

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4208081号
(P4208081)

(45) 発行日 平成21年1月14日(2009.1.14)

(24) 登録日 平成20年10月31日(2008.10.31)

(51) Int.Cl.		F I			
G06Q	50/00	(2006.01)	G06F	17/60	1 3 2
G06Q	30/00	(2006.01)	G06F	17/60	3 3 0
G06Q	10/00	(2006.01)	G06F	17/60	5 0 4
G06F	15/00	(2006.01)	G06F	15/00	3 1 0 A
			G06F	15/00	3 1 0 D

請求項の数 16 (全 22 頁)

(21) 出願番号	特願2004-377805 (P2004-377805)	(73) 特許権者	390009531
(22) 出願日	平成16年12月27日(2004.12.27)		インターナショナル・ビジネス・マシーンズ・コーポレーション
(65) 公開番号	特開2006-185160 (P2006-185160A)		INTERNATIONAL BUSINESS MACHINES CORPORATION
(43) 公開日	平成18年7月13日(2006.7.13)		アメリカ合衆国10504 ニューヨーク州 アーモンク ニュー オーチャードロード
審査請求日	平成19年11月12日(2007.11.12)	(74) 代理人	100086243 弁理士 坂口 博
早期審査対象出願		(74) 代理人	100091568 弁理士 市位 嘉宏
		(74) 代理人	100108501 弁理士 上野 剛史

最終頁に続く

(54) 【発明の名称】 複数のウェブサイトにパーソナライズされた価値を追加するためのシステム、ウェブサーバ、方法およびプログラム

(57) 【特許請求の範囲】

【請求項1】

第1のウェブサーバと、単一または複数からなる第2のウェブサーバとを備え、当該第1のウェブサーバおよび当該第2のウェブサーバにネットワークを介して接続されるクライアント装置のウェブブラウザにウェブページを表示可能なシステムであって、

前記第1のウェブサーバは、既存のウェブページに対して付加される新たな価値である付加価値として、同じ付加価値を持たせるウェブサーバの集合として定義された当該第1のウェブサーバおよび当該第2のウェブサーバのドメインと、当該第1のウェブサーバおよび当該第2のウェブサーバのそれぞれに当該付加価値を提供するファイルのURL (Uniform Resource Locators) とのコード情報を記憶装置に登録し、

前記クライアント装置が前記付加価値を提供するファイルのURLへアクセスすると、

前記第1のウェブサーバまたは前記第2のウェブサーバは、前記付加価値を提供するためのスクリプトプログラムを前記ウェブブラウザに送り、

前記ウェブブラウザは、前記第1のウェブサーバまたは前記第2のウェブサーバから送られた既存のウェブページを表示し、前記スクリプトプログラムに含まれるAPI (Application Program Interface) を操作して当該既存のウェブページに前記付加価値を提供することを特徴とするシステム。

【請求項2】

前記第1のウェブサーバは、ウェブページ上に前記付加価値を持たせる他のウェブサーバのウェブページへのリンクがある場合に、フレーム内にウェブページをロードしたとき

に、当該他のウェブサーバ上の付加価値を提供するURLに遷移するようにDOM(Document Object Model)を操作してAタグの属性を変更し、前記第2のウェブサーバは、前記第1のウェブサーバから取得したコード情報に基づいて、ウェブページ上に前記付加価値を持たせる他のウェブサーバのウェブページのリンクがある場合に、フレーム内にウェブページをロードしたときに、当該他のウェブサーバ上の付加価値を提供するURLに遷移するようにDOM(Document Object Model)を操作してAタグの属性を変更することを特徴とする請求項1記載のシステム。

【請求項3】

前記第1のウェブサーバおよび前記第2のウェブサーバの中から特定のウェブサーバが代表してクッキーの情報を管理し、前記付加価値に関するパーソナライズのための情報は、当該特定のウェブサーバを通して当該クッキーの情報にアクセスすることを特徴とする請求項1記載のシステム。

10

【請求項4】

第1のウェブサーバと、単一または複数からなる第2のウェブサーバとを備え、当該第1のウェブサーバおよび当該第2のウェブサーバにネットワークを介して接続されるクライアント装置のウェブブラウザにウェブページを表示可能なシステムであって、

前記第1のウェブサーバおよび前記第2のウェブサーバは、それぞれ固有のコンテンツが格納されると共に当該固有のコンテンツに付加価値を持たせるためにフレームを用いた文書ファイルを生成するための生成手段を備え、

前記第1のウェブサーバは、既存のウェブページに対して付加される新たな価値である付加価値として、同じ付加価値を持たせるウェブサーバの集合として定義された当該第1のウェブサーバおよび前記第2のウェブサーバのドメインと、当該第1のウェブサーバおよび当該第2のウェブサーバのそれぞれに当該付加価値を提供するファイルのURLとのコード情報を記憶装置に登録し、

20

前記クライアント装置が前記付加価値を提供するファイルのURLへアクセスすると、前記第1のウェブサーバまたは前記第2のウェブサーバは、前記付加価値を提供するためのスクリプトプログラムを前記ウェブブラウザに送り、

前記ウェブブラウザは、前記第1のウェブサーバまたは前記第2のウェブサーバから送られた既存のウェブページを表示し、前記スクリプトプログラムに含まれるAPI(Application Program Interface)を操作して当該既存のウェブページに前記付加価値を提供することを特徴とするシステム。

30

【請求項5】

前記第1のウェブサーバは、前記ウェブブラウザからパラメータとして渡されたURLが、同じ付加価値を持たせるウェブサーバの集合として事前に登録されたドメインのものである場合に、クッキーの情報を読むためのスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信することを特徴とする請求項4記載のシステム。

【請求項6】

前記第2のウェブサーバは、前記ウェブブラウザからパラメータとして渡された前記クッキーの情報を引数として関数を呼び出すコードが記述されたスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信することを特徴とする請求項5記載のシステム。

40

【請求項7】

前記付加価値は、表示されたウェブページに対して実行されるアクセシビリティのための機能であることを特徴とする請求項4記載のシステム。

【請求項8】

ネットワークを介して接続されるウェブブラウザに、既存のウェブページに対して付加される新たな価値である付加価値として同じ付加価値を提供する複数のウェブサーバの、主となるウェブサーバであって、

装置固有のオリジナルコンテンツを格納する格納手段と、

前記付加価値の提供のために、フレームを用いた文書ファイルを生成する生成手段と、

50

前記複数のウェブサーバにて生成された文書ファイルから参照され、前記ウェブブラウザで前記付加価値を実行させるための文書ファイルを送信する送信手段と、

前記付加価値を提供するウェブサーバの集合として定義された前記複数のウェブサーバのドメインと、当該複数のウェブサーバに当該付加価値を提供するスクリプトプログラムのURLとのコード情報を記憶装置に登録する手段とを含み、

クライアント装置が前記付加価値を提供するスクリプトプログラムのURLへアクセスすると、前記複数のウェブサーバが前記付加価値を提供するためのスクリプトプログラムをウェブブラウザに送った後、当該ウェブブラウザが当該複数のウェブサーバから送られた既存のウェブページを表示し、当該ウェブブラウザが当該スクリプトプログラムに含まれるAPI(Application Program Interface)を操作して当該既存のウェブページに前記付加価値を提供することを可能とするウェブサーバ。

10

【請求項 9】

前記ウェブブラウザからパラメータとして渡されたURLが事前に登録された前記複数のウェブサーバのドメインのものがどうかを判断し、事前に登録されたものである場合に、自らが提供するクッキーの情報を引数として関数を呼び出すコードが記述されたスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信する第2の送信手段を更に備えたことを特徴とする請求項8記載のウェブサーバ。

【請求項 10】

前記スクリプトプログラムは、前記クッキーの情報を引数としてスクリプトプログラムの関数を呼び出すコードが記述されていることを特徴とする請求項9記載のウェブサーバ

20

【請求項 11】

前記複数のウェブサーバを構成する他のウェブサーバのウェブページを表示している前記ウェブブラウザから前記クッキーの情報を書くスクリプトプログラムの要求を受信した場合に、当該クッキーの情報を書くスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信する第3の送信手段を更に備えたことを特徴とする請求項10記載のウェブサーバ。

【請求項 12】

複数のウェブサーバにネットワークを介してウェブブラウザが接続されるシステムにて、既存のコンテンツに修正を加えることなく、既存のウェブページに対して付加される新たな価値である付加価値を提供するための方法であって、

30

前記複数のウェブサーバを構成する特定のコンピュータ装置に、同じ付加価値を持たせるウェブサーバの集合として定義された第1のウェブサーバおよび第2のウェブサーバのドメインと、当該第1のウェブサーバおよび当該第2のウェブサーバのそれぞれに当該付加価値を提供するファイルのURL(Uniform Resource Locators)とのコード情報を記憶装置に登録し、同じ付加価値を持たせる集合として定義された当該複数のウェブサーバのドメインと当該付加価値を提供するスクリプトプログラムのURLとのコード情報を予めメモリに格納し、

前記ウェブブラウザからパラメータとして渡されたURLが、前記メモリに予め格納されたものであるか否かを当該メモリに格納された前記コード情報から確認し、予め当該メモリに格納されたものである場合に、クッキーの情報を読むためのスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信することを特徴とし、

40

クライアント装置が前記付加価値を提供するスクリプトプログラムのURLへアクセスすると、前記複数のウェブサーバが前記付加価値を提供するためのスクリプトプログラムをウェブブラウザに送った後、当該ウェブブラウザが当該複数のウェブサーバから送られた既存のウェブページを表示し、当該ウェブブラウザが当該スクリプトプログラムに含まれるAPI(Application Program Interface)を操作して当該既存のウェブページに前記付加価値を提供することを可能とする方法。

【請求項 13】

クッキーの情報が前記複数のウェブサーバの文書ファイルから参照され、アプリケーション

50

ョン固有の付加価値を前記ウェブブラウザにて実行させることを特徴とする請求項 1 2 記載の方法。

【請求項 1 4】

前記複数のウェブサーバを構成する他のウェブサーバのウェブページを表示している前記ウェブブラウザからクッキーの情報を書くスクリプトプログラムの要求を受信した場合に、当該クッキーの情報を書くスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信することを特徴とする請求項 1 2 記載の方法。

【請求項 1 5】

複数のウェブサーバにネットワークを介してウェブブラウザが接続されるシステムにて、既存のコンテンツに修正を加えることなく、既存のウェブページに対して付加される新たな価値である付加価値を提供するための当該ウェブサーバとして機能するコンピュータに、

10

同じ付加価値を持たせる集合として定義された前記複数のウェブサーバのドメインと当該付加価値を提供するスクリプトプログラムの URL とのコード情報を、前記コンピュータが備えるメモリに予め格納させる機能と、

前記ウェブブラウザからパラメータとして渡された URL が、前記メモリに予め格納されたものであるか否かを当該メモリに格納された前記コード情報から確認し、予め当該メモリに格納されたものである場合に、クッキーの情報を読むためのスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信する機能とを実現させ、

クライアント装置が前記付加価値を提供するスクリプトプログラムの URL へアクセスすると、前記複数のウェブサーバが前記付加価値を提供するためのスクリプトプログラムをウェブブラウザに送った後、当該ウェブブラウザが当該複数のウェブサーバから送られた既存のウェブページを表示し、当該ウェブブラウザが当該スクリプトプログラムに含まれる A P I (Application Program Interface) を操作して当該既存のウェブページに前記付加価値を提供することを可能とするプログラム。

20

【請求項 1 6】

前記複数のウェブサーバを構成する他のウェブサーバのウェブページを表示している前記ウェブブラウザからクッキーの情報を書くスクリプトプログラムの要求を受信した場合に、当該クッキーの情報を書くスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信する機能を更に実現させる請求項 1 5 記載のプログラム。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、複数のウェブ(Web)サイトにネットワークを介して接続されるウェブブラウザにてウェブページを表示可能なネットワークシステムに係り、より詳しくは、複数のウェブサイトにパーソナライズされた価値を追加するためのシステム等に関する。

【背景技術】

【0002】

近年、ウェブページのアクセシビリティ(Accessibility)が大きく注目されている。これは、ウェブを利用する全ての人々が、年齢や身体的制約、利用環境等の影響を受けることなく、ウェブ上で提供されている情報にアクセスでき、また、コンテンツや機能を利用できるようにする技術である。関連する従来技術として、例えば、ウェブサイトの読みたい文章の段落の冒頭にマウスを動かすだけで、自動的に音声で読み上げ、その文字を見易いように拡大表示させるものが存在する(例えば、非特許文献 1 参照。)。この非特許文献 1 に記載の技術では、読み上げるときの速度を変えたり、音量を調整することも可能である。色弱や白内障などの病気の方でも読み易いように、文字の色とその背景色を変えることもできる。そして、このソフトでは、P C (パーソナルコンピュータ) にインストールする手間を省くための数々の工夫がなされている。例えば、P C のユーザによって最初に表示される使用条件に対する承諾がなされると、自動的に ActiveX (登録商標) オブジェクトが起動し、前提ソフトウェアの有無をチェックして自動的にダウンロードおよびセットアッ

40

50

プが行われる。これによって、ユーザの手間をかけることなく、このソフトを直ぐに使用することができる。そして、このソフトを用いると、既存のウェブサイトを変更することなく、画面の拡大や読み上げを行うことが可能となる。また、JavaScript(登録商標)とActiveXを用いたアクセシビリティへの他の配慮技術として、例えば非特許文献2も存在する。

【0003】

また、従来、ドメインをJavaScriptで設定することにより、ドメインの範囲を広げる技術が存在する(例えば、非特許文献3参照。)。この技術では、フレームをまたがるセキュリティの実装として、同じドメイン内でホストされているページだけが、特定のページの内容を安全にスクリプトできる。例えば、それぞれのドキュメントにて、

```
document.domain = " ibm.com " ;
```

を実行すれば、w3.tri.ibm.comとw3.watson.ibm.comとのドキュメント間で、互いに相手を制御することができる。

ここで、非特許文献4によると、document.domainは設定できないことになっているが、Internet Explorer(登録商標)やMozilla(登録商標)などのブラウザでもこの方法により、設定することが可能である。

【0004】

更に、従来技術として、ファイルの拡張子を.htaとすることにより、セキュリティの制限を回避する技術が存在している(例えば、非特許文献5参照。)。

また、公報記載の従来技術として、パーソナル化サーバというものを設け、利用者は、複数のウェブサイトへアクセスするときに、このパーソナル化サーバを利用することで、共通のパーソナライズを可能とする技術が存在している(例えば、特許文献1参照。)。

更に、他の公報には、クッキー(Cookie)をサーバ側に保管することによって、複数のクライアントで同じクッキーを利用し、ウェブサイトにアクセスすることを可能とする技術が開示されている(例えば、特許文献2参照。)。

【0005】

【非特許文献1】“インターネット閲覧支援ソフトウェア 「らくらくウェブ散策」”〔online〕、日本IBM〔2004年12月14日検索〕、インターネット<http://www.research.ibm.com/tri/news/ibm_users/tritech_11.htm>

【非特許文献2】“ソフトウェアZoomSight(ズームサイト)”〔online〕、日立公共システムエンジニアリング株式会社〔2004年12月14日検索〕、インターネット<<http://www.gp.hitachi.co.jp/eigy/product/zoomsight/>>

【非特許文献3】“[InetSDK] フレームをまたがったのスクリプトで“アクセス拒否””〔online〕、Microsoft.com Japan サイト〔2004年12月14日検索〕、インターネット<<http://support.microsoft.com/default.aspx?scid=kb;JA;167796>>

【非特許文献4】“Document Object Model (DOM) Level 2 HTML Specification Version 1.0”〔online〕、W3C Recommendation〔2004年12月14日検索〕、インターネット<<http://www.w3.org/TR/2003/REC-DOM-Level-2-HTML-20030109/>>

【非特許文献5】“[InetSDK] クロスフレームスクリプトが可能なページをHTMLアプリケーションで作成する”〔online〕、Microsoft.com Japan サイト〔2004年12月14日検索〕、インターネット<<http://support.microsoft.com/default.aspx?scid=kb;ja;241754>>

【特許文献1】特表2003-520361号公報

【特許文献2】特開2004-103022号公報

【発明の開示】

【発明が解決しようとする課題】

【0006】

ここで、非特許文献1や非特許文献2に記載された技術では、ActiveXを使うことから、OS(Operating System)にWindows(登録商標)を使い、ウェブブラウザにInternet Explorerを使う必要がある。そのために、それ以外のOSやウェブブラウザを使用できず、汎

10

20

30

40

50

用性に欠ける問題がある。

また、非特許文献3を用いたドメインの設定方法では、それぞれのウェブページでこのJavaScriptを実行するように変更しなければならない。この変更をウェブサイト上の全てのウェブページに対して行おうとすると、非常に大きな手間がかかってしまう。また、非常に関連性の高いサイトであっても第2レベルのドメインが同じでなければ、この方法を用いることができないという問題も残る。

更に、非特許文献5を用いた方法では、OSにWindowsを使う必要があり、それ以外のOSを使うことができない。また、セキュリティの制限を一切回避してしまうので、利用者が本当に信頼しているサイトでしかHTMLアプリケーションを実行することができないという問題もある。

10

【0007】

一方、特許文献1に記載された方法では、サーバに利用者の個人情報を保管することになる。しかしながら、近年、個人情報の流出が大きな社会問題となっており、サーバで個人情報を保管する場合には、情報管理用の系を構築する必要性が生じ、情報の漏洩を防止するために、この管理を慎重に行うことが要求される。ここで注意すべき点は、ただ認証などの技術を用いた情報管理用の系を構築するだけでは不十分だという点である。問題になっている個人情報の流出事故では、個人情報にアクセスできる権限を持っている人が個人情報を不正に流出するということが多い。そのため、サーバに個人情報を保管することは非常に大きなリスクを抱えることになる。また、それぞれの利用者ごとの情報をサーバで保管するため、利用者の数が増えると、保管するデータが膨大になるという問題もある。

20

更に、特許文献2に記載された方法では、複数のウェブサイト間でパーソナライズのための情報を共有することができないという問題がある。また、サーバで個人情報を保管することから、上述の特許文献1で述べたものと同様な問題も抱えている。

【0008】

このように、近年、アクセシビリティを高めることなどを目的として、既存ウェブサイトに修正を加えることなく新たな価値を追加するという要求が増えてきたが、このときに、ActiveXのように特定のクライアント環境でしか利用できない技術ではなく、JavaScriptのように多くのクライアント環境で利用できる技術を使うことが望ましい。しかしながら、JavaScriptを用いて別のフレームやウィンドウを制御しようとする、ウェブブラウザのセキュリティの制限でそのフレームやウィンドウが同じドメインに存在する場合しか制御できない。そのため、複数のウェブサイトに統一的に価値を追加することができなかった。

30

【0009】

また、このように付加価値を加えるときに、この付加価値をパーソナライズすることができれば更に価値を高めることができる。パーソナライズのための情報はサーバで保存する方法とクライアントのクッキーに保存する方法があるが、個人情報をサーバで管理するには情報管理用の系を構築する必要があるとともに、情報が漏洩しないようにするために、この管理を慎重に行わなければならない。パーソナライズのための情報をクッキーに保存すればサーバで保存する必要はないが、この方法ではドメイン間でパーソナライズのための情報を共有することができない。

40

【0010】

本発明は、以上のような従来技術の問題点に鑑みてなされたものであって、その目的とするところは、複数の既存のウェブサイトに対する変更を最小限に留めた状態にて、アクセシビリティを高めるなどの新たな価値を追加(付加価値を提供)することにある。

また他の目的は、クライアントの環境が特定のOSやウェブブラウザに依存しないようにすることにある。

更に他の目的は、関連のある複数のウェブサイト間でページを遷移したときも利用者負担をかけることなく、新たに追加した価値を利用できるようにすることにある。

また更に他の目的は、関連のある複数のウェブサイト間でパーソナライズのための情報

50

をサーバに保管することなく共有することにある。

【課題を解決するための手段】

【0011】

かかる目的のもと、本発明では、既存の複数のウェブサイトに対し、少しのコンテンツを追加することで、既存のコンテンツに修正を加えることなく新たな価値を追加する。そのために、まず、同じ付加価値を持たせるウェブサイトの集合を定義しておき、それぞれのウェブサイトごとに、付加価値を提供するURLを登録しておく。そして、ウェブブラウザ内に既存のウェブページを表示し、そのウェブページについて別のフレームや別のウィンドウからDOMを操作することにより、付加価値を提供する。また、同じ付加価値を持たせるウェブサイトのウェブページへのリンクがある場合には、フレーム内にページをロードしたときに、そのウェブサイト上の付加価値を提供するURLに遷移するように、DOMを変更する。そして、パーソナライズのための情報は、同じ付加価値を持たせるウェブサイトの集合の中から代表してクッキーを管理するウェブサイトを1つだけ選んでおき、そのサイトを通してクッキーにアクセスすることを特徴としている。

10

【0012】

即ち、本発明は、第1のウェブサーバと、単一または複数からなる第2のウェブサーバとを備え、当該第1のウェブサーバおよび当該第2のウェブサーバにネットワークを介して接続されるクライアント装置のウェブブラウザにウェブページを表示可能なシステムであって、前記第1のウェブサーバは、既存のウェブページに対して付加される新たな価値である付加価値として、同じ付加価値を持たせるウェブサーバの集合として定義された当該第1のウェブサーバおよび当該第2のウェブサーバのドメインと、当該第1のウェブサーバおよび当該第2のウェブサーバのそれぞれに当該付加価値を提供するファイルのURL (Uniform Resource Locators)とのコード情報を記憶装置に登録し、前記第1のウェブサーバまたは前記第2のウェブサーバは、前記付加価値を提供するためのスクリプトプログラムを前記ウェブブラウザに送り、前記ウェブブラウザは、前記第1のウェブサーバまたは前記第2のウェブサーバから送られた既存のウェブページを表示し、前記スクリプトプログラムに含まれるAPI (Application Program Interface)を操作して当該既存のウェブページに前記付加価値を提供することを特徴としている。

20

【0013】

ここで、この第1のウェブサーバおよび第2のウェブサーバは、付加価値を持たせる他のウェブサーバのウェブページへのリンクがある場合に、フレーム内にウェブページをロードしたときに、この他のウェブサーバ上の付加価値を提供するURLに遷移するようにAPIを変更することを特徴とすることができる。また、この第1のウェブサーバおよび第2のウェブサーバの中から特定のウェブサーバが代表してクッキーを管理し、付加価値に関するパーソナライズのための情報は、この特定のウェブサーバを通してクッキーにアクセスすることを特徴とすることができる。尚、以下では、第1のウェブサーバを、このクッキーを管理する特定のウェブサーバとして選んだものとして説明している。

30

【0014】

他の観点から捉えると、本発明は、第1のウェブサーバと、単一または複数からなる第2のウェブサーバとを備え、当該第1のウェブサーバおよび当該第2のウェブサーバにネットワークを介して接続されるクライアント装置のウェブブラウザにウェブページを表示可能なシステムであって、前記第1のウェブサーバおよび前記第2のウェブサーバは、それぞれ固有のコンテンツが格納されると共に当該固有のコンテンツに付加価値を持たせるためにフレームを用いた文書ファイルを生成するための生成手段を備え、前記第1のウェブサーバは、既存のウェブページに対して付加される新たな価値である付加価値として、同じ付加価値を持たせるウェブサーバの集合として定義された当該第1のウェブサーバおよび前記第2のウェブサーバのドメインと、当該第1のウェブサーバおよび当該第2のウェブサーバのそれぞれに当該付加価値を提供するファイルのURLとのコード情報を記憶装置に登録し、前記第1のウェブサーバまたは前記第2のウェブサーバは、前記付加価値

40

50

を提供するためのスクリプトプログラムを前記ウェブブラウザに送り、前記ウェブブラウザは、前記第1のウェブサーバまたは前記第2のウェブサーバから送られた既存のウェブページを表示し、前記スクリプトプログラムに含まれるAPI(Application Program Interface)を操作して当該既存のウェブページに前記付加価値を提供することを特徴としている。

【0015】

ここで、この第1のウェブサーバは、ウェブブラウザからパラメータとして渡されたURLが、同じ付加価値を持たせるウェブサーバの集合として事前に登録されたドメインのものである場合に、クッキーを読むためのJavaScriptなどのスクリプトプログラムを含む文書ファイルをウェブブラウザに送信することを特徴としている。また、この第2のウェブサーバは、ウェブブラウザからパラメータとして渡されたクッキーの情報に基づいてスクリプトプログラムを含む文書ファイルをウェブブラウザに送信することを特徴としている。更に、この付加価値は、アクセシビリティを高めるものであることを特徴とすることができる。また、付加価値としては、翻訳や振り仮名表示、略語の展開なども考えられる。

【0016】

一方、本発明は、ネットワークを介して接続されるウェブブラウザに、既存のウェブページに対して付加される新たな価値である付加価値として同じ付加価値を提供する複数のウェブサーバの、主となるウェブサーバであって、装置固有のオリジナルコンテンツを格納する格納手段と、前記付加価値の提供のために、フレームを用いた文書ファイルを生成する生成手段と、前記複数のウェブサーバにて生成された文書ファイルから参照され、前記ウェブブラウザで前記付加価値を実行させるための文書ファイルを送信する送信手段と、前記付加価値を提供するウェブサーバの集合として定義された前記複数のウェブサーバのドメインと、当該複数のウェブサーバに当該付加価値を提供するスクリプトプログラムのURLとのコード情報を記憶装置に登録する手段とを含み、前記複数のウェブサーバが前記付加価値を提供するためのスクリプトプログラムをウェブブラウザに送った後、当該ウェブブラウザが当該複数のウェブサーバから送られた既存のウェブページを表示し、当該ウェブブラウザが当該スクリプトプログラムに含まれるAPI(Application Program Interface)を操作して当該既存のウェブページに前記付加価値を提供することを可能としている。

【0017】

ここで、この付加価値を提供するウェブサーバの集合として定義された複数のウェブサーバのドメインと付加価値を提供するスクリプトプログラムのURLを登録する手段を更に備えたことを特徴とすることができる。また、ブラウザからパラメータとして渡されたURLが事前に登録された複数のウェブサーバのドメインのものかどうかを判断し、事前に登録されたものである場合に、自らが提供するクッキーを読むためのスクリプトプログラムを含む文書ファイルをブラウザに送信する第2の送信手段を更に備えたことを特徴とすることができる。それに加えて、この複数のウェブサーバを構成する他のウェブサーバのウェブページを表示しているウェブブラウザからクッキーを書くスクリプトプログラムの要求を受信した場合に、このクッキーを書くスクリプトプログラムを含む文書ファイルをウェブブラウザに送信する第3の送信手段を更に備えたことを特徴とすることができる。

【0018】

本発明を方法のカテゴリから捉えると、本発明は、複数のウェブサーバにネットワークを介してウェブブラウザが接続されるシステムにて、既存のコンテンツに修正を加えることなく、既存のウェブページに対して付加される新たな価値である付加価値を提供するための方法であって、前記複数のウェブサーバを構成する特定のコンピュータ装置に、同じ付加価値を持たせるウェブサーバの集合として定義された第1のウェブサーバおよび第2

のウェブサーバのドメインと、当該第1のウェブサーバおよび当該第2のウェブサーバのそれぞれに当該付加価値を提供するファイルのURL (Uniform Resource Locators) とのコード情報を記憶装置に登録し、同じ付加価値を持たせる集合として定義された当該複数のウェブサーバのドメインと当該付加価値を提供するスクリプトプログラムのURL とのコード情報を予めメモリに格納し、前記ウェブブラウザからパラメータとして渡されたURL が、前記メモリに予め格納されたものであるか否かを当該メモリに格納された前記コード情報から確認し、予め当該メモリに格納されたものである場合に、クッキーの情報を読むためのスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信することを特徴とし、前記複数のウェブサーバが前記付加価値を提供するためのスクリプトプログラムをウェブブラウザに送った後、当該ウェブブラウザが当該複数のウェブサーバから送られた既存のウェブページを表示し、当該ウェブブラウザが当該スクリプトプログラムに含まれるAPI (Application Program Interface) を操作して当該既存のウェブページに前記付加価値を提供することを可能としている。

10

【0019】

ここで、この複数のウェブサーバの文書ファイルから参照され、アプリケーション固有の付加価値をウェブブラウザにて実行させることを特徴とすることができる。また、複数のウェブサーバを構成する他のウェブサーバのウェブページを表示しているウェブブラウザからクッキーを書くスクリプトプログラムの要求を受信した場合に、このクッキーを書くスクリプトプログラムを含む文書ファイルをウェブブラウザに送信することを特徴と

20

【0020】

更に本発明が適用されるプログラムは、複数のウェブサーバにネットワークを介してウェブブラウザが接続されるシステムにて、既存のコンテンツに修正を加えることなく、既存のウェブページに対して付加される新たな価値である付加価値を提供するための当該ウェブサーバとして機能するコンピュータに、同じ付加価値を持たせる集合として定義された前記複数のウェブサーバのドメインと当該付加価値を提供するスクリプトプログラムのURL とのコード情報を、前記コンピュータが備えるメモリに予め格納させる機能と、前記ウェブブラウザからパラメータとして渡されたURL が、前記メモリに予め格納されたものであるか否かを当該メモリに格納された前記コード情報から確認し、予め当該メモリに格納されたものである場合に、クッキーの情報を読むためのスクリプトプログラムを含む文書ファイルを当該ウェブブラウザに送信する機能とを実現させ、前記複数のウェブサーバが前記付加価値を提供するためのスクリプトプログラムをウェブブラウザに送った後、当該ウェブブラウザが当該複数のウェブサーバから送られた既存のウェブページを表示し、当該ウェブブラウザが当該スクリプトプログラムに含まれるAPI (Application Program Interface) を操作して当該既存のウェブページに前記付加価値を提供することを可能としている。

30

【0021】

また他の観点から捉えると、本発明が適用されるプログラムは、ウェブブラウザ内に既存のウェブページを表示し、他のフレームや他のウィンドウからDOMを操作することによってウェブページに付加価値を提供する機能と、この付加価値を持たせる他のウェブサーバのウェブページへのリンクがある場合に、フレーム内にウェブページをロードしたときに、他のウェブサーバ上の付加価値を提供するURL に遷移するようにDOMを操作することによりAタグの属性を変更する機能とを実現させる。

40

【0022】

尚、これらのプログラムをウェブサーバとして機能するコンピュータに対して提供する際に、例えばウェブサーバにインストールされた状態にて提供される場合の他、コンピュータに実行させるプログラムをコンピュータが読取可能に記憶した記憶媒体にて提供する形態が考えられる。この記憶媒体としては、例えば各種DVDやCD-ROM媒体等が該当し、各種DVDやCD-ROM読取装置等によってプログラムが読み取られ、フラッシ

50

メモリROM等にこのプログラムが格納されてCPUにて実行される。また、これらのプログラムは、例えば、プログラム伝送装置によってネットワークを介して提供される形態がある。

【発明の効果】

【0023】

本発明によれば、複数の既存のウェブサイトをほとんど変更することなく、アクセシビリティを高めるなどの新たな付加価値を提供することが可能となる。

【発明を実施するための最良の形態】

【0024】

以下、添付図面を参照して、本発明の実施の形態について詳細に説明する。

図1は、本実施の形態が適用されるネットワークシステムの全体構成を示した図である。図1に示すネットワークシステムは、ウェブ(Web)サーバにアクセスしてアクセシビリティなどの付加価値のあるウェブページの提供を受けるクライアント装置10と、付加価値のあるウェブページを提供する第1のウェブ(Web)サーバ20および第2のウェブ(Web)サーバ30とを備えている。そして、これらの各装置は、インターネットなどのWWW(World Wide Web)ベースのネットワーク40に接続されている。クライアント装置10の利用者は、ウェブブラウザ(後述)を用いて、HTTP(Hypertext Transfer Protocol)リクエストやHTTPS(Hypertext Transfer Protocol over SSL)リクエストを出力し、第1のウェブサーバ20や第2のウェブサーバ30に付加価値のあるウェブページを要求する。第1のウェブサーバ20や第2のウェブサーバ30は、HTTPレスポンス/HTTPSレスポンスによってクライアント装置10のウェブブラウザに付加価値のあるウェブページを送信する。プロトコルとしてHTTPを用いるかHTTPSを用いるかは、付加価値を付ける先の既存のウェブサイトが用いているプロトコルに合わせる。

【0025】

図2は、クライアント装置10、第1のウェブサーバ20、および第2のウェブサーバ30を実現するのに好適なコンピュータ装置のハードウェア構成例を模式的に示した図である。クライアント装置10、第1のウェブサーバ20、および第2のウェブサーバ30は、演算手段であるCPU(Central Processing Unit)101と、M/B(マザーボード)チップセット102およびCPUバスを介してCPU101に接続されたメインメモリ103と、同じくM/Bチップセット102およびAGP(Accelerated Graphics Port)を介してCPU101に接続されたビデオカード104とを備えている。また、PCI(Peripheral Component Interconnect)バスを介してM/Bチップセット102に接続されたハードディスクドライブ(HDD)106と、ネットワークインタフェース107とを備えている。更に、このPCIバスからブリッジ回路105およびISA(Industry Standard Architecture)バスなどの低速なバスを介してM/Bチップセット102に接続されたフレキシブルディスクドライブ108およびキーボード/マウス109を備える。

尚、図2は、本実施の形態を実現するコンピュータ装置のハードウェア構成を例示するに過ぎず、本実施の形態を適用可能であれば、他の種々の構成を取ることができる。例えば、ビデオカード104を設ける代わりに、ビデオメモリのみを搭載し、CPU101にてイメージデータを処理する構成としてもよいし、外部記憶装置として、ATA(AT Attachment)やSCSI(Small Computer System Interface)などのインタフェースを介してCD-R(Compact Disc Recordable)やDVD-RAM(Digital Versatile Disc Random Access Memory)などのドライブを設けてもよい。

【0026】

図3は、図1に示すシステムにて、図2に示すハードウェアを用いて実行される各装置の機能を示した図である。クライアント装置10は、メインメモリ103をワークエリアとしてCPU101にて実行されるソフトウェアであるウェブ(Web)ブラウザ11を備えており、第1のウェブサーバ20や第2のウェブサーバ30などのウェブサーバ上の情報に、ネットワークインタフェース107を用いネットワーク40を介してアクセスする。また、ユーザ情報やアクセス履歴などの情報をウェブブラウザ11とウェブサーバ(第1

10

20

30

40

50

のウェブサーバ20や第2のウェブサーバ30など)との間でやりとりするための仕組みとして、クッキー(Cookie)12を備えている。本実施の形態におけるクッキー12は、第1のウェブサーバ20が管理するヘッダ内容をクライアント装置10のローカルのHDD106にテキストファイルで保存している。

【0027】

第1のウェブサーバ20および第2のウェブサーバ30は、既存のウェブサイトにアクセシビリティなどの付加価値を持たせたものであり、ネットワーク40に接続されている。第1のウェブサーバ20は、例えば主となる単一のウェブサーバとして機能する。第2のウェブサーバ30は、例えば従となる単一または複数のウェブサーバで構成される。この第1のウェブサーバ20は、メインメモリ103をワークエリアとしてCPU101にて実行されるソフトウェアとして、HTTP/HTTPS処理部21と、付加価値用HTML生成部22と、個人情報処理部23と、ドメイン登録処理部24と、アプリケーション固有の付加価値の処理を行う付加価値処理部27とを備えている。また、HDD106には、オリジナルコンテンツ25と、登録ドメイン情報26とが格納される。これらのHDD106に格納される情報は、CPU101にて実行される各種ソフトウェアの指示に基づいて格納や読み出しが実行される。

同様に、第2のウェブサーバ30は、実行されるソフトウェアとして、HTTP/HTTPS処理部31と、付加価値用HTML生成部32と、個人情報処理部33とを備えている。また、HDD106には、オリジナルコンテンツ35が格納されている。

【0028】

HTTP/HTTPS処理部21,31は、クライアント装置10のウェブブラウザ11からのHTTPリクエスト/HTTPSリクエストを受信し、ウェブブラウザ11に対してHTTPレスポンス/HTTPSレスポンスを送信する。付加価値用HTML生成部22,32は、付加価値を持たせるためにフレームを用いたHTMLなどの文書ファイルを生成するための、ウェブサーバで実行されるプログラムであり、各々のウェブサーバ(第1のウェブサーバ20、第2のウェブサーバ30)で持つものである。第1のウェブサーバ20の個人情報処理部23は、第1のウェブサーバ20管理のクッキーを書くための各種処理を実行する。第2のウェブサーバ30の個人情報処理部33は、ウェブブラウザ11からパラメータとして渡されたクッキーの情報を通知するJavaScriptなどのスクリプトプログラムを含むHTMLを送信するための各種処理を実行する。第1のウェブサーバ20の登録ドメイン情報26には、同じ付加価値を持つウェブサーバのドメインとJSP(Java Server Pages)のURLを登録するJavaScriptのコード情報が格納されている。ドメイン登録処理部24では、この登録ドメイン情報26に格納されたコード情報を用いて、パラメータとして渡されたURLが事前に登録されたドメインのものであるかどうかの確認処理が実行される。オリジナルコンテンツ25,35には、各々のウェブサーバ固有のオリジナルのコンテンツが格納される。付加価値処理部27では、例えば、マウスの移動があったときにマウスのポインタが指すテキストを別フレームに拡大する、といったような付加価値の処理が実行される。この付加価値処理27は、第1のウェブサーバ20および第2のウェブサーバ30を含め、全てのウェブサーバが生成したHTMLから参照され、ウェブブラウザ11で実行されるプログラムで構成される。

【0029】

クライアント装置10の利用者は、ウェブブラウザ11を用いて、HTTPリクエスト/HTTPSリクエストによって第1のウェブサーバ20にアクセシビリティなどの付加価値のあるウェブページを要求する。第1のウェブサーバ20のHTTP/HTTPS処理部21は、HTTPレスポンス/HTTPSレスポンスによって、ウェブブラウザ11に付加価値のあるウェブページを送信する。付加価値のあるウェブページを受信したウェブブラウザ11では、単純にウェブページを表示するだけでなく、アクセシビリティのための機能などを実行することができる。付加価値としては、利用者がマウスで指し示す位置にある情報を別のフレームに拡大して表示することや音声で読み上げることが挙げられる。また、本実施の形態では、付加価値の機能をパーソナライズすることができる。例えば、キーボード

10

20

30

40

50

/マウス109のマウスで指し示す位置にある情報を別のフレームに拡大して表示するときの倍率を利用者の好みにより指定することができる。この情報は、第1のウェブサーバ20が管理するクッキー12により保管される。

【0030】

クライアント装置10の利用者は、ウェブページ上のリンクをクリックすることにより、別のウェブページに移動する。このとき、移動先が第1のウェブサーバ20上のウェブページであるか、第1のウェブサーバ20の登録ドメイン情報26に登録された同じ付加価値を持つウェブサーバ上のウェブページであれば、移動先でも同じ付加価値を利用することができる。例えば、利用者がウェブブラウザ11上で第2のウェブサーバ30上のウェブページへのリンクをクリックすると、HTTPリクエスト/HTTPSリクエストによって第2のウェブサーバ30にアクセシビリティなどの付加価値のあるウェブページを要求する。第1のウェブサーバ20と第2のウェブサーバ30には、同じ付加価値を持つウェブサーバであることをお互いに登録してある。第2のウェブサーバ30のHTTP/HTTPS処理部31は、HTTPレスポンス/HTTPSレスポンスによってウェブブラウザ11に付加価値のあるウェブページを送信する。

10

【0031】

図4は、ウェブブラウザ11の構成を示したブロック図である。図4に示す枠はフレームを持つHTMLを示しており、このフレームのHTMLに対してJavaScriptのコードが書かれている。ここでは、付加価値のための制御を行う制御部51と、手を加えていないウェブコンテンツが嵌め込まれて表示されるコンテンツ部52と、既存のウェブサイト上のウェブページに対して新たな価値を付加する付加価値部53と、個人情報データをクッキー12によりクライアントで保存する個人情報管理部54とを有している。

20

【0032】

制御部51では、コンテンツ部52にウェブページが読み込まれたときに、手を加えられていないコンテンツ部52のウェブページに対して必要な初期処理が行われる。この初期処理は、コンテンツ部52のためのFRAMEタグの中で設定されるonloadの中で行われる。このonloadは、リンクされた先のコンテンツの中身が読み込まれ、読み込まれ終わったことを教える。“W3C Recommendation 24 December 1999、HTML 4.01 Specification”〔2004年12月16日検索〕、インターネット<<http://www.w3.org/TR/HTML4/>>によると、FRAMEタグにonloadのイベントハンドラを設定できないことになっているが、Internet ExplorerでもMozillaでも、この方法により設定することができる。

30

この初期処理では、以下の3項目が実行される。

1．同じ付加価値を持つウェブサイト上のウェブページへのリンクがあるときには、付加価値を持たせたウェブページへのリンクに変更する。

2．コンテンツ部のスタイルを動的に変更させるためのCSS(Cascading Style Sheets)を追加する。

3．付加価値の機能を持たせるために、コンテンツ部内のイベントハンドラを設定する。

例えば、onmouseoverとonmouseoutのイベントハンドラを設定しておけば、HTMLのそれぞれのDOM(Document Object Model)要素の上に、マウスポインタが乗ったときとマウスカーソルが離れたときの処理を行うことができる。

40

【0033】

コンテンツ部52にて表示される既存のウェブサイト上のウェブページは、ウェブブラウザ11のフレームとして表示される。ウェブページが読み込まれたときに、制御部51によりイベントハンドラが設定される。ここで設定されたイベントハンドラは、必要に応じてDynamic HTMLを用いて、コンテンツ部52及び付加価値部53の表示を変化させる。イベントハンドラを記述するプログラミング言語としては、現在広く使われているウェブブラウザで動作することから、JavaScriptを用いることが適当である。しかしながら、本実施の形態は、イベントハンドラとしてJavaScriptの利用に限定されるものではない。将来、多くのウェブブラウザで動作する別のプログラミング言語が現れた場合には、そのプログラミング言語を用いることも可能である。

50

付加価値部 5 3 は、フレームや別ウィンドウ、あるいは両方として表示される。フレームや別ウィンドウの数は複数でも良い。単純な機能であれば、HTML と JavaScript のみで実装できる。より複雑な機能が必要であれば、Java アプレットなどを用いることになる。

個人情報管理部 5 4 では、個人情報がクッキーによりクライアントで保存され、例えばサイズを 0 に設定したフレームとして実現される。

【 0 0 3 4 】

次に、これらの各機能によって実行される処理の流れについて説明する。

図 5 は、第 2 のウェブサーバ 3 0 から送られたウェブページから第 1 のウェブサーバ 2 0 管理のクッキーを読む方法を示した図である。尚、この図 5 および後述する図 6 では、クッキーを管理する特定のウェブサーバを第 1 のウェブサーバ 2 0、それ以外のウェブサーバを第 2 のウェブサーバ 3 0 として取り扱っている。ウェブブラウザ 1 1 の付加価値部 5 3 は、第 1 のウェブサーバ 2 0 に対する HTTP リクエスト / HTTPS リクエストによって、第 1 のウェブサーバ 2 0 に対してクッキーを読むスクリプトプログラムを含む HTML を要求する (ステップ 1 0 1)。このとき、ターゲットとして個人情報管理部 5 4 のフレームを指定する。また、パラメータとして、第 2 のウェブサーバ 3 0 に対する個人情報管理部 5 4 からの HTTP リクエスト / HTTPS リクエストの宛て先となる URL が渡される。

【 0 0 3 5 】

HTTP リクエスト / HTTPS リクエストを受けた第 1 のウェブサーバ 2 0 のドメイン登録処理部 2 4 は、パラメータとして渡された URL が事前に登録されたドメインのものであるかを、HDD 1 0 6 に格納される登録ドメイン情報 2 6 を用いて確認する (ステップ 1 0 2)。登録されているものであれば、HTTP レスポンス / HTTPS レスポンスによってスクリプトプログラムを含む HTML を送信する (ステップ 1 0 3)。そのスクリプトプログラムは、クッキーを読み、その値をパラメータとして渡された URL に送るということを実行するコードが記述されている。登録されたドメインのものではない場合には、空の HTML を送信する (ステップ 1 0 4)。

【 0 0 3 6 】

個人情報管理部 5 4 では、送られたスクリプトプログラムを実行することによりクッキーを読む (ステップ 1 0 5)。そして、第 2 のウェブサーバ 3 0 に対して、読んだクッキーの情報を付加価値部 5 3 に通知するスクリプトプログラムを要求する (ステップ 1 0 6)。このとき、ターゲットとして個人情報管理部 5 4 のフレームを指定する。また、パラメータとして、読んだクッキーの情報が渡される。

【 0 0 3 7 】

この HTTP リクエスト / HTTPS リクエストを受けた第 2 のウェブサーバ 3 0 は、パラメータとして渡されたクッキーの情報を付加価値部 5 3 に通知するスクリプトプログラムを含む HTML を、HTTP レスポンス / HTTPS レスポンスによって送信する (ステップ 1 0 7)。そのスクリプトプログラムは、クッキーの情報を引数として、付加価値部 5 3 のスクリプトプログラムの関数を呼び出すというコードが記述されている。

【 0 0 3 8 】

ここで、個人情報管理部 5 4 が第 2 のウェブサーバ 3 0 からの HTTP レスポンス / HTTPS レスポンスを受け取ると、個人情報管理部 5 4 のフレームと付加価値部 5 3 のフレームの両方に第 2 のウェブサーバ 3 0 から送られたウェブページがあることになる。そのため、個人情報管理部 5 4 は、クッキーの情報をパラメータとして、付加価値部 5 3 のスクリプトプログラムを呼び出し (ステップ 1 0 8)、第 1 のウェブサーバ 2 0 管理のクッキーを付加価値部 5 3 に知らせることができる。付加価値部 5 3 では、得られたクッキーの情報をを用いて、アプリケーション固有の処理を実行して (ステップ 1 0 9)、処理が終了する。

【 0 0 3 9 】

尚、サーバで実行されるプログラムは、JSP の他に、Servlet や CGI などの任意の技術を用いることができる。また、ウェブブラウザ 1 1 で実行されるプログラムは、JavaScript が代表的なものであるが、多くのウェブブラウザで動作する別のプログラミング言

10

20

30

40

50

語が現れた場合には、そのプログラミング言語を用いても構わない。更に、同じ付加価値を持たせるウェブサイトの集合を変更する場合には、集合に追加あるいは削除されるウェブサイトと主となる単一のウェブサーバのみを変更すれば良い。この処理は、図3に示すドメイン登録処理部24によって実行され、その結果は登録ドメイン情報26に格納される。

【0040】

図6は、第2のウェブサーバ30から送られたウェブページから第1のウェブサーバ20管理のクッキーを書く方法を示した図である。付加価値部53は、第1のウェブサーバ20に対するHTTPリクエスト/HTTPSリクエストによって、第1のウェブサーバ20に対してクッキーを書くスクリプトプログラムを含むHTMLを要求する(ステップ201)。このとき、ターゲットとして個人情報管理部54のフレームを指定する。また、パラメータとしてクッキーに書く値を渡す。第1のウェブサーバ20は、HTTPレスポンス/HTTPSレスポンスによって、個人情報管理部54に対してクッキーを書くスクリプトプログラムを含むHTMLを送信する(ステップ202)。個人情報管理部54は、クッキーを書いても良いか否かを利用者に確認する(ステップ203)。確認が得られない場合には、クッキーを保存せずにクッキーを書くためのフローは終了し、次の処理へ移行する。確認が得られた場合には、クッキーを書き(ステップ204)、次の処理へ移行する。尚、クッキーを書くとき、悪意のあるウェブサイトから不正なHTTPリクエスト/HTTPSリクエストが第1のウェブサーバ20に対して送られた場合には、クッキーを書かないようにしなければならない。そのためには、第1のウェブサーバ20がHTTPリクエストのrefererフィールドを参照し、そのURLが事前に登録されたドメインのものであるかを確認し、登録されたものであればクッキーを書くという方法が考えられる。ただし、ウェブブラウザのバグなどでrefererフィールドを偽装されるという事態や、利用者がrefererフィールドを送信しないようにウェブブラウザを設定するという事象も考えられる。そこで、JavaScriptの中でクッキーを書く前にダイアログボックスを表示し、利用者にクッキーを保存しても良いかを確認する方法が確実である。尚、文字の拡大率などの、それほど重要でない情報をクッキーに書くときには、悪意のあるサイトにクッキーを書き換えられても影響が少ない。そこで、クッキーを書き換えるかどうかを利用者に確認する作業を省略することもできる。また、個人情報などの重要な情報をクッキーで扱うときには、図6に示す方法でクッキーを書くことはせず、特定のウェブサイトからだけクッキーを書くようにすることも望ましい。

【0041】

次に、具体的な実施例について説明する。

図7は、ウェブブラウザ11に、文字の拡大の付加価値のあるウェブサイト上のウェブページを表示した例を示した図である。図7に示す例では、目に見える3つのフレーム(第1のフレーム61、第2のフレーム62、および第3のフレーム63)と、見えない第4のフレーム64との4つのフレームを備えている。第1のフレーム61は拡大率の変更などを行うフレームであり、付加価値部53で用いられる。図示するような付加価値のためのボタン(拡大、縮小)などが配置される。第2のフレーム62はコンテンツの表示を行うフレームであり、コンテンツ部52で用いられる。ここでは、オリジナルのコンテンツが表示されている。第3のフレーム63は付加価値が施された画像を表示するフレームであり、付加価値部53で用いられる。ここでは、拡大された文字列が表示されている。更に、大きさを0にしているため見えないが、第4のフレーム64を備えており、この第4のフレーム64は個人情報管理部54にて用いられる。図7の例では、第2のフレーム62の所定の文字列にマウスがあるときに、その下のDOM要素を反転し、第3のフレーム63に拡大表示している。

【0042】

図8は、図7に示す例のJSPのコードを示した図である。図8に示す例では、hrefというパラメータを用いてHTMLを生成している。JSPは、一般に、HTMLファイルの内部にスクリプトで処理を記述し、ウェブブラウザからのリクエストのたびにスクリプトをサーバ側で実行し、HTML中のスクリプト部分をその処理結果で置き換えてウェブ

10

20

30

40

50

ブラウザに送り出す機構である。この図 8 に示されるような J S P は、主となる単一のウェブサーバである第 1 のウェブサーバ 2 0 と、従となる単一または複数のウェブサーバである第 2 のウェブサーバ 3 0 とに備えられる。図 8 では、コンテンツ部のFRAMEタグにonloadのイベントハンドラが設定されている。このイベントハンドラの引数としてFRAMEの名前が渡されている。コンテンツの指定には、hrefというパラメータでURLを渡すことがsrc属性に示されている。参照するスクリプトプログラム(JavaScript)は、それぞれのウェブサイトごとに置く必要はない。7行目の構成要素71にて参照されるJavaScriptは、例えば第1のウェブサーバ20などの単一のウェブサーバ(特定のウェブサイト)に置かれ、第1のウェブサーバ20や第2のウェブサーバの中から何れか一つのサーバを選べば良い。この構成要素71には、URLを登録するJavaScriptのコードのありかが示されている。そして、11行目に示す部分72が置き換えられる。特定のウェブサイトは、元のテンプレートは同じであるが、特定のサーバの記述に変更することで、この特定のサーバのJavaScriptを利用することができる。

【0043】

ここで、例えばフレームのページが第1のウェブサーバ20から来たもので、中身だけが第2のウェブサーバ30のコンテンツであるとすると、通常の方法であっては、その中身の情報を取得することができない。即ち、通常のセキュリティモデルによっては、ウェブサーバが異なる場合に、例えばマウスでポイントした箇所についてどのような文字列が書かれているか、ということを外のフレームから取得することができない。しかしながら、図8の例に示すような本実施の形態によれば、第1のウェブサーバ20が提供するウェブページから第2のウェブサーバ30にジャンプしたときには、第2のウェブサーバ30から全体をロードし直すことで、第2のウェブサーバ30のJavaScriptから第2のウェブサーバ30のコンテンツにアクセスするように切り換えることが可能となる。このように本実施の形態では、異なるドメイン間をまたがるジャンプの場合には、全体をリロードすることで、ウェブページを提供するドメインのJavaScriptによってコンテンツの中に書かれている文字列の情報を取得することができる。これによって、全体のズームとか全体の色の置き換えだけではなく、ポイントした箇所だけを拡大して読み上げる等のアクセシビリティを実現可能としている。

【0044】

図9は、同じ付加価値を持つウェブサーバのドメインとJ S PのURLを登録するJavaScriptのコードの例を示した図であり、図8の7行目で指定したJavaScriptファイルの内容の一部である。この図9に示すJavaScriptのコードは、図3に示す第1のウェブサーバ20の登録ドメイン情報26に格納されている。この図9に示す例では、

```
w3.trl.ibm.com
w3.watson.ibm.com
w3.almaden.ibm.com
```

が同じ付加価値を持つウェブサーバのドメインであり、これらのドメイン内のウェブページに付加価値を伴ってアクセスするには、それぞれ、

```
http://w3.trl.ibm.com/RakurakuWeb/rakuraku.jsp
http://w3.watson.ibm.com/RakurakuWeb/rakuraku.jsp
http://w3.almaden.ibm.com/RakurakuWeb/rakuraku.jsp
```

というJ S Pを介してアクセスするというを示している。例えば、

```
http://w3.trl.ibm.com/
```

に付加価値を伴ってアクセスするには

```
http://w3.trl.ibm.com/RakurakuWeb/rakuraku.jsp?href=http://w3.trl.ibm.com/
```

と指定する。

【0045】

図10は、コンテンツ部のFRAMEタグに設定されたonloadのイベントハンドラのJavaScriptのコードを示した図である。現在のドメイン以外へ移動するAタグのうちで、同じ付加価値を持つウェブサーバのドメイン上のウェブページに移動するものは、付加価値を持

たせるためにhref属性を変更し、また、target属性も_topに変更しておく。更に、コンテンツ部のDOM要素を反転するために、CSSもこのコンテンツに追加する。更に、onmouseoverとonmouseoutのイベントハンドラを設定する。

onmouseoverのコードの詳細は省略するが、イベントのソースのDOM要素を得て、そのDOM要素にクラスを設定してCSSを用いて反転表示し、また、下のフレームに拡大した文字を表示する。

onmouseoutのコードの詳細も省略するが、イベントのソースのDOM要素を得て、そのDOM要素に反転表示のために設定したクラスの設定を解除し、また、下のフレームに拡大して表示した文字の表示を取り消す。このように図10の3行目から9行目に示す構成によって、同じ付加価値を持たせるウェブサイトのウェブページへのリンクがある場合には、フレーム内にページをロードしたときに、そのウェブサイト上の付加価値を提供するURLに遷移するようにDOMが変更される。即ち、図8に示すcontentsFrameに新しいウェブページがロードされると、図10のinitが実行される。そして、このウェブページの中の全てのAタグを調べ、現在のページのホストとは別のページへのリンクだが、同じ付加価値を持たせるウェブサイトとして定義されているホスト上のページへのリンクについては、そのウェブサイト上の付加価値を提供するURLに遷移するように、DOMを操作することにより、Aタグの属性を変更する。

尚、図10には、必要に応じてテキストを分割するための構成要素81として、splitText()が追加されている。

【0046】

図11は、図10に示すHTML文書の元となった処理の流れを示すフローチャートである。ここでは、まず、現在のドメイン以外へ移動するAタグの内、同じ付加価値を持つウェブサーバのドメイン上へ移動するものは、付加価値を持たせるためにhref属性を変更し、target属性も_topに変更する(ステップ301)。そして、CSSを追加し(ステップ302)、必要に応じてテキストを分割する(ステップ303)。そして、onmouseoverとonmouseoutのイベントハンドラを設定して(ステップ304)、処理が終了する。

【0047】

図12(a),(b)は、ステップ303の、必要に応じてテキストを分割する場合の具体例を示した図である。オリジナルのHTMLよりも細かい単位で、マウスポインタの位置のテキストを検知するために、必要に応じてSPANタグを挿入する。このSPANタグの目的により、適切なclass属性も付けておく。どのような単位でSPANタグを挿入するのかということ、どのようなclass属性を用いるかということは、アプリケーションに依存する。図12に示す例では、文ごとにSPANタグに分割し、更に漢字は、文のSPANタグの子のSPANタグとしている。即ち、図12(a)に示すように、元の文章の「こんにちは。私の名前は、太郎です。」について、図12(b)に示すように、まず、「こんにちは。」と「私の名前は、太郎です。」2つのセンテンスに分割される。2つ目のセンテンスである「私の名前は、太郎です。」は、漢字とひらがなとに分け、漢字である「私」、「名前」、「太郎」は、文のSPANタグの子のSPANタグとされる。

【0048】

図13は、イベントのソースを得る関数のJavaScriptのコードを示した図である。onmouseoverとonmouseoutのイベントハンドラでは、処理を行うためにイベントのソースを得る必要があるが、ウェブブラウザごとにイベントのソースを得る方法が違うので、ここに示したようなコードが必要となる。図13に示すJavaScriptのコードでは、まず、myWindow.eventを使えるか否かの判断がなされる。使える場合には、myWindow.event.srcElementをイベントソースとする。使えない場合には、e.Targetをイベントソースとし、イベントのソースを返して処理が終了する。マウスポインタの下のHTML要素を取り出す際に、図12に示すようにSPANタグで分割しておけば、オリジナルのHTMLよりも細かい単位でHTML要素を取り出すことが可能となる。

【0049】

尚、上述した実施の形態では、付加価値としてアクセシビリティを例に挙げたが、付加

10

20

30

40

50

価値として、例えば、個人情報の入力補助にも利用できる。例えば、商取引のサイトで商品を購入しようとする、氏名、住所、電話番号などの情報の入力を求められる。これらの情報について、商品を購入する度に入力することは非常に煩わしい。しかしながら、本実施の形態を応用すると、クライアントにクッキーとしてこれらの個人情報を保管しておき、フォーム内の項目を自動的に入力することができる。これらの個人情報の入力補助は、事前に登録してある複数のウェブサイトで共有することが可能である。例えば、個人情報を登録するには、利用者が登録ページを表示し必要な項目を入力し、登録ボタンを押すとクッキーに書き出される。この登録された個人情報を利用するウェブサイトでは、個人情報を入力すべきページを表示したときに、図5の方法によりクッキーを読み取る。そして、個人情報管理部54から付加価値部53に対して実行されるステップ108のJavaScript呼び出しの中で、個人情報を入力すべきフォームの要素に対して呼び出したクッキーの値を代入すれば良い。

10

【0050】

また、他の応用例として、本実施の形態に示したものと同様な仕組みにより、文字の拡大の代わりに翻訳を行うように構成することもできる。文は、図10の構成要素81に示すsplitText()の処理で、図12に示すタグの中に入れておく。これによって、オリジナルのHTMLの構造単位だけではなく、アプリケーションにて必要な単位で翻訳を行うことが可能となる。

更に、同様な仕組みによって、文字の拡大の代わりに振り仮名の表示を行うことへの応用も可能である。漢字は、splitText()の処理で、図12に示すタグの中に入れておく。これによって、漢字だけを取り出して振り仮名表示をすることが可能となる。

20

また更に、同様な仕組みによって、文字の拡大の代わりに略語の展開を行うこともできる。略語は、splitText()の処理でタグの中に入れておく。

【0051】

以上、詳述したように、本実施の形態によれば、既存ウェブサイトをほとんど変更することなく、アクセシビリティを高めることや、上述した各種応用例などの新たな価値を追加(付加価値を提供)することができる。また、クライアントの環境が特定のOSやウェブブラウザに依存しないようにできる。更に、関連のある複数のウェブサイト間でページを遷移したときも利用者に負担をかけることなく、新たに追加した価値を利用可能とすることができる。また更に、関連のある複数のウェブサイト間でパーソナライズのための情報をサーバに保管することなく共有することも可能となる。

30

【図面の簡単な説明】

【0052】

【図1】本実施の形態が適用されるネットワークシステムの全体構成を示した図である。

【図2】クライアント装置、第1のウェブサーバ、および第2のウェブサーバを実現するのに好適なコンピュータ装置のハードウェア構成例を模式的に示した図である。

【図3】図1に示すシステムにて、図2に示すハードウェアを用いて実行される各装置の機能を示した図である。

40

【図4】ウェブブラウザの構成を示したブロック図である。

【図5】第2のウェブサーバから送られたウェブページから第1のウェブサーバ管理のクッキーを読む方法を示した図である。

【図6】第2のウェブサーバから送られたウェブページから第1のウェブサーバ管理のクッキーを書く方法を示した図である。

【図7】ウェブブラウザに、文字の拡大の付加価値のあるウェブサイト上のウェブページを表示した例を示した図である。

【図8】図7に示す例のJSPのコードを示した図である。

【図9】同じ付加価値を持つウェブサーバのドメインとJSPのURLを登録するJavaScriptのコードの例を示した図である。

50

【図10】コンテンツ部のFRAMEタグに設定されたonloadのイベントハンドラのJavaScriptのコードを示した図である。

【図11】図10に示すHTML文書の元となった処理の流れを示すフローチャートである。

【図12】(a),(b)は、ステップ303の、必要に応じてテキストを分割する場合の具体例を示した図である。

【図13】イベントのソースを得る関数のJavaScriptのコードを示した図である。

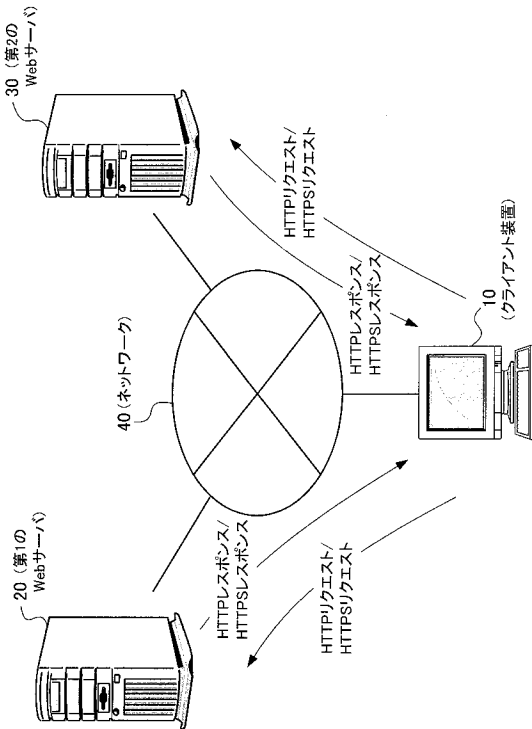
【符号の説明】

【0053】

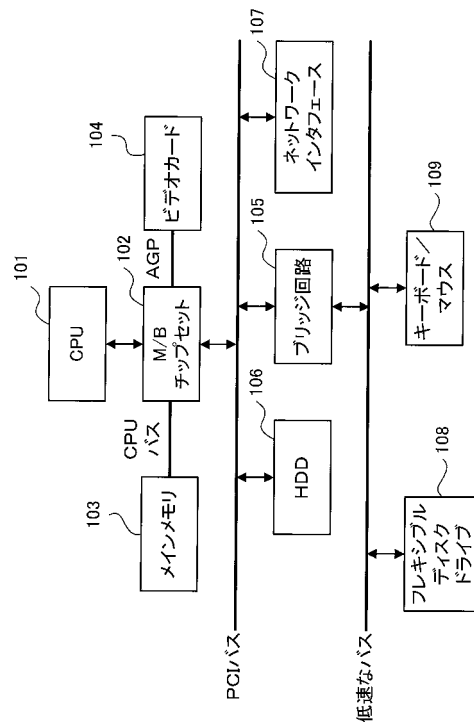
10 ... クライアント装置、 11 ... ウェブ(Web)ブラウザ、 12 ... クッキー(Cookie)、 20 ... 第1のウェブ(Web)サーバ、 21 ... HTTP/HTTPS 処理部、 22 ... 付加価値用HTML生成部、 23 ... 個人情報処理部、 24 ... ドメイン登録処理部、 25 ... オリジナルコンテンツ、 26 ... 登録ドメイン情報、 27 ... 付加価値処理部、 30 ... 第2のウェブ(Web)サーバ、 31 ... HTTP/HTTPS 処理部、 32 ... 付加価値用HTML生成部、 33 ... 個人情報処理部、 35 ... オリジナルコンテンツ、 40 ... ネットワーク、 51 ... 制御部、 52 ... コンテンツ部、 53 ... 付加価値部、 54 ... 個人情報管理部

10

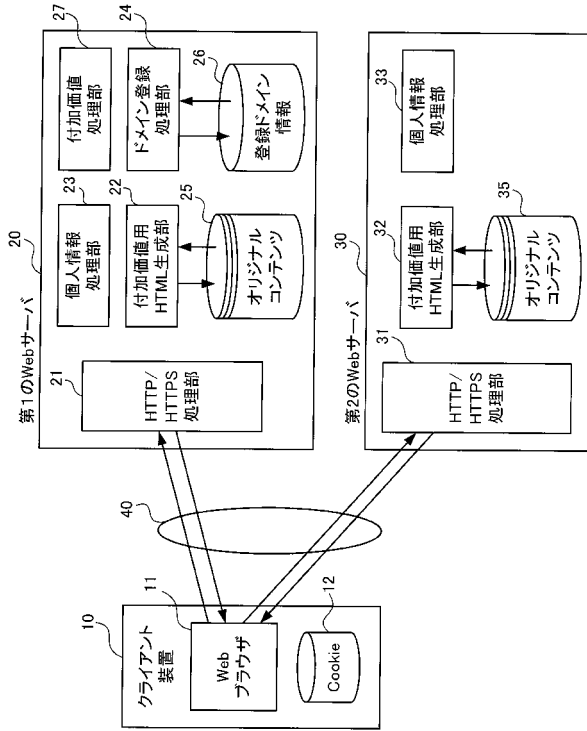
【図1】



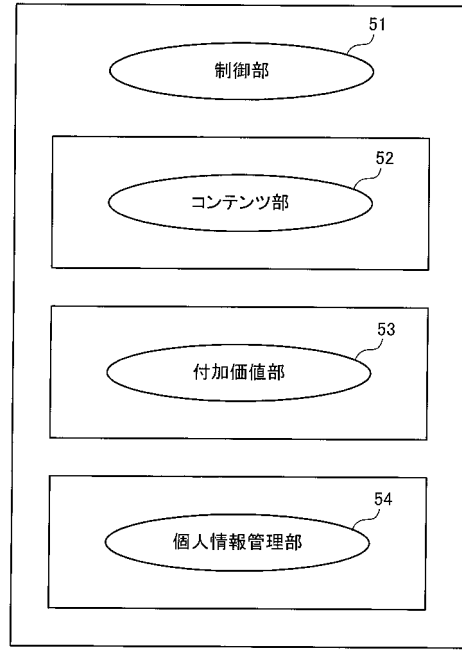
【図2】



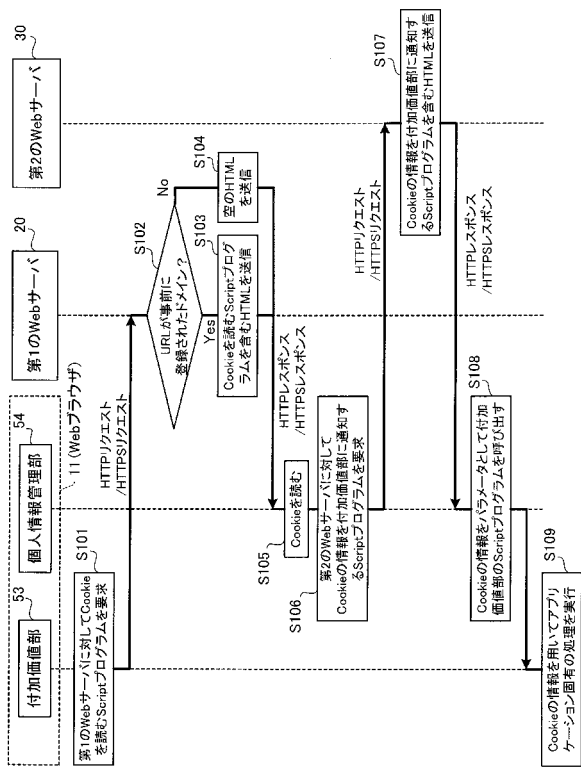
【図3】



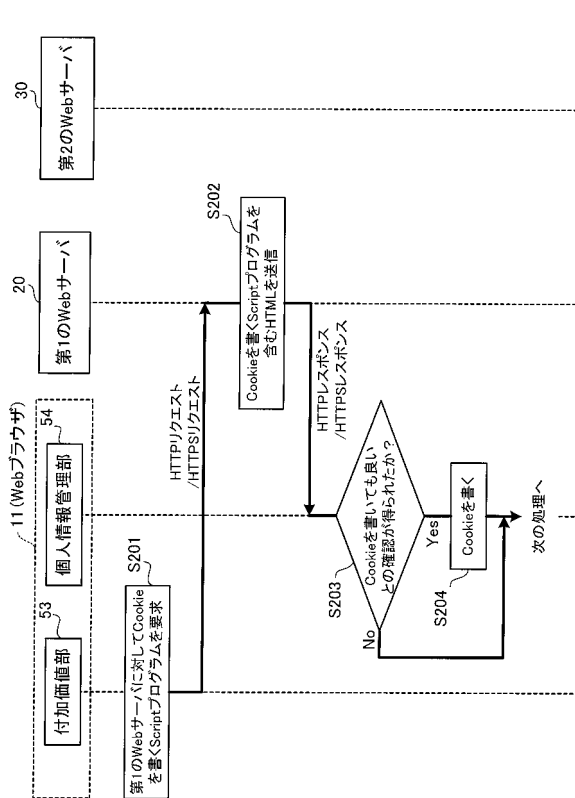
【図4】



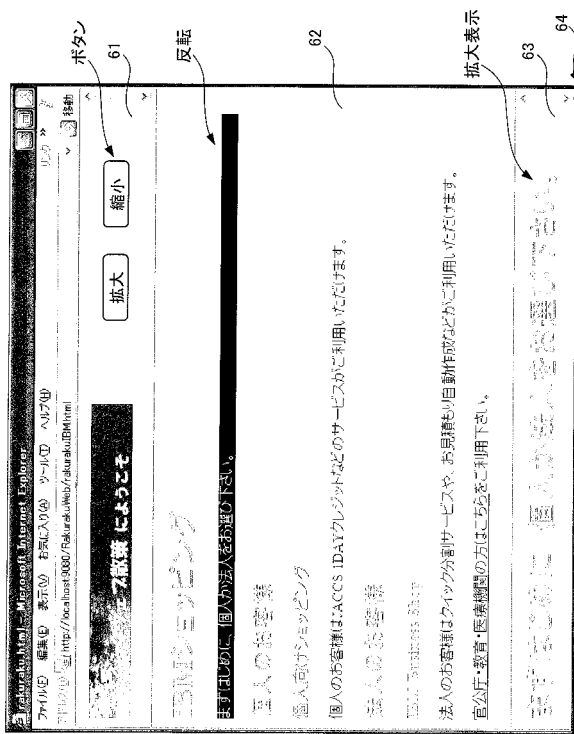
【図5】



【図6】



【図 7】



【図 9】

```

1  var RAKURAKU_JSPTS = new Object();
2  RAKURAKU_JSPTS["w3.tr.ibm.com"] = "http://w3.tr.ibm.com/RakurakuWeb/rakuraku.jsp";
3  RAKURAKU_JSPTS["w3.watson.ibm.com"] = "http://w3.watson.ibm.com/RakurakuWeb/rakuraku.jsp";
4  RAKURAKU_JSPTS["w3.almaden.ibm.com"] = "http://w3.almaden.ibm.com/RakurakuWeb/rakuraku.jsp";

```

【図 8】

```

1  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
2  <HTML>
3  <HEAD>
4  <meta http-equiv="Content-Type" content="text/html; charset=SHIFT_JIS" pageEncoding="SHIFT_JIS" >
5  <TITLE>Rakuraku.jsp</TITLE>
6  <SCRIPT type="text/javascript" src="http://w3.tr.ibm.com/RakurakuWeb/rakuraku.js"></SCRIPT>
7  </HEAD>
8  <FRAMESET rows="20% * 30% 0%" >
9  <FRAME src="controller.html" name="controllerFrame" onload="init(controllerFrame)" >
10 <FRAME src="request.getParameter("href").%" name="contentsFrame" onload="init(contentsFrame)" >
11 <FRAME src="magnifier.html" name="magnifierFrame" >
12 <FRAME name="personalizedDataFrame" >
13 </FRAMESET>
14 <BODY>
15 <P>This page uses frames. The current browser you are using does not support frames.</P>
16 </BODY>
17 </NOFRAMES>
18 </FRAMESET>
19 </HTML>
20

```

71

72

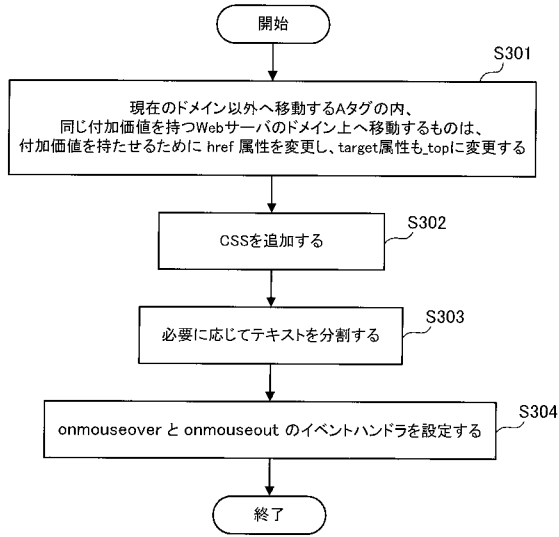
【図 10】

```

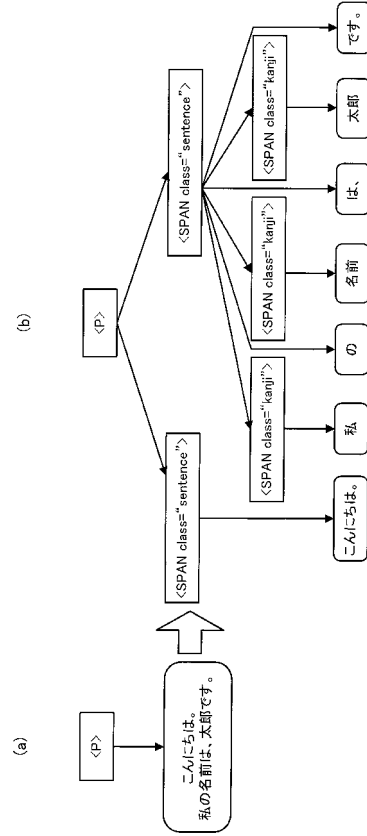
1  function init(myWindow) {
2  var anchors;
3  anchors = myWindow.document.getElementsByTagName("a");
4  for (var i = 0; i < anchors.length; i++) {
5  if (myWindow.location.hostname == anchors[i].hostname) && (anchors[i].hostname in RAKURAKU_JSPTS) {
6  anchors[i].href = RAKURAKU_JSPTS[anchors[i].hostname] + "?href=" + anchors[i].href;
7  anchors[i].target = "_top";
8  }
9  }
10 appendRakurakuCss();
11 splitText();
12 myWindow.document.onmouseover = function(e) {
13 81
14  var src;
15  src = getEventSource(myWindow, e);
16  ...
17 myWindow.document.onmouseout = function(e) {
18  var src;
19  src = getEventSource(myWindow, e);
20  ...
21  }
22 }

```

【図 1 1】



【図 1 2】



【図 1 3】

```

function getSource(myWindow, e) {
  var src;
  if (typeof(myWindow.event) != "undefined") { // Internet Explorer
  } else {
    src = myWindow.event.srcElement;
  }
  return src;
}

```

フロントページの続き

- (72)発明者 坂入 隆
神奈川県大和市下鶴間1623番地14 日本アイ・ピー・エム株式会社 東京基礎研究所内
- (72)発明者 大湖 卓也
神奈川県大和市下鶴間1623番地14 日本アイ・ピー・エム株式会社 東京基礎研究所内
- (72)発明者 牟田 英正
神奈川県大和市下鶴間1623番地14 日本アイ・ピー・エム株式会社 東京基礎研究所内

審査官 佐藤 智康

- (56)参考文献 特開2004-185541(JP,A)
特開2004-318816(JP,A)

- (58)調査した分野(Int.Cl., DB名)
G06Q 10/00-50/00
G06F 15/00