(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4486380号
(P4486380)
(45)発行日　平成22年6月23日(2010.6.23)　(24)登録日　平成22年4月2日(2010.4.2)

(51)Int.Cl.　　F I
G06Q 50/00 (2006.01)　　G06F 17/60 142
G06Q 30/00 (2006.01)　　G06F 17/60 302E

請求項の数 14　(全 43 頁)

(21)出願番号　特願2004-52372(P2004-52372)
(22)出願日　平成16年2月26日(2004.2.26)
(65)公開番号　特開2004-259283(P2004-259283A)
(43)公開日　平成16年9月16日(2004.9.16)
審査請求日　平成19年2月26日(2007.2.26)
(31)優先権主張番号　10/374,321
(32)優先日　平成15年2月26日(2003.2.26)
(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　ピーター　デビッド　ワックスマン
アメリカ合衆国　９８００４　ワシントン州　ベルビュー　ノースイースト　２８　プレイス　１００８

最終頁に続く

(54)【発明の名称】クロスフォレストディレクトリ情報に基づくコンテンツのディジタル権利管理（ＤＲＭ）ライセンスの発行

(57)【特許請求の範囲】
【請求項１】
組織からのユーザが前記組織内で事前に定義されたグループのメンバであるかどうかを判定する方法であって、前記グループは、１つ又は複数の前記グループを指名し、各グループに関して対応する権利のセットを指定する署名付き権利ラベルにおいて識別され、前記組織は、前記組織を論理的な形で分割した少なくともフォレストＡおよびフォレストＢを含むコンピュータネットワークを維持し、前記フォレストＡは、ディレクトリＡおよびディレクトリＡを照会することができる照会サーバＡを有し、前記フォレストＢは、ディレクトリＢおよびディレクトリＢを照会することができる照会サーバＢを有し、前記グループは前記フォレストＡ又は前記フォレストＢのどちらかに属し、前記方法は、
前記照会サーバＡが、前記フォレストＡ内のデジタルコンテンツの対応する部分をレンダリングするためのデジタルライセンスのために前記フォレストＡ内でユーザからの要求を受け取るステップであって、前記要求は、前記ユーザ識別及び前記グループ識別を含む、ステップと、
前記照会サーバＡが、前記グループのメンバシップについての情報を要求し、前記情報に応じて前記照会サーバＡが前記フォレストＢに方向付けされるために前記フォレストＡの前記ディレクトリＡを照会するステップであって、前記方向付けは、前記グループが前記フォレストＡに属さず、前記フォレストＢに属すことを示す、ステップと、
前記照会サーバＡが、前記フォレストＢの前記照会サーバＢと連絡し、前記グループが現在存在しているかどうかについて前記ディレクトリＢを照会するように前記照会サーバ

Ｂに要求するステップであって、前記グループが現在存在している場合、前記ユーザは前記フォレストＢにおいて前記グループのメンバであり、前記グループが存在していない場合、前記ユーザは前記フォレストＢにおいて前記グループのメンバでない、ステップと、
　前記照会サーバＡが、前記ユーザが実際に前記フォレストＢにおいて前記グループのメンバであるかどうかを照会サーバＢから受け取るステップと、
　前記照会サーバＡが、前記ユーザが前記フォレストＢにおいて前記グループのメンバであるかどうかに少なくとも部分的に基づいて前記フォレストＡ内の前記ユーザからの前記フォレストＡ内の前記要求を許可するステップと
　を含むことを特徴とする方法。
【請求項２】
　前記照会サーバＡが、前記フォレストＡの前記ディレクトリＡを照会するステップは、
　前記照会サーバＡが、前記グループと関連するすべてのオブジェクトを返すために前記フォレストＡの前記ディレクトリＡを照会するステップであって、前記グループは、前記フォレストＢの前記ディレクトリＢが前記グループに対応するレコードオブジェクトを有するように前記フォレストＢに属し、前記ディレクトリＢ内の前記グループに関する前記レコードオブジェクトは前記グループのすべての直接のメンバを含み、前記フォレストＡの前記ディレクトリＡは前記グループに対応するポインタオブジェクトを有し、前記ディレクトリＡ内の前記グループに関する前記ポインタオブジェクトは前記フォレストＢのアドレスを含む、ステップを含み、
　前記照会サーバＡが前記フォレストＢの前記照会サーバＢに連絡するステップは、
　前記照会サーバＡが前記グループに関する前記ポインタオブジェクトを前記ディレクトリＡから受け取るステップであって、前記グループが前記フォレストＡに属していないことを示す、ステップと、
　前記照会サーバＡが、受け取られた前記ポインタオブジェクトから前記フォレストＢの前記アドレスを検索するステップと、
　前記照会サーバＡが、前記フォレストＢの前記照会サーバＢのアドレスについての前記フォレストＢの前記ディレクトリＢを照会するために検索された前記アドレスを使用するステップと、
　前記照会サーバＡが、前記ディレクトリＢから前記照会サーバＢの前記アドレスを受け取るステップと、
　前記照会サーバＡが、前記照会サーバＢに前記アドレスで連絡し、前記ユーザが前記フォレストＢにおいて前記グループのメンバであるかどうかについて前記ディレクトリＢを照会するように前記照会サーバＢに要求するステップと
　を含むことを特徴とする請求項１に記載の方法。
【請求項３】
　前記照会サーバＢが、前記グループに関するすべてのオブジェクトを返すために前記フォレストＢの前記ディレクトリＢに照会するステップであって、前記グループは、前記フォレストＢの前記ディレクトリＢが前記グループに対応するレコードオブジェクトを有するように前記フォレストＢに属し、前記ディレクトリＢ内の前記グループに関する前記レコードオブジェクトは前記グループのすべての直接のメンバを含む、ステップと、
　前記照会サーバＢが、前記ディレクトリＢから前記グループに関する前記レコードオブジェクトを受け取るステップであって、前記グループは前記フォレストＢに属すことを示す、ステップと、
　前記照会サーバＢが、前記ユーザが前記グループの直接のメンバであるかどうかを判定するために受け取られた前記レコードオブジェクトで示される前記グループの前記直接のメンバを再検討するステップと
　によって前記ユーザが前記グループのメンバであるかどうかを判定するために前記照会サーバＢが前記フォレストＢの前記ディレクトリＢを照会するステップをさらに含むことを特徴とする請求項１に記載の方法。
【請求項４】

前記ユーザが前記グループの直接のメンバでないと判定された場合、
前記照会サーバＢが、前記ユーザに関するすべてのオブジェクトを返すために前記ディレクトリＢを照会し、前記ディレクトリＢ内で前記ユーザから前記グループへのメンバシップパスを見つけることを試みることによって前記ユーザが前記グループの間接のメンバであるかどうかの判定に進むステップであって、前記ユーザから前記グループへの見つかったメンバシップパスは、前記ユーザが実際に前記グループのメンバであることを確立するステップと、
前記照会サーバＢが、メンバシップパスが見つかったかどうかに基づいて前記フォレストＡ内の前記ユーザが実際に前記グループのメンバであるかどうかを前記照会サーバＡに報告するステップと
をさらに含むことを特徴とする請求項３に記載の方法。
【請求項５】
前記照会サーバＢはＤＲＭサーバであり、さらに、前記照会サーバＡはＤＲＭサーバであり、前記照会サーバＡの識別を前記照会サーバＢに送り、前記ユーザが前記グループのメンバであるかどうかについて知る資格を前記照会サーバＡが有することを、送られた前記識別に基づいて前記照会サーバＢが満足するステップを含むことを特徴とする請求項３に記載の方法。
【請求項６】
前記照会サーバＡはディジタル証明書および照会するエンティティによって認識されるオーソリティの信頼されるルートにつながる証明書チェーンと共に前記照会サーバＢに送り、前記照会サーバＢは前記証明書を検証するステップを含むことを特徴とする請求項５に記載の方法。
【請求項７】
組織からのユーザが前記組織内で事前に定義されたグループのメンバであるかどうかを判定する方法を実行するコンピュータ実行可能命令が保管されたコンピュータ読み取り可能記録媒体であって、前記グループは、１つ又は複数の前記グループを指名し、各グループに関して対応する権利のセットを指定する署名付き権利ラベルで識別され、前記組織は、前記組織を論理的な形で分割した少なくともフォレストＡおよびフォレストＢを含むコンピュータネットワークを維持し、前記フォレストＡは、ディレクトリＡおよびディレクトリＡを照会することができる照会サーバＡを有し、前記フォレストＢは、ディレクトリＢおよびディレクトリＢを照会することができる照会サーバＢを有し、前記グループは前記フォレストＡ又は前記フォレストＢのどちらかに属し、前記方法は、
前記照会サーバＡが、前記フォレストＡ内のデジタルコンテンツの対応する部分をレンダリングするためのデジタルライセンスのために前記フォレストＡ内でユーザからの要求を受け取るステップであって、前記要求は、前記ユーザ識別及び前記グループ識別を含む、ステップと、
前記照会サーバＡが、前記グループのメンバシップについての情報を要求し、前記情報に応じて前記照会サーバＡが前記フォレストＢに方向付けされるために前記フォレストＡの前記ディレクトリＡを照会するステップであって、前記方向付けは、前記グループが前記フォレストＡに属さず、前記フォレストＢに属すことを示す、ステップと、
前記照会サーバＡが、前記フォレストＢの前記照会サーバＢと連絡し、前記グループが現在存在しているかどうかについて前記ディレクトリＢを照会するように前記照会サーバＢに要求するステップであって、前記グループが現在存在している場合、前記ユーザは前記フォレストＢにおいて前記グループのメンバであり、前記グループが存在していない場合、前記ユーザは前記フォレストＢにおいて前記グループのメンバでない、ステップと、
前記照会サーバＡが、前記ユーザが実際に前記フォレストＢにおいて前記グループのメンバであるかどうかを前記照会サーバＢから受け取るステップと、
前記照会サーバＡが、前記ユーザが前記フォレストＢにおいて前記グループのメンバであるかどうかに少なくとも部分的に基づいて前記フォレストＡ内の前記ユーザからの前記フォレストＡ内の前記要求を許可するステップと

を含むことを特徴とするコンピュータ読み取り可能記録媒体。
【請求項８】
前記照会サーバＡが、前記フォレストＡの前記ディレクトリＡを照会するステップは、
前記照会サーバＡが、前記グループと関連するすべてのオブジェクトを返すために前記フォレストＡの前記ディレクトリＡを照会するステップであって、前記グループは、前記フォレストＢの前記ディレクトリＢが前記グループに対応するレコードオブジェクトを有するように前記フォレストＢに属し、前記ディレクトリＢ内の前記グループに関する前記レコードオブジェクトは前記グループのすべての直接のメンバを含み、前記フォレストＡの前記ディレクトリＡは前記グループに対応するポインタオブジェクトを有し、前記ディレクトリＡ内の前記グループに関する前記ポインタオブジェクトは前記フォレストＢのアドレスを含む、ステップを含み、
前記照会サーバＡが前記フォレストＢの前記照会サーバＢに連絡するステップは、
前記照会サーバＡが前記グループに関する前記ポインタオブジェクトを前記ディレクトリＡから受け取るステップであって、前記グループが前記フォレストＡに属していないことを示す、ステップと、
前記照会サーバＡが、受け取られた前記ポインタオブジェクトから前記フォレストＢの前記アドレスを検索するステップと、
前記照会サーバＡが、前記フォレストＢの前記照会サーバＢのアドレスについての前記フォレストＢの前記ディレクトリＢを照会するために検索された前記アドレスを使用するステップと、
前記照会サーバＡが、前記ディレクトリＢから前記照会サーバＢの前記アドレスを受け取るステップと、
前記照会サーバＡが、前記照会サーバＢに前記アドレスで連絡し、前記ユーザが前記フォレストＢにおいて前記グループのメンバであるかどうかについて前記ディレクトリＢを照会するように前記照会サーバＢに要求するステップと
を含むことを特徴とする請求項７に記載のコンピュータ読み取り可能記録媒体。
【請求項９】
前記照会サーバＢが、前記グループに関するすべてのオブジェクトを返すために前記フォレストＢの前記ディレクトリＢに照会するステップであって、前記グループは、前記フォレストＢの前記ディレクトリＢが前記グループに対応するレコードオブジェクトを有するように前記フォレストＢに属し、前記ディレクトリＢ内の前記グループに関する前記レコードオブジェクトは前記グループのすべての直接のメンバを含む、ステップと、
前記照会サーバＢが、前記ディレクトリＢから前記グループに関する前記レコードオブジェクトを受け取るステップであって、前記グループは前記フォレストＢに属すことを示す、ステップと、
前記照会サーバＢが、前記ユーザが前記グループの直接のメンバであるかどうかを判定するために受け取られた前記レコードオブジェクトで示される前記グループの前記直接のメンバを再検討するステップと
によって前記ユーザが前記グループのメンバであるかどうかを判定するために前記照会サーバＢが前記フォレストＢのディレクトリＢを照会するステップを含む方法を実行するコンピュータ実行可能命令が保管されたもう１つのコンピュータ読み取り可能記録媒体と組み合わされることを特徴とする請求項７に記載のコンピュータ読み取り可能記録媒体。
【請求項１０】
前記ユーザが前記グループの直接のメンバでないと判定された場合、前記方法は、
前記照会サーバＢが、前記ユーザに関するすべてのオブジェクトを返すために前記ディレクトリＢを照会し、前記ディレクトリＢ内で前記ユーザから前記グループへのメンバシップパスを見つけることを試みることによって前記ユーザが前記グループの間接のメンバであるかどうかの判定に進むステップであって、前記ユーザから前記グループへの見つかったメンバシップパスは、前記ユーザが実際に前記グループのメンバであることを確立するステップと、

前記照会サーバＢが、メンバシップパスが見つかったかどうかに基づいて前記フォレストＡ内の前記ユーザが実際に前記グループのメンバであるかどうかを前記照会サーバＡに報告するステップと

をさらに含むことを特徴とする請求項９に記載のもう１つのコンピュータ読み取り可能記録媒体。

【請求項１１】

前記照会サーバＢはＤＲＭサーバであり、さらに、前記照会サーバＡはＤＲＭサーバであり、前記方法は、前記照会サーバＡの識別を前記照会サーバＢに送り、前記ユーザが前記グループのメンバであるかどうかについて知る資格を前記照会サーバＡが有することを、送られた前記識別に基づいて前記照会サーバＢが満足するステップを含むことを特徴とする請求項１０に記載のコンピュータ読み取り可能記録媒体。

【請求項１２】

前記照会サーバＢはＤＲＭサーバであり、さらに、前記照会サーバＡはＤＲＭサーバであり、前記方法は、前記照会サーバＡの識別を前記照会サーバＢに送り、前記ユーザが前記グループのメンバであるかどうかについて知る資格を前記照会サーバＡが有することを、送られた前記識別に基づいて前記照会サーバＢが満足するステップを含むことを特徴とする請求項１０に記載のもう１つのコンピュータ読み取り可能記録媒体。

【請求項１３】

前記コンピュータ読み取り可能記録媒体の前記方法は、前記照会サーバＡが、ディジタル証明書および照会するエンティティによって認識されるオーソリティの信頼されるルートにつながる証明書チェーンと共に前記照会サーバＢに送るステップを含むことを特徴とする請求項１１に記載のコンピュータ読み取り可能記録媒体。

【請求項１４】

前記もう１つのコンピュータ読み取り可能記録媒体の前記方法は、前記照会サーバＢが前記証明書を検証するステップを含むことを特徴とする請求項１２に記載のコンピュータ読み取り可能記録媒体。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、ディジタル権利管理（ｄｉｇｉｔａｌ　ｒｉｇｈｔｓ　ｍａｎａｇｅｍｅｎｔ、ＤＲＭ）システムに関する。具体的には、本発明は、オフィスまたは会社または類似物などの組織内のコンテンツのレンダリングおよび使用を、対応する使用条件またはライセンス条件に従って制約できるようにする、組織内のディジタルコンテンツのパブリッシュ（ｐｕｂｌｉｓｈ）へのＤＲＭシステムの使用に関する。さらに具体的には、本発明は、別のフォレストからのディレクトリから得たグループ情報に基づく、対応するディジタルコンテンツのディジタルライセンスの発行に関する。

【背景技術】

【０００２】

ディジタル権利管理およびディジタル権利実施は、ディジタルオーディオ、ディジタルビデオ、ディジタルテキスト、ディジタルデータ、ディジタルマルチメディアなどのディジタルコンテンツに関連して、そのようなディジタルコンテンツが１つまたは複数のユーザに配布される場合に非常に望ましい。ディジタルコンテンツは、たとえばテキストドキュメントなどの静的なディジタルコンテンツ、または生イベントのストリーミングオーディオ／ビデオなどのストリーミングされるディジタルコンテンツとすることができる。配布の通常の態様には、磁気（フロッピー（登録商標））ディスク、磁気テープ、光学（コンパクト）ディスク（ＣＤ）などの有形のデバイス、および電子掲示板、電子ネットワーク、インターネットなどの無形メディアが含まれる。ユーザによる受信の際に、そのようなユーザは、パーソナルコンピュータ上のメディアプレイヤまたは類似物などの適当なレンダリングデバイスの助けを得てディジタルコンテンツをレンダリングまたは「プレイ」する。

【０００３】
１つのシナリオでは、著作者、出版社、放送会社などのコンテンツオーナーまたは権利オーナーが、ライセンス料またはなんらかの他の報酬と交換で多数のユーザまたは受取人のそれぞれにそのようなディジタルコンテンツを配布することを望む。そのようなシナリオでは、コンテンツが、曲、曲のアルバム、ムービーなどである可能性があり、配布の目的が、ライセンス料を生成するためである。そのようなコンテンツオーナーは、選択肢を与えられた場合に、そのような配布されるディジタルコンテンツに関してユーザが行うことができることを制限することを望みたがる。たとえば、コンテンツオーナーは、少なくともコンテンツオーナーが第２ユーザからのライセンス料を拒否する形で、ユーザが第２ユーザへのそのようなコンテンツのコピーおよび再配布を行うことを制限したがる。
【０００４】
さらに、コンテンツオーナーは、異なるライセンス料で異なるタイプの使用ライセンスを購入する柔軟性をユーザに提供すると同時に、実際にどのタイプのライセンスが購入されたかに関してユーザを拘束することを望む可能性がある。たとえば、コンテンツオーナーは、配布されるディジタルコンテンツを、限られた回数だけ、あるトータル時間だけ、あるタイプのマシンでのみ、あるタイプのメディアプレイヤでのみ、あるタイプのユーザによってのみ、などでプレイできるようにすることを望む可能性がある。
【０００５】
もう１つのシナリオでは、組織の従業員またはメンバなどのコンテンツデベロッパが、そのようなディジタルコンテンツを、組織内の１つまたは複数の他の従業員またはメンバ、あるいは組織外の他の個人に配布することを望むが、他者がそのコンテンツをレンダリングできなくすることを望む。この場合に、そのコンテンツの配布は、ライセンス料またはなんらかの他の報酬と交換のブロードベースの配布ではなく、秘密の形または制限された形での組織ベースのコンテンツ共有により類似する。
【０００６】
したがって、そのようなシナリオにおいて、コンテンツを、オフィスセッティング内で交換することができるドキュメントプレゼンテーション、スプレッドシート、データベース、電子メール、または類似物などのドキュメントプレゼンテーション、スプレッドシート、データベース、電子メール、または類似物とすることができ、コンテンツデベロッパは、コンテンツが、組織またはオフィスセッティング内に留まり、たとえば競争者または反対者など、許可されない個人によってレンダリングされないことを保証することを望む可能性がある。やはり、そのようなコンテンツデベロッパは、受取人がそのように配布されるディジタルコンテンツに関して行うことができることを制限することを望む。たとえば、コンテンツオーナーは、少なくともコンテンツをレンダリングすることを許可されなければならない個人の範囲の外にコンテンツが露出される形で、ユーザがそのようなコンテンツをコピーし、第２ユーザに再配布することを制限したがる。
【０００７】
さらに、コンテンツデベロッパは、さまざまな受取人に異なるレベルのレンダリング権利を与えることを望む可能性がある。たとえば、コンテンツデベロッパは、あるクラスの個人に関してプロテクトされたディジタルコンテンツを表示可能にするが、印刷可能にせず、別のクラスの個人に関して表示可能であり印刷可能にできるようにすることを望む可能性がある。
【０００８】
しかし、どちらのシナリオにおいても、配布が行われた後に、そのようなコンテンツのオーナー／デベロッパは、ディジタルコンテンツに対する制御を、あるとしてもごくわずかだけ有する。これは、実用上すべてのパーソナルコンピュータに、そのようなディジタルコンテンツの正確なディジタルコピーを行い、そのような正確なディジタルコピーを書き込み可能磁気ディスクまたは書き込み可能光学ディスクにダウンロードし、またはインターネットなどのネットワークを介して任意の宛先へそのような正確なディジタルコピーを送信するのに必要なソフトウェアおよびハードウェアが含まれるという事実に鑑みて、

特に問題である。

【０００９】

もちろん、コンテンツが配布されるトランザクションの一部として、コンテンツのオーナー／デベロッパは、ディジタルコンテンツのユーザ／受取人が、歓迎されない形でそのようなディジタルコンテンツを再配布しないことを約束することを要求することができる。しかし、そのような約束は、簡単になされ、簡単に破られる。コンテンツのオーナー／デベロッパは、通常は暗号化および暗号化解除が用いられる、複数の既知のセキュリティデバイスのいずれかを介してそのような再配布を防ぐことを試みることができる。しかし、穏やかに決心したユーザが、暗号化されたディジタルコンテンツを暗号解除し、そのようなディジタルコンテンツを暗号化されない形態で保存し、その後、これを再配布することを防ぐことの可能性は非常に低い。

【発明の開示】

【発明が解決しようとする課題】

【００１０】

任意の形態のディジタルコンテンツの制御されたレンダリングまたはプレイを可能にし、そのような制御が、柔軟であり、そのようなディジタルコンテンツのコンテンツオーナー／デベロッパによって定義可能である、ディジタル権利管理（ＤＲＭ）およびディジタル権利実施アーキテクチャならびにディジタル権利管理（ＤＲＭ）およびディジタル権利実施方法の必要性が存在する。具体的に言うと、特に、ドキュメントが個人または個人のクラスの定義されたグループの間で共有される、オフィスまたは組織の環境または類似物での、そのような制御されたレンダリングを可能にし、促進するアーキテクチャの必要が存在する。さらに具体的に言うと、組織内の事前に定義されるディビジョンまたは「フォレスト」にわたりコンテンツをライセンスできるようにするアーキテクチャの必要性が存在する。

【課題を解決するための手段】

【００１１】

前述の必要性は、組織のユーザが、組織内の事前に定義されるグループのメンバであるかどうかが判定され、組織が、少なくともフォレストＡおよびフォレストＢを含むコンピュータネットワークを維持し、フォレストＡが、ディレクトリＡおよびディレクトリＡに照会することができる照会のエンティティＡを有し、フォレストＢが、ディレクトリＢおよびディレクトリＢに照会することができる照会のエンティティＢを有する、本発明によって少なくとも部分的に満足される。

【００１２】

この方法では、照会のエンティティＡが、ユーザから要求を受け取り、そのユーザがグループのメンバであるかどうかに少なくとも部分的に基づいて、要求を許可するかどうかを判断する。したがって、照会のエンティティＡは、グループに関する情報を返すためにフォレストＡのディレクトリＡに照会し、それに応答して、フォレストＢに向けられる。照会のエンティティＡは、したがって、フォレストＢのエンティティＢに照会し、照会のエンティティＢに、ユーザがグループのメンバであるかどうかをディレクトリＢに照会するように要求する。照会のエンティティＡは、それに応答して、照会のエンティティＢから、ユーザが実際にグループのメンバであるかどうかを受け取り、ユーザがグループのメンバであるかどうかに少なくとも部分的に基づいてユーザからの要求を許可する。

【００１３】

前述の要約ならびに以下の本発明の実施形態の詳細な説明は、添付図面と共に読まれる場合によりよく理解される。本発明の例示において、図面に、現在好ましい実施形態を示す。しかし、理解されるように、本発明は、図示の正確な配置および手段に限定されない。

【発明を実施するための最良の形態】

【００１４】

コンピュータ環境

図１および以下の説明は、本発明を実施することができる適切なコンピューティング環境の短い全般的な説明を提供することを意図されている。しかし、ハンドヘルドコンピューティングデバイス、ポータブルコンピューティングデバイス、およびすべての種類の他のコンピューティングデバイスが、本発明と共に使用されることを企図されていることを理解されたい。汎用コンピュータを下で説明するが、これは、１つの例にすぎず、本発明は、ネットワークサーバインターオペラビリティおよびネットワークサーバインタアクションを有するシンクライアントだけを必要とする。したがって、本発明は、非常に少ないクライアントリソースまたは最小限のクライアントリソースが関係するネットワークホステッドサービスの環境、たとえば、クライアントデバイスがワールドワイドウェブへのブラウザまたはインターフェースとしてのみ働くネットワーク化された環境で、実施することができる。

【００１５】

必要ではないが、本発明を、デベロッパによる使用のためのアプリケーションプログラミングインターフェース（ＡＰＩ）を介して実施するか、クライアントワークステーション、サーバ、または他のデバイスなどの１つまたは複数のコンピュータによって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全般的な文脈で説明されるネットワークブラウジングソフトウェアに含めるか、その両方を行うことができる。一般に、プログラムモジュールには、特定のタスクを実行するか、特定の抽象データ型を実施するルーチン、プログラム、オブジェクト、コンポーネント、データ構造、および類似物が含まれる。通常、プログラムモジュールの機能性は、さまざまな実施形態での望みに応じて、組み合わせるか分散することができる。さらに、当業者は、本発明を他のコンピュータシステム構成で実践することができることを諒解するであろう。本発明と共に使用するのに適する可能性がある他の周知のコンピューティングシステム、コンピューティング環境、および／またはコンピューティング構成には、パーソナルコンピュータ（ＰＣ）、自動預金払戻機、サーバコンピュータ、ハンドヘルドデバイス、ラップトップデバイス、マルチプロセッサシステム、マイクロプロセッサベースシステム、プログラマブル消費者電子製品、ネットワークＰＣ、ミニコンピュータ、メインフレームコンピュータ、および類似物が含まれるが、これに限定はされない。本発明は、通信ネットワークまたは他のデータ伝送メディアを介してリンクされるリモートプロセッシングデバイスによってタスクが実行される分散コンピューティング環境で実践することもできる。分散コンピューティング環境では、プログラムモジュールを、メモリストレージデバイスを含む、ローカルコンピュータストレージメディアおよびリモートコンピュータストレージメディアの両方に配置することができる。

【００１６】

したがって、図１に、本発明を実施することができる適切なコンピューティングシステム環境１００の例を示すが、上で明らかにしたように、コンピューティングシステム環境１００は、適切なコンピューティング環境の１つの例に過ぎず、本発明の使用の範囲または機能性に関する限定を提案することを意図されてはいない。コンピューティング環境１００を、例示的なオペレーティング環境１００に示されたコンポーネントのいずれかまたはその組合せに関する依存性または要件を有するものと解釈してもならない。

【００１７】

図１に関して、本発明を実施する例示的なシステムに、コンピュータ１１０の形態の汎用コンピューティングデバイスが含まれる。コンピュータ１１０のコンポーネントには、処理ユニット１２０、システムメモリ１３０、およびシステムバス１２１が含まれるがこれに限定はされず、システムバス１２１によって、システムメモリを含むさまざまなシステムコンポーネントが、処理ユニット１２０に結合される。システムバス１２１は、メモリバスまたはメモリコントローラ、周辺バス、およびさまざまなバスアーキテクチャのいずれかを使用するローカルバスを含む複数のタイプのバス構造のいずれかとすることができる。制限ではなく例として、そのようなアーキテクチャに、Ｉｎｄｕｓｔｒｙ　Ｓｔａｎｄａｒｄ　Ａｒｃｈｉｔｅｃｔｕｒｅ（ＩＳＡ）バス、マイクロチャネルアーキテクチ

ャ（ＭＣＡ）バス、Ｅｎｈａｎｃｅｄ　ＩＳＡ（ＥＩＳＡ）バス、Ｖｉｄｅｏ　Ｅｌｅｃｔｒｏｎｉｃｓ　Ｓｔａｎｄａｒｄｓ　Ａｓｓｏｃｉａｔｉｏｎ（ＶＥＳＡ）ローカルバス、およびＰｅｒｉｐｈｅｒａｌ　Ｃｏｍｐｏｎｅｎｔ　Ｉｎｔｅｒｃｏｎｎｅｃｔ（ＰＣＩ）バス（メザニンバスとも称する）が含まれる。

【００１８】

コンピュータ１１０には、通常は、さまざまなコンピュータ可読メディアが含まれる。コンピュータ可読メディアは、コンピュータ１１０によってアクセスすることができるすべての使用可能なメディアとすることができ、コンピュータ可読メディアには、揮発性メディアおよび不揮発性メディアと、取り外し可能メディアおよび取り外し不能メディアの両方が含まれる。制限ではなく例として、コンピュータ可読メディアに、コンピュータストレージメディアおよび通信メディアを含めることができる。コンピュータストレージメディアには、コンピュータ可読命令、データ構造、プログラムモジュール、または他のデータなどの情報のストレージに関する任意の方法またはテクノロジで実施される、揮発性および不揮発性の両方の、取り外し可能メディアおよび取り外し不能メディアが含まれる。コンピュータストレージメディアには、ＲＡＭ、ＲＯＭ、ＥＥＰＲＯＭ、フラッシュメモリ、または他のメモリテクノロジ、ＣＤＲＯＭ、ディジタル多用途ディスク（ＤＶＤ）、または他の光学ディスクストレージ、磁気カセット、磁気テープ、磁気ディスクストレージ、または他の磁気ストレージデバイス、あるいは所望の情報を保管するのに使用することができ、コンピュータ１１０によってアクセスすることができる他のあらゆるメディアが含まれるが、これに制限はされない。通信メディアによって、通常は、搬送波または他のトランスポートメカニズムなどの変調されたデータ信号でコンピュータ可読命令、データ構造、プログラムモジュール、または他のデータが実施され、通信メディアには、すべての情報配信メディアが含まれる。用語「変調されたデータ信号」は、信号内で情報を符号化する形でその特性の１つまたは複数を設定または変更された信号を意味する。制限ではなく例として、通信メディアには、ワイヤードネットワークまたは直接配線接続などのワイヤードメディアと、音響、ＲＦ、赤外線、および他のワイヤレスメディアなどのワイヤレスメディアが含まれる。上記のいずれかの組合せも、コンピュータ可読メディアの範囲に含まれなければならない。

【００１９】

システムメモリ１３０には、読取専用メモリ（ＲＯＭ）１３１およびランダムアクセスメモリ（ＲＡＭ）１３２などの揮発性および／または不揮発性のメモリの形態のコンピュータストレージメディアが含まれる。起動中などにコンピュータ１１０内のエレメントの間での情報の転送を助ける基本ルーチンを含む基本入出力システム１３３（ＢＩＯＳ）が、通常はＲＯＭ１３１に保管される。ＲＡＭ１３２には、通常は、即座にアクセス可能であるおよび／または処理ユニット１２０によって現在操作されているデータおよび／またはプログラムモジュールが含まれる。制限ではなく例として、図１に、オペレーティングシステム１３４、アプリケーションプログラム１３５、他のプログラムモジュール１３６、およびプログラムデータ１３７が図示されている。

【００２０】

コンピュータ１１０には、他の取り外し可能／取り外し不能の揮発性／不揮発性コンピュータストレージメディアを含めることもできる。例としてのみ、図１に、取り外し不能不揮発性磁気メディアから読み取り、これに書き込むハードディスクドライブ１４１、取り外し可能不揮発性磁気ディスク１５２から読み取り、これに書き込む磁気ディスクドライブ１５１、およびＣＤ　ＲＯＭまたは他の光学メディアなどの取り外し可能不揮発性光ディスク１５６から読み取り、これに書き込む光学ディスクドライブ１５５が図示されている。この例示的オペレーティング環境で使用することができる他の取り外し可能／取り外し不能の揮発性／不揮発性コンピュータストレージメディアには、磁気テープカセット、フラッシュメモリカード、ディジタル多用途ディスク、ディジタルビデオテープ、ソリッドステートＲＡＭ、ソリッドステートＲＯＭ、および類似物が含まれるが、これに限定はされない。ハードディスクドライブ１４１は、通常は、インターフェース１４０などの

取り外し不能メモリインターフェースを介してシステムバス１２１に接続され、磁気ディスクドライブ１５１および光学ディスクドライブ１５５は、通常は、インターフェース１５０などの取り外し可能メモリインターフェースによってシステムバス１２１に接続される。

【００２１】

上で説明した、図１に図示されたドライブおよびそれに関連するコンピュータストレージメディアによって、コンピュータ１１０のコンピュータ可読命令、データ構造、プログラムモジュール、および他のデータのストレージが提供される。図１では、たとえば、ハードディスクドライブ１４１が、オペレーティングシステム１４４、アプリケーションプログラム１４５、他のプログラムモジュール１４６、およびプログラムデータ１４７を保管する状態で図示されている。これらのコンポーネントを、オペレーティングシステム１３４、アプリケーションプログラム１３５、他のプログラムモジュール１３６、およびプログラムデータ１３７と同一または異なるの、いずれかとすることができる。オペレーティングシステム１４４、アプリケーションプログラム１４５、他のプログラムモジュール１４６、およびプログラムデータ１４７は、最低限、これらが異なるコピーであることを示すために、本明細書では異なる符号を与えられている。ユーザは、キーボード１６２およびポインティングデバイス１６１（通常は、マウス、トラックボール、またはタッチパッドと称する）などの入力デバイスを介してコンピュータ１１０にコマンドおよび情報を入力することができる。他の入力デバイス（図示せず）に、マイクロホン、ジョイスティック、ゲームパッド、衛星放送受信用パラボラアンテナ、スキャナ、または類似物を含めることができる。上記および他の入力デバイスは、しばしば、システムバス１２１に結合されたユーザ入力インターフェース１６０を介して処理ユニット１２０に接続されるが、パラレルポート、ゲームポート、またはｕｎｉｖｅｒｓａｌ　ｓｅｒｉａｌ　ｂｕｓ（ＵＳＢ）などの、他のインターフェースおよびバス構造によって接続されることが可能である。

【００２２】

モニタ１９１または他のタイプのディスプレイデバイスも、ビデオインターフェース１９０などのインターフェースを介してシステムバス１２１に接続される。ノースブリッジなどのグラフィックスインターフェース１８２も、システムバス１２１に接続することができる。ノースブリッジは、ＣＰＵまたはホスト処理ユニット１２０と通信するチップセットであり、ａｃｃｅｌｅｒａｔｅｄ　ｇｒａｐｈｉｃｓ　ｐｏｒｔ（ＡＧＰ）通信の責任を負う。１つまたは複数のグラフィックス処理ユニット（ＧＰＵ）１８４によって、グラフィックスインターフェース１８２と通信することができる。これに関して、ＧＰＵ１８４には、一般に、レジスタストレージなどのオンチップメモリストレージが含まれ、ＧＰＵ１８４によって、ビデオメモリ１８６との通信が行われる。しかし、ＧＰＵ１８４は、コプロセッサの１つの例にすぎず、したがって、さまざまなコプロセッシングデバイスを、コンピュータ１１０に含めることができる。モニタ１９１または他のタイプのディスプレイデバイスも、ビデオインターフェース１９０などのインターフェースを介してシステムバス１２１に接続され、このビデオインターフェース１９０によって、ビデオメモリ１８６と通信することができる。モニタ１９１の他に、コンピュータに、スピーカ１９７およびプリンタ１９６など、出力周辺インターフェース１９５を介して接続することができる他の周辺出力デバイスも含めることができる。

【００２３】

コンピュータ１１０は、リモートコンピュータ１８０などの１つまたは複数のリモートコンピュータへの論理接続を使用して、ネットワーク化された環境で動作することができる。リモートコンピュータ１８０は、パーソナルコンピュータ、サーバ、ルータ、ネットワークＰＣ、ピアデバイス、または他の一般的なネットワークノードとすることができ、リモートコンピュータ１８０には、通常は、コンピュータ１１０に関して上述したエレメントの多数またはすべてが含まれるが、図１にはメモリストレージデバイス１８１だけが図示されている。図１に示された論理接続には、ローカルエリアネットワーク（ＬＡＮ）

１７１および広域ネットワーク（ＷＡＮ）１７３が含まれるが、他のネットワークも含めることができる。そのようなネットワーキング環境は、オフィス、会社全体のコンピュータネットワーク、イントラネット、およびインターネットでありふれている。

【００２４】

ＬＡＮネットワーキング環境で使用される場合に、コンピュータ１１０は、ネットワークインターフェースまたはネットワークアダプタ１７０を介してＬＡＮ１７１に接続される。ＷＡＮネットワーキング環境で使用される場合に、コンピュータ１１０には、通常は、インターネットなどのＷＡＮ１７３を介する通信を確立する、モデム１７２または他の手段が含まれる。モデム１７２は、内蔵または外付けとすることができるが、ユーザ入力インターフェース１６０または他の適当なメカニズムを介してシステムバス１２１に接続することができる。ネットワーク化された環境では、コンピュータ１１０に関して示されたプログラムモジュールまたはその諸部分を、リモートメモリストレージデバイスに保管することができる。制限ではなく例として、図１に、メモリデバイス１８１に常駐するリモートアプリケーションプログラム１８５を図示する。図示のネットワーク接続が、例示的であり、コンピュータの間の通信リンクを確立する他の手段を使用することができることを諒解されたい。

【００２５】

当業者は、コンピュータ１１０または他のクライアントデバイスを、コンピュータネットワークの一部として展開することができることを諒解することができる。これに関して、本発明は、任意の個数のメモリまたはストレージユニット、および任意の数のストレージユニットまたはボリュームにまたがって発生する任意の数のアプリケーションおよびプロセスを有するあらゆるコンピュータシステムに関係する。本発明は、リモートストレージまたはローカルストレージを有する、ネットワーク環境内で展開されたサーバコンピュータおよびクライアントコンピュータを伴う環境に適用することができる。本発明は、プログラミング言語機能性、解釈機能、および実行機能を有する独立型コンピューティングデバイスにも適用することができる。

【００２６】

分散コンピューティングによって、コンピューティングデバイスとシステムとの間の直接交換によるコンピュータリソースおよびサービスの共有が容易になる。これらのリソースおよびサービスには、情報の交換、キャッシュストレージ、およびファイル用のディスクストレージが含まれる。分散コンピューティングでは、ネットワーク接続性が利用され、クライアントが、集合的な威力を活用して企業全体に利益をもたらすことができるようになる。これに関して、さまざまなデバイスが、信頼されるグラフィックスパイプラインに関する本発明の認証テクニックを含めるために相互作用することができるアプリケーション、オブジェクト、またはリソースを有する。

【００２７】

図２に、例示的なネットワーク化されたコンピューティング環境または分散コンピューティング環境の概略図を示す。分散コンピューティング環境には、コンピューティングオブジェクト１０ａ、１０ｂなどおよびコンピューティングオブジェクトまたはコンピューティングデバイス１１０ａ、１１０ｂ、１１０ｃが含まれる。これらのオブジェクトに、プログラム、メソッド、データストア、プログラマブルロジックなどを含めることができる。オブジェクトに、ＰＤＡ、テレビジョン、ＭＰ３プレイヤ、テレビジョン、パーソナルコンピュータなどの同一のデバイスまたは異なるデバイスの一部を含めることができる。各オブジェクトは、通信ネットワーク１４によって別のオブジェクトと通信することができる。このネットワーク自体に、図２のシステムにサービスを提供する、他のコンピューティングオブジェクトおよびコンピューティングデバイスを含めることができる。本発明の態様によれば、各オブジェクト１０または１１０に、信頼されるグラフィックスパイプラインに関する本発明の認証テクニックを要求する可能性があるアプリケーションを含めることができる。

【００２８】

１１０ｃなどのオブジェクトを、別のコンピューティングデバイス１０または１１０上でホスティングすることができることも諒解することができる。したがって、図示の物理的環境では、接続されたデバイスがコンピュータとして図示されているが、そのような図示は、単に例示的であり、代わりに、ＰＤＡ、テレビジョン、ＭＰ３プレイヤなどのさまざまなディジタルデバイス、またはインターフェース、ＣＯＭオブジェクト、および類似物などのソフトウェアオブジェクトを含む物理的環境を図示するか説明することができる。

【００２９】

分散コンピューティング環境をサポートするさまざまなシステム、コンポーネント、およびネットワーク構成がある。たとえば、コンピューティングシステムを、ワイヤラインシステムまたはワイヤレスシステムによって、ローカルネットワークまたは広い範囲に分散したネットワークによって、一緒に接続することができる。現在、ネットワークの多くが、インターネットに結合され、このインターネットによって、広範囲に分散したコンピューティングネットワークのインフラストラクチャが提供され、このインターネットに、多数の異なるネットワークが含まれる。

【００３０】

ホームネットワーキング環境には、電力線、データ（ワイヤレスおよびワイヤードの両方）、音声（たとえば電話）、およびエンターテイメントメディアなどの独自のプロトコルをそれぞれがサポートすることができる少なくとも４つの異なるネットワークトランスポートメディアがある。照明スイッチおよび照明器具などのホームコントロールデバイスのほとんどによって、接続に電力線を使用することができる。データサービスは、ブロードバンド（たとえば、ＤＳＬモデムまたはケーブルモデムのいずれか）として家庭に入ることができ、家庭内でワイヤレス接続性（たとえば、ＨｏｍｅＲＦまたは８０２．１１ｂ）またはワイヤード接続性（たとえば、Ｈｏｍｅ　ＰＮＡ、Ｃａｔ　５、ｅｖｅｎ　ｐｏｗｅｒ　ｌｉｎｅ）のいずれかを使用してアクセス可能である。音声トラフィックは、ワイヤード（たとえば、Ｃａｔ　３）またはワイヤレス（たとえば、セル電話）のいずれかとして家庭に入ることができ、家庭内でＣａｔ　３配線を使用して配布することができる。エンターテイメントメディアは、衛星またはケーブルのいずれかを介して家庭に入ることができ、通常は、家庭内で同軸ケーブルを使用して配布される。ＩＥＥＥ　１３９４およびＤＶＩも、メディアデバイスのクラスタのディジタル相互接続として現れつつある。これらのネットワーク環境およびプロトコル標準規格として現れる可能性がある他の環境のすべてを、インターネットによって外部の世界に接続することができるイントラネットを形成するように相互接続することができる。短く言うと、データのストレージおよび伝送のためのさまざまな異なるソースが存在し、その結果、前に進むと、コンピューティングデバイスは、そのデータ処理パイプラインのすべての部分で、コンテンツを保護する形を必要とするようになる。

【００３１】

「インターネット」は、一般に、コンピュータネットワーキングの技術で周知の、ＴＣＰ／ＩＰプロトコルスイートを使用するネットワークおよびゲートウェイの集合を指す。ＴＣＰ／ＩＰは、「Ｔｒａｎｓｐｏｒｔ　Ｃｏｎｔｒｏｌ　Ｐｒｏｔｏｃｏｌ／Ｉｎｔｅｒｆａｃｅ　Ｐｒｏｇｒａｍ（転送制御プロトコル／インターフェースプログラム）」の頭字語である。インターネットは、ユーザがネットワークを介して相互作用でき、情報を共有できるようにする、ネットワーキングプロトコルを実行するコンピュータによって相互接続された、地理的に分散したリモートコンピュータネットワークのシステムとして説明することができる。そのような広範囲に広がった情報共有のゆえに、インターネットなどのリモートネットワークは、これまで、全般的に、デベロッパが、本質的に制限なしで特殊化されたオペレーションまたはサービスを実行するソフトウェアアプリケーションを設計することができるオープンシステムに進化してきた。

【００３２】

したがって、ネットワークインフラストラクチャによって、クライアント／サーバアー

キテクチャ、ピアツーピアアーキテクチャ、またはハイブリッドアーキテクチャなどのネットワークトポロジのホストが可能になる。「クライアント」は、それが関係しない別のクラスまたはグループのサービスを使用する、あるクラスまたはグループのメンバである。したがって、コンピューティングにおいて、クライアントは、別のプログラムによって提供されるサービスを要求するプロセスすなわち、おおまかには命令の組またはタスクである。クライアントプロセスによって、他のプログラムまたはサービス自体に関する機能の詳細を一切「知る」必要なしに、要求されたサービスが使用される。クライアント／サーバアーキテクチャ、特にネットワークシステムにおいて、クライアントは、通常は、たとえばサーバなどの別のコンピュータによって提供される共有ネットワークリソースにアクセスするコンピュータである。図２の例において、コンピュータ１１０ａ、１１０ｂなどを、クライアントと考えることができ、コンピュータ１０ａ、１０ｂなどを、サーバと考えることができ、ここで、サーバ１０ａ、１０ｂなどによって、クライアントコンピュータ１１０ａ、１１０ｂなどに複製されるデータが維持される。

【００３３】

サーバは、通常は、インターネットなどのリモートネットワークを介してアクセス可能なリモートコンピュータシステムである。クライアントプロセスは、第１コンピュータシステム内でアクティブとすることができ、サーバプロセスは、第２コンピュータシステム内でアクティブとすることができ、これらが通信メディアを介して互いに通信し、したがって、分散機能性がもたらされ、複数のクライアントがサーバの情報収集機能を利用できるようになる。

【００３４】

クライアントおよびサーバは、プロトコルレイヤによって提供される機能性を使用して互いに通信する。たとえば、ハイパーテキスト転送プロトコル（ＨＴＴＰ）は、ワールドワイドウェブ（WWW）と共に使用される一般的なプロトコルである。通常、Ｕｎｉｖｅｒｓａｌ　Ｒｅｓｏｕｒｃｅ　Ｌｏｃａｔｏｒ（ＵＲＬ）またはインターネットプロトコル（ＩＰ）アドレスなどのコンピュータネットワークアドレスが、サーバコンピュータまたはクライアントコンピュータがお互いを識別するのに使用される。ネットワークアドレスを、Ｕｎｉｖｅｒｓａｌ　Ｒｅｓｏｕｒｃｅ　Ｌｏｃａｔｏｒアドレスと呼ぶことができる。たとえば、通信を、通信メディアを介して提供することができる。具体的に言うと、クライアントおよびサーバを、大容量通信に関するＴＣＰ／ＩＰ接続を介して互いに結合することができる。

【００３５】

したがって、図２に、サーバがネットワーク／バスを介してクライアントコンピュータと通信する、本発明を実施することができる、例示的なネットワーク化された環境または分散環境が示されている。さらに詳細に言うと、本発明にしたがって、複数のサーバ１０ａ、１０ｂなどが、ＬＡＮ、ＷＡＮ、イントラネット、インターネットなどとすることができる通信ネットワーク／バス１４を介して、ポータブルコンピュータ、ハンドヘルドコンピュータ、シンクライアント、ネットワーク化された機器、またはＶＣＲ、ＴＶ、オーブン、照明、ヒータ、および類似物などの他のデバイスなどの複数のクライアントコンピューティングデバイスまたはリモートコンピューティングデバイス１１０ａ、１１０ｂ、１１０ｃ、１１０ｄ、１１０ｅなど相互接続される。したがって、本発明を、それに関して信頼されるソースからのセキュアコンテンツを処理し、保管し、またはレンダリングすることが望ましいあらゆるコンピューティングデバイスに適用することができることが企図されている。

【００３６】

通信ネットワーク／バス１４がインターネットであるネットワーク化された環境では、たとえば、サーバ１０を、ウェブサーバとすることができ、このウェブサーバに、クライアント１１０ａ、１１０ｂ、１１０ｃ、１１０ｄ、１１０ｅなどが、ＨＴＴＰなどの複数の既知のプロトコルのいずれかを介して通信する。サーバ１０は、クライアント１１０として働くこともでき、これは、分散コンピューティング環境の特性とすることができる。

通信は、適当な場合に、ワイヤードまたはワイヤレスとすることができる。クライアントデバイス１１０は、通信ネットワーク／バス１４を介して通信してもしなくてもよく、独立の通信を関連付けられることができる。たとえば、ＴＶまたはＶＣＲの場合に、その制御に対するネットワーク化された態様がある場合もそうでない場合もある。各クライアントコンピュータ１１０およびサーバコンピュータ１０は、さまざまなアプリケーションプログラムモジュールまたはオブジェクト１３５を備えることができ、さまざまなタイプのストレージエレメントまたはオブジェクトへの接続またはアクセスを備えることができ、この接続またはアクセスを介して、ファイルを保管するか、ファイルの一部をダウンロードまたは移植することができる。したがって、本発明を、通信ネットワーク／バス１４にアクセスし、これと相互作用することができるクライアントコンピュータ１１０ａ、１１０ｂなどと、コンピュータ１１０ａ、１１０ｂなどおよび他のデバイス１１１およびデータベース２０と相互作用することができるサーバコンピュータ１０ａ、１０ｂとを有するコンピュータネットワーク環境で使用することができる。

【００３７】

ディジタル権利管理（ＤＲＭ）の概要

既知のように、図１１を参照すると、ディジタル権利管理（ＤＲＭ）およびディジタル権利実施は、ディジタルオーディオ、ディジタルビデオ、ディジタルテキスト、ディジタルデータ、ディジタルマルチメディアなどのディジタルコンテンツ１２に関して、そのようなディジタルコンテンツ１２がユーザに配布される場合に、非常に望ましい。ユーザによって受け取られる際に、そのようなユーザは、パーソナルコンピュータ１４のメディアプレイヤまたは類似物などの適当なレンダリングデバイスの助けを得てディジタルコンテンツをレンダリングまたは「プレイ」する。

【００３８】

通常、そのようなディジタルコンテンツ１２を配布するコンテンツオーナーまたはコンテンツデベロッパ（以下では「オーナー」と称する）は、ユーザがそのように配布されるディジタルコンテンツ１２に関して行うことができることを制限することを望む。たとえば、コンテンツオーナーは、ユーザが、そのようなコンテンツ１２をコピーし、第２のユーザに再配布することを制限することを望む可能性があり、あるいは、配布されたディジタルコンテンツ１２を、限られた回数だけ、あるトータル時間だけ、あるタイプのマシンでのみ、あるタイプのメディアプレイヤでのみ、あるタイプのユーザによってのみ、などでプレイできるようにすることを望む可能性がある。

【００３９】

しかし、配布が行われた後には、そのようなコンテンツオーナーは、ディジタルコンテンツ１２に対する制御を、あるとしてもごくわずかだけ有する。ＤＲＭシステム１０を用いると、ディジタルコンテンツ１２の任意の形での制御されたレンダリングまたはプレイが可能になり、そのような制御は、柔軟であり、そのようなディジタルコンテンツのコンテンツオーナーによって定義可能である。通常、コンテンツ１２は、適当な配布チャネルによってパッケージ１３の形でユーザに配布される。配布されるディジタルコンテンツパッケージ１３に、対称暗号化／暗号化解除キー（ＫＤ）（すなわち（ＫＤ（ＣＯＮＴＥＮＴ））を用いて暗号化されたディジタルコンテンツならびに、コンテンツを識別する情報、そのようなコンテンツのライセンスを獲得する方法などの他の情報を含めることができる。

【００４０】

信頼ベースのＤＲＭシステム１０を用いると、ディジタルコンテンツ１２のオーナーが、そのようなディジタルコンテンツ１２がユーザのコンピューティングデバイス１４でのレンダリングを許可される前に満足されなければならないライセンスルールを指定できるようになる。そのようなライセンスルールに、前述の時間的要件を含めることができ、そのようなライセンスルールは、ユーザ／ユーザのコンピューティングデバイス１４（以下、このような用語は、状況が他の形を必要としない限り、交換可能である）がコンテンツオーナーまたはそのエージェントから入手しなければならないディジタルライセンスまた

は使用ドキュメント（以下では「ライセンス」と称する）１６内で実施することができる。そのようなライセンス１６に、おそらくはユーザのコンピューティングデバイスによって暗号化解除可能なキーによって暗号化されたディジタルコンテンツを暗号化解除する暗号化解除キー（ＫＤ）も含まれる。

【００４１】

ディジタルコンテンツ１２の１つのコンテンツオーナーは、ユーザのコンピューティングデバイス１４が、そのようなコンテンツオーナーによってライセンス１６内で指定されたルールおよび要件を守ることすなわち、ディジタルコンテンツ１２が、ライセンス１６内のルールおよび要件が満足されない限りレンダリングされないことを信頼しなければならない。ユーザのコンピューティングデバイス１４が、ディジタルコンテンツ１２に関連し、ユーザによって入手されるライセンス１６で実施されるライセンスルールに従う場合を除いてディジタルコンテンツ１２がレンダリングされない、信頼されるコンポーネントまたは信頼されるメカニズム１８を備えることが好ましい。

【００４２】

信頼されるコンポーネント１８は、通常は、ライセンスエバリュエータ２０を有し、このライセンスエバリュエータ２０によって、とりわけ、ライセンス１６が有効であるかどうかが判定され、そのような有効なライセンス１６のライセンスルールおよびライセンス要件が再検討され、再検討されたライセンスルールおよびライセンス要件に基づいて、要求元のユーザが、要求されたディジタルコンテンツ１２を求められる形でレンダリングする権利を有するかどうかが判定される。理解されるように、ライセンスエバリュエータ２０は、ＤＲＭシステム１０内で信頼されて、ライセンス１６のルールおよび要件に従ってディジタルコンテンツ１２のオーナーの望みを実行し、ユーザは、極悪なまたは他の形のすべての目的のためにそのような信頼されるエレメントを簡単に変更することができてはならない。

【００４３】

理解されなければならないように、ライセンス１６のルールおよび要件によって、ユーザが、そのユーザが誰であるか、そのユーザがどこに位置するか、そのユーザが使用しているコンピューティングデバイスのタイプが何であるか、どのレンダリングアプリケーションによってＤＲＭシステムが呼び出されているか、日付、時刻などを含む複数の要因に基づいて、ディジタルコンテンツ１２をレンダリングする権利を有するかどうかを指定することができる。さらに、ライセンス１６のルールおよび要件によって、たとえば事前に決定されるプレイ回数または事前に決定されるプレイ時間にライセンス１６を制限することができる。

【００４４】

ルールおよび要件は、任意の適当な言語および構文に従ってライセンス１６内で指定することができる。たとえば、この言語によって、満足されなければならない属性および値を単純に指定することができ（たとえば、ＤＡＴＥ ｍｕｓｔ ｂｅ ｌａｔｅｒ ｔｈａｎ Ｘ）、あるいは、指定されたスクリプトによる関数の実行を要求することができる（たとえば、ＩＦ ＤＡＴＥ ｇｒｅａｔｅｒ ｔｈａｎ Ｘ，ＴＨＥＮ ＤＯ ．．．）。

【００４５】

ライセンスエバリュエータ２０が、ライセンス１６が有効であり、ユーザがそのライセンスのルールおよび要件を満足すると判定する際に、ディジタルコンテンツ１２をレンダリングすることができる。具体的に言うと、コンテンツ１２をレンダリングするために、暗号化解除キー（ＫＤ）が、ライセンス１２から入手され、コンテンツパッケージ１３からの（ＫＤ（ＣＯＮＴＥＮＴ））に適用されて、実際のコンテンツ１２がもたらされ、実際のコンテンツ１２が、実際にレンダリングされる。

【００４６】

ディジタルコンテンツのパブリッシュ

図３は、ディジタルコンテンツをパブリッシュするシステムおよび方法の機能ブロック

図である。本明細書で使用される用語としての「パブリッシュ」は、アプリケーションまたはサービスが、信頼されるエンティティと共に、そのエンティティがそのコンテンツを発行できる権利および条件の組ならびにその権利および条件を発行できる相手を確立するために従うプロセスである。本発明によれば、パブリッシュプロセスに、ディジタルコンテンツを暗号化することと、コンテンツの作者がコンテンツのすべての可能なユーザについて企図した永続的で実施可能な権利のリストを関連付けることが含まれる。このプロセスをセキュアな形で実行して、コンテンツの作者によって意図されないすべての権利またはコンテンツへのアクセスを禁止することができる。

【００４７】

具体的には、３つのエンティティが、セキュアディジタルコンテンツのパブリッシュに使用される。すなわち、クライアント３００で実行され、パブリッシュのためにコンテンツを準備するコンテンツ準備アプリケーション３０２、やはりクライアントデバイス３００に常駐するディジタル権利管理（ＤＲＭ）アプリケーションプログラムインターフェース（ＡＰＩ）３０６、および、インターネット、ローカルエリアネットワーク、広域ネットワーク、またはこれらの組合せなどの通信ネットワーク３３０を介してクライアント３００に通信的に結合されるＤＲＭサーバ３２０である。

【００４８】

コンテンツ準備アプリケーション３０２は、ディジタルコンテンツを作る任意のアプリケーションとすることができる。たとえば、アプリケーション３０２は、ディジタルテキストファイル、ディジタルミュージック、ビデオ、または他のそのようなコンテンツを作る、ワードプロセッサまたは他のパブリッシャとすることができる。コンテンツに、ライブイベントまたはテープに記録されたイベントまたは例のストリーミングされたオーディオ／ビデオなどの、ストリーミングされたコンテンツを含めることもできる。アプリケーション３０２は、ディジタルコンテンツを暗号化し、したがって、暗号化されたディジタルコンテンツファイル３０４を形成する暗号キーを備え、ユーザは、ディジタルコンテンツファイル３０４の暗号化されたコンテンツに緊密に関連付けられる権利データを提供する。権利データには、ディジタルコンテンツの権利を有するエンティティごとの識別ならびに各識別されたエンティティの権利および条件の組が含まれる。

【００４９】

そのようなエンティティは、たとえば、個人、個人のクラス、またはデバイスとすることができる。そのような権利に、ディジタルコンテンツを読み取る権利、ディジタルコンテンツを編集する権利、ディジタルコンテンツをコピーする権利、ディジタルコンテンツを印刷する権利などを含めることができる。条件に、最小システム要件、日付制限、回数制限、プレイカウント、および類似物を含めることができる。

【００５０】

クライアントＡＰＩ３０６によって、暗号化されたディジタルコンテンツデータおよび権利データが、ＤＲＭサーバ３２０に渡される。下で詳細に説明するプロセスを使用して、ＤＲＭサーバ３２０によって、そのＤＲＭサーバ３２０によって権利データを実施できるかどうかが判定され、そうである場合には、ＤＲＭサーバ３２０によって、権利データに署名して、署名付き権利ラベル（ＳＲＬ）３０８が形成される。しかし、一般に、信頼されるエンティティであればどれであっても、好ましくはＤＲＭサーバ３２０によって信頼されるキーを使用して、権利データに署名することができる。たとえば、クライアントが、ＤＲＭサーバ３２０によってクライアントに供給されるキーを使用して権利データに署名することができる。

【００５１】

権利ラベル３０８に、権利記述、暗号化されたコンテンツキー、ならびに権利記述および暗号化されたコンテンツキーに対するディジタル署名を表すデータを含めることができる。ＤＲＭサーバ３２０によって権利ラベルに署名される場合には、ＤＲＭサーバ３２０によって、署名付き権利ラベル３０８が、クライアントＡＰＩ３０６を介してクライアントに渡され、このクライアントによって、署名付き権利ラベル３０８が、クライアントデ

バイス３００で保管される。コンテンツ準備アプリケーション３０２によって署名付き権利ラベル３０８が、たとえば権利管理されたコンテンツファイル３１０を形成するための連結によるなど、暗号化されたディジタルコンテンツファイル３０４に関連付けられる。しかし、コンテンツファイル３１０を形成するためにコンテンツファイル３０４に連結されたＳＲＬ３０８への参照を用いて、ＳＲＬ３０８を、コンテンツファイル３０４と別々に既知の位置に保管することができることに留意されたい。

【００５２】

図４を参照すると、権利管理されたディジタルコンテンツをパブリッシュする方法の１つが示されている。ステップ４０２で、アプリケーション３０２によって、ディジタルコンテンツを暗号化するのに使用されるコンテンツキー（ＣＫ）が生成される。コンテンツキー（ＣＫ）は、通常は、対称キーであるが、ディジタルコンテンツの暗号化に任意のキーを使用することができる。既知のように、対称キーは、対称キーアルゴリズムによって、暗号化および暗号化解除の両方に使用される。したがって、（ＣＫ）は、送信側と受信側の間で共有される場合に、きちんと隠蔽されなければならない。

【００５３】

ステップ４０４で、アプリケーション３０２によって、（ＣＫ）を用いてディジタルコンテンツを暗号化して、暗号化されたコンテンツファイル３０４（すなわち（ＣＫ（ｃｏｎｔｅｎｔ））を形成する。さらに、（ＣＫ（ｃｏｎｔｅｎｔ））に対応する権利データが、コンテンツのパブリッシャまたは別のエンティティのいずれかによって生成される。そのような権利データを、カスタム権利データまたは、事前に定義されたテンプレートから入手される権利データとすることができることに留意されたい。上で説明したように、権利データに、コンテンツを消費する権利を与えられるエンティティのリスト、コンテンツに関してエンティティのそれぞれが所有する特定の権利、およびこれらの権利に押し付けることができるすべての条件を含めることができる。

【００５４】

ステップ４０６で、ＡＰＩ３０６によって、第２の暗号化キー（ＤＥＳ１）が生成され、このキーはコンテンツキー（ＣＫ）を暗号化するのに使用される。（ＤＥＳ１）も、対称鍵であることが好ましい。ステップ４０８で、ＡＰＩ３０６によって、（ＤＥＳ１）を用いて（ＣＫ）を暗号化して、（ＤＥＳ１（ＣＫ））をもたらす。ステップ４１０で、ＡＰＩ３０６によって、（ＣＫ）を破棄し、その結果、（ＣＫ）は、（ＤＥＳ１（ＣＫ））を暗号化解除することによってのみ入手できるようになる。（ＣＫ（ｃｏｎｔｅｎｔ））が中央ＤＲＭサーバ３２０に保護されることと、コンテンツに関するすべての「ライセンス要求」が権利データに従って集中して行われることを保証するために、ステップ４１２で、ＡＰＩ３０６によって、提供されるＤＲＭサーバ３２０に連絡し、その公開キー（ＰＵ－ＤＲＭ）を検索する。ステップ４１４で、ＡＰＩ３０６によって、（ＰＵ－ＤＲＭ）を用いて（ＤＥＳ１）を暗号化して、（ＰＵ－ＤＲＭ（ＤＥＳ１））をもたらす。したがって、（ＣＫ）を（ＰＵ－ＤＲＭ））に保護して、ＤＲＭサーバ３２０が、（ＣＫ（ｃｏｎｔｅｎｔ））の暗号化解除に必要な（ＣＫ）へのアクセスを得ることができる唯一のエンティティであることを保証する。ステップ４１６で、ＡＰＩ３０６によって、（ＤＥＳ１）を用いて権利データ（すなわち、許可されるエンティティのリストならびにリスト内の許可されるエンティティのそれぞれに関連するめいめいの権利および条件）を暗号化して、（ＤＥＳ１（ｒｉｇｈｔｓｄａｔａ））をもたらす。

【００５５】

代替実施形態では、（ＣＫ）を使用して、権利データを直接に暗号化して、（ＣＫ（ｒｉｇｈｔｓｄａｔａ））をもたらすことができ、（ＰＵ－ＤＲＭ）を使用して、（ＣＫ）を直接に暗号化して、（ＰＵ－ＤＲＭ（ＣＫ））をもたらすことができ、これによって、（ＤＥＳ１）の使用を完全になしで済ませることができる。しかし、（ＤＥＳ１）を使用して権利データおよび（ＣＫ）を暗号化することによって、そのような（ＤＥＳ１）を、ＤＲＭサーバに従うようにすることができる任意の特定のアルゴリズムに従わせることができ、（ＣＫ）を、ＤＲＭサーバから独立のエンティティによって指定することができ、

ＤＲＭサーバに従う（ＣＫ）としなくてよい。

【００５６】

ステップ４１８で、コンテンツ保護アプリケーション３０２によって、署名用の権利ラベルとしてＤＲＭサーバ３２０に（ＰＵ－ＤＲＭ（ＤＥＳ１））および（ＤＥＳ１（ｒｉｇｈｔｓｄａｔａ））がサブミットされる。代わりに、クライアント自体が、下で示す形で権利データに署名することができる。権利データが、署名のためにサーバにサブミットされている場合には、ステップ４２０で、ＤＲＭサーバ３２０によって、権利データがアクセスされ、サブミットされた権利ラベルの権利および条件をＤＲＭサーバ３２０によって実施できることが検証される。ＤＲＭサーバ３２０によって権利を実施できることを検証するために、ＤＲＭサーバ３２０によって、（ＰＵ－ＤＲＭ）に対応する秘密キー（ＰＲ－ＤＲＭ）を（ＰＵ－ＤＲＭ（ＤＥＳ１））に適用して、（ＤＥＳ１）をもたらし、次に、（ＤＥＳ１）を（ＤＥＳ１（ｒｉｇｈｔｓｄａｔａ））に適用して、平文の権利データをもたらす。次に、サーバ３２０によって、ポリシ検査を実行して、権利データで指定されたユーザ、権利、および条件が、サーバ３２０によって実施されるすべてのポリシに含まれることを検証する。サーバ３２０によって、最初にサブミットされた、（ＰＵ－ＤＲＭ（ＤＥＳ１））および（ＤＥＳ１（ｒｉｇｈｔｓｄａｔａ））を含む権利ラベルに署名して、署名付き権利ラベル（ＳＲＬ）３０８をもたらし（この署名は、ＤＲＭサーバ３２０の秘密キー（ＰＲ－ＤＲＭ）に基づく）、ＳＲＬ３０８がＡＰＩ３０６に返され、ＡＰＩ３０６によって、返されたＳＲＬ３０８が、クライアントアプリケーション３０２に提示される。

【００５７】

ＳＲＬ３０８は、ディジタル署名されたドキュメントであり、このディジタル署名によって、ＳＲＬ３０８が耐タンパにされる。さらに、ＳＲＬ３０８は、コンテンツを暗号化するのに使用された実際のキーのタイプおよびアルゴリズムと独立であるが、それが保護するコンテンツに対する強い１対１関係が維持される。図４Ａを参照すると、本発明の一実施形態で、ＳＲＬ３０８に、とりわけ、おそらくはコンテンツのＩＤ；（ＰＵ－ＤＲＭ（ＤＥＳ１））と、ネットワークでＤＲＭサーバを突き止めるためのＵＲＬおよびＵＲＬに障害が発生する場合のフォールバック情報などの参照情報とを含む、ＳＲＬ３０８に署名するＤＲＭサーバに関する情報；ＳＲＬ３０８自体を説明する情報；（ＤＥＳ１（ｒｉｇｈｔｓｄａｔａ））；（ＤＥＳ１（ＣＫ））；およびディジタル署名（Ｓ（ＰＲ－ＤＲＭ））を含むＳＲＬ３０８の基礎であるコンテンツに関する情報を含めることができる。

【００５８】

署名付き権利ラベル３０８を作るために、信頼されるエンティティが権利データに署名することを保証することによって、ＤＲＭサーバ３２０によって、権利ラベル３０８の権利データで説明されるパブリッシャによって示される条件に従ってコンテンツに関するライセンスが発行されることが、ＤＲＭサーバ３２０によって主張されつつある。諒解されなければならないように、ユーザは、特にライセンスにコンテンツキー（ＣＫ）が含まれる限り、コンテンツをレンダリングするためにライセンスを入手することを要求される。ユーザが、暗号化されたコンテンツに関するライセンスを入手することを求める場合には、ユーザは、コンテンツのＳＲＬ３０８およびＤＲＭサーバ３２０または他のライセンスを発行するエンティティへのユーザの信任状を検証する証明書を含むライセンス要求を提示することができる。ライセンスを発行するエンティティは、（ＰＵ－ＤＲＭ（ＤＥＳ１））および（ＤＥＳ１（ｒｉｇｈｔｓｄａｔａ））を暗号化解除して、権利データを作ることができ、ライセンスを要求するエンティティに対して作者（存在する場合に）によって許可されるすべての権利をリストすることができ、これらの特定の権利だけを有するライセンスを構築することができる。

【００５９】

上で示したように、アプリケーション３０２がＳＲＬ３０８を受け取る際に、そのようなアプリケーション３０２によって、署名付き権利ラベル３０８に、対応する（ＣＫ（ｃｏｎｔｅｎｔ））３０４が連結されて、権利管理されたディジタルコンテンツが形成され

る。代わりに、権利データが、既知の位置に保管され、その位置への参照が、暗号化されたディジタルコンテンツと共に提供される。したがって、ＤＲＭ対応であるレンダリングアプリケーションによって、そのレンダリングアプリケーションによってレンダリングが試みられつつあるコンテンツの部分を介して、署名付き権利ラベル３０８を発見することができる。このディスカバリによって、レンダリングアプリケーションによる、ＤＲＭライセンシングサーバ３２０に対するライセンス要求の開始がトリガされる。パブリッシュするアプリケーション３０２によって、たとえば、ＵＲＬをＤＲＭライセンシングサーバ３２０に保管することができ、あるいは、ＤＲＭライセンシングサーバ３２０によって、権利ラベルにディジタル署名する前に、それ自体のＵＲＬをメタデータの一部として権利ラベルに組み込むことができ、その結果、レンダリングアプリケーションによって呼び出されるＤＲＭクライアントＡＰＩ３０６によって、正しいＤＲＭライセンシングサーバ３２０を識別できるようになる。

【００６０】

パブリッシュされたコンテンツに関するライセンスの入手

図５を参照すると、権利管理されたディジタルコンテンツをライセンシングするシステムおよび方法が示されている。本明細書で使用される用語としての「ライセンシング」は、ライセンスで指定される条件に従って、ライセンスで名前を指定されるエンティティがコンテンツを消費できるようにするライセンスを要求し、受け取るためにアプリケーションまたはサービスが従うプロセスを指す。ライセンシングプロセスへの入力に、ライセンスが要求されているコンテンツに関連する署名付き権利ラベル３０８およびそれに対するライセンスが要求されているエンティティの公開キー証明書を含めることができる。ライセンスを要求するエンティティが、必ずしもそれに関するライセンスが要求されているエンティティでないことに留意されたい。通常、ライセンスには、ＳＲＬ３０８からの権利記述、暗号化されたコンテンツを暗号化解除することができる暗号化されたキー、ならびに、合法性を主張し、タンパリングを防ぐための、権利記述および暗号化されたキーに対するディジタル署名が含まれる。

【００６１】

予備的に、クライアントＡＰＩ３０６によって、権利管理されたコンテンツ３１０の署名付き権利ラベル３０８が、通信ネットワーク３３０を介してＤＲＭサーバ３２０に転送される。上述したように、権利ラベル３０８には、ＤＲＭサーバ３２０の公開キー（ＰＵ－ＤＲＭ）に従って暗号化されたコンテンツキー（ＣＫ）（すなわち（ＰＵ－ＤＲＭ（ＣＫ））が含まれる。ライセンス発行のプロセスで、ＤＲＭサーバ３２０によって、（ＰＲ－ＤＲＭ）が（ＰＵ－ＤＲＭ（ＣＫ））に適用されて、（ＣＫ）が入手される。ＤＲＭサーバ３２０によって、ライセンス要求で渡される公開キー証明書の公開キー（ＰＵ－ＥＮＴＩＴＹ）が使用されて、（ＣＫ）が再暗号化される（すなわち（ＰＵ－ＥＮＴＩＴＹ（ＣＫ））。新たに暗号化された（ＰＵ－ＥＮＴＩＴＹ（ＣＫ））が、ライセンスに置かれる。したがって、ライセンスを、（ＣＫ）を露出する危険性なしに呼出し側に返すことができる。というのは、（ＰＵ－ＥＮＴＩＴＹ）に対応する秘密キー（ＰＲ－ＥＮＴＩＴＹ）の所有者だけが、（ＰＵ－ＥＮＴＩＴＹ（ＣＫ））から（ＣＫ）を回復できるからである。クライアントＡＰＩ３０６によって、（ＣＫ）が使用されて、暗号化されたコンテンツが暗号化解除されて、暗号化解除されたディジタルコンテンツ３１２が形成される。クライアントアプリケーション３０２によって、ライセンス内で供給される権利に従って、暗号化解除されたディジタルコンテンツ３１２を使用することができる。

【００６２】

代わりに、下で詳細に示すように、パブリッシュするクライアントなどのクライアントによって、たとえば、コンテンツを消費するための使用ライセンスをそれ自体に発行することができる。

【００６３】

図６Ａおよび６Ｂに移ると、権利管理されたディジタルコンテンツをライセンシングする方法が示されている。ステップ６０２で、ＤＲＭサーバ３２０などのライセンスを発行

するエンティティによって、公開キー証明書あるいは１つまたは複数の要求されるライセンシーのそれぞれの識別のいずれかを含むライセンス要求が受け取られる。おそらく、識別が指定される場合に、ＤＲＭサーバ３２０によって、ディレクトリ、データベース、または類似物から対応する公開キー証明書を獲得することができる。１つのライセンシーだけに関してライセンスが要求された場合には、１つの証明書だけまたは１つの識別だけが、名前を指定される。ライセンスが、複数のライセンシーについて要求された場合には、証明書または識別を、潜在的なライセンシーごとに名前を指定することができる。ステップ６０４で、望まれる場合に、要求するエンティティ（すなわち、ライセンス要求を行うエンティティ）が認証される。ステップ６０６で、やはり望まれる場合に、エンティティがライセンスを要求することを許可されるかどうかを判定する。

【００６４】

ステップ６０８で、発行するエンティティによって、公開キー証明書がライセンス要求に含まれていないかどうかが判定され、そうである場合に、発行するエンティティによって、指定された識別を使用して、適当な公開キー証明書についてディレクトリサービスまたはデータベースのルックアップが実行される。ステップ６１０で、発行するエンティティによって、証明書がディレクトリ内にあると判定される場合に、ステップ６１２で、その証明書を検索する。証明書を、要求内またはディレクトリ内のいずれでも所与の潜在的なライセンシーについて見つけることができない場合には、ライセンスサーバは、その潜在的なライセンシーに関するライセンスを生成せず、ステップ６１４で、要求元エンティティにエラーを返す。

【００６５】

ＤＲＭサーバ３２０が、少なくとも１つの潜在的なライセンシーの公開キー証明書を有すると仮定すると、ステップ６１６で、そのようなＤＲＭサーバ３２０によって、各ライセンシー証明書の信頼が検証される。検証されない場合に、ＤＲＭサーバ３２０によって、ライセンシー証明書の発行者が信頼される発行者のリストにないことが判定され、その後、要求がそのライセンシーについて失敗し、ステップ６１４でエラーが生成される。したがって、信頼される発行者によって発行された証明書を有しないすべての潜在的なライセンシーが、ライセンスを受け取らない。

【００６６】

さらに、ＤＲＭサーバ３２０によって、信頼される発行者証明書から個々のライセンシー公開キー証明書に進む証明書チェーン内のすべてのエンティティに関するディジタル証明書検証が実行されることが好ましい。チェーン内のディジタル証明書を検証するプロセスは、周知のアルゴリズムである。所与の潜在的なライセンシーの公開キー証明書が検証されない場合、またはチェーン内の証明書が検証されない場合に、その潜在的なライセンシーが信頼されず、したがって、ライセンスは、その潜在的なライセンシーに発行されない。そうでない場合には、ステップ６１８で、ライセンスを発行することができる。このプロセスは、それに関してライセンスが要求されたすべてのエンティティが処理されるまで、ステップ６２０で繰り返される。

【００６７】

図６Ｂからわかるように、ＤＲＭサーバ３２０は、ライセンス要求で受け取った署名付き権利ラベル３０８の検証に進む。一実施形態では、ＤＲＭサーバ３２０は、それによって署名されるすべての権利ラベルのマスタコピーを有する。ライセンスの際に（ステップ６２２で）、ＤＲＭサーバ３２０によって、マスタ権利ラベルのコピーを検索することができる。マスタ権利ラベルは、ライセンス要求で送られる権利ラベルのコピーより新しい可能性があり、したがって、要求されたライセンスを作るのに使用される権利ラベルになる。マスタ権利レベルが見つからない場合には、ＤＲＭサーバ３２０によって、ステップ６２４で、事前に定義されたポリシに従って、要求の権利ラベルに基づいてライセンスを発行するかどうかが判定される。ポリシによって許可されない場合には、ステップ６２６で、ライセンス要求が失敗し、ステップ６２８で、エラーがＡＰＩ３０６に返される。

【００６８】

ステップ６３０で、ＤＲＭサーバ３２０によって、ＳＲＬ３０８、特にそのディジタル署名が検証される。ＳＲＬ３０８が検証されない場合には、ステップ６２６でライセンス要求が失敗し、ステップ６２８でエラーがＡＰＩ３０６に返される。

【００６９】

すべての検証が行われた後に、ＤＲＭサーバによって、ＳＲＬ３０８に基づいて承認されたライセンスごとにライセンスが構築される。ステップ６３２で、ＤＲＭサーバ３２０によって、各ライセンシーに発行されるライセンスのめいめいの権利記述が生成される。ライセンシーごとに、ＤＲＭサーバ３２０によって、そのライセンシーの公開キー証明書で名前を指定された識別が、権利ラベルの権利記述で名前を指定された識別に対して評価される。ステップ６３６で、ＤＲＭサーバ３２０によって、ＳＲＬ３０８から（ＰＵ－ＤＲＭ（ＤＥＳ１））および（ＤＥＳ１（ＣＫ））が入手され、（ＰＲ－ＤＲＭ）が適用されて、（ＣＫ）が入手される。発行するエンティティによって、次に、ライセンシーの公開キー証明書からの（ＰＵ－ＥＮＴＩＴＹ）を使用して（ＣＫ）が再暗号化されて、（ＰＵ－ＥＮＴＩＴＹ（ＣＫ））がもたらされる。ステップ６３８で、ＤＲＭサーバ３２０によって、生成された権利記述と（ＰＵ－ＥＮＴＩＴＹ（ＣＫ））が連結され、結果のデータ構造が、（ＰＲ－ＤＲＭ）を使用してディジタル署名される（すなわち、Ｓ（ＰＲ－ＤＲＭ））。したがって、この署名されたデータ構造が、この特定のライセンシーのライセンスである。

【００７０】

ステップ６４０で、ＤＲＭサーバ３２０によって、特定の要求に関して生成されるライセンスがまだあるかどうかが判定される。生成されるラインセスは、その後、ステップ６４２で、ライセンスを信頼されるオーソリティに結び付ける適当な証明書チェーンと一緒に、要求元のエンティティに返される。

【００７１】

ディレクトリによるライセンシング

保護されたコンテンツのライセンスを発行する場合に、ラインセスを発行するエンティティ（以下では「ライセンサ」と称する）は、コンテンツからの送られたＳＲＬ３０８を調べて、どのユーザ／グループ／クラスタ／ディビジョン／プラットフォーム／など（以下では「エンティティ」と称する）に権利を与えるかを判定し、送られた証明書を調べて、ライセンスリクエスタを識別する。それに基づいて、ライセンサによって、ＳＲＬ３０８にリストされた権利のどれをリクエスタに発行しなければならないかが判定される。概念上、ライセンサによって、ＳＲＬ３０８にリストされたエンティティが検査され、そのようなエンティティが、リクエスタと比較される。したがって、ＳＲＬ３０８によって、特定のグループがライセンスを受け取ることが指定され、リクエスタが、そのようなグループのメンバである場合に、リクエスタは、ＳＲＬ３０８のグループについて示された権利を伴うラインセスを許可される。同様に、ＳＲＬ３０８によって、特定のユーザが、ライセンスを受け取ることが指定され、リクエスタが、そのようなユーザである場合には、リクエスタは、ＳＲＬ３０８でそのようなユーザについて示された権利を伴うライセンスを許可される。諒解できるように、特定のＳＲＬ３０８によって、複数のエンティティおよびそれに対する権利をリストすることができ、特定のリクエスタに、１つまたは複数のエンティティのメンバであることに基づいてライセンスを許可することができる。

【００７２】

本発明の一実施形態では、図７に示されているように、リクエスタが、送られた証明書１２０２内で、識別子１２０４によって識別され、識別子１２０４は、たとえば、それを介してリクエスタが組織のディレクトリ１２０６内で識別されるエイリアスとすることができる。それに対応して、ＳＲＬ３０８によって、各権利を与えられるエンティティが、そのような識別子１２０４に従ってリストされる。したがって、ライセンス１２０８に関する要求の処理の一部として、ライセンサ１２１０（通常はＤＲＭサーバ３２０である）によって、証明書１２０２からリクエスタの識別子１２０４が入手され、入手された識別子１２０４が、送られたＳＲＬ３０８にリストされたすべての識別子１２０４と比較され

る。一致が見つかる場合には、ライセンサ１２１０によって、リクエスタの識別子１２０４についてＳＲＬ３０８で指定される権利を用いて、リクエスタにライセンス１２０８が発行される。

【００７３】

さらに、ディレクトリ１２０６の利用可能性を用いて、ディレクトリ１２０６に他のエンティティ内のリクエスタのメンバシップ状況を反映する適当な相互参照情報が含まれると仮定して、ライセンサ１２１０によって、リクエスタがＳＲＬ３０８にリストされた他のそのようなエンティティのメンバであるかどうかを判定することもできる。通常、ディレクトリ１２０６には、リクエスタごとに、その識別子１２０４だけではなく、リクエスタがそのメンバである各グループ／クラスタ／ディビジョン／プラットフォーム／他のエンティティ／などの識別子１２０８もリストされる。ディレクトリ１２０６に、メールアドレス、代替メールアドレス、ＩＤ、代替ＩＤ、グループメンバシップ、ヒストリック識別子、および／または類似物などの識別子１２０８を含めることができることに留意されたい。

【００７４】

証明書１２０２がその識別子１２０４と共にリクエスタから受け取られ、ＳＲＬ３０８からの権利データがリクエスタから受け取られた状態で、一般的に言って、とりわけ、リクエスタがＳＲＬ３０８にリストされたエンティティのメンバであることをディレクトリ１２０６から判定できる場合に、ライセンサ１２１０によって、リクエスタにライセンス１２０８が発行される。そうである場合に、ライセンシングに関するすべての他の条件が満足されると仮定して、ライセンサ１２１０によって、ＳＲＬ３０８から入手されたリストされたエンティティに関して指定された権利および条件を用いてリクエスタにライセンス１２０８が発行される。

【００７５】

本発明の一実施形態では、ＳＲＬ３０８にリストされたエンティティによって、個人のグループまたはクラスタあるいは他の集合（以下では「グループ」と称する）が表され、そのようなグループは、ディレクトリ１２０６を用いて適当に表現される。そのようなグループに、配布リストまたはメールエイリアスなどのメール対応グループ、あるいはネットワークオペレーティングシステムまたは類似物に関して定義することができるセキュリティグループを含めることができる。

【００７６】

特筆すべきことに、グループに従ってＳＲＬ３０８で権利を指定し、リクエスタのグループメンバシップに従ってリクエスタにライセンス１２０８を発行することによって、企業セッティングまたは組織セッティングでのディジタル権利管理が実現される。たとえば、ドキュメントまたは電子メールをＤＲＭ保護し、その結果、所与の部署のすべてのメンバが、ドキュメントまたは電子メールを読む権利を有するようにすることができる。そのような部署のグループ（たとえば電子メールエイリアス）が、組織のディレクトリ１２０６に存在すると仮定すると（ほとんどの場合にそうである）、ドキュメントまたは電子メールの作者は、個人ではなくグループに基づいて権利を与える。諒解できるように、そのようなグループ単位の権利付与の長所に、作者が権利を有する個人のクラスを指定する際に使用することの容易さが含まれる。さらに、グループに従って権利を指定することによって、指定された権利が、新しい個人がグループに参加し、古い個人がグループから離脱する際に、指定された権利が「古く」ならない。その代わりに、グループのすべての現在のメンバが、そのようなグループのメンバシップが組織ディレクトリ１２０６で最新状態に保たれる限り、権利を行使することができる。

【００７７】

リクエスタがグループのメンバであるかどうかの判定

通常、ディレクトリ１２０６は、組織にとって重要である有用な情報のリポジトリまたはその一部である。具体的に言うと、ディレクトリ１２０６によって、ユーザがメンバシップを有するすべてのグループを含む、組織内の各ユーザに関する複数の重要な情報が保

持される。実際、ユーザのそのようなメンバシップを、少なくとも部分的にユーザに従ってディレクトリ１２０６にリストすることによって、定義することができる。

【００７８】

組織が特に大きいか、他の理由で必要である情況では、その中でのコンピューティングを、ある論理的な形で複数のディビジョンまたは「フォレスト」に分割することができることに留意されたい。たとえば、各フォレストは、組織が、たとえばペンシルバニア州ウィルクスバリに１つのオフィスを有し、フロリダ州ボイントンビーチに別のオフィスを有する場合に、地理に基づくものとすることができ、この場合には、各そのようなオフィスに、別々のフォレストが与えられる。その代わりに、たとえば、各フォレストを、たとえば組織が会計課および生産課を有する場合に、機能に基づくものとすることができ、この場合には、各そのような部署に、別々のフォレストが与えられる。もちろん、フォレストは、本発明の趣旨および範囲から逸脱せずに、他の判断基準に従って定義することができる。

【００７９】

重要なことに、各フォレストがそれ自体のディレクトリ１２０６またはディレクトリ１２０６のグループ（以下では「ディレクトリ１２０６」と称する）を有する場合に、各フォレストが、少なくとも１つのＤＲＭサーバ３２０を有すると仮定する。図８を参照すると、２つのフォレスト、フォレストＡおよびフォレストＢが、組織について示されており、フォレストＡは、ディレクトリＡおよびＤＲＭサーバＡを有し、フォレストＢは、ディレクトリＢおよびＤＲＭサーバＢを有する。諒解されるように、各フォレストは、保護エンクレーブ（ｐｒｏｔｅｃｔｉｖｅ　ｅｎｃｌａｖｅ）として働き、この保護エンクレーブ内で、対応するディレクトリおよびＤＲＭサーバが展開される。したがって、フォレストＡ内のＤＲＭサーバＡは、少なくともセキュリティの観点から、比較的容易にディレクトリＡ内の情報にアクセスすることができなければならず、フォレストＢ内のＤＲＭサーバＢは、やはり少なくともセキュリティの観点から、ほぼ、そのように比較的容易に、フォレストＡ内のディレクトリＡ内の情報にアクセスすることができる。しかし、相互動作する必要を有する複数のフォレストでは、通常は、その間で情報が複製され、任意選択として、その間の明示的な信頼感形をセットアップすることができることに留意されたい。

【００８０】

フォレスト内のグループメンバシップ判定

さしあたり、組織の全体が図８のフォレストＡによって表されることと、図８のフォレストＢが存在しないことを仮定し、フォレストＡのディレクトリＡによって、すべてのユーザおよびこれらのユーザが属するすべてのグループを含む、組織に関係するすべてのディレクトリ情報が維持されることを諒解されたい。ここで、そのようなグループに、実際にはグループの複数のレイヤを含めることができることに留意されたい。したがって、図９からわかるように、グループ１が、直接のメンバとしてグループ２およびグループ３を有し、グループ３が、直接のメンバとしてユーザ１を有する場合がある。そのような場合には、ユーザ１が、グループ３の直接のメンバであり、グループ１の間接のメンバであり、グループ２のメンバでない。諒解できるように、ＳＲＬ３０８に、グループ１または３のメンバに関する権利がリストされる場合に、ユーザ１は、他のすべての条件が満たされると仮定して、ユーザ１が、そのような権利に基づいてライセンス１２０８を入手することができるが、ＳＲＬ３０８に、グループ２のメンバに関する権利だけがリストされている場合には、ユーザ１は、そのようなライセンス１２０８を入手することができない。一般的に言って、ＳＲＬ３０８で名前を指定されるメンバの直接のメンバまたは間接のメンバであるユーザは、どのような権利および条件がそのようなグループに許可される場合であっても、ライセンス１２０８を入手することができる。

【００８１】

図１０からわかるように、グループ１から３およびユーザ１のそれぞれに関するディレクトリＡ内のレコードオブジェクト／レコードが示されている。通常はそうであるが、各レコードは、エンティティがグループであると仮定して、エンティティのすべての直接の

メンバおよび、エンティティがユーザまたは別のグループのサブグループのいずれであれ、エンティティが所有するすべての直接のメンバシップの両方を含む、めいめいのエンティティに関するリスティングの形態である。したがって、グループ１は、直接のメンバとしてグループ２および３を有し、他のグループの直接のメンバではなく、グループ２は、メンバを持たず、グループ１の直接のメンバであり、グループ３は、直接のメンバとしてユーザ１を有し、グループ１の直接のメンバであり、ユーザとしてのユーザ１は、メンバを持たないが、グループ３の直接のメンバである。もちろん、図９および１０に示されたディレクトリＡは、全体に過剰に単純化されており、したがって、実際には、より多数のユーザおよびグループを有するはるかに大きいディレクトリである可能性が非常に高い。それでも、図示のディレクトリなどのディレクトリＡは、本発明を説明するのに適する。

【００８２】

ユーザ１がＤＲＭサーバＡにＳＲＬ３０８をサブミットする過程で（サブミットされるＳＲＬ３０８に１つまたは複数のグループおよびそれに関する権利がリストされる）、ＤＲＭサーバＡによって、少なくとも部分的に、ディレクトリＡを参照して、ユーザ１がＳＲＬ３０８にリストされたグループのいずれかの直接のメンバまたは間接のメンバであるかどうかを判定することによって、ライセンス１２０８をユーザ１に発行するかどうかが決定される。諒解されるように、そのような判定は、２つの一般的な形すなわち、リストされたグループからユーザへ、またはユーザからリストされたグループへ、実行することができる。

【００８３】

グループからユーザへの判定を実行することは、通常はより単純なプロセスであるが、グループが多数のメンバを有することができるので、見つかったすべてのユーザを調べて一致が存在するかどうかを判定することに、比較的長い時間が費やされる可能性が高い。ユーザからグループへの判定を実行することは、通常はより単純ではないプロセスであるが、ユーザが、通常は多数のグループのメンバではないので、見つかったすべてのグループを調べて一致が存在するかどうかを判定することに、比較的短い時間が費やされる可能性が高い。どの場合でも、ユーザからグループへの判定が、より経済的であり、総合的に望ましいことが、少なくとも経験的にわかっている。

【００８４】

図９および１０を参照しながら、図１２も参照すると、ユーザ１がグループ１のメンバであるかどうかに関するグループメンバシップ判定を行うために、ＤＲＭサーバＡは、下記のように進行する。準備として、ＤＲＭサーバＡによって、グループ１が存在するかどうかについてディレクトリＡが調べられる（ステップ１２０１）。諒解されるように、そのような照会に対する応答によって、グループ１が存在しないことが示される場合には、このプロセスは、ユーザ１が存在しないグループのメンバであることが不可能なので、終了する（ステップ１２０３）。したがって、判定がユーザからグループへ実行されるという事実にもかかわらず、グループが存在するかどうかを調べる準備ステップが、不要な場合の追加ステップの実行を避けるために実行される。

【００８５】

もちろん、図からわかるように、グループ１が実際に存在し、したがって、応答は、実際には肯定になる。したがって、ＤＲＭサーバＡによって、ユーザ１がメンバであるすべてのグループについてディレクトリＡが調べられ、ディレクトリＡによって、ユーザ１がグループ３のメンバであることの情報が返される（ステップ１２０５、１２０７）。もちろん、グループ３は、グループ１ではなく、したがって、ＤＲＭサーバＡは、ユーザ１がグループ１のメンバであるかどうかをまだ判定していない（ステップ１２０９）。それでも、グループ３の直接のメンバであることと、グループ３がグループ１の直接のメンバまたは間接のメンバであることとによって、ユーザ１がグループ１の間接のメンバであることが後にわかる可能性がある場合に、ＤＲＭサーバＡは継続しなければならない。

【００８６】

したがって、ＤＲＭサーバＡは、グループ３がメンバであるすべてのグループについて

ディレクトリＡに尋ねることによって継続し、ディレクトリＡによって、グループ３がグループ１のメンバであるという情報が返される（ステップ１２１１、１３１３）。その結果、ＤＲＭサーバＡは、グループ３のメンバであることによって、ユーザ１が実際にグループ１の間接のメンバであると判定する（ステップ１２１５）。

【００８７】

ここで諒解されるように、ＤＲＭサーバＡは、より多数のグループおよびより多数の反復を介して再帰的に照会して、最終的にユーザ１が実際にグループ１のメンバであることを判定しなければならない場合がある。代わりに、ユーザ１が、実際にはグループ１のメンバでない場合には、ＤＲＭサーバＡは、ユーザ１がグループ１のメンバでないと最終的に判定する前に、直接および間接の、ユーザ１に関するすべての可能なグループメンバシップの全体を介して照会しなければならない。下で詳細に説明するように、そのような照会は、高価になる可能性があり、連続的な基礎で実行される可能性があるので、そのような探査の結果を、１つまたは複数のキャッシュに保管することができる。

【００８８】

フォレストにまたがるグループメンバシップ判定

組織が、展開された複数のフォレストならびにフォレスト境界にまたがって使用可能にする必要があるユーザおよびグループメンバシップ情報を有する場合に、特に、照会するＤＲＭサーバ３２０が、そのディレクトリ内に必要な情報をネイティブに保持しないフォレストにある場合に、特定のグループに関するユーザのグループメンバシップを判定する単純な形はない。その代わりに、ＤＲＭサーバ３２０によって、そのような情報が実際にネイティブに保持される別のフォレストを調べなければならない。

【００８９】

具体的に言うと、特に重要なのは、図８のＤＲＭサーバＡなどのＤＲＭサーバ３２０が、グループをリストするＳＲＭ３０８に基づいてユーザにラインセンスを発行することを求められるが、グループが、ＤＲＭサーバＡが存在するフォレストＡにネイティブでない場合である。その代わりに、そのようなグループが、フォレストＢにネイティブであり、したがって、フォレストＢのディレクトリＢ内にレコードを有する。その結果、問題のユーザが問題のグループのメンバであるかどうかに関するグループメンバシップ判定を、そのディレクトリＢがそのグループのグループ情報を有する問題のグループのレコードを有する限り、フォレストＢのディレクトリＢに関して行わなければならない。

【００９０】

しかし、重要なことに、フォレストＡのＤＲＭサーバは、通常は、本質的なディレクトリ情報を得るためにフォレストＢのディレクトリＢに直接に照会することができない。というのは、そのようなクロスフォレスト照会が、通常は、セキュリティの懸念から制限されるか妨げられるからである。その代わりに、本発明の一実施形態では、ＤＲＭサーバＡは、フォレストＢのＤＲＭサーバＢに、フェレストＢのディレクトリＢを照会し、照会の結果をＤＲＭサーバＡに報告するように求める。ディレクトリＢへのＤＲＭサーバＢの照会が、クロスフォレスト照会ではなく、したがって、その制限をこうむらないことに留意されたい。

【００９１】

しかし、ＤＲＭサーバＡが、照会を実行するようにＤＲＭサーバＢに求めるのに先立って、問題のグループがフォレストＢにネイティブである限り、ＤＲＭサーバＢが、実際に、照会を実行するのに正しいＤＲＭサーバ３２０であることを実際に知らなければならないことを諒解しなければならない。簡単に言うと、ＤＲＭサーバＡは、ディレクトリＢに照会するようにＤＲＭサーバＢに求めるのに先立って、問題のグループがフォレストＢにネイティブであることを知らなければならない。しかし、ＤＲＭサーバＡは、ディレクトリＡに照会することだけができ、したがって、そのようなディレクトリＡが問題のグループのネイティブフォレストに関する情報を有しない限り、問題のグループのネイティブフォレストについて知らない。

【００９２】

したがって、本発明の一実施形態では、ディレクトリＡが、図８に示されているように、実際にそのような情報を有する。具体的に言うと、ディレクトリＡは、問題のグループに関するポインタオブジェクトを有し、このポインタオブジェクトによって、問題のグループに関して照会するすべてのエンティティが、フォレストＢに向けられる。そのようなポインタオブジェクトに、たとえば、ＤＲＭサーバＡが問題のグループに関する情報を要求する際に、照会するＤＲＭサーバＡに返されるフォレストＢのアドレスを含めることができる。おそらく、フォレストＢのディレクトリＢが、問題のグループに関するレコードオブジェクトを有する。したがって、ＤＲＭサーバＡは、そのようなアドレスを使用して、フォレストＢを見つけ、そこから進行することができる。

【００９３】

本発明の一実施形態では、フォレストＢのアドレスが、具体的には、フォレストＢ内のディレクトリＢのアドレスである。そのようなアドレスを有するＤＲＭサーバＡは、それに基づいてディレクトリＢに連絡する。しかし、ＤＲＭサーバＡは、フォレストＡ内にあり、ディレクトリＢは、フォレストＢ内にあるので、上でほのめかしたように、ディレクトリＢは、通常は、フォレストにまたがってＤＲＭサーバＡに本質的な情報を提供しない。それでも、ディレクトリＢは、ＤＲＭサーバＡにＤＲＭサーバＢのアドレスを提供することができる。

【００９４】

したがって、本発明の一実施形態では、ＤＲＭサーバＡが、ディレクトリＢからＤＲＭサーバＢのアドレスを受け取り、それに基づいて、ＤＲＭサーバＡの代わりにディレクトリＢに照会するようにＤＲＭサーバＢに求める。具体的にいうと、ＤＲＭサーバＡは、問題のユーザが問題のグループのメンバであるかどうかについてディレクトリＢを照会し、その回答をＤＲＭサーバＡに返すように、ＤＲＭサーバＢに要求する。おそらく、そのような要求には、たとえばＤＲＭサーバＢによって認識されるオーソリティの信頼されるルートにつながる証明書チェーンを有する識別する証明書など、ＤＲＭユニバース内のＤＲＭサーバ３２０としてのＤＲＭサーバＡのある種類の識別が含まれる。したがって、ＤＲＭサーバＢは、識別する証明書が検証される場合に限って、ＤＲＭサーバＡの代わりにディレクトリＢの照会を実行する。

【００９５】

ＤＲＭサーバＢが、図１２に関して上で示した形などで、ＤＲＭサーバＡの代わりにディレクトリＢの照会を実際に実行すると仮定すると、ＤＲＭサーバＢは、最終的に、問題のユーザが問題のグループのメンバであるかどうかを判定し、その情報を、ＤＲＭサーバＡからの要求に対する応答としてＤＲＭサーバＡに返す。

【００９６】

ＤＲＭサーバＢが、問題のユーザが問題のグループのユーザであるかどうかをディレクトリＢに尋ねる場合に、その問題のユーザが、レコードオブジェクトによらない場合には少なくともポインタオブジェクトによって、ディレクトリＢ内で表現されていなければならない。さらに、特に、上で説明したように、グループメンバシップがユーザからグループへ判定される場合にｍｅｍｂｅｒ－ｏｆ情報が参照される限り、そのようなポインタオブジェクトに、問題のユーザがそのメンバであるグループのそれぞれに関する情報が含まれなければならない。

【００９７】

図１３に移り、要約すると、ＤＲＭサーバＡおよびＤＲＭサーバＡの代わりのＤＲＭサーバＢは、フォレストにまたがってグループメンバシップを判定するために、下記のステップを実行する。

【００９８】

予備的に、ＤＲＭサーバＡによって、あるコンテンツに対応する使用ライセンス１２０８に関するユーザからの要求が受け取られる（ステップ１３０１）。この要求には、ユーザを識別する証明書およびコンテンツに関連するＳＲＬ３０８が含まれる。これに応答して、ＳＲＬ３０８によってその中のグループが識別されると仮定し、適当なキャッシング

された情報が存在しないことも仮定すると、ＤＲＭサーバＡによって、ディレクトリＡが照会されて、識別されたグループのオブジェクトが返される（ステップ１３０３）。現在のシナリオでは、グループがフォレストＢにネイティブであり、したがって、グループが実際にフォレストＢをポイントするポインタオブジェクトとしてディレクトリＡ内で表され、ＤＲＭサーバＡが、ディレクトリＡからフォレストＢをポイントするポインタオブジェクトを受け取ると仮定され、したがって、このポインタによって、グループがフォレストＢにネイティブであることが示され（ステップ１３０４）、受け取られたポインタオブジェクトからフォレストＢのアドレスが検索される（ステップ１３０５）。

【００９９】

その後、ＤＲＭサーバＡによって、検索されたアドレスが使用されて、ＤＲＭサーバＢのアドレスについてフォレストＢ内のディレクトリＢが照会され、ディレクトリＢによって、そのようなアドレスが、それに応答してＤＲＭサーバＡに返される（ステップ１３０７、１３０９）。ＤＲＭサーバＢのアドレスを得る特定のメカニズムは、既知であるか、当業者に明白であり、したがって、本明細書で詳細に説明する必要はない。したがって、本発明の趣旨および範囲から逸脱せずに、任意の適当なメカニズムを使用することができる。

【０１００】

ＤＲＭサーバＢのアドレスを用いて、ＤＲＭサーバＡによって、ＤＲＭサーバＢに、そのユーザがそのグループのメンバであるかどうかをディレクトリＢに照会するように要求する（ステップ１３１１）。やはり、要求をＤＲＭサーバＢに配信するのにＤＲＭサーバＡによって使用される特定のメカニズムは、既知であるか、当業者に明白であり、したがって、本明細書で詳細に説明する必要はない。したがって、本発明の趣旨および範囲から逸脱せずに、任意の適当なメカニズムを使用することができる。

【０１０１】

ＤＲＭサーバＡに関して、やはり適当なキャッシングされた情報が存在しないと仮定すると、ＤＲＭサーバＢは、ディレクトリＢに照会して、識別されたグループのすべてのオブジェクトを返す（ステップ１３１３）。このシナリオでは、グループが、フォレストＢのネイティブであり、したがって、そのようなグループは、実際に、レコードオブジェクトとしてディレクトリＢ内で表され、ＤＲＭサーバＢは、ディレクトリＢからそのレコードオブジェクトを受け取ると仮定され、したがって、グループがフォレストＢのネイティブであることが示される（ステップ１３１５）。ＤＲＭサーバＢによって、この点で、受け取られたオブジェクトで示されるグループのメンバを再検討して、ユーザがそのグループの直接のメンバであるかどうかを判定する（ステップ１３１７）ことができることに留意されたい。そうである場合には、判定は肯定であり、ＤＲＭサーバＢによって、それがＤＲＭサーバＡに報告される（ステップ１３１９）。そうでない場合には、ＤＲＭサーバＢは、ディレクトリＢに照会してユーザに関するすべてのオブジェクトを返し、したがって、図１２に関して示した形に似た形で、ディレクトリＢ内でユーザからグループへトラバースすることを試みることによる、ユーザがグループのメンバであるかどうかの判定に進む（ステップ１３２３）。試みられたトラバーサルを終了した際に、ＤＲＭサーバＢは、ユーザとグループの間にメンバシップ関係が存在するかどうかを判定しており、これをＤＲＭサーバＡに報告する（ステップ１３２５）。

【０１０２】

もちろん、そのような判定に基づいて、ＤＲＭサーバＡによって、ステップ１３０１で受け取られたユーザからの要求を尊重するかどうかが判断される（ステップ１３２７）。諒解されるように、ユーザがグループのメンバでない場合には、もちろん、ユーザが受け取られたＳＲＬ３０８で識別される別のグループのメンバであることがわかった場合を除いて、要求が拒否される。同様に、ユーザがメンバのグループである場合には、要求を尊重するための他のすべての条件が満たされると仮定して、要求が尊重される。

【０１０３】

キャッシングされた情報によるユーザグループメンバシップの判定

これまでに上述したように、ＤＲＭサーバは、ＳＲＬ３０８に基づいてユーザにライセンス１２０８を発行する過程で、ＤＲＭサーバ３２０のフォレストまたは別のフォレスト内のディレクトリ１２０６を参照することによって、ユーザがＳＲＬ３０８で識別されるグループのメンバであるかどうかを判定することができる。しかし、諒解されるように、判定を行うたびにディレクトリ１２０６を参照することは、特に追加のネットワークトラフィックおよび、特に行われる判定の数が比較的多い場合のディレクトリ１２０６によって実行される必要がある作業に関して、かなり高価であり、面倒である。

【０１０４】

したがって、本発明の一実施形態では、１つまたは複数のディレクトリ１２０６から導出されるユーザ－グループ情報が、ＤＲＭサーバ３２０からアクセス可能な１つまたは複数の位置でキャッシングされる。さらに、本発明の一実施形態では、ＤＲＭサーバ３２０は、ユーザ－グループ判定を行う場合に、ｌｏｗｅｓｔ－ｃｏｓｔ－ｆｉｒｓｔアルゴリズムを使用して、ディレクトリ１２０６への参照に先立って、最低コストから最高コストへの順で複数のキャッシュ位置を再検討する。したがって、より高いコストの動作が、できる限り延期され、判定速度が高まり、ネットワークトラフィックが減る。

【０１０５】

ユーザ－グループ情報のキャッシング

図１２に関して開示した判定など、ユーザがグループのメンバであるかどうかをディレクトリ１２０６から判定する過程で、ＤＲＭサーバ３２０は、潜在的に、ユーザが直接のメンバまたは間接のメンバである少なくともいくつかの他のグループを突き止める。メンバシップパスが実際に見つかったならばトラバーサルが停止する限り、ユーザのすべてのグループが見つからない可能性が高いことに留意されたい。たとえば、問題のグループのレコードオブジェクトの検査で、ユーザがそのグループの直接のメンバであることがわかった場合に、そのユーザが直接のメンバまたは間接のメンバである他のグループは、見つけられない。同様に、ユーザとグループの間のメンバシップパスを見つける試みでユーザからグループにトラバースする必要がある場合に、少なくとも、ユーザが直接のメンバであるグループが見つかるが、いくつかの間接グループは、事前にメンバシップパスが見つかったので見つけられない場合がある。それでも、本発明の一実施形態では、ＤＲＭサーバ３２０によって、ユーザならびにその直接のグループメンバシップおよび間接のグループメンバシップに関してＤＲＭサーバが有するすべての知識が、将来に必要になった場合に備えて１つまたは複数のキャッシュに保管される。

【０１０６】

図１４を参照すると、キャッシュ１４０４の複数のエントリ１４０２が示されている。具体的に言うと、キャッシュ１４０４の１つのエントリ１４０２から、ユーザ３について、間接的にまたは直接に、のいずれかで、そのようなユーザ３がグループ４、６、および７のメンバであり、グループ４が、グループ６および７のメンバであることが見つかっていることがわかる。同様に、キャッシュ１４０４の別のエントリ１４０２から、ユーザ４について、そのユーザ４がグループ８のみのメンバであることが見つかっていることがわかる。そのようなユーザ３および４ならびにグループ４がそのグループの直接のメンバまたは間接のメンバであるかどうかは、重要でなく、したがって、詳細に叙述されないが、必要または有用であることがわかった場合には、そのような情報をそのように詳細に叙述することができる。重要なのは、ユーザ／グループのすべてのグループメンバシップが、キャッシングされることである。というのは、特定のユーザ－グループ関係が、後の時点で、同一のユーザ／グループまたは異なるユーザ／グループについてユーザ－グループ関係を確立するのに有用になる可能性があるからである。

【０１０７】

諒解されるように、ディレクトリ１２０６から入手されるキャッシュ１４０４内の情報は、ある期間の時間の後に「古く」なる可能性がある。すなわち、経時的に、ユーザが、ディレクトリ１２０６から削除される可能性があり、追加のグループおよび追加のグループメンバシップが、定義される可能性があり、同様の可能性があり、それでも、キャッシ

ュ１４０４は、そのように変更された情報を反映するように自動的には更新されない。したがって、図１４からわかるように、キャッシュ１４０４の各エントリ１４０２には、作成の時刻が含まれる。そのような作成時刻に基づいて、キャッシュ１４０４を照会する照会するエンティティは、エントリ１４０４が、頼るには古すぎるかどうかを判断することができる。それに加えてまたは代わりに、キャッシュ１４０４が、それ自体で、作成時刻に基づいて古すぎると思われる、そのキャッシュ内の各エントリ１４０４を削除することができる。

【０１０８】

図１５および１６に移ると、図１４のキャッシュ１４０４を、複数の構成で実施することができる。１つの構成では、あるフォレスト内のライセンスを発行するＤＲＭサーバ３２０のそれぞれが、比較的基本的な軽量インメモリキャッシュであるキャッシュ１４０４を有し、フォレストも、比較的拡張されたより重いインメモリキャッシュであるキャッシュ１４０４を有する専用キャッシュサーバ１５０２（図１５）を有する。さらに、専用キャッシュサーバ１５０２によって、データベースキャッシュ１４０４を使用すること、および／またはライセンスを発行するＤＲＭサーバ３２０の代わりにディレクトリ１２０６に対する照会を実行することを行うことができる。

【０１０９】

もう１つの構成では、あるフォレストのライセンスを発行するＤＲＭサーバ３２０のそれぞれが、比較的拡張された重量インメモリキャッシュ（図１６）を有し、ディレクトリ１２０６に対する照会を実行する。専用キャッシュサーバ１５０２は存在しないが、データベースキャッシュ１４０４を、そのフォレスト内のすべてのライセンスを発行するＤＲＭサーバ３２０の間で共有することができる。どの構成でも、各基本キャッシュ１４０４は、ユーザ情報だけを有するが、各拡張キャッシュ１４０４は、ユーザ情報およびグループ情報を有する。

【０１１０】

ｌｏｗｅｓｔ－ｃｏｓｔ－ｆｉｒｓｔアルゴリズム

キャッシュ１４０４の構成にかかわらず、ユーザがグループのメンバであるかどうかを判定しようとするＤＲＭサーバ３２０は、本発明のｌｏｗｅｓｔ－ｃｏｓｔ－ｆｉｒｓｔアルゴリズムを使用して、最低コストから最高コストの順でキャッシュ１４０４およびディレクトリ１２０６を検索する。仮定として、ＤＲＭサーバ３２０に関連するキャッシュ１４０４は、最低のコストを関連付けられ、専用サーバ１５０２に関連するキャッシュ１４０４およびデータベースキャッシュ１４０４は、より高いコストを関連付けられ、ディレクトリ１２０６は、最高のコストを関連付けられる。もちろん、コストを主観的とすることができ、いずれにしても、特定のエンティティが、本発明の趣旨および範囲から逸脱せずに、特定のコストを有することができる。

【０１１１】

図１７Ａおよび１７Ｂに移ると、このアルゴリズムの主要な特性は、より高いコストを有すると思われる動作が、より低いコストを有すると思われるすべての動作が使い尽くされる前には試行されないことであることがわかる。一般に、コストの昇順の動作が、ライセンシングＤＲＭサーバ３２０のインメモリキャッシュ１４０４内でルックアップされ、データベースキャッシュ１４０４または専用キャッシュサーバ１５０２のキャッシュ１４０４内でルックアップされ、ＡＤ内でルックアップされる。このアルゴリズムを実行する際に、ユーザおよび１つまたは複数のターゲットグループが、サブミットされ、ユーザが、直接にまたは間接的にのいずれかで、ターゲットグループのいずれかのメンバであるかどうかの判定が行われる。本質的に、このアルゴリズムでは、やはり高コスト動作を実行する前に必ず低コスト動作を実行して、キャッシュ１４０４およびディレクトリ１２０６からのすべての使用可能なメンバシップ情報によってユーザからターゲットグループのいずれかへトラバースする。このアルゴリズムでは、ユーザがターゲットグループの１つのメンバである場合に「ｙｅｓ」が結果として返され、ユーザがすべてのターゲットグループのメンバでない場合に「ｎｏ」が返され、ユーザがディレクトリ１２０６内にない場合

に「ｕｓｅｒ　ｎｏｔ　ｆｏｕｎｄ」が返され、ターゲットグループのどれもが、ディレクトリ１２０６内で見つからない場合に「ｎｏ　ｔａｒｇｅｔ　ｇｒｏｕｐｓ　ａｒｅ　ｆｏｕｎｄ」が返される。

【０１１２】

図１７Ａからわかるように、このプロセスは、問題のユーザおよび１つまたは複数のターゲットグループを定義することによって開始される（ステップ１７０１）。とりあえず、ＤＲＭサーバ３２０が、ローカルキャッシュ１４０４を有し、リモートキャッシュ１４０４（専用キャッシュサーバ１５０２またはデータベースキャッシュ１４０４とすることができる）およびディレクトリ１２０６へのアクセスを有すると仮定すると、ライセンスを発行するＤＲＭサーバ３２０によって使用されるアルゴリズムでは、下記の複数のビンおよびキューが使用される。

・１つまたは複数のターゲットグループがその中に常駐するターゲットビン（Ｔ）

・検証される１つまたは複数のターゲットグループがその中に常駐する検証済みターゲットビン（Ｖ）

・検索を待っているグループがその中に常駐する検索ビン（Ｓ）

・ローカルキャッシュ１４０４内で検索されるグループがその中に常駐するローカルキュー（Ｑ１）

・リモートキャッシュ１４０４内で検索されるグループがその中に常駐するリモートキュー（Ｑ２）

・ディレクトリ１２０６内で検索されるグループがその中に常駐するディレクトリキュー（Ｑ３）

・既に処理されたグループがその中に常駐する破棄ビン（Ｄ）。

【０１１３】

したがって、このアルゴリズムは、ターゲットビン内のターゲットグループごとに、そのターゲットグループのレコードまたはエントリ１４０２を見つけるための検索することによって、１つまたは複数のターゲットグループを検証することによって進行する。本質的に、同一のプロセスが、各ストレージデバイス（ローカルキャッシュ１４０４に対応するＳ１、リモートキャッシュ１４０４に対応するＳ２、ディレクトリ１２０６に対応するＳ３）に関して実行されるので、そのようなプロセスは、包括的なストレージＳｘに関して説明されることだけを必要とする。

【０１１４】

具体的に言うと、このアルゴリズムでは、まず、Ｓ１すなわちローカルキャッシュ１４０４を調べ、ターゲットビン（Ｔ）内のターゲットグループごとに、そのようなターゲットグループがローカルキャッシュ１４０４内のエントリ１４０２として見つかる場合に、そのターゲットグループが（Ｔ）から除去され、そのターゲットグループが、検証済みターゲットビン（Ｖ）に置かれ、そのターゲットグループのエントリ１４０２が、下位レベルキャッシュ１４０４に置かれるが、下位レベルキャッシュは、この例では存在しない（ステップ１７０３）。その後、このアルゴリズムでは、Ｓ２すなわちリモートキャッシュ１４０４に関してステップ１７０３を繰り返し、（Ｔ）に残っているターゲットグループごとに、そのようなターゲットグループがリモートキャッシュ１４０４内のエントリ１４０２として見つかる場合に、ターゲットグループを（Ｔ）から除去し、そのターゲットグループを（Ｖ）に置き、そのターゲットグループのエントリ１４０２を下位レベルキャッシュ１４０４に置くが、この下位レベルキャッシュは、この場合にはローカルキャッシュ１４０４である。最後に、このアルゴリズムは、Ｓ３すなわちディレクトリ１２０６に関してステップ１７０３をもう一度繰り返し、（Ｔ）にまだ残っているターゲットグループごとに、そのようなターゲットグループがディレクトリ１２０６内のレコードとして見つかる場合に、そのターゲットレコードを（Ｔ）から除去し、そのターゲットレコードを（Ｖ）に置き、そのターゲットグループのエントリ１４０２を下位レベルキャッシュ１４０４に置くが、この下位レベルキャッシュは、この場合にはローカルキャッシュおよびリモートキャッシュ１４０４である。

【０１１５】
ストレージデバイスごとにステップ１７０３を実行した際に、検証済みターゲットビン（Ｖ）に、キャッシュ１４０４またはディレクトリ１２０６のいずれかに存在することがわかったすべてのターゲットグループが含まれなければならない。ターゲットビンに残っているすべてのターゲットグループは、不良またはもはや存在しなと仮定することができ、したがって、無視することができる。（Ｖ）が空である場合には、検証済みターゲットグループが存在しないので、このアルゴリズムは終了し、このアルゴリズムによって、「ｎｏ ｔａｒｇｅｔ ｇｒｏｕｐｓ ａｒｅ ｆｏｕｎｄ」が返される（ステップ１７０５）。注記すべきこととして、レコードまたはエントリ１４０２をディレクトリ１２０６またはリモートキャッシュ１４０４からすべての下位レベルキャッシュ１４０４にコピーすることによって、そのような下位レベルキャッシュ１４０４が、ユーザがグループのメンバであるかどうかの将来の判定に使用することができるキャッシュ情報で満たされる。

【０１１６】
その後、少なくとも１つの検証済みターゲットが（Ｖ）に存在すると仮定すると、このアルゴリズムによって、ユーザが検証される。具体的に言うと、このアルゴリズムでは、まずＳ１すなわちローカルキャッシュ１４０４、次にＳ２すなわちリモートキャッシュ１４０４、次にＳ３すなわちディレクトリを調べることによって、問題のユーザに関するレコードまたはエントリ１４０２を検索する（ステップ１７０９）。ステップ１７０９の反復のいずれかでユーザに関するレコードまたはエントリ１４０２（以下では「アイテム」と称する）が見つかる際に、このアルゴリズムでは、ユーザがそのメンバであるグループのそれぞれが、検索ビン（Ｓ）に置かれ（ステップ１７１１）、さらに、ユーザのエントリ１４０２が、すべての下位レベルキャッシュ１４０４に移植される（ステップ１７１３）。ユーザに関するアイテムが、ステップ１７０９のすべての反復から見つからない場合には、検証されたユーザが存在しないので、このアルゴリズムは終了し、このアルゴリズムによって、「ｕｓｅｒ ｎｏｔ ｆｏｕｎｄ」が返される（ステップ１７１５）ことに留意されたい。

【０１１７】
諒解されるように、ステップ１７０９は、最低コスト動作から最高コスト動作に対応すると知覚される順序で反復され、この順序は、この例ではＳ１、次にＳ２、次にＳ３に関する。したがって、ユーザが検証されたならば、より高いコストの動作が回避される。本発明の趣旨および範囲から逸脱せずに、１つまたは複数のターゲットグループに先立ってユーザを検証することができることに留意されたい。

【０１１８】
（Ｓ）にあるグループによって表される少なくとも１つのグループのメンバである、検証されたユーザが見つかり、少なくとも１つの検証済みターゲットグループが、（Ｖ）の対応するアイテムによって表されるターゲットグループとして見つかると仮定し、図１７Ｂに移ると、このアルゴリズムは、（Ｓ）および（Ｖ）から、（Ｓ）のグループが（Ｖ）のグループ（すなわち、１つまたは複数の検証済みターゲットグループ）と一致するかどうかを判定することによって継続される（ステップ１７２３）。実際に、（Ｓ）のグループが（Ｖ）のアイテムと一致することがわかった場合には、このアルゴリズムは、一致と共に終了し、このアルゴリズムによって、「ｙｅｓ」が返される（ステップ１７２５）。そうでない場合には、このアルゴリズムは、（Ｓ）のすべてのグループをローカルキュー（Ｑ１）に移動することによって継続される（ステップ１７２７）。

【０１１９】
その後、このアルゴリズムでは、反復プロセスを実行して、ユーザからすべてのターゲットグループへのメンバシップパスについて検索する。この反復プロセスでは、グループメンバシップを判定するために、ローカルキャッシュおよびリモートキャッシュ１４０４ならびにディレクトリ１２０６での複数の検索が必要である。本質的に同一のプロセスが、キュー（ローカルキャッシュ１４０４に対応するＱ１、リモートキャッシュ１４０４に対応するＱ２、ディレクトリ１２０６に対応するＱ３）のそれぞれに関して実行されるの

で、そのようなプロセスは、図１７Ｂに示されているように、包括的な（Ｑｘ）および包括的なストレージＳｘに関して説明することだけが必要であり、ここで、Ｓ１はローカルキャッシュ１４０４、Ｓ２はリモートキャッシュ１４０４、Ｓ３はディレクトリ１２０６である。

【０１２０】

具体的に言うと、さしあたり図１７Ｂの（Ｑｘ）およびＳｘが（Ｑ１）およびＳ１すなわちローカルキャッシュ１４０４であると仮定すると、（Ｑ１）が空であるかどうかの判定が、まず行われる（ステップ１７２９）。もちろん、（Ｑ１）は、当初は空であってはならないが、このプロセスは反復的なので、（Ｑ１）が実際に空になる点が来る可能性がある。（Ｑ１）が、当初は空でなく、１グループを有すると仮定すると、そのようなグループが、選択され、（Ｑ１）から除去され、そのグループが既に破棄ビン（Ｄ）にあるかどうかが判定される（ステップ１７３１）。当初は、もちろん、選択され（Ｑ１）から除去されるグループは、そのグループがまだ処理されていない限りは（Ｄ）に含まれないが、このプロセスが反復される際に、おそらくは前に処理されていないグループを含む他のグループが（Ｑ１）に置かれる可能性があり、各グループが処理される際に、そのグループが実際に（Ｄ）に移動される。

【０１２１】

選択され（Ｑ１）から除去されるグループが、既に処理されており、したがって（Ｄ）に含まれる場合には、このアルゴリズムは、ステップ（１７２９）に戻り、もう一度（Ｑ１）が空であるかどうかを判定する。しかし、選択され（Ｑ１）から除去されるグループが、まだ処理されておらず、したがって（Ｄ）に含まれない場合には、このアルゴリズムは、選択されたグループのエントリ１４０２を、（Ｑ１）に対応するキャッシュ１４０４またはディレクトリ１２０６内で検索することによって継続され、このキャッシュまたはディレクトリは、この例ではローカルキャッシュ１４０４である（ステップ１７３３）。

【０１２２】

選択されたグループに関するそのようなエントリ１４０２がローカルキャッシュ１４０４で見つからない場合には、このプロセスは、そのグループを（Ｑ１）からリモートキュー（Ｑ２）に移動することと（ステップ１７３５）、ステップ（１７２９）に戻ることによって継続され、このステップ（１７２９）では、もう一度（Ｑ１）が空であるかどうかを判定する。しかし、選択されたグループに関するそのようなエントリ１４０２がローカルキャッシュ１４０４内で見つかる場合に、このアルゴリズムでは、選択されたグループがそのメンバである新たに発見されたグループのそれぞれを、検索ビン（Ｓ）に置き、選択されたグループを廃棄ビン（Ｄ）に移動し（ステップ１７３９）、さらに、すべての下位レベルキャッシュ１４０４に、選択されたグループに対応するエントリ１４０２を移植する（ステップ１７３７）。もちろん、（Ｑ１）およびローカルキャッシュ１４０４に関して、そのような下位レベルキャッシュ１４０４が存在しないことに留意されたい。その後、このアルゴリズムは、ステップ１７２３に戻ることによって継続され、このステップ１７２３では、（Ｓ）のグループのいずれか（すなわち、新たに発見されたグループ）が（Ｖ）のグループ（すなわち、１つまたは複数の検証済みターゲットグループ）と一致するかどうかが判定される。もちろん、実際に（Ｓ）のグループが（Ｖ）のアイテムと一致することがわかった場合には、このアルゴリズムは、一致で終了し、このアルゴリズムによって、ステップ１７２５で「ｙｅｓ」が返される。そうでない場合には、このアルゴリズムは、ステップ１７２７でＳのすべてのグループを（Ｑ１）に移動することによって継続され、このアルゴリズムは、継続される。

【０１２３】

それを行う際に、このアルゴリズムでは、（Ｑ１）によって、ユーザから検証済みターゲットグループのいずれかへ、ローカルキャッシュ１４０４内で可能な範囲まで、すべての可能なパスを展開する。諒解されるように、一致が見つからないと仮定すると、（Ｑ１）は、プロセスが反復される際により新たに発見されたグループで満たされ、リモートキャッシュ１４０４によって処理される破棄ビン（Ｄ）または（Ｑ２）のいずれかへローカ

ルキャッシュ１４０４によって（Ｑ１）のすべてのグループが処理されるまで、空にされる。したがって、より高コストと仮定されるリモートキャッシュ１４０４での動作は、ローカルキャッシュ１４０４でのすべての可能な動作が使い果たされる（ステップ１７２９で（Ｑ１）が空であることがわかる場合に発生する）までは行われない。

【０１２４】

実際に、ステップ１７２９で（Ｑ１）が空であることがわかる場合に、処理は、（Ｑ１）に関して行われる形に似た形で、リモートキュー（Ｑ２）に向かう。具体的に言うと、（Ｑｘ）が図１７ＢのＳｘであり、（Ｑ２）がＳ２すなわちリモートキャッシュ１４０４であると仮定すると、まず、（Ｑ２）が空であるかどうかの判定が行われる（ステップ１７２９）。ここで、（Ｑ２）は、当初は空でない可能性が高いが、このような事象は、ローカルキュー１４０４自体が（Ｑ１）からのすべてのグループを処理できる場合に発生する可能性がある。（Ｑ２）が、当初は空ではなく、１グループを有すると仮定すると、そのようなグループが、選択され、（Ｑ２）から除去され、そのグループが、既に処理されたので破棄ビン（Ｄ）に既にあるかどうかを判定する（ステップ１７３１）。

【０１２５】

選択され（Ｑ２）から除去されたグループが、既に処理されており、したがって（Ｄ）にある場合には、このアルゴリズムは、ステップ（１７２９）に戻り、ここで、（Ｑ２）が空であるかどうかをもう一度判定する。しかし、選択され（Ｑ２）から除去されたグループが、まだ処理されておらず、したがって（Ｄ）にない場合には、このアルゴリズムは、（Ｑ２）に対応するキャッシュ１４０４またはディレクトリ１２０６（この例ではリモートキャッシュ１４０４）の選択されたグループに関するエントリ１４０２を検索することによって継続される（ステップ１７３３）。

【０１２６】

選択されたグループに関するそのようなエントリ１４０２がリモートキャッシュ１４０４で見つからない場合には、このプロセスは、グループを（Ｑ２）からディレクトリキュー（Ｑ３）に移動することによって継続され（ステップ１７３５）、その後、ステップ（１７２９）に戻って、（Ｑ２）が空であるかどうかをもう一度判定する。しかし、選択されたグループに関するそのようなエントリ１４０２がリモートキャッシュ１４０４で見つかる場合には、このアルゴリズムでは、選択されたグループがそのメンバである新たに発見されたグループのそれぞれを検索ビン（Ｓ）に置き、選択されたグループを破棄ビン（Ｄ）に移動し（ステップ１７３９）、さらに、選択されたグループに対応するエントリ１４０２をすべての下位レベルキャッシュ１４０４に移植する（ステップ１７３７）。もちろん、（Ｑ２）およびリモートキャッシュ１４０４に関して、ローカルキャッシュ１４０４だけがそのように移植される必要があることに留意されたい。その後、このアルゴリズムは、ステップ１７２３に戻ることによって継続され、そのステップで、（Ｓ）のグループ（すなわち新たに発見されたグループ）が、（Ｖ）のグループ（すなわち、１つまたは複数の検証済みターゲットグループ）と一致するかどうかが判定される。もちろん、（Ｓ）のグループが、実際に（Ｖ）のアイテムと一致することがわかった場合には、このアルゴリズムは、一致で終了し、このアルゴリズムによって、ステップ１７２５で「ｙｅｓ」が返される。そうでない場合には、このアルゴリズムは、ステップ１７２７でＳのすべてのグループを（Ｑ１）に移動することによって継続され、このアルゴリズムは、継続され、（Ｑ１）が処理される。

【０１２７】

それを行う際に、このアルゴリズムでは、（Ｑ２）によって、ユーザから検証済みターゲットグループのいずれかへ、リモートキャッシュ１４０４内で可能な範囲まで、すべての可能なパスを展開する。諒解されるように、一致が見つからないと仮定すると、（Ｑ２）は、プロセスが反復される際により新たに発見されたグループで満たされ、ディレクトリ１４０４によって処理される破棄ビン（Ｄ）または（Ｑ３）のいずれかへリモートキャッシュ１４０４によって（Ｑ２）のすべてのグループが処理されるまで、空にされる。したがって、より高コストと仮定されるディレクトリ１２０６での動作は、リモートキャッ

シュ１４０４でのすべての可能な動作が使い果たされる（（Ｑ２）が空である場合に発生する）までは行われない。

【０１２８】

重要なことに、（Ｑ２）およびリモートキャッシュ１４０４によって新たに発見されるすべてのグループが、まず、（Ｑ１）およびローカルキャッシュ１４０４によって処理される。したがって、やはり、より高コストと仮定されるリモートキャッシュ１４０４での動作は、ローカルキャッシュ１４０４での可能なすべての動作が使い果たされる（（Ｑ１）が空の場合に発生する）までは、新たに発見されたグループに関して行われない。

【０１２９】

実際に、ステップ１７２９で（Ｑ２）が空であることがわかる場合に、処理は、（Ｑ１）および（Ｑ２）に関して行われる形に似た形で、ディレクトリキュー（Ｑ３）に向かう。具体的に言うと、（Ｑｘ）が図１７ＢのＳｘであり、（Ｑ３）がＳ３すなわちディレクトリ１２０６であると仮定すると、まず、（Ｑ３）が空であるかどうかの判定が行われる（ステップ１７２９）。ここで、やはり、（Ｑ３）は、当初は空でない可能性が高いが、このような事象は、ローカルキューおよびリモートキュー１４０４自体が（Ｑ１）および（Ｑ２）からのすべてのグループを処理できる場合に発生する可能性がある。（Ｑ３）が、当初は空ではなく、１グループを有すると仮定すると、そのようなグループが、選択され、（Ｑ３）から除去され、そのグループが、既に処理されたので破棄ビン（Ｄ）に既にあるかどうかが判定される（ステップ１７３１）。

【０１３０】

選択され（Ｑ３）から除去されたグループが、既に処理されており、したがって（Ｄ）にある場合には、このアルゴリズムは、ステップ（１７２９）に戻り、ここで、（Ｑ３）が空であるかどうかをもう一度判定する。しかし、選択され（Ｑ３）から除去されたグループが、まだ処理されておらず、したがって（Ｄ）にない場合には、このアルゴリズムは、（Ｑ３）に対応するキャッシュ１４０４またはディレクトリ１２０６（この例ではディレクトリ１２０６）の選択されたグループに関するエントリ１４０２を検索することによって継続される（ステップ１７３３）。

【０１３１】

選択されたグループに関するそのようなエントリ１４０２がディレクトリ１２０６で見つからない場合には、このプロセスは、グループを（Ｑ３）から（Ｄ）に移動することによって継続され（ステップ１７３５）、その後、ステップ（１７２９）に戻って、（Ｑ３）が空であるかどうかをもう一度判定する。（Ｑ３）の後には次のキューがないので、選択されたグループをさらに処理することができないことに留意されたい。しかし、選択されたグループに関するそのようなエントリ１４０２がリモートキャッシュ１４０４で見つかる場合には、このアルゴリズムでは、選択されたグループがそのメンバである新たに発見されたグループのそれぞれを検索ビン（Ｓ）に置き、選択されたグループを破棄ビン（Ｄ）に移動し（ステップ１７３９）、さらに、選択されたグループに対応するエントリ１４０２をすべての下位レベルキャッシュ１４０４に移植する（ステップ１７３７）。もちろん、（Ｑ３）およびディレクトリ１２０６に関して、ローカルキャッシュおよびリモートキャッシュ１４０４だけがそのように移植される必要があることに留意されたい。その後、このアルゴリズムは、ステップ１７２３に戻ることによって継続され、そのステップで、（Ｓ）のグループ（すなわち新たに発見されたグループ）が、（Ｖ）のグループ（すなわち、１つまたは複数の検証済みターゲットグループ）と一致するかどうかが判定される。もちろん、（Ｓ）のグループが、実際に（Ｖ）のアイテムと一致することがわかった場合には、このアルゴリズムは、一致で終了し、このアルゴリズムによって、ステップ１７２５で「ｙｅｓ」が返される。そうでない場合には、このアルゴリズムは、ステップ１７２７でＳのすべてのグループをローカルキュー（Ｑ１）に移動することによって継続され、このアルゴリズムは、継続され、（Ｑ１）が処理される。

【０１３２】

それを行う際に、このアルゴリズムでは、（Ｑ３）によって、ユーザから検証済みター

ゲットグループのいずれかへ、ディレクトリ１２０６で可能な範囲まで、すべての可能なパスを展開する。諒解されるように、一致が見つからないと仮定すると、（Ｑ３）は、プロセスが反復される際により新たに発見されたグループで満たされ、破棄ビン（Ｄ）へディレクトリ１２０６によって（Ｑ３）のすべてのグループが処理されるまで、空にされる。

【０１３３】

重要なことに、（Ｑ３）およびディレクトリ１２０６によって新たに発見されるすべてのグループが、まず、（Ｑ１）およびローカルキャッシュ１４０４によって処理される。したがって、やはり、より高コストと仮定されるディレクトリ１２０６での動作は、ローカルキャッシュおよびリモートキャッシュ１４０４での可能なすべての動作が使い果たされる（（Ｑ１）および（Ｑ２）が空の場合に発生する）までは、新たに発見されたグループに関して行われない。

【０１３４】

実際に、ステップ１７２９で、（Ｑ３）が空であることがわかった場合には、処理は、この特定のシナリオで（Ｑ４）がない限り、これ以上進行することはできず、したがって、完了する。具体的に言うと、このアルゴリズムは、一致なしで終了し、このアルゴリズムによって、「ｎｏ」が返される（ステップ１７４１）。

【０１３５】

本発明のアルゴリズムが、より高コストの動作を実行する前に、まずより低コストの動作を実行するように設計されていることを諒解されたい。諒解されねばならぬように、このアルゴリズムは、３レベルのストレージ（すなわち、ローカルキャッシュおよびリモートキャッシュ１４０４ならびにディレクトリ１２０６）に関して開示されたが、その代わりに、２、４、５、６、７、および類似する数を含む任意の他のレベル数のストレージに、同様の数のキューを定義することによって適用することができる。したがって、本発明のアルゴリズムは、本発明の趣旨および範囲から逸脱せずに、キャッシングされたユーザーグループ情報のすべての複数レベルストレージ配置に適用することができる。

【０１３６】

結論

本発明に関連して実行されるプロセスを実現するのに必要なプログラミングは、比較的単純であり、当業者に明白である。したがって、そのようなプログラミングは、本明細書に添付しない。任意の特定のプログラミングを使用して、本発明の趣旨および範囲から逸脱せずに本発明を実現することができる。

【０１３７】

本発明では、ディジタル権利管理（ＤＲＭ）およびディジタル権利実施のアーキテクチャおよび方法によって、任意の形態のディジタルコンテンツの制御されたレンダリングまたは制御されたプレイが可能になり、そのような制御は、柔軟であり、そのようなディジタルコンテンツのコンテンツオーナー／デベロッパによって定義可能である。このアーキテクチャによって、そのような制御されたレンダリングが、特にオフィスまたは組織環境あるいは、ドキュメントが個人の異なるグループまたは個人のクラスの間で共有される場合に、可能になり、促進される。そのようなアーキテクチャによって、コンテンツのライセンスがフォレスト中から得られる。

【０１３８】

本発明の概念から逸脱せずに、上述の実施形態に対して変更を行うことができることを諒解されたい。たとえば、本開示では、ユーザからグループへのグループメンバシップ判定を説明したが、代わりに、そのような判定を、本発明の趣旨および範囲から逸脱せずに、グループからユーザへ行うことができる。したがって、本発明が開示された特定の実施形態に限定されるのではなく、請求項によって定義される本発明の趣旨および範囲に含まれる修正形態を含むことが意図されていることを理解されたい。

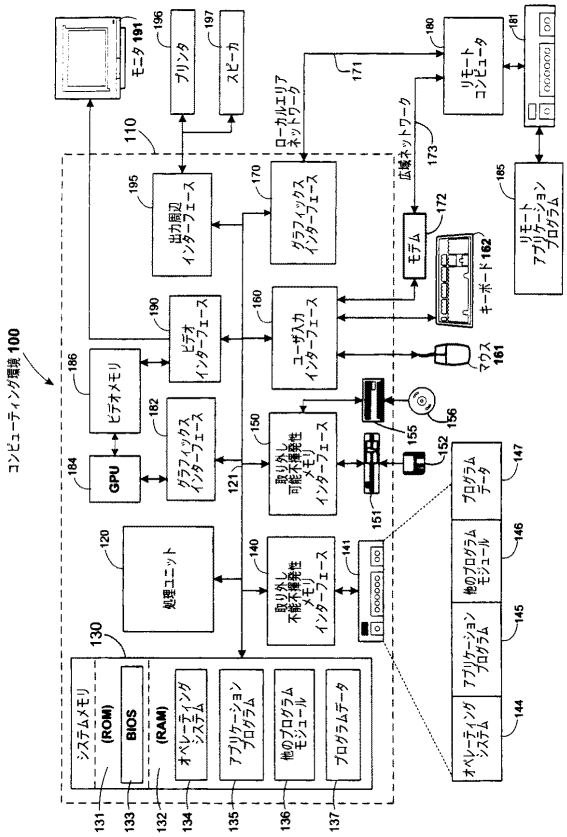
【図面の簡単な説明】

【０１３９】

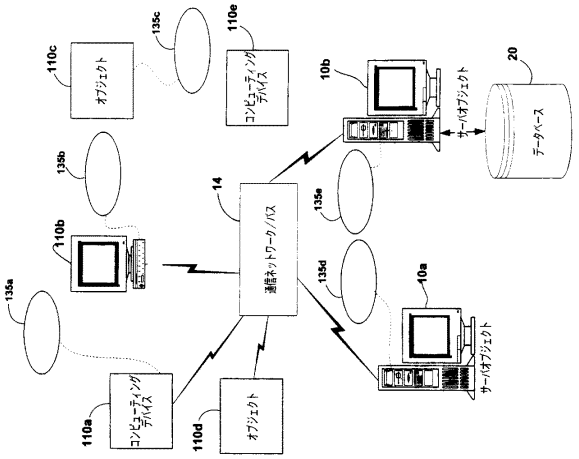
【図１】本発明を実施することができる例示的であり非制限的なコンピューティング環境を表すブロック図である。
【図２】本発明を実施することができるさまざまなコンピューティングデバイスを有する例示的なネットワーク環境を表すブロック図である。
【図３】ディジタルコンテンツをパブリッシュする、本発明によるシステムおよび方法の好ましい実施形態を示す機能ブロック図である。
【図４】権利管理されたディジタルコンテンツをパブリッシュする、本発明による方法の好ましい実施形態を示す流れ図である。
【図４Ａ】図４の方法によって作られる署名付き権利ラベルの構造を示すブロック図である。
【図５】権利管理されたディジタルコンテンツをライセンシングする、本発明によるシステムおよび方法の好ましい実施形態を示すブロック図である。
【図６Ａ】権利管理されたディジタルコンテンツをライセンシングする、本発明による方法の好ましい実施形態を示す流れ図である。
【図６Ｂ】権利管理されたディジタルコンテンツをライセンシングする、本発明による方法の好ましい実施形態を示す流れ図である。
【図７】本発明の一実施形態による、リクエスタによってライセンサに送られる証明書および権利ラベルと、組織ディレクトリを参照するライセンサを示すブロック図である。
【図８】複数のディビジョンまたはフォレストに編成された組織であって、各フォレストが、本発明の一実施形態による、少なくとも１つのＤＲＭサーバおよびディレクトリを有する、組織を示すブロック図である。
【図９】定義されたグループメンバシップによって関連するさまざまなエンティティを示すブロック図である。
【図１０】図９に示されたエンティティに関する、図８のディレクトリ内のディレクトリエントリを示すブロック図である。
【図１１】信頼ベースシステムの例の実施アーキテクチャを示すブロック図である。
【図１２】ユーザがＤＲＭサーバのフォレストにネイティブであるグループのメンバであるかどうかを判定する場合に図８のＤＲＭサーバによって実行されるキーステップを示す流れ図である。
【図１３】本発明の一実施形態による、ユーザがＤＲＭサーバのフォレストにネイティブでないグループのメンバであるかどうかを判定する場合に図８のＤＲＭサーバによって実行されるキーステップを示す流れ図である。
【図１４】本発明の一実施形態によるキャッシュ内のエントリを示すブロック図である。
【図１５】本発明の一実施形態による、図１４のキャッシュを使用するキャッシュアーキテクチャの変形形態を示すブロック図である。
【図１６】本発明の一実施形態による、図１４のキャッシュを使用するキャッシュアーキテクチャの変形形態を示すブロック図である。
【図１７Ａ】ユーザが本発明の一実施形態によるグループのメンバであるかどうかを判定するために図１５および１６のキャッシュアーキテクチャと共に図８のライセンシングＤＲＭサーバによって使用されるアルゴリズムによって実行されるキーステップを示す流れ図である。
【図１７Ｂ】ユーザが本発明の一実施形態によるグループのメンバであるかどうかを判定するために図１５および１６のキャッシュアーキテクチャと共に図８のライセンシングＤＲＭサーバによって使用されるアルゴリズムによって実行されるキーステップを示す流れ図である。
【符号の説明】
【０１４０】
３００　クライアント
３０２　コンテンツ準備アプリケーション
３０４　暗号化されたディジタルコンテンツファイル

３０６　ディジタル権利管理（ＤＲＭ）アプリケーションプログラムインターフェース（ＡＰＩ）
３０８　署名付き権利ラベル（ＳＲＬ）
３１０　権利管理されたコンテンツファイル
３２０　ＤＲＭサーバ
３３０　通信ネットワーク

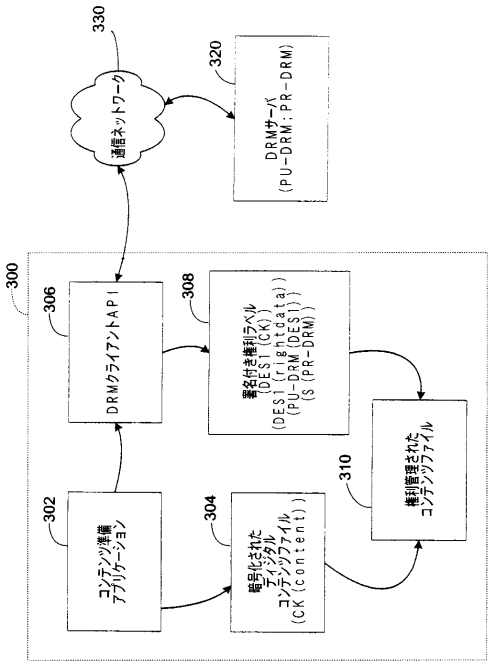
【図１】



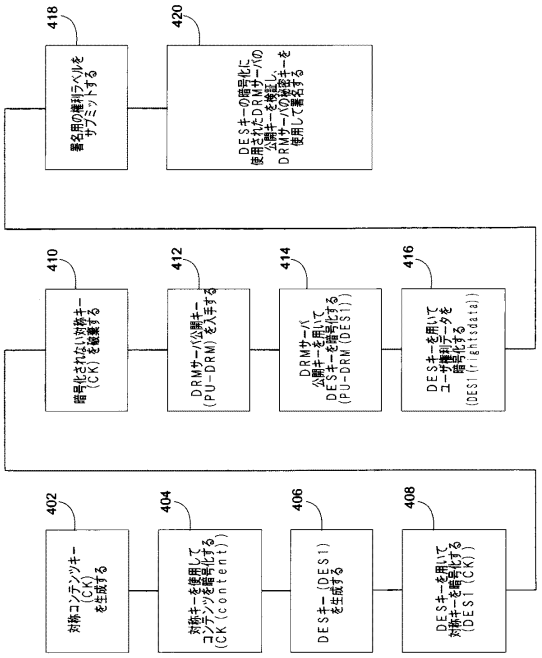
【図２】



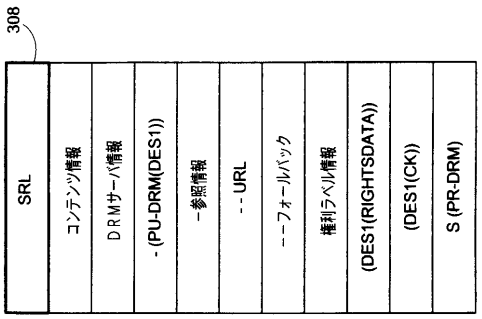
【図３】


330
通信ネットワーク
320
DRMサーバ
(PU-DRM;PR-DRM)
300
306
DRMクライアントAPI
308
署名付き権利ラベル
(DES1(rightsdata))
(PU-DRM(DES1))
(S(PR-DRM))
302
コンテンツ準備
アプリケーション
304
暗号化された
ディジタル
コンテンツファイル
(CK(content))
310
権利管理された
コンテンツファイル

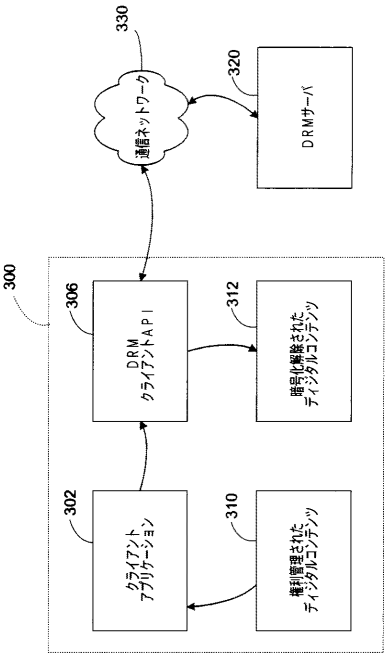
【図４】


418
署名用の権利ラベルを
サブミットする
420
DESキーの暗号化に
使用されたDRMサーバの
公開キーを検証し、
DRMサーバの秘密キーを
使用して署名する
410
暗号化されない対称キー
(CK)を破棄する
412
DRMサーバ(公開キー
(PU-DRM)を入手する
414
DRMサーバ
公開キーを用いて
DESキーを暗号化する
(PU-DRM(DES1))
416
DESキーを用いて
ユーザ権利データを
暗号化する
(DES1(rightsdata))
402
対称コンテンツキー
(CK)
を生成する
404
対称キーを使用して
コンテンツを暗号化する
(CK(content))
406
DESキー(DES1)
を生成する
408
DESキーを用いて
対称キーを暗号化する
(DES1(CK))

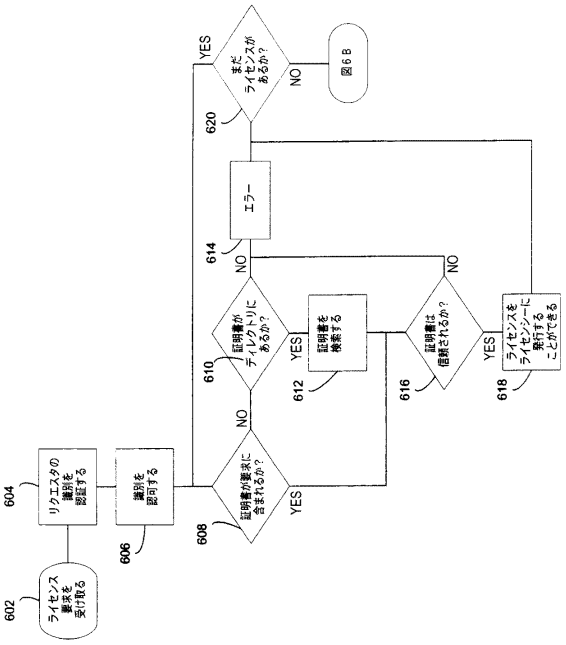
【図４Ａ】


308
SRL
コンテンツ情報
DRMサーバ情報
-(PU-DRM(DES1))
-参照情報
--URL
--フォールバック
権利ラベル情報
(DES1(RIGHTSDATA))
(DES1(CK))
S(PR-DRM)

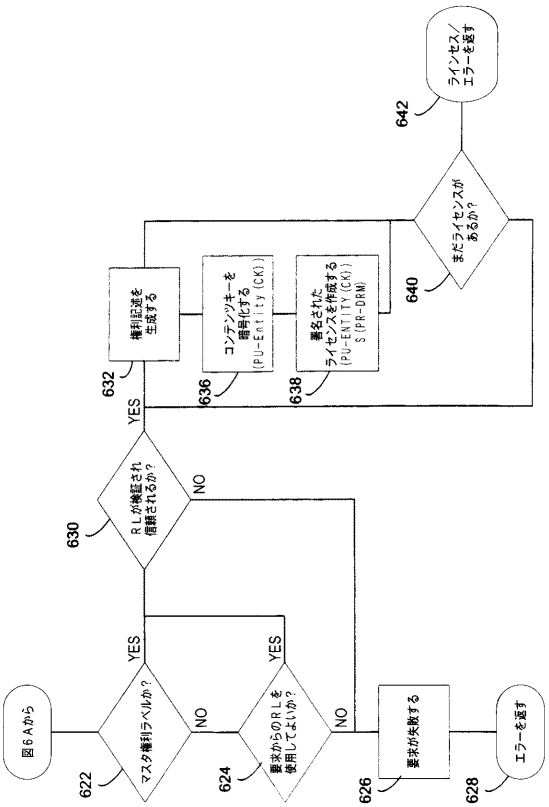
【図５】


330
通信ネットワーク
320
DRMサーバ
300
306
DRM
クライアントAPI
312
暗号化解除された
ディジタルコンテンツ
302
クライアント
アプリケーション
310
権利管理された
ディジタルコンテンツ

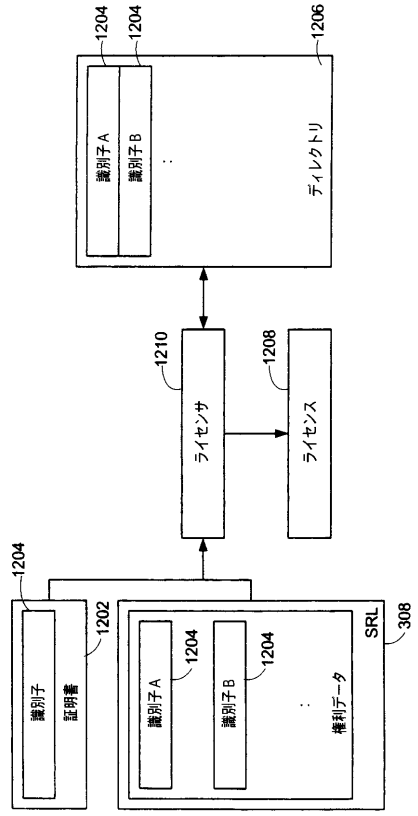
【図６Ａ】



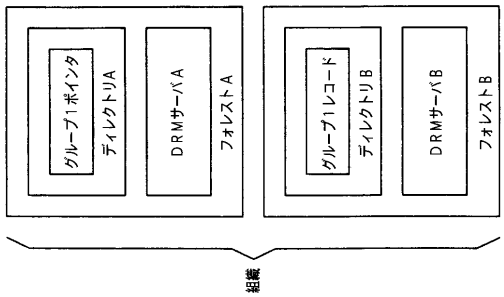
【図６Ｂ】



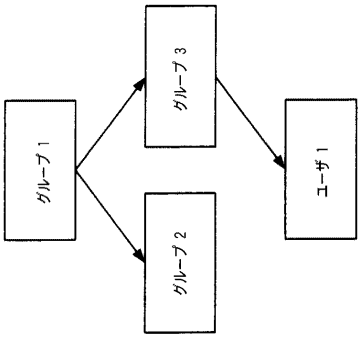
【図７】



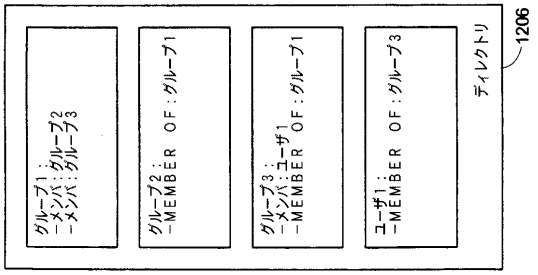
【図８】



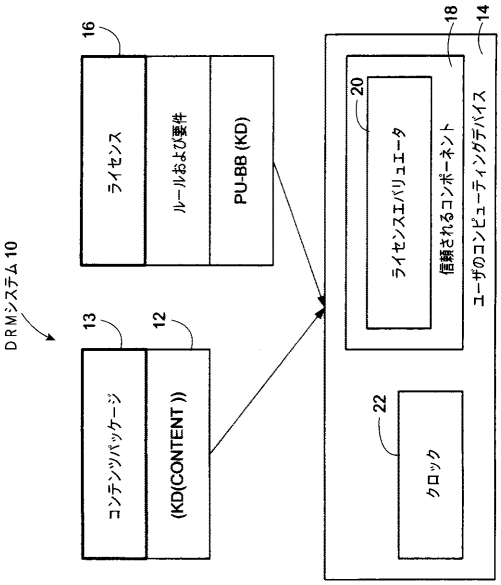
【図９】



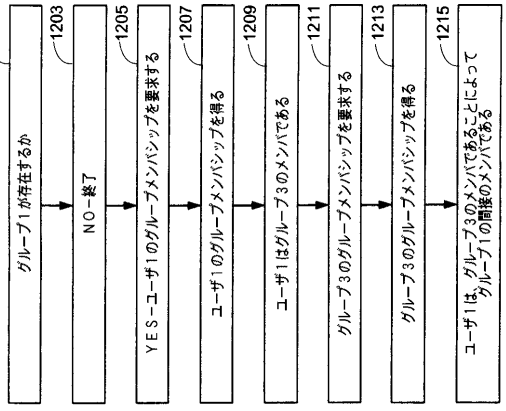
【図１０】



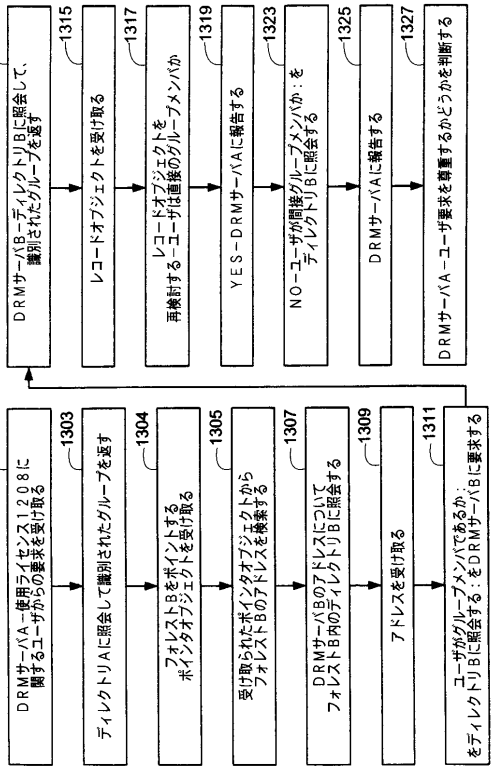
【図１１】



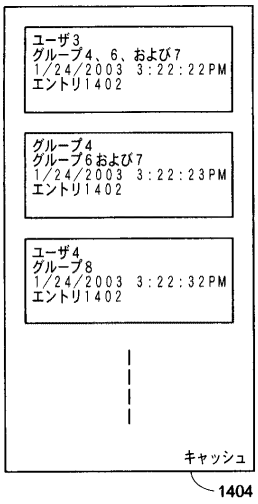
【図１２】



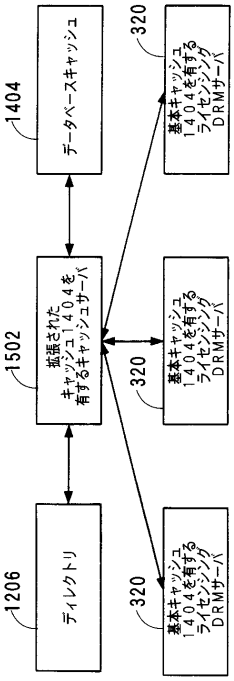
【図１３】



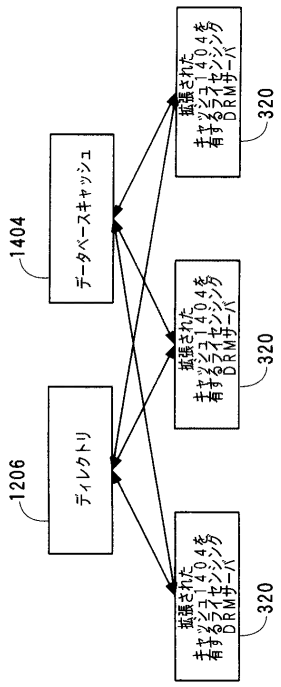
【図１４】



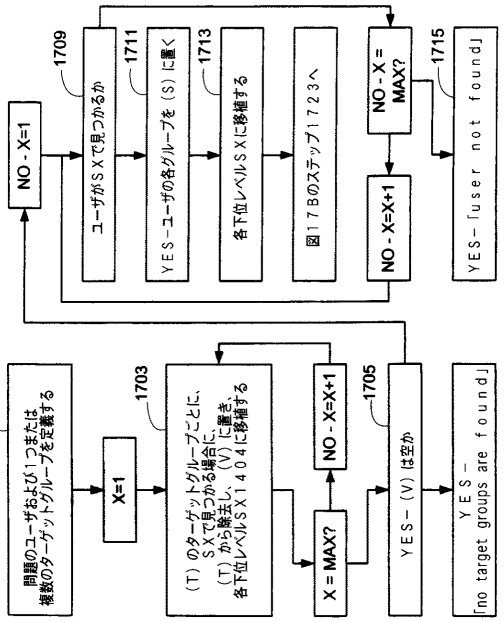
【図１５】



【図１６】



【図１７Ａ】



【図１７Ｂ】

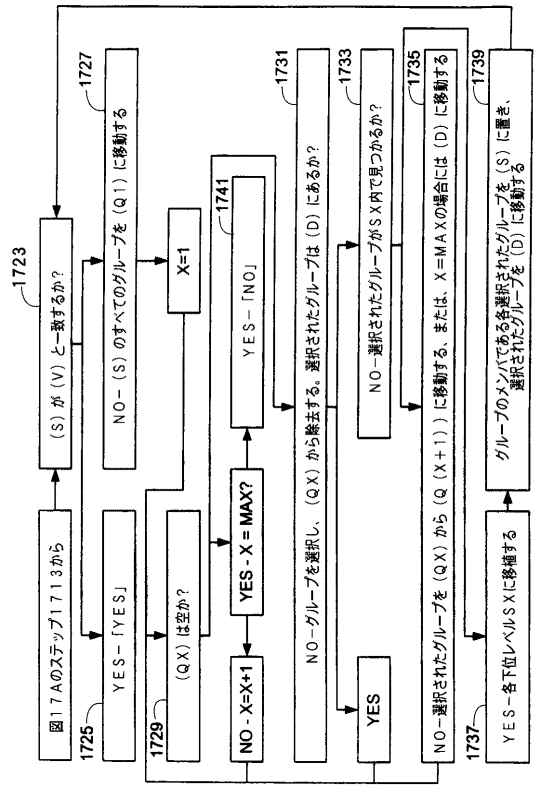


図17Aのステップ1713から
(S)が(V)と一致するか?
1723
YES-「YES」
1725
NO-(S)のすべてのグループを(Q1)に移動する
1727
(QX)は空か?
1729
X=1
NO-X=X+1
YES-X=MAX?
YES-「NO」
1741
NO-グループを選択し、(QX)から除去する。選択されたグループは(D)にあるか?
1731
YES
NO-選択されたグループがSX内で見つかるか?
1733
NO-選択されたグループを(QX)から(Q(X+1))に移動する。または、X=MAXの場合には(D)に移動する
1735
YES-各下位レベルSXに移植する
1737
グループのメンバである各選択されたグループを(S)に置き、
選択されたグループを(D)に移動する
1739

フロントページの続き

(72)発明者　チャンドラモウリ　ベンカテシュ
アメリカ合衆国　９８０７４　ワシントン州　サマミッシュ　２１３　プレイス　サウスイースト　４１４
(72)発明者　エフゲニー　（ユージン）　ローゼンフェルド
アメリカ合衆国　９８００７　ワシントン州　ベルビュー　ノースイースト　１３　プレイス　１５２０２　ナンバー２７１４
(72)発明者　アチッラ　ナリン
アメリカ合衆国　９８０１１　ワシントン州　ボセル　ノースイースト　１４４　コート　８７４１

審査官　篠原　功一

(56)参考文献　米国特許出願公開第２００３／００１８４９１（ＵＳ，Ａ１）

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