

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6211722号

(P6211722)

(45) 発行日 平成29年10月11日(2017.10.11)

(24) 登録日 平成29年9月22日(2017.9.22)

(51) Int.Cl.

F I

G 0 6 F 13/00 (2006.01)

G 0 6 F 13/00 5 4 0 B

請求項の数 16 (全 23 頁)

(21) 出願番号	特願2016-574382 (P2016-574382)	(73) 特許権者	502208397
(86) (22) 出願日	平成26年6月26日(2014.6.26)		グーグル インコーポレイテッド
(65) 公表番号	特表2017-527005 (P2017-527005A)		アメリカ合衆国 カリフォルニア州 94
(43) 公表日	平成29年9月14日(2017.9.14)		043 マウンテン ビュー アンフィシ
(86) 国際出願番号	PCT/CN2014/080801		アター パークウェイ 1600
(87) 国際公開番号	W02015/196405	(74) 代理人	100108453
(87) 国際公開日	平成27年12月30日(2015.12.30)		弁理士 村山 靖彦
審査請求日	平成28年12月28日(2016.12.28)	(74) 代理人	100110364
			弁理士 実広 信哉
早期審査対象出願		(74) 代理人	100133400
			弁理士 阿部 達彦

最終頁に続く

(54) 【発明の名称】最適化されたブラウザレンダリング処理

(57) 【特許請求の範囲】

【請求項1】

コンピュータシステムであって、
少なくとも1つのプロセッサと、
前記少なくとも1つのプロセッサによって実行された場合、前記システムに対して、
フェッチレコードに記憶された複数のURLに共通のURLパターンを識別させ、
前記複数のURLからURLのサンプルを選択させ、
前記サンプルにおける各URLについて、
前記URLのためのコンテンツを使用して、前記URLのエンベッタのための第1のレンダリング結果を生成し、
前記URLのための前記コンテンツを使用することなく、前記エンベッタのための第2のレンダリング結果を生成し、
前記第1のレンダリング結果を前記第2のレンダリング結果と比較することによって、
前記URLのための類似性スコアを計算し、
前記類似性スコアが類似性しきい値よりも大きい場合、前記URLがオプションであると判定する
ことによって、前記URLがオプションであるか否かを判定させ、
前記サンプルにおけるあらかじめ定められた数の前記URLが、オプションである場合、
オプションリソースパターンのデータストアに前記URLパターンを記憶させる、命令を記憶したメモリと

10

20

を備える、システム。

【請求項2】

前記命令はさらに、前記少なくとも1つのプロセッサによって実行された場合、前記システムに対して、

要求された埋込リソースのコンテンツを求める要求を受信させ、

前記要求された埋込リソースが前記データストア内の前記URLパターンに一致するか否かを判定させ、

前記要求された埋込リソースが前記URLパターンに一致する場合、前記要求された埋込リソースがオプションであるというインジケーションを返させる

命令を含む、請求項1に記載のシステム。

10

【請求項3】

前記インジケーションは、エラーの発見されていないURLである、請求項2に記載のシステム。

【請求項4】

前記URLの前記数は、前記サンプルにおけるURLの数に等しい、請求項1に記載のシステム。

【請求項5】

前記フェッチレコードに記憶された前記複数のURLに共通の前記URLパターンを識別するステップは、

前記URLから少なくともクエリ文字列の一部を削除することによって、前記フェッチレコードに記憶された各々のURLのためのグループURLを生成するステップと、

グループURLによって前記URLをクラスタ化するステップとを含む、請求項1に記載のシステム。

20

【請求項6】

前記フェッチレコードに記憶された前記複数のURLに共通の前記URLパターンを識別するステップは、

前記URLから少なくともクエリ文字列の一部を削除することによって、前記フェッチレコードに記憶された各々のURLのためのグループURLを生成するステップと、

グループURLによって前記URLをクラスタ化するステップと、

最多数のメンバを有するクラスタの前記グループURLを前記URLパターンとして選択するステップと

を含む、請求項1に記載のシステム。

30

【請求項7】

前記類似性スコアを計算するステップは、

前記第1のレンダリング結果のDOMツリー、および前記第2のレンダリング結果のDOMツリーのための最長共通シーケンスを決定するステップと、

前記類似性スコアを決定するために前記最長共通シーケンスを使用するステップとを含む、請求項1に記載のシステム。

【請求項8】

少なくとも1つのプロセッサを使用して、フェッチレコードに記憶された複数のURLに共通のURLパターンを識別するステップと、

前記少なくとも1つのプロセッサを使用して、前記複数のURLからURLのサンプルを選択するステップと、

前記サンプルにおける各URLについて、

前記URLのためのコンテンツを使用して前記URLのエンベッタのための第1のレンダリング結果を生成し、前記URLのための前記コンテンツを使用することなく前記エンベッタのための第2のレンダリング結果を生成し、

前記第1のレンダリング結果と前記第2のレンダリング結果とを比較することによって前記URLのための類似性スコアを計算する

ことによって、前記URLがオプションであるか否かを判定するステップであって、前記類

40

50

似性スコアが類似性しきい値よりも大きい場合、前記URLはオプションである、判定するステップと、

前記サンプルにおける前記URLの大多数がオプションである場合、オプションリソースパターンのデータストアに、前記URLパターンを記憶するステップとを備える方法。

【請求項 9】

要求された埋込リソースのコンテンツを求める要求を受信するステップと、

前記要求された埋込リソースが前記データストア内の前記URLパターンに一致するか否かを判定するステップと、

前記要求された埋込リソースが、前記URLパターンと一致する場合、前記要求された埋込リソースがオプションであるというインジケーションを返すステップとをさらに備える、請求項8に記載の方法。

10

【請求項 10】

前記サンプルにおけるURLがオプションであるか否かを判定するステップの前に、前記複数のURLによって表される数が、サイズしきい値を超えることを判定するステップをさらに備える、請求項8に記載の方法。

【請求項 11】

前記サンプルにおける前記URLのすべてがオプションである場合、オプションリソースパターンの前記データストアに前記URLパターンを記憶するステップをさらに備える、請求項8に記載の方法。

20

【請求項 12】

前記フェッチレコードにおける前記複数のURLに共通の前記URLパターンを識別するステップは、

前記フェッチレコードにおける各々のURLについて、前記URLから少なくともクエリ文字列の一部を削除することによって前記フェッチレコード内の前記URLのためのグループURLを生成するステップと、

グループURLによってクラスタ化するステップとを含む、請求項8に記載の方法。

【請求項 13】

前記フェッチレコードにおける前記複数のURLに共通の前記URLパターンを識別するステップは、

30

前記フェッチレコードにおける各々のURLについて、前記URLから少なくともクエリ文字列の一部を削除することによって前記フェッチレコードにおける前記URLのためのグループURLを生成するステップと、

グループURLによってクラスタ化するステップと、

クラスタにおけるメンバの数がしきい値を満足する場合、前記クラスタの前記グループURLを選択するステップとを含む、請求項8に記載の方法。

【請求項 14】

ウェブページをレンダリングする要求を受信するステップと、

40

少なくとも1つのプロセッサを使用して、フェッチを要求する、前記ウェブページ内の少なくとも1つの埋込リソースを識別するステップと、

前記埋込リソースのためのURLが、オプションリソースのデータストア内のパターンに一致すると判定することによって、前記埋込リソースがオプションリソースであると判定するステップと、

前記少なくとも1つのプロセッサを使用して、前記埋込リソースのためのコンテンツをフェッチすることなく、前記埋込リソースが利用不可能であるかのように前記ウェブページをレンダリングするステップとを備える方法。

【請求項 15】

50

サービスから、オプションリソースのためのパターンの前記データストアを受信するステップをさらに備える、請求項14に記載の方法。

【請求項16】

前記埋込リソースがオプションリソースであると判定するステップは、

前記URLから、少なくともクエリ文字列の一部を削除することによって、前記埋込リソースのための前記URLをリライトするステップと、

前記リライトされたURLを、オプションリソースの前記データストア内の前記パターンに一致させるステップであって、前記リライトされたURLが前記パターンに一致する場合、前記埋込リソースはオプションである、一致させるステップを含む、請求項14に記載の方法。

10

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、最適化されたブラウザレンダリング処理に関する。

【背景技術】

【0002】

多くのウェブページが動的に生成される。これは、ウェブブラウザのユーザによって見られた最後のページが、ブラウザレンダリング処理によって少なくとも部分的に生成されることを意味する。スクリプト、画像、およびスタイルシートのような埋込オブジェクトは、しばしば、ウェブページ内に埋め込まれ、レンダリングされたウェブページのコンテンツおよび外観に影響を与え得るか、または、動的なコンテンツを生成することに無関係な他の目的のために含まれ得る。たとえば、Google Analytics JavaScript（登録商標）コードは、ウェブサイトへのトラフィックの追跡を助けるがコンテンツに影響を与えない埋込オブジェクトであり得る。埋込オブジェクトはそれ自身、追加の埋込オブジェクトを含み得る。同様に、いくつかの埋込オブジェクトは、秘匿され、ユーザに見えないことがあり得る。ウェブページは何百もの埋込オブジェクトを有し得、ブラウザのレンダリングエンジンがユーザのためにウェブページをレンダリングするために費やす時間の多くは、ブラウザがこれら埋込リソースをフェッチするのを待つことに費やされる。

20

【発明の概要】

【課題を解決するための手段】

30

【0003】

実施は、オプションの埋込リソースを識別し、レンダリングエンジンが、そのようなリソースをスキップすることを可能にする。これは、レンダリング時間を改善する。オプションリソースは、レンダリングされたウェブページのコンテンツに影響を与えることなく、意味のある、または、顕著な手法でスキップされ得る埋込オブジェクトを表す。ブラウザは、レンダリング処理の間、そのようなオプションの埋込リソースをスキップし得る。その結果、より高速なロード時間となる。

【0004】

1つの態様では、システムは、少なくとも1つのプロセッサと、少なくとも1つのプロセッサによって実行された場合に、システムに対して動作を実行させる命令を記憶するメモリとを含む。これら動作は、フェッチレコードに記憶された複数のURLに共通のURLパターンを識別するステップと、複数のURLからURLのサンプルを選択するステップとを含み得る。これら動作はまた、URLのためのコンテンツを使用してURLのエンベッタのための第1のレンダリング結果をレンダリングし、URLのためのコンテンツを使用することなくエンベッタのための第2のレンダリング結果をレンダリングし、第1のレンダリング結果を第2のレンダリング結果と比較することによってURLのための類似性スコアを計算することによって、URLがオプションであるか否かを、サンプル内の各URLについて判定するステップを含む。この方法はまた、類似性スコアが類似性しきい値よりも大きい場合、URLがオプションであると判定するステップと、サンプル内のあらかじめ決定された数のURLがオプションである場合、オプションリソースパターンのデータストアに、URLパターンを記憶す

40

50

るステップとを含む。

【0005】

本明細書で説明される主題の実施の1つまたは複数は、以下の特徴のうちの1つまたは複数を含み得る。たとえば、これら動作はまた、要求された埋込リソースのコンテンツを求める要求を受信するステップと、要求された埋込リソースがデータストア内のパターンに一致するか否かを判定するステップと、要求された埋込リソースがパターンに一致する場合、要求された埋込リソースがオプションであることを示すインジケーションを返すステップとを含み得る。いくつかの実施では、インジケーションは、エラーが発見されていないURLであり得る。別の例として、フェッチレコードに記憶された複数のURLに共通のURLパターンを識別するステップは、URLからクエリ文字列の少なくとも一部を削除することによって、フェッチレコードに記憶された各々のURLのためのグループURLを生成するステップと、グループURLによってURLをクラスタ化するステップとを含み得る。別の例として、フェッチレコードに記憶された複数のURLに共通のURLパターンを識別するステップは、URLから少なくともクエリ文字列の一部を削除し、グループURLによってURLをクラスタ化し、最多数のメンバを有するクラスタのグループURLをURLパターンとして選択することによって、フェッチレコードに記憶された各々のURLのためのグループURLを生成するステップを含み得る。別の例として、類似性スコアを計算するステップは、第1のレンダリング結果のDOMツリー、および第2のレンダリング結果のDOMツリーのための最長共通シーケンスを決定するステップと、類似性スコアを決定するために最長共通シーケンスを使用するステップとを含み得る。

【0006】

別の態様では、コンピュータによって実施される方法は、フェッチレコードに記憶された複数のURLに共通のURLパターンを識別するステップと、少なくとも1つのプロセッサを使用して、複数のURLからURLのサンプルを選択するステップとを含む。この方法はまた、URLのためのコンテンツを使用して、URLのエンベッタのための第1のレンダリング結果を生成し、URLのためのコンテンツを使用することなく、エンベッタのための第2のレンダリング結果を生成し、第1のレンダリング結果と第2のレンダリング結果とを比較することによってURLのための類似性スコアを計算することによって、サンプルにおける各URLについて、URLがオプションであるか否かを判定するステップを含み得る。類似性スコアが類似性しきい値よりも大きい場合、URLはオプションである。この方法はまた、サンプルにおけるURLの大多数がオプションである場合、オプションリソースパターンのデータストアにURLパターンを記憶するステップを含み得る。

【0007】

本明細書で説明される主題の実施の1つまたは複数は、以下の特徴のうちの1つまたは複数を含み得る。たとえば、方法はまた、サンプルにおけるURLがオプションであるか否かを判定するステップの前に、複数のURLによって表される数が、サイズしきい値を超えるか否かを判定するステップをも含み得る。別の例として、方法は、サンプル内のすべてのURLがオプションである場合、オプションリソースパターンのデータストアにURLパターンを記憶するステップを含み得る。別の例として、フェッチレコードにおける複数のURLに共通のURLパターンを識別するステップは、フェッチレコードにおける各々のURLについて、URLから少なくともクエリ文字列の一部を削除し、グループURLによってクラスタ化し、クラスタにおけるメンバの数がしきい値を満足する場合、クラスタのグループURLを選択することによって、フェッチレコードにおけるURLのためのグループURLを生成するステップを含み得る。

【0008】

別の態様では、方法は、ウェブページをレンダリングする要求を受信するステップと、フェッチを要求する、ウェブページ内の少なくとも1つの埋込リソースを識別するステップと、埋込リソースがオプションリソースであると判定するステップと、埋込リソースのためのコンテンツをフェッチすることなく、埋込リソースが利用不可能であるかのようにウェブページをレンダリングするステップとを含む。

【0009】

本明細書で説明される主題の実施の1つまたは複数は、以下の特徴のうちの1つまたは複数を含み得る。たとえば、埋込リソースがオプションリソースであると判定するステップは、埋込リソースのためのURLが、オプションリソースのデータストア内のパターンに一致すると判定するステップを含み得る。別の例として、方法はまた、オプションリソースのためのパターンのデータストアを、サービスから受信するステップを含み得、埋込リソースがオプションリソースであると判定するステップは、埋込リソースのためのURLが、データストア内のパターンと一致すると判定するステップを含む。別の例として、埋込リソースがオプションリソースであると判定するステップは、URLからクエリ文字列部分を削除することによって、埋込リソースのためのURLをリライトするステップと、リライトされたURLを、オプションリソースのデータストア内のパターンに一致させるステップとを含み得、リライトされたURLがこのパターンに一致する場合、埋込リソースはオプションである。別の例として、埋込リソースがオプションリソースであると判定するステップは、少なくともクエリ文字列の一部をURLから削除することによって、埋込リソースのためのURLをリライトするステップと、リライトされたURLを、オプションリソースのデータストア内のパターンに一致させるステップとを含み得、リライトされたURLがこのパターンに一致する場合、埋込リソースはオプションである。

10

【0010】

別の態様では、非一時的なコンピュータ読取可能な媒体は、コンピュータシステムに対して、上述された方法のうちの1つまたは複数を実行させる、基板に形成された少なくとも1つのプロセッサによって実行可能な命令を含み得る。

20

【0011】

本明細書で説明される主題の実施の1つまたは複数は、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施され得る。一例として、オプションの埋込リソースをスキップするステップによって、レンダリングエンジンは、コンテンツを喪失することなく、ウェブページをより高速にレンダリングできるようになる。ブラウザは、オプションの埋込リソースをフェッチするステップをスキップし得るのみならず、ブラウザはまた、スキップされたオプションの埋込リソースに埋め込まれたオブジェクトをフェッチするステップをもスキップする。したがって、開示された実施は、ネットワーク帯域幅およびバッテリー消費量を節約する。ウェブページは、何百もの埋込オブジェクトを含み得るので、そのような最適化は本質的になり得る。

30

【図面の簡単な説明】

【0012】

【図1】開示された主題に従う例示的なシステムを例示する図である。

【図2】実施に従って、レンダリングされたウェブページのためのレンダリング結果のブロック図である。

【図3】実施に従って、オプションリソースパターンを識別するための例示的な処理を例示するフローチャートである。

【図4】実施に従って、2つのレンダリング結果が類似しているか否かを判定するための例示的な処理を例示するフローチャートである。

40

【図5】実施に従って、最適化されたレンダリングのための情報を提供するための例示的な処理を例示するフローチャートである。

【図6】説明された技術を実施するために使用され得るコンピュータデバイスの例を示す図である。

【図7】説明された技術を実施するために使用され得る分散されたコンピュータデバイスの例を示す図である。

【発明を実施するための形態】

【0013】

開示された実施は、オプションの埋込オブジェクト、または、レンダリングされたウェブページのコンテンツに対して顕著な影響を与えない埋込オブジェクトを識別し、これに

50

よって、ブラウザレンダリングエンジンは、オプションの埋込オブジェクトをフェッチすることなく、そのようなウェブページをレンダリングできるようになる。これは、フェッチシステムにおけるレンダリング時間および負荷を改善する。これは、スマートフォンおよびタブレットのような制限されたリソースしか有さないコンピュータデバイスにとって重要である。オプションの埋込オブジェクトとともにレンダリングされたエンベッダウェブページのレンダリング結果が、オプションの埋込オブジェクトなしでレイダリングされたエンベッダウェブページのレンダリング結果に類似している場合、埋込リソースは、最適化されたレンダリングのための候補となり得る。類似性は、ページにおけるトークンの類似性、アウトリンクの類似性、ページレイアウトの類似性等によって判定され得る。いくつかの実施では、システムは、レンダリング結果が類似しているか否かを判定するために、最長共通シーケンス測定を使用し得る。いくつかの実施では、スナップショットの主要な構成要素が、類似性しきい値を満足する類似性スコアを有している場合、このスナップショットは類似していると考えられ得る。埋込リソースがオプションであるとシステムが判定すると、システムは、オプションの埋込リソースのリストに埋込リソース(たとえば、そのユニフォームリソースロケータすなわちURL)を記憶し得る。いくつかの実施では、システムは、他のオプションの埋込リソースを識別するために使用され得るパターンを識別するために、オプションの埋込リソースURLをアグリゲートし得る。

10

【0014】

ブラウザがウェブページのレンダリングを開始した場合、埋込リソースのうちの何れかがオプションであれば、ブラウザは、サービスを要求し得る。いくつかの実施では、これは、フェッチ要求の一部として行われ得る。サービスは、正確な一致、または、データストアにおけるパターンの一致の何れかによって、要求された埋込リソースのURLを、オプションの埋込オブジェクトのデータストアへ一致させ得る。サービスが一致を発見すると、サービスは、ブラウザに対して、どの埋込リソースがオプションであるかを伝え、ブラウザは、これら埋込リソースをフェッチすることなく、ウェブページをレンダリングし得る。そのように最適化されたブラウザレンダリングは、ブラウザレンダリング処理を、よりさらにリソース効率的にする。いくつかの実施では、オプションの埋込リソースデータストアは、たとえばモバイルブラウザおよびデスクトップブラウザのように、ブラウザタイプを区別し得る。

20

【0015】

図1は、例示的な実施に従うシステムのブロック図である。システム100は、オプションの埋込リソースとオプションリソースパターンとを識別し、ブラウザレンダリング処理を最適化するためにこれらパターンを使用するために使用され得る。システム100は、多くの異なるデバイスの形態をとるコンピューティングデバイスを含み得る。たとえば、システム100は、図6に描写されたようなコンピュータデバイス600、および図7に描写されたようなコンピュータデバイス700の例であるコンピューティングデバイスを含み得る。

30

【0016】

システム100は、バッチレンダリングシステム110を含み得る。バッチレンダリングシステム110は、標準的なサーバ、そのようなサーバのグループ、クライアント-サーバシステム、またはラックサーバシステムであり得る。それに加えて、バッチレンダリングシステム110は、パーソナルコンピュータにおいて実施され得る。バッチレンダリングシステム110は、メモリに記憶された、1つまたは複数のマシン実行可能な命令、またはいくつかのソフトウェア、ファームウェア、またはこれらの組合せを実行するように構成された1つまたは複数のプロセッサを含み得る。たとえば、バッチレンダリングシステム110は、レンダリングエンジン120、フェッチサービス122、ウェブ-クロールエンジン124、およびオプションリソース識別エンジン126を含み得る。バッチレンダリングシステム110は、レンダリングされたウェブページを、バッチモードで、たとえば、インデクス処理の一部として生成し、この処理において、フェッチレコード132を生成し得る。これは、バッチレンダリングシステム110が、オプションリソースパターン130を生成するために使用し得る。

40

50

【0017】

ダウンストリーム処理は、バッチレンダリングシステム110がウェブページをバッチモードでレンダリングすることを要求し得る。いくつかの実施では、バッチレンダリングシステム110は、オプションリソース識別エンジン126またはインデクスエンジン(図示せず)のようなダウンストリーム処理を含み得る。いくつかの実施では、ダウンストリーム処理は、バッチレンダリングシステム110とは異なるコンピューティングデバイス上で実行され得る。たとえば、ダウンストリーム処理は、遠隔手順コールを介してバッチレンダリングシステムへ要求を送るインデクスエンジンまたは広告プラットフォームであり得る。ダウンストリーム処理は、バッチレンダリングエンジン120のうちの1つが、特定のウェブページのレンダリング結果を生成することを要求し得る。各バッチレンダリングエンジン120は、パーソナルウェブブラウザのためにレンダラをエミュレートするように構成され得るが、バッチレンダリングのために最適化されている。バッチレンダリングシステム110は、何千ものバッチレンダリングエンジン120を含み得る。そして、特定の要求へ応答するバッチレンダリングエンジン120のうちの1つを選択するために、負荷平準化を使用し得る。要求されたウェブページは、スタイルシート、JavaScript(登録商標)、画像等のような埋込オブジェクトを含み得る。バッチレンダリングエンジン120は、フェッチサービス122を使用して、埋込オブジェクトのためのコンテンツを要求し得る。

10

【0018】

フェッチサービス122は、どの埋込リソースがホストサーバ190からフェッチされる必要があるのか、どの埋込リソースがキャッシュから戻され得るのか、および、どのリソースが、戻される必要がないのか、を判定し得る。ホストサーバ190は、1つまたは複数のウェブページ、または、1つもしくは複数のウェブページに埋め込まれたリソースをホストする、インターネットを介してアクセス可能な任意のタイプのコンピューティングデバイスであり得る。埋込リソースがフェッチされる必要があるのであれば、フェッチサービス122は、従来技術を使用して、ウェブ-クロールエンジン124を介して埋込オブジェクトのためのコンテンツを要求し得る。インデクスエンジンのようなダウンストリーム処理はまた、ウェブ-クロールエンジン124を介してサーバ190からコンテンツを要求し得る。ウェブ-クロールエンジン124を介したフェッチ要求の結果として、バッチレンダリングエンジン110は、フェッチレコード132を生成し得る。フェッチレコード132は、どのウェブページおよび埋込オブジェクトが要求されホストサーバ190から検索されたのかに関する情報を含み得る。フェッチレコード132はまた、もしあれば、要求の時間、エンベッダウェブページ等のような追加の情報をも含み得る。

20

30

【0019】

ウェブ-クロールエンジン124、バッチレンダリングエンジン120、およびフェッチサービス122は、ワールドワイドウェブにおいて発見され得るウェブページのような多数のウェブページを効率的にレンダリングするために共に動作する。ウェブページのレンダリングは、レンダリング結果である。これは、ダウンストリーム要求処理に対して有用な、さもないと、利用不可能な、様々なデータ要素を含む。オプションリソース識別エンジン126は、オプションリソースパターン130を生成するために、バッチレンダリングエンジン120を使用し得る。オプションリソース識別エンジン126は、入力としてフェッチレコード132を使用して、周期的に(たとえば、毎日、週に2回等)実行し得る。オプションリソース識別エンジン126は、フェッチレコード132を分析し、前の周期にフェッチされたURLのためのパターンを生成し得る。たとえば、オプションリソース識別エンジン126は、フェッチレコードにおけるURLのクエリ文字列を取り除き、各々のURLのためのグループURLを生成する。クエリ文字列は、URLにおける疑問符(?)の後の任意のキャラクタであり得る。いくつかの実施では、クエリ文字列の一部のみが、URLのためのグループURLを生成するために取り除かれ得る。その後、オプションリソース識別エンジン126は、グループURLをソートまたはクラスタ化し得、時間周期から、どのグループURLが最多数のフェッチ要求を有していたのかを判定し得る。

40

【0020】

50

オプションリソース識別エンジン126は、最多数のフェッチ要求を有するグループURLを、潜在的なパターンとして選択し得る。そのようなパターンは、最も頻繁にフェッチされるので、ブラウジング処理を最適化するための最大の可能性を有する埋込リソースを表す。オプションリソース識別エンジンは、潜在的なパターンに一致するURLが、または、潜在的なパターンに一致するURLのサンプルがオプションであるか否かを判定し得る。たとえば、オプションリソース識別エンジン126は、フェッチレコード132からの潜在的なパターンに一致する埋込リソースを識別し得、バッチレンダリングエンジン120が、埋込リソースのためのエンベッダウェブページの第1のレンダリング結果をレンダリングすることを要求し得る。エンベッダウェブページは、たとえば、埋込リソースのためのフェッチレコードから識別され得る。オプションリソース識別エンジン126は、その後、バッチレンダリングエンジン120が、埋込リソースをフェッチすることなく、エンベッダウェブページの第2のレンダリング結果をレンダリングすることを要求し得る。

10

【0021】

特定の埋込リソースをスキップすることによって、エンベッダウェブページのレンダリングされたコンテンツが影響されているか否かを判定するために、オプションリソース識別エンジン126は、第1のレンダリング結果を、第2のレンダリング結果と比較し得る。いくつかの実施では、コンテンツが顕著に影響を受けているのであれば、特定の埋込リソースが、要求されたりソースのリストへ追加され得る。コンテンツが顕著に影響を受けているのではない(たとえば、コンテンツが類似している)のであれば、オプションリソース識別エンジン126は、埋込リソースを、オプションリソースとして識別し得る。いくつかの実施では、埋込リソース(たとえば、そのURL)は、オプションリソースパターン130のようにデータストアに記憶され得る。いくつかの実施では、埋込リソースは、オプションリソースのためのパターンを決定するために、後の時間において使用されるオプションリソースの一時的なリストに記憶され得る。オプションリソース識別エンジン126は、フェッチレコード内の潜在的なパターンに一致するすべての埋込リソースについて、または、潜在的なパターンに一致する埋込リソースのサンプルのために、このテスト(レンダリング結果の比較)を実行し得る。いくつかの実施では、潜在的なパターンに一致する埋込リソースの何れかが適切であれば、潜在的なパターンは、オプションリソースパターンではない。いくつかの実施では、潜在的なパターンに一致する僅かなパーセンテージ(たとえば、1%またはそれ未満)の埋込リソースのみが、要求されたりソースであれば、潜在的なパターンは、オプションリソースパターンであると考えられ、オプションリソース識別エンジン126は、潜在的なパターンを、オプションリソースパターン130におけるパターンとして含み得る。もちろん、潜在的なパターンに一致するテストされたすべての埋込リソースがオプションであれば、オプションリソース識別エンジン126は、潜在的なパターンを、オプションリソースパターン130へ追加し得る。

20

30

【0022】

いくつかの実施では、オプションリソース識別エンジン126はまた、オプションリソースパターン130におけるパターンが未だにオプションであることを検証し得る。たとえば、オプションリソース識別エンジン126は、パターンに一致するURLのサンプルを選択し、パターンに一致する埋込リソースを用いて、および、埋込リソースなしで、レンダリング結果をレンダリングするようにレンダリングエンジンに対して要求し、これら2つのレンダリング結果を比較し得る。オプションリソースパターン130におけるパターンと一致するURLが、もはやオプションではないのであれば、このパターンは削除され得る。

40

【0023】

いくつかの実施では、オプションリソース識別エンジン126は、パターンが、フルブラウザから分離しているモバイルブラウザのためのオプションであるか否かを判定し得る。たとえば、いくつかのウェブサイトは、ウェブページのモバイルバージョンおよびフルバージョンのための異なるコンテンツをロードし、埋込リソースは、モバイルブラウザのためのオプションであり得るが、フルブラウザのために必要とされ得る。したがって、オプションリソース識別エンジン126は、埋込リソースのために4つのレンダリング結果をレン

50

ダリングし得る。最初の2つのレンダリング結果は、フルブラウザをエミュレートするレンダリングエンジンによってレンダリングされ得る一方、最後の2つのレンダリング結果は、スマートフォンまたはタブレットのようなモバイルデバイスにおけるモバイルブラウザをエミュレートするレンダリングエンジンによってレンダリングされ得る。したがって、オプションリソースパターン130は、パターンが、たとえば、モバイルブラウザのため、または、フルブラウザのためのように、ブラウザタイプによってオプションであるか否かを示すデータを含み得る。

【0024】

簡潔のために図1に図示されていないが、いくつかの実施では、バッチレンダリングシステム110は、複数の個別のコンピューティングデバイスに分散され得る。それに加えて、バッチレンダリングエンジン120、フェッチデバイス122、ウェブ-クロールエンジン124、およびオプションリソース識別エンジン126のうちの1つまたは複数が、1つまたは複数のコンピューティングデバイスにわたって分散され得る。いくつかの実施では、バッチレンダリングエンジン120、フェッチサービス122、ウェブ-クロールエンジン124、およびオプションリソース識別エンジン126のうちの1つまたは複数は、メモリまたはハードウェアプロセッサのようなリソースを、バッチレンダリングシステム110の他の構成要素と共有し得る。同様に、フェッチレコード132およびオプションリソースパターン130もまた、多数のコンピューティングデバイスにわたって分散されたメモリに記憶され得る。いくつかの実施では、バッチレンダリングシステム110の様々な構成要素は、コンピューティングデバイスのハードウェア構成要素を共有し得るか、または、同じコンピューティングデバイスの論理区分であり得る。

【0025】

バッチレンダリングシステム110は、クライアント180およびサーバ190とネットワーク160を介して通信し得る。ネットワーク160は、たとえばインターネットであり得るか、または、ネットワーク160は、たとえばゲートウェイデバイス、ブリッジ、スイッチ等を使用して実施される、ワイヤまたはワイヤレスローカルエリアネットワーク(LAN)、広域ネットワーク(WAN)、これらの組合せ等であり得る。ネットワーク160を介して、バッチレンダリングシステム110は、クライアント180および/またはホストサーバ190と通信し、クライアント180および/またはホストサーバ190へ/からデータを送信し得る。たとえば、バッチレンダリングシステム110は、オプションリソースパターン130をクライアント180へ提供し得るか、または、特定の埋込リソースがオプションであるか否かを尋ねる要求をクライアント180から受信し得、オプションリソースパターン130に基づいて応答を提供し得る。

【0026】

クライアント180は、図6に描写されたコンピューティングデバイス600のように、インストールされたパーソナルウェブブラウザ140を備えるパーソナルコンピューティングデバイスであり得る。パーソナルウェブブラウザの例は、スマートフォンまたはタブレットのようなモバイルデバイス用、または、ラップトップまたはデスクトップのようなパーソナルコンピュータ用であるかに関わらず、CHROME、SAFARI、INTERNET EXPLORER、FIREFOX等を含む。ブラウザ140は、ウェブページをレンダリングした場合に、特定の埋込リソースをフェッチするか否かを判定するためにサービスを使用するように構成され得る。いくつかの実施では、システムは、ブラウザ140による使用のために、オプションリソースパターン130のコピーを、クライアント180へプッシュし得る。この観点において、サービスは、ローカルサービスであり得る。いくつかの実施では、ブラウザ140は、特定の埋込リソースが、オプションリソースパターン130におけるパターンのうちの1つと一致するか否かを判定するために、サーバ-ベースのサービスを求めるように構成され得る。ブラウザ140は、ウェブページをレンダリングする場合に、オプション埋込リソースをスキップするために、このサービスを使用することによって、最適化されたレンダリングを実行する。したがって、ブラウザ140は、コンテンツが影響されない場合、レンダリング時間を改善し、リソースを節約するために、埋込リソースを選択的にスキップするように構成され得

る。

【0027】

図2は、実施に従って、バッチレンダリングエンジン120によってレンダリングされたレンダリング結果200のブロック図である。レンダリング結果200は、そのすべてが図2に例示されている訳ではないが、様々な構成要素を含み得る。たとえば、レンダリング結果200は、レンダリングされたページの画像205を含み得る。画像205は、ディスプレイデバイスを介して、ウェブブラウザを介してユーザへ表示されるであろう画像であり得る。画像205は、たとえば、レンダリングされたページのサムネイルをユーザへ表示するために、および、ディスプレイ上のどこにウェブページの要素が現れるのかを(たとえば、そのx座標およびy座標に基づいて)判定するために、使用され得る。レンダリング結果200はまた、ドキュメントオブジェクトモデル(DOM)ツリー210を含み得る。DOMツリー210は、ウェブページのHTML構造を表す。レンダリング結果200はまた、レイアウト215をも含み得る。レイアウト215は、一般に、ウェブページの要素のためのボックスを含み、このボックスは、画像205における要素のx座標およびy座標を規定する。したがって、レイアウト215は、ウェブページ上のどこに要素が現れるのか、それがウェブページ上でどれだけ多くのスペースを使うのか、等を示すインジケーションを提供する。したがって、レイアウト215は、ウェブページのうちのどれくらいの量が広告なのか、パラグラフがどのように際立っているのか(たとえば、折り目の上、または、折り目の下)、要素が目に見えるか否か等に関する情報を提供する。言い換えると、レイアウト215は、レンダリングされたウェブページの要素に関する幾何学情報を提供する。このレンダリング結果200はまた、エラー220を含み得る。エラー220は、たとえばJavaScript(登録商標)のようなスクリプトオブジェクトの実行の結果として遭遇したエラーを含む。レンダリング結果200はまた、レンダリング中にフェッチされた埋込リソース225のリストを含み得る。そして、レンダリング処理の一部として生成された他の要素(図示せず)を含み得る。したがって、レンダリング結果200は、ホストサーバからのコンテンツのフェッチによって、単独では利用可能ではない情報を提供する。オプションリソース識別エンジンのようなダウンストリーム要求処理は、実行中のスクリプトオブジェクトが、レンダリングされているウェブページのコンテンツを顕著に変更するか否かを判定するために、様々な目的のためにレンダリング結果情報を使用し得る。たとえば、ウェブページの広告部分におけるコンテンツは、重要であると考えられないことがあり得るので、URLは、広告部分に相違がある場合であっても未だにオプションであり得る。いくつかの実施では、レンダリング結果の主要な構成要素における相違のみが、重要であると考えられ得る。主要な構成要素は、レイアウト215における最大のボックス(たとえば、最大の高さおよび幅を有するボックス)であり得る。

【0028】

図3は、実施に従って、オプションリソースパターンを識別するための例示的な処理300を例示するフローチャートである。処理300は、図1のシステム110のようなシステムによって実行され得る。システムは、フェッチレコードからオプションリソースのためのパターンを識別するために処理300を使用し得る。システムは、リソースありでレンダリングされたエンベッダウェブページと、リソースなしでレンダリングされたエンベッダウェブページとのレンダリング結果比較に基づいて、オプションリソースのためのパターンを決定し得る。これらパターンは、データストアに記憶され、フルブラウザ、モバイルブラウザ、またはバッチレンダリングエンジンであるかに関わらず、ブラウザが、オプションリソースをスキップする(たとえば、フェッチしない)ことによってレンダリングを最適化できるようにするサービスの一部として使用され得る。

【0029】

処理300は、システムが、フェッチされた埋込リソースを潜在的なパターンによってクラスタ化することで始まり得る(305)。システムは、たとえば、一日、二日、一週間、最後の時間処理300が実行されてから等のような、ある前の時間周期にフェッチされたすべての埋込リソースを選択し得る。潜在的なパターンは、フェッチレコードにおいて発見されたURLからのクエリ文字列のすべてまたは一部を取り除くことによって生成され得る。

フェッチレコードは、ウェブ-クロールエンジンまたはフェッチサービスによって生成され得る。いくつかの実施では、フェッチレコードは、たとえば、インターネットを介して利用可能なドキュメントのためのインデクス処理のようなインデクス処理の一部として生成され得る。埋込リソースは、URLのような識別子によって、フェッチレコードにおいて識別され得る。したがって、ウェブページまたは埋込リソースはまた、URLとして称され得、埋込リソース(またはウェブページ)およびURLへの参照は、本明細書で使用されるように、一般に同意語であると理解される。潜在的なパターンは、クエリ文字列のすべてまたは一部を取り除くことによってURLのために生成され得、クエリ文字列は、埋込リソースのためのURLにおける疑問符(?)に続く任意のキャラクタである。潜在的なパターンはまた、サブドメイン名を取り除くまたは無視する、パスの構成要素(たとえば、最初の
10 フォワードスラッシュ(「/」)およびクエリ文字列(「?」))を無視する、または、クエリ文字列内のあるパラメータを無視することによって、埋込リソースのために生成され得る。いくつかの実施では、フェッチレコードからの埋込リソースが、そのURLから生成された1つよりも多くの潜在的なパターンに関連付けられ得る。潜在的なURLは、ソートすることによって、または、他の既知のクラスタ化技術によって、クラスタ化され得る。

【0030】

システムは、潜在的なパターンのうちの1つによって表されるクラスタを選択し(310)、クラスタメンバの数が、しきい値以上であるか否かを判定し得る(315)。クラスタメンバの数は、潜在的なパターンに一致する埋込リソースが、前の時間周期中にフェッチされた
20 回数数を表す。したがって、1つの特定の埋込リソースは、一度よりも多くフェッチされているのであれば、その数における複数回数表され得る。クラスタの数がしきい値未満(315、No)であれば、システムは、次のクラスタへ進み得る(355)。潜在的なパターンが、十分なメンバの数を有しているのであれば(315、Yes)、システムは、潜在的なパターンに一致する埋込リソースのサンプルを選択し得る(320)。言い換えれば、システムは、クラスタのユニークなメンバをサンプルし得る。もちろん、いくつかの実施では、システムは、クラスタ内のすべてのユニークな埋込リソースをサンプルとして選択し得る(たとえば、サンプルサイズ100%)。システムは、その後、サンプルにおける埋込リソースのうちの1つを選択し得(325)、選択された埋込リソースを用いてリソースを埋め込むウェブページ(たとえば、エンベッダウェブページ)の第1のレンダリング結果を生成し得る(330)。エンベッダウェブページは、フェッチレコードから決定され得る。システムはまた、選択され
30 た埋込リソースをフェッチすることなく、エンベッダウェブページの第2のレンダリング結果を生成し得る(330)。システムは、たとえばバッチレンダリングエンジンを使用して、2つのレンダリング結果を生成し得る。システムは、その後、第1のレンダリング結果と第2のレンダリング結果とを比較し、これらレンダリング結果が類似しているか否かを判定し得る(335)。

【0031】

システムは、図4に関してより詳細に説明されるように、これらレンダリング結果が類似しているか否かを判定するために、様々なテストを使用し得る。これらレンダリング結果が類似していない(335、No)のであれば、システムは、次のクラスタへ移動し得る(355)。これらレンダリング結果が類似している(335、Yes)のであれば、システムは、クラスタ
40 サンプル内に、検査すべき他の埋込リソースがあるか否かを判定し得る(340)。他の埋込リソースが存在する(340、No)のであれば、システムは、サンプルのための次の埋込リソースを選択し(345)、ステップ330および335を繰り返し得、2つのレンダリング結果を生成し、これら結果を比較する。サンプル内のすべての埋込リソースが検査されているのであれば(340、Yes)、システムは、クラスタのための潜在的なパターンを、オプションリソースパターンデータストアへ追加し得る(350)。このデータストアは、図5に関してより詳細に説明されるように、フェッチされる必要のない埋込リソースを識別するためにサービスによって使用され得る。システムは、その後、検査すべき他のクラスタがあるか否かを判定し得る(355)。分析すべき残っている他のクラスタが存在するのであれば(355、No)、システムは、次のクラスタを選択し得、次のクラスタの潜在的なパターンに一致する埋込リ
50

ソースのためにステップ315～355を繰り返す。分析すべきさらなるクラスタが存在しないのであれば(355、Yes)、処理300は完了する。

【0032】

例示的な処理300は、オプションではない1つの埋込リソースが、オプションリソースパターンデータストアから省かれている潜在的なパターンになる(たとえば、335、No)であろう実施を例示する。しかしながら、いくつかの実施では、システムは、オプションである潜在的なパターンのための埋込リソースのリストと、要求されている潜在的なパターンのための埋込リリースのリスト(たとえば、これら2つのレンダリング結果は、類似しているとは考えられていない)とを維持し得る。そのような実施では、システムは、要求されている潜在的なパターンに一致する埋込リソースのパーセンテージを計算し得る。このパーセンテージが非常に低い(たとえば、1%以下)場合、システムは未だに、潜在的なパターンを、オプションリソースパターンデータストアへ追加し得る(350)。したがって、実施は、図3に例示された正確な処理300に限定されない。

10

【0033】

いくつかの実施では、システムは、異なるブラウザタイプによってレンダリングされたレンダリング結果を用いて、処理300のいくつかまたはすべてを繰り返し得る。たとえば、ブラウザタイプは、フルブラウザまたはモバイルブラウザであり得る。第1および第2のレンダリング結果が、フルブラウザによってレンダリングされているのであれば、システムは、モバイルブラウザを使用してステップ330～350を繰り返し得る。したがって、オプションリソースパターンのリストは、ブラウザタイプのインジケーションを含み得、データストアにおけるパターンもまた、ブラウザタイプのインジケーションを含み得る。これによって、システムは、埋込リソースが、1つのブラウザタイプ(たとえば、モバイルブラウザ)についてオプションであり、別のブラウザタイプ(たとえば、フルブラウザ)についてオプションではないか否かを示すことが可能となる。

20

【0034】

図4は、実施に従って、2つのレンダリング結果が類似しているか否かを判定するための例示的な処理400を例示するフローチャートである。処理400は、たとえば、図3のステップ335の一部として実行され得る。処理400は、レンダリング結果類似性を判定するための3つのテストを例示しているが、実施は、例示されたテストのうちの1つ、2つ、またはすべてを含み得、例示されていない追加の類似性テストを含み得ることが理解される。

30

【0035】

処理400は、2つのレンダリング結果におけるトークンを比較するステップを含み得る(405)。これらトークンは、たとえば、ユーザに見えるワードのような、ドキュメントのテキストを含む。トークンは、図2のDOMツリー210のようなDOMツリーを処理することにより生成され得る。いくつかの実施では、ストップワードおよび数は、比較を実行する前に削除され得る。第1のレンダリング結果のトークンが、第2のレンダリング結果のトークンとは異なる(410、Yes)のであれば、システムは、オプションとして、作動されたスクリプト記述を伴うレンダリングによって追加されたユニークなトークンの数が、トークンしきい値未満であるか否かを判定し得る(415)。言い換えれば、作動されたスクリプト記述を伴うレンダリングが、小数のユニークなトークンにしかならないのであれば、システムは、相違を些細なものとして考慮し得る。いくつかの実施では、トークンしきい値は、5であり得る。追加されたユニークなトークンの数が、トークンしきい値を満足する(415、No)のであれば、システムは、レンダリング結果を、類似していないと考慮し得る(430)。レンダリング結果が類似していない場合、レンダリング結果を生成するために使用される埋込リソースはオプションではない。いくつかの実施では、システムは、テスト415をスキップし得、トークンにおけるあらゆる相違は顕著であり、レンダリング結果は類似していない、と考慮し得る。その数が、トークンしきい値を満足しない(415、Yes)のであれば、システムは、トークンにおける相違を些細なものとして考慮し得る。したがって、第1のレンダリング結果のためのユニークなトークンが、第2のレンダリング結果におけるトークンと同じである(410、No)、または、ユニークなトークンの数が、トークンしきい値を満足

40

50

していない(415、Yes)のであれば、レンダリング結果は、類似であると考慮され、システムは、他のテストの実行を継続し得る。トークンテストのみを含む実施では、システムは、ステップ445へ直接進み得、ここでは、レンダリング結果が類似しているとの判定がなされる。

【0036】

いくつかの実施では、トークンが同じ(410、No)または(415、Yes)であれば、システムはアウトリンクを比較し得る(420)。アウトリンクは、レンダリングされたページから他のウェブページまたは他のドキュメントへのリンクを表す。アウトリンクはまた、レンダリング結果のDOMツリーのアンカタグノード(<a>タグ)から抽出され得る。システムがアウトリンクにおける相違を発見する(425、Yes)と、システムは、相違が顕著であり、レンダリング結果は類似していないと考慮し得る(430)。したがって、埋込リソースは、オプションとは考慮されない。2つのレンダリング結果におけるアウトリンクが同じ(425、No)であれば、システムは、(たとえば、ステップ450へ進むことによって)レンダリング結果を類似のものと考慮し得るか、または、他の類似性テストを実行することに進む。いくつかの実施では、システムは、ステップ405および415の前に、ステップ405および415と独立してステップ420および425を実行し得る。

【0037】

他のテストに加えて、または、他のテストの代わりに実行され得る別の類似性テストでは、システムは、画像(たとえば、スナップショット)、DOMツリー、またはレイアウト間の類似性を判定し得る。たとえば、システムは、レンダリング結果のDOMツリー、レイアウト、画像、または別の構成要素のための最長共通シーケンス(LCS)を計算し得る(435)。システムは、同じである構成要素のパーセンテージを表す類似性スコアを計算するためにLCSを使用し得る(440)。パーセンテージがしきい値を満足する場合(445、Yes)、システムは、レンダリング結果が類似していると判定し得る(450)。パーセンテージがしきい値を満足しない場合(445、No)、システムは、レンダリング結果が類似していないと判定し得る(430)。レンダリング結果が類似している(450)か、または類似していない(430)かをシステムが判定すると、処理400は終了する。

【0038】

もちろん、システムは、処理400の一部として他のテストを実行し得る。たとえば、別の類似性テスト(図4に図示せず)では、システムは、レイアウトの主要な構成要素を判定し得る。レンダリング結果のレイアウトは、各々がスクリーン座標によって定義されるボックスからなる。一般に、各ボックスは、ウェブページの各々の要素に対応する。たとえば、レイアウトは、(すべてのDOM要素に対応するレンダラボックスを有し得る訳ではないが、)DOMツリーにおけるDOMノードのボックス表現を含み得る。ボックスは、レンダラツリーとしても知られているツリー構造で体系化され得る。したがって、たとえば、テーブルは、レイアウト内のボックスによって表現され得、パラグラフは、レイアウト内の別のボックスによって表現され得る。ウェブページの主要な構成要素は、スクリーン座標によって定義されるように、最大ボックスを持つレイアウト内のこれら要素である。いくつかの実施では、システムは、たとえば、セットが、主要な構成要素のうちの最大を含むように、あらかじめ決定された数の、セット内の主要な構成要素を含み得る。いくつかの実施では、システムは、セット内のスクリーンのパーセンテージを占める主要な構成要素を含み得る。そのような実施では、最大の主要な構成要素は、そのパーセンテージ超を占め、最大の主要な構成要素は、セットのメンバのみであり得る。いくつかの実施では、しきいサイズを超えるボックスサイズを有する何れの構成要素も、主要な構成要素のセットに含まれ得る。主要な構成要素のセットにない構成要素は、重要ではない構成要素であると考えられ得る。いくつかの実施では、システムは、主要な構成要素を発見するために、オニオンピーリング技術を使用し得る。たとえば、システムは、レンダラツリーのルートボックスで始まる横型探索を実行し、ルートボックスの最大の子ボックスを識別し得る。システムはその後、最大の子を選択し、より深く進み、現在のボックスの最大の子ボックス(たとえば、ルートボックスの最大の子ボックス)を発見し得る。システムは、子ボックス

10

20

30

40

50

の何れもが支配的、たとえば、親ボックスのエリアの半分超を占有すること、ではない場合、より深く進むことを停止し得る。システムが、より深く進むことを停止する場合、主要な構成要素は、支配している子を有しないボックスである。

【0039】

システムは、セット内の主要な構成要素間の類似性スコアを計算し得る。たとえば、システムは、主要な構成要素のボックスのオーバーラップスコアを使用し得る。オーバーラップスコアでは、システムは、主要な構成要素のオーバーラップエリアを計算し得る。これは、第2のレンダリング結果における対応する主要な構成要素のエリアとオーバーラップする第1のレンダリング結果における主要な構成要素のエリアを表す。システムは、その後、類似性スコアを計算し得る。これは、オーバーラップの調和平均が、各主要な構成要素の総エリアに関連していることを表す。たとえば、システムは、以下の式を使用し得る。

【0040】

【数1】

$$\frac{2}{((a1/oa) + (a2/oa))}$$

【0041】

ここでoaは、オーバーラップエリアであり、a1は、第1のレンダリング結果における主要な構成要素ボックスの総エリアであり、a2は、第2のレンダリング結果における主要な構成要素ボックスの総エリアである。もちろん、システムは、このスコアを計算するために他の類似性メトリクスを使用し得る。そのような類似性メトリクスの例は、限定されないが、Katz類似性を含み得る。類似性スコアが、類似性しきい値を満足するのであれば、レンダリング結果は、類似していると考えられ、したがって、埋込リソースはオプションである。いくつかの実施では、類似性しきい値は、たとえば80%以上のように、高いことがあり得る。類似性スコアがそのしきい値を満足しないのであれば、システムは、レンダリング結果が類似していないと考えし得る。いくつかの実施では、類似性スコアが、類似性しきい値を満足しているのであれば、システムは、図4に例示されていないレンダリング結果に基づいて、追加の類似性テストを実行し得る。いくつかの実施では、システムはまた、重要ではない構成要素における相違のための類似性スコアを計算し得るが、これら相違をさほど重要視せず、たとえば、このレンダリング結果については、最終類似性スコアのうちの、ごく一部とする。

【0042】

いくつかの実施では、システムは、処理400に対する不確定の効果を最小化または削除することを試み得る。不確定は、埋込リソースが同一である場合でさえ、レンダリング結果が異なる場合に生じる。不確定は、オプションである埋込リソースが、あたかも必要とされているかのように見えるようにし得る。したがって、不確定のための考慮は、オプションリソースのカバレッジを顕著に増加させ得る。不確定を考慮するために、システムは、第3のレンダリング結果を生成し得る。第3のレンダリング結果は、(たとえば、サンプルURLを含む)第1のレンダリング結果の生成から返されたリソースを使用したレンダリングであり得る。第3のレンダリング結果と第1のレンダリング結果との間で生じる何れの相違も、不確定による。システムは、不確定によるあらゆる相違(たとえば、第1および第3のレンダリング結果における相違である、アウトリンク、トークン、画像、レンダラツリー等における相違)を削除することによって、類似性を除去し得る。たとえば、システムは、第1のレンダリング結果の画像と、第3のレンダリング結果の画像との間の相違であるピクセルの数を表す第1の数と、第1のレンダリング結果の画像と、第2のレンダリング結果の画像との間の相違であるピクセルの数を表す第2の数とを計算し得る。システムは、相違を生成するために、第2の数から第1の数を減じ得る。この相違が、第2の数に近づくほど、第1のレンダリング結果と第2のレンダリング結果との間の何れの相違も、不確定に

10

20

30

40

50

よらない可能性がより高くなる。したがって、システムは、その相違に基づいて、LCSの類似性スコアを調節し得る(たとえば、相違が第2の数に等しいのであれば調節をせず、相違が第2の数の半分であれば、半分まで調節する等である)。別の例として、システムは、第1のレンダリング結果と第3のレンダリング結果との間の相違であるDOMノードを判定し得、第1のレンダリング結果を第2のレンダリング結果と比較する(たとえば、LCSを計算する)場合、これらノードを無視し得る。

【0043】

図5は、実施に従って、最適化されたレンダリングのための情報を提供するための例示的な処理500を例示するフローチャートである。処理500は、ブラウザのためのサービスとして実行され得る。ブラウザは、図1のブラウザ140のような、モバイルまたはフルであるクライアントブラウザであり得るか、または、図1のバッチレンダリングエンジン120のようなバッチレンダリングエンジンであり得る。いくつかの実施では、サービスは、クラウド-ベースのサービスであり得る。言い換えれば、ブラウザは、ネットワークを介してクラウド-ベースのサービスへ要求を送信し得る。クラウド-ベースのサービスは、ブラウザへ応答を提供し得る。いくつかの実施では、サービスは、図1のバッチレンダリングシステム110のようなサーバ上で実行され得る。他の実施では、サービスは、ブラウザが実行するコンピューティングデバイスに局所的であり得る。たとえば、オプションリソースパターンのデータストアが、ブラウザを実行するコンピューティングデバイスへプッシュされ得、このサービスが、コンピューティングデバイスにおいて実行され得る。いくつかの実施では、サービスは、図1のクライアント180のようなクライアント上で実行され得る。

【0044】

処理500は、サービスがブラウザからURLを受信することで始まる(505)。ブラウザは、モバイルブラウザ、フルブラウザであり得、クライアント上で、または、バッチレンダリングエンジンとして、実行され得る。サービスは、この要求からのURLが、オプションリソースパターンに一致するか否かを判定し得る(510)。いくつかの実施では、パターンは、フルURLを表し得る。これによって、要求されたURL全体に対する一致がなされる。いくつかの実施では、パターンは、部分的なURLを、たとえば、除去されたクエリ文字列と、または、ワイルドカードによって置き換えられた様々な部分とともに表し得る。要求されたURLが、オプションリソースデータストア内の少なくとも1つのパターンと一致しない(510、No)のであれば、サービスは、URLがオプションであり、コンテンツをフェッチすることなくエンベッダウェブページがレンダリングされ得ることを示す応答を提供する(515)。いくつかの実施では、これは、リソースが位置決定され得ない場合に、ブラウザが受信し得る「URLが発見されません」という応答に類似した応答であり得る。このインジェクションを受信することに応じて、ブラウザは、埋込リソースをスキップするエンベッダウェブページをレンダリングし得、処理リソースおよび帯域幅リソースを節約する。要求されたURLが、オプションリソースデータストアにおけるパターンと一致しない(510、No)のであれば、サービスは、埋込リソースが要求されていることを示す応答を提供する(520)。いくつかの実施では、サービスは、実際にフェッチを実行し、埋込リソースのためのコンテンツを提供し得る。いくつかの実施では、サービスは、ブラウザに対して、埋込リソースを求めるフェッチ要求を進めるように伝える応答を提供し得る。したがって、ブラウザは、埋込リソースを求めるコンテンツを用いてエンベッダウェブページをレンダリングし得る。その後、処理500は、レンダリング処理を最適化するためにブラウザが使用し得る情報を提供して、終了する。いくつかの実施では、ブラウザは、ブラウザタイプをサービスへ提供し得、サービスは、オプションリソースデータストアにおけるパターンが一致した場合に、このブラウザタイプを使用し得る。

【0045】

図6は、本明細書で説明された技術とともに使用され得る、図1のバッチレンダリングシステム110および/またはクライアント180として操作され得る一般的なコンピュータデバイス600の例を図示する。コンピューティングデバイス600は、ラップトップ、デスクトップ、ワークステーション、携帯情報端末、セルラ電話、スマートフォン、タブレット、サ

ーバ、および、ウェアラブルデバイスを含む他のコンピューティングデバイスのようなコンピューティングデバイスの様々な例示的な形態を表すことが意図される。本明細書に図示された構成要素、それらの接続と関係、およびそれらの機能は、例のみであることが意図され、本書において説明および/または特許請求された発明の実施を限定することは意図されていない。

【0046】

コンピューティングデバイス600は、インターフェース608を介して接続された、たとえばシリコンベースのハードウェアプロセッサのようなプロセッサ602、メモリ604、記憶デバイス606、および拡張ポート610を含む。いくつかの実施では、コンピューティングデバイス600は、他の構成要素の中でも、インターフェース608を介して接続された、トランシーバ646、通信インターフェース644、およびGPS(全地球測位システム)受信機モジュール648を含み得る。デバイス600は、通信インターフェース644を介してワイヤレスに通信し得る。これは、必要な場合、デジタル信号処理回路を含み得る。構成要素602、604、606、608、610、640、644、646、および648の各々は、共通のマザーボードに、または他の方式で適切に搭載され得る。

10

【0047】

プロセッサ602は、コンピューティングデバイス600内の実行のための命令を処理し得る。この命令は、ディスプレイ616のような外部入力/出力デバイスにおけるGUIのためのグラフィック情報を表示するために、メモリ604に、または、記憶デバイス606に記憶された命令を含む。ディスプレイ616は、モニタ、または、フラットタッチスクリーンディスプレイであり得る。いくつかの実施では、多数のプロセッサおよび/または多数のバスが、多数のメモリおよび多数のタイプのメモリとともに適切に使用され得る。また、(たとえば、サーババンク、ブレードサーバのグループ、または、マルチ-プロセッサシステムとして)各デバイスが必要な動作の一部を提供する多数のコンピューティングデバイス600が接続され得る。

20

【0048】

メモリ604は、コンピューティングデバイス600内に情報を記憶する。1つの実施では、メモリ604は、揮発性メモリユニットである。別の実施では、メモリ604は、不揮発性メモリユニットである。メモリ604はまた、磁気または光ディスクのような別の形態のコンピュータ読取可能な媒体であり得る。いくつかの実施では、メモリ604は、拡張インターフェースを介して提供される拡張メモリを含み得る。

30

【0049】

記憶デバイス606は、コンピューティングデバイス600のための大容量ストレージを提供することができる。1つの実施では、記憶デバイス606は、フロッピーディスクデバイス、ハードディスクデバイス、光ディスクデバイス、またはテープデバイス、フラッシュメモリまたは他の類似のソリッドステートメモリデバイス、または、記憶エリアネットワークまたは他の構成におけるデバイスを含むデバイスのアレイのようなコンピュータ読取可能な媒体であり得るか、または、これらを含み得る。コンピュータプログラム製品は、そのようなコンピュータ読取可能な媒体内に、明確に具体化され得る。コンピュータプログラム製品はまた、上述したように、実行された場合に、1つまたは複数の方法を実行する命令をも含み得る。コンピュータ-またはマシン-読取可能な媒体は、メモリ604、記憶デバイス606、または、プロセッサ602におけるメモリのような、記憶デバイスである。

40

【0050】

インターフェース608は、コンピューティングデバイス600のための帯域幅消費型動作を管理する高速コントローラ、または、より低い帯域幅消費型動作を管理する低速コントローラ、または、そのようなコントローラの組合せ、であり得る。外部インターフェース640は、デバイス600の他のデバイスとの近傍エリア通信を可能にするように提供され得る。いくつかの実施では、コントローラ612は、記憶デバイス606および拡張ポート614へ結合され得る。様々な通信ポート(たとえば、USB、Bluetooth(登録商標)、イーサネット(登録商標)、ワイヤレスイーサネット(登録商標))を含み得る拡張ポートは、キーボー

50

ド、ポインティングデバイス、スキャナ、または、スイッチまたはルータのようなネットワークデバイスのような1つまたは複数の入力/出力デバイスへ、たとえばネットワークアダプタを介して結合され得る。

【0051】

図面に図示されるように、コンピューティングデバイス600は、多くの異なる形態で実施され得る。たとえば、それは、標準的なサーバ630として、または、そのようなサーバのグループ内で何度も実施され得る。また、それは、ラックサーバシステムの一部として実施され得る。それに加えて、それは、ラップトップコンピュータ632、デスクトップコンピュータ634、または、タブレットまたはスマートフォン636のようなペルソナコンピューティングデバイス内で実施され得る。システム全体は、互いに通信する多数のコンピューティングデバイス600から構成され得る。他の構成も可能である。

10

【0052】

図7は、本明細書で説明された技術とともに使用され得る、図1のシステム110であり得る、一般的なコンピュータデバイス700の例を図示する。コンピューティングデバイス700は、サーバ、ブレードサーバ、データセンター、メインフレーム、および他の大規模なコンピューティングデバイスのような大規模なデータ処理デバイスの様々な例示的な形態を表すことが意図される。コンピューティングデバイス700は、1つまたは複数の通信ネットワークによって相互接続された、恐らくはネットワークに結合された記憶ノードを含む、多数のプロセッサを有する分散システムであり得る。本明細書に図示された構成要素、それらの接続および関係、およびそれらの機能は、単なる例であることが意図され、本書において説明および/または特許請求された発明の実施を限定することは意図されていない。

20

【0053】

分散型コンピューティングデバイス700は、任意の数のコンピューティングデバイス780を含み得る。コンピューティングデバイス780は、ローカルまたは広域ネットワーク、専用光リンク、モデム、ブリッジ、ルータ、スイッチ、ワイヤまたはワイヤレスネットワーク等を介して通信する、サーバまたはラックサーバ、メインフレーム等を含み得る。

【0054】

いくつかの実施では、各コンピューティングデバイスは、多数のラックを含み得る。たとえば、コンピューティングデバイス780aは、多数のラック758a~758nを含む。各ラックは、プロセッサ752a~752nおよび762a~762nのような1つまたは複数のプロセッサを含み得る。プロセッサは、データプロセッサ、ネットワーク結合された記憶デバイス、または、他のコンピュータ制御されたデバイスを含み得る。いくつかの実施では、1つのプロセッサは、マスタプロセッサとして動作し得、スケジューリングおよびデータ配信タスクを制御し得る。プロセッサは、1つまたは複数のラックスイッチ758を介して相互接続され得、1つまたは複数のラックは、スイッチ778を介して接続され得る。スイッチ778は、接続された多数のコンピューティングデバイス700間の通信を取り扱い得る。

30

【0055】

各ラックは、メモリ754およびメモリ764のようなメモリと、756および766のようなストレージとを含み得る。ストレージ756および766は、大規模ストレージを提供し得、ネットワークに結合されたディスク、フロッピーディスク、ハードディスク、光ディスク、テープ、フラッシュメモリ、または、他の類似のソリッドステートメモリデバイス、または、記憶エリアネットワークまたは他の構成におけるデバイスを含むデバイスのアレイ、のような揮発性または不揮発性のストレージを含み得る。ストレージ756または766は、多数のプロセッサ、多数のラック、または多数のコンピューティングデバイス間で共有され得、プロセッサのうちの1つまたは複数によって実行可能な命令を記憶するコンピュータ読取可能な媒体を含み得る。メモリ754および764は、たとえば、揮発性メモリユニット、不揮発性メモリユニット、および/または、磁気ディスクまたは光ディスク、フラッシュメモリ、キャッシュ、ランダムアクセスメモリ(RAM)、読取専用メモリ(ROM)、およびその組合せのような他の形態のコンピュータ-読取可能な媒体を含み得る。メモリ754のようなメモ

40

50

りもまた、プロセッサ752a~752n間で共有され得る。インデックスのようなデータ構造は、たとえば、ストレージ756およびメモリ754にわたって記憶され得る。コンピューティングデバイス700は、コントローラ、バス、入力/出力デバイス、通信モジュール等のような図示しない他の構成要素を含み得る。

【0056】

システム110のようなシステム全体は、互いに通信する多数のコンピューティングデバイス700から構成され得る。たとえば、デバイス780aは、デバイス780b、780c、および780dと通信し得、これらはシステム100として集合的に知られ得る。別の例として、図1におけるシステム100は、1つまたは複数のコンピューティングデバイス700を含み得る。コンピューティングデバイスのうちのいくつかは、地理的に互いに接近して位置され得、他は、地理的に離れて位置され得る。コンピューティングデバイス700のレイアウトは、単なる例であって、このシステムは、他のレイアウトまたは構成を呈し得る。

【0057】

様々な実施は、基板に形成され、特別目的または一般目的であり得、記憶システム、少なくとも1つの入力デバイス、および少なくとも1つの出力デバイスからデータおよび命令を受信し、記憶システム、少なくとも1つの入力デバイス、および少なくとも1つの出力デバイスへデータおよび命令を送信するために結合された、少なくとも1つのプログラマブルプロセッサを含むプログラマブルシステムにおいて実行可能および/または解釈可能である1つまたは複数のコンピュータプログラムにおける実施を含み得る。

【0058】

(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、またはコードとしても知られる)これらコンピュータプログラムは、プログラマブルプロセッサのためのマシン命令を含み、高レベルの手続型および/またはオブジェクト指向のプログラム言語で、および/または、アセンブリ/マシン言語で、実施され得る。本明細書で使用されるように、「マシン-読取可能な媒体」、「コンピュータ-読取可能な媒体」という用語は、マシン命令および/またはデータをプログラマブルプロセッサへ提供するために使用される、任意の非一時的なコンピュータプログラム製品、装置、および/または、デバイス(たとえば、磁気ディスク、光ディスク、(リードアクセスメモリを含む)メモリ、プログラマブル論理デバイス(PLD))を称する。

【0059】

本明細書で説明されるシステムおよび技術は、バックエンド構成要素(たとえば、データサーバ)を含む、または、ミドルウェア構成要素(たとえば、アプリケーションサーバ)を含む、または、フロントエンド構成要素(たとえば、ユーザが、本明細書で説明されたシステムおよび技術の実施とインタラクトでき得るグラフィックユーザインターフェースまたはウェブブラウザを有するクライアントコンピュータ)を含む、コンピューティングシステムにおいて、または、そのようなバックエンド構成要素、ミドルエンド構成要素、またはフロントエンド構成要素の任意の組合せにおいて、実施され得る。システムの構成要素は、(たとえば通信ネットワークのような)デジタルデータ通信の任意の形態または媒体によって相互接続され得る。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」)、広域ネットワーク(「WAN」)、およびインターネットを含む。

【0060】

コンピューティングシステムは、クライアントおよびサーバを含み得る。クライアントおよびサーバは、一般に、互いに離れており、一般に、通信ネットワークを介してインタラクトする。クライアントとサーバとの関係は、各々のコンピュータにおいて実行し、互いにクライアント-サーバ関係を有している、コンピュータプログラムによって生じる。

【0061】

多くの実施が説明された。しかしながら、様々な修正が、本発明の精神およびスコープから逸脱することなくなされ得る。それに加えて、図面に描写された論理フローは、所望される結果を達成するために、必ずしも、図示された特定の順序、または、連続した順序を必要としない。それに加えて、他のステップが提供され得るか、または、ステップは、

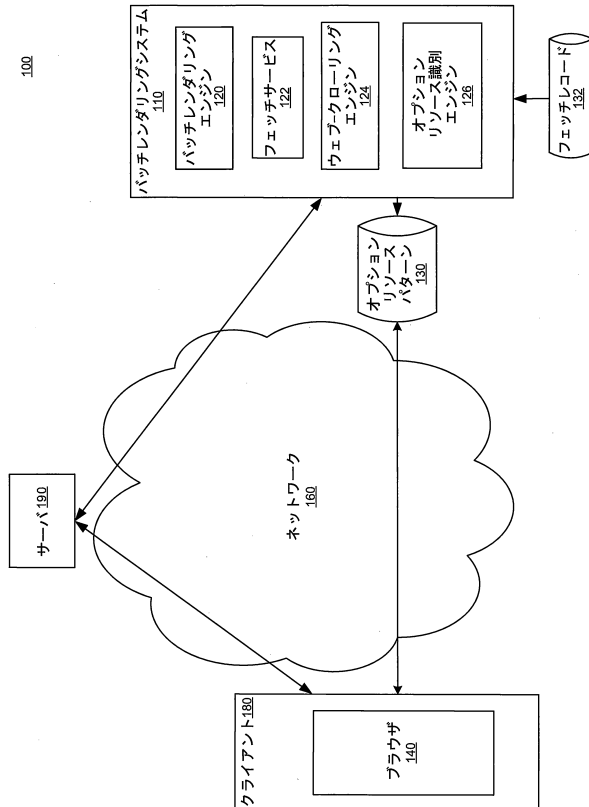
説明されたフローから削除され得、他の構成要素が、説明されたシステムへ追加され得るか、または、説明されたシステムから削除され得る。したがって、他の実施は、以下の特許請求の範囲のスコープ内である。

【符号の説明】

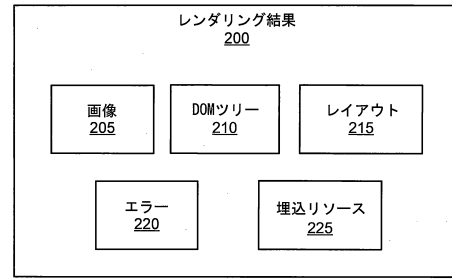
【0062】

100	システム	
110	バッチレンダリングシステム	
120	バッチレンダリングエンジン	
122	フェッチサービス	
124	ウェブ-クローリングエンジン	10
126	オプションリソース識別エンジン	
130	オプションリソースパターン	
132	フェッチレコード	
140	ブラウザ	
160	ネットワーク	
180	クライアント	
190	ホストサーバ	
200	レンダリング結果	
205	画像	
210	DOMツリー	20
215	レイアウト	
220	エラー	
225	埋込リソース	
600	コンピュータデバイス	
602	プロセッサ	
604	メモリ	
606	記憶デバイス	
608	インターフェース	
610	拡張ポート	
612	コントローラ	30
614	拡張ポート	
616	ディスプレイ	
630	サーバ	
632	ラップトップコンピュータ	
634	デスクトップコンピュータ	
636	スマートフォン	
640	外部インターフェース	
644	通信インターフェース	
646	トランシーバ	
648	GPS(全地球測位システム)受信機モジュール	40
700	コンピュータデバイス	
752	プロセッサ	
754	メモリ	
756	ストレージ	
758	ラック、ラックスイッチ	
762	プロセッサ	
764	メモリ	
766	ストレージ	
778	スイッチ	
780	デバイス	50

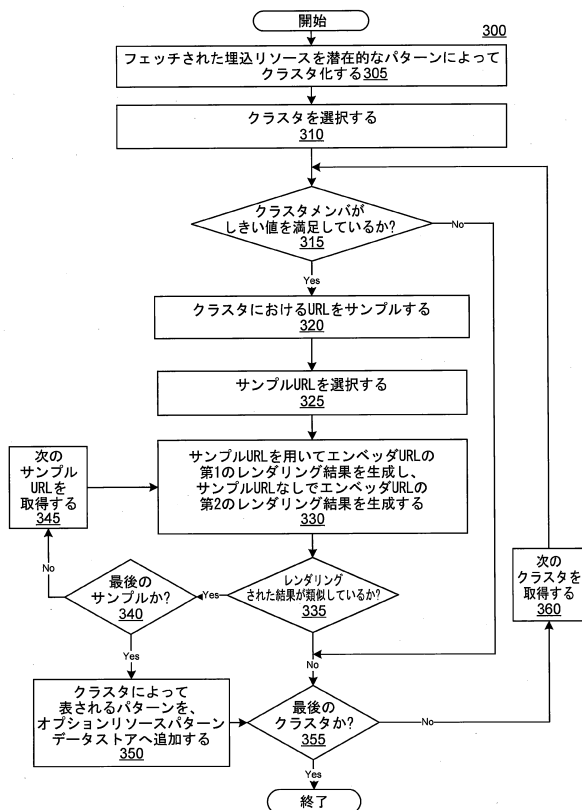
【図 1】



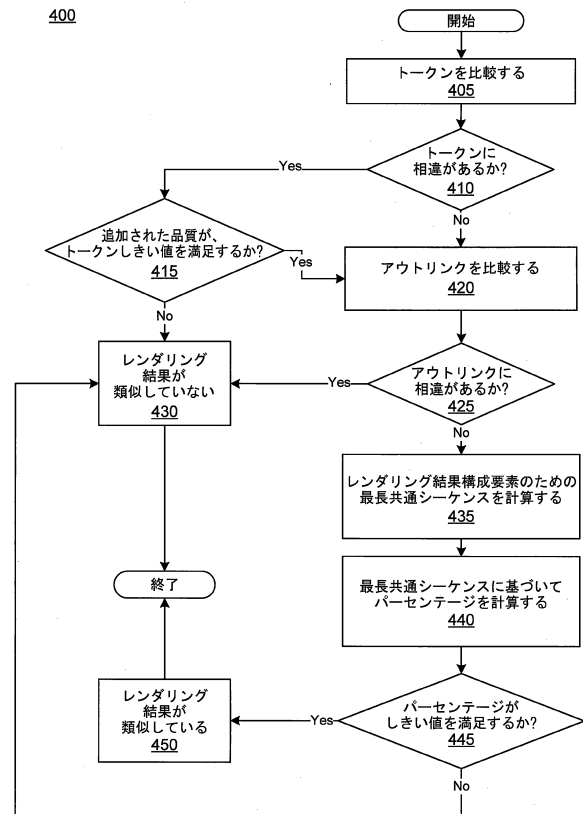
【図 2】



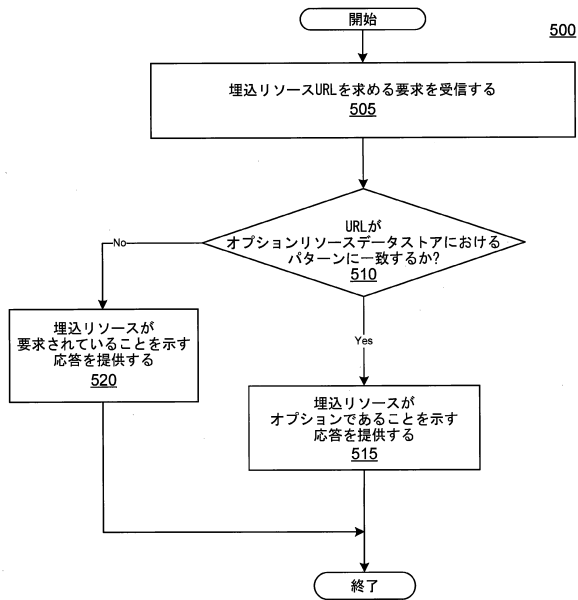
【図 3】



【図 4】



【図 5】



【図 6】

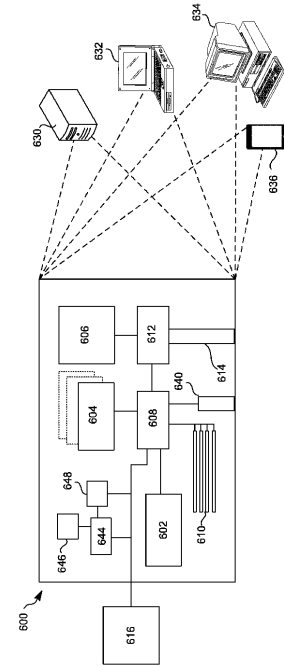


FIG. 6

【図 7】

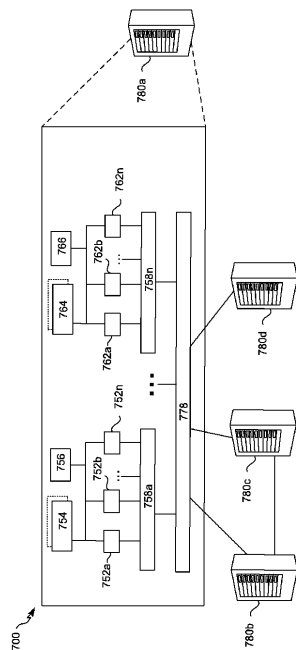


FIG. 7

フロントページの続き

- (72)発明者 ビピン・メン
中華人民共和国・１０００８４・ベイジン・ハイディエン・ディストリクト・ジョングアンチュン
・イースト・ロード・ナンバー・１・ツィンファ・サイエンス・パーク・ビルディング・６
- (72)発明者 マイケル・イースタム
アメリカ合衆国・カリフォルニア・マウンテン・ビュー・アンフィシアター・パークウェイ・１６
００
- (72)発明者 フイ・シュウ
アメリカ合衆国・カリフォルニア・マウンテン・ビュー・アンフィシアター・パークウェイ・１６
００
- (72)発明者 シャオボ・ジョウ
中華人民共和国・１０００８４・ベイジン・ハイディエン・ディストリクト・ジョングアンチュン
・イースト・ロード・ナンバー・１・ツィンファ・サイエンス・パーク・ビルディング・６

審査官 佐々木 洋

- (56)参考文献 米国特許出願公開第２０１１／００５５３１４（US，A１）
米国特許出願公開第２００３／０２２９６７７（US，A１）
特開２００２－０５５８７２（JP，A）
特表２０１２－５２２３２２（JP，A）
特表２００２－５１０８１９（JP，A）

- (58)調査した分野(Int.Cl.，DB名)
G 0 6 F １ ３／０ ０