

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6580708号
(P6580708)

(45) 発行日 令和1年9月25日(2019.9.25)

(24) 登録日 令和1年9月6日(2019.9.6)

(51) Int. Cl. F I
G06F 16/28 (2019.01) G O 6 F 16/28
G06F 16/80 (2019.01) G O 6 F 16/80
G06F 13/00 (2006.01) G O 6 F 13/00 5 4 O F

請求項の数 20 (全 21 頁)

(21) 出願番号	特願2017-556914 (P2017-556914)	(73) 特許権者	502208397
(86) (22) 出願日	平成28年8月5日(2016.8.5)		グーグル エルエルシー
(65) 公表番号	特表2018-525698 (P2018-525698A)		アメリカ合衆国 カリフォルニア州 94
(43) 公表日	平成30年9月6日(2018.9.6)		043 マウンテン ビュー アンフィシ
(86) 国際出願番号	PCT/US2016/045722		アター パークウェイ 1600
(87) 国際公開番号	W02017/034780	(74) 代理人	100108453
(87) 国際公開日	平成29年3月2日(2017.3.2)		弁理士 村山 靖彦
審査請求日	平成29年11月30日(2017.11.30)	(74) 代理人	100110364
(31) 優先権主張番号	14/835, 182		弁理士 実広 信哉
(32) 優先日	平成27年8月25日(2015.8.25)	(74) 代理人	100133400
(33) 優先権主張国・地域又は機関	米国 (US)		弁理士 阿部 達彦
		(72) 発明者	ドン・ハ・リ
			アメリカ合衆国・カリフォルニア・940
			43・マウンテン・ビュー・アンフィシ
			アター・パークウェイ・1600
			最終頁に続く

(54) 【発明の名称】 対応するリソースへのアプリケーション部分ディープリンク

(57) 【特許請求の範囲】

【請求項1】

データ処理装置によって、リソースアドレスによって識別されるリソースがブラウザ環境においてレンダリングされるべきコンテンツを規定すると決定するステップであって、前記リソースが複数の部分識別子およびそれぞれの対応するネイティブアプリケーション部分ディープリンクを含み、各部分識別子が各他の部分識別子とは異なり、部分識別子ごとに、

前記部分識別子が、前記ブラウザ環境においてレンダリングされるべき前記コンテンツの適切なサブセットであるコンテンツの一部を識別し、前記コンテンツの適切なサブセットが前記他の部分識別子によって識別される前記コンテンツの適切なサブセットとは異なり、

前記ネイティブアプリケーション部分ディープリンクが、前記コンテンツの適切なサブセットを識別するために、前記部分識別子によって識別される前記コンテンツの一部をレンダリングするために使用されるべきネイティブアプリケーションと、前記部分識別子に一意に対応する前記ネイティブアプリケーションのための部分ディープリンクとを識別する、

ステップと、

前記データ処理装置によって、前記リソースアドレスを前記ネイティブアプリケーション部分ディープリンクにマッピングするマッピングを生成するステップと、

前記マッピングをマッピングインデックスに記憶するステップと

を備える、コンピュータ実装方法。

【請求項2】

ユーザデバイスにインストールされている前記ネイティブアプリケーションによる検索クエリに応答して、前記リソースの前記コンテンツの適切なサブセットのうちの1つを参照するネイティブアプリケーションディープリンクが前記ユーザデバイスに提供されるべきであると決定するステップと、

前記ネイティブアプリケーションディープリンクが前記ユーザデバイスに提供されるべきであるとの決定に応答して、前記コンテンツの適切なサブセットのうちの前記1つを参照する前記ネイティブアプリケーション部分ディープリンクを選択するために、前記マッピングインデックスにアクセスするステップと、

前記ネイティブアプリケーション部分ディープリンクを前記ユーザデバイスに提供するステップと

をさらに備える、請求項1に記載の方法。

【請求項3】

前記コンテンツの適切なサブセットのうちの前記1つを参照する前記ネイティブアプリケーション部分ディープリンクを選択するために前記マッピングインデックスにアクセスするステップが、

前記リソースの前記リソースアドレスがネイティブアプリケーション部分ディープリンクにマッピングされているかどうかを判定するために、前記マッピングインデックスにアクセスするステップと、

肯定的な判定に**応答して**、前記コンテンツの適切なサブセットへの前記参照に一致する部分識別子を含む前記ネイティブアプリケーション部分ディープリンクを選択するステップと

を備える、請求項2に記載の方法。

【請求項4】

前記検索クエリに対する検索結果が、前記ネイティブアプリケーション部分ディープリンクと、前記ネイティブアプリケーション部分ディープリンクにマッピングされた前記リソースアドレスへのリンクとを含む、請求項3に記載の方法。

【請求項5】

前記部分識別子が、バイトインデックスであるか、または前記リソースのコード内のフラグメント属性によって識別される、請求項1から4のいずれか一項に記載の方法。

【請求項6】

前記データ処理装置によって、前記リソース内の前記部分識別子によって指定されたコンテンツをキャプチャするステップであって、前記リソースが、前記ネイティブアプリケーション部分ディープリンクにマッピングされた前記リソースアドレスに関連付けられる、ステップと、

前記データ処理装置によって、前記リソース内の前記部分識別子によって指定された前記キャプチャされたコンテンツを記憶するステップと

をさらに含む、請求項1から5のいずれか一項に記載の方法。

【請求項7】

前記データ処理装置によって、前記ネイティブアプリケーション部分ディープリンクを含むページ内の前記コンテンツの少なくとも一部を識別するステップと、

前記データ処理装置によって、前記ページの前記識別されたコンテンツが、前記ネイティブアプリケーション部分ディープリンクにマッピングされた前記リソースアドレスに関連付けられる前記リソース内の前記コンテンツと一致すると判定するステップと

をさらに含む、請求項1から6のいずれか一項に記載の方法。

【請求項8】

前記マッピングインデックスから、前記リソースアドレスがマッピングされるリソースアドレスとネイティブアプリケーション部分ディープリンクとを備える少なくとも1つのアドレスペアにアクセスするステップと、

10

20

30

40

50

前記アドレスペアの前記ネイティブアプリケーション部分ディープリンクの選択に続いてネイティブアプリケーションによって提示されたコンテンツと、前記アドレスペアの前記リソースアドレスによって規定される前記リソースの一部との間の一貫性の検証を実行するステップと

をさらに含む、請求項1から7のいずれか一項に記載の方法。

【請求項9】

前記アドレスペアの前記ネイティブアプリケーション部分ディープリンクの選択に続いて前記ネイティブアプリケーションによって提示された前記コンテンツが、前記アドレスペアの前記リソースアドレスによって規定された前記リソースの前記一部と一致するという判定に対して、後で検索クエリに回答して検索結果を決定し提供するために使用するために、前記アドレスペアの前記ネイティブアプリケーション部分ディープリンクをインデックス付けすることによって回答するステップをさらに含む、請求項8に記載の方法。

10

【請求項10】

前記アドレスペアの前記ネイティブアプリケーション部分ディープリンクの選択に続いて前記ネイティブアプリケーションによって提示された前記コンテンツが、前記アドレスペアの前記リソースアドレスによって規定された前記リソースの前記一部と一致しないという判定に対して、検索クエリに回答して前記ネイティブアプリケーション部分ディープリンクが検索結果の一部として提示されることを防止するか、そうでなければ無効にすることによって回答するステップをさらに含む、請求項8または9に記載の方法。

20

【請求項11】

プロセッサと、

前記プロセッサに結合され、かつ命令を記憶したコンピュータ可読媒体と

を備え、前記命令は、前記プロセッサによって実行されると、前記プロセッサに、

リソースアドレスによって識別されるリソースが、ブラウザ環境においてレンダリングされるべきコンテンツを規定すると決定することであって、前記リソースが、複数の部分識別子およびそれぞれの対応するネイティブアプリケーション部分ディープリンクを含み、各部分識別子が各他の部分識別子とは異なり、部分識別子ごとに、

前記部分識別子が前記ブラウザ環境においてレンダリングされるべき前記コンテンツの適切なサブセットであるコンテンツの一部を識別し、前記前記コンテンツの適切なサブセットが前記他の部分識別子によって識別される前記コンテンツの適切なサブセットとは異なり、

30

前記ネイティブアプリケーション部分ディープリンクが、前記コンテンツの適切なサブセットを識別するために、前記部分識別子によって識別される前記コンテンツの一部をレンダリングするために使用されるべきネイティブアプリケーションと、前記部分識別子に一意に対応する前記ネイティブアプリケーションの部分ディープリンクとを識別する、決定することと、

前記リソースアドレスを前記ネイティブアプリケーション部分ディープリンクにマッピングするマッピングを生成することと、

前記マッピングをマッピングインデックスに記憶することと

を備える動作を実行させる、システム。

40

【請求項12】

前記動作が、

ユーザデバイスにインストールされている前記ネイティブアプリケーションによる検索クエリに回答して、前記リソースの前記コンテンツの適切なサブセットのうちの1つを参照するネイティブアプリケーションディープリンクが前記ユーザデバイスに提供されるべきであると決定することと、

前記ネイティブアプリケーションディープリンクが前記ユーザデバイスに提供されるべきであるとの決定に回答して、前記コンテンツの適切なサブセットのうちの前記1つを参照する前記ネイティブアプリケーション部分ディープリンクを選択するために、前記マッピングインデックスにアクセスすることと、

50

前記ネイティブアプリケーション部分ディープリンクを前記ユーザデバイスに提供することと

をさらに備える、請求項11に記載のシステム。

【請求項13】

前記コンテンツの適切なサブセットのうちの前記1つを参照する前記ネイティブアプリケーション部分ディープリンクを選択するために前記マッピングインデックスにアクセスすることが、

前記リソースの前記リソースアドレスがネイティブアプリケーション部分ディープリンクにマッピングされているかどうかを判定するために、前記マッピングインデックスにアクセスすることと、

肯定的な判定に回答して、前記コンテンツの適切なサブセットへの前記参照に一致する部分識別子を含む前記ネイティブアプリケーション部分ディープリンクを選択することとを備える、請求項12に記載のシステム。

【請求項14】

前記検索クエリに対する検索結果が、前記ネイティブアプリケーション部分ディープリンクと、前記ネイティブアプリケーション部分ディープリンクにマッピングされた前記リソースアドレスへのリンクとを含む、請求項13に記載のシステム。

【請求項15】

前記部分識別子が、バイトインデックスであるか、または前記部分識別子は、前記リソースのコード内のフラグメント属性によって識別される、請求項11から14のいずれか一項に記載のシステム。

【請求項16】

前記動作が、

データ処理装置によって、前記リソース内の前記部分識別子によって指定されたコンテンツをキャプチャすることであって、前記リソースが、前記ネイティブアプリケーション部分ディープリンクにマッピングされた前記リソースアドレスに関連付けられる、キャプチャすることと、

前記データ処理装置によって、前記リソース内の前記部分識別子によって指定された前記キャプチャされたコンテンツを記憶することと

をさらに含む、請求項11から15のいずれか一項に記載のシステム。

【請求項17】

前記動作が、

データ処理装置によって、前記ネイティブアプリケーション部分ディープリンクを含むページ内の前記コンテンツの少なくとも一部を識別することと、

前記データ処理装置によって、前記ページの前記識別されたコンテンツが、前記ネイティブアプリケーション部分ディープリンクにマッピングされた前記リソースアドレスに関連付けられる前記リソース内の前記コンテンツと一致すると判定することと

をさらに含む、請求項11から16のいずれか一項に記載のシステム。

【請求項18】

プロセッサによって実行されると、前記プロセッサに、

リソースアドレスによって識別されるリソースが、ブラウザ環境においてレンダリングされるべきコンテンツを規定すると決定することであって、前記リソースが、複数の部分識別子およびそれぞれの対応するネイティブアプリケーション部分ディープリンクを含み、各部分識別子が各他の部分識別子とは異なり、部分識別子ごとに、

前記部分識別子が、前記ブラウザ環境においてレンダリングされるべき前記コンテンツの適切なサブセットであるコンテンツの一部を識別し、前記コンテンツの適切なサブセットが、前記他の部分識別子によって識別される前記コンテンツの適切なサブセットとは異なり、

前記ネイティブアプリケーション部分ディープリンクが、前記コンテンツの適切なサブセットを識別するために、前記部分識別子によって識別される前記コンテンツの一部を

10

20

30

40

50

レンダリングするために使用されるべきネイティブアプリケーションと、前記部分識別子に一意に対応する前記ネイティブアプリケーションのための部分ディープリンクとを識別する、

決定することと、

前記リソースアドレスを前記ネイティブアプリケーション部分ディープリンクにマッピングするマッピングを生成することと、

前記マッピングをマッピングインデックスに記憶することと

を備える動作を実行させる命令を記憶する、コンピュータ可読媒体。

【請求項19】

コンピュータ実装方法であって、

ユーザデバイスにおいて、検索クエリに回答して、複数の検索結果を受信するステップであって、少なくとも1つの検索結果が部分ディープリンクを含み、

前記部分ディープリンクが、前記部分ディープリンクとは異なるリソースアドレスによって識別されるリソース内の部分識別子によって識別されるコンテンツの一部をレンダリングするために使用されるべきネイティブアプリケーションを識別し、

前記コンテンツの一部が、前記リソースアドレスにおいて前記リソースの要求に回答するとき、ユーザデバイスによってレンダリングされるべきコンテンツのサブセットであり、

前記検索結果の選択に回答して、前記部分ディープリンクが、前記ユーザデバイスに前記ネイティブアプリケーションをインスタンス化させ、かつ前記部分識別子によって識別される前記コンテンツの一部を要求させる、ステップと、

前記検索結果のうちの前記部分ディープリンクを含む前記少なくとも1つの検索結果を選択するステップと、

前記部分ディープリンクを含む前記少なくとも1つの検索結果の選択に回答して、前記コンテンツの一部を要求および受信し、かつ前記ネイティブアプリケーションのインスタンス内に前記コンテンツの一部を表示するステップと

を備える、方法。

【請求項20】

前記部分識別子が部分識別子マークアップタグであるか、または前記部分識別子がバイトオフセットである、請求項19に記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、対応するリソースへのアプリケーション部分ディープリンクに関する。

【背景技術】

【0002】

インターネットは、広範な情報へのアクセスを提供する。たとえば、デジタル画像ファイル、ビデオおよび/またはオーディオファイル、ならびに特定の話題または特定のニュース記事のウェブページリソースは、インターネットを介してアクセス可能である。ウェブページリソースに関しては、これらのリソースの多くは、バンキング、ホテル予約、ショッピングなどの特定の機能を実行することを容易にするように、またはオンライン百科事典、映画データベースなどの構造化情報を提供するように設計されている。

【0003】

インターネットを介してアクセス可能な特定のリソースを識別するための、様々な検索エンジンが利用可能である。これらの検索エンジンは、様々なウェブページリソースをクロールしてインデックス付けする。次いで、検索エンジンは、どのリソースが検索クエリに最も応答しているかを決定し、クエリに回答してリソースにリンクする検索結果を提供するために、インデックスを使用する。

【0004】

タブレットコンピュータおよびスマートフォンの出現により、ウェブページリソースの

10

20

30

40

50

使用によって容易にされた同じまたは同様の機能の実行を容易にする、および同じまたは同様のコンテンツを提供するネイティブアプリケーションが、現在大量に提供されている。

【発明の概要】

【課題を解決するための手段】

【0005】

本明細書は、ウェブページ上の対応するコンテンツを有するネイティブアプリケーションコンテンツを識別することに関する。

【0006】

一般に、本明細書に記載される主題の1つの革新的な態様は、リソースアドレスによって識別されるリソースが、ブラウザ環境においてレンダリングされるべきコンテンツを規定すると決定するステップであって、リソースが、複数の部分識別子およびそれぞれの対応するネイティブアプリケーション部分ディープリンクを含み、各部分識別子が他の部分識別子とは異なり、部分識別子ごとに、部分識別子が、ブラウザ環境においてレンダリングされるべきコンテンツの適切なサブセットであるコンテンツの一部を識別し、コンテンツの適切なサブセットが、他の部分識別子によって識別されるコンテンツの適切なサブセットとは異なり、ネイティブアプリケーション部分ディープリンクが、コンテンツの適切なサブセットを識別するために、部分識別子によって識別されるコンテンツの一部をレンダリングするために使用されるべきネイティブアプリケーションと、部分識別子に一意に対応するネイティブアプリケーションの部分ディープリンクとを識別する、ステップと、データ処理装置によって、リソースアドレスをネイティブアプリケーション部分ディープリンクにマッピングするマッピングを生成するステップと、マッピングをマップインデックスに記憶するステップとの動作を含む方法において具体化することができる。

【0007】

本明細書に記載される主題の別の革新的な態様は、検索クエリに応答してユーザデバイスにおいて複数の検索結果を受信するステップであって、少なくとも1つの検索結果が部分ディープリンクを含み、部分ディープリンクが、部分ディープリンクとは異なるリソースアドレスによって識別されるリソース内の部分識別子によって識別されるコンテンツの一部をレンダリングするために使用されるべきネイティブアプリケーションを識別し、コンテンツの一部が、リソースアドレスにおいてリソースの要求に応答するとき、ユーザデバイスによってレンダリングされるべきコンテンツのサブセットであり、検索結果の選択に応答して、部分ディープリンクが、ユーザデバイスにネイティブアプリケーションをインスタンス化させ、部分識別子によって識別されるコンテンツの一部を要求させる、ステップと、部分ディープリンクを含む検索結果を選択するステップと、選択に応答して、コンテンツの一部を要求および受信し、ネイティブアプリケーションのインスタンス内のコンテンツの一部を表示するステップとの動作を含む方法において具体化することができる。

【0008】

本明細書に記載される主題の特定の実施形態は、以下の利点のうちの1つまたは複数を実現するように実装することができる。ネイティブアプリケーション環境をウェブページの対応する部分にマッピングすることによって、検索エンジンによる、より正確で効率的なコンテンツ検証が可能になる。コンテンツ検証により、コンテンツのネイティブアプリケーションのリンクをインデックス付けする前に、アプリケーションによって提示されたコンテンツが、ウェブサイトの対応する部分のコンテンツと一貫性があることが保証される。検証によって、クエリに応答してネイティブのアプリケーション検索結果を提供する際に、検索エンジンがユーザの情報ニーズを満たす情報へのアクセスを提供することが保証される。ユーザの情報ニーズのより良い/より迅速な満足は、計算資源、特に、通常であれば情報ニーズを満たすためにユーザによって実行される追加の/より長いブラウジング中に使用される計算資源の使用を減少させる可能性がある。加えて、コンテンツ検証はウェブページの一部に基づいているので、検証プロセスは、対応するウェブページのすべ

10

20

30

40

50

てのデータを考慮する検証プロセスよりもロバストである。これにより、コンテンツ検証の失敗が少なくなり、検証プロセスのステップが少なくなり、ひいては、コンテンツの検証およびインデックス付けのために必要なコンピュータリソースが解放される。これにより、コンテンツのインデックス付けおよび検索の技術分野が改善される。

【0009】

いくつかの実装形態では、検索エンジンは、ウェブサイトおよびネイティブアプリケーションによって一貫した方法で提供されるコンテンツを1つのインデックスにインデックス付けするだけでよく、ウェブリソースのインデックスのためにあるインデックスを、およびネイティブアプリケーションのために別のインデックスを別々にインデックス付けする必要はない。これにより、2つのコーパスの全体的なメンテナンスが削減される。

10

【0010】

いくつかの実装形態では、検証によって、ウェブリソースに一貫したコンテンツを提供することが知られているネイティブアプリケーションのネイティブアプリケーション検索結果の提供が容易になる。したがって、ウェブリソースはネイティブアプリケーションのスコアリングプロキシとして使用することができる。これにより、検索結果のスコアリングとランク付けを行う際に、ネイティブアプリケーションと対応するリソースとの間の一貫したスコアリングが可能になる。

【0011】

最後に、様々なコンテンツ検証技法を使用することによって、検証プロセスは、ネイティブアプリケーションと対応するウェブリソースとの間の様々なフォーマットおよび表示の違いに関してロバストになる。

20

【0012】

本明細書に記載される主題の1つまたは複数の実施形態の詳細は、添付の図面および以下の説明に記載される。主題の他の特徴、態様、および利点は、説明、図面、および特許請求の範囲から明らかになるであろう。

【図面の簡単な説明】

【0013】

【図1A】ネイティブアプリケーション部分ディープリンクがリソースアドレスにマッピングされる環境のブロック図である。

【図1B】ウェブページにおけるコンテンツ部分、およびコンテンツ部分のうちのいくつかのネイティブアプリケーションのおよび対応する環境インスタンスを示すブロック図である。

30

【図2A】ネイティブアプリケーション部分ディープリンクをリソースアドレスにマッピングする例示的なプロセスのフローチャートである。

【図2B】部分ディープリンクのネイティブアプリケーションコンテンツを検証するための例示的なプロセスの流れ図である。

【図3】リソースのコンテンツの適切なサブセットの1つを参照するネイティブアプリケーション部分ディープリンクをユーザデバイスに提供するための例示的なプロセスのフローチャートである。

【図4】検索結果におけるネイティブアプリケーション部分ディープリンクを提供するための例示的なプロセスのフローチャートである。

40

【発明を実施するための形態】

【0014】

様々な図面における同様の参照番号および名称は、同様の要素を示す。

【0015】

本明細書で使用される場合、ネイティブアプリケーションは、ネイティブアプリケーションの環境内のユーザデバイス上に表示するための環境インスタンスを生成し、ユーザデバイス上のブラウザアプリケーションとは独立して動作する。ネイティブアプリケーションは、特定のユーザデバイスオペレーティングシステムおよびマシンファームウェア上で動作するように特に設計されたアプリケーションである。したがって、ネイティブアプリ

50

ケーションは、ブラウザベースのアプリケーションおよびブラウザレンダリングされたリソースとは異なる。後者は、インスタンス化またはレンダリングされるたびにウェブサーバからダウンロードされた、すべての、または少なくともいくつかの要素あるいは命令を必要とする。さらに、ブラウザベースのアプリケーションおよびブラウザレンダリングされたリソースは、ブラウザ内のすべてのウェブ対応モバイルデバイスによって処理することができ、したがってオペレーティングシステム固有のものではない。

【0016】

モバイルデバイスおよびタブレットデバイスのネイティブアプリケーションは、通常、デスクトップデバイスまたはラップトップデバイスの表示領域よりもはるかに小さい表示領域を備える。したがって、デスクトップサイズのディスプレイ用に公開され、複数の異なるセクションを含むリソースは、ディスプレイがより小さいデバイス上では、ビューポート外でレンダリングされるべきコンテンツの複数のセクションを持つことがよくある。ユーザにとって関心がある可能性がある情報を見つけるために、ユーザはビューポート内をナビゲートしなければならないため、これはユーザの経験を低下させる。この問題は、ユーザが面白いと感じた情報の断片をそれ自体が含む可能性がある検索結果の選択に回答してリソースが提供されたときに悪化する。

【0017】

本明細書では、ネイティブアプリケーション部分ディープリンクの存在を決定し、ネイティブアプリケーション部分ディープリンクに対応するリソースアドレスにマッピングするシステムを開示する。動作中、システムは、ブラウザ環境においてレンダリングされるべきコンテンツを規定するリソースアドレスによって識別されるリソースにアクセスする。本システムは、リソースが、部分識別子およびそれぞれの対応するネイティブアプリケーション部分ディープリンクを含むと決定する。各部分識別子は各他の部分識別子とは異なり、ブラウザ環境においてレンダリングされるべきコンテンツの一部を識別する。各部分識別子によって識別されるコンテンツは、他の部分識別子によって識別されるコンテンツとは異なる。ネイティブアプリケーション部分ディープリンクが、コンテンツの適切なサブセットを識別するために、部分識別子によって識別されるコンテンツの一部をレンダリングするために使用されるべきネイティブアプリケーションと、部分識別子に一一对応するネイティブアプリケーションの部分ディープリンクとを識別する。本システムは、リソースアドレスをネイティブアプリケーション部分ディープリンクにマッピングし、そのマッピングをインデックスに記憶する。次いで、本システムは、部分ディープリンクにアクセスするときにネイティブアプリケーションによって提供されるコンテンツを、対応する部分識別子によって識別されるリソースのコンテンツと比較することによって、ネイティブアプリケーション部分ディープリンクのコンテンツを検証してもよい。

【0018】

たとえば、レストランは、ウェブサイトおよび対応するネイティブアプリケーションを有してもよい。レストランウェブサイトでは、リソース(たとえば、ウェブページ)は、特定のレストランに関する情報を含んでもよく、またリソースの部分内にその特定のレストランのレビューが含まれてもよい。部分識別子は、リソースにおいて提供されるコンテンツの特定の部分を識別するために、リソースのハイパーテキストマークアップ言語(HTML)に含まれてもよい(たとえば、部分識別子がリソースのレビュー部分を識別してもよい)。部分識別子によって識別されるネイティブアプリケーション部分ディープリンクと、コンテンツの特定の部分が位置するリソースの関連するリソースアドレスとの間のマッピングを生成するために、たとえば、クローリング動作によってマッピングが実行されてもよい。後述するように、レストランのネイティブアプリケーションのネイティブアプリケーション部分ディープリンクが選択されると、ネイティブアプリケーションはディープリンクによって参照されるコンテンツを提示する。レストランレビュー部分コンテンツの場合、検索エンジンは、ネイティブアプリケーションのコンテンツを検証するために、部分識別子によってラベル付けされたウェブページのコンテンツ、ならびにネイティブアプリケーションによって提示されたコンテンツにアクセスしてもよい。

10

20

30

40

50

【 0 0 1 9 】

これらの特徴および追加の特徴は、以下でより詳細に説明される。

【 0 0 2 0 】

図1Aは、ネイティブアプリケーション部分ディープリンクがリソースアドレスにマッピングされる例示的な環境100のブロック図である。インターネットなどのコンピュータネットワーク102は、リソース発行者ウェブサイト104、アプリケーション発行者106、ユーザデバイス108、および検索システム130を接続する。

【 0 0 2 1 】

リソース発行者ウェブサイト104は、ドメインに関連付けられ、1つまたは複数の場所における1つまたは複数のサーバによってホストされる、1つまたは複数のウェブリソース105を含む。一般に、リソース発行者ウェブサイトは、テキスト、画像、マルチメディアコンテンツ、およびプログラミング要素を含むことができるハイパーテキストHTMLでフォーマットされたウェブページの集合体である。各ウェブサイト104は、ウェブサイト104を制御、管理、および/または所有するエンティティであるコンテンツ発行者によって維持される。

10

【 0 0 2 2 】

ウェブページリソースは、ネットワーク102を介して発行者ウェブサイト104によって提供されてもよく、リソースアドレス、たとえば、ユニフォームリソースロケータ(URL)を有する任意のデータである。ウェブリソースは、いくつかの例を挙げると、HTMLページ、画像ファイル、ビデオファイル、オーディオファイル、およびフィードソースであってもよい。リソースは、埋込み情報、たとえば、メタ情報およびハイパーリンク、ならびに/または埋込み命令、たとえばクライアント側スクリプトを含んでもよい。より一般的には、「リソース」はネットワークを介して識別可能なものであり、またネイティブアプリケーションも含むことができる。

20

【 0 0 2 3 】

アプリケーション発行者ウェブサイト106はまた、1つまたは複数のウェブリソース105を含んでもよく、またネイティブアプリケーション107を提供する。上述のように、ネイティブアプリケーション107は、特定のユーザデバイスオペレーティングシステムおよびマシンファームウェア上で動作するように特に設計されたアプリケーションである。ネイティブアプリケーション107は、異なるプラットフォーム上で動作するように設計された複数のバージョンを含んでもよい。たとえば、映画データベースウェブサイトに対応するネイティブアプリケーションは、第1のタイプのスマートフォン上で動作する第1のネイティブアプリケーション、第2のタイプのスマートフォン上で動作する第2のネイティブアプリケーション、第1のタイプのタブレット上で動作する第3のネイティブアプリケーションなどを含んでもよい。

30

【 0 0 2 4 】

本明細書で使用される場合、「環境インスタンス」は、ネイティブアプリケーション内の表示環境であり、そこにテキスト、画像などのコンテンツが表示される。環境インスタンスは、特定のネイティブアプリケーションに固有であり、ネイティブアプリケーションは、ユーザデバイス108の特定のオペレーティングシステムに固有である。環境インスタンスは、環境インスタンスがネイティブアプリケーション内で生成され、ネイティブアプリケーションに固有のものである点で、レンダリングされたウェブリソースとは異なるが、ウェブリソースは、ウェブページリソースに互換性があり、ユーザデバイスのオペレーティングシステムから独立している任意のブラウザ環境においてレンダリングされてもよい。

40

【 0 0 2 5 】

ユーザデバイス108は、ユーザの制御下にある電子デバイスである。ユーザデバイス108は、典型的には、ネットワーク102を介してウェブページリソース105およびネイティブアプリケーション107を要求および受信することができる。例示的なユーザデバイス108は、パーソナルコンピュータ、モバイル通信デバイス、およびタブレットコンピュータを含む

50

【0026】

様々なウェブサイトおよびネイティブアプリケーション(および、任意選択で、他のデータソース)を検索するために、ユーザデバイス108は、検索クエリを検索システム130に提出する。クエリに回答してウェブリソース105およびネイティブアプリケーション107を検索するために、検索システム130は、アプリケーションインデックス114およびウェブインデックス116にアクセスする。アプリケーションインデックス114は、ネイティブアプリケーション107のアプリケーションページのインデックスであり、アプリケーションデータ抽出器およびプロセッサ110、ならびにインデクサ112を使用して構築される。ウェブインデックス116は、たとえば、発行者ウェブサイト104をクロールすることから構築されたウェブリソース105のインデックスである。各クエリに回答して、検索システム130は、クエリに関連するリソースおよびネイティブアプリケーションを識別するために、ウェブインデックス116、および任意選択でアプリケーションインデックス114にアクセスする。別個のインデックスとして示されているが、ウェブインデックス116およびアプリケーションインデックス114は、単一のインデックスに組み合わせることができる。

10

【0027】

検索システム130は、たとえば、リソースおよびアプリケーションをそれぞれウェブリソース検索結果およびネイティブアプリケーション検索結果の形式で識別してもよい。生成された後で、検索結果は、クエリが受信されたユーザデバイス108に提供される。

【0028】

ウェブリソース検索結果は、検索システム130によって生成され、特定の検索クエリを満たすリソースのコンテンツに基づいてウェブリソースを識別するデータである。リソースのウェブリソース検索結果は、ウェブページタイトル、リソースから抽出されたテキストの断片、およびリソースのユニフォームリソース識別子(URI)、たとえばウェブページのユニフォームリソースロケータ(URL)を含むことができる。ウェブリソース検索結果は、ユーザデバイスにおいて選択されると、ユーザデバイスに、URLに位置するリソースを求める要求を生成させる。次いで、受信されたウェブリソースは、ブラウザアプリケーションに表示される。

20

【0029】

ネイティブアプリケーションの検索結果は、ネイティブアプリケーションと、任意選択でネイティブアプリケーションディープリンクを指定する。ネイティブアプリケーションの検索結果は、ユーザデバイスにおいて選択されると、ユーザデバイスにインストールされたネイティブアプリケーションに、一貫したコンテンツを要求させる。ネイティブアプリケーションが要求されたコンテンツを受信すると、ネイティブアプリケーションは、ネイティブアプリケーションのユーザインターフェースにコンテンツを表示する。

30

【0030】

いくつかの実装形態では、ウェブページの発行者は、ウェブページコード内に埋め込まれたネイティブアプリケーションの部分識別子およびディープリンクの使用によって、ウェブページ自体内のウェブページの特定の部分に対応するネイティブアプリケーションディープリンクを指定してもよい。そのようなディープリンクは、各部分ディープリンクがウェブページのコンテンツのサブ部分(すなわち、ウェブページコンテンツの適切なサブセット)に対応するので、「部分ディープリンク」と呼ばれる。マッピングシステム120は、ネイティブアプリケーションの部分識別子およびディープリンクにアクセスし、ウェブページリソースをそれぞれの部分ディープリンクにマッピングするマッピングデータ122を生成してもよい。

40

【0031】

部分識別子を有するウェブページと、対応するネイティブアプリケーションとの間の関係は図1Bに示されており、図1Bは、ウェブページ150内のコンテンツ部分152、154、および156、ならびにコンテンツ部分のうちのいくつかのネイティブアプリケーションの対応する環境インスタンス164および166を示すブロック図である。ウェブページ150をレンダ

50

リングするために使用されるコードによって規定されるコンテンツ部分154および156は、対応するネイティブアプリケーションの部分識別子および部分ディープリンクによって識別される。部分ディープリンクは、ネイティブアプリケーションによってそれぞれアクセスされると、ネイティブアプリケーションに、対応するコンテンツ164および166を表示させる。ウェブページ150の別の部分152は、部分識別子を有さず、したがって、ウェブページ150の部分152内のコンテンツに対応する部分ディープリンクは存在しない。

【 0 0 3 2 】

マッピングシステム120は、ウェブページURLとアプリケーションURIとの間の関係を識別するために、ウェブインデックス116およびアプリケーションインデックス114にアクセスしてもよい。具体的には、マッピングシステム120は、リソースアドレスとネイティブアプリケーション部分ディープリンクとの間の関係を識別し、その結果をマッピングデータ122に記憶してもよい。

10

【 0 0 3 3 】

マッピングデータを生成するための1つの例示的なプロセスを、図2Aを参照して説明する。図2Aは、ネイティブアプリケーション部分ディープリンクをリソースアドレスにマッピングする例示的なプロセス200のフローチャートである。プロセス200は、たとえば、マッピングシステム120によって実装され得る。いくつかの実装形態では、例示的なプロセス200の動作は、非一時的コンピュータ可読媒体に記憶された命令として実装されてもよく、命令は、データ処理装置に例示的なプロセス200の動作を実行させる。

【 0 0 3 4 】

20

プロセス200は、リソースが、複数の部分識別子および対応するネイティブアプリケーション部分ディープリンクを含むと決定する(202)。たとえば、マークアップ言語リソース、たとえば、XHTML、HTML、XMLの文脈において、1つまたは複数の部分識別子がリソースに含まれてもよい。いくつかの実装形態では、部分識別子は、リソースのコード内のフラグメント属性によって識別されてもよい。他のタイプのリソースコードについては、リソースのコンテンツのサブ部分を識別する任意のデータ構造が、部分識別子として使用されてもよい。部分識別子は、任意のノード、サブノード、部分ノード、または任意の他のタイプのハイパーテキスト要素であってもよい。加えて、特定の部分の開始点および終了点のバイトインデックスなどの他のタイプの部分識別子が使用されてもよい。

【 0 0 3 5 】

30

たとえば、android.example.comのURLを有するリソースの次のマークアップ言語を考えてみる。

```
<html>
<head>
...
<link rel="alternate" href="androidapp://com.example.android/foo" />
<link rel="alternate" deepLinkId="bar" href="androidapp:// com.example.android/fo
oo_bar" />
<link rel="alternate" deepLinkId="car" href="androidapp:// com.example.android/fo
oo_car" />
<link rel="alternate" deepLinkId="dar" href="androidapp:// com.example.android/fo
oo_dar" />
...
innerHTML A
</head>
<body>
innerHTML B
<deepLinkId id="bar" ...>
innerHTML C
</div>
```

40

50

```

<deeplinkid id="car" ...>
innerHTML D
</div>
<deeplinkid id="dar" ...>
innerHTML E
</div>
</body>

```

【 0 0 3 6 】

リソースは、リソースのコンテンツ全体に対する対応するディープリンクと、リソースの一部に対する3つの部分ディープリンクを含む。"link"タグは、ネイティブアプリケーション"androidapp"のページ全体のディープリンク"androidapp://com.example.android/foo"に使用される。ここでは、ページ全体のディープリンクは"foo"というステムを含む。このディープリンクの部分識別子がないため、ディープリンクはウェブページリソースに含まれるすべてのコンテンツを含む必要がある。

10

【 0 0 3 7 】

また、ネイティブアプリケーション用の3つの追加のディープリンク、"androidapp://com.example.android/foo_bar"、"androidapp://com.example.android/foo_car"、および"androidapp://com.example.android/foo_dar"がある。各それぞれの部分ディープリンクを宣言するために使用される各リンク属性は、属性"deeplinkid"によって識別される部分識別子を含む。マークアップ言語の場合、部分識別子はマークアップタグの形式であってもよい。それぞれの部分属性は、"bar"、"dar"、"car"である。ウェブページのコードでは、div属性"bar"、"dar"、および"car"があり、それぞれがHTMLのセクションを規定する。特に、div属性"bar"はinnerHTML Cに対応し、div属性"car"はinnerHTML Dに対応し、div属性"dar"はinnerHTML Dに対応する。

20

【 0 0 3 8 】

図示されている実装形態では、部分識別子の部分ディープリンクは、ディープリンク全体と、部分ディープリンクに付加された部分識別子とによって形成される。これは、システム120に、ネイティブアプリケーションによって示されるそれぞれのコンテンツを検証するために使用されるべきコンテンツは、ディープリンク"androidapp://com.example.android/foo_bar"にはinnerHTML Cのコンテンツであり、ディープリンク"androidapp://com.example.android/foo_car"にはinnerHTML Dのコンテンツであり、"androidapp://com.example.android/foo_dar"にはinnerHTML Eのコンテンツであることを示す。しかし、他の実装形態では、部分ディープリンクはまた、他の何らかの一意的テキストを有する部分識別子に一意的に関連付けることもできる。たとえば、以下のスキームを使用することができる。

30

```

<html>
<head>
...
<link rel="alternate" href="androidapp://com.example.android/foo" />
<link rel="alternate" deeplinkid="bar" href="androidapp:// com.example.android/A
A" />
<link rel="alternate" deeplinkid="car" href="androidapp:// com.example.android/B
B" />
<link rel="alternate" deeplinkid="dar" href="androidapp:// com.example.android/C
C" />
...
innerHTML A
</head>
<body>
innerHTML B

```

40

50

```

<deeplinkid id="bar" ...>
innerHTML C
</div>
<deeplinkid id="car" ...>
innerHTML D
</div>
<deeplinkid id="dar" ...>
innerHTML E
</div>
</body>

```

10

【 0 0 3 9 】

ここで、値AA、BB、およびCCは、部分識別子bar、car、およびdarにそれぞれ一意に対応する。

【 0 0 4 0 】

マッピングシステム120は、リソースに埋め込まれた部分識別子およびネイティブアプリケーションディープリンクを識別するために、検索システム130によって生成されたウェブインデックス116にアクセスしてもよい。

【 0 0 4 1 】

プロセス200は、リソースアドレスをネイティブアプリケーション部分ディープリンクにマッピングするマッピングを生成する(204)。部分識別子が見つかり、リソースアドレスとネイティブアプリケーション部分ディープリンクがマッピングされてもよく、マッピングデータ122のマッピングインデックスに記憶されてもよい。上記の例では、マッピングシステム120は、部分ディープリンクについて以下のマッピングを生成してもよい。

20

【表 1】

URL	"androidapp"の部分ディープリンク
Android.example.com#bar	com.example.androidapp/foo_bar
Android.example.com#car	com.example.androidapp/foo_car
Android.example.com#dar	com.example.androidapp/foo_dar

30

【 0 0 4 2 】

いくつかの実装では、マッピングシステム120は、リソースアドレスとネイティブアプリケーション部分ディープリンクとの間のマッピングとともに、ネイティブアプリケーションディープリンクの部分識別子によって識別されるリソースのコンテンツをマッピングデータ122に記憶してもよい。上記の例では、マッピングシステム120は、各deeplinkid属性(たとえば、部分識別子"bar"からの"innerHTML C"のコンテンツなど)によって識別されるデータおよびHTMLコンテンツをキャプチャしてもよい。

【 0 0 4 3 】

いくつかの実装形態では、リソースのコンテンツは、ディープリンクがアクセスされるときにネイティブアプリケーションによって実際に提供されるコンテンツであってもよく、すなわち、ネイティブアプリケーションは、部分識別子によって識別されるコードフラグメントをレンダリングしてもよい。他の実装形態では、リソースの一部のコンテンツは、ネイティブアプリケーションの部分ディープリンクによって提供されるコンテンツを検証するために使用される。検証は、ウェブリソースで提供されるネイティブアプリケーション部分ディープリンクが、それが対応するウェブリソースの一部と一貫性があるコンテンツを実際に参照することを保証するために使用されてもよい。

40

【 0 0 4 4 】

ネイティブアプリケーション部分ディープリンクと、リソースにおけるコンテンツの対応する適切なサブセットとの間で分析されたコンテンツは一貫性がないと判定された場合

50

、マッピングシステム120は、ネイティブアプリケーション部分ディープリンクが検索結果の一部として提示されることを防止するか、そうでなければ無効にしてもよい。

【0045】

図2Bは、部分ディープリンクのネイティブアプリケーションコンテンツを検証するための例示的なプロセス250の流れ図である。プロセス250は、プロセス200が実装されるのと同じ方法で、データ通信における1つまたは複数のコンピュータのデータ処理装置において実装され得る。

【0046】

プロセス250は、ネイティブアプリケーション用の第1の部分ディープリンクとウェブリソース用の第2のアドレスとのアドレスペアにアクセスする(252)。ペアごとに、第1の部分ディープリンクはネイティブアプリケーションによってアクセス可能であり、ネイティブアプリケーションが第1のコンテンツを受信する場所を指定し、各第2のアドレスは、それぞれが第2のコンテンツを提供するウェブリソースのアドレスである。やはり、アドレスペアは、上述したようにマッピングシステム120によって決定されてもよい。

【0047】

プロセス250は、第1の部分ディープリンクの第1のコンテンツを受信する(254)。マッピングシステム120は、たとえば、ネイティブアプリケーションを実行し、ネイティブアプリケーションのコンテンツ要求に回答して受信されるデータを監視してもよい。コンテンツは様々な方法で収集することができる。たとえば、データ処理装置は、ネイティブアプリケーションのテキストハンドラ、リストハンドラ、およびビューハンドラに、処理のための余分なテキストおよび画像データを問い合わせるように働くことができる。代替的または追加的に、コンテンツ要求に対する応答を監視し、応答からデータを抽出することができる。ネイティブアプリケーションに提供される、および/またはネイティブアプリケーションによって表示されるデータを収集する他の適切な方法も使用することができる。

【0048】

プロセス250は、第2のアドレスの第2のコンテンツを受信する(256)。ウェブリソースの場合、ウェブリソース内の部分識別子によって識別されるコンテンツが第2のコンテンツとして抽出される。

【0049】

プロセス250は、第1のコンテンツおよび第2のコンテンツが一貫したコンテンツであるかどうかを判定する(258)。エンティティマッチング、n-グラム類似性、フレーズマッチング、および特徴類似性などの1つまたは複数の比較プロセスが、第1のコンテンツと第2のコンテンツとの間の一貫性のしきい値レベルを示す場合、コンテンツは一貫性があると判定される。

【0050】

プロセス250が、第1のコンテンツと第2のコンテンツとが一貫したコンテンツであると判定した場合、プロセス250は、アドレスペアを検証し、検証データをリソースインデックスに記憶する(260)。たとえば、ウェブインデックス116内のURLのレコードは、ネイティブアプリケーション部分ディープリンクのアドレスと、ネイティブアプリケーションの識別子とを含むように更新されてもよい。検索システム130は、検索結果を後で処理するとき、URLによって参照されるウェブリソースがウェブ検索結果としてユーザデバイスに提供されるべきである場合、対応するネイティブアプリケーション検索結果を生成することができる。

【0051】

逆に、プロセス250が第1のコンテンツと第2のコンテンツとが一貫したコンテンツであると判定しない場合、プロセス250はアドレスペアを検証しない(262)。したがって、対応するウェブリソースと一貫性があるコンテンツを有していないと判定されたネイティブアプリケーションアドレスについては、そのアドレスに対するネイティブアプリケーション検索結果は生成されないことになる。

【0052】

10

20

30

40

50

部分ディープリンクが検証されると、検索システム130は、部分ディープリンクが対応するリソースのコンテンツがクエリに回答すると判定された場合に、部分ディープリンクを提供してもよい。図3は、リソースのコンテンツの適切なサブセットの1つを参照するネイティブアプリケーション部分ディープリンクをユーザデバイス108に提供するための例示的なプロセス300のフローチャートである。いくつかの実装形態では、例示的なプロセス300の動作は、非一時的コンピュータ可読媒体に記憶された命令として実装することができ、命令は、データ処理装置に例示的なプロセス300の動作を実行させる。

【0053】

リソースのコンテンツの適切なサブセットのうちの1つを参照するネイティブアプリケーションディープリンクは、ユーザデバイスに提供されると決定される(302)。たとえば、検索システム130は、部分ディープリンクが対応するリソースのコンテンツがクエリに回答すると決定され、ネイティブアプリケーション検索結果(または、ディープリンク)がユーザデバイスに提供されるべきであると決定してもよい。前者の決定は、部分識別子によって特定されたコンテンツの適切なサブセットのクエリに対する関連性スコアに基づいて行われてもよい。後者の決定は、たとえば、ユーザデバイスにインストールされているネイティブアプリケーションに基づいてもよい。

10

【0054】

プロセス300は、コンテンツの特定の適切なサブセットを参照するネイティブアプリケーション部分ディープリンクを選択するために、マッピングインデックスにアクセスする(304)。部分識別子によって識別されるコンテンツの適切なサブセットを参照するネイティブアプリケーション部分ディープリンクを突き止めるために、マッピングデータ122内のマッピングインデックスがアクセスされてもよい。

20

【0055】

次いで、プロセス300は、ネイティブアプリケーション部分ディープリンクをユーザデバイスに提供する(306)。部分ディープリンクは、別個のネイティブアプリケーション検索結果の一部として提供されてもよく、ウェブリソース検索結果における第2のリンクとして提供されてもよい。

【0056】

図4は、検索結果におけるネイティブアプリケーション部分ディープリンクを提供するための別の例示的なプロセス400のフローチャートである。プロセス400は、ウェブリソースコンテンツに含まれている部分識別子に回答して、部分ネイティブアプリケーション検索結果を提供することができる。

30

【0057】

プロセス400は、マッピングデータにアクセスし(402)、マッピングデータから、リソースのリソースアドレスがネイティブアプリケーション部分ディープリンクにマッピングされているかどうかを判定する(404)。

【0058】

リソースアドレスがネイティブアプリケーション部分ディープリンクにマッピングされている場合、コンテンツの適切なサブセットへの参照と一致する部分識別子を含むマッピングされたネイティブアプリケーション部分ディープリンクが選択され(406)、検索結果に提供される。しかしながら、検索結果に提供されたリソースアドレスがネイティブアプリケーション部分ディープリンクにマッピングされていないと判定された場合、リソースのリソースアドレスのみが検索結果に提供される(408)。

40

【0059】

本明細書に記載される主題および動作の実施形態は、本明細書に開示された構造およびそれらの構造的等価物を含む、デジタル電子回路、あるいはコンピュータソフトウェア、ファームウェア、またはハードウェア、あるいはそれらの1つまたは複数の組合せに実装することができる。本明細書に記載される主題の実施形態は、データ処理装置によって実行されるか、またはデータ処理装置の動作を制御するためにコンピュータ記憶媒体上に符号化された、1つまたは複数のコンピュータプログラム、すなわちコンピュータプログラ

50

ム命令の1つまたは複数のモジュールとして実装することができる。代替的または追加的に、本プログラム命令は、人工的に生成された伝播信号、たとえばデータ処理装置による実行のための適切なレシーバ装置への送信のために情報を符号化するために生成された機械生成電気信号、光信号、または電磁信号において符号化することができる。コンピュータ記憶媒体は、コンピュータ可読記憶デバイス、コンピュータ可読記憶基板、ランダムまたはシリアルアクセスメモリアレイまたはデバイス、あるいはそれらの1つまたは複数の組合せであり得るか、またはそれらに含まれ得る。さらに、コンピュータ記憶媒体は伝播信号ではないが、コンピュータ記憶媒体は、人工的に生成された伝播信号において符号化されたコンピュータプログラム命令の発信元または宛先であり得る。コンピュータ記憶媒体はまた、1つまたは複数の別個の物理的コンポーネントまたは媒体(たとえば、複数のCD 10、ディスク、または他の記憶デバイス)であり得るか、またはそれらに含まれ得る。

【0060】

本明細書に記載される動作は、1つまたは複数のコンピュータ可読記憶デバイスに記憶されたデータ、または他のソースから受信したデータに対してデータ処理装置によって実行される動作として実装することができる。

【0061】

「データ処理装置」という用語は、例として、プログラム可能なプロセッサ、コンピュータ、チップ上のシステム、または前述のものの複数のもの、あるいはそれらの組合せを含む、データを処理するためのあらゆる種類の装置、デバイス、および機械を包含する。本装置はまた、ハードウェアに加えて、問題のコンピュータプログラムの実行環境を生成 20するコード、たとえば、プロセッサファームウェアを構成するコード、プロトコルスタック、データベース管理システム、オペレーティングシステム、クロスプラットフォームランタイム環境、仮想マシン、またはそれらの1つまたは複数の組合せを含むことができる。本装置および実行環境は、ウェブサービス、分散コンピューティングおよびグリッドコンピューティングインフラストラクチャなどの、様々な異なるコンピューティングモデルインフラストラクチャを実現することができる。

【0062】

コンピュータプログラム(プログラム、ソフトウェア、ソフトウェアアプリケーション、スクリプト、またはコードとしても知られる)は、コンパイラ型言語またはインタープリタ型言語、宣言型言語または手続き型言語を含む、任意の形式のプログラミング言語で 30記述することができ、また、スタンドアロンプログラムとして、あるいはコンピューティング環境における使用に適したモジュール、コンポーネント、サブルーチン、オブジェクト、またはその他のユニットとして、を含む、任意の形式で展開することができる。コンピュータプログラムは、ファイルシステム内のファイルに対応してもよいが、必ずしも対応する必要はない。プログラムは、他のプログラムまたはデータ(たとえば、マークアップ言語文書に記憶された1つまたは複数のスクリプト)を保持するファイルの一部、問題のプログラム専用の単一のファイル、あるいは複数の調整されたファイル(たとえば、1つまたは複数のモジュール、サブプログラム、またはコードの一部を記憶するファイル)に記憶 40することができる。コンピュータプログラムは、1台のコンピュータ、または1つのサイトに配置されるか複数のサイトに分散され、通信ネットワークによって相互接続された複数のコンピュータ上で実行されるように展開することができる。

【0063】

本明細書に記載されるプロセスおよび論理フローは、入力データを操作して出力を生成することによって動作を実行するために1つまたは複数のコンピュータプログラムを実行する1つまたは複数のプログラム可能なプロセッサによって実行することができる。コンピュータプログラムの実行に適したプロセッサは、一例として、汎用マイクロプロセッサおよび専用マイクロプロセッサの両方、ならびに任意の種類デジタルコンピュータの任意の1つまたは複数のプロセッサを含む。一般に、プロセッサは、読み出し専用メモリまたはランダムアクセスメモリ、あるいはその両方から命令およびデータを受信する。コンピュータの本質的要素は、命令に従って動作を実行するためのプロセッサと、命令およびデ 50

ータを記憶するための1つまたは複数のメモリデバイスとである。一般に、コンピュータはまた、データを記憶するための1つまたは複数の大容量記憶デバイス、たとえば磁気ディスク、光磁気ディスク、または光ディスクを含むか、それらからデータを受信するように、またはそれらにデータを転送するように、あるいはその両方を行うように動作可能に結合される。しかしながら、コンピュータはそのようなデバイスを有する必要はない。さらに、コンピュータは、別のデバイス、たとえばいくつか例を挙げると、モバイル電話、携帯情報端末(PDA)、モバイルオーディオまたはビデオプレーヤ、ゲーム機、全地球測位システム(GPS)レシーバ、あるいはポータブル記憶デバイス(たとえば、ユニバーサルシリアルバス(USB)フラッシュドライブ)に組み込まれ得る。コンピュータプログラム命令およびデータを記憶するために適したデバイスは、例としてたとえば、EPROM、EEPROM、およびフラッシュメモリデバイスなどの半導体メモリデバイスを含む、すべての形態の不揮発性メモリ、媒体、およびメモリデバイス、内部ハードディスクまたはリムーバブルディスクなどの磁気ディスク、光磁気ディスク、ならびにCD-ROMおよびDVD-ROMディスクを含む。プロセッサおよびメモリは、専用論理回路によって補完されてもよく、それに組み込まれてもよい。

【0064】

ユーザとの対話を提供するために、本明細書に記載される主題の実施形態は、たとえば、情報をユーザに表示するためのCRT(陰極線管)またはLCD(液晶ディスプレイ)モニタなどのディスプレイデバイス、ならびにユーザがコンピュータに入力を提供することができる、キーボード、および、たとえばマウスまたはトラックボールなどのポインティングデバイスを有するコンピュータ上で実装することができる。ユーザとの対話を提供するために他の種類のデバイスをも使用することができる、たとえば、ユーザに提供されるフィードバックは、任意の形態の感覚フィードバック、たとえば、視覚フィードバック、聴覚フィードバック、または触覚フィードバックであってもよく、ユーザからの入力は、音響、音声、または触覚入力を含む任意の形態で受信することができる。加えて、コンピュータは、ユーザによって使用されるデバイスとの間で文書を送受信することによって、たとえば、ウェブブラウザから受信した要求に回答してユーザのユーザデバイス上のウェブブラウザにウェブページを送信することによって、ユーザと対話することができる。

【0065】

本明細書に記載される主題の実施形態は、たとえば、データサーバとしてのバックエンドコンポーネントを含む、あるいは、たとえばアプリケーションサーバなどのミドルウェアコンポーネントを含む、あるいは、フロントエンドコンポーネント、たとえばグラフィカルユーザインターフェースを有するユーザコンピュータ、またはユーザが本明細書に記載される主題の実装と対話することができるウェブブラウザを含むコンピューティングシステム、あるいは1つまたは複数のそのようなバックエンド、ミドルウェア、またはフロントエンドコンポーネントの任意の組合せにおいて実装することができる。本システムのコンポーネントは、任意の形式または媒体のデジタルデータ通信、たとえば、通信ネットワークによって相互接続することができる。通信ネットワークの例は、ローカルエリアネットワーク("LAN")およびワイドエリアネットワーク("WAN")、相互接続ネットワーク(たとえば、インターネット)、ならびにピアツーピアネットワーク(たとえば、アドホックピアツーピアネットワーク)を含む。

【0066】

コンピューティングシステムは、ユーザおよびサーバを含むことができる。ユーザおよびサーバは、一般に、互いに遠隔であり、典型的には通信ネットワークを通じて対話する。ユーザとサーバとの関係は、それぞれのコンピュータ上で動作し、互いにユーザ-サーバ関係を有するコンピュータプログラムによって生じる。いくつかの実施形態では、サーバは、(たとえば、ユーザデバイスと対話するユーザにデータを表示し、また、そこからユーザ入力を受信する目的で)ユーザデバイスにデータ(たとえば、HTMLページ)を送信する。ユーザデバイスにおいて生成されたデータ(たとえば、ユーザ対話の結果)は、サーバにおいてユーザデバイスから受信することができる。

10

20

30

40

50

【 0 0 6 7 】

本明細書は、多くの具体的な実装形態の詳細を含むが、これらは、任意の発明の範囲または特許請求される可能性のあるものの範囲の限定として解釈されるべきではなく、むしろ特定の発明の特定の形態に特有の特徴の説明として解釈されるべきである。別個の実施形態の文脈において本明細書に記載される特定の形態もまた、単一の実施形態において組み合わせられて実装することができる。逆に、単一の実施形態の文脈において記載される様々な特徴もまた、複数の実施形態において別々に、または任意の適切なサブコンビネーションにおいて実装することができる。さらに、特徴は、特定の組合せにおいて作用するものとして上記で説明されており、最初はそのように特許請求されているが、場合によっては、特許請求された組合せからの1つまたは複数の特徴を組合せから切り出すことができ、特許請求された組合せは、サブコンビネーション、またはサブコンビネーションの変形に導かれてよい。

10

【 0 0 6 8 】

同様に、動作は特定の順序で図面に示されているが、これは、所望の結果を達成するために、そのような動作が示された特定の順序で、または順番に実行されること、あるいは図示されたすべての動作が実行されることを必要とするものとして理解されるべきではない。特定の状況では、マルチタスク処理および並列処理が有利な場合がある。さらに、上述の実施形態における様々なシステムコンポーネントの分離は、すべての実施形態においてそのような分離を必要とするものとして理解されるべきではなく、記載されたプログラムコンポーネントおよびシステムは、一般に、単一のソフトウェア製品と一緒に統合することができ、または複数のソフトウェア製品にパッケージ化することができることが理解されるべきである。

20

【 0 0 6 9 】

このように、主題の特定の形態について説明した。他の実施形態は、添付の特許請求の範囲内にある。場合によっては、特許請求の範囲に記載された動作を異なる順序で実行することができ、依然として所望の結果を達成することができる。加えて、添付の図面に示されるプロセスは、所望の結果を達成するために、示された特定の順序、または順番を必ずしも必要としない。特定の形態では、マルチタスク処理および並列処理が有利な場合がある。

30

【 符号の説明 】

【 0 0 7 0 】

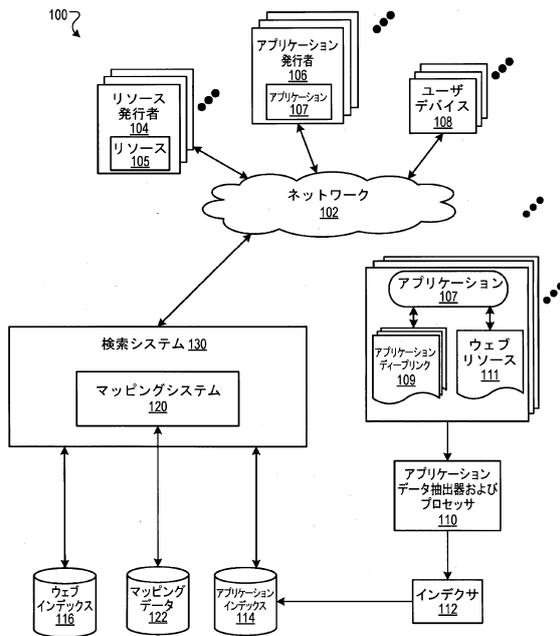
- 100 環境
- 102 コンピュータネットワーク
- 104 リソース発行者ウェブサイト
- 105 ウェブリソース
- 106 アプリケーション発行者
- 106 アプリケーション発行者ウェブサイト
- 107 ネイティブアプリケーション
- 108 ユーザデバイス
- 110 アプリケーションデータ抽出器およびプロセッサ
- 112 インデクサ
- 114 アプリケーションインデックス
- 116 ウェブインデックス
- 120 マッピングシステム
- 122 マッピングデータ
- 130 検索システム
- 150 ウェブページ
- 152 コンテンツ部分
- 154 コンテンツ部分
- 156 コンテンツ部分

40

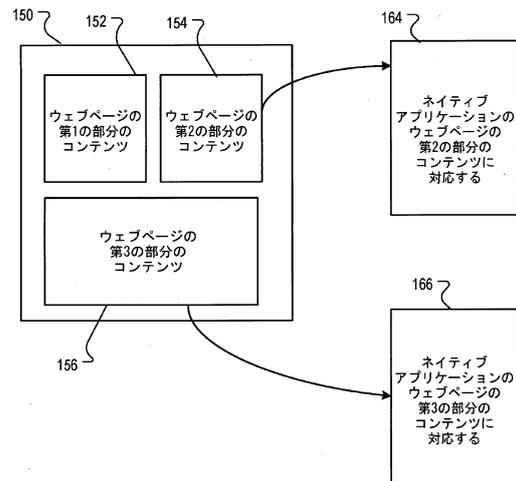
50

- 164 環境インスタンス
- 166 環境インスタンス
- 200 プロセス
- 250 プロセス
- 300 プロセス
- 400 プロセス

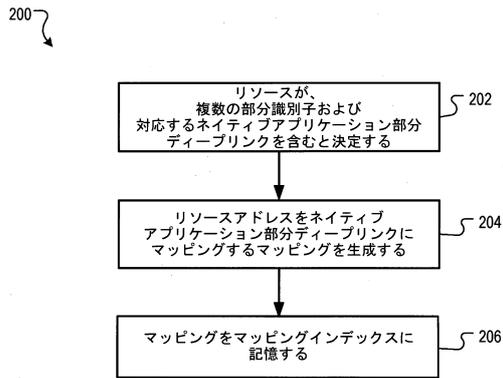
【図1A】



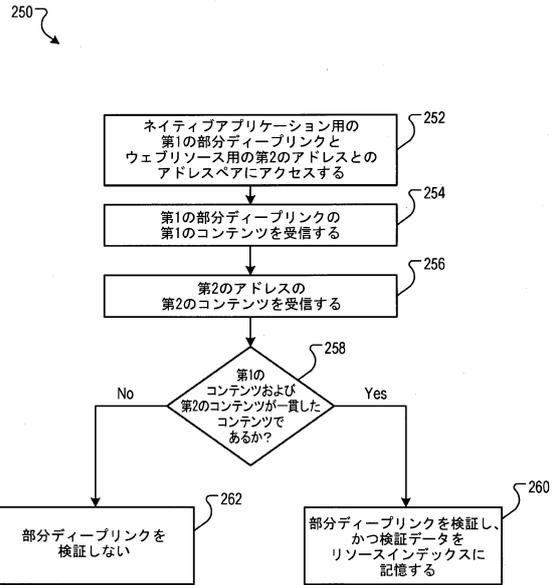
【図1B】



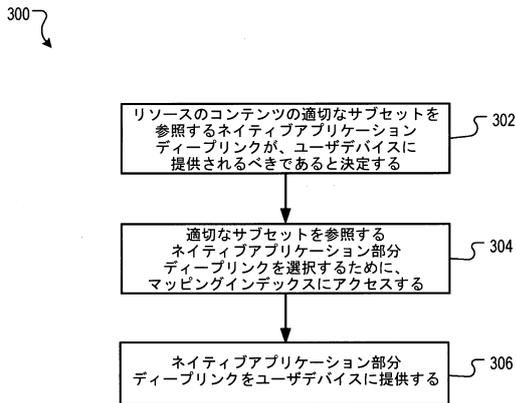
【図 2 A】



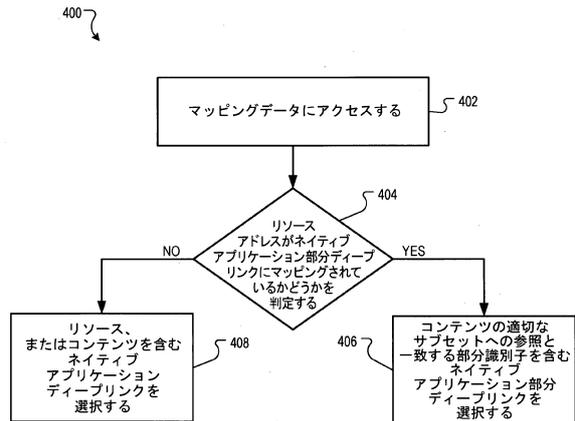
【図 2 B】



【図 3】



【図 4】



フロントページの続き

(72)発明者 ジェヒョン・ヨム
アメリカ合衆国・カリフォルニア・94043・マウンテン・ビュー・アンフィシアター・パーク
ウェイ・1600

審査官 鹿野 博嗣

(56)参考文献 米国特許出願公開第2013/0304729 (US, A1)
米国特許第08042036 (US, B1)
特表2015-501496 (JP, A)
米国特許出願公開第2014/0250106 (US, A1)
特表2009-530736 (JP, A)

(58)調査した分野(Int.Cl., DB名)
G06F 16/28
G06F 13/00
G06F 16/80