(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4950439号
(P4950439)
(45)発行日　平成24年6月13日(2012.6.13)　(24)登録日　平成24年3月16日(2012.3.16)

(51)Int.Cl.　　F I
G06F 13/00 (2006.01)　　G06F 13/00 520D
G06F 15/00 (2006.01)　　G06F 15/00 440A

請求項の数 30　(全 24 頁)

(21)出願番号　特願2005-139956 (P2005-139956)
(22)出願日　平成17年5月12日 (2005.5.12)
(65)公開番号　特開2005-327291 (P2005-327291A)
(43)公開日　平成17年11月24日 (2005.11.24)
審査請求日　平成20年5月9日 (2008.5.9)
(31)優先権主張番号　10/845,625
(32)優先日　平成16年5月14日 (2004.5.14)
(33)優先権主張国　米国 (US)

(73)特許権者　500046438
マイクロソフト　コーポレーション
アメリカ合衆国　ワシントン州　９８０５２－６３９９　レッドモンド　ワン　マイクロソフト　ウェイ
(74)代理人　100077481
弁理士　谷　義一
(74)代理人　100088915
弁理士　阿部　和夫
(72)発明者　チャ　ジャン
アメリカ合衆国　９８０５２　ワシントン州　レッドモンド　ワン　マイクロソフト　ウェイ　マイクロソフト　コーポレーション内

最終頁に続く

(54)【発明の名称】部分的複製を使用する、Ｗｅｂコンテンツの分散型ホスティング

(57)【特許請求の範囲】
【請求項１】
ホスト・コンピューティング・デバイスおよび１または複数のピア・コンピューティング・デバイスを含むコンピュータ・クラスタ内の複数のコンピューティング・デバイスに、Ｗｅｂサイトのコンテンツを分配する方法であって、前記コンテンツは、複数のファイルを含み、前記方法は、
前記複数のファイルのそれぞれについて、前記Ｗｅｂサイトのディレクトリ間の相対的重要度を表すサイト重み（Ｓ）に、コンテンツのタイプに応じた重要度を表すタイプ重み（Ｔ）を乗算したファイル重み（ＦＷ）を計算すること、
前記複数のファイルのそれぞれの複製係数に基づいて決定されている前記コンテンツの複数の部分集合を作成することであって、前記複製係数は、前記部分集合の１つを格納するコンピュータ・デバイスの記憶容量に基づく複製量（Ｄ）に基づいて、下式により決定されること、
【数１】

$$D(\lambda_i) = \sum_m |Pm| \max\{1, FW_m \times \lambda_i\}$$

ここで、Ｄ（λｉ）はピア・コンピューティング・デバイスに複製される総量であり、λｉはピア・コンピューティング・デバイスについての複製係数であり、ＦＷｍは前記Ｗｅｂサイトの内のｍ番目のファイルのファイル重みであり、ＰｍはＷｅｂサイトの内のｍ番目のファイルのファイルサイズであること、および

前記複数の部分集合の１つを前記複数のコンピューティング・デバイスのそれぞれに分配すること
を備えることを特徴とする方法。
【請求項２】
前記コンピュータ・クラスタは、サーバ・アレイを備えることを特徴とする請求項１に記載の方法。
【請求項３】
前記コンピュータ・クラスタは、ピア・ツー・ピア・ネットワークを備えることを特徴とする請求項１に記載の方法。
【請求項４】
前記タイプ重みは、ファイル拡張子に基づくことを特徴とする請求項１に記載の方法。
【請求項５】
前記サイト重みは、前記Ｗｅｂサイトのディレクトリに割り当てられ、前記ディレクトリ内のそれぞれのファイルは、前記ディレクトリの前記サイト重みを持つことを特徴とする請求項１に記載の方法。
【請求項６】
前記ディレクトリのサブ・ディレクトリは、前記ディレクトリに割り当てられている前記サイト重みを継承することを特徴とする請求項５に記載の方法。
【請求項７】
前記複数のファイルは、複数のメッセージに分割され、それぞれの部分集合は、少なくとも１つのメッセージを備えることを特徴とする請求項１に記載の方法。
【請求項８】
前記複数のメッセージの大部分は、固定サイズであることを特徴とする請求項７に記載の方法。
【請求項９】
さらに、前記複数のファイルを複数のオリジナル・メッセージに分割すること、
イレージャ・コーディング方式を適用して複数の符号化メッセージを派生させること、および
前記符号化メッセージのうちの１つまたは複数をそれぞれの部分集合に追加すること
を備えることを特徴とする請求項１に記載の方法。
【請求項１０】
前記イレージャ・コーディング方式は、リード・ソロモン・コーデックを備えることを特徴とする請求項９に記載の方法。
【請求項１１】
前記複数の部分集合のうちの１つを分配することは、前記イレージャ・コーディング方式に関連付けられているキーに基づき前記部分集合を分配することを備えることを特徴とする請求項９に記載の方法。
【請求項１２】
請求項１ないし１１のいずれかに記載の方法を実行するためのコンピュータ実行可能命令を備えることを特徴とするコンピュータ可読記録媒体。
【請求項１３】
ホスト・コンピューティング・デバイスおよび１または複数のピア・コンピューティング・デバイスを含むコンピュータ・クラスタ内の複数のコンピューティング・デバイスから、Ｗｅｂサイトのコンテンツを取り出す方法であって、
前記コンテンツは、複数のファイルを含み、
前記複数のファイルのそれぞれに、前記Ｗｅｂサイトのディレクトリィ間の相対的重要度を表すサイト重み（Ｓ）に、コンテンツのタイプに応じた重要度を表すタイプ重み（Ｔ）を乗算したファイル重み（ＦＷ）が割り当てられ、
前記複数のファイルは、下式により決定される複製比に基づいて前記複数のコンピューティング・デバイスに分配されており、

$w_{m,i}=max\{1, FW_m\times\lambda_i\}$
ここで、$w_{m,i}$はピアｉに対するｍ番目のファイルの複製比であり、λｉはピアｉについて計算された複製係数であり、ＦＷｍは前記Ｗｅｂサイトの内のｍ番目のファイルのファイル重みであり、前記方法は、
前記ＷｅｂサイトのコンテンツのＵＲＬを受け取ること、
前記コンテンツの部分集合であるファイルを格納しているコンピューティング・デバイスであって、前記コンテンツを取得することになる複数のコンピューティング・デバイスを決定すること、および
前記複数のコンピューティング・デバイスとのネゴシエーションにより、前記コンテンツに含まれる前記複数のファイルを取得して、前記Ｗｅｂサイトの前記コンテンツを再作成すること
を備えたことを特徴とする方法。
【請求項１４】
前記ＵＲＬは、前記ＵＲＬが前記分散型Ｗｅｂサイトに関連付けられていることを示すタグを含むことを特徴とする請求項１３に記載の方法。
【請求項１５】
前記複数のコンピューティング・デバイスを決定することは、前記ＵＲＬを大きな整数値に変換し、前記複数のコンピューティング・デバイスを識別することを備えることを特徴とする請求項１３に記載の方法。
【請求項１６】
前記大きな整数値は、ＧＵＩＤ（Global Unifier Identifier）であることを特徴とする請求項１５に記載の方法。
【請求項１７】
前記複数のコンピューティング・デバイスを決定することは、さらに、前記大きな整数値であるキーを使用する分散型ハッシュ・テーブル（ＤＨＴ）プロトコルを使用することを備えることを特徴とする請求項１５に記載の方法。
【請求項１８】
さらに、前記ＵＲＬに関連付けられているサマリ・ファイルを取得することを含み、前記サマリ・ファイルにより、前記Ｗｅｂファイルを備えるメッセージのインデックスおよび前記Ｗｅｂファイルの修正時刻を識別することを特徴とする請求項１３に記載の方法。
【請求項１９】
前記サマリ・ファイルは、ローカルでキャッシュされることを特徴とする請求項１８に記載の方法。
【請求項２０】
前記サマリ・ファイルは、前記複数のコンピューティング・デバイスのうちの少なくとも１つから取得された複数のアイテムから再作成されることを特徴とする請求項１８に記載の方法。
【請求項２１】
さらに、前記複数のファイルを複数のオリジナル・メッセージに分割すること、
イレージャ・コーディング方式を使用して複数の符号化メッセージを派生させること、および
前記符号化メッセージのうちの１つまたは複数をそれぞれの部分集合に追加すること
を備えることを特徴とする請求項１３に記載の方法。
【請求項２２】
前記イレージャ・コーディング方式は、リード・ソロモン・コーデックを備えることを特徴とする請求項２１に記載の方法。
【請求項２３】
前記複数のコンピューティング・デバイスとのネゴシエーションを実行することは、前記イレージャ・コーディング方式に関連付けられたキーを送信することを含むことを特徴とする請求項２１に記載の方法。

【請求項２４】
さらに、前記再作成されたファイルをブラウザで表示すること備えることを特徴とする請求項１３に記載の方法。
【請求項２５】
請求項１３ないし２４のいずれかに記載の方法を実行するためのコンピュータ実行可能命令を備えることを特徴とするコンピュータ可読記録媒体。
【請求項２６】
ホスト・コンピューティング・デバイスおよび１または複数のピア・コンピューティング・デバイスを含むコンピュータ・クラスタ内の複数のコンピューティング・デバイスに、Ｗｅｂサイトのコンテンツを分配しているシステムであって、
前記コンテンツは、複数のファイルを含み、
前記複数のファイルのそれぞれに、前記Ｗｅｂサイトのディレクトリィ間の相対的重要度を表すサイト重み（Ｓ）に、コンテンツのタイプに応じた重要度を表すタイプ重み（Ｔ）を乗算したファイル重み（ＦＷ）が割り当てられ、
前記複数のファイルのそれぞれの複製係数に基づいて決定されている前記コンテンツの複数の部分集合が作成され、前記複数のコンピューティング・デバイスのそれぞれに分配されていることであって、前記複製係数は、前記部分集合の１つを格納するコンピュータ・デバイスの記憶容量に基づく複製量（Ｄ）に基づいて、下式により決定され、
【数２】

$$D(\lambda_i) = \sum_m |Pm| \max\{1, FW_m \times \lambda_i\}$$

ここで、Ｄ（λｉ）はピア・コンピューティング・デバイスに複製される総量であり、λｉはピア・コンピューティング・デバイスについての複製係数であり、ＦＷｍは前記Ｗｅｂサイトの内のｍ番目のファイルのファイル重みであり、ＰｍはＷｅｂサイトの内のｍ番目のファイルのファイルサイズであることを特徴とするシステム。
【請求項２７】
さらに、前記ファイルを複数のメッセージに分割することを含み、前記ファイルを送信することは、前記ファイルに関連付けられている前記複数のメッセージのうちの少なくとも１つを前記１つのコンピューティング・デバイスに送信することを備えることを特徴とする請求項２６に記載のシステム。
【請求項２８】
さらに、イレージャ・コーディング方式を適用して複数の符号化メッセージを派生させ、前記符号化メッセージを前記メッセージとすることを特徴とする請求項２７に記載のシステム。
【請求項２９】
ホスト・コンピューティング・デバイスおよび１または複数のピア・コンピューティング・デバイスを含むコンピュータ・クラスタ内の複数のコンピューティング・デバイスから、Ｗｅｂサイトのコンテンツを取り出すクライアント・コンピューティング・デバイスであって、
前記コンテンツは、複数のファイルを含み、
前記複数のファイルのそれぞれに、前記Ｗｅｂサイトのディレクトリィ間の相対的重要度を表すサイト重み（Ｓ）に、コンテンツのタイプに応じた重要度を表すタイプ重み（Ｔ）を乗算したファイル重み（ＦＷ）が割り当てられ、
前記複数のファイルは、下式により決定される複製比に基づいて前記複数のコンピューティング・デバイスに分配されており、
$w_{m,i} = \max\{1, FW_m \times \lambda_i\}$
ここで、$w_{m,i}$はピアｉに対するｍ番目のファイルの複製比であり、λｉはピアｉについて計算された複製係数であり、ＦＷｍは前記Ｗｅｂサイトの内のｍ番目のファイルのファイル重みであり、前記クライアント・コンピューティング・デバイスは、
前記ＷｅｂサイトのコンテンツのＵＲＬを受け取り、

前記コンテンツの部分集合であるファイルを格納しているコンピューティング・デバイスであって、前記コンテンツが取り出されることになる複数のコンピューティング・デバイスを決定して、

前記複数のコンピューティング・デバイスとのネゴシエーションにより、前記コンテンツに含まれる前記複数のファイルを取得して、前記Ｗｅｂサイトの前記コンテンツを再作成することを特徴とするクライアント・コンピューティング・デバイス。

【請求項３０】

前記複数のコンピューティング・デバイスを決定することは、前記ＵＲＬを大きな整数に変換すること、分散型ハッシュ・テーブル（ＤＨＴ）プロトコルを適用して、前記大きな整数をキーとして保持している前記コンピューティング・デバイスの位置を突きとめることを備えることを特徴とする請求項２９に記載のクライアント・コンピューティング・デバイス。

【発明の詳細な説明】

【技術分野】

【０００１】

本明細書は、概して、Ｗｅｂコンテンツのホスティングに関し、より詳細には、コンピュータ・クラスタ内のＷｅｂコンテンツのホスティングに関する。

【背景技術】

【０００２】

インターネットで容易に入手可能な情報量は過去数年の間に著しく増大した。最近まで、ほとんどの情報は企業Ｗｅｂサイトにより提供されていた。しかし、今日では、多くの個人ユーザがパーソナルＷｅｂページとして情報を公開（publish）している。これらのパーソナルＷｅｂページは、日記、Ｗｅｂログ（通例、ブログ（blog）と呼ばれる）、パーソナル写真／ビデオ・コレクション、個人的な意見（パーソナル・アドバイス）、個人的な体験（パーソナル・エクスペリエンス）などの様々なＷｅｂコンテンツを含む場合がある。インターネットは、このようなＷｅｂコンテンツを公開する優れた手段を提供するが、Ｗｅｂコンテンツをパブリッシュする個人ユーザの能力に影響を与える様々な障害がある。

【０００３】

一般に、個々のコンテンツ所有者は、自分のＷｅｂコンテンツのホスティングを行ううえで、（１）商用データ・センター上でのホスティングすること、または（２）自分のインターネットリンクを使用して自分のパーソナル・コンピュータ上でのホスティングすること、の２つのオプションが与えられる。商用データ・センターのオプションでは、信頼できるサーバおよび帯域幅のリンクが提供される。しかしながら、いくつかの短所もある。例えば、コンテンツ所有者は、ホスト・サービスについて追加料金を支払わなければならない。彼らは、データ・センターでホスティングできるコンテンツの量において制約を受ける。また、彼らは、データ・センターを通じて利用できる毎日および毎月のトラフィック量も制約を受ける。コンテンツ所有者は、お気に入りのアプリケーションまたはツールを、データ・センター側でそれらのアプリケーションまたはツールをサポートしていないという理由で、利用することができないかも知れない。さらに、データ・センターを利用する多数の人々の間で強力なサーバおよび高帯域リンクが共有されるけれども、個々のユーザの要求に応じて割り当てられる計算能力およびネットワーク帯域幅は制限される可能性がある。

【０００４】

一方、上述のオプション２では、コンテンツ所有者は、自分のコンピュータ上でＷｅｂコンテンツをホスティングすることができる。所有するコンピュータを使用してＷｅｂコンテンツをホスティングすれば、追加料金の発生が回避され、利用できるホスティング・スペースのサイズは事実上無制限であり（使用ハード・ドライブのサイズのみにより制限される）、所有者はアプリケーション／データベースを自由に選択できる。しかし、そこには障害もある。障害の１つは、Ｗｅｂコンテンツを他のユーザに配信する際に、信頼性

が低い、不十分であるという特性である。Ｗｅｂコンテンツに連続したアクセスを提供するためには、所有者のホーム・コンピュータおよびインターネット接続が、常時、稼働し、正常に機能していなければならない。ホーム・コンピュータで障害が発生したり、またはコンピュータの電源が不注意に切られたりすると、コンテンツを提供できなくなる。さらに、所有者のインターネット接続がダウンした場合も、コンテンツを提供できなくなる。コンピュータおよびインターネット接続が絶対にダウンしないようにすることが可能であっても、さらに、十分な帯域幅を確保するなどの、克服すべき他の障害が残っている。インターネット・サービス・プロバイダ（ＩＳＰ）がユーザのホーム・コンピュータからのインターネット接続のアップロード速度を制限するのはまれなことではない。このアップロード速度がＷｅｂコンテンツを他のユーザに高速に配信するのに十分であるということはめったにない。例えば、ブロードバンド接続の場合であっても、アップロード速度は、通常、１２８Ｋｂｐｓに制限されるのはふつうであり、これはＷｅｂコンテンツ・アクセス要求に応えるのに十分な帯域幅とはいえない。

【０００５】

Ｗｅｂコンテンツを公開する企業は、高価なサーバ・アレイおよびより高速なインターネット接続に投資することでこのような障害を克服することも可能である。しかし、このような選択肢は、費用が高く、大半の個人ユーザにとっては利用できないものである。幸いなことに、いくつかの種類のＷｅｂコンテンツについては、代替えとなる費用効果の高いソリューションが出現している。この代替えソリューションは、ピア・ツー・ピア（Ｐ２Ｐ）ネットワークを構築するものである。Ｐ２Ｐコンシューマ・アプリケーションの例として、「ＮＡＰＳＴＥＲ」、「ＫＡＺＡＡ」、および「ｇｎｕｔｅｌｌａ」がある。これらのＰ２Ｐコンシューマ・アプリケーションはそれぞれ、複数のコンピュータ間でファイルを共有することに重点を置いている。ファイルの共有はＷｅｂサイトの共有に似ているように見えるが、Ｗｅｂサイトの共有には固有の問題点がある。

【０００６】

【非特許文献１】"Reed-Solomon Codes and their applications", by S. B. Wicker and V. K. Bhargava, IEEE Press, New York, 1994

【非特許文献２】Bayardo, Jr., R. et al., "YouServ: A Web-Hosting and Content Sharing Toot for the Masses", Proceedings of the International World Wide Web Conference, pp. 345-354, May 2002.

【非特許文献３】'Ratnasamy, S., et al., "Routing Algorithms for DHTs: Some Open Questions", IPTPS '02, pp. 45-52, 2002.

【非特許文献４】Petar Maymounkov et al., "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric", IPTPS 2002, pp. 52-65, July 2002.

【非特許文献５】Ngan, et al., "Enforcing Fair Sharing of Peer-to-Peer Resources", IPTPS '03, pp. 149-159, 2003.

【非特許文献６】Cardeliini, V., et at. "The state of the art in locally distributed web-server systems", ACM Computing Surveys, Vol. 34, No. 2, pp. 263-311, September 2002.

【発明の開示】

【発明が解決しようとする課題】

【０００７】

問題点の１つは、共有されなければならない情報の量である。Ｗｅｂサイトは、複数のＷｅｂページを有し、それぞれのＷｅｂページはファイルの集合からなる。複数のＷｅｂページの１つの関連付けられているハイパーリンクが選択された場合、ファイルの集合全体が直ちに使用できなければならない。したがって、Ｗｅｂサイトの共有は、１個のファイルを共有するのと比べて、大量の記憶領域を必要とし、また広い帯域幅を消費する。他の問題点としては、ファイルの集合を提供する際の取り出し速度および応答時間がある。現在のＰ２Ｐコンシューマ・アプリケーションでは、１つのファイルの取り出しの実行は、きわめて低速であり、数時間から数日を要することさえある。さらに、取り出しは、ネ

ットワークが混雑していない時間に、またはコンテンツを保持しているコンピュータがオンラインになるとすぐに、スケジュールすることができる。要求側クライアントがＷｅｂページを表示するまで何時間も待つ、またはコンピュータがオンラインになるまで待つことはありえないため、Ｗｅｂページを取り出すときに、このように取り出し速度が遅いのは受け入れがたいことである。

【０００８】

これらの問題点を克服するために、Ｗｅｂコンテンツを複数のコンピュータに複製する試みがいくつかなされた。すると、クライアントによりコンテンツがアクセスされたときに、コンテンツは、所有者のホーム・コンピュータから、またはコンテンツ全体をホスティングする他のコンピュータのうちの１つからアクセス可能ある。複数のコンピュータにＷｅｂコンテンツを複製すると、すべてのコンピュータおよびその関連するネットワークリンクが同時にダウンする可能性は低いため、Ｗｅｂコンテンツの信頼性が向上する。しかし、コンテンツを提供する帯域幅は、コンテンツ全体がまだ１台のコンピュータおよびその関連するネットワーク接続から取り出されるので、同じままである。このタイプのシステムでは、Ｗｅｂページへのアクセスの信頼性は向上するが、システムは、依然として、コンテンツ全体を格納するため大量の記憶容量を必要とし、またコンテンツ全体を配信するために広い帯域幅を必要とする。したがって、現在まで、一般大衆が使用するのに好適な、Ｗｅｂコンテンツを公開するための満足のいくソリューションは得られていない。

【課題を解決するための手段】

【０００９】

本明細書で説明されている手法およびメカニズムは、部分的複製を使用する複数のコンピューティング・デバイス上で、Ｗｅｂサイトのコンテンツをホスティングすることを対象とする。

【００１０】

Ｗｅｂサイトに関連付けられているそれぞれのファイルに対する相対的重要度が計算される。この相対的重要度を使用して、サーバ・アレイ、ピア・ツー・ピア・ネットワークなどの、コンピュータ・クラスタ内の複数のデバイスに分配されるコンテンツの複数の部分集合を計算する。これらの部分集合は、１つまたは複数のファイルの一部分を含むパケットに関してイレージャ・コーディング方式（erasure coding scheme）を使用して作成された符号化メッセージを含むことができる。ファイルの取得時、一定数のはっきりと識別可能なパケット（distinct packets）が、イレージャ・コーディング方式に基づいてデバイスから取り出される。ファイルは、これらのはっきりと識別可能なパケットにより再作成される。複数のデバイスがコンテンツを保持するので、Ｗｅｂサイトに対する取り出しをかなり高速化することができ、どのコンピューティング・デバイスも大量の記憶領域または帯域幅を必要とせずに、信頼性が向上する。

【００１１】

以下の図を参照しつつ、非限定的非網羅的実施形態を説明するが、類似の参照番号は、特に断りのない限り、様々な図面全体を通して類似の部分を指す。

【発明を実施するための最良の形態】

【００１２】

要するに、本発明のＷｅｂホスティング・メカニズムは、それぞれがＷｅｂコンテンツの部分集合を格納するコンピュータ・クラスタ内の複数のコンピューティング・デバイスからＷｅｂコンテンツを取り出すことをサポートするということである。本発明のＷｅｂホスティング・メカニズムは、さらに、コンピュータ・クラスタ内の複数のコンピューティング・デバイス上に格納される複数の部分集合にＷｅｂコンテンツを分散させることもサポートする。以下で詳述するように、この分配方法では、Ｗｅｂコンテンツを複製する際に必要な記憶領域のサイズを極力抑える。さらに、この分配および取り出し方法は、Ｗｅｂコンテンツに対する信頼性およびアクセス時間を向上させる。これらおよびその他の利点は、以下の詳細な説明を読むと明らかになるであろう。

【００１３】

図１は、本発明のＷｅｂホスティング・メカニズムを実装するためのシステムの一実施例を示している。システムは、コンピューティング・デバイス１００などのコンピューティング・デバイスを備える。最も基本的な構成では、コンピューティング・デバイス１００は、少なくとも１つの処理ユニット１０２およびシステム・メモリ１０４を備えるのがふつうである。コンピューティング・デバイスの正確な構成と種類に応じて、システム・メモリ１０４は揮発性（ＲＡＭなど）、不揮発性（ＲＯＭ、フラッシュ・メモリなど）、またはこれら２つの何らかの組合せとすることができる。システム・メモリ１０４は、通常、オペレーティング・システム１０５、１つまたは複数のプログラム・モジュール１０６を格納し、プログラム・データ１０７を含むこともある。プログラム・モジュール１０６は、コンテンツを複数のコンピューティング・デバイスに分配し、それらからコンテンツを取り出す本発明のＷｅｂホスティング・メカニズムを実装するモジュール１３０を備える。さらに、システム・メモリ１０４は、Ｗｅｂページを特定し、表示するためのブラウザを備える。基本構成は、図１において点線１０８内のコンポーネントにより示されている。

【００１４】

コンピューティング・デバイス１００は、さらに特徴または機能を追加することもできる。例えば、コンピューティング・デバイス１００は、磁気ディスク、光ディスク、またはテープなどの追加データ記憶デバイス（取り外し可能および／または取り外し不可能）を備えることもできる。このような追加記憶装置は、図１では、取り外し可能記憶装置１０９および取り外し不可能記憶装置１１０により例示されている。コンピュータ記憶媒体は、コンピュータ可読命令、データ構造体、プログラム・モジュール、またはその他のデータなどの情報を格納する方法または技術で実装される揮発性および不揮発性、取り外し可能および取り外し不可能媒体を含むことができる。システム・メモリ１０４、取り外し可能記憶装置１０９、および取り外し不可能記憶装置１１０は、すべてコンピュータ記憶媒体の実施例である。したがって、コンピュータ記憶媒体としては、限定するものではないが、ＲＡＭ、ＲＯＭ、ＥＥＰＲＯＭ、フラッシュ・メモリもしくはその他のメモリ技術、ＣＤ－ＲＯＭ、デジタル多目的ディスク（ＤＶＤ）もしくはその他の光ディスク記憶装置、磁気カセット、磁気テープ、磁気ディスク記憶装置もしくはその他の磁気記憶デバイス、または所望の情報を格納するために使用することができコンピューティング・デバイス１００によりアクセスできるその他の媒体がある。このような任意のコンピュータ記憶媒体をデバイス１００の一部とすることができる。さらにコンピューティング・デバイス１００は、キーボード、マウス、ペン、音声入力デバイス、タッチ入力デバイスなどの入力デバイス１１２を備えることもできる。ディスプレイ、スピーカ、プリンタなどの出力デバイス１１４を備えることもできる。これらのデバイスは、当業ではよく知られているため、本明細書でさらに詳しい説明をする必要はない。

【００１５】

また、コンピューティング・デバイス１００は、デバイスがネットワークなどを経由して他のコンピューティング・デバイス１１８と通信するために使用する通信接続１１６も含むことができる。（複数の）通信接続１１６は、通信媒体の一実施例である。通信媒体は、通常、搬送波もしくはその他のトランスポート・メカニズムなどの変調データ信号を介して、コンピュータ可読命令、データ構造体、プログラム・モジュール、またはその他のデータによって具現化することができ、情報配信媒体を含む。「変調データ信号」という用語は、信号内の情報を符号化する方法によりその特性のうち１つまたは複数が設定または変更された信号を意味する。例えば、通信媒体としては、限定するものではないが、有線ネットワークまたは直接配線接続などの有線媒体、ならびに、音響、ＲＦ、赤外線、およびその他の無線媒体などの無線媒体がある。コンピュータ可読媒体は、コンピュータによりアクセスできる入手可能な媒体であればどのようなものでもよい。例えば、コンピュータ可読媒体は、限定するものではないが、「コンピュータ記憶媒体」および「通信媒体」を含むことができる。

【００１６】

様々なモジュールおよび手法は、１つまたは複数のコンピュータまたはその他のデバイスにより実行される、プログラム・モジュールなどのコンピュータ実行可能命令の一般的なコンテクストで、本明細書で説明することができる。一般に、プログラム・モジュールは、特定のタスクを実行する、または特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、コンポーネント、データ構造などを含む。これらのプログラム・モジュールなどは、ネイティブ・コードとして実行するか、または仮想マシンまたはその他のジャスト・イン・タイム・コンパイル実行環境などで、ダウンロードし実行することができる。通常、プログラム・モジュールの機能は、様々な実施形態で望まれているように組み合わせるか、または分散させることができる。これらのモジュールおよび手法の実装は、コンピュータ可読媒体に格納するか、または何らかの形のコンピュータ可読媒体で伝送することができる。

【００１７】

図２は、図１に示されている、コンピューティング・デバイス１００などの２つ以上のコンピュータデバイスが本発明の分散型Ｗｅｂホスティング・メカニズムの手法およびメカニズムを実装するように構成されているネットワークの図である。ネットワークは、コンピュータ・クラスタ２００と呼ぶことができる。一実施形態では、コンピュータ・クラスタ２００は、サーバ・アレイとして構成することができる。他の実施形態では、コンピュータ・クラスタ２００は、ピア・ツー・ピア（Ｐ２Ｐ）ネットワークを使用して構成することができる。一般に、コンピュータ・クラスタ２００は、複数のコンピューティング・デバイス２０２～２１２を備える。それぞれのコンピューティング・デバイス２０２～２１２は、他のコンピューティング・デバイス２０２～２１２のうちの１つまたは複数と通信するように構成されている。通信路は、図２では、コンピューティング・デバイス２００２～２１２のうちの２つの間の実線により表されている。通信路は、ローカル・エリア・ネットワーク、インターネット、無線などを経由することができる。コンピューティング・デバイスのうちの１つ（例えば、コンピューティング・デバイス２０２）は、オリジナルのＷｅｂコンテンツ（例えば、Ｗｅｂコンテンツ２２２）を保持する。以下の説明全体を通して、コンピューティング・デバイス２０２は、ホスト・コンピュータ２０２と呼ばれる。他のコンピューティング・デバイス（例えば、コンピューティング・デバイス２０４～２１２）は、それぞれ、Ｗｅｂコンテンツ２２２の部分集合（例えば、部分集合２２４～２３２）を保持する。他のコンピューティング・デバイス２０４～２１２はホスト・コンピュータのＷｅｂコンテンツの部分集合を保持するので、コンピューティング・デバイス２０４～２１２は、以下の説明全体を通してホスト・コンピュータ２０２のピア２０４～２１２と呼ばれる。ピアという用語を使用するといっても、コンピュータ・クラスタ２００がピア・ツー・ピア・ネットワークとして構成されている必要はないことに留意されたい。むしろ、ピアという用語は、ピアが他のコンピューティング・デバイスのためにコンテンツを保持することを反映する。

【００１８】

概して、ホスト・コンピュータ２０２では、Ｗｅｂコンテンツ２２２を、クライアントコンピューティングデバイス２６０などの１つまたは複数のコンピューティング・デバイス（これ以降クライアント２６０と呼ぶ）に配信することを望む。クライアント２６０は、コンピュータ・クラスタ２００の一部として示されていないことに留意されたい。これは標準的な構成であるが、本発明の分散型Ｗｅｂホスティング・メカニズムは、クライアント２６０がコンピュータ・クラスタ２００の一部である場合と同様に等しく機能する。いずれの構成でも、クライアント２６０は、以下で説明される本発明の分散型Ｗｅｂホスティング・メカニズムと連携し、よく知られている手法使用してインターネット経由でピア上の部分集合にアクセスすることができる。パーソナル・コンテンツを公開するための従来の実装では、クライアント２６０はホスト・コンピュータ２０２、またはＷｅｂコンテンツ２２２全体を複製した他のコンピューティング・デバイスからＷｅｂコンテンツ２０３全体を取得する。しかし、上述のように、これらの実装では、かなりの記憶領域を消費し、ホスト２０２または他のコンピューティング・デバイスの帯域幅のみを利用する。

したがって、本発明の分散型Ｗｅｂホスティング・メカニズムは、ピア２０４～２１２のそれぞれにＷｅｂコンテンツ２２２の部分集合２２４～２３２を複製することを重視する。それぞれの部分集合２２４～２３２は、コンテンツ２２２の異なる集合を含むことができる。ピア２０４～２１２はＷｅｂコンテンツ２２２全体を格納しないため、クライアント２６０は、本発明の分散型Ｗｅｂホスティング・メカニズムにより複数のピア（例えば、ピア２０８～２１２）からＷｅｂコンテンツ２２２全体を取得する。

【００１９】

Ｗｅｂコンテンツ２２２の部分集合を決定し、これらの部分集合２２４～２３２をピア２０４～２１２に分配するメカニズムは、図３に示されている流れ図３００で例示される。概して、分配プロセス３００は、様々なピア上にコンテンツのどの部分集合を複製するかを決定する。コンテンツ全体がピアのそれぞれに分配されるわけではないため、ピアに対する格納コストは極力抑えられ、部分集合をピアに分配することに関連するコスト／時間も極力抑えられる。

【００２０】

ブロック３０２で、タイプ重みがＷｅｂサイト階層内の各ファイルについて割り当てられる。Ｗｅｂコンテンツは、テキスト、画像、ビデオなどのいくつかのタイプのコンテンツを含むことができる。これらのいくつかのタイプのコンテンツは、それぞれ、コンテンツのタイプに関連付けられた異なる属性を持つことができる。本発明の分散型Ｗｅｂホスティング・メカニズムにより、それぞれのタイプのコンテンツに特定の重み（つまり、タイプ重み）が割り当てられる。タイプ重みは、他のタイプのコンテンツに関連した、そのタイプのコンテンツの重要度を反映する。一実施形態では、重みは大きいほど、このタイプのコンテンツを含むＷｅｂページのどれかのページが取り出されるときにこのタイプのコンテンツが欠落していないことを保証したいことを示す。例えば、アイコンをＷｅｂページに表示させることは、Ｗｅｂページ上にテキストを表示させることに比べればあまり重要でないといえる。したがって、アイコン・コンテンツに割り当てられるタイプ重みは、テキスト・コンテンツに割り当てられるタイプ重みよりも低くできる。一実施形態では、ユーザはコンテンツのそれぞれの個々のタイプに対し、このタイプ重みを割り当てることができる。例えば、ユーザは、各タイプのファイル拡張子に対しタイプ重みを割り当てることができる。その後、分配プロセスは、ファイル拡張子が認識されると、コンテンツに対し指定されたタイプ重みを割り当てる。他の実施形態では、異なるタイプのコンテンツに対し、既定のタイプ重み設定を設定できる。ユーザは、後から、既定の設定を指定変更することもできる。処理は、ブロック３０４に続く。

【００２１】

ブロック３０４で、サイト重みがＷｅｂサイト階層内の各ファイルについて割り当てられる。一般に、Ｗｅｂサイトは、興味、時間／イベントなどのトピック別にまとめることができる。特定のＷｅｂページに関連付けられているファイルは、ディレクトリに入れることにより、記憶媒体上に編成することができる。したがって、Ｗｅｂサイト階層は、図４に示されているように、階層型ディレクトリ・ツリーとして図形で例示することができる。

【００２２】

図４では、Ｗｅｂサイト階層例は、ｗｏｒｋディレクトリ４１０およびｐｅｒｓｏｎａｌディレクトリ４３０の２つのサブ・ディレクトリを含むｒｏｏｔディレクトリ４０２を含む。これらのサブ・ディレクトリは両方とも、さらに、サブ・ディレクトリを持つ。ｗｏｒｋディレクトリ４１０は、ｐｒｏｊｅｃｔ　１サブ・ディレクトリ４１２とｐｒｏｊｅｃｔ　２サブ・ディレクトリ４１８を持つ。ｐｒｏｊｅｃｔ　１とｐｒｏｊｅｃｔ　２は両方とも、それぞれｄａｔａ　４１４および４２０ならびにｓｕｍｍａｒｙ　４１６および４２２の２つのサブ・ディレクトリを持つ。ｐｅｒｓｏｎａｌディレクトリ４２０は、ｔｒｉｐ　２００２　４３２およびｔｒｉｐ　２００４　４３４の２つのサブ・ディレクトリを持つ。Ｗｅｂサイト階層の実際のレイアウトに特に興味があるわけではないが、それぞれのディレクトリはそのディレクトリに関連付けられたサイト重みを持つことには

留意されたい。ディレクトリのサイト重みは、Ｗｅｂサイト内の様々なディレクトリ間の相対的重要度を反映するように、Ｗｅｂサイト階層の所有者により割り当てられる。ディレクトリ間の相対的重要度は、Ｗｅｂサイトを閲覧している人が、特定のコンテンツが欠落している場合に経験するであろう困惑の様々なレベルを反映する。相対的重要度は、さらに、特定のコンテンツへのアクセスの可能性が他のコンテンツへのアクセスに比べて高いことも反映する。例えば、最近作成されたＷｅｂページは、古いＷｅｂページよりもアクセスされる可能性が高い。

【００２３】

サイト重みの割り当ては、図４に示されているようなグラフィックを用いたＷｅｂサイト階層を形成し、ユーザがそれぞれのディレクトリまたはファイルを選択し、サイト重みを割り当てられるようにすることで実現できる。サイト重みを割り当てるプロセスを減らすため、サイト重み割り当てを少数のディレクトリだけに絞るとよい。そこで、未割り当てディレクトリは、親ディレクトリから自サイト重みを継承することができる。したがって、ディレクトリまたはファイルが自サイト重みを割り当てられていない場合、そのディレクトリまたはファイルは、親からサイト重みを継承することができる。例えば、ｗｏｒｋディレクトリ４１０には、サイト重み２．０が割り当てられている。ｐｒｏｊｅｃｔ２ディレクトリおよびそのサブ・ディレクトリには、具体的サイト重みは割り当てられていない。したがって、ｐｒｏｊｅｃｔ２　４１８およびｄａｔａ　４２０、およびｓｕｍｍａｒｙ　４２２は、ｗｏｒｋディレクトリ４１０からサイト重み２．０を継承する。他の実施形態では、サイト重みをファイル内にリストとして記述することができる。サイト重みを割り当てるためのこれらの変更形態およびその他の変更形態により、本発明の分散型Ｗｅｂホスティング・メカニズムがサイト重みを入手することができる。サイト重みおよびタイプ重みが割り当てられた後、処理は図３のブロック３０６に続く。

【００２４】

ブロック３０６で、ファイル重みＦＷがＷｅｂサイト階層内の各ファイルについて計算される。一般に、以下に示されるように、ファイル重みＦＷは、関連付けられているファイルｍに対する複製比$w_{m,i}$の計算に影響を及ぼす。複製比$w_{m,i}$は、コンテンツ（またはファイル）がピア上に何回複製されるかを決定する。したがって、複製比$w_{m,i}$は、特定のファイルｍに対する取り出し信頼度に影響を及ぼす。ファイルｍのファイル重みＦＷは、以下の式に示されているように、ファイルのサイト重みＳにタイプ重みＷを掛けることで計算される。

$$FW_m = S_m \times T_m \qquad \text{（式１）}$$

ただし、ｍはＷｅｂサイト内のｍ番目のファイルを表す。一般に、以下に示されるように、ファイル重みＦＷを２倍にすると、コンテンツ複製比も２倍になり、その結果得られるファイルは提供帯域幅および取り出し信頼度の２倍で取り出される。個々のファイル重みＦＷがＷｅｂサイト階層内の各ファイルｍについて計算された後、処理はブロック３０８に続く。

【００２５】

ブロック３０８で、相対的ピア複製係数λがピアについて決定される（例えば、図２に示されているピア２０４～２１２）。相対的ピア複製係数λは、ホスト・コンピューティング・デバイスについてピアが格納することに同意した同意複製量Ｄに基づく。相対的ピア複製係数λは、相対的ピア複製係数λについて以下の方程式を解くことにより決定することができる。

【００２６】

【数１】

$$D(\lambda_i) = \sum_m |Pm| \max\{1, FW_m \times \lambda_i\} \qquad (式2)$$

【００２７】

ここで、Ｄ（λ_i）は、ピアｉに複製されるコンテンツの総量を表し、λ_iは、ピアｉ

について計算されるピア複製係数を表し、FW_mは、Ｗｅｂサイト内のｍ番目のファイルに対するファイル重みを表し、P_mは、Ｗｅｂサイト内のｍ番目のファイルのファイル・サイズを表す。したがって、ピアｉに複製されるコンテンツの総量Ｄ（λ_i）は同意した量であり、ファイル・サイズおよびファイル重みは、Ｗｅｂサイト内のファイル毎に知られているため、相対的ピア複製係数λ_iを決定できる。

【００２８】

二分探索法（bi-sectional search）を実行することによって、相対的ピア複製係数を決定する１つの方法がある。上記の方程式は、相対的ピア複製係数λ_iが増加する場合に単調増加関数となるので、二分探索法は成功する。同意した複製量について、各ピアとネゴシエートすることができる。例えば、同意した複製量は、ホスト・コンピュータがピアのＷｅｂサイトの複製をサポートするためホスト・コンピュータ上の同意した量の複製をやり取りするというホスト・コンピュータによる双方の合意を表すことができる。複製係数λがピアとの格納契約、つまりＤ（式２を参照）に依存することに留意されたい。すべてのピアの同意した量がＷｅｂサイト全体に必要な最低限の記憶装置サイズに等しいという制約はない。実際、本発明のホスティング・メカニズムによれば、各ピアにより格納される複製データの量がどうあれ、信頼性および取り出し速度は改善される。さらに、複製係数λが小さくても、ファイル重みFW_mの大きいいくつかのファイルを大規模に複製することができ、それにより、ファイルの信頼性および取り出し速度が向上する。個々のピアに対するピア複製係数が決定された後、処理はブロック３１０に続く。

【００２９】

ブロック３１０で、複製のためピアに送られるコンテンツを決定するため、Ｗｅｂサイト内のそれぞれのファイルのピア複製比を計算する。それぞれのファイルのピア複製比は、以下の式を使用して計算される。

$$w_{m,i} = \max\{1, FW_m \times \lambda_i\} \qquad \text{（式３）}$$

ここで、$w_{m,i}$は、ピアｉに対するｍ番目のファイルのピア複製比を表し、λ_iは、ピアｉについて計算された相対的ピア複製係数を表し、FW_mは、Ｗｅｂサイト内のｍ番目のファイルに対するファイル重みを表す。ピア複製比$w_{m,i}$は、ｍ番目のファイルのサイズ｜P_m｜に比例する、ピアｉに送られるｍ番目のファイルに関係するコンテンツの量として見ることができる。ピア複製率が決定されると、処理はブロック３１２に続く。

【００３０】

ブロック３１２で、これらのファイルはメッセージ内に配置される。一時的に、図５を参照すると、これらのメッセージの作成が説明されている。すでに述べたように、Ｗｅｂサイトはページの集合である。それぞれのＷｅｂページは、複数のファイルを含む（つまり、Ｗｅｂファイル５０１）。これらのＷｅｂファイル５０１は、多数のパケットに分けられる（例えば、パケット５０２～５１０）。例えば、大きなファイルを複数のパケットに分割したり、または複数の小さなファイルを１つのパケットにまとめたりすることができる。一実施形態では、これらのパケット５０２～５１０は、データ操作がしやすいように固定サイズである。しかし、この実施形態では、パケットの一部が固定サイズでなくてよいことに留意されたい。例えば、大きなファイルを複数のパケットに分割するときにこのようなことが生じることがある。最後のパケットは、他のパケットよりも小さくなることがある。一般に、この実施形態については、パケットの大半は固定サイズである。これらのパケットはそれぞれ、ｋ個のメッセージ（例えば、メッセージ５１２～５２０）にさらに分割できる。再び、これらｋ個のメッセージは、サイズ固定としてよい。１つのメッセージ（例えば、メッセージ５１２）は、オリジナルのパケット５０６のコンテンツに応じて、オリジナルの１つのＷｅｂサイト・ファイルの一部を含むか、または複数のＷｅｂサイト・ファイルの一部分を持つことができる。処理は、図３のブロック３１６に続く。

【００３１】

図３を参照すると、ブロック３１６で、多数のこれらｋ個のメッセージがピアに分配される。ホスト・コンピュータとピアｉとの間の接続上のトラフィックが最小の場合にメッ

セージを分配すると都合がよい。一実施形態では、ランダムな数ｋのメッセージがピアｉに送られる。メッセージのランダムな個数は、ｍ番目のファイルについてピアｉに対し計算されたピア複製比$w_{m,i}$に比例することができる。例えば、パケットＡが１６このメッセージに分割され、ピアＺに対するパケットＡに関連付けられているファイルに対するピア複製比は、．５であり、これら１６個のメッセージのうちの半分（つまり、８個のメッセージ）がピアＺに送られる。概して、図６について以下で詳述するように、取り出し時に、クライアントは、パケットＡを再作成するためにコンピュータ・クラスタ内のピアのうちの１つまたは複数から１６個のメッセージのそれぞれを特定する。この実施形態にはいくつかの短所がある。複数のピアに多数のメッセージが分配する場合でも、クライアントがパケットＡに対する要求を送信する時点で、そのコンピュータ・クラスタが、パケットＡに関連付けられているそれぞれの特定のメッセージを利用可能な状態で持たない場合がある。このようなことになったら、パケットＡだけでなく、パケットＡを含むファイルも、取り出せない。各パケットを再作成できる確率を高める方法の１つは、分配する前に、メッセージに関してイレージャ・コーディングを実行することである。そこで、ブロック３１６の前にブロック３１４を実行できる。

【００３２】

ブロック３１４で、任意選択で、メッセージに関してイレージャ・コーディングを実行することができる。以下に示されるように、イレージャ・コーディングをファイル重みと連携して適用すると、さらに、Ｗｅｂコンテンツの信頼性および取り出し速度が向上する。図５を再び参照すると、メッセージ５１２～５２０は、イレージャ・コーディングと呼ばれるよく知られている数学的ツールを使用して処理される。イレージャ・コーディングはデータの符号化に関してよく知られているが、Ｗｅｂサイトのコンテンツへの応用については、これまで考えられたことがなかった。概して、メッセージ５１２～５２０は、（ｎ，ｋ）イレージャ・コーデックを通じて処理され、ｎ個の符号化メッセージ（例えば、符号化メッセージ５２２～５５０）から成るイレージャ・コーディング空間を形成する。リード・ソロモン・イレージャ・コーデック、トーネイド・コーデック、およびＬＰＤＣコーデックなど、使用可能な複数のイレージャ・コーディング技術がよく知られている。

【００３３】

メッセージ誤り訂正符号として、（ｎ，ｋ）イレージャ・レジリエント符号（erasure resilient code）のオペレーションは、ガロア体ＧＦ（ｐ）上の行列乗算を介して記述することができる。

【００３４】

【数２】

$$\begin{bmatrix} c_0 \\ c_1 \\ M \\ M \\ c_{n-1} \end{bmatrix} = G \begin{bmatrix} x_0 \\ x_1 \\ M \\ x_{k-1} \end{bmatrix} \qquad (式4)$$

【００３５】

ここで、ｐはガロア体の次数であり、$\{x_0, x_1, \ldots, x_{k-1}\}$はオリジナル・メッセージであり、$\{c_0, c_1, \ldots, c_{n-1}\}$は符号化メッセージであり、Ｇは生成行列（generator matrix）である。一実施形態では、符号化されたメッセージは、すべて同時に生成されるわけではない。むしろ、本発明のホスティング・メカニズムでは、生成行列Ｇを使用して、符号化メッセージ空間を定義する。クライアントがｋ個の符号化メッセージ$\{c'_0, c'_1, \ldots, c'_{k-1}\}$を受信した場合、これらｋ個の符号化メッセージは以下のように表すことができる。

【００３６】

【数３】

$$\begin{bmatrix} c'_0 \\ c'_1 \\ M \\ c'_{k-1} \end{bmatrix} = G_k \begin{bmatrix} x_0 \\ x_1 \\ M \\ x_{k-1} \end{bmatrix} \qquad (式5)$$

【００３７】

ここで、G_kは、符号化メッセージに対応する生成行列Ｇのｋ個の行（row）により形成される部分生成行列である。この部分生成行列G_kが最大階数ｋを持つ場合、行列G_kは反転させる（inverse）ことができ、したがって、オリジナル・メッセージは復号化できる。

【００３８】

それぞれのイレージャ・コーティング技術は、それ独自の利点を有している。例えば、リードソロモン符号は、ｋ個のはっきりと識別可能な符号化メッセージが取り出される限り復号化を保証する最大距離分離可能（ＭＤＳ；maximum distance separable）特性を持つ。Ｗｅｂホスティング・アプリケーションにおける誤りの主要な形態は、ネットワーク伝送における接続の途絶またはパケットの喪失により引き起こされる符号化メッセージの喪失であるため、リードソロモン符号は、本発明の分散型Ｗｅｂホスティング・メカニズムに特に好適である。リードソロモン符号の詳細については、書籍を参照されたい（例えば、非特許文献１を参照）。

【００３９】

イレージャ符号のパラメータｋは、パケットの粒度とともにイレージャ・コーディング空間のサイズを決定する。オリジナルのパケット５０２～５１０は、それぞれ、ｋ個のサイズの等しいメッセージに分解されるため、パラメータｋの値が大きいほど、それぞれのパケットから得られるメッセージ５１２～５２０の個数は多くなる。このため、アクセスの粒度とイレージャ・エンコーディングのオーバーヘッドの両方が増大することになる。他方、ｋはユーザがコンテンツを同時に取り出すことができるピアの最大数を決める。したがって、ユーザが、多数のピアからコンテンツを取り出して、可能な最大の速度増大が得られるようにするために、適度なサイズｋを選択することが有益である。パラメータｎは、イレージャ符号から生成可能な符号化メッセージの個数を決定する。ｎに対し十分に大きな値を取ることで、異なるピアが異なる符号化メッセージを保持するようにできる。例示されているパラメータ群は、ｋ＝１６およびｎ＝2^k＝６５５３６である。このパラメータ群を使って、４０９６（６５５３６／１６）個のピアに対応できる。

【００４０】

図３をもう一度参照して、ブロック３１６で、これらの符号化メッセージ５２２～５５０の部分集合が、オリジナルのメッセージの代わりに、ピアに分配される。イレージャ・コーディングが適用されると、ブロック３１４で、それぞれのピアは、イレージャ・コーディング空間内のｎ個の符号化メッセージからＺ個のはっきりと識別可能な符号化メッセージを受け取る。個数Ｚは、ｍ番目のファイルおよびピアｉについて以下のように計算されるピア複製比$W_{m,i}$に基づく。

$$Z_i = W_{m,i} \times k \qquad （式６）$$

ここで、Z_iは、符号化メッセージのはっきりと識別可能な個数を表し、$W_{m,i}$は、ｍ番目のファイルとピアｉに対するピア複製比を表す。例えば、ピアが複製比０．５を持つ場合、そのピアはパケットに対するオリジナル・メッセージの個数の半分に等しい個数の符号化メッセージを受け取らなければならないことを意味する（つまり、５０％の重複率）。はっきりと識別可能な符号化メッセージの個数Z_iは、分数値でもよい。このような場合、Ｚｉは、

【００４１】

【数４】
$\lfloor x \rfloor$

【００４２】
をフロア関数とすると、確率
【００４３】
【数５】
$(1+\lfloor Z_i \rfloor - Z_i)$

【００４４】
であれば
【００４５】
【数６】
$\lfloor Z_i \rfloor$

【００４６】
であり、確率
【００４７】
【数７】
$(Z_i - \lfloor Z_i \rfloor)$

【００４８】
であれば
【００４９】
【数８】
$\lfloor Z_i \rfloor + 1$

【００５０】
である、と単に解釈されるだけである。したがって、この確率の調整により、いくつかのピアは１つ多い符号化メッセージを持ち、それ以外のピアは余分な符号化メッセージを持たないことになる。ピアに分配される符号化メッセージが一意であることを保証するために、異なるイレージャ・コーディング・キーをそれぞれのピアに割り当てることができる。イレージャ・コーディング・キーは、イレージャ符号に関連付けられている行列の行（row）インデックスから導くことができる。ファイルP_mの集計されたコンテンツ複製比は、C_m、即ち
【００５１】
【数９】
$$C_m = \sum_i W_{m,i} \qquad (式7)$$

【００５２】
と表され、コンピュータ・クラスタ内で複製されたファイルP_mのコピーの総数である。それらのピアに分配されるＷｅｂサイト・コンテンツのこれらの部分集合に関して、クライアントは要求を開始することができ、ｋ個のはっきりと識別可能な符号化メッセージが見つかることになる。
【００５３】
図６は、ピアの１つまたは複数から欠落していたアイテム（missing item）を取り出すための取り出しプロセス６００を例示する流れ図である。欠落しているアイテムは、実際のＷｅｂサイト・ファイル、パケット、メッセージ、または符号化メッセージである可能性がある。概して、それぞれのＷｅｂページは複数のピアから取り出されるため、Ｗｅｂ

ページを取り出すスループットは増大する。さらに、要求されたＷｅｂページを取得する信頼性も、所望のコンテンツを複数のコンピュータ（つまり、ピア）から取得できるため向上する。取り出し時の一般的原理は、オリジナルの個数（つまり、ｋ個）のアイテム（例えば、メッセージ、符号化メッセージ）をそれぞれのパケットに対する任意の数のピアから取り出すというものである。その後、取り出されたアイテムを組み合わせて、オリジナル・パケットを再作成し、これらを組み合わせてオリジナルのＷｅｂファイルを作成する。アイテムが符号化メッセージであれば、符号化メッセージをイレージャ復号化して、オリジナル・メッセージを取得し、続いて、これらを上述のように組み合わせる。個々のパケットについては、オンラインのすべてのピアに格納されているはっきりと識別可能なメッセージの個数がそのパケットに対するオリジナル・メッセージの個数（つまり、ｋ）よりも大きい限り、パケットを取り出すことができる。それぞれのオリジナルＷｅｂファイルは、それを構成する複数の要素パケットが取り出し可能である限り、復元できる。この場合、イレージャ・コーディングを使用せずに、各オリジナル・メッセージを取り出して、パケットを再作成することができる。しかし、イレージャ・コーディングを使用すると、ｋをオリジナル・メッセージの個数とした場合、ｋ個のはっきりと識別可能な符号化メッセージのみを取り出して、パケットを再作成する。上述のように、リード・ソロモン・コーデックを使用するイレージャ・コーディングが適用され、パラメータｋが１６に設定された場合、１６個のはっきりと識別可能な符号化メッセージが、オリジナル・パケットの復元を可能にする。

【００５４】

取り出しプロセス６００は、ブラウザが要求を送出したときに開始する。その要求の中で、ホストが、そのホストの下部の相対アドレスを加えて、識別される。例えば、典型的な要求は、＜ｄｗｅｂ＞ｗｗｗ．ｘｙｚ．ｃｏｍのように表示される。ここに示すように、＜ｄｗｅｂ＞などのタグは、ホスト・アドレスと一緒にされる。このタグは、その後に続くホスト・アドレスが分散型Ｗｅｂサイトであることを示す。このようなタグがないと、ブラウザ要求を受け取ったプログラムは、分散型Ｗｅｂサイトと単一サーバによりホスティングされている通常のＷｅｂサイトとを区別できない場合がある。一実装形態では、このプログラムはプロキシとして実装される。この実装形態では、このプログラムはプロキシ・ポートを介して複数のピアと通信する。しかし、プログラムは、本発明の分散型Ｗｅｂホスティング・メカニズムの精神から逸脱することなく他の構成で実装することができる。例えば、このプログラムは、他のブラウザ・コンポーネントとともにインストールされるブラウザ内のコンポーネント（例えば、ツールバー）とすることができる。また、ブラウザのオプションの取り出しブロックとして実装することも可能である。任意の構成について、このプログラムにより実行される取り出しプロセスは、図６に例示されている流れ図に関連して詳細に説明される。

【００５５】

そこで、ブロック６０２で、www.xyz.com/personal/trip2004/picturel.jpgなどのＵＲＬ（Uniform Resource Locator）を受け入れる。このＵＲＬは、分散型Ｗｅｂコンデンサを含むものとして認識される。処理は、ブロック６０４に続く。

【００５６】

ブロック６０４で、このＵＲＬに関連付けられているＷｅｂコンテンツの一部分を保持するピアのリストが取得される。ＵＲＬは、ＧＵＩＤ（Global Unifier Identifier）などの大きな整数値に変換される。このＵＲＬを保持しているピアのリストは、ＧＵＩＤに関連付けられているピアのリストである。一実施形態では、ピアのリストを決定することは、ルート・パスを指定する１つの列（カラム）とＧＵＩＤを指定する第２の列（カラム）を持つローカルのＧＵＩＤテーブルをチェックすることにより行われる。このローカルＧＵＩＤテーブルは、クライアントが分散型Ｗｅｂサイトにアクセスすると必ず作成され、更新されるようにできる。その後、Ｗｅｂコンテンツの一部分を格納しているコンピュータのリストを取得するため、ＧＵＩＤテーブル・リストがコンピュータ・クラスタ全体を通して送信される。他の実施形態では、ＧＵＩＤを保持するピアのリストは、分散型ハ

ッシュ・テーブル（ＤＨＴ）方式を使用してルックアップを実行することにより決定される。ＧＵＩＤに関連付けられているピアを識別するＤＨＴ手法は、当業ではよく知られている。処理は、ブロック６０６に続く。

【００５７】

ブロック６０６で、このＵＲＬに関連付けられているサマリ・ファイルが取得される。一般に、階層型Ｗｅｂサイトのディレクトリ毎にサマリ・ファイルが用意される。つまり、サマリ・ファイルは、Ｗｅｂサイトの構造についての提示（glimpse）を与える。サマリ・ファイルを使用することにより、クライアントは、Ｗｅｂページ／ファイルを構成するパケットの個数およびインデックスを決定し、そうして、目的のＷｅｂページ／ファイルに対する正しいパケットを取り出すことができる。要求されたファイルを取り出すために、その要求されたファイルが配置されているディレクトリに関連付けられているサマリ・ファイルを取得する。そこで、正しいサマリ・ファイルを特定するために、Ｗｅｂサイト階層がトラバースされる。サマリ・ファイルは、ファイル／子ディレクトリ毎に１エントリを含む。サマリ・ファイルの各エントリは、ファイル／子ディレクトリの名前を識別し、変更プロパティ、ファイル／フォルダ識別子、およびファイルの長さを含む。さらに、各エントリは、Ｗｅｂページ／ファイルを構成するパケットを識別するイレージャコーディング・パケット識別子を含むことができる。変更プロパティは、関連するファイル／子ディレクトリが最後に更新された時点のタイム・スタンプである。変更プロパティがピア毎に異なる場合、古いタイム・スタンプを持つピアに格納されているアイテムは取り出されない。最初に、サマリ・ファイルがチェックされ、ローカルに置かれているかどうかを調べられる。サマリ・ファイルは、このディレクトリの下のファイルのどれかがすでにクライアントによりアクセスされていた場合にローカルに置かれる。すでにアクセスされていた場合、サマリ・ファイルはピアから取り出す必要はない。アクセスされていなかった場合、サマリ・ファイルは、ピアの１つまたは複数から取り出され、後続の要求のためローカルのクライアントに格納される。したがって、漸進的なキャッシュ・プロセスがある。サマリ・ファイル自体は、複数のパケットに分割され、イレージャ符号化されたメッセージに符号化されるようにできる。取り出し時に、クライアントは、サマリ・ファイルに関連付けられているイレージャ符号化されたメッセージを取り出し、この符号化メッセージを複数のパケットに復号化し、サマリ・ファイルを組み立て直す。サマリ・ファイルが利用可能になると、処理はブロック６０８に続く。

【００５８】

ブロック６０８で、複数のピアから取得されたサマリ・ファイルが検証される。Ｗｅｂページ／ファイルの古い（つまり、以前の）バージョンを保持していることをサマリ・ファイルが示しているピアは、取り出しプロセスには関わらない。サマリ・ファイルが検証された後、処理はブロック６１０に続く。

【００５９】

ブロック６１０で、ネゴシエーション・プロセスが利用可能なピアにより実行される。ネゴシエーション・プロセスを使用して、Ｗｅｂページ／ファイルおよび上述のサマリ・ファイルの両方を取り出す。一実施形態では、ネゴシエーション・プロセスにより、利用可能なピアから実際のＷｅｂサイト・ファイルを取得する。他の実施形態では、ネゴシエーション・プロセスにより、利用可能なピアからＷｅｂサイト・ファイルを構成するパケットを取得する。さらに他の実施形態では、ネゴシエーション・プロセスにより、利用可能なピアからメッセージまたは符号化メッセージを取得する。この符号化メッセージは、その後、パケットに復号化され、パケットはさらに組み立てられ、オリジナルのＷｅｂページ／ファイルが再作成される。

【００６０】

一時的に、図７を参照すると、利用可能なピアから欠落していたファイルまたはメッセージを取得する例示的なネゴシエーション・プロセス７００を示す、シーケンシャルな流れ図が説明されている。図７の「アイテム」という用語は、要求された情報がＷｅｂサイト・ファイル、パケット、メッセージ、または符号化メッセージであることを示すために

使用されている。左のラインは、クライアント（つまり、Ｗｅｂコンテンツを見たい、分散型ＷｅｂサイトのＵＲＬを送信した、コンピューティング・デバイス）が受信または送信した通信を表す。右のラインは、ピアの１つが受信または送信した通信を表す。ネゴシエーション・プロセス７００は、必要なアイテムが取り出されるまで識別されたピア毎に実行することができる。

【００６１】

クライアントは、プロキシを介して、どのアイテムが必要かを通知する（７０２）。それらのアイテムが符号化メッセージの場合、クライアントはパケット識別子およびキーを供給する。それに応じて、ピアは、要求された、欠落していたアイテムに関連付けられている利用可能なアイテムに関する情報を提供する。再び、それらのアイテムが符号化メッセージであれば、ピアは、ピア上にローカルで格納されている符号化メッセージのパケット識別子およびキーを供給する。７０６で、クライアントは、特定の欠落していたアイテム、即ち通知されたピアがそのアイテムの利用が可能であると通知した特定の欠落アイテム、を求める要求を送信する。７０８で、そのピアは、特定の欠落していたアイテムをクライアントに送信する。

【００６２】

図６に戻ると、ネゴシエーション・プロセスで欠落していたアイテムが取得された後、処理はブロック６１２に続く。ブロック６１２で、情報は組み立てられ、関連するＷｅｂページが表示される。情報を組み立てることには、符号化メッセージをパケットに復号化し、その後、それらのパケットをＵＲＬに関連する特定のＷｅｂファイルに組み立てることを必要とする。

【００６３】

このようにして、図６および７に関して上で説明したように、１つのＷｅｂファイルの異なる部分が、異なる場所で実行している複数のコンピューティング・デバイスから、取得されることが可能になる。通常、どのコンピューティング・デバイスも、Ｗｅｂファイルの完全なコピーを持たない。このことは、特にイレージャ・コーディングが実行される場合、特に当てはまる。したがって、クライアントは、複数の部分を取り出して、それらを１つにまとめ上げて要求されたＷｅｂファイルを作成し、Ｗｅｂファイルをブラウザに表示するする役目を負う。

【００６４】

本発明の分散型ホスティング・メカニズムに関する実験が実行された。これらの実験では、３つのシナリオ、１）複数のピア上にＷｅｂサイト全体を複製すること、２）イレージャ・コーディングを使用せずに複数のピア上にＷｅｂサイトの一部分を複製すること、および３）イレージャ・コーディングを使用して複数のピア上にＷｅｂサイトの一部分を複製すること、をテストした。図８および９は、上の３つの特定のシナリオについて、それぞれ、テスト結果８０１、８０２、８０３、ならびに９０１、９０２、および９０３を例示している。シナリオ２については、それぞれのパケットはｋ個の断片に分割される。しかし、複製段階では、オリジナルのメッセージ断片が、イレージャ・コーディング無しで、他のピアに送信される。すべてのシナリオについて、コンテンツ（つまり、集計されたコンテンツ複製比Ｃ）を分配しホスティングするために同じ量のネットワークおよび記憶資源が使用されると仮定した。もう１つの仮定は、コンピュータ・クラスタ内のピアのそれぞれが、同一の提供帯域幅を持ち、オンラインになっている確率として独立した値を持ってクライアントにサービスを提供するというものであった。コンピュータ・クラスタ（例えば、Ｐ２Ｐネットワーク）内のＷｅｂサイトを取り出すのに成功する確率は、図８に示されている。様々なシナリオを使用した際のＷｅｂサイトを取り出すための平均速度増大は、図９に示されている。図８および９の両方について、横軸は、ピアがオンラインである確率である。コンテンツ分配のパラメータはＣ＝８（コンテンツの８コピーがホスティングされている）、ｋ＝１６に設定された。次に、これらの図のそれぞれについて詳述する。

【００６５】

図８は、様々なシナリオにおいて、オンラインになる確率に対する取り出し失敗率、に関連するテスト結果を例示するグラフである。上述のように、曲線８０１は、Ｗｅｂサイト全体が複製されるシナリオ１を表し、曲線８０２は、Ｗｅｂサイト全体のうちの一部分がイレージャ・コーディング無しで複製されるシナリオ２を表し、曲線８０３は、Ｗｅｂサイト全体のうちの一部分がイレージャ・コーディングを使用して複製されるシナリオ３を表す。曲線８０３では、同じ量のネットワークおよび記憶資源を使用する曲線８０１および８０２との比較で、取り出しの信頼性が著しく向上していることに留意されたい。実際、ピアがオンラインである確率が０．１３よりも大きいと、イレージャ符号化されたコンテンツに関するＷｅｂサイトの取り出し失敗率は、Ｗｅｂ複製曲線８０１全体、およびイレージャ・コーディング無しの部分的Ｗｅｂ複製曲線８０２、と比べて数千倍小さかった。

【００６６】

図９は、様々なシナリオにおいて、オンラインになる確率に対する取り出し速度向上、に関連するテスト結果を例示するグラフである。上述のように、曲線９０１は、Ｗｅｂサイト全体が複製されるシナリオ１を表し、曲線９０２は、Ｗｅｂサイト全体のうちの一部分がイレージャ・コーディング無しで複製されるシナリオ２を表し、曲線９０３は、Ｗｅｂサイト全体のうちの一部分がイレージャ・コーディングを使用して複製されるシナリオ３を表す。イレージャ符号化曲線９０３は、それぞれ全Ｗｅｂ複製（曲線９０１）およびイレージャ・コーディング無しの部分的Ｗｅｂ複製（曲線９０２）と比べて、１６倍および１～１０倍、取り出し速度が高速化されることに注目されたい。

【００６７】

イレージャ符号化コンテンツ分配、および不均等な重み割り当てを使用する階層型コンテンツ編成によるＰ２Ｐ　Ｗｅｂホスティングシステムが設計された。Ｗｅｂサイトは、７つのピア上に複製された。オリジナルのＷｅｂサイトでは、２２８メガバイトを消費した。複製時に、それぞれのピアは、そのＷｅｂサイトのうちの６０メガバイト分をホスティングすることに同意しており、その結果、平均複製比は０．２６となった。Ｗｅｂファイルの重み付けは不等なので、実際のＷｅｂファイルに対するピア複製比は０．２５から１．０と変動する。Ｗｅｂページ時に、クライアントは、７つのピアから同時にＷｅｂコンテンツを取り出し、イレージャ符号化メッセージを復号化し、Ｗｅｂページをレンダリングした。

【００６８】

したがって、説明したように、本発明の分散型ホスティング・メカニズムは、Ｗｅｂコンテンツにアクセスする取り出し速度を向上させ、またコンテンツを取り出す信頼性を向上させる。さらに、分散型ホスティング・メカニズムは、それぞれの個別ピア上に複製されるコンテンツの量を低減する。そのため、ピアでは、記憶領域の追加または帯域幅の増大に関わる膨大な費用は発生しない。本発明のＷｅｂホスティング・メカニズムは、ホスト・コンポーネント、ピア・コンポーネント、およびクライアント・コンポーネントの３つのコンポーネントを含む。ホスト・コンポーネントは、Ｗｅｂページ／ファイルの一部分（つまり、部分集合）をピアに分配する。ピア・コンポーネントは、Ｗｅｂページ／ファイルの分配された部分を受け取り、その後、クライアントから要求があったら再分配する。クライアント・コンポーネントでは、複数のピアからオリジナル・メッセージまたはイレージャ符号化メッセージの形でＷｅｂページ／ファイルの一部分を取り出す作業を調整する。その後、オリジナル・メッセージは組み立てられ、要求されたＷｅｂページ／ファイルを形成する。符号化メッセージはパケットにイレージャ復号化され、パケットはさらに組み立てられ、要求されたＷｅｂページ／ファイルが形成される。

【００６９】

本明細書全体を通して、「一実施形態」、または「一実施例」と記述されている場合、これは、特定の説明されている機能、構造、または特性が本発明の少なくとも一実施形態に含まれることを意味する。したがって、このようなフレーズを使用した場合、１つの実施形態というよりも複数の実施形態を指している。さらに、説明されている機能、構造、

または特性は、１つまたは複数の実施形態において、適当な方法により組み合わせることができる。

【００７０】

ただし、当業者であれば、本発明は、特定の詳細の１つまたは複数を使用せずに、または他の方法、資源、材料などを使用して、実践できることは理解できるであろう。一方、よく知られている構造、資源、またはオペレーションについては、単に本発明の態様をわかりにくくすることを避けるために、詳細に示したり、あるいは説明したりしてはいない。

【００７１】

複数の実施例およびアプリケーション例を例示し、説明したが、本発明は、上述の正確な構成および資源に限定されないことが理解されるであろう。請求の範囲から逸脱することなく、本明細書に開示されている本発明の方法およびシステムの配置、オペレーション、および詳細に対し、様々な修正、変更、および変形を加えられることは、当業者には明白なことであろう。

【図面の簡単な説明】

【００７２】

【図１】本明細書で説明されている手法およびメカニズムを実装するために使用することができるコンピューティング・デバイスの図である。

【図２】図２に示されている２つ以上のコンピュータデバイスが本明細書で説明されている分散型Ｗｅｂホスティング・メカニズムを実装するように構成されているネットワークの図である。

【図３】複製されたＷｅｂコンテンツの部分集合を図２に示されている複数のコンピューティング・デバイスのうちの１つに分配する分配プロセスを例示する流れ図である。

【図４】図３に示されている分配プロセスで使用するのに好適なＷｅｂコンテンツにサイト重みを割り当てることを例示するツリー図である。

【図５】図３に示されている分配プロセスで使用するのに好適なメッセージにＷｅｂファイルを変換することを図形で例示するブロック図である。

【図６】図２に示されている複数のコンピューティング・デバイスのうちの１つからＷｅｂコンテンツの部分集合を取り出す取り出しプロセスを例示する流れ図である。

【図７】図６に示されている取り出しプロセスで使用するのに好適な欠落している項目を取得するネゴシエーション・プロセスを例示する逐次的流れ図である。

【図８】様々なシナリオにおいて取り出し失敗率対オンラインになる確率に関連するテスト結果を例示するグラフである。

【図９】本発明の分散型Ｗｅｂホスティング・メカニズムにより、様々なシナリオにおいて取り出し速度向上対オンラインになる確率に関連するテスト結果を例示するグラフである。

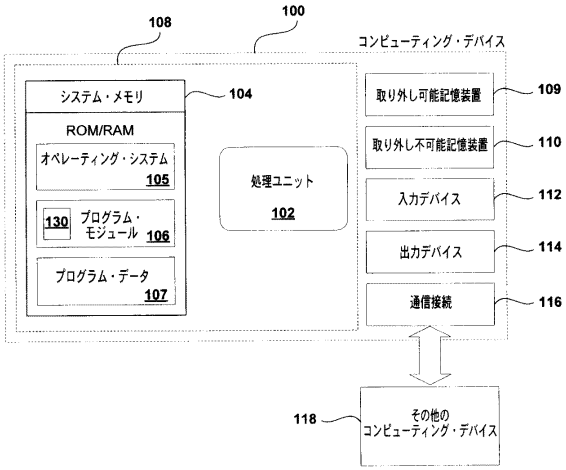
【符号の説明】

【００７３】

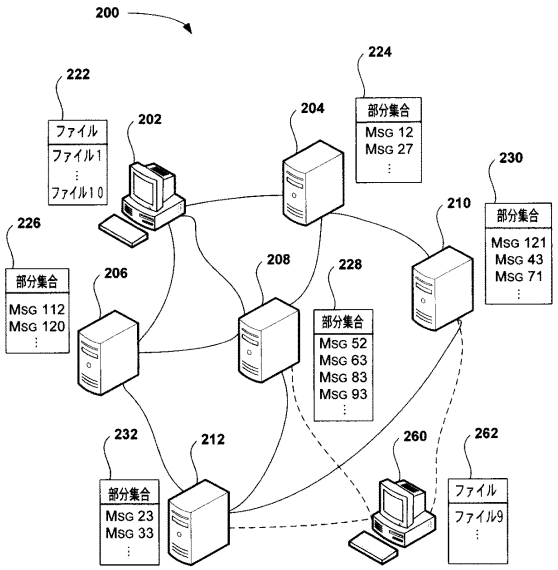
１００　コンピューティング・デバイス
１０２　処理ユニット
１０４　システム・メモリ
１０５　オペレーティング・システム
１０６　プログラム・モジュール
１０７　プログラム・データ
１０８　基本構成
１０９　取り外し可能記憶装置
１１０　取り外し不可能記憶装置
１１２　入力デバイス
１１４　出力デバイス
１１６　通信接続

１１８　その他のコンピューティング・デバイス
２００　ネットワーク（コンピュータ・クラスタ）
２０２～２１２　コンピューティング・デバイス
２６０　コンピューティング・デバイス
８０１、８０２、８０３　テスト結果
９０１、９０２、９０３　テスト結果

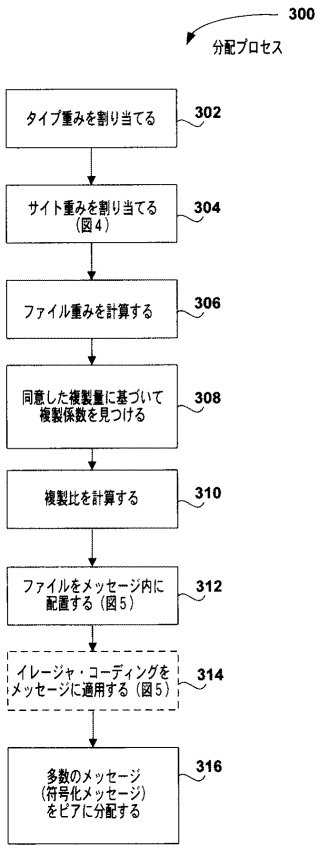
【図１】



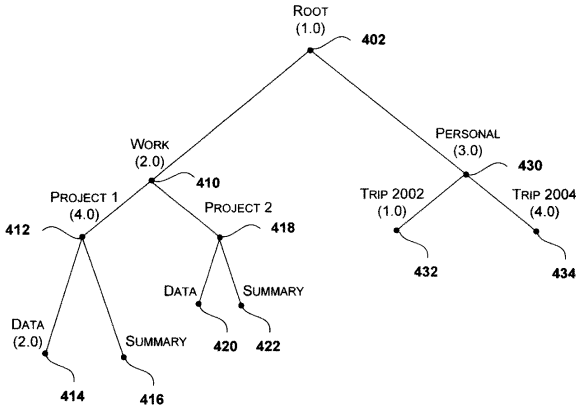
【図２】



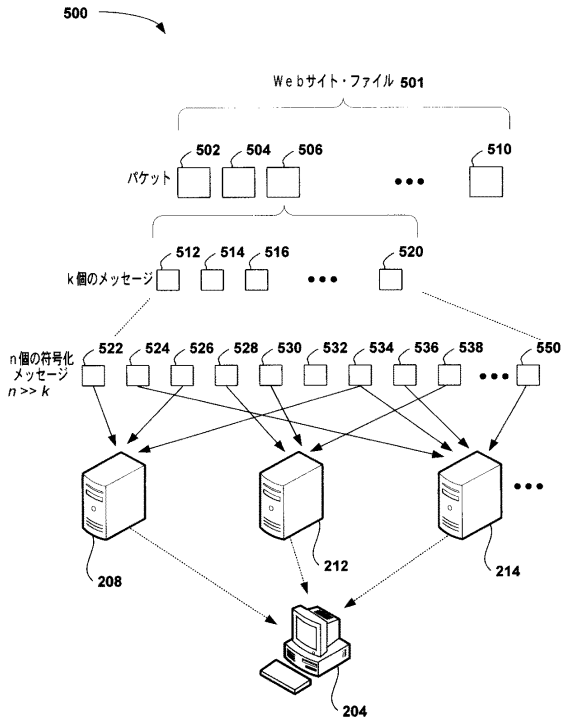
【図３】


300
分配プロセス
タイプ重みを割り当てる
302
サイト重みを割り当てる（図４）
304
ファイル重みを計算する
306
同意した複製量に基づいて複製係数を見つける
308
複製比を計算する
310
ファイルをメッセージ内に配置する（図５）
312
イレージャ・コーディングをメッセージに適用する（図５）
314
多数のメッセージ（符号化メッセージ）をピアに分配する
316

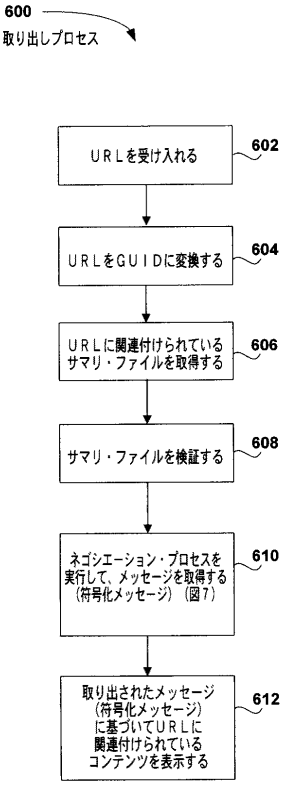
【図４】


ROOT (1.0)
402
WORK (2.0)
410
PERSONAL (3.0)
430
PROJECT 1 (4.0)
412
PROJECT 2
418
TRIP 2002 (1.0)
TRIP 2004 (4.0)
DATA (2.0)
SUMMARY
DATA
SUMMARY
414
416
420
422
432
434

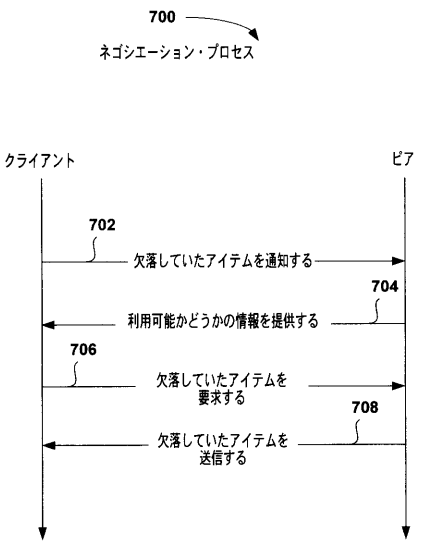
【図５】


500
Webサイト・ファイル 501
パケット
502 504 506 510
k個のメッセージ
512 514 516 520
n個の符号化メッセージ
n >> k
522 524 526 528 530 532 534 536 538 550
208
212
214
204

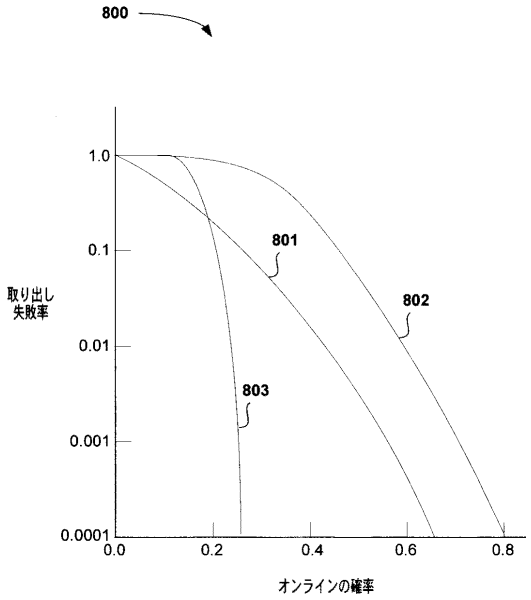
【図６】


600
取り出しプロセス
URLを受け入れる
602
URLをGUIDに変換する
604
URLに関連付けられているサマリ・ファイルを取得する
606
サマリ・ファイルを検証する
608
ネゴシエーション・プロセスを実行して、メッセージを取得する（符号化メッセージ）（図７）
610
取り出されたメッセージ（符号化メッセージ）に基づいてURLに関連付けられているコンテンツを表示する
612

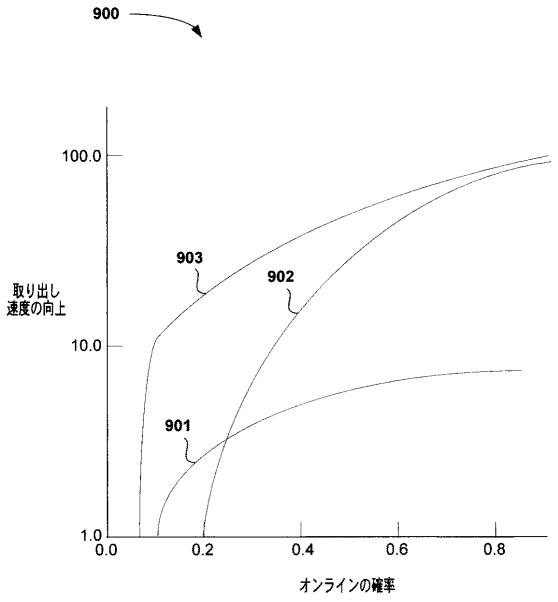
【図７】


700
ネゴシエーション・プロセス
クライアント
ピア
702
欠落していたアイテムを通知する
704
利用可能かどうかの情報を提供する
706
欠落していたアイテムを
要求する
708
欠落していたアイテムを
送信する

【図８】


800
1.0
0.1
0.01
0.001
0.0001
取り出し
失敗率
801
802
803
0.0
0.2
0.4
0.6
0.8
オンラインの確率

【図９】


900
100.0
10.0
1.0
取り出し
速度の向上
903
902
901
0.0
0.2
0.4
0.6
0.8
オンラインの確率

フロントページの続き

(72)発明者　ジン　リ
アメリカ合衆国　９８０５２　ワシントン州　レッドモンド　ワン　マイクロソフト　ウェイ　マイクロソフト　コーポレーション内

審査官　鈴村　理絵子

(56)参考文献　特開２００２－０３２２８０（ＪＰ，Ａ）
特開２００３－２１６５２１（ＪＰ，Ａ）
特開平１０－０９３４４６（ＪＰ，Ａ）
特開２００２－２７８８５９（ＪＰ，Ａ）
国際公開第０１／０９０９４３（WO，Ａ１）
特表２００２－５２０７３５（ＪＰ，Ａ）
特開２００３－０６７２７９（ＪＰ，Ａ）
国際公開第０３／０７１８００（WO，Ａ１）
特開２００２－０６３０６４（ＪＰ，Ａ）
特開２００１－３３３０３２（ＪＰ，Ａ）
特開平１０－１９８５９０（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１３／００
Ｇ０６Ｆ　１５／００