

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2007-287146

(P2007-287146A)

(43) 公開日 平成19年11月1日(2007.11.1)

(51) Int. Cl.	F I	テーマコード (参考)
G06F 13/00 (2006.01)	G06F 13/00 550A	5B009
G06F 17/21 (2006.01)	G06F 17/21 530T	

審査請求 未請求 請求項の数 4 O L (全 9 頁)

(21) 出願番号	特願2007-103702 (P2007-103702)	(71) 出願人	596170170
(22) 出願日	平成19年4月11日 (2007. 4. 11)		ゼロックス コーポレーション
(31) 優先権主張番号	11/379, 090		XEROX CORPORATION
(32) 優先日	平成18年4月18日 (2006. 4. 18)		アメリカ合衆国 コネチカット州 スタン
(33) 優先権主張国	米国 (US)		フォード、ロング・リッジ・ロード 80
			O
		(74) 代理人	100082005
			弁理士 熊倉 禎男
		(74) 代理人	100067013
			弁理士 大塚 文昭
		(74) 代理人	100086771
			弁理士 西島 孝喜
		(74) 代理人	100109070
			弁理士 須田 洋之

最終頁に続く

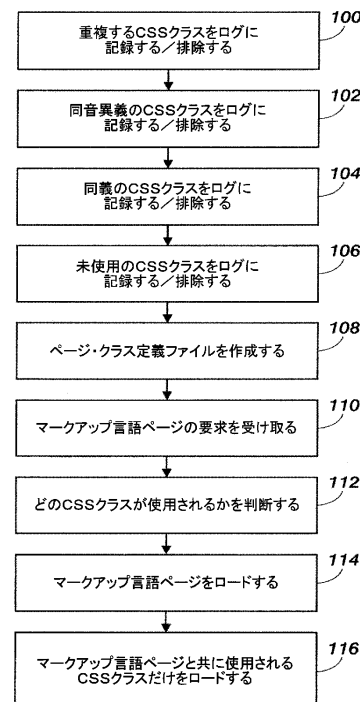
(54) 【発明の名称】 マークアップ言語ファイルの格納及び送信の最適化

(57) 【要約】

【課題】 マークアップ言語ファイルをサーバからブラウザにロードし、格納するとき、速度を増し、コンピュータ・リソースの要求を減少させる。

【解決手段】 サーバにおいて、ブラウザから少なくとも1つのマークアップ言語ページの要求を受け取り、どのカスケーディング・スタイル・シート (CSS) クラスが前記マークアップ言語ページと共に使用されるかを判断し、前記マークアップ言語ページを前記ブラウザにロードし、前記マークアップ言語ページと共に使用されるCSSクラスだけを前記ブラウザにロードし、該ブラウザが、前記CSSクラスを該マークアップ言語ページ内の表示可能なオブジェクトに適用し、かつ、該マークアップ言語ページを表示することを可能にする、ステップを含む。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

サーバにおいて、ブラウザから少なくとも 1 つのマークアップ言語ページの要求を受け取り、

どのカスケーディング・スタイル・シート（C S S）クラスが前記マークアップ言語ページと共に使用されるかを判断し、

前記マークアップ言語ページを前記ブラウザにロードし、

前記マークアップ言語ページと共に使用される C S S クラスだけを前記ブラウザにロードし、該ブラウザが、前記 C S S クラスを該マークアップ言語ページ内の表示可能なオブジェクトに適用し、かつ、該マークアップ言語ページを表示することを可能にする、
ステップを含むことを特徴とする方法。

10

【請求項 2】

サーバにおいて、ブラウザから少なくとも 1 つのマークアップ言語ページの要求を受け取り、

どのカスケーディング・スタイル・シート（C S S）クラスが前記マークアップ言語ページと共に使用されるかを判断し、

同じ名前及び同じプロパティを有する重複する C S S クラスをログに記録し、

同じ名前を有するが、異なるプロパティを有する同音異義の C S S クラスをログに記録し、

同じプロパティを有するが、異なる名前を有する同義の C S S クラスをログに記録し、

20

前記マークアップ言語ページを前記ブラウザにロードし、

前記マークアップ言語ページと共に使用される C S S クラスだけを前記ブラウザにロードし、該ブラウザが、前記 C S S クラスを該マークアップ言語ページ内の表示可能なオブジェクトに適用し、かつ、該マークアップ言語ページを表示することを可能にする、
ステップを含むことを特徴とする方法。

【請求項 3】

マークアップ言語ページの要求を受け取る前に、プロセス内の前記マークアップ言語ページと共に使用されるカスケーディング・スタイル・シート（C S S）クラスを処理する方法であって、

同じ名前及び同じプロパティを有する重複する C S S クラスをログに記録し、

30

同じ名前を有するが、異なるプロパティを有する同音異義の C S S クラスをログに記録し、

同じプロパティを有するが、異なる名前を有する同義の C S S クラスをログに記録し、

いずれのマークアップ言語ページによっても使用されない未使用の C S S クラスをログに記録し、

前記マークアップ言語ページの各々によって使用される C S S クラスを識別するページ・クラス定義ファイルを作成し、

サーバにおいて、ブラウザから少なくとも 1 つの前記マークアップ言語ページの前記要求を受け取り、

前記ページ・クラス定義ファイルを用いて、どの C S S クラスが前記マークアップ言語ページと共に使用されるかを判断し、

40

前記マークアップ言語ページを前記ブラウザにロードし、

前記マークアップ言語ページと共に使用される C S S クラスだけを前記ブラウザにロードし、該ブラウザが、前記 C S S クラスを該マークアップ言語ページ内の表示可能なオブジェクトに適用し、かつ、該マークアップ言語ページを表示することを可能にする、
ステップを含むことを特徴とする方法。

【請求項 4】

コンピュータ・プログラム製品であって、

コンピュータによって実行されるとき、コンピュータに、

サーバにおいて、ブラウザから少なくとも 1 つのマークアップ言語ページの要求を受け

50

取り、

どのカスケーディング・スタイル・シート（ＣＳＳ）クラスが前記マークアップ言語ページと共に使用されるかを判断し、

前記マークアップ言語ページを前記ブラウザにロードし、

前記マークアップ言語ページと共に使用されるＣＳＳクラスだけを前記ブラウザにロードし、該ブラウザが、前記ＣＳＳクラスを該マークアップ言語ページ内の表示可能なオブジェクトに適用し、かつ、該マークアップ言語ページを表示することを可能にする、ステップを含む方法を実行させる命令を格納するコンピュータ使用可能データ記憶媒体を備えることを特徴とするコンピュータ・プログラム製品。

【発明の詳細な説明】

10

【技術分野】

【０００１】

本明細書の実施形態は、一般に、マークアップ言語ファイルをサーバからブラウザにロードすることに関し、より具体的には、そうしたファイルをそのように処理し、格納するとき、速度を増し、コンピュータ・リソースの要求を減少させる方法に関する。

【背景技術】

【０００２】

近年、ウェブ・ベースのアプリケーションは、モデル・ビュー・コントローラ・アーキテクチャに移行してきた。このアーキテクチャのビュー（view）部分は、UIウィジェットを配置し、種々の表示可能なオブジェクトをフォーマット設定するカスケーディング・スタイル・シート（Cascading Style Sheet、CSS）クラスの使用によって主として支配されてきた。今や大部分のブラウザが豊富なCSS能力をサポートしているので、マークアップ・ページにおいて、「ビュー」情報はほとんど必要とされない。代わりに、サーバに、ページについてのフォーマット設定を供給する１組のCSSクラスのロードを要求するために、マークアップ・ページが必要とされる。

20

【発明の開示】

【発明が解決しようとする課題】

【０００３】

従来より、ブラウザがこれらのCSSクラスを受け取るとき、ブラウザは、ページを表示する前に、クラスを構文解析し、それらを種々の表示可能なオブジェクトに適切に適用しなければならない。このようなページをロードし、CSSクラスをロードし、クラスを構文解析し、それらを表示可能なオブジェクトに適用するサイクルが、表示されるページの各々について繰り返される。往々にして、多数のCSSクラスが単一のファイル内に格納され、次に、それらが幾つかのマークアップ言語ページにロードされる。所望のCSSクラスにアクセスするために、幾つかのCSSファイルを単一のマークアップ言語ページにロードする必要があることもある。しばしば、ロードされた幾つかのクラスが、要求中のページに使用されていないことさえある。結局、ブラウザは、必要とされるジョブの実行に必要なものより、CSSクラスのロード及び構文解析により多くの処理サイクルを費やさなければならない傾向が強い。このことは、平均的なディストロップ環境においては問題にならないが、ブラウザ及びサーバが、限られたメモリを備え、ディスクを有していない、同じ安価で動力不足の内蔵プロセッサにおいて実行されるとき、このことが、潜在的なユーザ・インターフェース性能の問題をもたらすことがある。

30

40

【課題を解決するための手段】

【０００４】

本明細書の実施形態は、各々のマークアップ言語ページが、そのページを表示するのに必要とされる最小の組のカスケーディング・スタイル・シート（CSS）クラスを自動的に受け取り、必要とされないCSSクラスは受け取らないことを保証するプロセスを説明する。ロード要求を用いて、全ての必要とされるクラスをブラウザからサーバにロードすることができる。このことは、ブラウザの処理要件を減少させ、CSSクラス・データを格納するのに必要とされるメモリ・フットプリントを最小にしながら達成される。

50

【0005】

本明細書の実施形態は、マークアップ言語ページと共に使用されるCSSクラスを処理する方法を含む。この方法は、同じ名前及び同じプロパティを有する重複するCSSクラスを排除し、同じ名前を有するが、異なるプロパティを有する同音異議のCSSクラスを排除し、同じプロパティを有するが異なる名前を有する同義のCSSクラスを排除し、いずれのマークアップ言語ページによっても使用されない未使用のCSSクラスを排除する。この方法はまた、各々のマークアップ言語ページによって使用されるCSSクラスを識別する使用法マップ(usage map)もログに記録し、このログを用いて、各々の異なるマークアップ言語ページ(サーバが利用できる)によって使用されるCSSクラス(サーバが利用できる)を列挙するページ・クラス定義ファイルを作成する。

10

【0006】

上述の処理の後(潜在的には、上述の処理中又は処理の前)、方法は、サーバにおいて、ブラウザからマークアップ言語ページの少なくとも1つの要求を受け取る。次に、方法は、ページ・クラス定義ファイルを用いて、どのCSSクラスが、要求されたマークアップ言語ページと共に使用されるかを判断することができる。次に、方法は、マークアップ言語ページをロードし、マークアップ言語ページと共に使用されるCSSクラスだけをブラウザにロードする。

【0007】

サーバが、CSSクラスを生成するために使用されるデータをロードしているとき、方法は、各CSSクラスの1つのインスタンスだけをロードする。さらに、各CSSクラスは、少なくとも1つのプロパティを含み、CSSクラスをロードすることは、各プロパティの1つのインスタンスだけをロードすることをさらに含む。ブラウザからの単一のロード要求を用いて、マークアップ言語ページと共に使用される全てのCSSクラスをブラウザにロードすることができる。CSSクラスは、マークアップ言語ページについてのフォーマット設定情報を提供する。

20

【0008】

このことは、重複するCSSクラス、同音異議のCSSクラス、同義のCSSクラス、又は未使用のCSSクラスを処理する必要なく、ブラウザが、CSSクラスをマークアップ言語ページ内の表示可能なオブジェクトに適用し、かつ、マークアップ言語ページを表示することを可能にする。さらに、要求されるマークアップ言語ページが使用しない他の適切なCSSクラスも処理されない。こうしたCSSクラスを処理から排除することによって、ブラウザ及びサーバの限られたリソース(メモリ等)がより有効に用いられ、これにより速度が増し、大きなメモリを備えた、高価な付加的なプロセッサに対する必要性が低減される。

30

【発明を実施するための最良の形態】

【0009】

添付の図面を参照して、システム及び方法の種々の例示的な実施形態が、以下に詳細に説明される。

本明細書の実施形態は、CSSクラスを組織化する方法を含み、CSSクラスを参照するページの数に関係なく、クラス属性及びクラスが一度だけ定められる。このことにより、CSSクラスをホストするのに必要とされるメモリの量が減少される。本明細書の実施形態はまた、ページに必要とされる最小の組のクラスだけをブラウザに供給し、これにより未使用のクラスを処理するオーバーヘッドが減少される。

40

【0010】

このプロセスには、2つのコンポーネント、すなわち前処理コンポーネント及び実行時コンポーネントがある。CSS前処理コンポーネントは、CSSの参照のためにマークアップ言語ページを走査し、使用法レポートを生成する。実行時サーバ・コンポーネントは、非常に高速のアルゴリズムを使用し、圧縮されたフォーマットからブラウザ読み取り可能なCSSクラスを生成する。

以下に説明されるプロセスは、性能を最適化し、開発者がCSSクラスの生成を管理す

50

るのを助けるように設計される。最近定められたクラスが、別の開発者によって以前に定められているかどうか、又はクラス名が異なるプロパティと重複するかどうか、すぐに分かる。また、マークアップ言語が変わると、このことが、以前に定められたクラスを時代遅れにすることがある。デバッグのために最も困難な状況の1つは、同じクラス名が定められているが異なるプロパティを有している場合である。CSS最適化ログ・ファイルは、これらの状況を早期に報告し、何らかのデバッグの試みが費やされる前に、それらの状況を無くすことを可能にする。

【0011】

図1に示されるように、本明細書の一実施形態が、マークアップ言語ページと共に使用されるカスケーディング・スタイル・シート(CSS)クラスを処理する方法を提供する。この方法は、項目100において、同じ名前及び同じプロパティを有する重複するCSSクラスをログに記録する及び／又は排除し、項目102において、同じ名前を有するが異なるプロパティを有する同音異義のCSSクラスをログに記録する及び／又は排除し、項目104において、同じプロパティを有するが、異なる名前を有する同義のCSSクラスをログに記録する及び／又は排除し、項目106において、いずれのマークアップ言語も用いていない未使用のCSSクラスをログに記録する及び／又は排除する。この方法は、各々のマークアップ・ページによって用いられるCSSクラスを識別する使用法マップもログに記録し、項目108において、このログを使用し、ページ・クラス定義ファイルを作成する。これらの処理点は、一般に、CSS前処理コンポーネントを構成する。

【0012】

項目110において、上記の前処理の後(潜在的には、上記の処理中又は処理の前)、方法の実行時コンポーネントが、サーバにおいて、ブラウザからマークアップ言語ページの少なくとも1つの要求を受け取る。次に、項目112において、方法は、ページ・クラス定義ファイルを用いて、どのCSSクラスが、要求されたマークアップ言語ページと共に使用されるかを判断することができる。次に、方法は、マークアップ言語ページをロードし(項目114)、項目116において、該マークアップ言語ページと共に使用されるCSSクラスだけをブラウザにロードする。

【0013】

項目116においてCSSクラスをロードするとき、方法は、各CSSクラスの1つのインスタンスだけをロードする。さらに、各CSSクラスは、少なくとも1つのプロパティを含み、項目116においてCSSクラスをロードすることは、プロパティの各々の1つのインスタンスだけをロードすることをさらに含む。ブラウザからの単一のロード要求(項目110)を用いて、マークアップ言語ページと共に使用されるCSSクラスの全てをブラウザにロードすることができる。CSSクラスは、マークアップ言語ページについてのフォーマット設定情報を提供する。

【0014】

このことは、重複するCSSクラス、同音異義のCSSクラス、同義のCSSクラス、又は未使用のCSSクラスを処理する必要なしに、ブラウザが、CSSクラスをマークアップ言語ページ内の表示可能なオブジェクトに適用し、かつ、マークアップ言語ページを表示することを可能にする。さらに、要求されているマークアップ言語ページが使用しない他の適切なCSSクラスも処理されない。処理からこうしたCSSクラスを排除することによって、ブラウザ及びサーバの限られたリソースがより効率的に使用され、これにより速度が増し、付加的なメモリを備えた、より高価でより高速の付加的なプロセッサに対する必要性が低減される。

【0015】

図2は、所定のサーバ210(本明細書に説明される方法を実行するための命令を格納するコンピュータ使用可能データ記憶媒体を含む)が利用できる全てのマークアップ言語ページ204が、サーバ210が利用できる全てのCSSクラス定義200に対して比較される、CSS最適化処理コンポーネントを示す。各々のマークアップ言語ページが、CSS要素クラスの用法について走査される。各々のCSSクラス定義ファイル200が

構文解析される。CSS最適化プロセスは、これらの入力进行分析し、次いで、マークアップ言語ページ又はCSSクラス定義ファイル200における問題を識別するために、開発者206に有用なレポートが作成される。

【0016】

より具体的には、項目206において、以下の情報、すなわち、重複するクラス（同じ名前及びプロパティを使用するクラス）、同音異義のクラス（同じ名前を使用するが、異なるプロパティを使用するクラス）、同義のクラス（異なる名前を使用するが、以前に定められたクラスと同じプロパティを使用するクラス）、未使用のクラス（いずれの分析されたマークアップ言語ページも使用していないクラス）、及びどのページがどのCSSクラスを使用するかを識別する使用法マップが、ログに記録される。さらに、所望であれば、本明細書の実施形態は、いずれの重複するCSSクラス、同音異義のCSSクラス、同義のCSSクラス、又は未使用のCSSクラスも自動的に消去又は削除することができる。

【0017】

開発者206のための情報ログに加えて、使用法マップを用いて、最適化されたCSSページ・クラス定義を含むファイル212が生成される。実行時に、要求されるページによって使用されるCSSクラスを生成するために、ウェブ・サーバによってCSSページ・クラス定義ファイル212が用いられる。ページ・クラス定義212ファイルは、サーバ210が読み取るデータを含む。多くのマークアップ言語ページにおいて多数のCSSクラスを用いるシステムにおいて、本明細書の実施形態の場合は、メモリ・フットプリントが、同じ能力を達成するのに従来必要とされるCSSファイルより著しく小さい。

【0018】

図3により詳細に示される、共通入力ファイル202の逐語的なコピーも、ページ・クラス・ファイル212内に格納される。共通ファイルのコンテンツは、あらゆるページに与えられるCSSクラスを含む。これらのクラスは、圧縮されず、ブラウザに送られる前にサーバによって更に処理する必要がある。ページ・クラス・ファイルの残りは、ページ名302に対する少なくとも1つのページ名参照を含む。各々のページ名302は、少なくとも1つのページ名参照を有する。各々のクラス名304は、少なくとも1つのプロパティ名参照を有する。各々のプロパティ名306は、プロパティ値（例えば、フォーマット特性）を定める。このように、共通入力ファイル202は、マークアップ言語内の要素クラス注釈と共に定めることができないCSSのために使用される。幾つかの形態のCSS構文は、マークアップ言語の要素と関連付けられたクラス属性に基づいて適用されない。CSS命令は、マークアップの状態に基づいて適用される。このタイプのCSS構文は、擬似要素として知られており、時として、この方法の前処理段階中に検出できないことがある。共通ファイルは、圧縮されていない形でサーバ内に格納されており、付加的な処理を必要としない各マークアップ・ページ要求、又は全てのページに共通のCSSクラスの各マークアップ・ページ要求に送信される。肝心なのは、開発者が何を共通ファイル内に置こうと、ページ特有のCSSクラスに加えてCSS情報が与えられたときに、直接ブラウザに行くという点である。このことは、マークアップ・ページが、依然として非要素ベースのCSS命令を受け取ることを可能にする。

【0019】

CSS最適化プロセスの実行時コンポーネントは、迅速に要素にアクセスできるように、ウェブ・サーバに、ページ・クラス定義ファイルをインポートし、これを格納するように要求する。ブラウザにロードされたとき、マークアップ言語ページは、マーク・ページの名前と同じ名前を有するCSS文書を参照する。例えば、マークアップ言語ページの参照が、<http://localhost:8081/myPage.xhtml>であった場合、関連したCSS文書への参照は、<http://localhost:8081/myPage.css>となる。これは、一例にすぎず、異なるシステムは、CSS文書にアクセスするために異なる機構を定めることができる。

【0020】

図4は、ブラウザ400とサーバ402との間のフローを示す。項目404は、上述の

10

20

30

40

50

ように、サーバ 402 が、どのクラスが各ページと関連付けられているかを知ることができるページ・クラス定義ファイル 212 をインポートするプロセスを表す。項目 406 は、ページ要求（CSS クラス要求とみなすことができる）を表す。項目 410 において、サーバ 402 は、ページ・クラス定義ファイル 212 を用いて、要求されたページについての CSS クラスを生成する。次に、項目 408 は、ブラウザ 400 へのページ及び CSS クラスの戻りを表す。

【0021】

このように、本明細書の実施形態の場合、ウェブ・サーバは、CSS 文書をマークアップ言語ページに供給していることを識別すると、どのクラス及びプロパティが必要とされるかを迅速に参照し、必要とされるクラス（例えば、必要とされるクラスの正確な組）だけをブラウザに送ることができる。この手法の幾つかの利点は、必要とされる CSS クラス、各プロパティの 1 つのインスタンス、各クラスの 1 つのインスタンスだけがロードされるため、かつ、マークアップ言語ページごとにクラスへの参照だけがなされるため、減少されたメモリ・フットプリントが生成されることである。さらに、本明細書の実施形態の場合、ブラウザは、所定のページについて使用されないクラスを構文解析するか又は他の方法で処理する必要がないので、本明細書の実施形態は、ブラウザによる処理サイクルを減少させる。代わりに、本明細書の実施形態の場合、必要とされるクラスだけがブラウザに与えられる。本明細書の実施形態の場合、CSS クラスの開発全体にわたって、より多くの制御がなされ、上述のログ能力に起因して CSS クラスの競合が生じる可能性は少ない。

10

20

【0022】

本明細書の実施形態は、何らかの目的のために表示機能を実行する、デジタル複写機、製本機械、ファクシミリ装置、多機能機械等のようないずれかの装置と共に用いることができる。本発明は、プリンタに限定されるものではなく、ユーザ・インターフェースを表示／フォーマット設定するために CSS を用いる全てのシステムに適用される。

【図面の簡単な説明】

【0023】

【図 1】 1 つの方法の実施形態を示すフロー図である。

【図 2】 ここに説明される実施形態を用いるシステムの概略図である。

【図 3】 図 2 に示されるページ・クラス定義ファイルの概略図である。

【図 4】 ここに説明される実施形態を用いる実行時環境の概略図である。

30

【符号の説明】

【0024】

200 : CSS クラス定義ファイル

202 : 共通ファイル

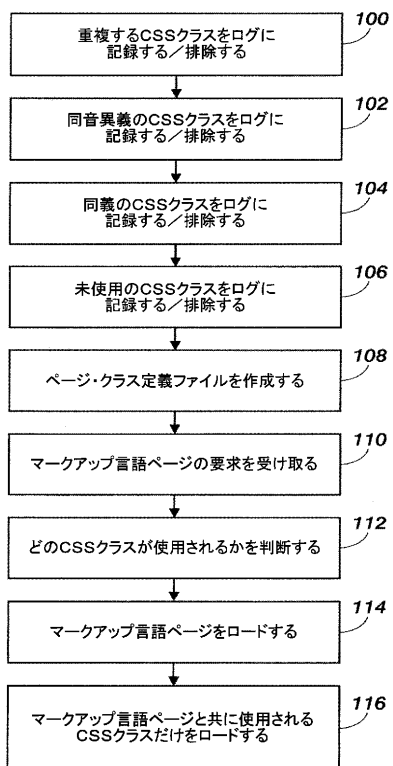
206 : 開発者

210、402 : サーバ

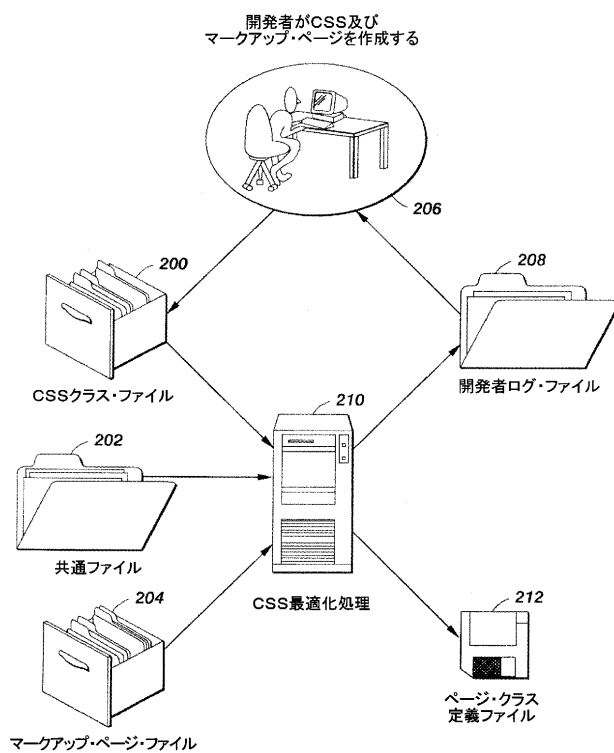
212 : CSS ページ・クラス定義ファイル

400 : ブラウザ

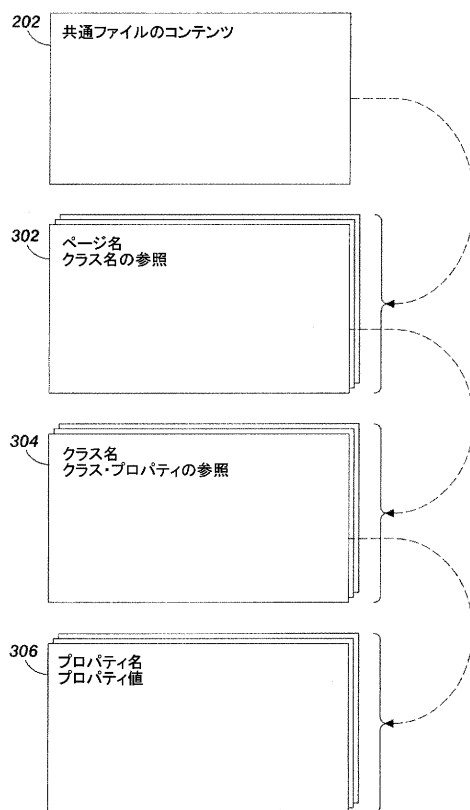
【図 1】



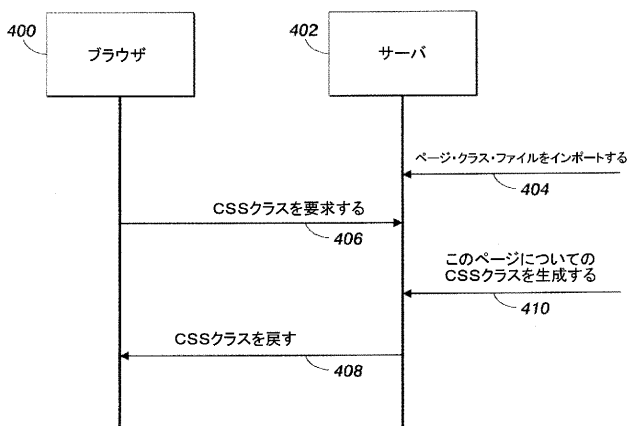
【図 2】



【図 3】



【図 4】



フロントページの続き

(72)発明者 ウェンデル エル キブラー

アメリカ合衆国 ニューヨーク州 1 4 5 1 9 オンタリオ バーク ロード 2 9 1

Fターム(参考) 5B009 NA03 NA11 QA06 RB35