

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6533871号
(P6533871)

(45) 発行日 令和1年6月19日(2019.6.19)

(24) 登録日 令和1年5月31日(2019.5.31)

(51) Int. Cl. F I
G06F 21/41 (2013.01) G O 6 F 21/41
G06F 13/00 (2006.01) G O 6 F 13/00 5 1 0 A

請求項の数 15 (全 30 頁)

(21) 出願番号	特願2018-522725 (P2018-522725)	(73) 特許権者	516114710
(86) (22) 出願日	平成28年11月4日 (2016.11.4)		ナスダック, インコーポレイテッド
(65) 公表番号	特表2018-536232 (P2018-536232A)		NASDAQ, INC.
(43) 公表日	平成30年12月6日 (2018.12.6)		アメリカ合衆国 ニューヨーク州 100
(86) 国際出願番号	PCT/US2016/060473		06, ニューヨーク, ブロードウェイ
(87) 国際公開番号	W02017/079523		165, ワン リバティ プラザ
(87) 国際公開日	平成29年5月11日 (2017.5.11)		One Liberty Plaza,
審査請求日	平成30年6月27日 (2018.6.27)		165 Broadway, New York
(31) 優先権主張番号	15/342, 923	(74) 代理人	100076428
(32) 優先日	平成28年11月3日 (2016.11.3)		弁理士 大塚 康徳
(33) 優先権主張国	米国 (US)	(74) 代理人	100115071
(31) 優先権主張番号	62/252, 295		弁理士 大塚 康弘
(32) 優先日	平成27年11月6日 (2015.11.6)		
(33) 優先権主張国	米国 (US)		
早期審査対象出願			最終頁に続く

(54) 【発明の名称】 ウェブアプリケーションへのサインオンを制御するためのシステムおよび方法

(57) 【特許請求の範囲】

【請求項1】

少なくともひとつのクライアントデバイスとウェブアプリケーションのサーバ側プロセスを実行する少なくともひとつのサーバデバイスを備えるシステムであって、

前記少なくともひとつのクライアントデバイスが少なくともひとつのハードウェアプロセッサを有する第1処理システムを含み、前記第1処理システムが前記ウェブアプリケーションの第1クライアント側部分を提供する第1クライアントアプリケーションと前記ウェブアプリケーションの第2クライアント側部分を提供する第2クライアントアプリケーションとを実行するよう構成され、前記第1クライアントアプリケーションがブラウザであり、前記第2クライアントアプリケーションがネイティブアプリケーションであり、

前記少なくともひとつのサーバが少なくともひとつのハードウェアプロセッサを有する第2処理システムを含み、前記第2処理システムが前記ウェブアプリケーションの前記サーバ側部分を実行するよう構成され、

前記第2処理システムは、

前記第1クライアントアプリケーションから受けた第1アクセス要求に応じて、前記第1クライアントアプリケーションに、前記ウェブアプリケーションの前記サーバ側部分によって提供されるアプリケーションデータにアクセスする前にサインオンプロセスを実行させることと、

行われる前記サインオンプロセスに関連して、第1セッションを生成することであって、後に前記第1クライアントアプリケーションに前記第1セッションを用いて前記アプリ

セッションデータへのアクセスが提供される、生成することと、

前記第2クライアントアプリケーションから第2要求が受け付けられた場合、前記第2要求に含まれる少なくともひとつの識別子と前記第2要求に関連付けられたブラウザタイプとを用いて、前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能か否かを判定することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能であると判定された場合、前記第2クライアントアプリケーションに、前記第1セッションを用いた前記アプリケーションデータへのアクセスを提供することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能でないと判定された場合、前記第2クライアントアプリケーションに、前記アプリケーションデータにアクセスする前にサインオンプロセスを実行させ、前記第2クライアントアプリケーションが前記サインオンプロセスを実行した後に、前記第2クライアントアプリケーションによって行われた前記サインオンプロセスに関連して生成された第2セッションを用いて前記第2クライアントアプリケーションに前記アプリケーションデータへのアクセスを提供することと、

他のクライアントデバイスで実行される、前記第1クライアントアプリケーションまたは前記第2クライアントアプリケーションのいずれかのインスタンスによるサインオンに関連して、第3セッションを生成することであって、前記他のデバイスで実行される該インスタンスに前記第3セッションを用いた前記アプリケーションデータへのアクセスを後に提供する、生成することと、

前記第3セッションを生成することに応じて、前記第1セッションを不活性化し、前記少なくともひとつのクライアントデバイスで実行される前記第1クライアントアプリケーションおよび前記第2クライアントアプリケーションによる前記アプリケーションデータへのアクセスを防ぐことと、を含む動作を行うよう構成されるシステム。

【請求項2】

前記判定することは、前記少なくともひとつのサーバデバイスのメモリに、前記識別子に対応する値が、複数のセッションレコードのうちの少なくともひとつに関連付けて保持されているか否かを判定することを含む請求項1に記載のシステム。

【請求項3】

前記判定することは、前記ウェブアプリケーションに関連付けられた、対応する以前に格納された資格を特定することをさらに含み、

前記判定することはさらに前記特定された資格に基づく請求項2に記載のシステム。

【請求項4】

前記第1処理システムはさらに、

前記第2クライアントアプリケーションが、前記ウェブアプリケーションの前記サーバ側部分と通信するために、前記第2クライアントアプリケーションによって用いられるひとつ以上の候補ブラウザコントロールを決定することと、

前記第2クライアントアプリケーションが前記サーバデバイスに、前記第2要求の第1インスタンスを送信することであって、前記第1インスタンスが前記ひとつ以上の候補ブラウザコントロールのうちの選択された第1候補ブラウザコントロールに対応する、送信することと、

前記第2要求の送信された前記第1インスタンスに対して受信された応答に基づいて、前記第2クライアントアプリケーションが、前記アプリケーションデータにアクセスするのか、または、前記第2要求の第2インスタンスを送信するのか、を判定することであって、前記第2インスタンスが前記ひとつ以上の候補ブラウザコントロールのうちの別のひとつに対応する、判定することと、を行うよう構成される請求項1に記載のシステム。

【請求項5】

(a) 前記第1クライアントアプリケーションのブラウザタイプが前記ひとつ以上の候

10

20

30

40

50

補ブラウザコントロールのうちの選択された前記第 1 候補ブラウザコントロールのブラウザタイプと同じである場合、前記受信された応答は前記第 1 セッションの存在を示し、(b) 前記第 1 クライアントアプリケーションの前記ブラウザタイプが前記ひとつ以上の候補ブラウザコントロールのうちの選択された前記第 1 候補ブラウザコントロールの前記ブラウザタイプと異なる場合、前記受信された応答は前記第 2 クライアントアプリケーションが使用可能な既存のセッションの発見に失敗したことを表す請求項 4 に記載のシステム。

【請求項 6】

前記第 1 処理システムはさらに、

前記受信された応答が前記第 1 セッションの前記存在を示し、かつ、前記ひとつ以上の候補ブラウザコントロールのうちの現在選択されている候補ブラウザコントロールのブラウザタイプが所定のブラウザタイプとは異なる場合、前記応答に含まれる第 1 セッション識別子を取得することと、

前記所定のブラウザタイプのブラウザコントロールをインスタンス化することと、

前記所定のブラウザタイプの前記インスタンス化されたブラウザコントロールと前記第 1 セッション識別子とを用いて前記アプリケーションデータにアクセスすることと、を行うよう構成される請求項 5 に記載のシステム。

【請求項 7】

前記第 1 処理システムはさらに、

前記第 2 要求のひとつ以上のインスタンスを送信した後に、前記第 2 要求のインスタンスについて受信した応答に関連して、前記第 2 クライアントアプリケーションが前記サインオンを実行することを行うよう構成される請求項 5 に記載のシステム。

【請求項 8】

前記第 2 処理システムはさらに、

前記第 2 要求のインスタンスを受信することに応じて、前記第 1 セッションが前記第 2 要求に現在関連付けられているブラウザタイプに対応するか否かを判定することと、

前記判定することが前記第 1 セッションが前記第 2 要求に現在関連付けられているブラウザタイプに対応すると判定する場合、前記第 2 クライアントアプリケーションに、前記第 1 セッションに関連付けられた第 1 セッション識別子を返すことと、

前記判定することが前記第 1 セッションが前記第 2 要求に現在関連付けられている前記ブラウザタイプに対応しないと判定する場合、前記第 2 クライアントアプリケーションに、前記第 2 クライアントアプリケーションが使用可能なセッションが見つからなかったことを示すものを返すことと、を行うよう構成される請求項 1 に記載のシステム。

【請求項 9】

ウェブアプリケーションのサーバ側部分を実行するよう構成された少なくともひとつのハードウェアプロセッサを含むサーバデバイスであって、前記少なくともひとつのプロセッサは、

前記ウェブアプリケーションの第 1 クライアント側部分を有する第 1 クライアントアプリケーションであって第 1 クライアントデバイスによって実行される第 1 クライアントアプリケーションから受けた第 1 アクセス要求に応じて、前記第 1 クライアントアプリケーションに、前記ウェブアプリケーションの前記サーバ側部分によって提供されるアプリケーションデータにアクセスする前にサインオンプロセスを実行させることであって、前記第 1 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である、実行させることと、

行われる前記サインオンプロセスに関連して、第 1 セッションを生成することであって、後に前記第 1 クライアントアプリケーションに前記第 1 セッションを用いて前記アプリケーションデータへのアクセスが提供される、生成することと、

前記ウェブアプリケーションの第 2 クライアント側部分を有する第 2 クライアントアプリケーションであって第 1 クライアントデバイスによって実行される第 2 クライアントアプリケーションから第 2 要求が受け付けられた場合、前記第 2 要求に含まれる少なくとも

10

20

30

40

50

ひとつの識別子と前記第2要求に関連付けられたブラウザタイプとを用いて、前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能か否かを判定することと、前記第2クライアントアプリケーションがネイティブアプリケーションである、判定することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能であると判定された場合、前記第2クライアントアプリケーションに、前記第1セッションを用いた前記アプリケーションデータへのアクセスを提供することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能でないと判定された場合、前記第2クライアントアプリケーションに、前記アプリケーションデータにアクセスする前にサインオンプロセスを実行させ、前記第2クライアントアプリケーションが前記サインオンプロセスを実行した後に、前記第2クライアントアプリケーションによって行われた前記サインオンプロセスに関連して生成された第2セッションを用いて前記第2クライアントアプリケーションに前記アプリケーションデータへのアクセスを提供することと、

他のクライアントデバイスで実行される、前記第1クライアントアプリケーションまたは前記第2クライアントアプリケーションのいずれかのインスタンスによるサインオンに関連して、第3セッションを生成することと、前記他のデバイスで実行される該インスタンスに前記第3セッションを用いた前記アプリケーションデータへのアクセスを後に提供する、生成することと、

前記第3セッションを生成することに応じて、前記第1セッションを不活性化し、前記第1クライアントデバイスで実行される前記第1クライアントアプリケーションおよび前記第2クライアントアプリケーションによる前記アプリケーションデータへのアクセスを防ぐことと、を含む動作を行うよう構成されるサーバデバイス。

【請求項10】

前記少なくともひとつのプロセッサはさらに、

前記第2要求のインスタンスを受信することに応じて、前記第1セッションが前記第2要求に現在関連付けられているブラウザタイプに対応するか否かを判定することと、

前記判定することが前記第1セッションが前記第2要求に現在関連付けられているブラウザタイプに対応すると判定する場合、前記第2クライアントアプリケーションに、前記第1セッションに関連付けられた第1セッション識別子を返すことと、

前記判定することが前記第1セッションが前記第2要求に現在関連付けられている前記ブラウザタイプに対応しないと判定する場合、前記第2クライアントアプリケーションに、前記第2クライアントアプリケーションが使用可能なセッションが見つからなかったことを示すものを返すことと、を行うよう構成される請求項9に記載のサーバデバイス。

【請求項11】

少なくともひとつのプロセッサを含むサーバデバイスによって行われる方法と、ウェブアプリケーションのサーバ側部分を実行することと、

前記ウェブアプリケーションの第1クライアント側部分を有する第1クライアントアプリケーションであって第1クライアントデバイスによって実行される第1クライアントアプリケーションから受けた第1アクセス要求に応じて、前記第1クライアントアプリケーションに、前記ウェブアプリケーションの前記サーバ側部分によって提供されるアプリケーションデータにアクセスする前にサインオンプロセスを実行させることと、前記第1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である、実行させることと、

前記行われるサインオンプロセスに関連して、第1セッションを生成することと、後に前記第1クライアントアプリケーションに前記第1セッションを用いて前記アプリケーションデータへのアクセスが提供される、生成することと、

前記ウェブアプリケーションの第2クライアント側部分を有する第2クライアントアプリケーションであって第1クライアントデバイスによって実行される第2クライアントア

10

20

30

40

50

アプリケーションから第2要求が受け付けられた場合、前記第2要求に含まれる少なくともひとつの識別子と前記第2要求に関連付けられたブラウザタイプとを用いて、前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能か否かを判定することと、前記第2クライアントアプリケーションがネイティブアプリケーションである、判定することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能であると判定された場合、前記第2クライアントアプリケーションに、前記第1セッションを用いた前記アプリケーションデータへのアクセスを提供することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能でないと判定された場合、前記第2クライアントアプリケーションに、前記アプリケーションデータにアクセスする前にサインオンプロセスを実行させ、前記第2クライアントアプリケーションが前記サインオンプロセスを実行した後に、前記第2クライアントアプリケーションによって行われた前記サインオンプロセスに関連して生成された第2セッションを用いて前記第2クライアントアプリケーションに前記アプリケーションデータへのアクセスを提供することと、

他のクライアントデバイスで実行される、前記第1クライアントアプリケーションまたは前記第2クライアントアプリケーションのいずれかのインスタンスによるサインオンに関連して、第3セッションを生成することとであって、前記他のデバイスで実行される該インスタンスに前記第3セッションを用いた前記アプリケーションデータへのアクセスを後に提供する、生成することと、

前記第3セッションを生成することに応じて、前記第1セッションを不活性化し、前記第1クライアントデバイスで実行される前記第1クライアントアプリケーションおよび前記第2クライアントアプリケーションによる前記アプリケーションデータへのアクセスを防ぐことと、を含む方法。

【請求項12】

それに保持されるプログラムインストラクションを有する非一時的コンピュータ可読保持媒体であって、前記プログラムインストラクションはサーバデバイスのコンピュータによって実行された場合、前記サーバデバイスに、

ウェブアプリケーションのサーバ側部分を実行することと、

前記ウェブアプリケーションの第1クライアント側部分を有する第1クライアントアプリケーションであって第1クライアントデバイスによって実行される第1クライアントアプリケーションから受けた第1アクセス要求に応じて、前記第1クライアントアプリケーションに、前記ウェブアプリケーションの前記サーバ側部分によって提供されるアプリケーションデータにアクセスする前にサインオンプロセスを実行させることとであって、前記第1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である、実行させることと、

行われる前記サインオンプロセスに関連して、第1セッションを生成することとであって、後に前記第1クライアントアプリケーションに前記第1セッションを用いて前記アプリケーションデータへのアクセスが提供される、生成することと、

前記ウェブアプリケーションの第2クライアント側部分を有する第2クライアントアプリケーションであって第1クライアントデバイスによって実行される第2クライアントアプリケーションから第2要求が受け付けられた場合、前記第2要求に含まれる少なくともひとつの識別子と前記第2要求に関連付けられたブラウザタイプとを用いて、前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能か否かを判定することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能であると判定された場合、前記第2クライアントアプリケーションに、前記第1セッションを用いた前記アプリケーションデータへのアクセスを提供することと、

10

20

30

40

50

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能でないと判定された場合、前記第2クライアントアプリケーションに、前記アプリケーションデータにアクセスする前にサインオンプロセスを実行させ、前記第2クライアントアプリケーションが前記サインオンプロセスを実行した後に、前記第2クライアントアプリケーションによって行われた前記サインオンプロセスに関連して生成された第2セッションを用いて前記第2クライアントアプリケーションに前記アプリケーションデータへのアクセスを提供することと、

他のクライアントデバイスで実行される、前記第1クライアントアプリケーションまたは前記第2クライアントアプリケーションのいずれかのインスタンスによるサインオンに関連して、第3セッションを生成することであって、前記他のデバイスで実行される該インスタンスに前記第3セッションを用いた前記アプリケーションデータへのアクセスを後に提供する、生成することと、

10

前記第3セッションを生成することに応じて、前記第1セッションを不活性化し、前記第1クライアントデバイスで実行される前記第1クライアントアプリケーションおよび前記第2クライアントアプリケーションによる前記アプリケーションデータへのアクセスを防ぐことと、を含む動作を行わせる非一時的コンピュータ可読保持媒体。

【請求項13】

少なくともひとつのハードウェアプロセッサを含むクライアントデバイスであって、前記少なくともひとつのハードウェアプロセッサは、ウェブアプリケーションの第1クライアント側部分を提供するブラウザアプリケーションである第1クライアントアプリケーションと前記ウェブアプリケーションの第2クライアント側部分を提供するネイティブアプリケーションである第2クライアントアプリケーションとを実行するよう構成され、前記少なくともひとつのハードウェアプロセッサは、

20

前記ウェブアプリケーションのサーバ側部分へ第1要求を送信した前記第1クライアントアプリケーションが、前記ウェブアプリケーションの前記サーバ側部分へのサインオンを行い、行われた前記サインオンの後に、行われた前記サインオンに関連して生成された第1セッションを用いて前記ウェブアプリケーションの前記サーバ側部分からアプリケーションデータにアクセスすることと、

前記第2クライアントアプリケーションがサーバデバイスへ、前記ウェブアプリケーションの前記サーバ側部分と通信するために、前記第2クライアントアプリケーションによって用いられるひとつ以上の候補ブラウザコントロールのうちの選択されたひとつを含む第2要求を送信することと、

30

送信された前記第2要求に対して受信された応答に基づいて、前記第2クライアントアプリケーションが、前記ウェブアプリケーションの前記サーバ側部分のアプリケーションデータにアクセスするのか、または、前記第2要求の別のインスタンスを送信するのか、を判定することと、を含む動作を行うよう構成され、

前記ウェブアプリケーションの前記サーバ側部分は、

前記第2クライアントアプリケーションから前記第2要求が受け付けられた場合、前記第2要求に含まれる少なくともひとつの識別子と前記第2要求に関連付けられたブラウザタイプとを用いて、前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能か否かを判定することと、

40

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能であると判定された場合、前記第2クライアントアプリケーションに、前記第1セッションを用いた前記アプリケーションデータへのアクセスを提供することと、

前記第2クライアントアプリケーションが前記アプリケーションデータにアクセスするのに前記第1セッションを使用可能でないと判定された場合、前記第2クライアントアプリケーションに、前記アプリケーションデータにアクセスする前にサインオンプロセスを実行させ、前記第2クライアントアプリケーションが前記サインオンプロセスを実行した後に、前記第2クライアントアプリケーションによって行われた前記サインオンプロセス

50

に関連して生成された第2セッションを用いて前記第2クライアントアプリケーションに前記アプリケーションデータへのアクセスを提供することと、

他のクライアントデバイスで実行される、前記第1クライアントアプリケーションまたは前記第2クライアントアプリケーションのいずれかのインスタンスによるサインオンに関連して、第3セッションを生成することと、前記他のクライアントデバイスで実行される該インスタンスに前記第3セッションを用いた前記アプリケーションデータへのアクセスを後に提供する、生成することと、

前記第3セッションを生成することに応じて、前記第1セッションを不活性化し、前記クライアントデバイスで実行される前記第1クライアントアプリケーションおよび前記第2クライアントアプリケーションによる前記アプリケーションデータへのアクセスを防ぐことと、を行うよう構成されるクライアントデバイス。

10

【請求項14】

(a) 前記第1クライアントアプリケーションのブラウザタイプが前記ひとつ以上の候補ブラウザコントロールのうちの選択された前記候補ブラウザコントロールのブラウザタイプと同じである場合、前記受信された応答は前記第1セッションの存在を示し、(b) 前記第1クライアントアプリケーションの前記ブラウザタイプが前記ひとつ以上の候補ブラウザコントロールのうちの選択された前記候補ブラウザコントロールの前記ブラウザタイプと異なる場合、前記受信された応答は前記第2クライアントアプリケーションが使用可能な既存のセッションの発見に失敗したことを表す請求項13に記載のクライアントデバイス。

20

【請求項15】

前記少なくともひとつのハードウェアプロセッサはさらに

前記受信された応答が前記第1セッションの前記存在を示し、かつ、前記ひとつ以上の候補ブラウザコントロールのうちの選択された前記候補ブラウザコントロールのブラウザタイプが所定のブラウザタイプとは異なる場合、前記応答に含まれる第1セッション識別子を取得することと、

前記所定のブラウザタイプのブラウザコントロールをインスタンス化することと、

前記所定のブラウザタイプの前記インスタンス化されたブラウザコントロールと前記第1セッション識別子とを用いて前記アプリケーションデータにアクセスすることと、を行うよう構成される請求項14に記載のクライアントデバイス。

30

【発明の詳細な説明】

【技術分野】

【0001】

関連出願へのクロスリファレンス

本願は、2015年11月6日に提出された米国特許仮出願第62/252,295号の優先権の利益を享受する。その出願の開示全体は、参照により本明細書に組み入れられる。

【背景技術】

【0002】

ウェブアプリケーションの展開は、多数のユーザが、ひとつ以上のウェブアプリケーションおよび/またはウェブアプリケーションによって制御されるリソースにアクセスすることを可能とする。例えば、大企業はその企業ネットワークや他のインターネットアクセス可能コンピュータの中にあるひとつ以上のサーバに企業ソフトウェアアプリケーション(「ウェブアプリケーション」)を展開し、全ての従業員および/またはクライアントがウェブを介してそのアプリケーションにアクセスすることを可能にしてもよい。そのようなアプリケーションのウェブアクセス性は、いつでも、また、クライアントデバイスがネットワーク接続を有する場所ならどこからでも、そのアプリケーションにアクセスすることができる能力を従業員および/またはクライアントに提供する。

40

【0003】

また、ウェブアプリケーションは、異なるタイプの情報処理デバイスを用いてアクセス可能である。同じユーザが異なる情報処理デバイスを用いてウェブアプリケーションにア

50

クセスする場合がある。例えば、ユーザは彼のスマートフォンを用いてウェブアプリケーションへのアクセスを試みる一方、同時に、デスクトップコンピュータを介してそのウェブアプリケーションにアクセスするかもしれない。ウェブアプリケーションは同じユーザによって二つの異なるブラウザ（例えば、グーグル社のクロームや、マイクロソフト社のインターネットエクスプローラ）を用いてアクセスされる場合がある。

【0004】

したがって、ウェブアプリケーションの展開は、アクセス性および可用性に関する数多くの利点を提供する。同じユーザが複数のデバイスやブラウザおよび/または他のクライアントアプリケーションを用いて同時にウェブアプリケーションにアクセスすることができる能力は、ユーザとウェブアプリケーションとの相互作用の性質、例えば効率および利便性、を改善することができる。

10

【0005】

しかしながら、ソフトウェア技術およびネットワーク技術の発展により提供されるウェブアプリケーションの利便性の増大は、それが適切に制御されない場合、アクセス権限の誤用を生じさせる。例えば、プロバイダがユーザごとにライセンス料やアクセス料を課金するウェブアプリケーションの場合、あるユーザは、そのウェブアプリケーションにアクセスするのに他のユーザの権限を用いることで、料金全体を支払うことを避けようとし、したがって潜在的にプロバイダからある程度の収入を奪うことが考えられる。アクセス権限のそのような誤用は、また、料金を支払っているユーザが利用可能なシステム容量に悪影響を与える可能性があり、および/または、ユーザが体験するシステム性能に悪影響を与える可能性がある。

20

【0006】

したがって、増大する商用的価値を有するより多くの企業ソフトウェアアプリケーションがウェブアプリケーションとして展開されている状況において、そのようなアプリケーションのアクセス性を改善し、アプリケーションプロバイダのリソースの保護を改善するための技術が必要とされている。

著作権表示

【0007】

本特許書類の開示の一部は著作権保護の対象となるものを含む。著作権者は、任意の者による本特許書類または本特許開示の特許商標庁にて現れる態様での複製について反対することはないが、そうでない場合は何であれ全ての著作権を留保する。

30

【発明の概要】

【0008】

開示される技術は、ウェブアプリケーションおよびそのようなアプリケーションによって制御されるリソースへのアクセスを制御することに関する。開示される技術はシングルサインオン能力を提供する。そこでは、クライアントアプリケーションからウェブアプリケーションに既にサインオンしているユーザは、後に彼/彼女が同じまたは別のクライアントアプリケーションからウェブアプリケーションにアクセスする必要がある場合に、再度のサインオンを要求されないであろう。シングルサインオン能力は、ユーザがサインオンプロセスを行わなければならない回数を低減することにより、ウェブアプリケーションに関連付けられたユーザ体験を改善することができる。開示される技術は、サーバに複数ログイン防止能力を提供する。複数ログイン防止を用いることで、潜在的に疑わしいサインオンシナリオを検出し、疑わしいサインオンが検出された場合にはひとつ以上のアクティブなセッションを無効化することができる。

40

【0009】

本概要は、発明の詳細な説明で以下にさらに説明される単純化された概念の選択を紹介するために提供される。本概要は、クレームに係る主題の主要な特徴または不可欠な特徴を識別することと、或いは、クレームに係る主題の範囲を限定するために用いられることとのいずれをも意図するものではなく、むしろ本概要は本書面で説明される主題の概観を提供することを意図するものである。したがって、上述した特徴は例示にすぎず、本明細

50

書で説明される主題の他の特徴、態様、および利点は、後述する詳細な説明、図面、および請求項から明らかになるであろうということは理解されたい。

【図面の簡単な説明】

【0010】

【図1】ある例示的な実施の形態に係る、ひとつ以上のクライアントデバイスと、ウェブアプリケーションを実行するひとつ以上のサーバデバイスと、を有する計算環境を示す。

【0011】

【図2】ある例示的な実施の形態に係る、図1のひとつ以上のサーバによって維持されてもよいデータテーブルのうちの一つを示す。

【0012】

【図3】ある例示的な実施の形態に係る、シングルサインオン動作が行われる場合の、クライアントデバイスとサーバデバイスとの間のやりとりを示すフロー図を示す。

【0013】

【図4】ある例示的な実施の形態に係る、複数ログイン防止動作が行われる場合の、クライアントデバイスとサーバデバイスとの間のやりとりを示すフロー図を示す。

【0014】

【図5A】図5（図5Aおよび図5B）は、ある例示的な実施の形態に係る、ユーザがサインインする場合の、クライアントデバイスとサーバデバイスとの間のやりとりを示すフロー図を示す。

【図5B】図5（図5Aおよび図5B）は、ある例示的な実施の形態に係る、ユーザがサインインする場合の、クライアントデバイスとサーバデバイスとの間のやりとりを示すフロー図を示す。

【0015】

【図6】ある例示的な実施の形態に係る、シングルサインオン動作の後に複数ログイン防止が続く場合の、クライアントデバイスとサーバデバイスとの間のやりとりを示すフロー図を示す。

【0016】

【図7】ある例示的な実施の形態に係る、ネイティブアプリケーションが起動時にウェブアプリケーションにサインオンするために用いるプロセスを示す。

【0017】

【図8】ある例示的な実施の形態に係る、クライアントのデータ要求についてのやりとりのフローチャートを示す。

【0018】

【図9】ある例示的な実施の形態に係る、クライアントデバイス上のネイティブアプリケーションの上に重ね合わされて提示されているサインオンウィンドウを示す。

【0019】

【図10】ある例示的な実施の形態に係る、クライアントデバイスおよび/またはサーバを実装可能な計算環境を模式的に示す。

【発明を実施するための形態】

【0020】

本明細書で説明される技術はとりわけひとつ以上のウェブアプリケーションへのユーザアクセスを制御することに関する。ある例示的な実施の形態はシングルサインオン（SSO）能力を提供し、SSO能力は、ログインプロセスを一度実行すれば、同じアプリケーションや関連アプリケーションの他の側面への後のアクセスをより少ない認証で、またはさらにサインオンアクションを行う必要なしに、行うことができるという利便性をユーザに提供する。ある例示的な実施の形態は複数ログイン防止（MLP）能力を含んでもよく、MLP能力は、ウェブアプリケーションにより生成される収入の流れを保護することの助けとなり、また、ウェブアプリケーションを実行するサーバの処理容量を保護することの助けとなりうる。MLP能力を用いることで、例えば、同時に複数のエンドポイントからあるウェブアプリケーションにアクセスするために同じ識別子/資格証明（例えば、ユ

10

20

30

40

50

ーザIDや特定ユーザの資格証明)が用いられたときを検出し、ウェブアプリケーションへのアクセス能力の誤用を低減するためにひとつ以上のセッションをキャンセルするよう動作することができる。ある実施の形態により提供されるSSO能力および/またはMLP能力は、任意のタイプのウェブアプリケーションにおいて効率やアクセススピードやユーザ利便性を向上させるが、多数のアプリケーションに頻繁にかつ利便性高くアクセスすることができる能力をユーザに提供するシングルページアプリケーション(SPA)フレームワークを伴うウェブアプリケーションにおいてさらに有益でありうる。

【0021】

図1は、ある実施の形態に係る、ひとつ以上のサーバ102(本明細書では「サーバインフラストラクチャ」とも称される)とひとつ以上のクライアント104とを含む非限定的計算環境100を示す。ひとつ以上のサーバ102はクライアント104およびひとつ以上の外部サーバと通信し、その結果、クライアント104のユーザはひとつ以上のサーバ102上で実行されるウェブアプリケーション112にアクセスすることができる。また、サーバ102はドキュメント管理システム108と通信してもよい。サーバ102、クライアント104、外部サーバ106、およびドキュメント管理システム108間の通信はインターネットや他の任意の通信ネットワークを介するものであってもよい。

10

【0022】

サーバ102は、ネットワークを介して互いに通信可能に接続されるひとつ以上の物理的なサーバコンピュータに実装されてもよい。ひとつ以上の物理的なサーバコンピュータは地理的に同じ場所にあってもよいし、分散していてもよい。サーバ102は、データベース管理システム110と、ひとつ以上のサーバ側ウェブアプリケーション112(例えば、112aおよび112b)と、コアサービスアプリケーション114と、を含んでもよい。

20

【0023】

ウェブサーバ、アプリケーションサーバおよびデータベースからなる三層モデルにしたがって各ウェブアプリケーション112を設計し、動作させてもよい。従来のシステムと同様に、各ウェブアプリケーション112のウェブサーバは、HTTPを介して、およびJSON、HTML、XML、JavaScript、Cascading Style Sheets(CSS)などの他のプロトコルを介して、外部エンティティと通信する能力を含む。各ウェブアプリケーション112のアプリケーションサーバはアプリケーションの処理ロジックを提供し、各ウェブアプリケーション112のデータベースはデータ格納およびデータ読み出しを提供する。

30

【0024】

ウェブアプリケーション112は、ユーザ認証を管理するためにコアサービスアプリケーション114とやりとりし、クライアント104から入力を受信しかつクライアント104に送信するためにクライアント104(例えば104aおよび104b)とやりとりし、クライアント104上で実行されている要求元のクライアントアプリケーションに提供されるべき情報を取得するためにデータベース管理システム110および/または外部サーバ106とやりとりしてもよい。ある実施の形態では、要求元のクライアントに提供される情報のいくらかまたは全ては、ウェブアプリケーションそのものおよび/または同じひとつ以上のサーバ102がローカルで実行している他のウェブアプリケーションによって生成されてもよい。

40

【0025】

コアサービスアプリケーション114は上述の三層モデルにしたがい設計され動作してもよく、また、ウェブアプリケーション112によって共通に用いられるひとつ以上のサービスを提供してもよい。コアサービスアプリケーション114によって提供されてもよい例示的なサービスは、ユーザの認証、セッションの管理等を含む。

【0026】

ある実施の形態では、コアサービスアプリケーション114はまた、外部サーバ106に対してセッション管理を提供することができる。例えば、二つ以上のウェブアプリケー

50

ション112が特定の外部サーバ106からデータを取得するユースケースでは、コアサービスアプリケーション114は、サーバ102と外部サーバ106との間のセッションを、クライアント104とサーバ102との間の対応する他のセッションにしたがって管理する能力を提供してもよい。各外部サーバはサーバ102で維持されているセッションテーブルの一部のコピー119を保持してもよい。

【0027】

ウェブアプリケーション112は、クライアント104から要求を受信し、データベース110および/または外部サーバ106からの情報の処理を行いおよび/または該情報を取得し、該処理および/または取得されたデータからの結果でクライアント104に回答するよう動作する。ウェブアプリケーション112はユーザおよびユーザセッションを管理するためにコアサービス114を用いてもよい。

10

【0028】

ウェブアプリケーションは、ひとつ以上のクライアント側コンポーネントと、ひとつ以上のサーバ側コンポーネントと、を備えてもよい。ウェブアプリケーションのクライアント側コンポーネントは、ユーザインタフェースデバイス上での情報の提供(例えば、表示)を行うことによるユーザインタフェースの取り扱いと、ユーザ入力の受け付けと、等を提供するよう動作してもよい。サーバ側コンポーネントは、認証、サービス測定、受け付けたユーザ入力にしたがいユーザに提供されるべき情報の生成または取得、を提供してもよい。

【0029】

20

実施の形態はウェブアプリケーションの特定のタイプに限定されない。実施の形態で使用可能なウェブアプリケーションは、シングルページアプリケーション(SPA)モデルまたは任意の非SPAモデルもしくはそれらの組み合わせにしたがい設計されたものを含んでもよい。

【0030】

SPAは、単一のウェブページ内で動作するウェブアプリケーションである。SPAでは、単一のウェブページのコンテンツはウェブサーバによってウェブブラウザに送信され、そのページはロード/描画され、これは従来のウェブアプリケーションについて上述した通りである。その後、ユーザがアプリケーション内の異なるコンテンツを見ることを希望する場合、ユーザはハイパーリンクをクリックするかページに入力する。しかしながら、非SPAウェブアプリケーションでそうであるように異なるページに導く代わりに、同じページがロードされ続け、そのコンテンツが動的に更新される。この動的更新は異なる複数の方法で達成可能である。それは、例えば、ウェブブラウザが新たなコンテンツのバックグラウンドHTTPフェッチを行うことや、(JavaScriptコードを介して)ページのドキュメントオブジェクトモデル(DOM)を更新することや、および/または他の技術を含んでもよい。

30

【0031】

AngularJS(登録商標)はSPAを生成するために使用されるウェブアプリケーションフレームワークである。ウェブブラウザにおいて、AngularJS JavaScriptライブラリがロードされ、AngularJS JavaScriptおよび他のAngularJSコーディングコンストラクトと共に埋め込まれたHTMLテンプレートを解釈する。その結果、得られるページはテンプレートで規定される通りに動作する。また、SPAアプリケーションのために他のフレームワーク(例えば、Backbone.js、Ember.js、およびReact)を用いてもよい。

40

【0032】

ある非SPAウェブアプリケーションモデルでは、ウェブアプリケーションは異なる複数のウェブページを含む。アプリケーション内の特定のウェブページを描画するために、以下のやりとりの集合が行われる。クライアントデバイスのウェブブラウザが(ハイパーテキストトランスファプロトコル(HTTP)メッセージを用いて)ウェブサーバに特定のウェブページを要求し、それに回答して、ウェブサーバが(HTTPを用いて)ページのコードをウェブブラウザに返信し、そのコードが例えばHTML、JavaScript

50

t (登録商標)、およびカスケーディングスタイルシート(CSS)コードを含み、ウェブブラウザが次いでコードをロードしてページを描画し、それによりユーザがページを見て相互作用できるようにする。ユーザが後にアプリケーション内の異なるコンテンツを見ることを希望する場合、ユーザはアプリケーション内の異なるページを指定するハイパーリンクをクリックするかそれを指定する入力をページに対して行い、次いで、その異なるページに対して上述の要求/応答/ロード/描画手順が行われる。

【0033】

データベース管理システム110(本明細書において単にデータベースと称される場合がある)は市販のDBMSや他のデータ記録管理システムであってもよい。単一のDBMSとして示されているが、DBMS110はひとつ以上の別個のデータベースを含んでもよい。実施の形態はデータベース管理システムのタイプに限定されない。

10

【0034】

クライアント104aおよび104bは同じまたは異なるクライアントアプリケーションを実行するよう構成されてもよい。図1に示される例示的な実施の形態では、第1クライアント104aはウェブブラウザA116とウェブブラウザB117とを含む。クライアント104aはまた、ネイティブアプリケーションであるクライアント側アプリケーション120aをそれに保持する形で有してもよい。ブラウザA116を用いてウェブアプリケーション112にアクセスする場合、ウェブアプリケーション112のクライアント側コード118はブラウザA116内で実行される。同じ例示的な実施の形態では、第2クライアント104bは、ネイティブアプリケーションであってもよいアプリケーション120bを実行するよう構成される。クライアント104bはネイティブアプリケーション120bに加えてブラウザ(例えば、ブラウザAやブラウザCなどとして示される)を有してもよい。クライアント側コード118およびアプリケーション120は、サーバ102上の対応するウェブアプリケーションのためのクライアント側処理を実行してもよい。

20

【0035】

図1に示されるように、クライアントアプリケーション(例えば、116/118や120a/120b)がウェブアプリケーション112と通信するとき、ウェブアプリケーション112はクライアントが要求した任意の情報をひとつ以上の外部サーバ106から取得し、クライアントアプリケーションに提供してもよい。ある実施の形態では、要求元のクライアント104に提供される情報のいくらかまたは全てはサーバ102によってローカルで生成されてもよい。

30

【0036】

クライアント104は、パーソナルコンピュータ、モバイルコンピュータ、タブレット、スマートフォン、および他の電子デバイスを含んでもよい。ある例示的な実施の形態では、少なくともディスプレイとユーザ入力のための入力デバイスとサーバデバイスとの通信のための通信インタフェースとを含む任意の電子計算デバイスはクライアントデバイスとして動作する。

【0037】

外部サーバ106(例えば、106a、106b、106c)は、ひとつ以上のサーバ102を制御するエンティティとは異なるエンティティ(例えば、管理エンティティ)によって制御されるひとつ以上のサーバを含んでもよい。例えば、外部サーバ106のうちのひとつ以上は、サービスプロバイダまたはベンダによって管理されてもよい。サービスプロバイダまたはベンダは、所定の(例えば、企業とベンダとによって以前に合意された)サービス明細の下、クライアントデバイス104のユーザに種々のアプリケーションデータへのアクセスおよび/または解析を提供する。サービス明細は、外部サーバへのアクセスのユーザ数、情報のタイプおよび量、利用期間、利用頻度等のうちのひとつ以上に基づくものであってもよい。

40

【0038】

図1に示されるソフトウェアモジュールがハードウェアコンポーネント(例えば、プロ

50

セッサやメモリ)によって保持され実行されることは理解されるべきであり、さらに、ソフトウェアモジュールが任意の動作を行うと本明細書で説明するときにはいつでも、それは説明を簡単にするためにのみなされるものであって、実際は、その動作はソフトウェアモジュールを構成するインストラクションおよびデータにしたがい下層のハードウェアが行うものであることは理解されるべきである。本明細書で説明されるフィーチャを実装するために用いられてもよい例示的なハードウェアコンポーネントに関するさらなる詳細は、本明細書の他の場所と共に、図10を参照して後に提供される。

【0039】

例示的な実装では、計算環境100はナスダックコーポレイションなどの企業に関連付けられてもよい。例示的なウェブアプリケーション112はリアルタイムマーケット解析アプリケーションとクライアントアカウント状態アプリケーションとを含んでもよい。ウェブアプリケーション112のユーザは、金融アナリストや企業の他の従業員を含んでもよい。コアサービスアプリケーション114は、管理(例えば、ユーザアカウント生成、資格管理等)、認証(例えば、ユーザが所定のサービスにアクセスするためのセッションの生成/管理等)、ユーザ承認(例えば、ユーザが所定のサービスやフィーチャにアクセスする権限があるかのチェック等)、などの共通サービスを提供してもよい。サーバ102は、ウェブアプリケーション112、コアサービスアプリケーション114および関連ソフトウェアを実行するために企業が用いるひとつ以上のサーバおよび関連インフラストラクチャを表してもよい。ドキュメント管理システム108は、企業のユーザにサービスを届けるためにウェブアプリケーションおよび/またはコアサービスアプリケーションと通信する顧客関係管理アプリケーションを含んでもよい。

【0040】

この例示的な実装では、外部サーバ106はそれぞれ、対応するアプリケーションデータベンダによって運用されてもよい。各ベンダは、ある種のサービス明細(例えば、データタイプや利用量やユーザ数などに基づく合意された価格)の下、ナスダックコーポレイションなどのエンティティにリアルタイム金融マーケット関連情報などのアプリケーションデータを提供してもよい。

【0041】

アナリストがクライアント104aを用いてサーバ102上のリアルタイムマーケット解析アプリケーションにアクセスすると、SPAがクライアントデバイスに表示され、ベンダ(例えば、外部サーバ106を運用するもの)および/または企業の内部アナリスト等からの種々のリアルタイムまたは付加価値付き情報が、表示されているSPAのひとつ以上の部分に表示されうる。

【0042】

外部サーバ106(例えば、ベンダ)は、要求された情報を提供しそのサービスにアクセスしているユーザを特定することができるが、例えば企業とベンダとの間で合意されたサービス明細で規定される利用条件にユーザが従うことを確認するために、企業(「ベンダオブレコード」として)に依存してもよい。例えば、特定のベンダとの合意に基づいて、コアサービスアプリケーションを用いることで、その特定のベンダにアクセスし/その特定のベンダからデータを取得するための資格(アクセス可能なデータのタイプ/量の指定やもしあれば他の制限)をユーザおよび/またはウェブアプリケーションに割り当てることができる。コアサービス114はベンダのデータに対する正当な資格を有する対応するユーザについて、そのベンダとのセッションを確立してもよい。コアサービス114は、各ベンダにおいて、セッションテーブル(例えば、116)をもたらずか維持してもよい。例えばサーバ102上のウェブアプリケーションのうちのひとつを通じて、クライアントデバイス114のユーザがベンダにデータを要求する場合、ある実施の形態では、ベンダはそのセッションテーブルのコピーをチェックし、要求元のユーザの資格が正当であるかの検証をコアサービス114に要求し、資格が正当であれば要求されたデータを提供するようにしてもよい。

【0043】

10

20

30

40

50

図2は、ある例示的な実施の形態に係る、コアサービスアプリケーション114によって維持されてもよくおよび/またはコアサービスアプリケーション114がアクセス可能なデータテーブル200の例示的な集合を示す。「テーブル」と称されてはいるが、図2に関連して議論されるデータは、コアサービスアプリケーション114および/またはデータを利用するよう構成される他のアプリケーションがアクセス可能な任意の形態やデータ構造で保持されてもよい。

【0044】

ユーザ認証情報202はユーザ認証情報のテーブルを含んでもよい。このテーブルは、例えばウェブアプリケーションのうちの一つ以上へのサインオン中にユーザを認証するためにコアサービス114によってアクセスされる。ユーザ認証情報202において、各ユーザについて、ユーザID、パスワード、ユーザ特定クッキー、およびユーザに関連付けられた資格などのうちのいずれかなどの特定情報が維持されてもよい。

10

【0045】

ユーザセッション情報204は、クライアントデバイス104のユーザなどのユーザがサーバ102上のウェブアプリケーションと現在アクティブなセッションにより通信するときのその現在アクティブなセッションの記録を含む。ユーザセッション情報204は、各行がアクティブなセッションを表すテーブルの形態であってもよい。ある実施の形態では、非アクティブセッション（例えば、最近不活性化された/キャンセルされたセッション）もまたテーブルにあってもよい。ユーザセッション情報テーブル204は、各セッションについて、ユーザID、セッションを利用するウェブアプリケーション、クライアント、認証セッション識別子および/または一つ以上のクッキーなどのフィールド/列を含んでもよい。テーブルはまた、セッションの開始時刻および終了時刻と現在の状態とを含んでもよい。状態は、例えば、セッションが断絶されたかまたは現在アクティブであるか、および断絶されているのであれば断絶の理由（例えば、MLPによる断絶）、を示してもよい。

20

【0046】

外部サーバセッション情報206は、対応する外部サーバ106に関するユーザセッションを保持する。ある実施の形態では、外部サーバセッション情報206は、外部サーバ106ごとのデータ構造で維持されてもよい。ある実施の形態では、各テーブル206を対応する外部サーバ106に提供することができる。この場合、要求が与えられたときに、外部サーバが、要求に関連付けられた許可および/または資格をローカルでチェックすることができる。ある実施の形態では、サーバデバイス102と外部サーバ106との間のセッションはユーザセッション情報204に基づくものであってもよく、別個のテーブルは不要でありうる。

30

【0047】

資格情報208はウェブアプリケーションへのアクセスに関連付けられた資格に関する情報を含む。上述の通り、資格は、ユーザまたはユーザのグループに対して、一つ以上の外部サーバ（または他のリソース）へのアクセスの範囲を指定する。例えば、資格情報208は、各ユーザまたはユーザの各グループがアクセスを有する外部サーバ（ベンダ）およびデータのタイプを指定してもよい。資格はまた、同じユーザによる、複数のクライアントアプリケーションによる同時アクセス（例えば、ブラウザおよびネイティブアプリケーションを介した同時アクセス）が特定のウェブアプリケーションについて許されているかを指定してもよい。

40

【0048】

図3は、ある例示的な実施の形態に係る、クライアント104aのユーザについてのシングルサインオンプロセス300で生じうる、クライアント104aとサーバ102との間のやりとりを示すフローチャートである。

【0049】

プロセス300は、ユーザがクライアント104aを用いてブラウザを、サーバ102で実行されているウェブアプリケーションへと至らせたときに始ってもよい。例えば、

50

ユーザはウェブブラウザに、ウェブアプリケーションにアクセスするための所定のURL (uniform resource locator) を入力してもよい。あるいはまた、ブラウザは、ユーザがウェブページに表示されるリンクを「クリック」したときにウェブアプリケーションへと導かれてもよい。ウェブアプリケーションへの最初の到着のときに出会うウェブアプリケーションのウェブページは「ランディングページ」と称される。

【0050】

ウェブブラウザ(例えば、インターネット 익스プローラブラウザ 312)を用いてウェブアプリケーションのランディングページに辿り着いた後、動作302において、ユーザはウェブアプリケーション(例えば、324)にサインオンされる(「ログイン」とも称される)。サインオンプロセスは図5との関連で後に詳述される。ユーザのサインオンプロセスが成功すると、セッションが生成され、クッキーが生成される。セッションの生成は、例えばユーザセッション情報204などのセッションテーブルへのエントリの追加を含む。クッキーは生成されたセッションについて一意であってもよく、従来技術を用いて生成された文字列であってもよい。クッキーはサーバ102によってクライアント104aに返され、将来の使用のためにクライアント104aに保存される。ウェブアプリケーション324に対するクライアント104aを用いたユーザによる以後の要求(例えば、HTTP要求)はクッキーを含み、サーバ102は受信したクッキーを用いて関連要求の追跡を行ってもよい。この時点で、ユーザはウェブアプリケーションに適切にサインオンし、ブラウザコンテンツにおいて動作しているウェブアプリケーションのクライアント側部分はウェブアプリケーションのサーバ側部分とメッセージおよび/またはデータを交換することができる。

【0051】

図3において、動作302は、ブラウザA312のフローレーンおよびウェブアプリケーション324のフローレーンおよびコアサービス326のフローレーンを実線の円でマークし、ブラウザB314のフローレーンおよびアプリケーション316のフローレーンが対応する破線の円を有するように、示されている。動作302などの動作におけるフローレーン内の実線の円は、そのフローレーンに対応するシステムコンポーネントがこの動作に能動的に参加している(例えば、他のシステムコンポーネントとメッセージを交換する)ことを示す。さらに、動作はそれが生じる時間的な順番にしたがいページの上から下へ並べられるのが一般的ではあるが、ある実施の形態では、動作のうちのひとつ以上は同時に生じてもよい。ある実施の形態は、図面に示されていない追加的なひとつ以上の動作を含んでもよく、および/または示されているひとつ以上の動作を含まなくてもよい。所定の動作においてシステムコンポーネントレベルのアクティビティを示すこの態様は、図3-6で用いられる。

【0052】

動作304において、ネイティブアプリケーション316などの第2クライアントアプリケーション(例えば、316)がクライアント104aで起動される。ネイティブアプリケーション316はサーバ102上のウェブアプリケーション324にアクセスするアプリケーションである。ある例示的な実施の形態によると、ネイティブアプリケーションは、動作中にサーバ102上のウェブアプリケーション324からデータを取得するおよび/または該ウェブアプリケーション324とデータを交換するスプレッドシートアプリケーション(例えば、マイクロソフト社のエクセル)である。

【0053】

ネイティブアプリケーション316は、例えばウェブアプリケーション324に関連付けられたウェブサーバとHTTPメッセージを交換することにより、HTTPなどを介してサーバ102と通信するために、ひとつ以上のブラウザコントローラのうちのいずれかを用いるよう構成されてもよい。ネイティブアプリケーションがひとつ以上のブラウザコントローラを利用可能であってもよい。ブラウザコントローラは、エクセルプログラムなどのアプリケーションが、ウェブを介した制御およびデータ交換のために用いられるHTTP、JSONおよび/または他のプロトコルを利用する能力を提供する。例えば、ネ

10

20

30

40

50

ティブアプリケーションは、インターネットエクスペローラ、クローム、ファイヤフォックス、サファリなどのうちのひとつ以上用の、市販のまたは無料で利用可能なブラウザコントローラを組み入れてもよい。ネイティブアプリケーションはまた、SSO機能に関連付けられた能力を提供するカスタマイズされたアドインコンポーネント(例えば、318)を含んでもよい。例えば、アドイン318は、ネイティブアプリケーション316内のエクセルスプレッドシートが、ひとつ以上のブラウザコントローラを用いてウェブアプリケーション324と相互作用することを可能とする。ある例示的な実施の形態では、カスタムアドイン318はマイクロソフトオートメーションやマイクロソフトCOMを用いて生成される。

【0054】

ネイティブアプリケーション316が起動された後、動作306において、ネイティブアプリケーションはクライアントにインストールされているブラウザ(例えば、312および314)のいずれかについて少なくともひとつの既存のセッションがあるか否かを特定する。インストール中および/またはアプリケーション起動中に、クライアントストレージのうち各特定のブラウザについて知られているディレクトリ構成の場所を探すことや所定の設定を探すことなどの技術によって、クライアント上で利用可能なブラウザを、ネイティブアプリケーションに知らせてもよい。代替的にまたは追加的に、ネイティブアプリケーションは、クライアントにインストールされるブラウザのそれぞれについてのブラウザコントロール(例えば、320および322)を特定し、自動的に組み入れてもよい。実施の形態によると、カスタムアドイン318はネイティブアプリケーションによる使用に利用可能なブラウザコントローラを繰り返す。例えば、利用可能なブラウザコントローラのそれぞれについて、設定可能な所定の順序で、またはネイティブアプリケーション316によってブラウザコントローラが発見された順序で、ネイティブアプリケーションはブラウザコントローラをインスタンス化し、サーバ102にHTTP要求を送信する。要求は、ユーザ、クライアントデバイス、ウェブアプリケーション、および現在選択されているブラウザコントロールに対応するブラウザの全てまたは組み合わせを特定してもよい。セッションが既に存在するブラウザタイプが見つかるまで、または、既存のセッションを見出すことなく全てのブラウザタイプが探索されてしまうまで、このプロセスを繰り返す。セッションが既に存在するブラウザタイプを特定することに関するさらなる詳細は図7に関して後述される。

【0055】

サーバ102がクライアント104aから要求を受け付けると、それは受け付けられた要求と共に提供された情報に対応する(例えば、HTTP要求に含まれているクッキー、ユーザ、クライアント、ウェブアプリケーション、およびブラウザのうちのいずれかまたはそれらの組み合わせのうちのいずれかに対応する)アクティブなセッションが存在するか否かを判定する。対応するセッションを特定するためにセッションテーブルを検索することにより、この判定がなされてもよい。例示的な実施の形態では、セッションが確立されるときに生成されるクッキーは、クライアントによって、サーバへの以後のHTTP要求のそれぞれにおいて送信され、サーバは受け付けたクッキーを用いることでセッションテーブル内の対応するセッションを一意に特定する。対応するセッションが現在アクティブであると判定された場合、対応するセッションに関連付けられたセッション情報および/またはクッキーがネイティブアプリケーションに返される。

【0056】

サーバにおいて、受け付けられたHTTP要求に対応するセッションが存在しないと判定された場合、サーバはネイティブアプリケーションに、その影響の通知を返す。ある例示的な実施の形態では、例えば、HTTPリダイレクト(例えば、「302リダイレクト」)メッセージが返される。ネイティブアプリケーションは、前のHTTP要求が既存の接続を特定するという結果を生じなかったとの通知を受けると、試行対象のブラウザのリスト内の次のブラウザに関する情報を含む新たなHTTP要求を形成する。

【0057】

10

20

30

40

50

ネイティブアプリケーションが利用可能なブラウザコントロールのいずれについても、それに対応する既存のセッションが見つからなかった場合、動作308において、サインオンの実行が成功し、新たなセッションおよび関連クッキーが生成される。サインオンは図5に関連して以下に詳述される。

【0058】

続いて、動作310において、ネイティブアプリケーションは、動作306において特定された既存のセッションまたは新たに生成されたセッション（動作308において生成されたもの）のいずれかを用いて動作する。ネイティブアプリケーションの動作は、ネイティブアプリケーションについて必要なデータを求めるHTTP要求を送信することおよびHTTP/JSONフォーマットのそのようなデータを受信することを含んでもよい。しかしながら、ネイティブアプリケーションは、ウェブアプリケーションなどのサーバ側アプリケーションと通信するためにHTTP/JSONを用いることに限定されない。サーバデバイスがウェブアプリケーションに対するデータ要求を受信すると、セッションの正当性がチェックされてもよい。

10

【0059】

図4は、複数ログイン防止（MLP）プロセス400が行われるときに生じる、クライアントデバイス104a、クライアントデバイス104bおよびサーバデバイス102の間のやりとりを示す。例えば、ユーザがあるクライアントデバイス（例えばクライアントデバイス104a）から既にログインしてウェブアプリケーション（例えば324）にアクセスしている状況で、そのユーザが第2クライアントデバイス（例えばクライアントデバイス104b）からサインインして同じウェブアプリケーションにアクセスする場合に、プロセス400は始まってもよい。

20

【0060】

動作402は、正当なクッキーを用いる確立されたセッションを介した、クライアントデバイス104aのユーザとサーバデバイス102上のウェブアプリケーションとの間の通信を表す。示されるように、クライアントデバイス104a上のブラウザおよび/またはネイティブアプリケーションのうちのひとつは確立されたセッションを介した通信を行ってもよい。

【0061】

動作402が続いている間のある時点で、動作404において、ユーザはクライアントデバイス104bでサインオンする。例えば、ユーザはクライアントデバイス104bのネイティブアプリケーションを介してログインしてもよい。

30

【0062】

動作406において、クライアントデバイス104bのネイティブアプリケーションは、HTTP要求を送信し、JSONフォーマットのデータを受信することによって、サーバデバイス上のウェブアプリケーションと通信する。上述の通り、要求元のネイティブアプリケーションに要求されたデータを送信する前に、セッションの正当性に対するサーバ側チェックが行われる。プロバイダデータへのアクセスおよびそのアクセスに関連付けられた認証についてのさらなる説明は、図8に関連して後述される。

【0063】

動作408はクライアントデバイス104aとサーバデバイス102との間のセッションのキャンセルを表す。以前の既存のセッションのキャンセルは、新たなセッションが生成されるのとはほぼ同時に行われてもよい。クライアントデバイス104aのセッションがキャンセルされた後は、クライアントデバイス104aからの、ウェブアプリケーションにデータを求める以降のHTTP要求は失敗する。クライアントデバイス104aのユーザインタフェースは、ウェブアプリケーションからのデータ取得に今後失敗することを示すよう、および/またはクライアントデバイス104aのセッションが他のロケーションからのログインに起因してキャンセルされたことをユーザに通知するよう、更新されてもよい。

40

【0064】

50

図5は、ある例示的な実施の形態に係る、サインオンプロセス500を実行している間に生じる、クライアントデバイス504（例えば、クライアントデバイス104a、104bと同様のもの）とサーバデバイス102との間のやりとりを示す。

【0065】

プロセス500は、ユーザがブラウザをウェブアプリケーションのいわゆる「ランディングページ」に辿り着かせるときに始ってもよい。例えば、「IR Insights」という名前のウェブアプリケーションについて、ランディングページはユーザに知られている「irinsight.nasdaq.com」などのアドレスにあってもよい。ランディングページのURLを入力することによって、または他のウェブページから対応するリンクをクリックすることによって、ナビゲーションが行われてもよい。

10

【0066】

動作502において、クライアントデバイス上のブラウザがランディングページに辿り着くと、ウェブアプリケーションは、対応する現在アクティブなセッションおよびクッキーが存在するか否かを判定する。アクティブなセッション（および/またはクッキー）が見つからなかった場合、ウェブアプリケーションは一時的なトークンを生成し、そのトークンをクライアントデバイスに送信する。ある実施の形態では、一時的なトークンは、ウェブアプリケーションのドメインクッキーと共にクライアントにプッシュされてもよい。

【0067】

動作504において、ウェブアプリケーションはクライアントデバイスをログインアプリケーションへリダイレクトする。ある実施の形態によると、クライアントデバイス上のブラウザをログインアプリケーションへ導く302HTTPリダイレクトメッセージをクライアントに送信することによって、リダイレクトが行われてもよい。リダイレクトメッセージと共にウェブアプリケーションの既知の識別子が渡されてもよい。この場合、ログインプロセスが完了した後、ログインアプリケーションは続いて、クライアントをウェブアプリケーションへと戻すようリダイレクトすることができる。ある実施の形態によると、ウェブアプリケーションに割り当てられたグローバル一意識別子（GUID）を、リダイレクトメッセージと共に渡される識別子として用いてもよい。

20

【0068】

動作506において、ブラウザはウェブアプリケーションからリダイレクトメッセージを受信しており、そのリダイレクトメッセージにおいて示されているアドレスにあるログインアプリケーションへと辿り着く。ログインアプリケーションにアクセスする際、ブラウザはHTTP要求に対応するGUIDを含めることにより、ウェブアプリケーション（これはブラウザをリダイレクトしたものである）を特定してもよい。ログインアプリケーションはユーザにログイン資格証明を促す。実施の形態によると、ログイン資格証明はユーザ名およびパスワードを含む。

30

【0069】

動作508において、クライアントはログイン資格証明を送信する。

【0070】

動作510において、ログインアプリケーションはログイン資格証明を検証する。例えば、ユーザ名およびパスワードは保持されているログインおよび/または認証情報に対して検証される。ユーザのログインが承認された後、ひとつ以上の現存のセッションはキャンセルされてもよい。この場合、関連するサービス明細に違反しない。ある実施の形態によると、ユーザがウェブアプリケーションにアクセスするための既存のセッションがキャンセルされてもよい。例えば、新たなセッションが確立される際のセッションのキャンセルに関して、先入れ先出し（FIFO）ポリシーが実装されてもよい。したがって、例えば、サービス明細が、特定のウェブアプリケーションに同時にアクセスできるクライアントアプリケーションの数を1に制限している場合、新たなセッションが生成されると、以前に確立されたセッションはキャンセルされる。他の実施の形態では、ウェブアプリケーションにアクセスするためのユーザの既存のセッションは、それらが新たなサインオンに関連付けられた同じ「領域（realm）」にある場合、キャンセルされる。「領域（realm）」

40

50

」は、GUIDおよび/またはGUID - クライアント組み合わせをグループ化するために用いられる概念である。例えば、ウェブアプリケーション324およびクライアントタイプ（例えば、デスクトップコンピュータ、スマートフォン）の特定の組み合わせを異なる領域に配置することによって、デスクトップコンピュータおよびスマートフォンの同じユーザによる同じウェブアプリケーション324への同時サインオンが許可されうる。

【0071】

動作512において、ログインアプリケーションはウェブアプリケーションに一時的なトークンを提供する。これはHTTP POSTメッセージを用いて、キー/値ペアの形態で一時的なトークンを指定することで、行われてもよい。ある実施の形態では、一時的なトークンと共に偽造防止トークンを提供してもよい。

10

【0072】

動作514において、ログインアプリケーションは、HTTPリダイレクトメッセージおよび以前に受信したウェブアプリケーションの識別子（例えば、GUID）を用いて、クライアントをウェブアプリケーションにリダイレクトする。ログインアプリケーションはまた、一時的なトークンを含むクッキーを生成してもよい。ある実施の形態では、クッキーはまた、偽造防止クッキーを含んでもよい。

【0073】

動作516において、クライアントはログインアプリケーションによってリダイレクトされており、ウェブアプリケーションに辿り着いて、一時的なトークンを含むひとつ以上のクッキーをウェブアプリケーションに提示する。

20

【0074】

動作518において、ウェブアプリケーションはトークンを検証する。

【0075】

動作520において、ウェブアプリケーションは永続的なトークンの生成をログインアプリケーションに要求する。

【0076】

動作522において、ログインアプリケーションは永続的なトークンを生成する。

【0077】

動作524において、ログインアプリケーションはウェブアプリケーションに生成された永続的なトークンを返す。

30

【0078】

動作526において、ウェブアプリケーションはクライアントに永続的なトークンを送信する。

【0079】

動作528において、クライアントは将来の使用のために永続的なトークンを格納する。ある実施の形態では、永続的なトークンを、対応するセッションを特定するためのクッキーとして用いられる。

【0080】

動作530は、クライアントによるHTTP要求の後続プロセスと、HTTP/JSONまたは他のフォーマットのメッセージを用いたサーバデバイスによる要求されたデータの返信を表す。ウェブアプリケーションは、要求されたデータを返す前に、永続的なトークンの引き続き正当性をチェックしてもよい。ある実施の形態では、例えば、クライアントによる個々全てのデータ要求について、ウェブアプリケーションおよび/または対応する外部サーバはセッションの正当性をチェックする。

40

【0081】

図6は、ある例示的な実施の形態に係る、SSOおよびMLPを合成された一連のやりとりで示す、クライアント104a、クライアント104bおよびサーバインフラストラクチャ102間のやりとりを示す。

【0082】

プロセス600は、ユーザがページ、ブラウザをウェブアプリケーションのいわゆる「

50

ランディングページ」に辿り着かせるときに始まってよい。ランディングページのURLを入力することによって、または他のウェブページから対応するリンクをクリックすることによって、ナビゲーションが行われてもよい。

【0083】

動作602において、クライアント104aのユーザはブラウザ312を介してウェブアプリケーション324にサインインする。サインオンプロセスは、上述の図5A-5Bにしたがい行われてもよい。図3の動作302に関連して説明した通り、ユーザがウェブアプリケーション324と相互作用するための新たなセッションが生成され、ひとつ以上の対応するクッキーが生成されてもよい。

【0084】

動作604において、ユーザがクライアント104bから有するアクティブなセッションは不活性化されるか除去される。例えば、動作602における新たなセッションのセットアップの一部として、同じユーザが他のクライアントから有するセッションは不活性化される。不活性化は、ひとつ以上のセッションがアクティブでないことを示すようセッションテーブルにマークをつけること、または、セッションテーブルからそのようなセッションを取り除いてしまうことを含んでもよい。

【0085】

動作602の時点でクライアント104bが継続中のアクティブなセッションを有していた場合、セッションテーブルにおけるそのセッションのエントリは除去されるか、または非アクティブとマークされてもよい。そのような除去または非アクティブとしてのマーキングの後、ウェブアプリケーション324に対するクライアント104bによる以降のデータ要求は成功しないであろう。これは、各データ要求に応じたサーバ側のセッション正当性チェックが、ウェブアプリケーションについてユーザおよびクライアント104bに関連付けられたアクティブなセッションを見つけれないことにより失敗するからである。これは、MLPの実装を示す。

【0086】

動作606において、クライアント104a上のアプリケーション316が起動する。この動作に関連付けられた処理は、図3の動作304に関連して説明されたものと同様である。

【0087】

動作608において、アプリケーションは既存のセッションを判定する。この動作に関連付けられた処理は、図3の動作306に関連して説明されたものと同様である。

【0088】

図7は、ある実施の形態に係る、ユーザがアプリケーションを介してウェブアプリケーションにアクセスすることを可能とするための、クライアント上で実行されるアプリケーションによって行われるプロセス700とひとつ以上のサーバ上で実行されるサーバプロセスとを示す。

【0089】

アプリケーション704（例えば、ウェブアプリケーション324にアクセスするための上述のネイティブアプリケーション316）がクライアント（例えば、クライアント104a）上で起動したときにプロセス700を開始してもよい。動作712において、起動時に、アプリケーション704のカスタムアドイン（例えば、アドイン318）は、使用可能なブラウザコントロールを決定する。ある実施の形態によると、ブラウザコントロールの所定の順序付けられたリストは使用可能なブラウザコントロールを表すものとして考慮されてもよい。他のある実施の形態では、クライアント上で利用可能なブラウザを自動的に発見してもよい。

【0090】

動作714において、第1ブラウザコントロールが選択される。ある実施の形態では、例えば、順序付けられたリストにおいて最初に現れるブラウザコントロールが選択される。例示的な実施の形態では、アプリケーションがエクセルプログラムを含む場合、インタ

10

20

30

40

50

ーネットエクスペローラブラウザコントローラを第1選択として選択する。これは、例えば、インターネットエクスペローラブラウザコントローラとエクセルプログラムとの間に高レベルの互換性が存在することが期待されるからである。

【0091】

動作716において、既存のセッションに対する要求がサーバに送信される。要求は、少なくとも、選択されたブラウザコントロールと、アクセス要求対象のウェブアプリケーションと、ユーザIDと、クライアント識別と、を示す。

【0092】

要求は、所定の指定ウェブアドレスに送信されてもよい。例えば、アプリケーションに組み入れられたカスタムアドインにより行われる処理を支援するために、企業はサーバ上のアプリケーションを所定の指定ウェブアドレスに設定してもよい。

10

【0093】

動作718において、指定ウェブアドレスにあるウェブサーバ706はクライアントにより送信された要求を受信する。ウェブサーバ706は(おそらくは、アプリケーションサーバ、データベースおよび/または他のプロセスと連携して)、クライアントから受信した要求に対応する既存のセッションに関する情報を要求している他のアプリケーションに、メッセージを、送信する。ウェブサーバ706および他のアプリケーションは、ある実施の形態では、コアサービス114内のアプリケーションであってもよい。

【0094】

動作720において、他のアプリケーション708はサーバ102上で実行されているアカウント管理アプリケーションであってもよく、他のアプリケーション708は要求を受信する。アプリケーション708は、既存のセッションが維持されているセッションテーブルや他のデータ構造をチェックすることで、受信した要求で指定されるパラメータに対応する既存のセッションを求めてもよい。受信した要求で指定されるパラメータは、ブラウザ(または上述の選択されたブラウザコントロール)のタイプ、ユーザ、クライアントおよび要求されたウェブアプリケーションの識別を含んでもよい。

20

【0095】

既存のセッションが見つかり、その既存のセッションに関連付けられたクッキーがアクセスされる。図2に関連して上述された通り、各既存のセッションは関連クッキーを有してもよい。

30

【0096】

動作722において、セッションが見つからなかった場合、アプリケーション708は、セッション発見に失敗したことを示しつつアプリケーション706に応答する。

【0097】

そうでなく、セッションが見つかった場合、動作722において、アプリケーション708は、既存のセッションについての情報および/またはアクセスされたクッキーのコピーを提供することによって、アプリケーション706に応答する。

【0098】

動作724において、アプリケーション706はアプリケーション708から受信した応答に基づいて、アプリケーション704に応答を送信する。したがって、動作720において既存のセッションが見つかった場合、既存のセッションについての情報および/または既存のセッションに対応するクッキーのコピーはアプリケーション704への応答に含まれる。動作720において既存のセッションが見つからなかった場合、応答は、アプリケーション708に、現在選択されているブラウザコントロールについて既存のセッションの発見に失敗した旨を知らせる。

40

【0099】

動作726において、アプリケーション704は、アドイン支援アプリケーション706から受信した最新の応答に基づいて、既存のセッションが発見されたか否かを判定する。

【0100】

50

既存のセッションが発見されなかった場合、アプリケーション704は、動作712で決定されたひとつ以上のブラウザコントロールからの次のブラウザコントロールの選択を試行する。次いで、既存のセッションについてまだテストされていない別のブラウザコントロールが存在する場合、動作728において、既存のセッションを決定すべき別のブラウザコントロールが選択され、処理は動作716に進む。動作716において、いまや、新たに選択されたブラウザコントロールを含む要求が生成されてもよく、処理は動作718に進んでもよい。

【0101】

動作726において、既存のセッションは発見されず、かつ、既存のセッションについてテストされるべきブラウザコントロールが残っていないと判定された場合、動作730において、アクセス対象のウェブアプリケーションについて、クライアントからのセッションは現在存在していないと判定され、アプリケーション704はサインオンを実行することができる。

10

【0102】

サインオンは、アプリケーション704がウェブアプリケーションの所定のアドレスにHTTP要求を送信することを含んでもよい。図5はクライアント上で実行されているブラウザがサインオンしているときに生じる例示的なプロセスを示す。アプリケーションがサインインするとき、サーバインフラストラクチャおよびクライアント上のアプリケーションによって、同様のプロセス（すなわち、同様の動作のシーケンス）が行われてもよい。サインオンプロセス中、例えば、アプリケーション704がログインプロンプト（コアサービス526から送信されるように示されている）を受信するとき、ユーザがログイン資格証明を入力するためのログインウィンドウが表示される。ある実施の形態によると、アプリケーションにおけるログインウィンドウは図9に関連して後述される。クライアント上のブラウザに対するセッションが生成される前に初めてアプリケーションがサインインするとき、新たなセッションが生成されるであろう。アプリケーションはこの新たなセッションを用いてウェブアプリケーションとやりとりする。

20

【0103】

一方、既存のセッションが見つかった場合、動作732において、アドイン支援アプリケーション706からの応答で受信したクッキーを、アプリケーションがアクセス可能なストレージに格納する。ウェブアプリケーションに対する以降の要求はそのクッキーを含んでもよい。したがって、既存のセッションを用いて、かつ、ユーザがアプリケーションを通じて別個のサインオンを行うことを要求せずに、ウェブアプリケーションがアプリケーション704にサービスを提供することが可能となる。

30

【0104】

ある実施の形態では、アプリケーション704がウェブアプリケーションによって提供されるアプリケーションデータにアクセスすることを始める前に、アプリケーション704によってさらに多くの動作が行われてもよい。ある実施の形態では、動作732の後（例えば、既存のセッションが発見され、ウェブアプリケーションにアクセスするためのクッキーがアプリケーションによって受信された後）、アプリケーション704は、別のブラウザコントロールをインスタンス化し、発見された既存のセッション（例えば、以前に取得されたクッキーを用いて）および新たにインスタンス化されたブラウザコントロールを用いて、ウェブアプリケーションにアクセスすることに進む。この能力は、クッキーをブラウザに亘って（例えば、ブラウザ間）で共有することを可能とする。例えば、クロームブラウザのサーバからクッキーを取得している状況で、しかしながらアプリケーションが他のタイプのブラウザコントローラ（例えば、インターネット 익스プローラブラウザコントローラ）で利用可能な所定のフィーチャを要求する場合、アプリケーションは、新たにインスタンス化された他のタイプのブラウザコントローラに既に取得されているクッキーを用いてもよい。

40

【0105】

図8は、ある例示的な実施の形態に係る、クライアント804上のアプリケーションま

50

たはクライアント側コードが外部サーバ810からのデータにアクセスするためのプロセス800を示す。

【0106】

プロセス800は、クライアント804上のアプリケーションまたはクライアント側コードがウェブアプリケーションにサインオンした後、クライアント上のアプリケーションまたはクライアント側コードによって外部サーバからのデータへのアクセスが要求されたときに始ってもよい。

【0107】

動作812において、クライアント804はサーバ102上のウェブアプリケーション806にデータ要求を送信する。要求は、セッションについて以前に取得されクライアントに保持されているクッキーを含んでもよい。要求は、要求されているデータを示す。

10

【0108】

動作814において、ウェブアプリケーション806はクライアント804からの要求を受信する。動作814における処理は、要求されているデータをどこから取得するかを決定することを含んでもよい。処理はまた、要求されているデータの少なくともいくらかをローカルで決定することを含んでもよい。

【0109】

動作816において、ウェブアプリケーションは、動作814において決定されたようによび/またはクライアント804からの要求で指定されるように、データ要求を外部サーバに送信する。

20

【0110】

動作818において、外部サーバ810は要求を受信する。ある実施の形態では、動作818における処理は正当なセッションについてローカルで維持されているテーブルをチェックすることを含んでもよい。チェックは、要求からのクッキーおよび/またはユーザ特定情報に基づくものであってもよい。ローカルで維持されているテーブルからのチェックが、ユーザがデータを受信することを承認することを示す場合、要求されているデータを提供するために処理は動作826に進んでもよい。

【0111】

他のある例示的な実施の形態では、ローカルで維持されているテーブルをチェックすることに加えてまたはその代わりに、動作820において、外部サーバはコアサービス808に、データ要求の確認を要求するための要求を送信する。

30

【0112】

動作822において、コアサービスアプリケーション808はセッションテーブルおよび/または他のデータ構造をチェックすることで、データ要求が承認されるか否かを判定する。

【0113】

動作824において、外部サーバに状態が報告される。

【0114】

動作826において、少なくともデータ要求が適切に承認されたかに基づいて、クライアントにデータを提供すると決定する。あるいはまた、要求が適切に承認されなかったと判定された場合、データを提供しないとの決定がなされる。

40

【0115】

動作828において、データがクライアントに提供されるべきか否かに基づいて、ウェブアプリケーション806に応答が送信される。応答は、要求されたデータを含むか、または失敗通知を含むか、のいずれかであってもよい。

【0116】

動作830において、ウェブアプリケーションは、クライアントの要求が成功したか失敗したかを判定する。

【0117】

動作832において、ウェブアプリケーションはクライアントに応答を送信する。

50

【 0 1 1 8 】

図 9 は、アプリケーション 1 2 0 a や 1 2 0 b などのアプリケーション内からユーザに提供されるログインウィンドウの例示的実装を示す。例えば、アプリケーション 1 2 0 a が実行されている（ウィンドウ 9 0 0 で示されるように）間に、クライアント 1 0 4 a のユーザが新たにサインインを行う必要があると判定された場合、ユーザのログイン資格証明を受け付けるためにログインウィンドウ 9 0 2 が表示される。

【 0 1 1 9 】

図 1 0 は、システム 1 0 0 のハードウェアアーキテクチャの非限定的例示的ブロック図を示す。図 1 0 に示される例では、クライアントシステム 1 0 1 0 はネットワーク 1 0 4 0 を介してサーバシステム 1 0 2 0 と通信する。ネットワーク 1 0 4 0 はインターネットなどの相互に接続された計算デバイスのネットワークを含みうる。ネットワーク 1 0 4 0 はローカルエリアネットワーク（LAN）を含んでもよく、またはクライアントシステム 1 0 1 0 とサーバシステム 1 0 2 0 との間のピアツーピア接続を含んでもよい。

【 0 1 2 0 】

例示的なクライアントシステム 1 0 1 0 とサーバシステム 1 0 2 0 とは、図 1 に示されるクライアント 1 0 4 とサーバ 1 0 2 とに対応しうる。すなわち、図 1 を参照して本明細書で説明され示された種々のソフトウェアコンポーネントおよび動作を実装するために、図 1 0 で説明されるハードウェア要素を用いることができる。例えば、図 1 0 のクライアントシステム 1 0 1 0 は、少なくともひとつのプロセッサ CPU 1 0 3 1 と、少なくともひとつのメモリ 1 0 3 2 と、少なくともひとつの入出力デバイス I/O 1 0 3 3 と、ユーザインタフェース UI 1 0 3 4 を生成して表示するためのコンポーネントと、を含んでもよい。少なくともひとつのメモリ 1 0 3 2 は、ランダムアクセスメモリ（RAM）やスタティック RAM やフラッシュメモリや磁気ディスクなどのコンピュータ可読保持媒体を含んでもよい。I/O デバイス 1 0 3 3 は、データを送受するトランシーバ（例えば、無線トランシーバ、有線トランシーバ）などの通信デバイスを全て包含し、それを含みうるものであってもよい。I/O デバイス 1 0 3 3 は、データを送受するために非一時的コンピュータ可読保持媒体をクライアントシステム 1 0 1 0 に接続するためのインタフェースを含んでもよい。

【 0 1 2 1 】

図 1 の例示的なウェブブラウザアプリケーション 1 1 7、1 1 8 およびアプリケーション 1 2 0 a を実装するためにクライアントシステム 1 0 1 0 の要素の組み合わせを用いることができることは、理解されるべきである。例えば、メモリ 1 0 3 2 はアプリケーションに関連付けられたファイル（例えば、HTML、XML、JavaScript ファイル）をロードし、CPU 1 0 3 1 はアプリケーションに関連付けられたインストラクションを実行するために用いられてもよい。I/O デバイス 1 0 3 3 を用いることで、サーバシステム 1 0 2 0 から SPA を含む種々の要素をフェッチすることができる、および/またはユーザと相互作用することができる。

【 0 1 2 2 】

サーバシステム 1 0 2 0 は、図 1 に示されるサーバ 1 0 2 または図 2 の 2 0 0 のソフトウェア要素を実装するために用いられる種々のハードウェアコンポーネントを備える。例えば、サーバシステム 1 0 2 0 は、少なくともひとつのプロセッサ CPU 1 0 2 1 と、少なくともひとつのメモリ 1 0 2 2 と、少なくともひとつの入出力デバイス I/O 1 0 2 3 と、のハードウェアコンポーネントを含んでもよい。少なくともひとつのメモリ 1 0 2 2 は、ランダムアクセスメモリ（RAM）やスタティック RAM やフラッシュメモリや磁気ディスクなどのコンピュータ可読保持媒体を含んでもよい。I/O デバイス 1 0 2 3 は、データを送受するトランシーバ（例えば、無線トランシーバ、有線トランシーバ）などの通信デバイスを全て包含し、それを含みうるものであってもよい。I/O デバイス 1 0 2 3 は、データを送受するために非一時的コンピュータ可読保持媒体をサーバシステム 1 0 2 0 に接続するためのインタフェースを含んでもよい。ひとつの例示的な実施の形態では、クライアントシステムの I/O デバイス 1 0 3 3 は、サーバシステムの I/O 1 0 2 3

10

20

30

40

50

とのネットワークを介した通信を行うことができる。

【 0 1 2 3 】

クライアントシステム 1 0 1 0 と同様に、サーバシステム 1 0 2 0 はアプリケーションを実装および/または実行することができる。例えば、ウェブアプリケーション 1 1 2 およびコアサービス 1 1 4 に関連付けられたウェブサーバおよびアプリケーションサーバによって用いられるコンポーネントおよびファイルに加えて、データベース 1 1 0 に情報を格納するためにメモリ 1 0 2 2 を用いることができる。CPU 1 0 2 1 は、クライアントシステム 1 0 1 0 により要求されそれに送信される対応モジュールを生成するのに必要なソフトウェアを実行する際に用いられてもよい。例えば、CPU 1 0 2 1 は、アプリケーションサーバによって生成される必要なモジュールを生成するために用いられてもよい。同様に、I/O デバイス 1 0 2 3 は、クライアントシステム 1 0 1 0 に異なるアプリケーション要素を送信するためにウェブサーバによって用いられてもよい。無論、これらの例は非限定的であり、システムはハードウェア要素を様々な態様で用いることを想定している。

10

【 0 1 2 4 】

説明された S S O および M L P に関して上で説明された技術は、とりわけ、ユーザが通信ネットワークを介してウェブアプリケーションとやりとりする際の効率を改善しつつ、ウェブアプリケーションのリソースの誤用を低減する能力を提供する。上で詳述された通り、ある実施の形態は、種々のネットワーク化リソースへのユーザアクセスを、そのようなりソースについての使用明細全体が守られるように、制御するために用いられうる。

20

【 0 1 2 5 】

本明細書の例では、説明を目的とし、限定を目的とせず、記載の技術の理解を提供するために、特定のノード、機能エンティティ、技術、プロトコル、規格等の具体的な詳細が説明される。他の実施の形態を、以下の詳細説明から離れて実施できることは当業者には明らかであろう。他の例では、周知の方法、デバイス、技術等の詳細説明は、不必要な詳細説明で説明を曖昧にしないために省略されている。個々の機能ブロックが図に示される。これらのブロックの機能は、個別のハードウェア回路（例えば、図 1 0 に示されるもの）を使用して、適切なプログラム化マイクロプロセッサあるいは汎用コンピュータと協働するソフトウェアプログラム（例えば、図 1 に示されるもの）およびデータを使用して、特定用途集積回路（A S I C）を使用して、および/または 1 つ以上のデジタル信号プロセッサ（D S P）を使用して、実現可能であることを当業者は理解するであろう。ソフトウェアプログラムインストラクションおよびデータは、コンピュータ可読保持媒体に保持されてもよく、このインストラクションがコンピュータあるいは他の適切なプロセッサ制御によって実行されると、コンピュータあるいはプロセッサは機能を実行する。以下でデータベースはテーブルとして示されている場合があるが、他のフォーマット（リレーショナルデータベース、オブジェクトベースモデル、および/または分散データベースを含む）を、データを保持し操作するために使用することができる。

30

【 0 1 2 6 】

処理ステップ、アルゴリズムあるいはその類が特定の順序で記載またはクレームされている場合があるが、このような処理は、異なる順序で動作するように構成設定することができる。換言すれば、明示されるかクレームされるいかなるシーケンスあるいはステップの順序も、そのステップがその順序で実行される必要性を示すものでは必ずしもない。本明細書で記載される処理のステップは、取り得る任意の順序で実行されうる。また、いくつかのステップは、非同時的に発生するように記載されているあるいは暗示されているが（例えば、あるステップは、他のステップの後に記載されているからである）、同時に実行されうる。さらに、図面における図示による処理の説明は、図示される処理が他の変形や変更について排他的であることを暗示しているのではなく、また、図示される処理あるいはその任意のステップが本技術に必要であることを暗示しているのではなく、また、図示される処理が好ましいことを暗示しているのでもない。

40

【 0 1 2 7 】

50

様々な形式のコンピュータ可読媒体 / 伝送は、プロセッサにデータ（例えば、インストラクションのシーケンス）を運ぶ際に含まれる。例えば、データは、(i) メモリからプロセッサへ送信されてもよく、(i i) 任意のタイプの送信媒体（例えば、有線、無線、光等）を介して運ばれてもよく、(i i i) イーサネット（あるいはIEEE 802.3）、ATP、ブルートゥース（登録商標）、TCP/IP、TDMA、CDMA、3G等のような、多くのフォーマット、規格あるいはプロトコルに従ってフォーマットおよび / または送信されてもよく、および / または(i v) 周知の様々な方法の任意のものでプライバシーを確保するためにあるいは流出を防止するために暗号化されてもよい。

【 0 1 2 8 】

本書で、動作が実行され「てもよく」、「ることができ」、もしくは「うる」ことと、機能、エレメント、もしくは構成が所与のコンテキストに含まれ「てもよく」、「ることができ」、もしくは「うる」、または適用可能なことと、所与のアイテムが所与の特性を有し「てもよく」、「ることができ」、もしくは「うる」こととが説明されるときはいつでも、または「～てもよい」、「～できる」もしくは「～しうる」という用語を含む任意の類似の語句が使用されるときはいつでも、所与の動作、特徴、エレメント、構成、特性、などが少なくとも1つの実施形態に示されているが、全ての実施形態に示される必要はないことが理解されたい。

10

【 0 1 2 9 】

AngularJSとの関連で技術が説明されたが、これは説明を容易にするためになされたものであって、本明細書で説明された技術が他のSPA技術や他のウェブ技術やおよび / または他のソフトウェア技術のコンテキストにおいて適用可能であることは理解される。

20

【 0 1 3 0 】

最も实际的で好適な実施の形態であると現時点で考えられているものとの関係で本技術が説明されたが、本技術は開示された実施の形態に限定されるものではなく、むしろ種々の変形例および等価な構成をカバーするよう意図されていることは理解されるべきである。

【図1】

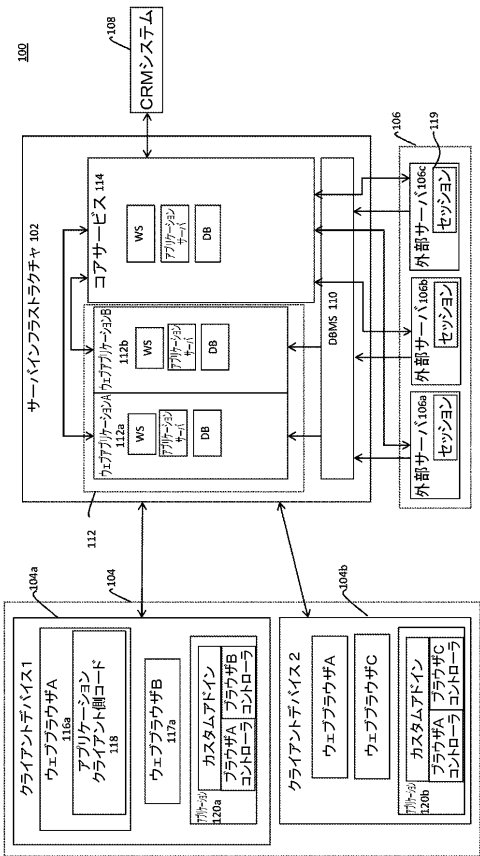


FIGURE 1

【図2】

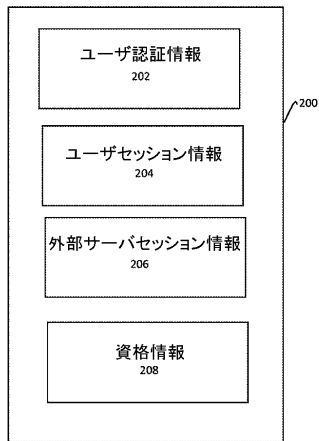


FIGURE 2

【図3】

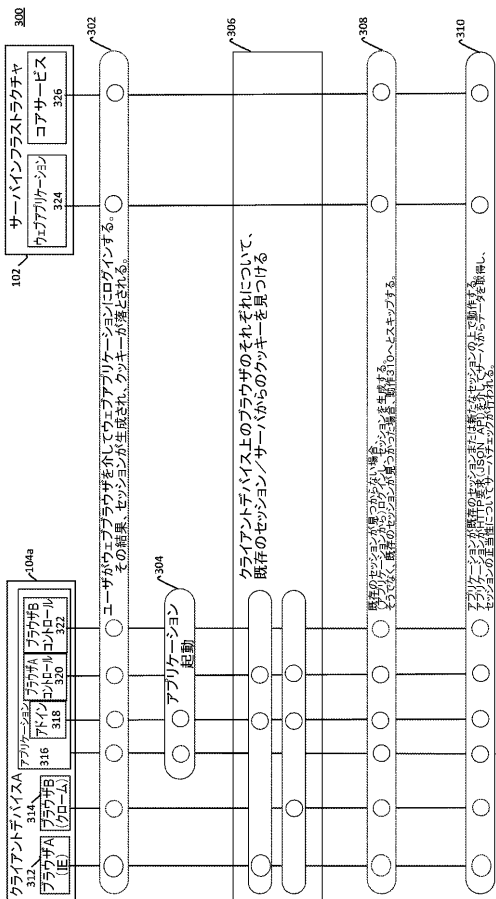


FIGURE 3

【図4】

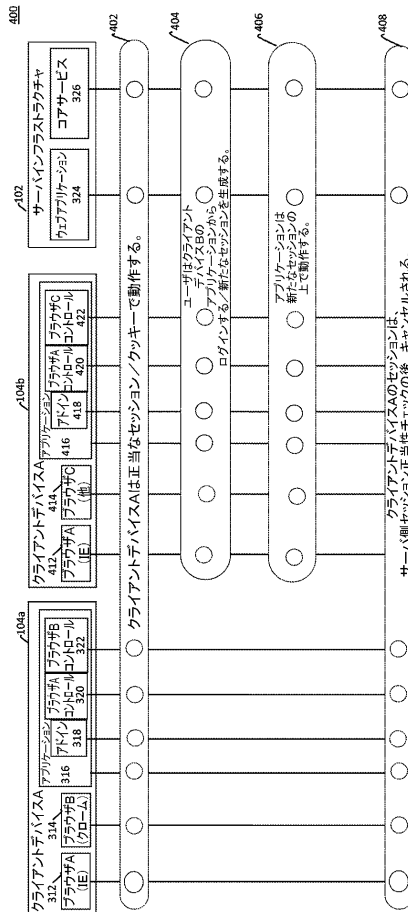


FIGURE 4

【 8 】

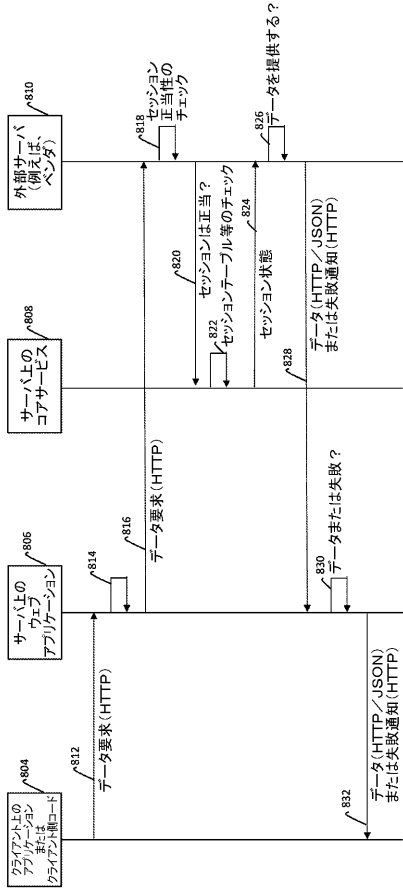


FIGURE 8

【 10 】

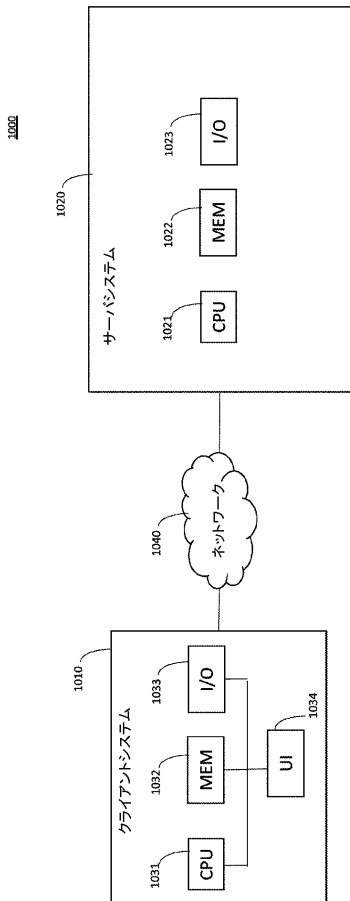


FIGURE 10

【 9 】

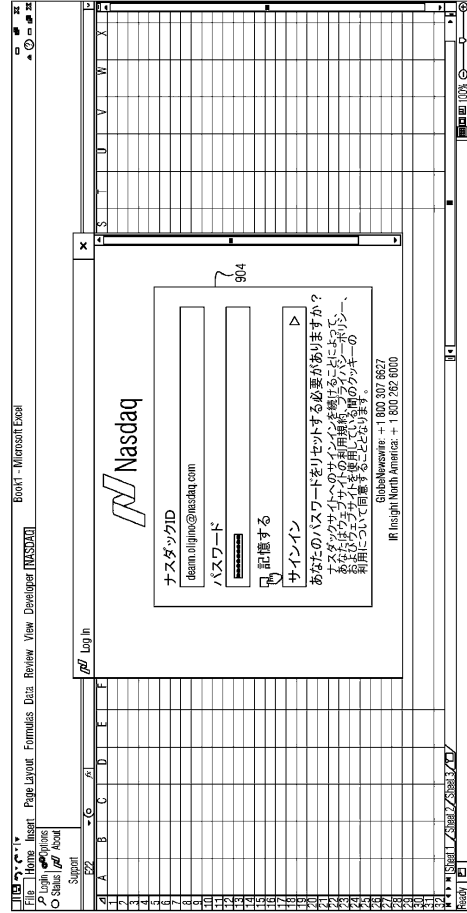


FIGURE 9

フロントページの続き

(74)代理人 100112508

弁理士 高柳 司郎

(74)代理人 100116894

弁理士 木村 秀二

(74)代理人 100130409

弁理士 下山 治

(74)代理人 100134175

弁理士 永川 行光

(72)発明者 ミテウスキ, ウラジミール

アメリカ合衆国 メリーランド州 20850, ロックヴィル, キング ファーム プールバ
ード 805, オージシー/アイピーアール デパートメント, ナスダック, インコーポ
レイテッド内

審査官 宮司 卓佳

(56)参考文献 米国特許出願公開第2012/0210413(US, A1)

米国特許出願公開第2014/0310173(US, A1)

米国特許出願公開第2006/0026260(US, A1)

特開2007-034956(JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 21/41

G06F 13/00