって利用される。この例においては、静的なタスクは、新たなメッセージの静的なタスクである。このAPIは、デスクトップ上に、webappアプリケーションファイル702を作成し、(図3の例におけるような)ナビゲーションドメイン、およびジャンプリストに関連付けられた情報などのその他の情報を格納する。

【0046】

[0091]図示され説明された実施形態においては、webappアプリケーションファイル702は、ウェブサイトが自分のサイトモード構成に関して定義した情報を含む。この情報は、スタートURLと、ウェブサイトが指定したナビゲーションドメインのすべて(具体的には図示されていない)と、ウェブアプリケーションタイトルと、いわゆるファビコンとを含む。その他の情報は、上述の新たなメッセージに関連付けられたカスタムタスクと、「custom list (カスタムリスト)」とを含む。この例においては、カスタムリスト要素は、ウェブアプリケーションがクライアント上で動作しているときにウェブサイトによって動的に投入されることが可能である動的な要素である。ここでは、「friends (友人)」の表示は、動的なコンテンツに関連付けられるヘッダを含む。したがって、この場合には、動的なコンテンツは、ユーザーの友人がオンライン状態であるときにユーザーに通知を提供することに関連付けられる。ここでは、カスタムリストは、ウェブアプリケーション内に存在しない別個のAPIである。しかしカスタムタスク

40

50

は、ウェブアプリケーション内に存在する。

【0047】

[0092]オペレーションにおいて、静的な要素に関連付けられたアイテムは、プリフェッチされ、ウェブアプリケーションが動作しているためのためにキャッシュに格納されることが可能である。その一方で、動的な要素に関連付けられたアイテムは、ウェブアプリケーションが動作しているときにウェブサイトによってオンザフライで提供されることが可能である。動的な要素に関連付けられたアイテムがどのようにしてウェブアプリケーションにオンザフライで提供されることが可能であるかについての実施例として、図8を考察されたい。

【0048】

[0093]図8は、メッセージが着信したことをユーザーがわかるようにするためにウェブサイトがどのようにしてカスタムジャンプリストと動的に対話することができるかを示す。この例においては、JavaScript 800は、ウェブサイトが、ウェブアプリケーションモードブラウザー内にホストされるページに更新をどのようにして送信することができるかを示す。ブラウザー内で実行されるクライアント側コードは、更新を受け取ること、スクリーン上のコンテンツを更新すること、およびジャンプリストにそのリストを更新したいという要求を送信することを担当する。この例においては、ウェブサイトは、リアルタイムの経験を提供するために情報をジャンプリストに押し出すことができる。ここでは、新たなメッセージ、たとえばNew Message 0（新たなメッセージ0）およびNew Message 1（新たなメッセージ1）がウェブサイトによって受け取られると、ジャンプリスト802を更新するためにウェブアプリケーションからJavaScript（登録商標）コールが行われることが可能である。少なくともいくつかの実施形態においては、関連する情報が受け取られたことをユーザーに知らせるための通知がタスクバー804内に現れることができる。通知は、タスクバー内の任意の適切な場所に現れることができ、任意の適切なタイプの通知を含むことができる。たとえば通知は、ウェブアプリケーションに関連付けられるアイコン806のところに、またはその付近に現れることができる。代替として、または追加として、通知は、ユーザーの注目を得るために点滅することができる。

【0049】

[0094]一例として、図9を考察されたい。そこでは、クライアントデスクトップの一部が、全体として900で示される。タスクバー901が、現在のウェブアプリケーションに関連付けられたアイコン902を含む。ここでは、オーバーレイアイコン904が、アイコン902内にレンダリングされていることに留意されたい。この例においては、ユーザーが新たなメッセージを受け取っており、オーバーレイアイコン904がアイコン902内にレンダリングされるようにするために、上述のように、ウェブサイトがウェブアプリケーションへのコールを行っている。これは、新たなメッセージが着信しているという通知をユーザーに提供する。オーバーレイアイコン904を見たことに応答して、ユーザーは、現在生じているまたは生じつつあるアポイントメントを示す要素908を表示するために、ジャンプリスト905にアクセスすることができる。要素906も表示されるということにも留意されたい。JavaScript（登録商標）の抜粋910は、オーバーレイアイコンを更新するために使用されることが可能であるコードの一例を示す。

【0050】

[0095]ウェブサイトとウェブアプリケーションの間における動的な対話は、その他のさまざまな方法で生じることができる。たとえば、少なくともいくつかの実施形態においては、ウェブサイトに関するユーザーのアクションに応答して、プレビューウィンドウが、ウェブサイトからジャンプリストに押し出されることが可能である。図示され説明された実施形態においては、プレビューウィンドウは、ウェブサイトによって提供されるミニチュアビューである。プレビューウィンドウは、任意の適切なタイプのユーザーアクションに応答して提供されることが可能である。たとえば、少なくともいくつかの実施形態においては、ウェブサイトに関連付けられたタスクバーアイコンの上でのマウスホバーに応答

10

20

30

40

50

して、プレビューウィンドウが、ウェブサイトからジャンプリストに押し出されることが可能である。代替として、または追加として、ウェブアプリケーションのタスクバーアイコンの上にカーソルを置いて左クリックすることによって、プレビューウィンドウが提供されることも可能である。

【0051】

[0096]一例として、図10を考察されたい。ここでは、クライアントデスクトップの一部が、全体として1000で示される。タスクバー1001が、現在のウェブアプリケーションに関連付けられたアイコン1002を含む。ここでは、ユーザーが自分のカーソルをアイコン1002の上に置いていることに留意されたい。それに応答して、プレビューウィンドウ1004がレンダリングされている。オペレーションにおいては、自分のカーソルをそのアイコンの上に置くというユーザーのアクションに応答して、イベントがウェブページに伝送される。次いで、そのイベントを受け取ったことに応答して、ウェブページは、プレビューを動的に提供すること、またはキャッシュに格納されていたプレビューウィンドウがレンダリングされるようにすることが可能である。

10

【0052】

[0097]少なくともいくつかの実施形態においては、プレビューウィンドウは、ツールバーボタンおよび関連付けられた動作を定義するためにウェブサイトによって使用されることも可能である。たとえば、図示され説明された実施形態においては、ウェブアプリケーションは、メディアプレーヤーアプリケーションを含み、3つのツールバーボタンが、プレビューウィンドウ1004の上にオーバーレイされるユーザーインターフェース手段1006内に現れる。これらのボタンは、一時停止ボタン、停止ボタン、および再生ボタンを含む。少なくともいくつかの実施形態においては、ツールバーボタンは、リモートサーバと対話しなければならない状況を回避するクライアント側コードを使用して実装されることが可能である。たとえば個々のボタンは、特定のウェブページ用として登録されることが可能である。それぞれのボタンは、設定されてIDを割り当てられる。1つの「リスナー」が、すべてのボタンイベントに関して登録される。ボタンが押されると、イベントが生成され、ブラウザーに返信され、次いでブラウザーは、そのイベントを、登録されたイベントリスナーに伝える。そのイベントは、押されたボタンを含む。これは、ボタンどうしの間におけるあいまい性の除去を可能にする。

20

【0053】

実施例

[0098]一実施例においては、ウェブ開発者は、ジャンプリスト内のカスタムリストを更新するために、およびタスクバーオーバーレイアイコンを更新するために、下記のJavaScript（登録商標）機能を使用することができる。

30

【0054】

リスト作成動作

[0099]これは、カスタムリストのタイトルであるリスト名を定義する。この値は、リストヘッダとして表示されることになる。任意選択で、アイテムの名前と、そのアイテムに関するURI値と、そのアイテムに関連付けられたイメージとを含むアイテムリストが、はじめにリストの投入を行うために提供されることが可能である。この機能は、ブラウザーがウェブアプリケーションモードで始動される場合にサポートされることが可能である。

40

【0055】

リスト更新動作

[0100]特定のリストアイテムを更新するために、リストアイテム値が提供される。リストアイテム値は、アイテムに関する名前と、そのアイテムに関するURI値と、アイテムに関連付けられたイメージとを含む。この機能は、ブラウザーがウェブアプリケーションモードで始動される場合にサポートされることが可能である。

【0056】

オーバーレイアイコンの設定

50

[00101] 現行のタスクバーアイコンのためのオーバーレイとして使用されることになるアイコンを指すURI値が指定される。この機能は、ブラウザーがウェブアプリケーションモードで始動される場合にサポートされることが可能である。

【0057】

プレビューイメージの設定

[00102] タスクバープレビューイメージ（またはサムバープレビュー（thumbbar preview））のための絵画表現として使用されることになるイメージを指すURI。ユーザーがマウスの左ボタンでタスクバーアイコンをクリックすると、プレビューが表示される。

【0058】

オーバーレイアイコンのクリア

10

[00103] これは、タスクバーアイコン上の現行のオーバーレイアイコンを取り除く。この機能は、ブラウザーがウェブアプリケーションモードで始動される場合にサポートされることが可能である。

【0059】

[00104] ウェブ開発者は、特定のウェブサイトのタスクバープレビューウィンドウ内に表示される一式のツールバーボタンを定義および修正するために、下記のJavaScript（登録商標）機能を使用することができる。

【0060】

ツールバーボタンのインストール

[00105] ボタンIDのリストが、ツールチップおよびイメージURLと共に指定される。イベントは、ユーザーがツールバーボタンを選択すると、処理用としてウェブサイトに渡される。次いでウェブサイトは、ボタンイベントどうしの間においてあいまい性を除去することができる。このコールは、ボタンが表示されるようにするためにサイトモードウィンドウが開かれると、少なくとも1回実行される。この機能は、ブラウザーがウェブアプリケーションモードで始動される場合にサポートされる。

20

【0061】

イメージの更新

[00106] これは、指定されるボタンIDの状態および可視性を識別する。状態は、有効にされるかまたは無効にされることが可能である。少なくともいくつかの実施形態においては、デフォルトでは、ボタンは有効にされる。ビューは、表示または非表示のいずれかとすることができる。デフォルトでは、定義されたボタンは可視である。この機能は、ブラウザーがウェブアプリケーションモードで始動される場合にサポートされる。

30

【0062】

[00107] 図11は、1つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアントデバイス上で実行するソフトウェアによって実行される。

【0063】

[00108] ステップ1100は、ウェブサイト統合機能を選択する。このステップは、任意の適切な方法で実行されることが可能である。典型的には、このステップは、ユーザーが、ウェブアプリケーションインストールプロセスを開始するために手段を選択するか、またはその他の形でアクションを起こすと、実行される。たとえばユーザーは、ウェブアプリケーションを統合できるようにするリンクを選択することができる。これがどのようにして行われることが可能であるかについての具体的な例が、以降で提供される。

40

【0064】

[00109] ステップ1102は、クライアントデバイス上で、ウェブアプリケーションファイル（ここでは、「webapp」ファイルと示される）を作成する。このファイルは、はじめに作成されたときには、ウェブサイトから、またはウェブサイトのために受け取られるコンテンツおよび情報が後に投入されることが可能であるアーチファクトまたは

50

シェルを構成することができる。ステップ1104は、この例においてはジャンプリストを含むウェブアプリケーションコンテンツをウェブアプリケーションファイルに投入する。このステップは、任意の適切な方法で実行されることが可能である。たとえば、このステップは、JavaScript APIの使用を通じて実行されることが可能であり、そのJavaScript APIの例は、上述のとおりである。代替として、または追加として、このステップの態様は、HTMLなどのマークアップの使用を通じて実行されることが可能である。

【0065】

[00110]クライアント上にウェブアプリケーションファイルを作成して投入を行うと、ウェブアプリケーションは、任意の適切なときに起動されることが可能になる。

10

[00111]図12は、1つまたは複数の実施形態による方法のステップを示す流れ図である。これらのステップは、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実行されることが可能である。少なくともいくつかの実施形態においては、この方法は、サーバ上で実行するソフトウェアと関連してクライアント上で実行するソフトウェアによって実行されることが可能である。

【0066】

[00112]ステップ1200は、ジャンプリストに関連付けられたユーザーアクションを受け取る。任意の適切なユーザーアクションが受け取られることが可能であり、それらのユーザーアクションの例は、上記で提供されている。たとえば、少なくともいくつかの実施形態においては、ユーザーアクションは、具体的に表示されたジャンプリストまたは表示されていないジャンプリストに関して受け取られることが可能である。代替として、または追加として、ユーザーアクションは、ウェブアプリケーションに関連付けられている表示されたアイコンに関して受け取られることが可能である。アイコンは、たとえばデスクトップタスクバーまたはその他の任意の適切な場所に表示されることが可能である。そのようなアクションの例は、上記で提供されている。

20

【0067】

[00113]ステップ1202は、ユーザーアクションに関連付けられたコンテンツを提示する。たとえば、提示されるコンテンツは、ジャンプリストそのものを含むことができる。ジャンプリストは、任意の適切なタイプのユーザーアクションにตอบสนองして提示されることが可能であり、そのユーザーアクションの例は、上記で提供されている。提示されるコンテンツは、ジャンプリストそのもの以外のコンテンツを含むこともできる。たとえばカスタムプレビューウィンドウが、たとえばタスクバーアイコンを左クリックすることなどのユーザーアクションにตอบสนองして提示されることが可能である。たとえばユーザーは、新たな電子メールメッセージを作成または編成することを選択することができる。

30

【0068】

[00114]図13は、1つまたは複数の実施形態による方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアント上で実行するソフトウェア、およびウェブサイトをサポートするサーバにおいて実行するソフトウェアによって実行される。したがって、この図の1つの列は、クライアントによって、またはクライアントにおいて実行されるステップを示すために「クライアント」と表示され、1つの列は、ウェブサイトによって、またはウェブサイトのために実行されるステップを示すために「ウェブサイト」と表示される。

40

【0069】

[00115]ステップ1300は、動的なジャンプリストアイテムに関連付けられた情報を受け取る。任意の適切なジャンプリストアイテムが、情報が受け取られる際の基礎として機能することができ、そのジャンプリストアイテムの例は、上記で提供されている。ステップ1302は、通知を作成し、ステップ1304は、ウェブアプリケーションを実行するクライアントデバイスにその通知を伝送する。

50

【0070】

[00116]ステップ1306は、その通知を受け取り、ステップ1308は、識別可能な通知をユーザーに提供する。任意の適切なタイプの識別可能な通知が提供されることが可能である。たとえば、少なくともいくつかの実施形態においては、識別可能な通知は、オーバーレイアイコンまたは点滅するウェブアプリケーションアイコンなどの視覚的に識別可能な通知とすることができる。代替として、または追加として、識別可能な通知は、ベルまたはブザーなどの可聴式に識別可能な通知を含むことができる。

【0071】

[00117]ジャンプリスト統合に関連付けられたさまざまな実施形態について考察してきたが、次いで、ウェブサイトが1つまたは複数の実施形態に従ってどのようにしてタスクバーなどのデスクトップ機能に「固定される」ようになることができるかに関する論考について考察されたい。

10

【0072】

タスクバーの固定

[00118]ウェブアプリケーションがクライアントデスクトップまたはタスクバーと統合されるようになることができるさまざまな方法がある。少なくともいくつかの実施形態においては、ウェブアプリケーションは、ドラッグアンドドロップオペレーションを通じてデスクトップと統合されることが可能である。代替として、または追加として、ウェブアプリケーションは、ウェブブラウザのメニュー選択を介して統合されることが可能である。代替として、または追加として、ウェブアプリケーションは、関連付けられたウェブ

20

サイトそのものによって統合されることが可能である。これらの実施形態のそれぞれは、以降でそれ自身の見出しのもとで論じられる。

【0073】

ドラッグアンドドロップオペレーションを通じた統合

[00119]1つまたは複数の実施形態においては、ウェブアプリケーションは、ドラッグアンドドロップオペレーションを通じてデスクトップまたはタスクバーと統合されることが可能である。一例として、図14を考察されたい。そこでは、クライアントデスクトップが、全体として1400で示される。ウェブブラウザウィンドウ1402は、ウェブサイトURLが中に表示された状態のアドレスバー1404を含む。URLに関連して、「ファビコン」と呼ばれるアイコン1406が示される。加えて、デスクトップ1400は、タスクバー1408を含む。

30

【0074】

[00120]図15は、1つまたは複数の実施形態によるドラッグアンドドロップオペレーションを図式的に示す。この例においては、カーソルがファビコン1406の上に置かれている。ファビコンを左クリックしてタスクバー1408へドラッグすることによって、関連付けられたウェブアプリケーション（この場合は、メッセージボードアプリケーション）が、デスクトップのタスクバー1408に固定されることが可能である。このドラッグアンドドロップオペレーションは、上述のようにウェブアプリケーションを統合する統合プロセスを始動し、ひいては、そのウェブアプリケーションをタスクバーに固定する。

【0075】

40

[00121]1つまたは複数の実施態様においては、ウェブアプリケーションに関連付けられたウェブページがブラウザ内でタブを開かれている場合には、ファビコンがタスクバー上にドロップされると、関連付けられたタブは、ブラウザのウィンドウから消えることができる。代替として、または追加として、タブが除去されるのではなく、代わりに、そのタブのコンテンツが「新たなタブ」のページに入れ替えられることが可能である。ブラウザウィンドウ内で単一のタブが開かれている場合には、そのブラウザウィンドウは、ファビコンがタスクバーに固定されると消えることになる。この時点で、元のサイトに含まれるタブは、ブラウザが閉じる前に、ただしウェブアプリケーションが固定された後で、除去されることが可能である。さらに、少なくともいくつかの実施形態においては、ドラッグオペレーションがタスクバーに入るときに、固定機能をユーザーに知らせる

50

ために、「タスクバーに固定する」という形態のツールチップが提示されることが可能である。

【0076】

[00122]さらに、タスクバーに固定されたウェブサイトまたはウェブアプリケーションの状態は、そのウェブアプリケーションが初めてインスタンス化されるときに、新たに表示されるウィンドウに移行されることが可能である。これは、ユーザーがそのアプリケーションを使用できるようになるためにサイトに証明書を再入力することを不要とすることができる。

【0077】

[00123]ウェブサイトまたはウェブアプリケーションをタスクバーに固定して、上述のようなインストールプロセスを完了すると、ウェブアプリケーションは、関連付けられたファビコンをクリックするだけでタスクバーから起動されることが可能となる。

【0078】

[00124]図16は、1つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアント上で実行するソフトウェアによって実行される。

【0079】

[00125]ステップ1600は、ウェブアプリケーションインストールに関連付けられたドラッグアンドドロップオペレーションの表示を受け取る。このステップは、任意の適切な方法で実行されることが可能である。すぐ上で説明された実施形態においては、このステップは、ユーザーが、関連付けられたウェブアプリケーションを自分のデスクトップに統合したいという希望をウェブサイトを示すために、ウェブサイトに関連付けられたファビコンなどのアイコンをドラッグアンドドロップすると、実行される。たとえばユーザーは、タスクバー、クイック起動エリア、または、デスクトップキャンバスなど、デスクトップ上のその他の何らかの場所にファビコンをドラッグアンドドロップすることができる。ステップ1602は、ドラッグアンドドロップオペレーションに応答してウェブアプリケーションファイルを作成する。図示され説明された実施形態においては、はじめに作成されるウェブアプリケーションファイルは、ジャンプリストタスク、スタートURL、ファビコン、または、静的なジャンプリストタスク、代替スタートURL、代替ファビコンなどのその他のプロパティーなど、関連付けられたウェブサイトからの情報をまだ含まないアーチファクトまたはシェルである。これらは、上述のような新たなマークアップおよび/またはJavaScript APIを通じて後から追加されることが可能である。JavaScript APIを採用する技術以外の技術が、特許請求される主題の趣旨および範囲から逸脱することなく利用されることが可能であるということを認識および理解されたい。

【0080】

ブラウザーメニュー選択を通じた統合

[00126]1つまたは複数の実施形態においては、ウェブアプリケーションは、ウェブブラウザーのメニュー選択を介して統合されることが可能である。一例として、図17を考察されたい。そこでは、クライアントデスクトップが、全体として1700で示される。ウェブブラウザーウィンドウ1702は、URLが表示された状態のアドレスバー1704を含む。さらにデスクトップ1700は、タスクバー1706を含む。ページメニューの形態のブラウザーメニューアイテム1708が示される。メニュー選択1710を表示するためにページメニューをドロップダウンすることによって、メニューアイテムまたは選択「Add to Start Menu (スタートメニューに追加)」が表示される。このオプションを選択することによって、ウェブサイトまたはウェブアプリケーションがデスクトップのスタートメニューに追加されることが可能であり、上述のようにインストールプロセスが開始されることが可能である。代替として、または追加として、インス

10

20

30

40

50

ツールプロセスの開始を可能にするために「Add to Task Bar (タスクバーに追加)」のメニューアイテムまたは選択が表示されることも可能である。

【0081】

[00127]図18は、1つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアント上で実行するソフトウェアによって実行される。

【0082】

[00128]ステップ1800は、ブラウザーメニュー選択を受け取る。このステップは、任意の適切な方法で実行されることが可能である。すぐ上で説明された実施形態においては、このステップは、ユーザーが特定のウェブサイトへナビゲートし、ブラウザーメニューをドロップダウンしてメニュー選択を表示し、そのウェブサイトに関連付けられたウェブアプリケーションのインストールを開始することに関連付けられたメニューアイテムを選択することによってアクションを起こすと、実行される。

【0083】

[00129]ステップ1802は、ブラウザーメニュー選択を受け取ったことに応答してウェブアプリケーションファイルを作成する。図示され説明された実施形態においては、はじめに作成されるウェブアプリケーションファイルは、ジャンプリストタスク、スタートURL、ファビコンなど、関連付けられたウェブサイトからの情報をまだ含まないアーチファクトまたはシェルである。これらは、上述のような新たなマークアップおよび/またはJavaScript APIを通じて後から追加されることが可能である。JavaScript APIを採用する技術以外の技術が、特許請求される主題の趣旨および範囲から逸脱することなく利用されることが可能であるということを認識および理解されたい。

【0084】

関連付けられたウェブサイトを通じた統合

[00130]1つまたは複数の実施形態においては、デスクトップとウェブアプリケーションの統合は、ウェブページから生じることができる。これらの実施形態においては、特定のウェブサイトが、ウェブアプリケーションをデスクトップ内に統合するための、JavaScript (登録商標)などのコードを使用することによる統合アクティビティーを選択することができる。これは、ウェブサイトが統合開始手段をコントロールできるようにする。

【0085】

[00131]一例として、図19を考察されたい。そこでは、クライアントデスクトップが、全体として1900で示される。ウェブブラウザーウィンドウ1902は、URLが中に表示された状態のアドレスバー1904を含む。加えて、デスクトップ1900は、タスクバー1906を含む。さらに、ブラウザーウィンドウ1902内に表示されたウェブページは、「Add to Desktop (デスクトップに追加)」と題されたリンク1908を含む。このリンクをクリックすることによって、ユーザーは、上述のようなウェブアプリケーションインストールプロセスを開始することができる。

【0086】

[00132]少なくともいくつかの実施形態においては、リンク選択が行われると、ユーザーが行おうとしているユーザーアクションと、新たに作成されたショートカットにアクセスするのはどこかとを説明するモーダル確認ダイアログが提示されることが可能である。この確認ダイアログは、提示されているページのソースURLをユーザーに提示することができる。表示されるURLは、ウェブサイトのフルパスを含むことができる。これは、ユーザーがインストールしたいと望むウェブサイトが正しいサイトから提供されていることをユーザーが確認できるようにすることができる。これは、悪意あるサブディレクトリに関連する状況を緩和することができる。

【0087】

[00133]さまざまな実施態様においては、デスクトップと統合されることになるウェブサイトのURLがチェックされて、そのウェブサイトが、そのウェブサイトを含むウェブページと同じドメイン上にあることが確認される。そのウェブサイトが、そのウェブサイトを含むウェブページと同じドメイン上にない場合には、エラーが表示されることが可能であり、そのオペレーションは、失敗に終わる可能性がある。ユーザーがオペレーションを確認すると、ダイアログが除去されることが可能であり、ウェブアプリケーションウィンドウが、正しいURLを伴って表示されることが可能である。

【0088】

[00134]図20は、1つまたは複数の実施形態によるインストール方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアント上で実行するソフトウェアによって実行される。

10

【0089】

[00135]ステップ2000は、統合リンクを伴うウェブページを表示する。このステップは、任意の適切な方法で実行されることが可能であり、その方法の例は、上記で提供されている。ステップ2002は、統合リンクの選択を受け取る。ステップ2004は、リンクの選択を受け取ったことに応答してウェブアプリケーションファイルを作成する。図示され説明された実施形態においては、はじめに作成されるウェブアプリケーションファイルは、ジャンプリストタスク、スタートURL、ファビコンなど、関連付けられたウェブサイトからの情報をまだ含まないアーチファクトまたはシェルである。これらは、上述のような新たなマークアップおよび／またはJavaScript APIを通じて後から追加されることが可能である。JavaScript APIを採用する技術以外の技術が、特許請求される主題の趣旨および範囲から逸脱することなく利用されることが可能であるということを認識および理解されたい。

20

【0090】

[00136]少なくともいくつかの実施形態においては、ウェブアプリケーションに関連付けられた複数のウィンドウまたはタブの視覚的な表示が、ユーザーに提供されることが可能である。一例として、図21を考察されたい。ここでは、デスクトップ2100が、ウェブアプリケーションアイコンが固定されているタスクバー2102を含む。アイコンをクリックすることによってウェブアプリケーションを起動するために、カーソルが使用されている。この例においては、ユーザーが、ウェブアプリケーションのスターティングページを使用して、複数のページヘナビゲートしたと想定する。ウェブアプリケーションは、ユーザーがナビゲートした先のウェブページの集合2104を示す視覚化を可能にすることができる。具体的には、この例においては、集合2104は、ウェブアプリケーションのためのスターティングページ2106と、ユーザーがスターティングページからナビゲートした先の後続のページ2108および2110とを含む。

30

【0091】

[00137]さまざまなウェブアプリケーション統合技術について考察してきたが、次いで、ユーザー証明書がどのようにしてウェブアプリケーションのためのログインセッションに関連付けられることが可能であるかに関する論考について考察されたい。

40

【0092】

証明書とログインセッションの関連付け

[00138]さまざまな実施形態は、ログイン／証明書情報(login or credential information)を利用するウェブサイトに関連付けられる1つまたは複数のウェブアプリケーションが、1つのウェブアプリケーションの別々のインスタンスにわたってログイン／証明書情報を保持する様式で統合されることを可能にする。

【0093】

[00139]ブラウザーが、ログイン／証明書情報を利用するウェブサイトヘナビゲートす

50

る場合には、そのログイン／証明書情報は、手動で入力されるか、または証明書ストアから検索されることが可能である。証明書ストアは、限定ではなく例として、ユーザー名およびパスワード、または特定のURLに関するユーザーの証明書など、ユーザーログイン情報を含むことができる。同一のURLまたはウェブサイトは、それぞれが別々のユーザーに関連付けられた複数のエントリーを有することができる。同様に、証明書ストアは、複数のURLに関するユーザーログイン情報または証明書を含むことができる。

【0094】

[00140]少なくともいくつかの実施形態においては、ユーザーがログインされているウェブサイトに関連付けられたウェブアプリケーションが、上述のようにデスクトップ上に統合され、対話されることが可能である。そのようなウェブアプリケーションが統合されるときに、プロセスは、そのウェブアプリケーションに関連付けられているのはどのウェブサイトかを、ならびにそのウェブサイトにログオンされているユーザーを特定する。プロセスは、関連付けられたログイン情報および／または証明書を求めて証明書ストアを検索する。次いでプロセスは、ユーザーと、作成されたウェブアプリケーションと、関連する証明書との間における関連付けを作成する。

【0095】

[00141]一例として、図22を考察されたい。図22は、1つまたは複数の実施形態による、ウェブサイトと、証明書ストアと、ウェブアプリケーションとの間における関係を示す。ブラウザー2200が、ログイン情報を利用するウェブサイトを表示する。URLを表示することに加えて、上述のようにウェブアプリケーションの統合を容易にするために選択されることが可能であるアイコン2210が表示される。証明書ストア2220が、複数のウェブサイトに関するログイン情報を含むエントリーを含む。そのような1つのエントリーが、2230で示される。ここでは、そのエントリーは、ウェブサイトURL、ユーザー名、およびパスワードを含む。図22は、ログイン情報を、ユーザー名およびパスワードを含むものとして示すが、他の形態のログイン情報または証明書が採用されることも可能であるということを認識および理解されたい。

【0096】

[00142]1つまたは複数の実施形態においては、アイコン2210は、上述のように、かつアイコン2240によって示されるように、選択されてタスクバー2235上にドラッグアンドドロップされることが可能である。この手順は、ウェブサイトに関連付けられたウェブアプリケーションの統合を開始する。選択が受け取られると、プロセスは、どのユーザーが現在そのウェブサイトにログオンされているかを特定し、関連付けられた証明書を求めて証明書ストア2220を検索する。しかし、関連付けられた証明書は、特許請求される主題の趣旨から逸脱することなく、その他の方法で特定および／または入手されることが可能であるということを認識および理解されたい。

【0097】

[00143]ユーザーの証明書および／またはログイン情報を入手すると、証明書と、ウェブアプリケーション証明書ストア2250内のウェブアプリケーションとの間において関連付けが作成される。ウェブアプリケーション証明書ストア2250は、ウェブサイトと、ウェブアプリケーションと、関連する証明書との間における関連付けに関連するデータを含む1つまたは複数のエントリーを含む。

【0098】

[00144]たとえば図22は、エントリー2260を示し、このエントリー2260は、ユーザーj s m i t hに関するウェブサイト「a. c o m」に関連するログイン情報を含むものとして示される。URL、ユーザー名、およびパスワード情報に加えて、エントリー2260は、ウェブアプリケーションを対応する証明書に関連付けるために使用されることが可能であるアプリケーションIDすなわち「A p p I D」も含む。ウェブアプリケーション証明書ストア2250は、同じウェブサイト「a. c o m」に関する一方で別のユーザー、b s m i t hに関する第2のエントリー2270も含む。このメカニズムは、同じウェブサイトからの個々のウェブアプリケーションを別々のユーザーおよびそれらの

ユーザーの関連付けられた証明書に関連付けることを可能にする。

【0099】

[00145]図22には示されていないが、さまざまな形態のユーザーログイン情報および／または証明書が、1つのウェブアプリケーションに関連付けられることが可能である。たとえば一実施形態においては、関連付けは、証明書ストア2220内の情報へ戻るポイントまたは参照を含むことができる。別の実施形態においては、ウェブアプリケーション証明書ストア2250は、証明書ストアからコピーされた情報を含むことができる。さらに別の実施形態においては、ユーザー名およびパスワードとは別個の証明書、またはユーザー名およびパスワードに追加した証明書が、1つのウェブアプリケーションに関連付けられることが可能である。たとえばバイオメトリック情報が、ウェブアプリケーション証明書ストア内で作成される関連付けの基礎を形成することができる。

10

【0100】

[00146]ウェブサイトと、証明書ストアと、ウェブアプリケーションとの間における関係について説明してきたが、次いで、関連付けられた証明書を採用するウェブサイトに関してウェブアプリケーションがどのようにして統合されることが可能であるかを考察されたい。

【0101】

関連付けられた証明書を伴うウェブアプリケーションの作成および起動

[00147]図23は、1つまたは複数の実施形態による方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアント上で実行するソフトウェアによって実行される。

20

【0102】

[00148]ステップ2300は、ウェブサイト統合機能の選択を受け取る。これがどのようにして行われることが可能であるかの例は、上記で提供されている。上述のように、ウェブサイト統合機能は、クライアントのデスクトップ上におけるウェブアプリケーションのインストールに関連付けられる。ウェブサイト統合機能の選択を受け取ったことに応答して、ステップ2302は、上述のように、クライアントのデスクトップ上においてウェブアプリケーションをインストールするためのインストールプロセスを開始する。具体的には、クライアント上におけるこのプロセスの開始は、上述のようにウェブアプリケーションファイルを作成することを含むことができる。ステップ2304は、ウェブサイトに関連付けられた現在のウェブセッションに関連付けられたセッション情報を入手する。このステップは、任意の適切な方法で実行されることが可能である。たとえば一実施形態においては、セッション情報は、ウェブサイトに関連付けられたコンテンツをレンダリングするブラウザと、インストールプロセスとの間における共有メモリーコンポーネントを使用して入手されることが可能である。別の実施形態においては、ウェブサイトは、ウェブサイト統合機能が選択されると、セッション情報を自動的に転送することができる。さらに別の実施形態においては、セッション情報がブラウザによって格納されて、後からクエリーされることが可能である。

30

40

【0103】

[00149]セッション情報を入手したことに応答して、ステップ2306は、そのセッション情報に関連した証明書を入手する。一実施形態においては、証明書ストアは、ユーザーがログインされるウェブサイトに関連したログインおよび／または証明書を求めてクエリーされることが可能である。たとえば証明書ストアは、現在のセッションのウェブサイトおよびユーザーに関連付けられたユーザー名およびパスワードを求めてクエリーされることが可能である。ステップ2308は、セッション情報に関連した証明書および／またはログイン情報を1つのウェブアプリケーションに関連付ける。このステップは、任意の適切な方法で実行されることが可能である。たとえば証明書は、その後の参照のためにウェブアプリケーション証明書ストアにコピーされることが可能である。代替として、また

50

は追加として、証明書ストア内の証明書へのポインタまたは参照が、ウェブアプリケーション証明書ストア内に配置されることも可能である。1つのウェブアプリケーションのそれぞれのインスタンスと、関連した証明書および／またはログイン情報とに関する一意のIDを作成するために、セッション情報および／または証明書に少なくとも部分的に基づいて1つの識別番号が生成されることが可能である。この情報は、取得された証明書および／またはログイン情報を1つのウェブアプリケーションに関連付けるために、ウェブアプリケーション証明書ストアのエントリーに加えられることが可能である。しかし、特許請求される主題の趣旨および範囲から逸脱することなく、証明書を1つのウェブアプリケーションに関連付けるために任意の適切な技術が採用されることが可能であるということ認識および理解されたい。

10

【0104】

[00150]上述のように、それぞれのウェブアプリケーションインスタンスごとの一意のIDは、複数のウェブアプリケーションの複数のインスタンスが同じURLまたはウェブサイトに関連付けられ、かつ、それぞれのインスタンスが別々のユーザー証明書に関連付けられている状態を可能にする。

【0105】

[00151]図24は、1つまたは複数の実施形態による方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せによって実行されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、クライアント上で実行するソフトウェアによって実行される。

20

【0106】

[00152]ステップ2400は、起動するウェブアプリケーションの選択を受け取る。このステップは、任意の適切な方法で実行されることが可能である。たとえばアイコンの選択は、上述のように、そのアイコンが固定されているタスクバーから行われることが可能である。代替として、または追加として、デスクトップスタートメニューまたはシステムツールバーからの選択が行われることも可能である。起動するウェブアプリケーションの選択を受け取ると、ステップ2402は、ウェブアプリケーションに関連付けられた証明書を検索する。たとえば、1つまたは複数の実施形態においては、現在ログインされているユーザーに関するログイン情報を検索する目的でウェブアプリケーション証明書ストアを参照するために、AppIDが使用されることが可能である。ステップ2404は、ウェブアプリケーションおよび関連する証明書に関連付けられたURLおよび／またはFormIDを検索する。ステップ2402および2404で説明された情報を取得すると、ステップ2406は、ユーザーの介入を伴わずに、関連付けられたウェブサイト に 自律的にログインする。ウェブサイトにログインすると、ステップ2408は、上述のように、スタートURLへナビゲートする。

30

【0107】

[00153]証明書に関連付けられた1つのウェブアプリケーションがどのようにして統合され起動されるかについて説明してきたが、次いで、関連付けられた証明書を伴う複数のウェブアプリケーションの複数のインスタンスに関する論考について考察されたい。

40

【0108】

関連付けられた証明書を伴う複数のウェブアプリケーションの複数のインスタンス

[00154]図25は、証明書に関連付けられる複数のウェブアプリケーションインスタンスの一例を示す。ウェブアプリケーション証明書ストア2500が、ウェブアプリケーションに関連するデータと、ウェブサイトに関する関連付けられた証明書とを含む。ウェブアプリケーション証明書ストア2500内に含まれるのは、エントリー2510および2520であり、これらはそれぞれ、別々のユーザーに関する。図25に示されるように、エントリー2510は、ウェブサイト「a.com」に関連し、関連付けられたウェブサイト に ログインするために使用されるユーザー名jsmithおよびパスワードbulldogsなど、さらなる情報を含む。エントリー2510は、AppIDを含み、このA

50

p p I Dは、そのエントリーをウェブアプリケーション2530に関連付けるために使用される。エントリー2520は、同様にウェブサイト「a. c o m」に関連し、その一方でユーザーb s m i t hに関するログイン情報を含み、ウェブアプリケーション2540に関連付けられる。図25は、エントリーを、U R L / F o r m I D、ユーザー名、パスワード、およびA p p I Dを含むように示すが、特許請求される主題の趣旨から逸脱することなく、別の形態の関連付けおよび証明書情報が使用されることが可能であるということ認識および理解されたい。したがって、ウェブアプリケーション証明書ストア内の複数のデータエントリーが、同じウェブサイトに関連付けられる別々のユーザー証明書をそれぞれが有してクライアントのデスクトップ上に統合される別々のウェブアプリケーションを識別するデータを含むことができる。

10

【0109】

[00155] 1つまたは複数の実施形態においては、同じウェブサイトに関連付けられる1つまたは複数のウェブアプリケーションが存在すること、または同時にアクティブ化されることが可能である。たとえば、ウェブアプリケーションが、デスクトップツールバーを介して起動されることと、そのウェブアプリケーションが起動されるときに、関連付けられた証明書を使用して、関連付けられたユーザーをウェブサイト自動的にログインさせることとを可能にするようにソフトウェアモジュールが構成されることが可能である。単一のウェブアプリケーションを起動することに加えて、第2のウェブアプリケーションが選択されて起動されるときに同じウェブサイトまたは別のウェブサイトのいずれかに関して別々のユーザー証明書を使用して第2のウェブアプリケーションが起動されることを可能にするようにソフトウェアモジュールが構成されることも可能である。

20

【0110】

[00156]たとえば、図25の2つのウェブアプリケーション2530、2540は、同時にアクティブ化される。上述のように、それぞれは、ウェブサイト「a. c o m」に関連し、その一方で別々の証明書に関連付けられている。ウェブアプリケーション2530は、選択されて起動される際に、ユーザーj s m i t hに関連付けられた証明書を使用して、ウェブサイト「a. c o m」自動的にログオンする。同様に、ウェブアプリケーション2540は、選択されて起動されるときに、ユーザーb s m i t hに関連付けられた証明書を使用して、ウェブサイト「a. c o m」にログオンする。このように、同じウェブサイトに関連付けられた複数のウェブアプリケーションの複数のインスタンスが、同時にアクティブ化され、別々の証明書に関連付けられることが可能である。

30

【0111】

[00157]関連付けられた証明書を伴うウェブアプリケーションの作成および起動という概念について説明してきたが、次いで、ウェブアプリケーションタスクセッションに関する論考について考察されたい。

【0112】

ウェブアプリケーションタスクセッション

[00158] 1つまたは複数の実施形態においては、ウェブアプリケーションに関連付けられた状態情報がシステムに保存されることを可能にするために、タスクセッションが作成されることが可能である。状態情報は、限定ではなく例として、セッションクッキー、J a v a S c r i p t（登録商標）状態、D O M状態、フォーム状態、タブおよびウィンドウの配置、ウィンドウサイズ、U R L、履歴などを含むことができる。

40

【0113】

[00159]特定のタスクセッションに関連付けられた状態情報が保存されるため、ウェブアプリケーションが、閉じられて、そのウェブアプリケーションに関する状態情報を復元またはリハイドレートするために後から再び開かれることが可能である。状態情報は、自動的に、または手動の選択プロセスを通じて保存されることが可能である。

【0114】

[00160]一例として、図26を考察されたい。そこでは、デスクトップ2600が、ウェブアプリケーションウィンドウ2602を含み、そのウェブアプリケーションウィンド

50

ウ 2 6 0 2 は、旅行を計画するために利用されている。デスクトップ 2 6 0 0 は、タスクバー 2 6 0 4 およびジャンプリスト 2 6 0 6 も含む。ウェブアプリケーションディレクトリー 2 6 0 8 が、ストレージ機能を提供し、このストレージ機能は、タスクセッション状態情報を格納するために利用されることが可能である。図示され説明された実施形態においては、ウェブアプリケーションディレクトリー 2 6 0 8 が、システムのユーザースペース内に作成される。この例においては、ユーザーは、保存されている 2 つのタスクセッション、すなわち、プエルトリコ旅行に関連付けられたタスクセッションと、アラスカ旅行に関連付けられたタスクセッションとを有する。

【0115】

[00161] オペレーションにおいて、ユーザーは、ウェブアプリケーションと対話するときに、タスクセッションを作成および保存することを、任意の適切な手段を通じて選択することができる。図示された例においては、ジャンプリスト 2 6 0 6 は、2 つのエントリーを含むメニューアイテム「Tasks (タスク)」を有する。第 1 のエントリー「New Task Session (新たなタスクセッション)」は、ユーザーが新たなタスクセッションを作成することを可能にする。第 2 のエントリー「Save Current Task (現在のタスクを保存)」は、ユーザーが現在のタスクを保存することを可能にする。現在のタスクを保存することによって、タスクに関連付けられた状態情報が、ウェブアプリケーションディレクトリー 2 6 0 8 に保存される。「Open Task Sessions (タスクセッションを開く)」と題されたメニューアイテムは、ウェブアプリケーションディレクトリー 2 6 0 8 に保存された前のタスクセッションをユーザーが復元またはリハイドレートすることを可能にするエントリーを含む。ここでは、2 つの前述した、前に保存されたタスクセッション、すなわちプエルトリコ旅行およびアラスカ旅行がある。

【0116】

[00162] 上記の例からわかるように、個々のウェブアプリケーションに関して複数のタスクセッションが作成および保存されることが可能である。タスクセッションが保存される場合には、ウェブアプリケーションに関連付けられたアプリケーション ID が、そのタスクセッションと共に保存されることが可能である。次いで、保存されたタスクセッションに関連付けられた情報をどのウェブアプリケーションが利用することになるかを特定するために、アプリケーション ID が使用されることが可能である。

【0117】

[00163] タスクセッションが作成および保存されることが可能にするために、任意の適切な技術およびアプローチが利用されることが可能である。少なくともいくつかの実施形態においては、本システムは、本システムのウェブブラウザーに関連付けられたクラッシュ回復システムを活用またはその他の形で利用することができる。この場合には、たとえばユーザーが、現在のタスクを保存すること、または新たなタスクセッションを作成することを選択すると、クラッシュ回復機能がトリガーされることが可能である。このクラッシュ回復機能は、「appdata」ファイルを作成することができ、この「appdata」ファイルは、ユーザーのアプリケーションデータディレクトリー内に存在し、タスクセッションに関連付けられた情報を保存するために使用されることが可能である。クラッシュ回復システムの具体的なオペレーションは、当業者によって理解されるであろう。したがって、簡潔にするために、そのようなシステムは、本明細書においては説明されない。

【0118】

[00164] 図 2 7 は、1 つまたは複数の実施形態による、タスクセッション状態情報を保存するための方法におけるステップを示す流れ図である。これらのステップは、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実行されることが可能である。少なくともいくつかの実施形態においては、この方法は、サーバ上で実行するソフトウェアと関連してクライアント上で実行するソフトウェアによって実行されることが可能である。

【0119】

[00165]ステップ2700は、タスクセッション状態情報を保存することに関連付けられた入力を受け取る。任意の適切な入力を受け取られることが可能である。たとえば、受け取られる入力は、ユーザーから受け取られる入力を含むことができる。代替として、または追加として、入力は、何らかのタイプの自動的な、プログラムによる入力を含むことができる。少なくともいくつかの実施形態においては、ユーザー入力は、ジャンプリストを介して受け取られることが可能である。代替として、または追加として、ユーザー入力は、ショートカットを介して受け取られることが可能である。図示され説明された実施形態においては、ユーザー入力は、ウェブアプリケーションに関連付けられたタスクセッション状態情報を保存したいとユーザーが望んでいることを示す。ステップ2702は、ウェブアプリケーションディレクトリー内のタスクセッションデータ構造を作成するか、またはその他の形でそのタスクセッションデータ構造にアクセスする。タスクセッションデータ構造は、タスクセッション状態情報を保存するために利用される。ステップ2704は、タスクセッション状態情報をデータ構造内に保存する。このステップは、任意の適切な方法で実行されることが可能である。たとえば、このステップは、ユーザーがタスクセッション状態情報を保存することを選択すると、実行されることが可能である。代替として、または追加として、このステップは、ウェブアプリケーションとユーザーの対話中に定期的に行われることが可能である。少なくともいくつかの実施形態においては、ステップ2702および2704は、ウェブブラウザのクラッシュ回復システムを利用することによって実行されることが可能である。しかし、特許請求される主題の趣旨および範囲から逸脱することなく、他の技術が利用されることが可能であるということを認識および理解されたい。

10

20

【0120】

[00166]図28は、1つまたは複数の実施形態による、状態情報が保存されているタスクセッションを復元するための方法におけるステップを示す流れ図である。これらのステップは、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実行されることが可能である。少なくともいくつかの実施形態においては、この方法は、サーバ上で実行するソフトウェアと関連してクライアント上で実行するソフトウェアによって実行されることが可能である。

【0121】

[00167]ステップ2800は、状態情報が前に保存されたタスクセッションを復元することに関連付けられたユーザー入力を受け取る。ステップ2802は、ウェブアプリケーションディレクトリー内のタスクセッションデータ構造にアクセスする。上述のように、タスクセッションの状態情報は、タスクセッションデータ構造内に保存される。ステップ2804は、ウェブアプリケーションディレクトリーからタスクセッション状態情報を検索する。ステップ2806は、関連付けられたウェブアプリケーションを起動し、ウェブアプリケーションディレクトリーから検索されたタスクセッション状態情報を使用してタスクセッションを復元する。

30

【0122】

[00168]タスクセッション状態情報を保存および再利用するという概念について説明してきたが、次いで、ウェブアプリケーションとブラウザの間においてどのようにして遷移が実行されることが可能であるかを考察されたい。

40

【0123】

ウェブアプリケーションとブラウザの間における遷移

[00169]1つまたは複数の実施形態においては、ウェブアプリケーションは、そのウェブアプリケーションを可能にするウェブアプリケーションモードブラウザによって提供されることが不可能であるブラウザ性能を活用できるようにするために、ブラウザ経験に遷移することができる。これは、いくつかの実施形態において、ウェブアプリケーションモードブラウザが、よりサイト固有の経験を開発者たちが提供できるようにするために、軽量化されたまたはクロムレスのブラウザであることに起因するということを想

50

起されたい。ウェブアプリケーションモードブラウザーから除外されているそのような他の性能としては、たとえば、お気に入り、ツールバー、および／またはその他のアドオンを含むことができる。

【0124】

[00170]少なくともいくつかの実施形態においては、個々のタブに関連付けられたコンテンツおよび状態が、ウェブアプリケーションからウェブブラウザーに移行されることが可能である。代替として、または追加として、複数のタブに関連付けられたコンテンツおよび状態、ならびに／またはウェブアプリケーションのすべてのコンテンツおよび状態が、ウェブアプリケーションからウェブブラウザーに移行されることも可能である。代替として、または追加として、個々のタブに関連付けられたセッションが、ウェブアプリケーションからブラウザーに移行されることも可能である。

10

【0125】

[00171]ユーザーが、自分のデスクトップからウェブアプリケーションを始動しており、その中でナビゲートしており、新たなタブにおいて1つまたは複数のリンクを開く状況を考えていただきたい。一例として、図29を考察されたい。そこでは、デスクトップ2900が、ウェブアプリケーションウィンドウ2902を含み、ウェブアプリケーションウィンドウ2902は、ウェブアプリケーションのうちで旅行を計画するために利用されている部分を含む。デスクトップ2900は、タスクバー2904も含み、ウェブアプリケーションは、そのタスクバーに固定されている関連付けられたアイコンをクリックすることなどによって、そのタスクバー2904から起動された。ウェブアプリケーションウィンドウ2902は、3つのタブ2906、2908、および2910を含む。この場合には、ユーザーは、タブ2906を選択しており、ユーザーが開くことができるリンク「Click here to search for flights（フライトを検索するためにはここをクリックしてください）」を有する。

20

【0126】

[00172]次いでユーザーは、関連付けられたリンクをクリックしてそのリンクを開き、そのリンクを開くと、タブ2906内に表示されているウェブサイトに関するお気に入りアイテムを作成したいと決心する、と想定する。この場合には、ユーザーは、ツールメニュー2912を開くか、またはその他の何らかのユーザーインターフェース手段を使用して、関連付けられたウェブブラウザー内にタブのコンテンツを開くというオプションを選択することができる。一例として、図30を考察されたい。図30は、図29と同様の番号を使用する。

30

【0127】

[00173]そこでは、ツールメニュー2912が、そのコンテンツ3000を表示するために開かれている。この例においては、ユーザーにとって、2つの選択、すなわち「Open Tab in Browser...（ブラウザー内にタブを開く...）」および「Open Web App Content in Browser...（ブラウザー内にウェブアプリケーションコンテンツを開く...）」が利用可能である。第1の選択は、ユーザーが、選択されたタブのコンテンツをウェブブラウザー内に開くことを可能にする。選択されたタブがウェブブラウザー内に開かれる場合には、そのタブのコンテンツおよび状態が、ウェブブラウザーに移行される。このウェブブラウザーは、オープンインスタンスを有するものとすることができ、または別法として、起動されるものとすることもできる。第2の選択は、ユーザーが、ウェブアプリケーションのすべてのコンテンツをウェブブラウザー内に開くことを可能にする。これが行われる場合には、そのウェブアプリケーションのコンテンツおよび状態が、ウェブブラウザーに移行される。

40

【0128】

[00174]コンテンツおよび状態をウェブアプリケーションタブまたはウェブアプリケーションからウェブブラウザーに移行するために、任意の適切な技術が利用されることが可能である。少なくともいくつかの実施形態においては、移行は、上述のようなウェブブラウザーのクラッシュ回復システムの使用を通じて生じる。具体的には、コンテンツおよび

50

状態をウェブアプリケーションからウェブブラウザに移行したいという希望をユーザーが示すと、コンテンツおよび状態は、たとえば、移行されることになる関連するデータを含む `appdata` ファイルを書き込むことによって、システムのディスクに書き込まれることが可能である。

【0129】

[00175] 加えて、少なくともいくつかの場合においては、通常はウェブブラウザのクラッシュ回復システムによって利用されない情報またはデータを移行するために、共有メモリーが利用されることが可能である。たとえば、証明書およびセッションクッキーなどのデータが共有メモリー内に格納されることが可能であり、そのようなデータがウェブブラウザによって使用されることを可能にするために、その共有メモリーが利用されることが可能である。

10

【0130】

[00176] ツールメニュー 2912 内に表示される特定のオプションをユーザーが選択すると、情報またはデータは、ウェブブラウザの最新のすなわち新たなインスタンスに移行されることが可能であり、ウェブアプリケーションウィンドウ 2902 内の関連付けられたタブは、閉じられることが可能である。1つまたは複数の実施形態においては、情報またはデータが移行されつつある元のタブが、ウェブアプリケーション内で開いている唯一のタブである場合には、ウェブアプリケーションは、移行が完了すると閉じられることが可能である。

【0131】

20

[00177] 上述のアプローチは、1つのプロセス境界をまたいで別々のプロセスにおいてウェブアプリケーションおよびウェブブラウザが実行する場合に功を奏する。すなわち、クラッシュ回復システムおよび共有メモリーを使用する移行は、プロセス境界をまたぐことに非常に適する。しかし、いくつかの場合には、プロセス境界を必ずしもまたぐ必要はない。むしろ、ウェブアプリケーションおよびその後続くウェブブラウザ機能は、同じプロセス内から表示されることが可能である。具体的には、この場合、ウェブブラウザユーザーインターフェースが、ウェブアプリケーションウィンドウ 2902 と共にインスタンス化されて使用されることが可能であり、ウェブアプリケーションモードブラウザを通じて利用できない機能がオンにされて、ウェブブラウザユーザーインターフェースを通じてアクセスできるようにされることが可能である。オペレーションにおいて、これを実施する1つの方法は、次のとおりである。ウェブアプリケーションは、はじめにいくつかのクラッシュ回復ファイルを作成する。新たなブラウザが始動され、それらのクラッシュ回復ファイルからクラッシュ回復情報をロードする。次いで、この情報は、その新たなブラウザの状態を構成するために使用される。ユーザーは、その新たなブラウザ内で作業するときには、そのブラウザの標準のユーザーインターフェースを介してブラウザ機能のすべてにアクセスできることになる。

30

【0132】

[00178] 図 31 は、一実施形態を示し、この実施形態においては、ユーザーが、1つのタブに関連付けられたコンテンツおよび状態を新たなブラウザインスタンスに移行することを選択している。図 29 の例と同様の番号が、利用されている。ここでは、ユーザーがタブ 2906 に関して「Open Tab in Browser (ブラウザ内にタブを開く)」のメニュー選択 (図 30) を選択していると想定する。それに応答して、そのタブのコンテンツおよび状態が、ウェブブラウザの新たなインスタンスに移行され、その関連付けられたユーザーインターフェースウィンドウが、3100 で示される。ユーザーインターフェースウィンドウ 3100 は、アドレスバー 3102 と、ウェブアプリケーションから移行されたタブに関連付けられたタブ 3104 とを含む。この例においては、ウェブアプリケーションウィンドウ 2902 においてタブ 2906 (図 30) が閉じられているが、複数のタブが開いているため、ウェブアプリケーションは開いたままであるということに留意されたい。

40

【0133】

50

[00179]図32は、1つまたは複数の実施形態による方法におけるステップを示す流れ図である。これらのステップは、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実行されることが可能である。少なくともいくつかの実施形態においては、この方法は、サーバ上で実行するソフトウェアと関連してクライアント上で実行するソフトウェアによって実行されることが可能である。

【0134】

[00180]ステップ3200は、ウェブアプリケーションのコンテンツおよび／または状態をウェブブラウザに移行することに関連付けられたユーザー入力を受け取る。任意の適切な入力を受け取られることが可能である。たとえば、少なくともいくつかの実施形態においては、入力は、ウェブアプリケーションによって表示されるツールメニューを通じて受け取られることが可能である。ステップ3202は、ウェブアプリケーションのコンテンツおよび／または状態をウェブブラウザに移行する。ウェブアプリケーションのコンテンツおよび／または状態を移行するために、任意の適切な技術が利用されることが可能である。加えて、任意の適切なレベルの細かさでコンテンツおよび／または状態が移行されることが可能である。たとえば、ウェブアプリケーションの個々のタブまたは複数のタブに関連付けられたコンテンツおよび／または状態が移行されることが可能である。代替として、または追加として、ウェブアプリケーションのコンテンツ全体が移行されることも可能である。さらに、少なくとも複数の実施形態においては、移行は、複数のプロセス境界をまたいで生じることが可能である。代替として、または追加として、移行は、同じプロセス内で生じることが可能である。

10

20

【0135】

[00181]図33は、1つまたは複数の実施形態による方法におけるステップを示す流れ図である。これらのステップは、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実行されることが可能である。少なくともいくつかの実施形態においては、この方法は、サーバ上で実行するソフトウェアと関連してクライアント上で実行するソフトウェアによって実行されることが可能である。

【0136】

[00182]ステップ3300は、ウェブアプリケーションのコンテンツおよび／または状態をウェブブラウザに移行することに関連付けられたユーザー入力を受け取る。任意の適切な入力を受け取られることが可能である。たとえば、少なくともいくつかの実施形態においては、入力は、ウェブアプリケーションによって表示されるツールメニューを通じて受け取られることが可能である。ステップ3302は、ウェブブラウザユーザーインターフェースをインスタンス化する。ステップ3304は、そのウェブブラウザユーザーインターフェースを使用して機能を表示する。機能を表示することは、ウェブブラウザユーザーインターフェースを通じたウェブアプリケーションコンテンツとの対話を可能にすることを含む。少なくともいくつかの実施形態においては、表示される機能は、ウェブアプリケーションまたはウェブアプリケーションモードブラウザを通じて利用できない機能で、かつウェブアプリケーションコンテンツと対話するために使用されることが可能である機能を含む。そのような機能の例は、上記で提供されている。図33の方法は、ウェブアプリケーションのコンテンツおよび／または状態の移行が同じプロセス内で生じる状況において役に立つことができる。

30

40

【0137】

[00183]図34は、1つまたは複数の実施形態による方法におけるステップを示す流れ図である。これらのステップは、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実行されることが可能である。少なくともいくつかの実施形態においては、この方法は、サーバ上で実行するソフトウェアと関連してクライアント上で実行するソフトウェアによって実行されることが可能である。

【0138】

[00184]ステップ3400は、ウェブアプリケーションのコンテンツおよび／または状態をウェブブラウザに移行することに関連付けられたユーザー入力を受け取る。任意の

50

適切な入力を受け取られることが可能である。たとえば、少なくともいくつかの実施形態においては、入力は、ウェブアプリケーションによって表示されるツールメニューを通じて受け取られることが可能である。ステップ 3402 は、ウェブアプリケーションコンテンツに関連付けられたデータを保存する。このステップは、任意の適切な方法で実行されることが可能である。たとえば、少なくともいくつかの実施形態においては、データのうちの少なくともいくつかは、システムのディスクに書き込まれることが可能である。データをシステムのディスクに書き込むために、任意の適切な技術が利用されることが可能である。たとえば、少なくともいくつかの実施形態においては、データをシステムのディスクに書き込むために、ウェブブラウザのクラッシュ回復システムが利用されることが可能である。さらに、少なくともいくつかの実施形態においては、ステップ 3402 は、ウェブアプリケーションとウェブブラウザとの間において共有される共有メモリーを使用することによって実行されることが可能である。

10

【0139】

[00185]ステップ 3404 は、ウェブブラウザが開いているかどうかを確認する。ウェブブラウザが開いている場合には、ステップ 3406 は、保存されたデータを使用して、ウェブアプリケーションコンテンツをウェブブラウザ内に提示する。その一方で、ウェブブラウザが開いていない場合には、ステップ 3408 は、ウェブブラウザを起動し、ステップ 3406 に戻って、保存されたデータを使用して、ウェブアプリケーションコンテンツを提示する。

【0140】

20

ブラウザからの一時的なウェブアプリケーションの作成

[00186]さまざまな実施形態が、1つまたは複数のいわゆる一時的なウェブアプリケーションの作成を可能にする。少なくともいくつかの実施形態においては、一時的なウェブアプリケーションをタスクバーに固定することなく、または上述のように一時的なウェブアプリケーションの関連付けられたファイルもしくは表示をクライアントのデスクトップにその他の形で統合することなく、一時的なウェブアプリケーションが作成されることが可能である。たとえば、一時的なウェブアプリケーションは、スタートメニュー上のショートカット、クライアントのタスクバー上のショートカットアイコンなど、その一時的なウェブアプリケーションが起動または再起動されることを可能にする方法を提供する、クライアントのデスクトップ上に統合されたユーザーインターフェース手段をまったく有さないことが可能である。しかし、そのような場合には、ユーザーは、一時的なウェブアプリケーションをその関連付けられたウェブサイトから起動することができ、そのウェブアプリケーションが上述のようにインストールまたは統合されていたならばそのウェブアプリケーションによって提供されることになるのと同じ機能にアクセスすることができる。一時的なウェブアプリケーションを使い終わると、その一時的なウェブアプリケーションは、ユーザーによって閉じられることが可能である。少なくともいくつかの実施形態においては、一時的なウェブアプリケーションが閉じられると、ユーザーは、もはやウェブアプリケーションのその特定のインスタンスにアクセスできなくなり、したがって、その一時的なウェブアプリケーションをクライアントのデスクトップツールバーまたはスタートメニューから再起動できなくする。これを行う1つの方法は、関連付けられたウェブサイトからウェブアプリケーションをユーザーが最初に起動したときにそのウェブアプリケーションに関して作成されたファイルまたはプロセスを削除することである。いくつかの実施形態においては、一時的なウェブアプリケーションは、インストールされたウェブアプリケーションに変換されることが可能であり、したがって、クライアントのデスクトップからそのウェブアプリケーションへのその後のアクセスを提供する。

30

40

【0141】

[00187]一例として、図35を考察されたい。図35は、一時的なウェブアプリケーションとブラウザの間における関係を示す。ここでは、ブラウザ3500が、タブ付きシステム (tabbed system) を通じた複数のウェブページへのアクセスを可能にし、タブ3510が、ウェブサイト「Any Search Page (任意の検索ページ)」に

50

関連付けられ、タブ 3 5 2 0 が、「A Second Open Page（第 2 の開いたページ）」に関連付けられる。1 つまたは複数の実施形態においては、一時的なウェブアプリケーションは、開いているウェブページから作成されることが可能である。図 3 5 においては、一時的なウェブアプリケーション 3 5 3 0 は、ユーザーがタブ 3 5 1 0 を選択してその選択をデスクトップ上のブラウザ 3 5 0 0 の外側にドラッグアンドドロップすることによって、生成または作成される。これが行われる場合、一時的なウェブアプリケーションファイルは、一時的な場所に作成されることが可能である。

【0 1 4 2】

[00188]しかし、特許請求される主題の趣旨および範囲から逸脱することなく、その他の方法で一時的なウェブアプリケーションが生成されることが可能であるということを認識および理解されたい。たとえばブラウザ 3 5 0 0 は、ウェブページの選択と、それに続く関連付けられた一時的なウェブアプリケーションの生成とを容易にするためのプルダウンメニューを有することができる。

10

【0 1 4 3】

[00189]本明細書の文脈においては、一時的なウェブアプリケーションは、本明細書において上述されている方法でクライアントのシステム上にインストールされないウェブアプリケーションである。しかしウェブサイトは、それでもなお、インストールされたウェブアプリケーションにおいて実行されることが可能であるのと同じ機能を一時的なウェブアプリケーションにおいても実行すること、およびそうした機能へのアクセスを提供することが可能である。たとえばウェブサイトは、一時的なウェブアプリケーションの独立したジャンプリストを修正すること、オーバーレイアイコンを設定およびクリアすることなどが可能である。代替として、または追加として、一時的なウェブアプリケーションは、上述のように、一時的なウェブアプリケーション内から開かれるタブまたはウィンドウの独立した集合を提供することなど、インストールされたウェブアプリケーションと同じ動作をサポートすることができる。

20

【0 1 4 4】

[00190]図 3 6 は、1 つまたは複数の実施形態による方法におけるステップを示す流れ図を示す。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実施されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、図 1 におけるような、クライアントデバイス上の適切に構成されたウェブブラウザおよび／またはソフトウェアモジュールによって実施されることが可能である。

30

【0 1 4 5】

[00191]ステップ 3 6 0 0 は、ウェブアプリケーションを取得できる元となるウェブサイトの選択を受け取る。これは、タブ付きウェブブラウザ上のタブの選択を通じて、プルダウンメニューを通じてなど、任意の適切な方法で達成されることが可能である。ウェブサイトの選択を受け取ると、ステップ 3 6 0 2 は、ウェブアプリケーションをクライアントデバイス上の一時的なウェブアプリケーションとして作成するための入力を受け取る。1 つまたは複数の実施形態においては、これは、ウェブアプリケーションを生成するために使用されることが可能である要求またはその他の情報を含むメッセージまたはコールを受け取ることを含むことができる。その他の実施形態においては、これは、ウェブサイトに関連付けられた何らかの表示をユーザーがドラッグアンドドロップすることから生成された入力を受け取ることを含むことができる。ステップ 3 6 0 4 は、選択されたウェブサイトに関連付けられる一時的なウェブアプリケーションを作成する。いくつかの実施形態においては、一時的なウェブアプリケーションを作成することによって、ウェブアプリケーションファイルおよび／またはプロセスが作成され、それらのウェブアプリケーションファイルおよび／またはプロセスまたは何らかの関連付けられた表示は、クライアントのデスクトップまたはスタートメニュー上に統合されない。たとえば、関連付けられたファイルは、統合されたウェブアプリケーションファイルが置かれる場所とは異なる一時的なファイルロケーションに保存されることが可能である。加えて、少なくともいくつかの

40

50

実施形態においては、一時的なウェブアプリケーションの作成は、ウェブサイトの状態をブラウザから一時的なウェブアプリケーションに転送することを含むことができる。

【0146】

[00192]図37は、1つまたは複数の実施形態による方法におけるステップを示す流れ図を示す。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実施されることが可能である。少なくともいくつかの実施形態においては、この方法の態様は、図1におけるような、適切に構成されたウェブブラウザおよび/またはソフトウェアモジュールによって実施されることが可能である。

【0147】

[00193]ステップ3700は、一時的なウェブアプリケーションを閉じるための入力を受け取る。このステップは、任意の適切な方法で実行されることが可能である。たとえば一実施形態においては、これは、開いている一時的なウェブアプリケーション上の閉じるボタンをユーザーが選択することから入力を受け取ることを含むことができる。別の実施形態においては、これは、一時的なウェブアプリケーションのプルダウンメニュー上の閉じるオプションをユーザーが選択することに基づいて入力を受け取ることを含むことができる。一時的なウェブアプリケーションを閉じるための入力を受け取ると、ステップ3702は、一時的なウェブアプリケーションを閉じる。一時的なウェブアプリケーションを閉じることは、一時的なウェブアプリケーションの関連付けられたファイルおよびプロセスを削除または除去することを含むことができる。したがって、一時的なウェブアプリケーションを閉じると、ユーザーは、上述のようにその一時的なウェブアプリケーションに再びアクセスするか、またはすぐ次に説明されるようにその一時的なウェブアプリケーションを一時的でないウェブアプリケーションとしてインストールしない限り、もはやその機能にアクセスすることはできない。

【0148】

[00194]一時的なウェブアプリケーションの作成および削除について説明してきたが、次いで、一時的なウェブアプリケーションが、1つまたは複数の実施形態に従って、固定されたまたはインストールされたウェブアプリケーションへとどのようにして変換されることが可能であるかについて考察されたい。

【0149】

インストールされたウェブアプリケーションへの一時的なウェブアプリケーションの変換

[00195]1つまたは複数の実施形態においては、一時的なウェブアプリケーションは、そのウェブアプリケーションが閉じられた後に永続的なアクセスを可能にするために、インストールされたまたは統合されたウェブアプリケーションへと変換されることが可能である。いったん変換されると、一時的なウェブアプリケーションは、一時的でないウェブアプリケーションと考えられることが可能である。

【0150】

[00196]図38は、ジャンプリストを含む一時的なウェブアプリケーションを示す。タスクバー3800が、クライアントデバイス上で開かれて動作している複数のプログラムを示す。一時的なウェブアプリケーション3810は、プログラム3840から生じたウェブアプリケーションである。一時的なウェブアプリケーション3810に関連付けられているのは、ジャンプリスト3820である。インストールされたまたは統合されたウェブアプリケーションの場合と同様に、ジャンプリスト3820は、インストールされたウェブアプリケーションに関連付けられたすべての潜在的な機能を有する。加えて、ジャンプリスト3820は、「Pin this program to taskbar (このプログラムをタスクバーに固定する)」と題されたアイテム3830を含む。このオプションを選択することは、ウェブアプリケーションをクライアントのタスクバーに固定し、ひいては、そのウェブアプリケーションおよび関連付けられたユーザーインターフェース手段を上述のようにクライアントのデスクトップ上にインストールする。これは、今や一時的ではなくなったウェブアプリケーションがデスクトップから再起動されることを可

能にする。しかし、特許請求される主題の趣旨および範囲から逸脱することなく、一時的なウェブアプリケーションを一時的でないウェブアプリケーションに変換するために、任意の適切な技術が採用されることが可能であるということを認識および理解されたい。たとえば、いくつかの実施形態においては、一時的なウェブアプリケーションは、そのウェブアプリケーションを統合およびインストールするためにクライアントのデスクトップのスタートメニューに追加されることが可能である。別の実施形態においては、一時的なウェブアプリケーションは、インストールプロセスを開始するオプションを伴うプルダウンメニューを有することができる。言うまでもなく、一時的なウェブアプリケーションを一時的でないウェブアプリケーションへと変換するために、多くの方法が利用されることが可能である。

10

【0151】

ウェブアプリケーションスーパーホームボタン

[00197]ウェブアプリケーションと対話しているときに、ユーザーは、そのウェブアプリケーションが関連付けられているウェブサイトに直接関連付けられたドメイン以外のドメインへナビゲートすることが可能である。たとえばユーザーは、電子メールウェブアプリケーションを始動してから、外部リンクを辿って、ニュース、ショッピング、または娯楽のサイトなどの別のサイトに至ることができる。

【0152】

[00198] 1つまたは複数の実施形態においては、ウェブアプリケーションホームボタンが、ユーザーインターフェース経験の一部として提供される。ウェブアプリケーションホームボタンは、結合目的を果たす。第1に、ウェブアプリケーションホームボタンは、特定のウェブアプリケーションモードブラウザー（「サイトモードブラウザー（site mode browser）」とも呼ばれる）インスタンスの目的が、関連付けられたウェブアプリケーションのためであるということを示す。ウェブアプリケーションホームボタンは、この情報を伝達するために、ブランド化およびその他の視覚的な手段を使用することができる。第2に、ウェブアプリケーションホームボタンは、ユーザーが、スタートURLにアクセスするためにウェブアプリケーションホームボタンをクリックするだけで、自分のウェブアプリケーション経験の起点に迅速かつ容易に戻ってスタートし直すことを可能にする。これは、関連付けられたウェブサイトに関するスタートURLにアクセスするために特定のウェブアプリケーションを閉じてから再起動しなければならない状況を緩和する。少なくともいくつかの実施形態においては、デフォルトで、スタートURLに関連付けられた値が、ユーザーがファビコンをタスクバー上にドラッグアンドドロップする元となるページから確認される。あるいはウェブ開発者は、スタートURLを記述するHTMLタグを自分のページの一部として定義することもできる。これは、ウェブ開発者が、自分が現在閲覧しているページと同じではない代替スタートURLを定義することを可能にする。

20

30

【0153】

[00199]一例として、図39を考察されたい。そこでは、ウェブアプリケーションモードブラウザー3900が、アドレスバー3902を含み、アドレスバー3902の中には、ウェブサイトに関するURLが現れている。加えて、ウェブアプリケーションホームボタン3904が、戻るおよび進むナビゲーションボタンの隣に現れている。ユーザーは、ウェブアプリケーションに関連付けられたウェブサイトの外側のドメインへナビゲートする際、いつでも、ウェブアプリケーションホームボタン3904をクリックするだけで、ウェブアプリケーションファイル内に記述されているウェブサイトのスタートURLへナビゲートすることができる。

40

【0154】

[00200]さらに、少なくともいくつかの実施形態においては、ナビゲーションの戻るボタンおよび進むボタンは、デフォルトブラウザーではなくウェブアプリケーション内の自らのコンテキストをユーザーに伝えるために、ウェブアプリケーションホームボタンを通じてウェブサイトのブランドの原色を利用または抽出することによって、そのサイトの独自性を帯びることができる。加えて、ウェブサイトがこれらのボタンの色を自らのHTML

50

L ページの一部として指定することを可能にするために、H T M L タグが利用されることが可能である。

【0155】

[00201]図40は、1つまたは複数の実施形態による方法におけるステップを示す流れ図である。この方法は、任意の適切なハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せと共に実施されることが可能である。少なくともいくつかの実施形態においては、この方法は、上述のようなサイトモードブラウザーによって実施されることが可能である。

【0156】

[00202]ステップ4000は、クライアントデバイス上にインストールされるウェブアプリケーションに関連付けられたウェブサイトのスタートURLへサイトモードブラウザーをナビゲートする。このステップは、任意の適切な方法で実行されることが可能であり、そうした方法の例は、上記で提供されている。ステップ4002は、別のウェブページへナビゲートする。このウェブページは、スタートURLに関連付けられたウェブサイトに関連付けられることもあり、または関連付けられないこともある。ステップ4004は、ウェブアプリケーションホームボタンの選択を受け取る。ウェブアプリケーションホームボタンの選択を受け取ったことに応答して、ステップ4006は、サイトモードブラウザーをウェブサイトのスタートURLへナビゲートする。

【0157】

例示的なシステム

[00203]図41は、上述のさまざまな実施形態を実装するために使用されることが可能である例示的なコンピューティングデバイス4100を示す。コンピューティングデバイス4100は、たとえば図1のコンピューティングデバイス102、またはその他の任意の適切なコンピューティングデバイスとすることができる。

【0158】

[00204]コンピューティングデバイス4100は、1つまたは複数のプロセッサまたは処理装置4102と、1つまたは複数のメモリーおよび／またはストレージコンポーネント4104と、1つまたは複数の入力／出力（I／O）デバイス4106と、さまざまなコンポーネントおよびデバイスどうしが相互に通信することを可能にするバス4108を含む。バス4108は、メモリバスまたはメモリコントローラと、ペリフェラルバスと、アクセラレイティッドグラフィックスポートと、さまざまなバスアーキテクチャーのうちのいずれかを使用するプロセッサまたはローカルバスとを含むいくつかのタイプのバス構造のいずれかのうちの1つまたは複数に相当する。バス4108は、有線バスおよび／またはワイヤレスバスを含むことができる。

【0159】

[00205]メモリー／ストレージコンポーネント4104は、1つまたは複数のコンピューター記憶媒体に相当する。コンポーネント4104は、揮発性の媒体（ランダムアクセスメモリー（RAM）など）、および／または不揮発性の媒体（読み取り専用メモリー（ROM）、フラッシュメモリー、光ディスク、磁気ディスクなど）を含むことができる。コンポーネント4104は、固定式の媒体（たとえば、RAM、ROM、固定式のハードドライブなど）、ならびに取り外し可能な媒体（たとえば、フラッシュメモリードライブ、取り外し可能なハードドライブ、光ディスクなど）を含むことができる。

【0160】

[00206]1つまたは複数の入力／出力デバイス4106は、ユーザーがコマンドおよび情報をコンピューティングデバイス4100に入力することを可能にし、また、情報がユーザーおよび／またはその他のコンポーネントもしくはデバイスに提示されることが可能にする。入力デバイスの例は、キーボード、カーソルコントロールデバイス（たとえばマウス）、マイクロフォン、スキャナなどを含む。出力デバイスの例は、ディスプレイデバイス（たとえば、モニターまたはプロジェクター）、スピーカー、プリンター、ネットワークカードなどを含む。

【0161】

[00207]さまざまな技術が、本明細書ではソフトウェアモジュールまたはプログラムモジュールという一般的なコンテキストにおいて説明されることが可能である。一般にソフトウェアは、特定のタスクを実行するかまたは特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、コンポーネント、データ構造などを含む。これらのモジュールおよび技術の実装形態は、何らかの形態のコンピューター可読媒体上に格納されることが、またはそうしたコンピューター可読媒体を介して伝送されることが可能である。コンピューター可読媒体は、コンピューティングデバイスによってアクセスされることが可能である任意の利用可能な1つまたは複数の媒体とすることができる。限定ではなく例として、コンピューター可読媒体は、「コンピューター可読記憶媒体」を含むことができる。

10

【0162】

[00208]「コンピューター可読記憶媒体」は、コンピューター可読命令、データ構造、プログラムモジュール、またはその他のデータなどの情報を格納するための任意の方法またはテクノロジーにおいて実装される揮発性の媒体および不揮発性の媒体、ならびに取り外し可能な媒体および取り外し不能な媒体を含む。コンピューター可読記憶媒体は、RAM、ROM、EEPROM、フラッシュメモリー、もしくはその他のメモリーテクノロジー、CD-ROM、デジタル多用途ディスク(DVD)、もしくはその他の光ストレージ、磁気カセット、磁気テープ、磁気ディスクストレージ、もしくはその他の磁気ストレージデバイス、または所望の情報を格納するために使用されることが可能で、コンピューターによってアクセスされることが可能であるその他の任意の媒体を含むが、それらには限

20

【0163】

結論

[00209]さまざまな実施形態が、エンドユーザーがウェブアプリケーションおよびウェブサイトを自分のデスクトップ上にインストールできるようにするメカニズムを提供する。1つまたは複数の実施形態によれば、ウェブサイトに関連付けられた開発者が、ユーザー対話に関連付けられた境界を定義すること、およびそれらの境界をランタイムエンジンによって施行させることを可能にするために、クライアント側コードが利用されることが可能である。少なくともいくつかの実施形態においては、開発者は、JavaScript（登録商標）コードを通じて、スタートメニューショートカットの作成、ナビゲーション、およびいわゆるジャンプリスト統合、ならびにその他のさまざまな機能のためのさまざまな構成を提供することができる。

30

【0164】

[00210]本主題は、構造的な特徴および／または方法論的な行為に特有の言葉で説明されたが、添付の特許請求の範囲において定義される本主題は、上述の特定の特徴または行為に必ずしも限定されるものではないということを理解されたい。むしろ、上述の特定の特徴および行為は、特許請求の範囲を実施する例示的な形態として開示されている。

【図 6】

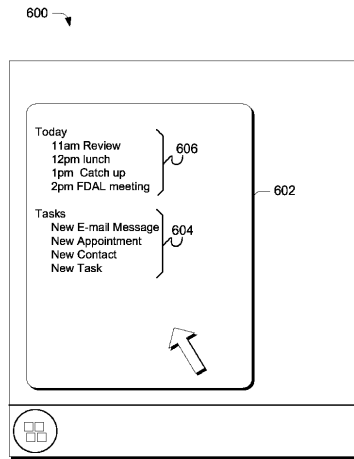


Fig. 6

【図 10】

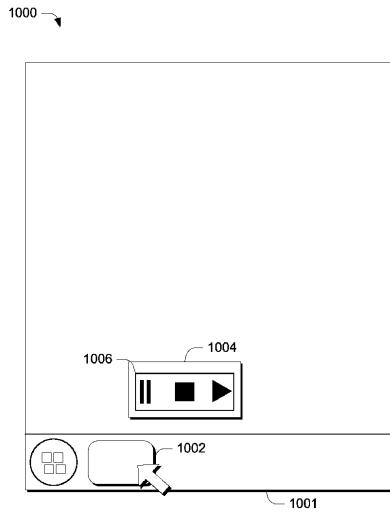


Fig. 10

【図 14】

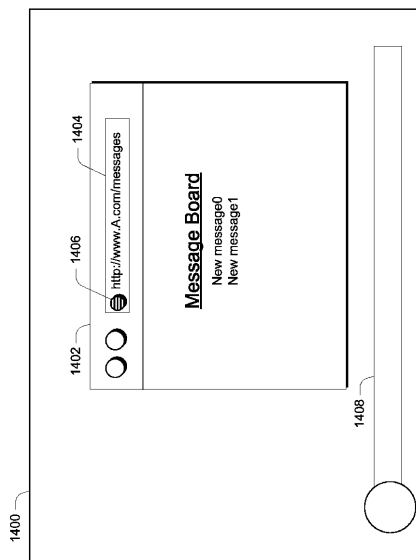


Fig. 14

【図 15】

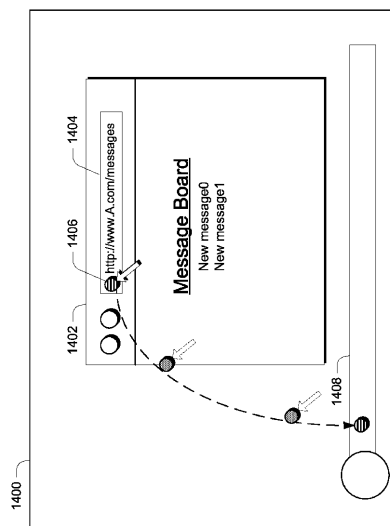


Fig. 15

【図 1 7】

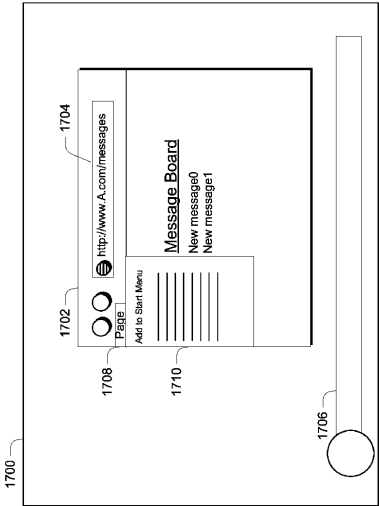


Fig. 17

【図 1 9】

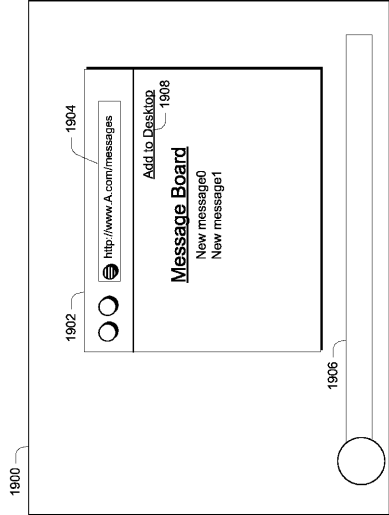


Fig. 19

【図 2 1】

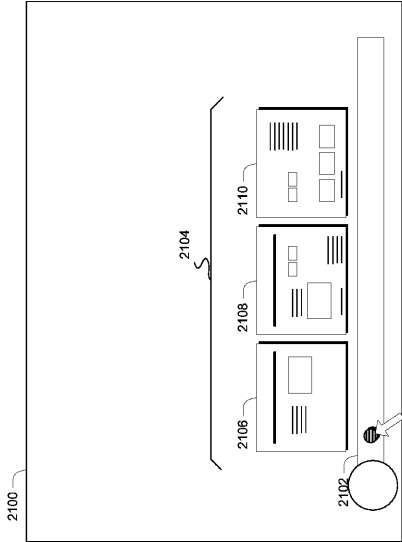


Fig. 21

【図 2 9】

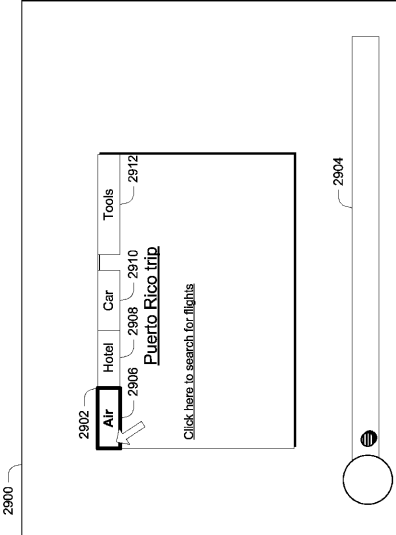


Fig. 29

【図 30】

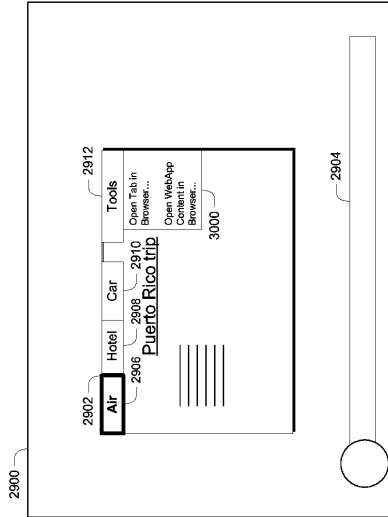


Fig. 30

【図 31】

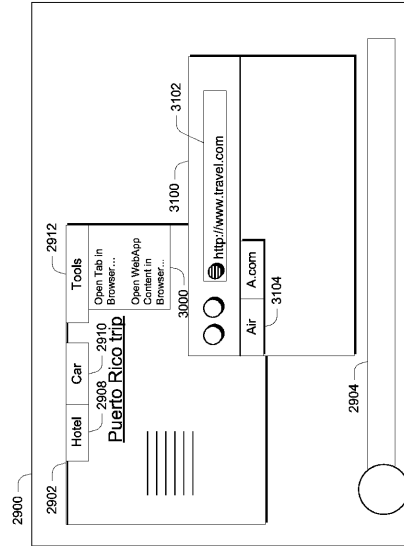


Fig. 31

【図 35】

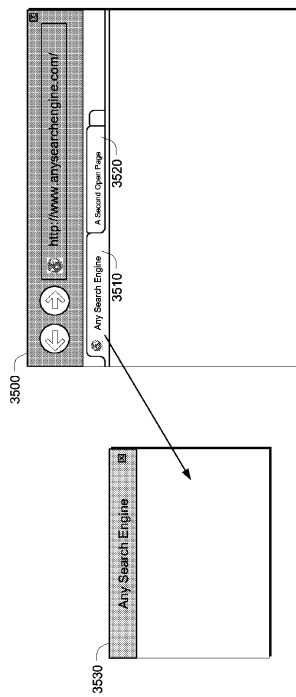


Fig. 35

【図 38】

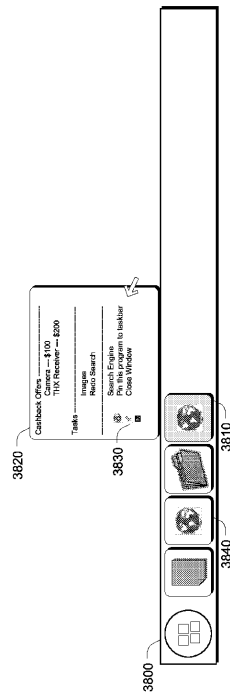
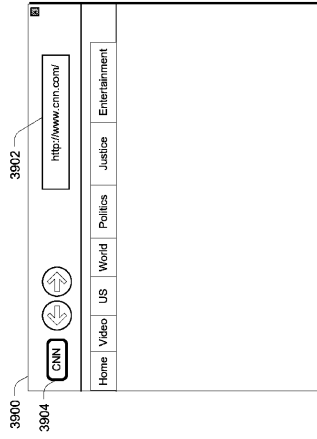
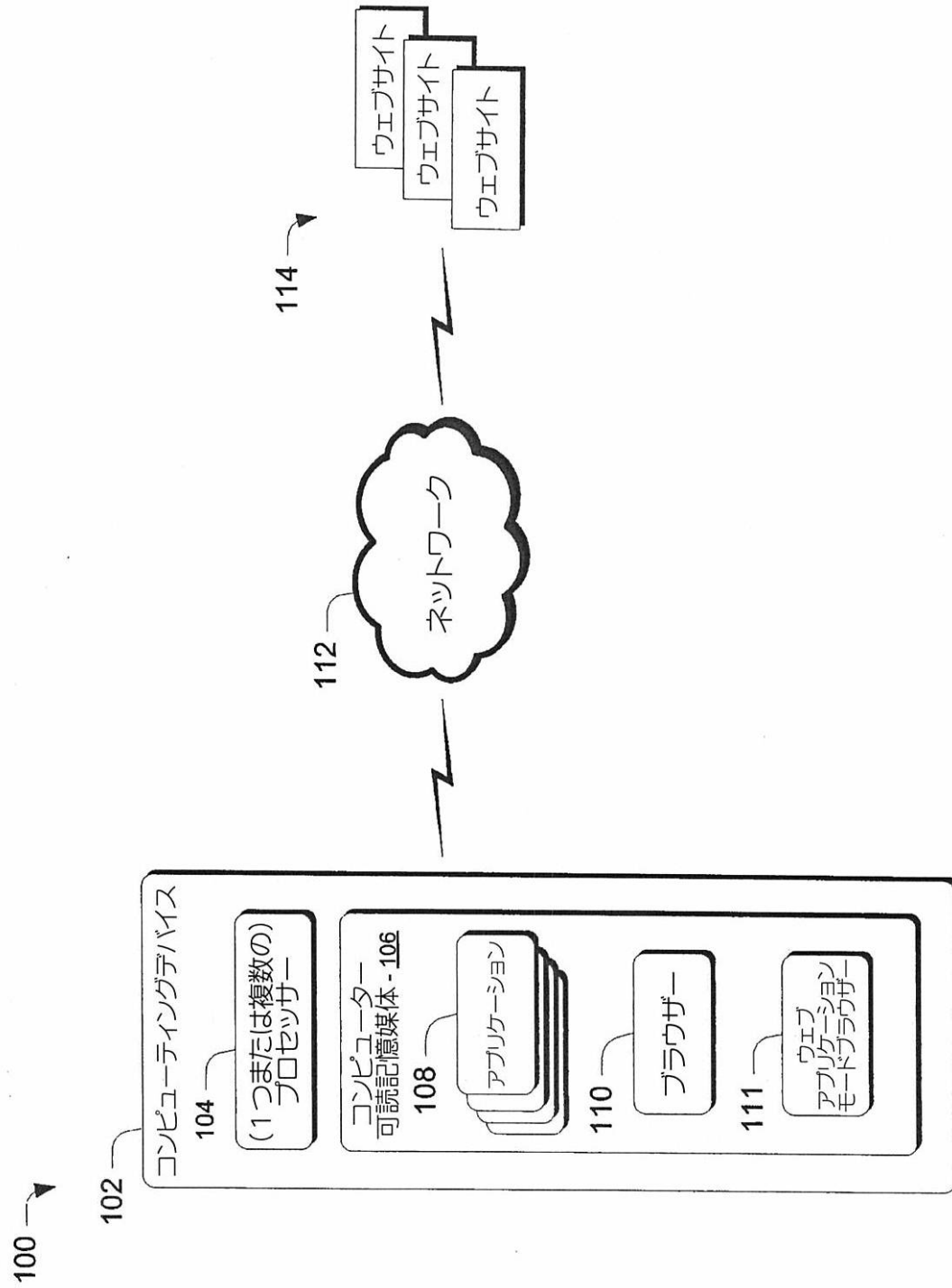


Fig. 38

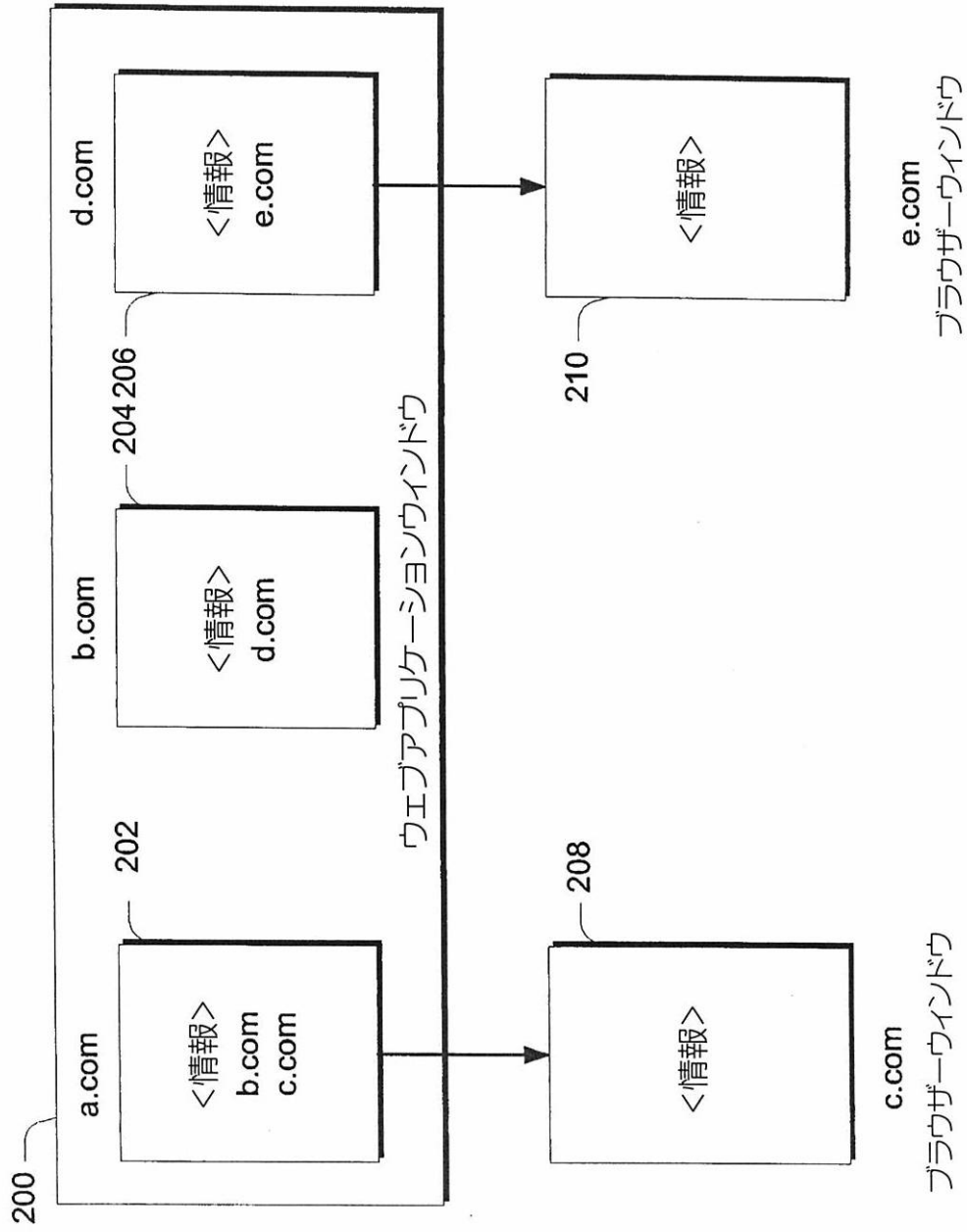
【図 39】

**Fig. 39**

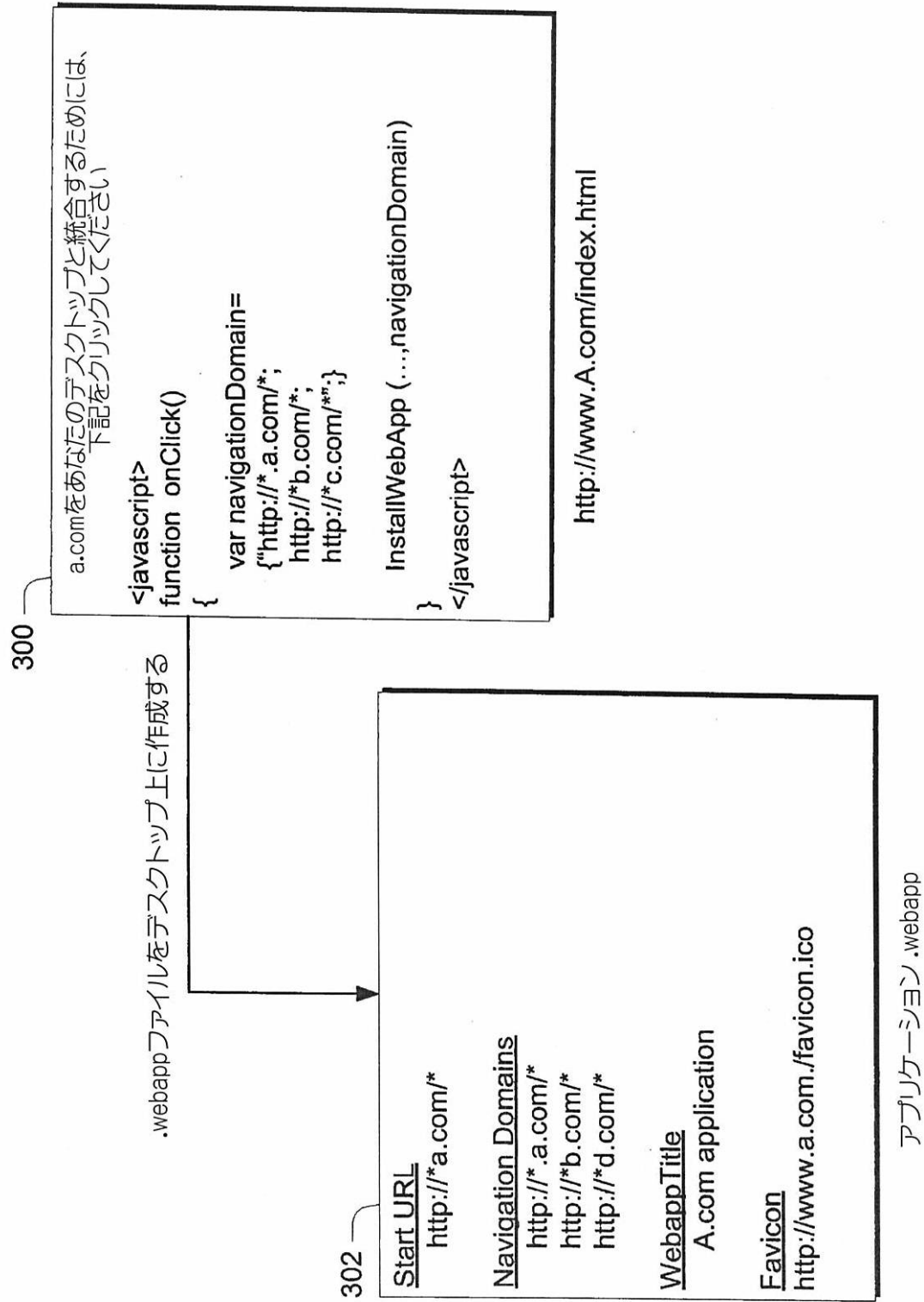
【図 1】



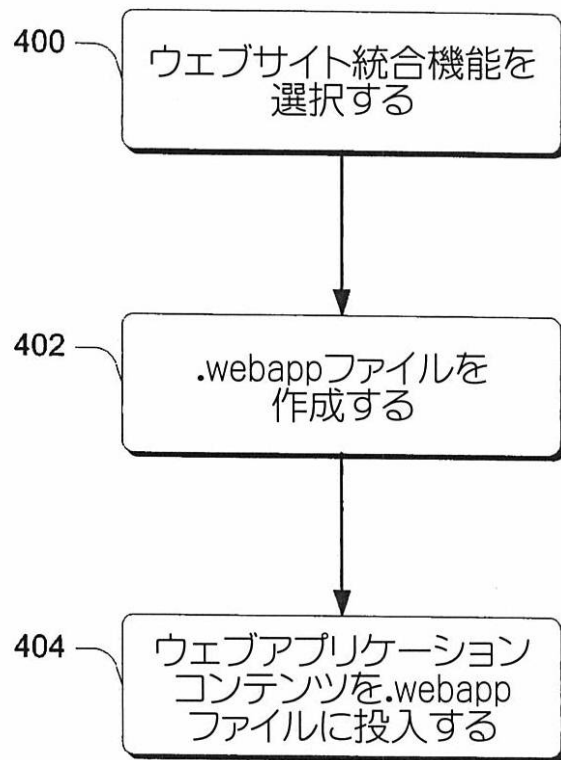
【図 2】



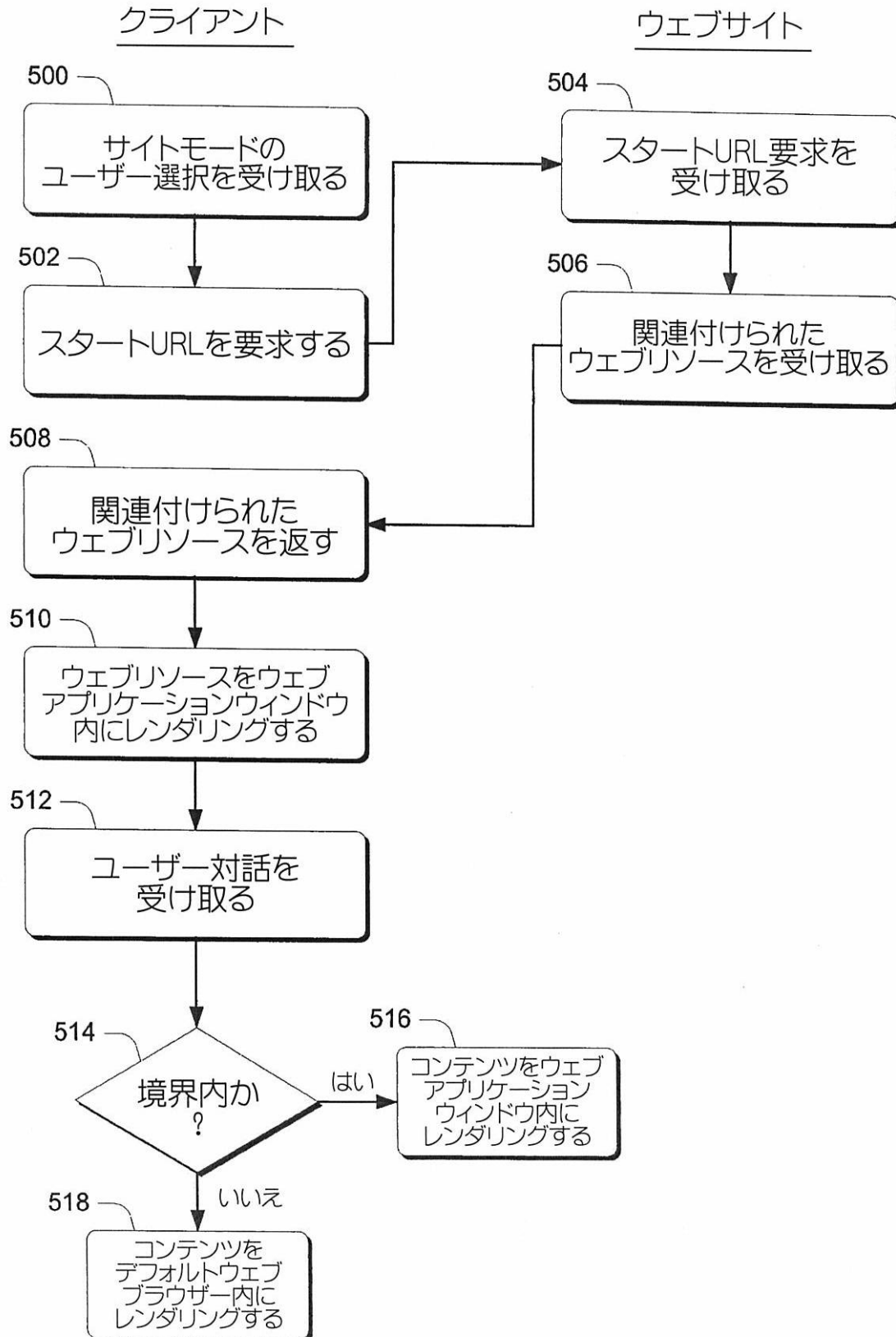
【図 3】



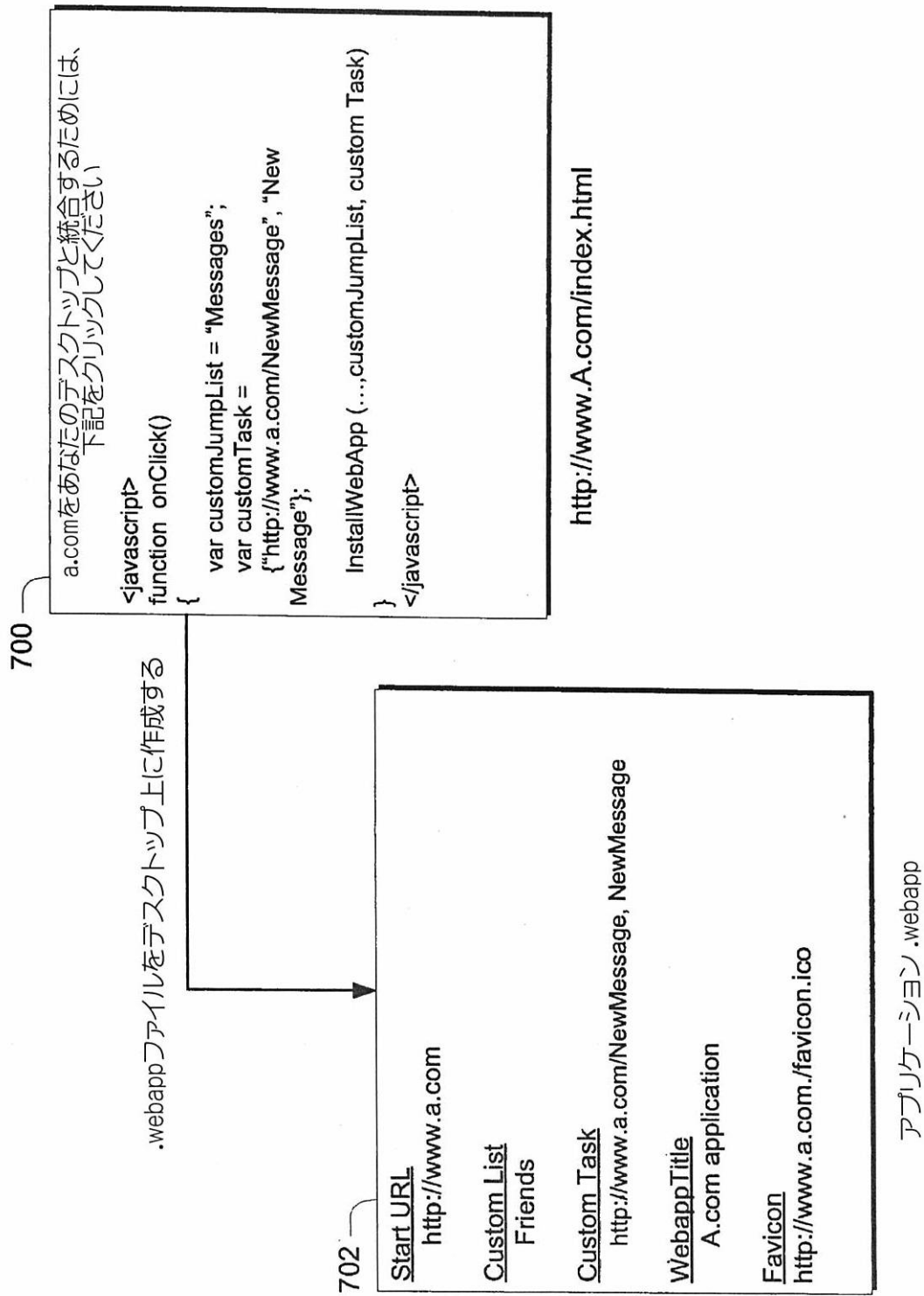
【図 4】



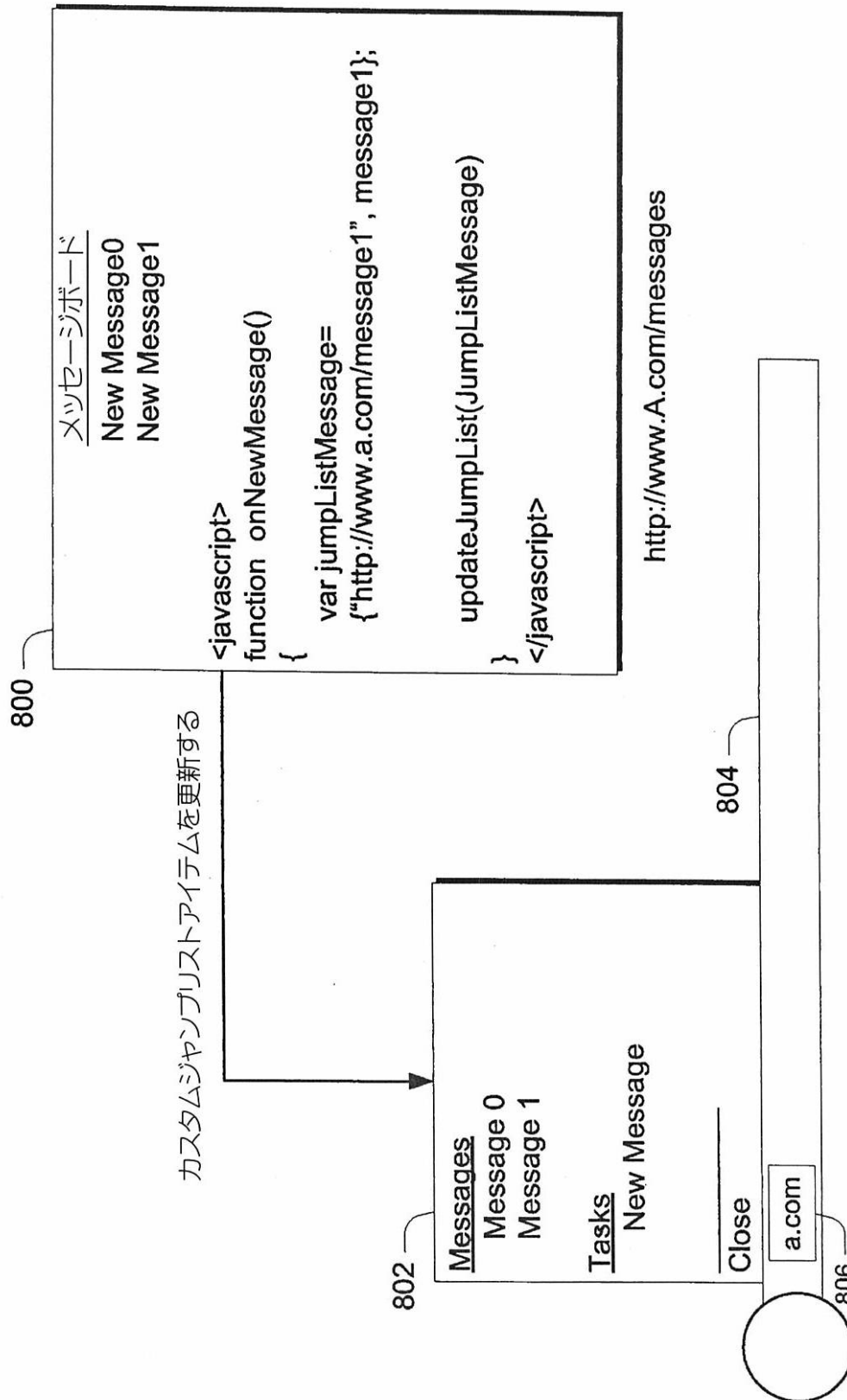
【図 5】



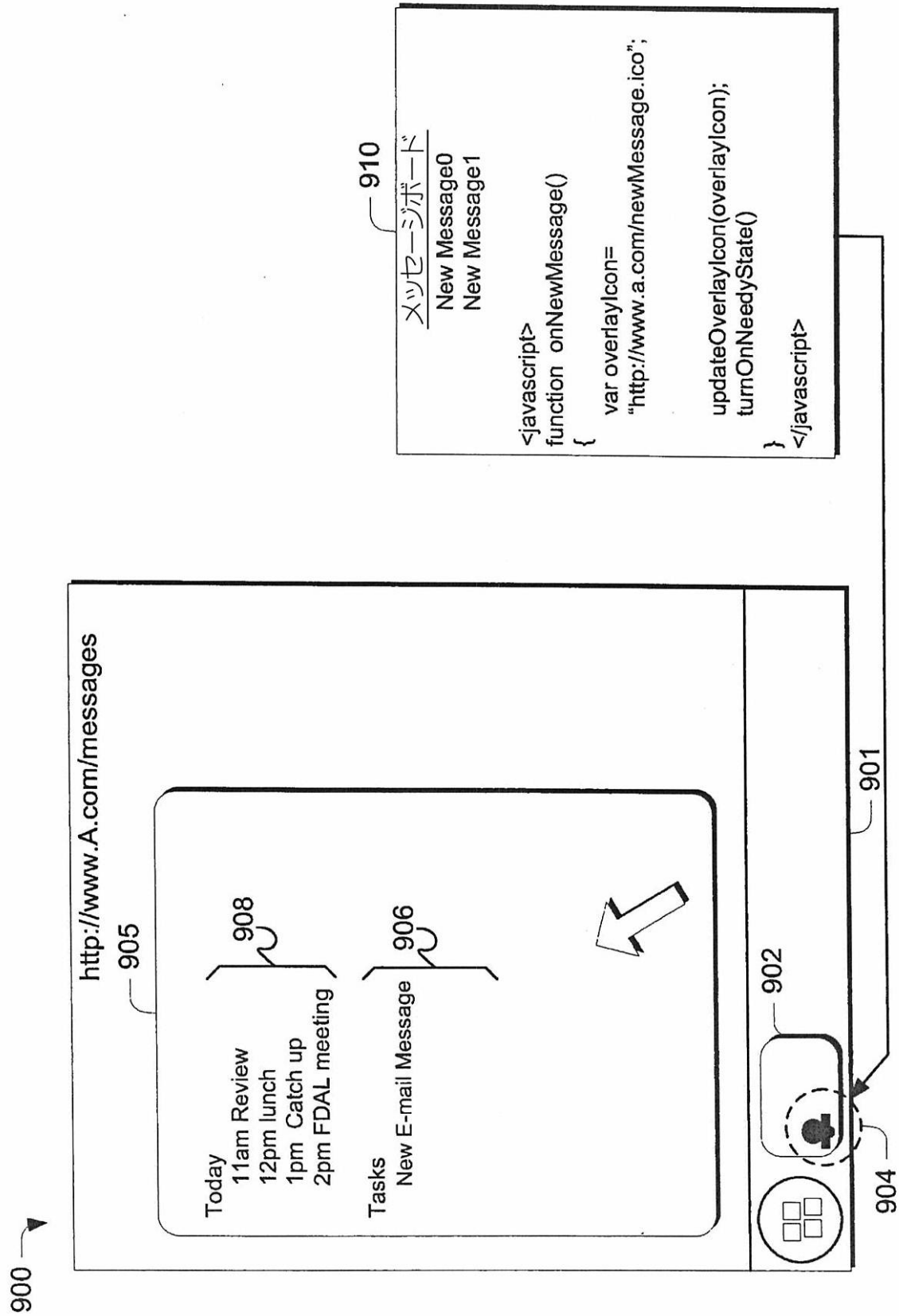
【図 7】



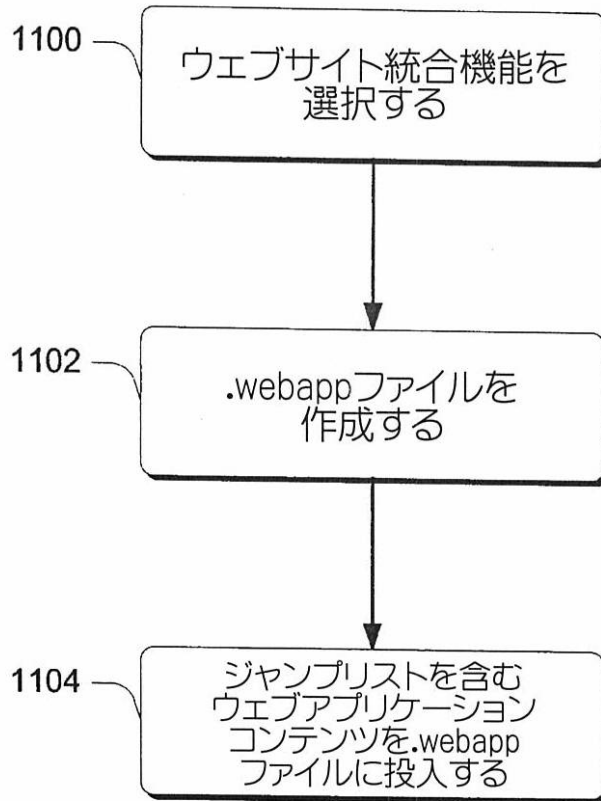
【図 8】



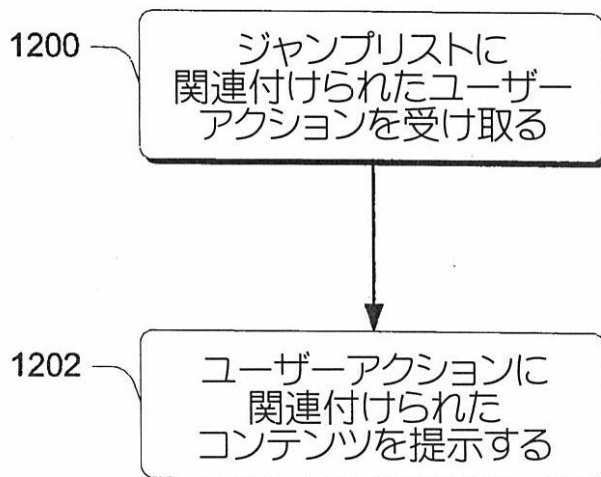
【図 9】



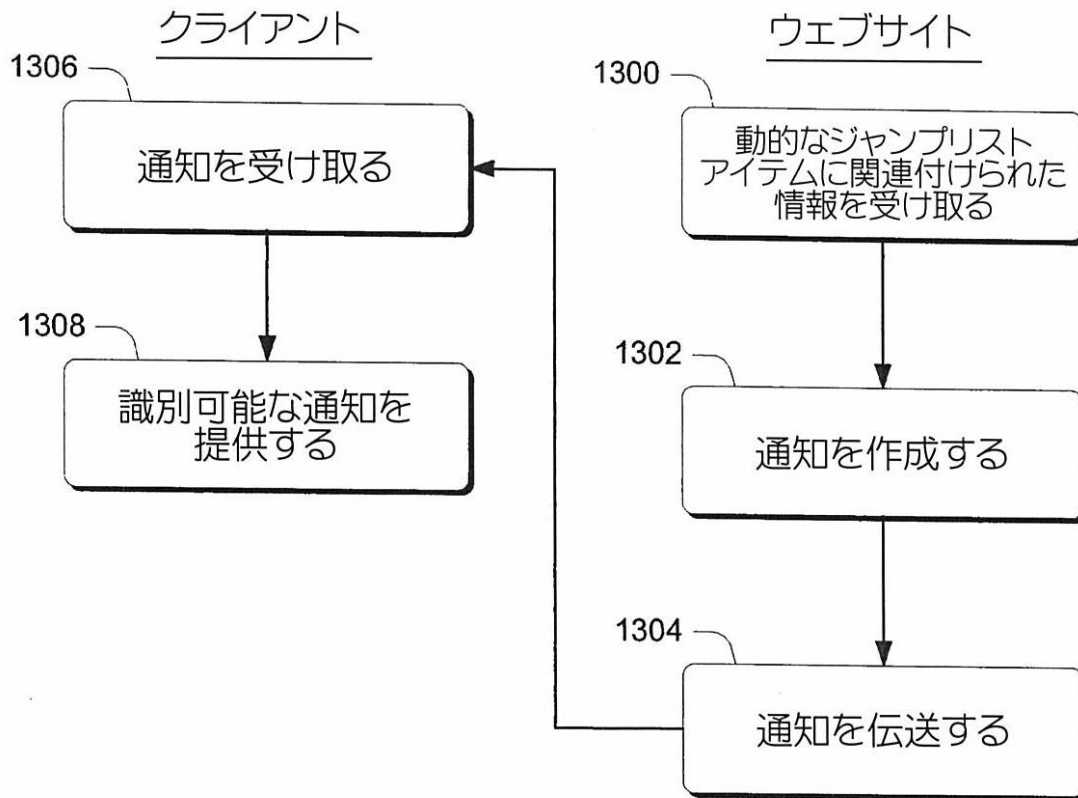
【図 1 1】



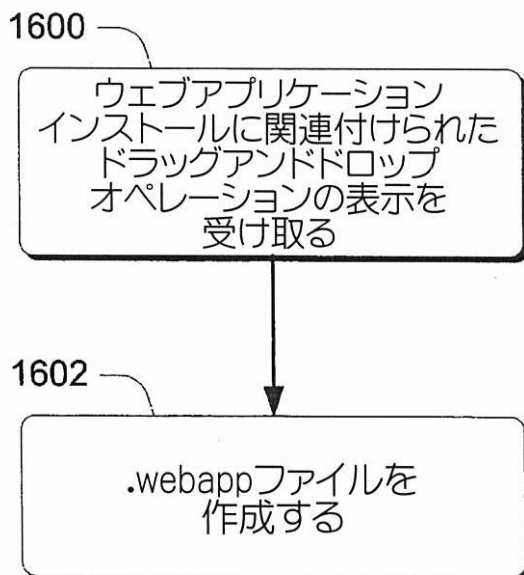
【図 1 2】



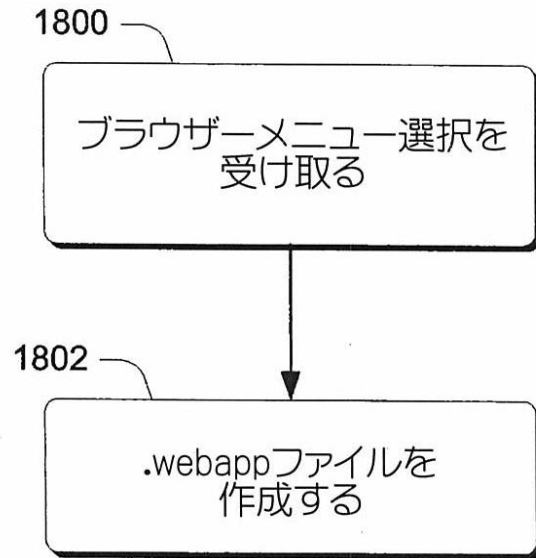
【図 13】



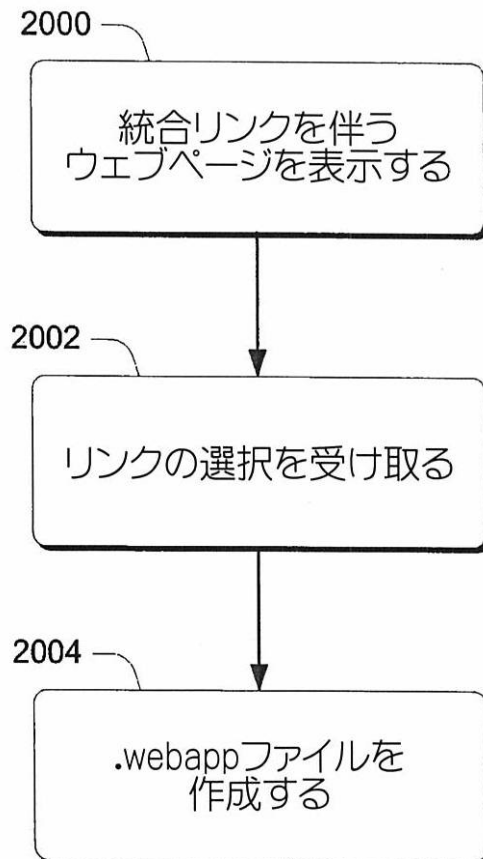
【図 16】



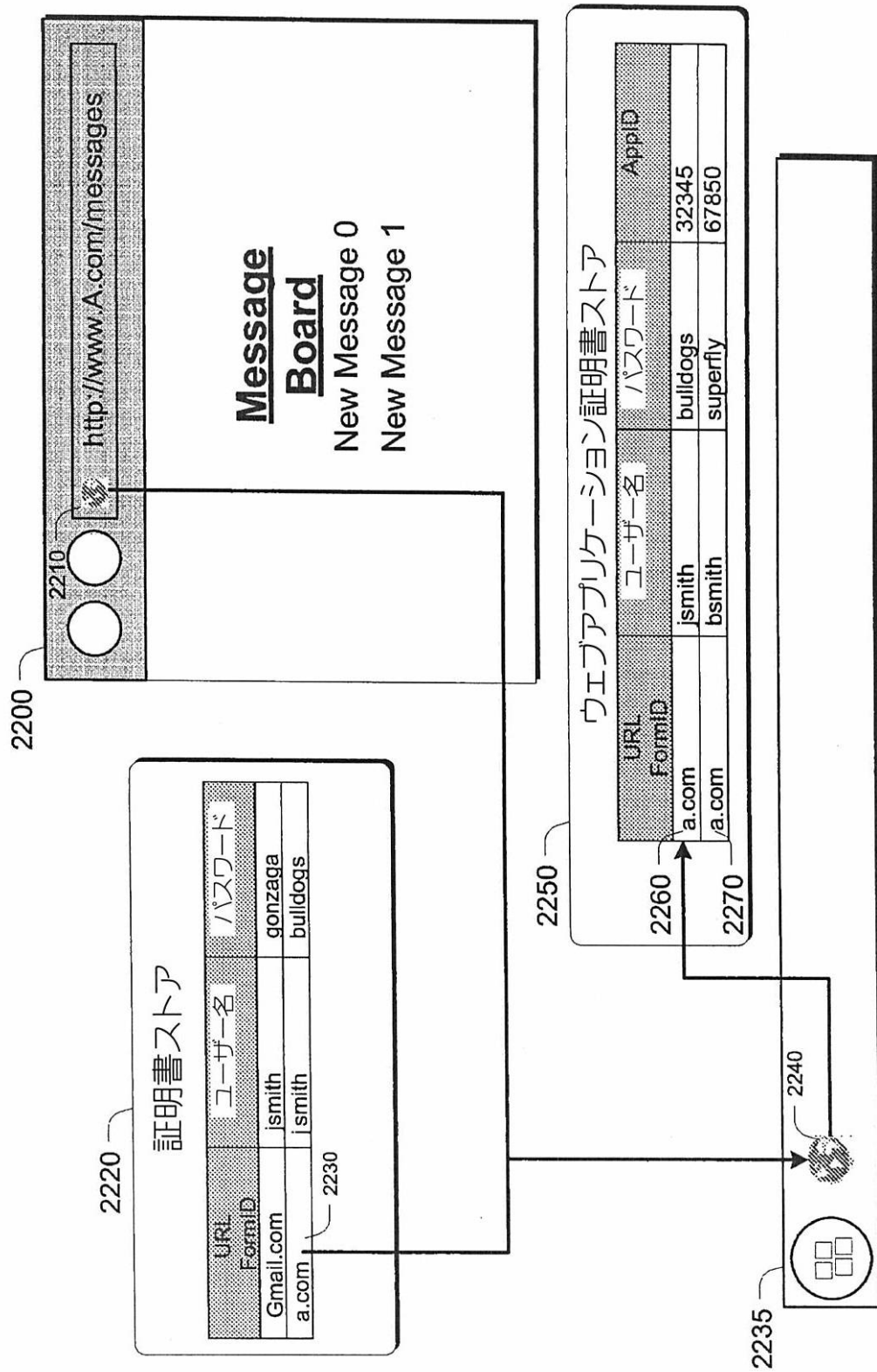
【図 18】



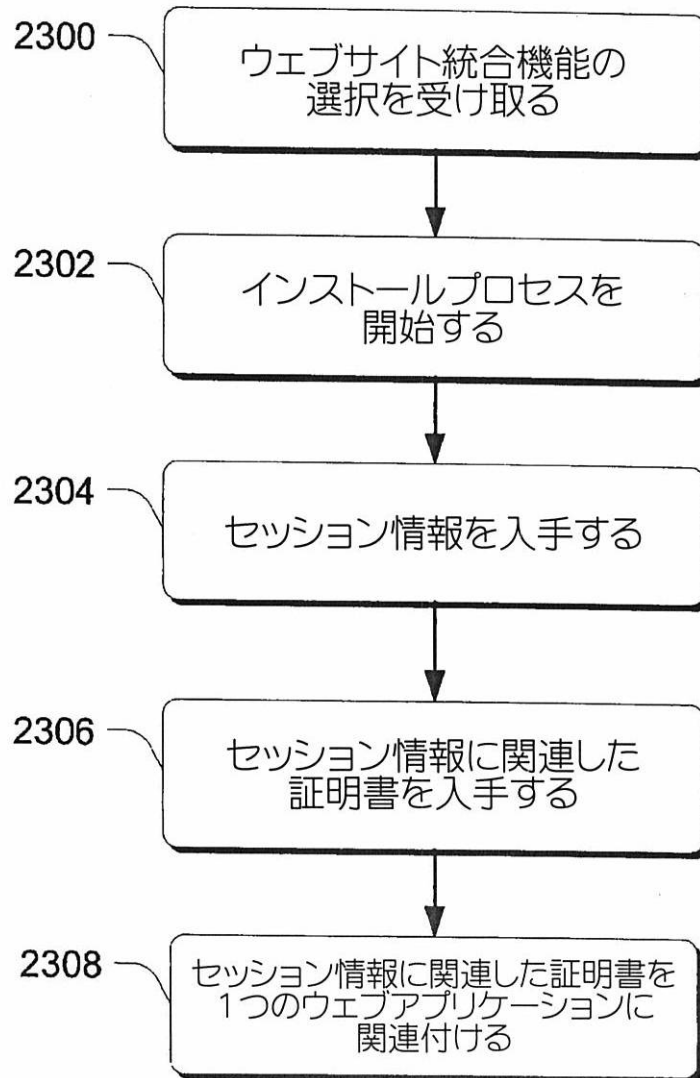
【図 20】



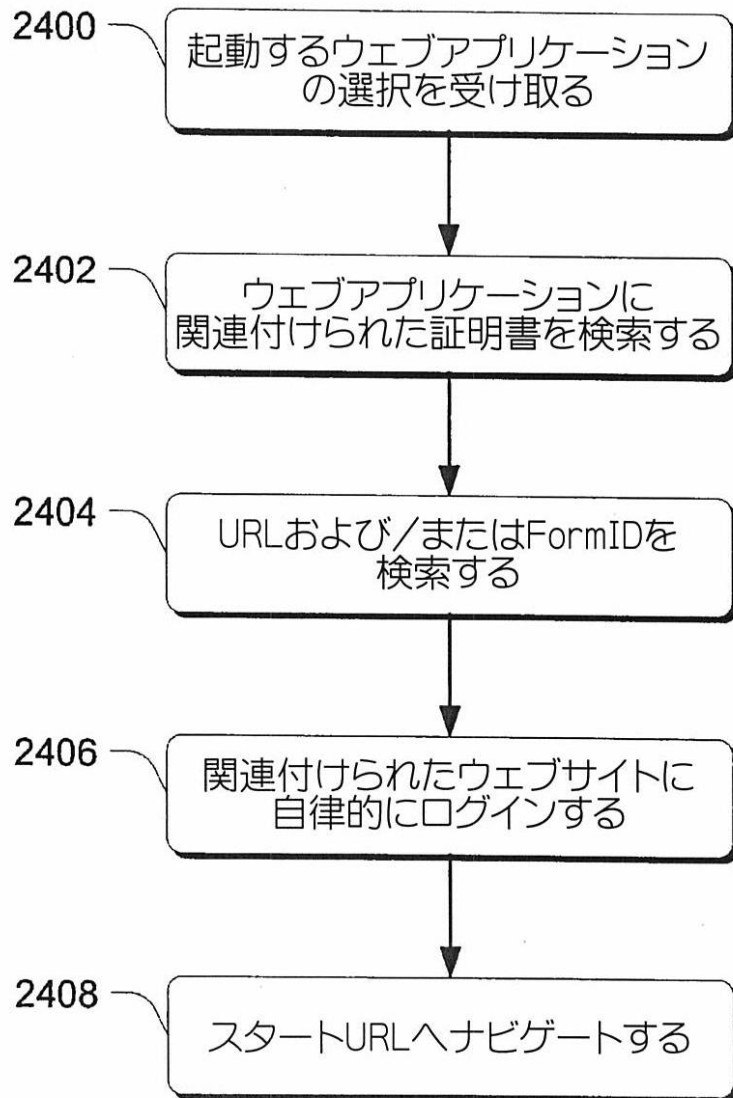
【図 22】



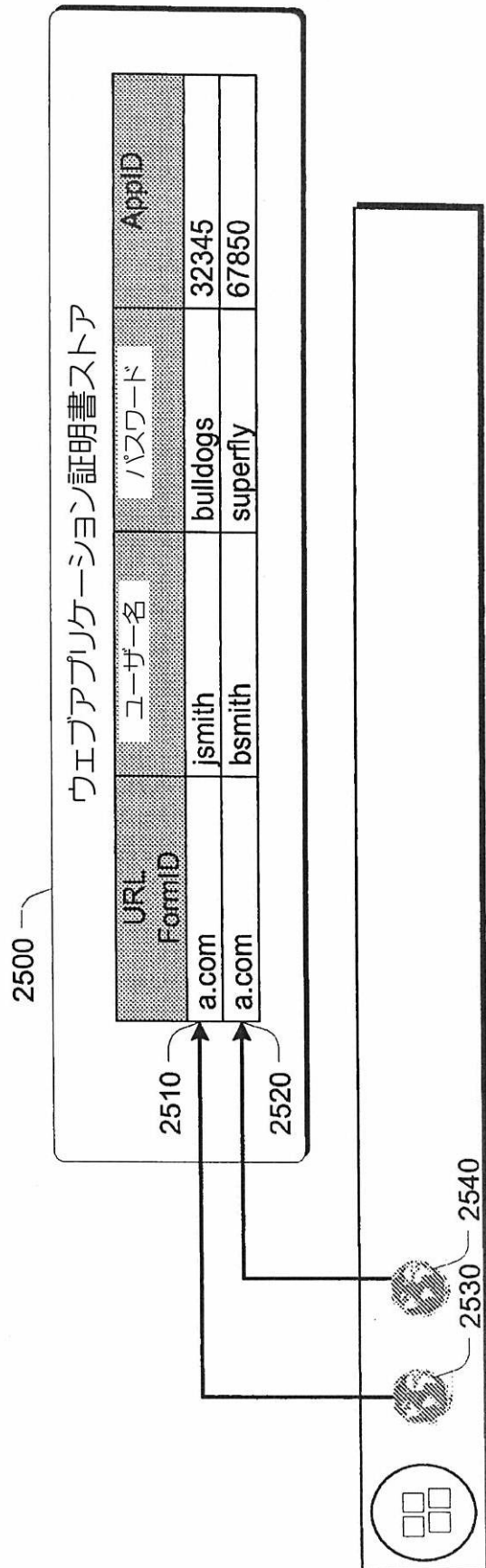
【図 2 3】



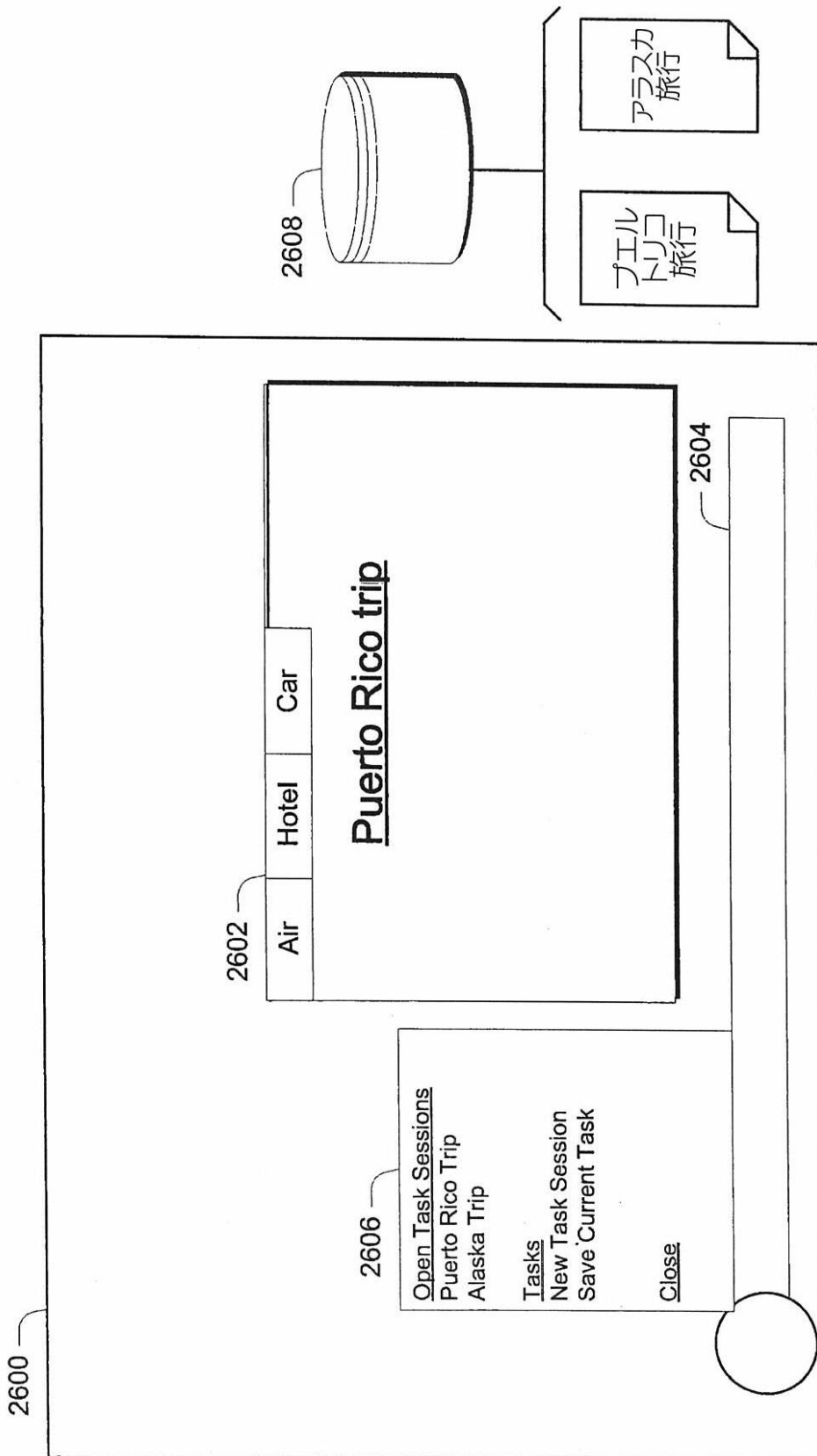
【図 24】



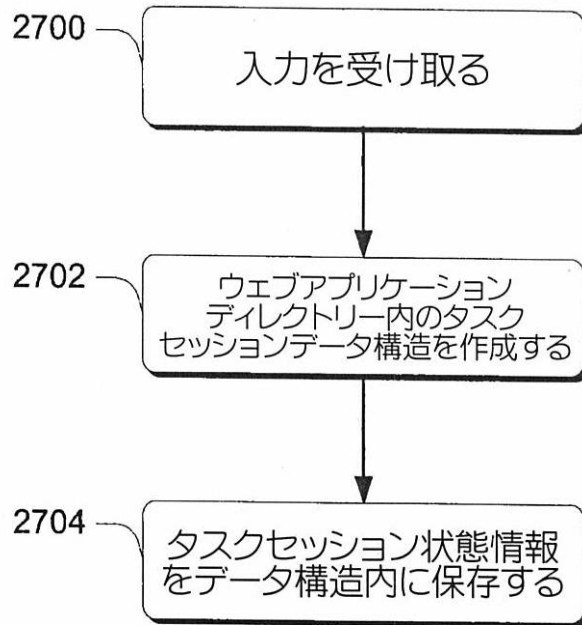
【図 25】



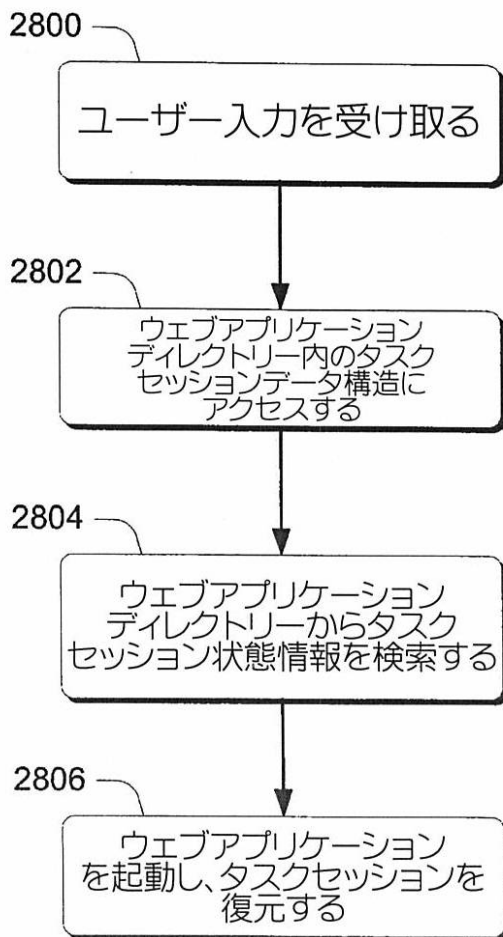
【図 26】



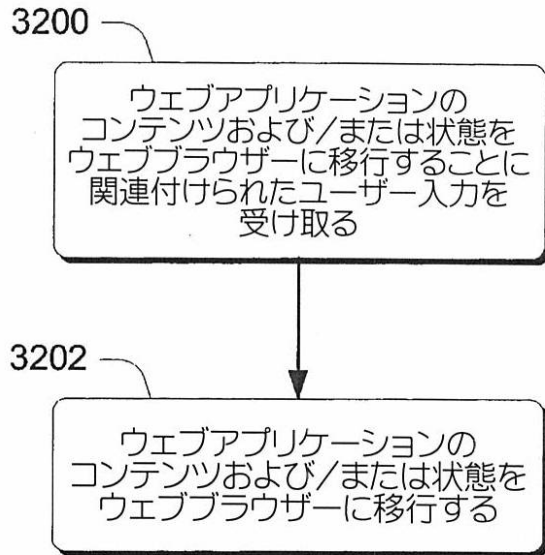
【図 27】



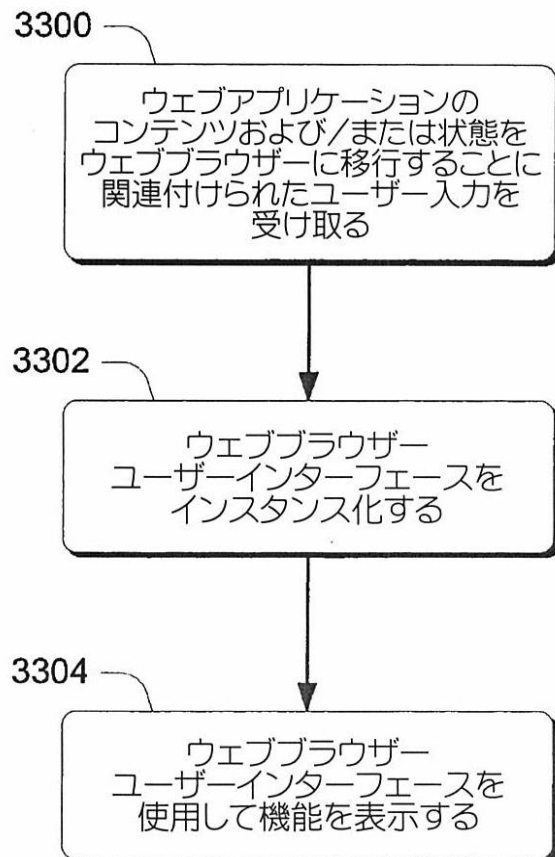
【図 28】



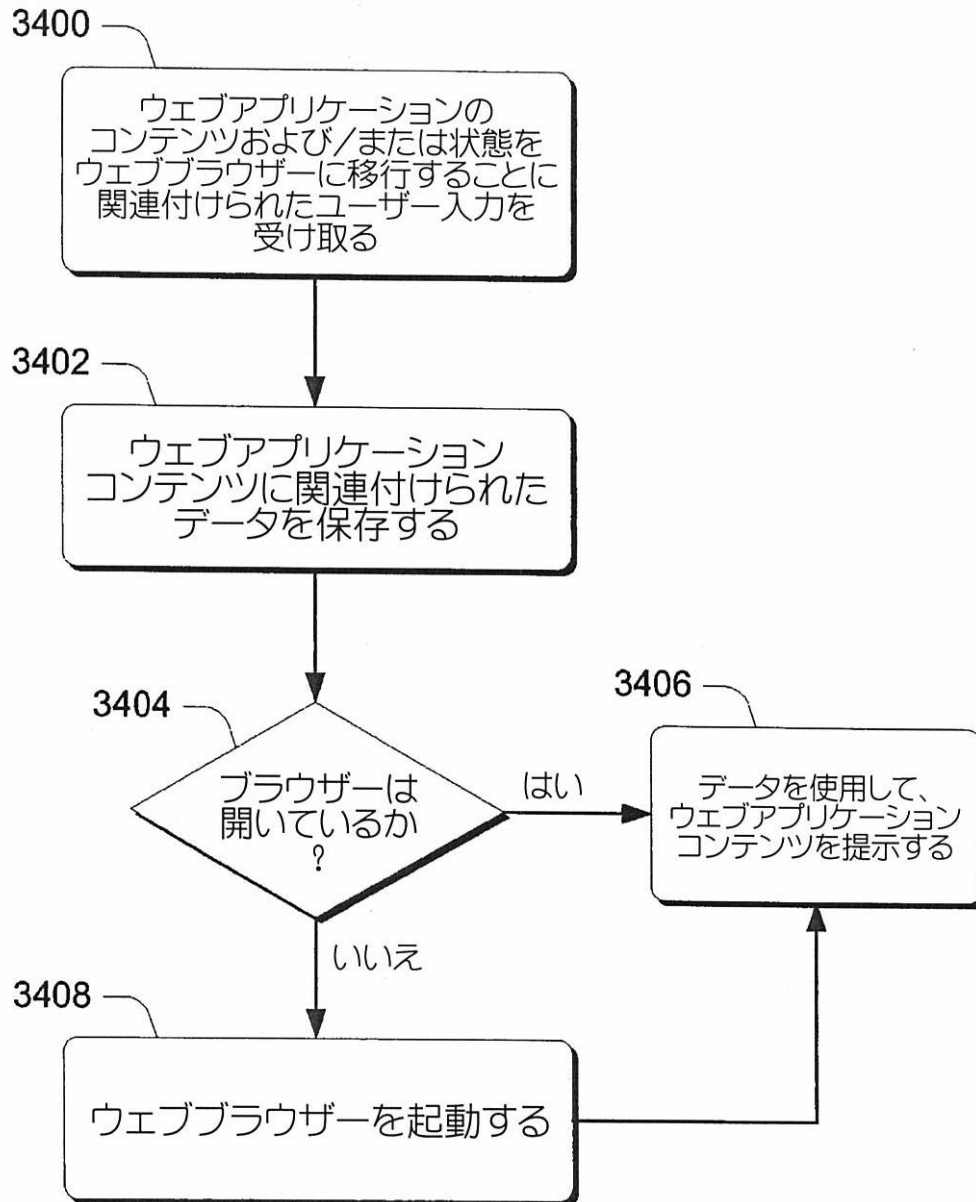
【図 3 2】



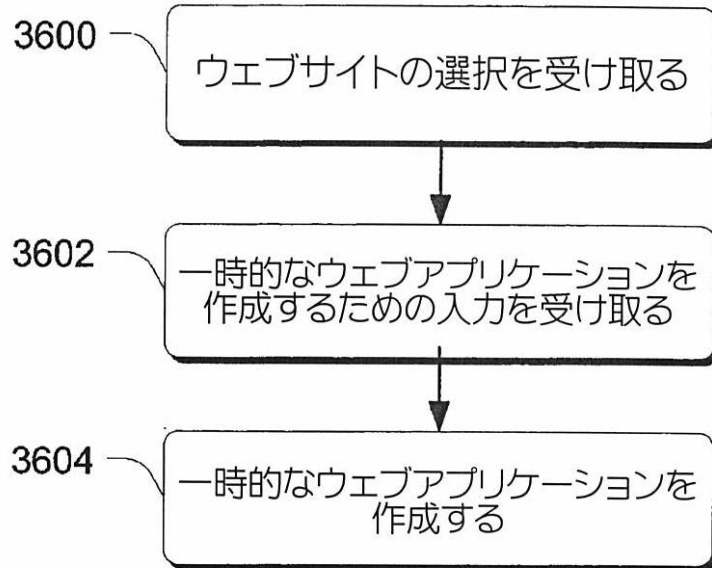
【図 3 3】



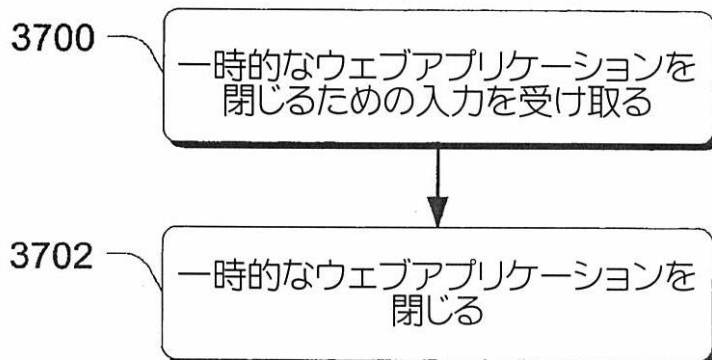
【図 3 4】



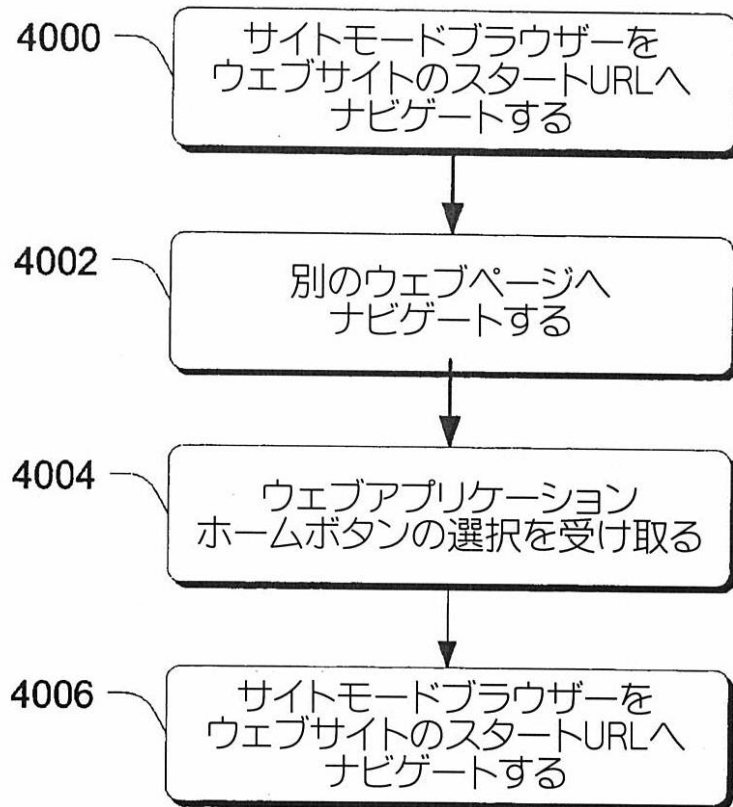
【図 3 6】



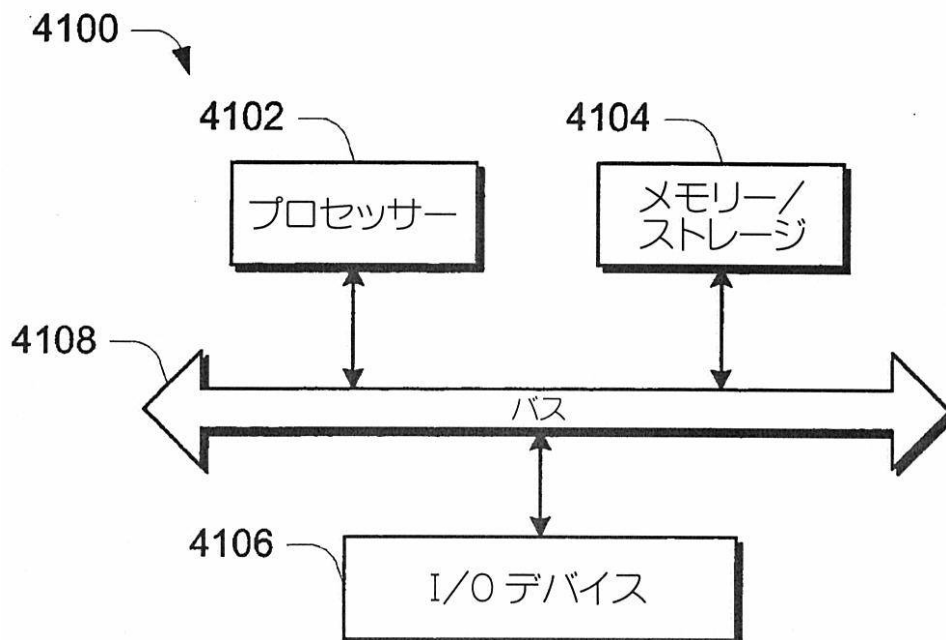
【図 3 7】



【図40】



【図41】



フロントページの続き

- (74)代理人 100153028
弁理士 上田 忠
- (74)代理人 100120112
弁理士 中西 基晴
- (74)代理人 100196508
弁理士 松尾 淳一
- (74)代理人 100147991
弁理士 鳥居 健一
- (74)代理人 100119781
弁理士 中村 彰吾
- (74)代理人 100162846
弁理士 大牧 綾子
- (74)代理人 100173565
弁理士 末松 亮太
- (74)代理人 100138759
弁理士 大房 直樹
- (72)発明者 キム, ジェーン・ティー
アメリカ合衆国ワシントン州98052-6399, 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, エルシーエイインターナショナル・パテント
- (72)発明者 ウィリアムズ, メアリー・リン
アメリカ合衆国ワシントン州98052-6399, 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, エルシーエイインターナショナル・パテント
- (72)発明者 ホール, マーティン・ジェイ
アメリカ合衆国ワシントン州98052-6399, 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, エルシーエイインターナショナル・パテント
- (72)発明者 ホルブルック, ジェス・エス
アメリカ合衆国ワシントン州98052-6399, 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, エルシーエイインターナショナル・パテント
- (72)発明者 モーガン, ブルース・エイ
アメリカ合衆国ワシントン州98052-6399, 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, エルシーエイインターナショナル・パテント
- (72)発明者 マレク, アレクサンダー・エイチ
アメリカ合衆国ワシントン州98052-6399, 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, エルシーエイインターナショナル・パテント
- (72)発明者 ヒレリオ, イスラエル
アメリカ合衆国ワシントン州98052-6399, 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, エルシーエイインターナショナル・パテント

審査官 宮久保 博幸

- (56)参考文献 米国特許第06823330 (US, B1)
特表2006-505078 (JP, A)
宮本朱美, IE7の6倍高速!?最新ベータ登場でリリース間近! Firefox3新機能大全, 週刊アスキー, 株式会社アスキー・メディアワークス, 2008年 4月 3日, 第20巻, 通巻682号, p. 144-150

- (58)調査した分野(Int.Cl., DB名)
G06F 13/00

G 0 6 F 1 5 / 0 0