

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2009-151560
(P2009-151560A)

(43) 公開日 平成21年7月9日(2009.7.9)

(51) Int.Cl.

G06F 9/50 (2006.01)

F I

G06F 9/46 462 Z

テーマコード (参考)

審査請求 未請求 請求項の数 11 O L (全 26 頁)

(21) 出願番号 特願2007-329043 (P2007-329043)
(22) 出願日 平成19年12月20日 (2007.12.20)

(71) 出願人 000005108
株式会社日立製作所
東京都千代田区丸の内一丁目6番6号
(74) 代理人 110000176
一色国際特許業務法人
(72) 発明者 加藤 英晴
神奈川県川崎市麻生区王禅寺1099番地
株式会社日立製作所システム開発研究所内
(72) 発明者 増岡 義政
神奈川県川崎市麻生区王禅寺1099番地
株式会社日立製作所システム開発研究所内

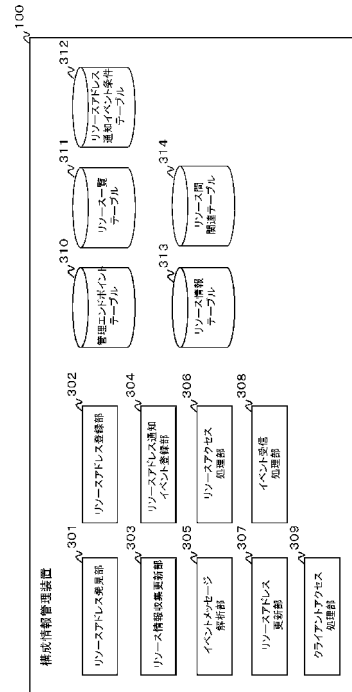
(54) 【発明の名称】 リソースの管理方法、情報処理システム、情報処理装置、及びプログラム

(57) 【要約】

【課題】 情報処理システムにおけるリソースを効率よく適切に管理する。

【解決手段】 情報処理システム1において、構成情報管理装置100が管理対象装置101のリソースの識別子と管理対象リソースのリソースアドレスとを対応づけて記憶し、管理対象装置101が管理対象リソースからイベント通知を受信する際に用いられる宛先情報（構成情報管理装置100のアドレス）を当該管理対象リソースの識別子と対応づけて記憶し、管理対象装置101が、構成情報管理装置100に管理対象リソースのイベント通知を送信するに際しリソースの識別子及び現在（最新）のリソースアドレスを送信し、構成情報管理装置100がイベント通知、リソースの識別子、及び現在のリソースアドレスを受信して、取得したリソースの識別子に対応づけて記憶しているリソースアドレスを、取得した現在のリソースアドレスに変更する。

【選択図】 図3



【特許請求の範囲】**【請求項 1】**

第 2 の情報処理装置と通信可能に接続する第 1 の情報処理装置が、
管理対象である前記第 2 の情報処理装置のリソースの識別子と前記リソースを特定する
情報であるリソースアドレスとを対応づけて記憶するステップと、
前記リソースからイベント通知を受信する際に用いられる宛先情報を当該リソースの識
別子と対応づけて前記第 2 の情報処理装置に記憶させるステップと、
前記リソースからのイベント通知の受信に際し、当該リソースの識別子と当該リソース
に現在付与されているリソースアドレスとを取得するステップと、
取得した前記リソースの識別子に対応づけて記憶している前記リソースアドレスを、取
得した前記現在のリソースアドレスに変更するステップと
を実行することを特徴とするリソースの管理方法。

10

【請求項 2】

請求項 1 に記載のリソースの管理方法であって、
前記第 1 の情報処理装置が、前記記憶しているリソースアドレスに対応づけてそのリ
ソースアドレスの有効/無効を示す情報を記憶するステップをさらに含むこと
を特徴とするリソースの管理方法。

【請求項 3】

請求項 1 に記載のリソースの管理方法であって、
前記第 1 の情報処理装置が、前記第 2 の情報処理装置に対し、前記イベント通知を前記
第 1 の情報処理装置に送信するか否かを決定する条件として、前記第 1 の情報処理装置が
記憶している前記リソースアドレスと前記現在のリソースアドレスとが異なるという条件
を前記宛先情報に対応づけて記憶させるステップをさらに含むこと
を特徴とするリソースの管理方法。

20

【請求項 4】

請求項 1 に記載のリソースの管理方法であって、
前記リソースアドレスは、WS-Addressing仕様に基づいて記述されること
を特徴とするリソースの管理方法。

【請求項 5】

通信可能に接続された複数の情報処理装置を含んで構成される情報処理システムにおけ
る情報処理装置のリソースの管理方法であって、
第 1 の情報処理装置が、管理対象となる第 2 の情報処理装置のリソースの識別子と前記
リソースを特定するリソースアドレスとを対応づけて記憶するステップと、
第 2 の情報処理装置が、前記リソースからイベント通知を受信する際に用いられる宛先
情報を当該リソースの識別子と対応づけて記憶するステップと、
前記第 2 の情報処理装置が、前記第 1 の情報処理装置に前記リソースについてのイベン
ト通知を送信する際、当該リソースの識別子、及び当該リソースに現在付与されているリ
ソースアドレスを送信するステップと、
前記第 1 の情報処理装置が、前記リソースからのイベント通知の受信に際し、前記イベ
ント通知、前記リソースの識別子、及び前記現在のリソースアドレスを受信するステップ
と、
前記第 1 の情報処理装置が、取得した前記リソースの識別子に対応づけて記憶している
前記リソースアドレスを、取得した前記現在のリソースアドレスに変更するステップと
を含むことを特徴とするリソースの管理方法。

30

40

【請求項 6】

請求項 5 に記載のリソースの管理方法であって、
前記第 2 の情報処理装置が、前記リソースからイベント通知を受信する際に用いられる
宛先情報を当該リソースの識別子と対応づけて記憶する前記ステップは、当該第 2 の情報
処理装置で動作している標準管理エージェントの機能を利用して行われること
を特徴とするリソースの管理方法。

50

【請求項 7】

請求項 5 に記載のリソースの管理方法であって、

前記第 2 の情報処理装置が、前記第 1 の情報処理装置に前記リソースについてのイベント通知を送信する際、当該リソースの識別子、及び当該リソースに現在付与されているリソースアドレスを送信する前記ステップは、当該第 2 の情報処理装置で動作している標準管理エージェントの機能を利用して行われること

を特徴とするリソースの管理方法。

【請求項 8】

通信可能に接続された複数の情報処理装置を含んで構成される情報処理システムであって、

第 1 の情報処理装置が、管理対象となる第 2 の情報処理装置のリソースの識別子と前記リソースを特定する情報であるリソースアドレスとを対応づけて記憶する手段と、

第 2 の情報処理装置が、前記リソースからイベント通知を受信する際に用いられる宛先情報を当該リソースの識別子と対応づけて記憶する手段と、

前記第 2 の情報処理装置が、前記第 1 の情報処理装置に前記リソースについてのイベント通知を送信する際、当該リソースの識別子、及び当該リソースに現在付与されているリソースアドレスを送信する手段と、

前記第 1 の情報処理装置が、前記イベント通知、前記リソースの識別子、及び前記現在のリソースアドレスを受信する手段と、

前記第 1 の情報処理装置が、取得した前記リソースの識別子に対応づけて記憶している前記リソースアドレスを、取得した前記現在のリソースアドレスに変更する手段と

を備えることを特徴とする情報処理システム。

【請求項 9】

請求項 8 に記載の情報処理システムにおける前記第 1 の情報処理装置の前記各手段を備えることを特徴とする情報処理装置。

【請求項 10】

請求項 8 に記載の情報処理システムにおける前記第 2 の情報処理装置の前記各手段を備えることを特徴とする情報処理装置。

【請求項 11】

第 2 の情報処理装置と通信可能に接続する第 1 の情報処理装置に、

管理対象である前記第 2 の情報処理装置のリソースの識別子と前記リソースを特定する情報であるリソースアドレスとを対応づけて記憶する機能と、

前記リソースからイベント通知を受信する際に用いられる宛先情報を当該リソースの識別子と対応づけて前記第 2 の情報処理装置に記憶させる機能と、

前記リソースからのイベント通知の受信に際し、当該リソースの識別子と当該リソースに現在付与されているリソースアドレスとを取得する機能と、

取得した前記リソースの識別子に対応づけて記憶している前記リソースアドレスを、取得した前記現在のリソースアドレスに変更する機能と

を実現するためのプログラム。

【発明の詳細な説明】**【技術分野】****【0001】**

この発明は、情報処理システムにおける情報処理装置のリソースの管理方法に関し、とくに管理対象であるリソースを識別又は特定する情報が変化する場合でもリソースを効率よく適切に管理できるようにする技術に関する。

【背景技術】**【0002】**

昨今、企業等における情報処理システムは大規模化 / 複雑化しており、情報処理システムが保有するリソースを効率よく簡単に運用し管理することが求められている。また、管理対象となる情報処理装置の多様化 / マルチベンダ化に伴い、情報処理システムの構成が

10

20

30

40

50

変化した場合にこれらリソースを効率よく迅速に対応できることが求められている。

【 0 0 0 3 】

ところで、情報処理システムにおける情報処理装置、オペレーティングシステム、アプリケーションソフトウェアなどのリソースの管理に際しては、管理対象となるリソースを確実に識別し特定する必要がある、リソースを特定する情報（以下、リソースアドレスと称する。）を適切に管理する必要がある（例えば特許文献 1 を参照）。

【特許文献 1】米国特許出願公開第20060136585号公報

【非特許文献 1】CMDBf (CMDB Federation)、[online]、[平成 1 9 年 1 1 月 2 2 日検索]、インターネット <URL:http://cmdbf.org/ >

【非特許文献 2】Distributed Management Task Force, Inc.、"Common Information Model (CIM) Standards"、[online]、[平成 1 9 年 1 1 月 2 2 日検索]、インターネット <URL:http://www.dmtf.org/standards/CIM/ >

10

【発明の開示】

【発明が解決しようとする課題】

【 0 0 0 4 】

しかしながら、大規模化 / 複雑化した情報処理システムにおいてリソースアドレスを適切に管理することは必ずしも容易ではない。例えば管理対象装置が移設されると IP アドレスやホスト名が変更されてしまうことがある。また NIC (Network Interface Card) が差し替えられれば MAC アドレスが変化する。また管理対象装置が仮想マシンであれば、いわゆるコールドマイグレーションが行われると管理対象装置の MAC アドレス (仮想 MAC アドレス) や IP アドレスが変化する。

20

【 0 0 0 5 】

従来、このようにリソースアドレスが変化する場合には、オペレータ等が手動でリソースアドレスの更新を行っており、大規模化 / 複雑化した情報処理システムにおいてはこのような更新作業により多大な運用負担が生じることとなる。

【 0 0 0 6 】

本発明はこのような事情に基づいてなされたもので、管理対象であるリソースを識別又は特定する情報が変化する場合でもリソースを効率よく適切に管理することが可能なリソースの管理方法、情報処理システム、情報処理装置、及びプログラムを提供することを目的とする。

30

【課題を解決するための手段】

【 0 0 0 7 】

上記課題を解決するための本発明のうちの主たる発明の一つは、リソースの管理方法であって、第 2 の情報処理装置と通信可能に接続する第 1 の情報処理装置が、管理対象である前記第 2 の情報処理装置のリソースの識別子と前記リソースを特定する情報であるリソースアドレスとを対応づけて記憶するステップと、前記リソースからイベント通知を受信する際に用いられる宛先情報を当該リソースの識別子と対応づけて前記第 2 の情報処理装置に記憶させるステップと、前記リソースからのイベント通知の受信に際し、当該リソースの識別子と当該リソースに現在付与されているリソースアドレスとを取得するステップと、取得した前記リソースの識別子と対応づけて記憶している前記リソースアドレスを、取得した前記現在のリソースアドレスに変更するステップとを実行することとする。

40

【発明の効果】

【 0 0 0 8 】

本発明によれば、管理対象であるリソースを識別又は特定する情報が変化する場合でもリソースを効率よく適切に管理することができる。

【発明を実施するための最良の形態】

【 0 0 0 9 】

以下、図面を参照しつつ実施の形態について詳細に説明する。

【 0 0 1 0 】

[実施例 1]

50

図 1 に本発明の実施の形態として説明する情報処理システム 1 の構成を示している。同図に示すように、この情報処理システム 1 は、通信ネットワーク 103 を介して通信可能に接続された、構成情報管理装置 100 (第 1 の情報処理装置)、構成情報管理装置 100 の管理対象となる情報処理装置である一台以上の管理対象装置 101 (第 2 の情報処理装置)、及び構成情報管理装置 100 が提供する情報を利用する情報処理装置である一台以上の利用装置 102 とを含んで構成されている。

【0011】

構成情報管理装置 100 は、管理対象装置 101 の構成管理に用いられる情報を管理対象装置 101 から収集する。構成情報管理装置 100 は、収集した情報を用いて生成した情報 (以下、構成情報と称する。) を利用装置 102 に提供する。構成情報管理装置 100 は、管理対象装置 101 から収集した情報以外にも、管理対象装置 101 の管理に関する様々な情報を記憶し、それらを利用装置 102 に提供する。

10

【0012】

図 2 に構成情報管理装置 100、管理対象装置 101、及び利用装置 102 として用いられる情報処理装置 (コンピュータ) のハードウェア構成の一例を示している。同図に示す情報処理装置 200 は、CPU 201、主記憶装置 202 (メモリ)、ハードディスク等の補助記憶装置 203、キーボードやマウスなどの入力装置 204、ディスプレイやプリンタなどの出力装置 205、ネットワーク 103 に接続するための通信インタフェースである通信制御装置 206、及びこれらの装置を通信可能に接続するバス 207 を含んで構成されている。

20

【0013】

なお、構成情報管理装置 100、管理対象装置 101、及び利用装置 102 は、夫々図 2 に示す構成からなる情報処理装置 200 を複数台用いて構成されていてもよい。

【0014】

図 3 に構成情報管理装置 100 によって実現される機能、及び構成情報管理装置 100 が記憶しているデータを示している。同図に示す各機能は、構成情報管理装置 100 のハードウェア、もしくは、CPU 201 が主記憶装置 202 や補助記憶装置 203 に記憶されているプログラムを実行することにより実現される。

【0015】

図 3 に示すように、構成情報管理装置 100 は、リソースアドレス発見部 301、リソースアドレス登録部 302、リソース情報収集更新部 303、リソースアドレス通知イベント登録部 304、イベントメッセージ解析部 305、リソースアクセス処理部 306、リソースアドレス更新部 307、イベント受信処理部 308、クライアントアクセス処理部 309 等の機能を有している。また、構成情報管理装置 100 は、管理エンドポイントテーブル 310、リソース一覧テーブル 311、リソースアドレス通知イベント条件テーブル 312、リソース情報テーブル 313、リソース間関連テーブル 314 を記憶している。

30

【0016】

図 4 に構成情報管理装置 100 の管理対象である管理対象装置 101 において実現される機能、及び管理対象装置 101 が記憶しているデータを示している。同図に示す各機能は、構成情報管理装置 100 のハードウェア、もしくは、CPU 201 が主記憶装置 202 や補助記憶装置 203 に記憶されているプログラムを実行することにより実現される。

40

【0017】

同図に示すように、管理対象装置 101 では標準管理エージェント 401 が機能している。標準管理エージェント 401 は、管理対象となるリソース (以下、管理対象リソースと称する。) に関する情報の提供、情報の更新、情報の操作等の機能を、標準的な運用管理技術に従ったインタフェースを介して提供する。後述するように、構成情報管理装置 100 は、各管理対象装置 101 で動作している標準管理エージェント 401 の標準管理インタフェースを利用して管理対象リソースに関する情報を収集する。上記運用管理技術の具体例として DMTF (Distributed Management Task Force) が標準化を行っている WS-

50

ManagementやC I M (Common Information Model)がある。

【0018】

管理対象装置101において、標準管理エージェント401は管理対象装置101で動作しているオペレーティングシステム411の機能を利用することにより管理対象リソースに関する様々な情報を取得する。例えば、管理対象装置101が図2に示すハードウェアで構成される場合、標準管理エージェント401は、管理対象装置101のハードウェア(CPU201、主記憶装置202、補助記憶装置203、入力装置204、出力装置205、通信制御装置206、及びバス207)に関する情報、オペレーティングシステム411の管理下で実行されるプロセスに関する情報、オペレーティングシステム411に関する情報などをオペレーティングシステム411から取得する。

10

【0019】

図4に示すように、標準管理エージェント401は、クライアントアクセス処理部403、オブジェクト管理部404、及び実リソースアクセス部405の各機能を有する。また、標準管理エージェント401は、管理オブジェクト定義情報406、管理オブジェクトインスタンス情報407を有する。同図に示す各機能は、管理対象装置101のハードウェア、もしくは、CPU201が主記憶装置202や補助記憶装置203に記憶されているプログラムを実行することにより実現される。

【0020】

標準管理エージェント401の機能のうちクライアントアクセス処理部403は、構成情報管理装置100など、標準管理エージェント401の機能を利用するシステムとの間でメッセージの送受信を行う。このメッセージの送受信は標準プロトコルに準拠して行われる。

20

【0021】

標準管理エージェント401の機能のうちオブジェクト管理部404は、管理対象リソースを表す管理オブジェクトに対する情報の参照、情報の更新、操作などの要求を、実際の管理対象リソースに対する要求として解釈して実リソースアクセス部405を呼び出す。実リソースアクセス部405は、実際の管理対象リソースとの接続機能を担い、実際の管理対象リソースから収集した情報を管理オブジェクトインスタンスとして応答する。

【0022】

標準管理エージェント401が管理するデータのうち管理オブジェクト定義情報406は、管理対象リソースを表現するためのモデルに関する定義情報である。管理対象リソースは、標準管理モデルの定義に従った管理オブジェクトとして表現される。また管理オブジェクトインスタンス情報407には、管理オブジェクトインスタンスが保持される。

30

【0023】

ところで、標準管理エージェント401は、以上に説明したようなオペレーティングシステム411と連携する構成以外にも、図5に示すようにDBMS412(データベース管理システム)と連携し、DBMS412が管理しているリソースに関する情報を取得/管理する構成であってもよい。また、図6に示すように、アプリケーションサーバ413と連携し、アプリケーションサーバ413が管理しているリソースに関する情報を取得/管理する構成であってもよい。

40

【0024】

管理対象装置101は必ずしも物理的な情報処理装置に限定される訳ではなく、例えば物理的な情報処理装置(物理マシン)で動作する仮想化環境(仮想マシンモタ)によって実現される仮想的な情報処理装置(仮想マシン)であってもよい(図7を参照)。この場合、仮想化環境は、物理マシンのCPU、主記憶装置、補助記憶装置、入力装置、出力装置、通信制御装置などの物理的なハードウェアを、仮想的な装置として仮想マシンに提供する。

【0025】

次に、構成情報管理装置100の各機能の動作について具体的に説明する。まず構成情報管理装置100が管理対象リソースのリソースアドレスを発見する処理について説明す

50

る。

【0026】

ここでリソースアドレスとは、管理対象装置101のリソースの識別や特定を行うための情報である。例えば管理対象リソースが情報処理装置（コンピュータ）である場合には、リソースアドレスはホスト名、IPアドレス、MACアドレス、製造社名、モデル名、シリアル番号、システムUUID（Universally Unique Identifier）などを用いて構成される。また、管理対象リソースがオペレーティングシステムである場合には、リソースアドレスはオペレーティングシステムの名称及びバージョン情報と情報処理装置の識別子などを用いて構成される。また、管理対象リソースがアプリケーションソフトウェアである場合には、アプリケーションソフトウェアの名称、バージョン情報、ソフトウェアがインストールされている情報処理装置又はオペレーティングシステムの識別子などを用いて構成される。

10

【0027】

構成情報管理装置100が管理対象リソースのリソースアドレスを発見する方法としては、構成情報管理装置100側から管理対象リソース側にリソースアドレスの取得要求を行うPull型、及び管理対象リソース側から構成情報管理装置100側にリソースアドレスの送信を行うPush型が考えられる。Pull型の場合のリソースアドレスの収集タイミングとしては、定期的に収集する、又はユーザの要求に応じて収集するなどがある。またPush型の場合の収集タイミングとしては、定期的に収集（又は送信）する、又はリソースの生成時に収集（又は送信）するなどがある。

20

【0028】

図9は構成情報管理装置100のリソースアドレス発見部301が、上記Pull型の方法で管理対象リソースのリソースアドレスを発見する処理を説明するフローチャートである。なお、以下の説明において符号の前に付した文字「S」はステップを意味する。

【0029】

同図に示すように、まずリソースアドレス発見部301が管理エンドポイントテーブル310から管理エンドポイントを取得する。管理エンドポイントは、構成情報管理装置100が管理対象装置101上の標準管理エージェント401に対してリクエストを送信する宛先を示す情報である。図15に管理エンドポイントテーブル310の一例を示す。標準管理エージェント401は管理エンドポイントを介して複数のリソースに関する管理機能を提供する。

30

【0030】

管理エンドポイントはオペレータ等が管理エンドポイントテーブル310に手動で登録してもよいし、構成情報管理装置100が標準管理エージェント401とSLP（Service Location Protocol）等のプロトコルを用いて通信することにより管理エンドポイントをプログラマティックに発見し自動登録するようにしてもよい。

【0031】

次にリソースアドレス発見部301は、リソースアクセス処理部306を通じて標準管理エージェント401と通信し、管理エンドポイントテーブル310から取得した各管理エンドポイントに対応づけて標準管理エージェント401が管理しているリソースのリソースアドレス一覧を取得する（S901）。

40

【0032】

このような仕組みを実現する技術としてWS-Addressingがある。WS-Addressingに基づくリソースアドレスは、管理エンドポイントを表すURL（Uniform Resource Locator）と管理エンドポイント内で一意にリソースを識別する情報との組み合わせで表現される。なお、以下の説明において、リソースアドレスはWS-Addressingに基づくものとする。

【0033】

図16にWS-Addressingに基づくリソースアドレスの一例を示している。同図に示すように、リソースアドレス1600は、管理エンドポイントを示すURL1601、及び管理エンドポイント内でリソースを特定するための情報1602を含んでいる。標準管理工

50

ージェント401がWS-Managementに準拠している場合、リソースアドレス一覧は例えばWS-Managementで規定されるEnumerateオペレーションを利用して取得することができる。

【0034】

次にリソースアドレス発見部301はリソースアドレス登録部302を呼び出し、S901で取得したリソースアドレスをリソース一覧テーブル311に登録する(S902)。

【0035】

図10は図9のS902においてリソースアドレス登録部302によって行われる処理を説明するフローチャートである。

【0036】

まずリソースアドレス登録部302は、取得したリソースアドレスに一致するリソースアドレスをリソース一覧テーブル311から検索する(S1000)。

【0037】

図17にリソース一覧テーブル311を例示している。同図において、リソース識別子1701は、構成情報管理装置100が個々の管理対象リソースに対して一意に付与する識別情報である。符号1702は、リソースアドレスである。また符号1703は、後述するリソースアドレス通知イベント登録の有無を示す情報である。符号1704は、リソースアドレスの状態を示す情報である。この符号1704には、該当のリソースアドレスを用いて管理対象リソースにアクセス可能であれば"有効"が、そうでない場合は"無効"が登録される。

【0038】

符号1704には、リソースアドレス通知イベント登録が"有"の場合にのみ内容が登録される。リソースアドレスが利用不可能な場合、通常は該当するレコードを削除するが、リソースアドレス通知イベント登録が"有"の場合はレコードを削除せず、符号1704に"無効"を登録してレコードは保持しておく。

【0039】

次に、リソースアドレス登録部302は、S1000の処理の結果、リソース一覧テーブル311に一致するリソースアドレスが存在するか否かを確認する(S1001)。一致するリソースアドレスが存在する場合には(S1001:YES)処理を終了する。

【0040】

一方、一致するリソースアドレスが存在しなかった場合には(S1001:NO)、以下の条件に合致するイベント通知登録情報の取得を入力されたリソースアドレスを宛先として要求する(S1002)。

条件1: イベント通知先アドレスに構成情報管理装置100のイベントメッセージ受信エンドポイントが含まれている。

条件2: イベント通知先アドレスにリソース識別子が含まれている。

【0041】

ここでイベント通知登録情報について説明する。標準管理エージェント401に対し、ある特定の条件に合致する場合にある特定の宛先に通知メッセージを送信することを要求することをイベント通知登録(サブスクライブ)と呼ぶ。例えば、「管理対象リソースがオペレーティングシステム411である場合、オペレーティングシステム411の稼働状態が"停止"となった場合に構成情報管理装置100に通知メッセージを送信させる」というイベント通知登録が考えられる。

【0042】

標準管理エージェント401は、構成情報管理装置100から受信したイベント通知登録をイベント通知登録情報(サブスクリプション)として保持して管理対象リソースを監視する。一般的なイベント通知登録情報には、イベント通知先アドレス、イベント通知登録対象リソースアドレス(サブスクライブ対象リソース)、及びイベント発火条件が含まれる。

【0043】

10

20

30

40

50

図 29 にイベント通知登録 (サブスクライブ) リクエストメッセージの一例を示す。同図において、符号 2900 は、イベント通知登録リクエストメッセージを示す。符号 2901 は、イベント通知登録対象リソース (サブスクライブ対象リソース) アドレスを示す情報である。符号 2902 は、イベント通知先アドレスを示す情報である。符号 2903 は、イベント発火条件を示す情報である。

【0044】

図 29 に示す例では、イベント通知登録対象リソース (サブスクライブ対象リソース) アドレスは SOAP メッセージ (SOAP : Simple Object Access Protocol) のヘッダ部分に存在しているが、SOAP メッセージのボディ部分に存在していても良い。図 30 は SOAP メッセージのボディ部分にイベント通知登録対象リソースを特定する情報が存在する場合におけるイベント通知登録リクエストメッセージの一例を示す図である。同図において、符号 3002 がイベント通知登録対象リソースを特定する情報である。

10

【0045】

標準管理エージェント 401 は、外部にイベント通知登録情報を提供する機能を有している。標準管理エージェント 401 が WS-Management と CIM に準拠している場合、イベント通知登録情報は CIM_IndicationFilter インスタンス、CIM_IndicationHandler インスタンス、CIM_IndicationSubscription インスタンスで表現され、これらは WS-Management で規定される Get オペレーションを利用して取得することができる。

【0046】

図 10 の S1003 では、リソースアドレス登録部 302 が上記条件に合致するイベント通知登録情報が発見されたか否かを確認している。イベント通知登録情報が発見されなかった場合には (S1003 : YES)、入力されたリソースアドレスに対してリソース識別子を付与し、リソース一覧テーブル 311 にリソース識別子、及び入力されたリソースアドレスを新規登録する (S1004)。新規登録に際しリソースアドレス通知イベント登録 1703 の値は "無"、状態 1704 の値は "有効" として登録する。

20

【0047】

一方、S1003 の結果、イベント通知登録情報が発見された場合には (S1003 : NO)、発見されたイベント通知登録情報からリソース識別子を取得する (S1005)。そしてリソースアドレス登録部 302 は、S1005 で取得したリソース識別子、及び入力されたリソースアドレスを新たにリソース一覧テーブル 311 に登録する。このときリソースアドレス通知イベント登録 1703 の値は "有"、状態 1704 の値は "有効" として登録する (S1006)。

30

【0048】

次に構成情報管理装置 100 は、以上のようにして発見 / 登録したリソースアドレスについて図 11 に示すリソースアドレス通知イベント登録処理を行う。同図に示すように、まずリソースアドレス通知イベント登録部 304 が、処理対象となるリソース (以下、処理対象リソースと称する。) を決定する (S1100)。

【0049】

ここで処理対象リソースの決定方法としては、リソース一覧テーブル 311 に管理されている全てのリソースを対象とする方法、オペレータ等が手動で対象リソースを指定する方法などがある。

40

【0050】

また、構成情報管理装置 100 が事前に構成情報管理装置 100 に設定しておいたポリシーに従ってリソース一覧テーブル 311 の中から対象とするリソースを選択するようにしてもよい。この場合には、例えば「管理対象リソースが仮想マシン上で動作するオペレーティングシステムである場合に処理対象とする」といった内容を上記ポリシーとして設定する。

【0051】

管理対象リソースが仮想マシン上のオペレーティングシステムであるか否かの判別は、リソース情報テーブル 313、リソース間関連テーブル 314 を参照して行う。例えば C

50

IMに準拠している場合、仮想マシンはCIM_ComputerSystemインスタンスとして表され、かつ、CIM_HostedDependencyインスタンス（関連）により、別のCIM_ComputerSystemインスタンスと関連付けられる。従って仮想マシン上のオペレーティングシステムであるか否かは、このようなCIM_ComputerSystemインスタンスとCIM_InstalledOSインスタンス（関連）によって関連付けられているCIM_OperatingSystemインスタンスであるか否かにより行うことができる。

【 0 0 5 2 】

次に、リソースアドレス通知イベント登録部 3 0 4 は、リソース一覧テーブル 3 1 1 から S 1 1 0 0 で決定した処理対象リソースのリソース識別子、及びリソースアドレスを取得する（S 1 1 0 1）。また、リソースアドレス通知イベント登録部 3 0 4 は、リソース
10
アドレス通知イベント条件テーブル 3 1 2 から、処理対象リソースの種別（リソース種別 1 8 0 1）に応じたイベント条件 1 8 0 2 を取得する（S 1 1 0 2）。

【 0 0 5 3 】

図 1 8 にリソースアドレス通知イベント条件テーブル 3 1 2 の一例を示している。リソースアドレス通知イベント条件テーブル 3 1 2 には、リソース種別 1 8 0 1 とイベント条件 1 8 0 2 とが対応づけて登録されている。リソース種別 1 8 0 1 にはリソースの種別を一意に識別できる情報が登録される。WS-ManagementやCIMを用いる場合には、リソース種別を示す値として、リソース情報を表現するために用いるXMLスキーマの名前空間名を利用し、例えばリソースの種別がオペレーティングシステムである場合、"http://schemas.dmtf.org/wbem/wsCIM/1/CIM-schema/2/CIM_OperatingSystem" という値を用いる
20

【 0 0 5 4 】

イベント条件 1 8 0 2 は、リソースアドレス通知イベントを送信させる条件を示す。例えばWS-ManagementやCIMを用いる場合、オペレーティングシステムの起動を示す条件は、XPathやCQL（CIM Query Language）を用いて、CIM_OperatingSystemインスタンスのEnabledStateプロパティの値が"Enabled"あるいは"Starting"であることを表現することにより指定する方法が考えられる。

【 0 0 5 5 】

次に、リソースアドレス通知イベント登録部 3 0 4 は、イベント通知登録リクエストメッセージを生成する（S 1 1 0 3）。この処理において、リソースアドレス通知イベント登録部 3 0 4 は、S 1 1 0 1 で取得したリソース識別子、及び構成情報管理装置 1 0 0 のイベントメッセージ受信エンドポイントを含むイベント通知先アドレスを生成する。
30

【 0 0 5 6 】

図 1 9 にWS-Managementを用いる場合のイベント通知先アドレスの一例を示す。同図において、符号 1 9 0 1 は、構成情報管理装置 1 0 0 のイベントメッセージ受信エンドポイントである。符号 1 9 0 2 は、S 1 1 0 1 で取得したリソース識別子である。

【 0 0 5 7 】

次に、リソースアドレス通知イベント登録部 3 0 4 は、S 1 1 0 1 で取得したリソースアドレスに対してS 1 1 0 3 で生成したイベント通知登録リクエストメッセージを送信する（S 1 1 0 4）。例えばWS-Managementを用いる場合は、リソースアドレス通知イベント登録部 3 0 4 はSubscribeリクエストを送信する。この際の標準管理エージェント 4 0 1 に対するリクエスト送信処理、及び標準管理エージェント 4 0 1 からのレスポンス受信処理はリソースアクセス処理部 3 0 6 が実施する。
40

【 0 0 5 8 】

次に、管理対象装置 1 0 1 からのリソース情報の収集、及びリソース情報の更新について説明する。ここでリソース情報の収集方法としては、構成情報管理装置 1 0 0 側から管理対象リソース側に対して情報収集要求を行うPull型と、管理対象リソース側から構成情報管理装置 1 0 0 側に情報送信を行うPush型の大きく 2 通りが考えられる。Pull型の場合のリソース情報の収集タイミングとしては、定期的に収集する、又はユーザの要求に応じて収集するなどが考えられる。また、Push型の場合の収集タイミングとしては、定期的に
50

収集（又は送信）する、値が変化した時に収集（又は送信）するなどが考えられる。

【 0 0 5 9 】

図 1 2 は、Pull型の場合におけるリソース情報収集更新部 3 0 3 の処理を説明するフローチャートである。同図に示すように、まずリソース情報収集更新部 3 0 3 は、リソース一覧テーブル 3 1 1 から、リソース識別子、リソースアドレス、及びリソースアドレス通知イベント登録の値を取得する（S 1 2 0 0）。

【 0 0 6 0 】

リソース情報収集更新部 3 0 3 は、S 1 2 0 0 で取得した各リソースアドレスに対してリソース情報取得リクエストを送信し、リソース情報を取得する（S 1 2 0 1）。例えば WS-Management を用いる場合には、Get リクエストを送信する。標準管理エージェント 4 0 1 に対するリクエスト送信処理、標準管理エージェント 4 0 1 からのレスポンス受信処理は、リソースアクセス処理部 3 0 6 が実施する。

10

【 0 0 6 1 】

次に、リソース情報収集更新部 3 0 3 は、リソース情報を取得する際に使用したリソースアドレスに対応するリソースが不明か否かを確認する（S 1 2 0 2）。リソースが不明であった場合、その旨を示すエラーメッセージが標準管理エージェント 4 0 1 から応答される。

【 0 0 6 2 】

リソース情報収集更新部 3 0 3 は、S 1 2 0 2 の結果、リソースが不明でなかった場合には（S 1 2 0 2 : N O）、S 1 2 0 0 で取得したリソース識別子を用いてリソース情報テーブル 3 1 3、又はリソース間関連テーブル 3 1 4 から該当するレコードを検索し、S 1 2 0 1 で取得したリソース情報を更新する（S 1 2 0 3）。

20

【 0 0 6 3 】

図 2 0 にリソース情報テーブル 3 1 3 の一例を示す。リソース情報テーブル 3 1 3 には、リソース識別子 2 0 0 1、リソース種別 2 0 0 2、及び詳細情報 2 0 0 3 が登録されている。

【 0 0 6 4 】

また、図 2 1 にリソース間関連テーブル 3 1 4 の一例を示す。リソース間関連テーブル 3 1 4 には、リソース識別子 2 1 0 1、関連種別 2 1 0 2、関連元リソース識別子 2 1 0 3、及び関連先リソース識別子 2 1 0 4 が登録されている。

30

【 0 0 6 5 】

関連種別 2 1 0 2 としては、例えば C I M で定義されている Association クラスで表される関連や、ベンダ独自の関連種別なども扱うことが考えられる。独自の関連種別としては、例えば管理対象サーバアプリケーションと管理対象クライアントアプリケーションとの関連を表す関連種別、管理対象サービスとサービスを構成する管理対象アプリケーションとの関連を表す関連種別、管理対象のアプリケーションとアプリケーションに関して発生したインシデント情報との関連を表す関連種別、管理対象リソースとそのリソースに関する変更履歴情報との関連を表す関連種別など、様々なオブジェクト間の関連を扱うことが考えられる。

【 0 0 6 6 】

次に、リソース情報収集更新部 3 0 3 は、S 1 2 0 0 で取得したリソースアドレス通知イベント登録の値を確認する（S 1 2 0 4）。S 1 2 0 4 の結果、値が“有”であれば（S 1 2 0 4 : Y E S）、リソース情報収集更新部 3 0 3 は S 1 2 0 0 で取得したリソース識別子を用いてリソース一覧テーブル 3 1 1 中の該当するレコードを検索し、状態 1 7 0 4 に“有効”を登録する（S 1 2 0 5）。一方、S 1 2 0 4 の結果、値が“無”であれば（S 1 2 0 4 : N O）、リソース情報収集更新部 3 0 3 は、処理を終了する。

40

【 0 0 6 7 】

一方、S 1 2 0 2 の結果、リソースが不明であった場合には（S 1 2 0 2 : Y E S）、リソース情報収集更新部 3 0 3 は、S 1 2 0 0 で取得したリソースアドレス通知イベント登録の値を確認する（S 1 2 0 6）。リソース情報収集更新部 3 0 3 は、S 1 2 0 6 の結

50

果、値が”有”であれば(S1206: YES)、S1200で取得したリソース識別子を用いてリソース一覧テーブル311中の該当するレコードを検索し、状態1704に”無効”を登録する(S1207)。一方、S1206の結果、値が”無”であれば(S1206: NO)、S1200で取得したリソース識別子を用いて、リソース一覧テーブル311中の該当するレコードを検索し、レコードを削除する(S1208)。すなわち、リソースアドレス通知イベント登録の値が”無”である場合にのみ、リソース一覧テーブル311中の該当レコードが削除される。従って、リソースアドレス通知イベント登録の値が”有”である限りレコードはリソース一覧テーブル311に残存し、次の検索(S1200)の対象となる。

【0068】

以上によりリソースアドレスの発見/登録、リソースアドレス通知イベント登録、リソース情報の収集/更新処理が完了する。

【0069】

次に、リソースアドレスの発見/登録、及びリソースアドレス通知イベント登録処理が完了した以降のある時点において、図7に示した構成からなる仮想マシンについてマイグレーションが行われた場合、具体的には図8に示すように物理マシン700上で動作する仮想化環境702上の仮想マシン701である管理対象装置101がコールドマイグレーションにより物理マシン700上で動作する仮想化環境702上に移動する場合を想定し、リソースアドレス通知イベントメッセージが、構成情報管理装置100に送信された場合に情報処理システム1において行われる処理について説明する。

【0070】

仮想マシン701のマイグレーションに伴う仮想マシン701の停止、再起動に際しては、オペレーティングシステムも停止、再起動するため、仮想マシン701上で動作するオペレーティングシステム411に対してオペレーティングシステム411の再起動を条件としてリソースアドレス通知イベント登録を行っている場合には、リソースアドレス通知イベントメッセージが送信される。

【0071】

また、構成情報管理装置100のイベントメッセージ受信エンドポイント宛に送信されるイベントメッセージは、イベント受信処理部308が受信する。イベントメッセージ解析部305は、イベント受信処理部308からイベントメッセージを受け取ると処理を開始する。

【0072】

図22はリソースアドレス通知イベントメッセージの一例を示す図である。符号2201は、イベント通知先エンドポイントであり、図11に示したリソースアドレス通知イベント登録処理において、イベント通知先アドレス内で指定した値である。また、符号2202は、前述のリソースアドレス通知イベント登録処理において、イベント通知先アドレス内で指定したリソース識別子である。これらの情報は、管理対象装置101のマイグレーションの前後を通して、管理対象装置101上の標準管理エージェント401によって、イベント通知登録情報(サブスクリプション)として保持されている。

【0073】

符号2203は、イベント送信元リソースのリソースアドレスである。同図の場合、イベント送信元リソースアドレスは、SOAPメッセージのヘッダ部分に存在しているが、SOAPメッセージのボディ部分に存在していても良い。図31はSOAPメッセージのボディ部分にイベント送信元リソースアドレスの情報が存在する場合におけるリソースアドレス通知イベントメッセージの一例である。同図において、符号3103がイベント送信元リソースアドレスの情報を示す。

【0074】

次に、構成情報管理装置100が管理対象装置101から送られてくるイベントメッセージを受信した場合にイベントメッセージ解析部305によって行われる処理について図13に示すフローチャートとともに説明する。

10

20

30

40

50

【 0 0 7 5 】

まずイベントメッセージ解析部 3 0 5 は、イベントメッセージ中からリソース識別子を取得する (S 1 3 0 0)。イベントメッセージ解析部 3 0 5 は、イベントメッセージ中にリソース識別子が存在するか否かを確認する (S 1 3 0 1)。リソース識別子が存在する場合には (S 1 3 0 1 : Y E S)、S 1 3 0 2 のリソースアドレス通知イベント処理に進む。リソース識別子が存在しない場合には (S 1 3 0 1 : N O)、処理を終了する。

【 0 0 7 6 】

図 1 4 は、リソースアドレス更新部 3 0 7 によって行われるリソースアドレス通知イベント処理 (S 1 3 0 2) を説明するフローチャートである。

【 0 0 7 7 】

リソースアドレス更新部 3 0 7 は、S 1 3 0 0 で取得したリソース識別子に一致するレコードをリソース一覧テーブル 3 1 1 から検索し、検索したレコードのリソースアドレス (以下、イベント通知登録対象リソースアドレスと称する。) を取得する (S 1 4 0 0)

次に、リソースアドレス更新部 3 0 7 は、イベントメッセージ中からイベント送信元リソースのリソースアドレスの取得処理を開始して (S 1 4 0 1)、リソースアドレス更新部 3 0 7 は、イベントメッセージ中にイベント通知の送信元のリソースのリソースアドレス (以下、イベント送信元リソースアドレスと称する。) が存在するか否かを確認する (S 1 4 0 2)。イベントメッセージ中にイベント送信元リソースアドレスが存在する場合には (S 1 4 0 2 : Y E S)、S 1 4 0 3 へ進む。イベントメッセージ中にイベント送信元リソースアドレスが存在しない場合には (S 1 4 0 2 : N O)、処理を終了する。

【 0 0 7 8 】

リソースアドレス更新部 3 0 7 は、S 1 4 0 0 で取得したイベント通知登録対象リソースアドレスと、S 1 4 0 1 で取得したイベント送信元リソースアドレス (現在のリソースアドレス) とを比較する (S 1 4 0 3)。比較の結果、イベント通知登録対象リソースアドレスとイベント送信元リソースアドレスの値が同じであれば (S 1 4 0 3 : Y E S)、すなわち、イベント通知登録対象リソースアドレスを変更する必要がなければ処理を終了する。一方、イベント通知登録対象リソースアドレスとイベント送信元リソースアドレスの値が異なれば (S 1 4 0 3 : N O)、S 1 4 0 4 へ進む。

【 0 0 7 9 】

S 1 4 0 4 では、リソースアドレス更新部 3 0 7 が、S 1 4 0 0 で検索したリソース一覧テーブル 3 1 1 の該当レコードのリソースアドレスを S 1 4 0 1 で取得したイベント送信元リソースアドレスの値に変更し、また状態 1 7 0 4 に " 有効 " を登録する。

【 0 0 8 0 】

リソース一覧テーブル 3 1 1 の各レコードのうち状態 1 7 0 4 に " 無効 " が登録される場合として、コールドマイグレーションにより仮想マシンが停止した際、仮想マシン上の標準管理エージェント 4 0 1 も停止し、標準管理エージェント 4 0 1 が公開するリソースアドレスが利用不可能となっていた場合などがある。

【 0 0 8 1 】

以上により、リソースアドレス通知イベント受信に伴って行われる管理対象リソースのリソースアドレス更新処理が完了する。

【 0 0 8 2 】

ところで、イベント通知登録対象リソースアドレスとイベント送信元リソースアドレスとが異なる場合にのみ、管理対象装置 1 0 1 から構成情報管理装置 1 0 0 にイベントメッセージが送信されるように、リソースアドレス通知イベント登録部 3 0 4 がイベント通知登録を行うようにしてもよい。この場合は S 1 4 0 2 の判断の結果、イベント送信元リソースアドレスが存在すれば (S 1 4 0 2 : Y E S)、S 1 4 0 3 の判断をすることなく S 1 4 0 4 に進むことになる。

【 0 0 8 3 】

図 3 2 にイベント通知登録リクエストメッセージの一例を示す。同図において、符号 3

10

20

30

40

50

203で示すイベント発火条件には「イベント通知登録対象リソースアドレスとイベント送信元リソースアドレスとが異なる」という条件が含まれている。

【0084】

[実施例2]

実施例1ではPull型のリソースアドレス発見方法を用いた例を示したが、本実施例ではPush型のリソースアドレス発見方法について説明する。

【0085】

図23はリソースアドレス発見部301の処理の一例を示すフローチャートである。同図において、まずリソースアドレス発見部301は、管理エンドポイントテーブル310から管理エンドポイント情報を取得する(S2300)。

10

【0086】

次に、リソースアドレス発見部301は、管理エンドポイントテーブル310から取得した各管理エンドポイントに対してインスタンス生成イベントの通知登録を行う(S2301)。例えば標準管理エージェント401がWS-Managementに準拠する場合には、Subscribeリクエストを送信する。実施例1と同様に、標準管理エージェント401に対するリクエスト送信処理、標準管理エージェント401からのレスポンスの受信処理は、リソースアクセス処理部306が実施する。

【0087】

以上により標準管理エージェント401において新たに管理対象リソースのインスタンスが生成された場合には、構成情報管理装置100に対してインスタンス生成イベントメッセージが送信されるようになる。

20

【0088】

次に、構成情報管理装置100がイベントメッセージを受信した場合に行われる処理について説明する。構成情報管理装置100のイベントメッセージ受信エンドポイント宛に送信されるイベントメッセージをイベント受信処理部308が受信すると、イベントメッセージ解析部305はイベント受信処理部308からイベントメッセージを受け取る。

【0089】

図24は、Push型の場合のリソースアドレス発見処理を示すフローチャートである。このフローチャートは図13で説明したイベントメッセージ解析処理にインスタンス生成イベント受信処理のために必要な処理を追加した構成になっている。同図におけるS2400~S2402の処理は、図13のS1300~S1302と同様である。

30

【0090】

S2403において、イベントメッセージ解析部305は、受信したイベントメッセージのイベント種別がインスタンス生成イベントであるか否かを判定する。S2403で判別したイベント種別がインスタンス生成イベントであれば(S2404: YES)、S2405へ進み、そうでなければ(S2404: NO)、処理を終了する。

【0091】

図25は、図24のS2405で示すリソースアドレス発見処理(Push型)(2)を説明するフローチャートである。同図に示すように、リソースアドレス発見部301は、インスタンス生成イベントメッセージから、生成されたインスタンスのリソースアドレスを取得する(S2500)。そしてリソースアドレス発見部301は、図10に示したリソースアドレス登録処理を呼び出す(S2501)。

40

【0092】

[実施例3]

実施例1ではPull型のリソース情報収集方法を用いた例を示したが、実施例3ではPush型リソース情報収集方法について説明する。

【0093】

図26はリソース情報収集更新部303の処理の一例を示すフローチャートである。同図において、まずリソース情報収集更新部303は、リソース一覧テーブル311からリソースアドレスを取得する(S2600)。

50

【 0 0 9 4 】

次に、リソース情報収集更新部 3 0 3 は、イベント通知登録リクエストメッセージを生成する (S 2 6 0 1)。この処理において、リソース情報収集更新部 3 0 3 は、 S 2 6 0 0 で取得したリソース識別子、及び構成情報管理装置 1 0 0 のイベントメッセージ受信エンドポイントを含むイベント通知先アドレスを生成する。

【 0 0 9 5 】

リソース情報収集更新部 3 0 3 は、 S 2 6 0 0 で取得した各リソースアドレスに対してインスタンス変更イベント通知登録を行う (S 2 6 0 2)。例えば標準管理エージェント 4 0 1 が WS-Management に準拠する場合、Subscribe リクエストを送信する。実施例 1 と同様に、標準管理エージェント 4 0 1 に対するリクエスト送信処理、標準管理エージェント 4 0 1 からのレスポンス受信処理は、リソースアクセス処理部 3 0 6 が実施する。

10

【 0 0 9 6 】

以上により、標準管理エージェント 4 0 1 において、管理対象リソースに関する変更があった場合、構成情報管理装置に対してインスタンス変更イベントメッセージが送信される。

【 0 0 9 7 】

次に、構成情報管理装置 1 0 0 がイベントメッセージを受信した場合に行われる処理について説明する。構成情報管理装置 1 0 0 のイベントメッセージ受信エンドポイント宛に送信されるイベントメッセージをイベント受信処理部 3 0 8 が受信すると、イベントメッセージ解析部 3 0 5 は、イベント受信処理部 3 0 8 からイベントメッセージを受け取る。

20

【 0 0 9 8 】

図 2 7 は、Push 型の場合のイベントメッセージ解析処理を示すフローチャートである。このフローチャートは図 1 3 で説明したイベントメッセージ解析処理にインスタンス変更イベントメッセージ受信処理のために必要な処理を追加した構成となっている。同図における S 2 7 0 0 ~ S 2 7 0 2 の処理は前述の S 1 3 0 0 ~ S 1 3 0 2 と同様である。

【 0 0 9 9 】

イベントメッセージ解析部 3 0 5 は、 S 2 7 0 2 の後、受信したイベントメッセージのイベント種別がインスタンス変更イベントであるか否かを判定する (S 2 7 0 3)。 S 2 7 0 3 で判別したイベント種別がインスタンス変更イベントであれば (S 2 7 0 4 : Y E S)、 S 2 7 0 5 に進み、そうでなければ (S 2 7 0 4 : N O)、処理を終了する。

30

【 0 1 0 0 】

図 2 8 は、図 2 7 の S 2 7 0 5 で示すリソース情報収集更新処理 (Push 型) (2) を示すフローチャートである。まずリソース情報収集更新部 3 0 3 は、取得したインスタンス変更イベントメッセージからインスタンス情報を取得する (S 2 8 0 0)。

【 0 1 0 1 】

次に、リソース情報収集更新部 3 0 3 は、 S 2 7 0 0 で取得したリソース識別子を用いてリソース情報テーブル 3 1 3、又はリソース間関連テーブル 3 1 4 の該当するレコードを検索し、 S 2 8 0 0 で取得したインスタンス情報を更新する (S 2 8 0 1)。

【 0 1 0 2 】

リソース情報収集更新部 3 0 3 は、 S 2 7 0 0 で取得したリソース識別子を用い、リソース一覧テーブル 3 1 1 から、リソースアドレス通知イベント登録の値を取得する (S 2 8 0 2)。

40

【 0 1 0 3 】

リソース情報収集更新部 3 0 3 は、 S 2 8 0 2 で取得したリソースアドレス通知イベント登録の値を確認する (S 2 8 0 3)。ここで値が " 有 " であれば (S 2 8 0 3 : Y E S)、 S 2 7 0 0 で取得したリソース識別子を用いてリソース一覧テーブル 3 1 1 の該当するレコードを検索し、状態 1 7 0 4 に " 有効 " を登録し (S 2 8 0 4)、値が " 無 " であれば (S 2 8 0 3 : N O) 処理を終了する。

【 0 1 0 4 】

以上に説明した実施例 1 ~ 実施例 3 に示した構成によれば、管理対象であるリソースを

50

識別又は特定する情報であるリソースアドレスが変化した場合でもリソースを効率よく適切に管理することができる。

【0105】

また、構成情報管理装置100は、仮想マシンである管理対象装置101上で動作するオペレーティングシステム411を、仮想マシンのマイグレーションの前後を通して継続的に識別することができる。換言すれば、仮想マシンのマイグレーションにより、仮想マシン上で動作していたオペレーティングシステム411が提供していたリソースアドレスが変化したとしてもマイグレーション以前に管理していた管理対象リソースと同一のリソースであることを判別することができる。従って、構成情報管理装置100は、構成情報管理装置100の利用者に対し、継続的かつ一貫した情報を提供することができる。

10

【0106】

より具体的には、構成情報管理装置100は、管理対象装置101の構成情報を収集・管理する場合に、管理対象装置101の情報に変化した場合でも、管理対象装置101上のリソースを別リソースと誤認識することなく正しく識別できる。このため、異なるネットワーク環境への移設、マイグレーション、共有ディスクからオペレーティングシステムを起動する際のブレードサーバの変更などが行われた場合でも、構成情報管理装置100或いは利用装置102において、管理対象リソースに関してそれまで蓄えてきた既存の管理情報を修正することなく利用し続けることができる。

【0107】

また、以上に説明したように、本実施形態の構成情報管理装置100は、管理対象リソースを一意に識別するリソース識別子を管理対象リソースに付与し、管理対象リソースの環境が変化した後、標準管理エージェント401のイベント通知に関する機能を利用して管理対象リソースからリソース識別子とこれに対応づけられている現在のリソースアドレスとを取得し、構成情報管理装置100がリソース識別子に対応づけて記憶しているリソースアドレスを現在のリソースアドレスに変更する。

20

【0108】

より具体的には、構成情報管理装置100は、標準管理エージェント401の機能を利用してイベント通知を受信するための情報とリソース識別子とを管理対象リソース側に記憶させ、標準管理エージェント401からのイベント通知の受信に際しリソース識別子と管理対象リソースに現在付与されているリソースアドレスとを取得し、リソース識別子に対応づけて記憶しているリソースアドレスを現在のリソースアドレスに変更する。

30

【0109】

このように、管理対象装置101へのリソース識別子の記憶や、管理対象リソースからのリソース識別子と現在付与されているリソースアドレスの取得は標準管理エージェント401の機能を用いて行われる。このため、本実施形態の情報処理システム1を実現するにあたり、特別なエージェントを管理対象装置101に用意する必要がなく、情報処理システム1が複雑化したり運用負担を増大させたりすることもなく、管理対象リソースの構成情報を効率よく管理することができる。

【0110】

ところで、標準管理エージェント401が、管理対象リソースに関する状態情報（プロパティ）に対する外部からの情報の設定（書込み）を許可している場合には、状態情報を用いて構成情報管理装置100がリソース識別子を保持するようにすることも考えられる。

40

【0111】

しかしながら、セキュリティ上、標準管理エージェント401は外部からの情報書込みを許可されていないことが多く、このような方法を採用できないことも多い。また、複数の構成情報管理装置100が同一の管理対象リソースにアクセスする情報処理システム1では、各構成情報管理装置100に同一のプロパティを用いてリソース識別子を保持させることはできない。

【0112】

50

これに対し本実施形態の構成では、構成情報管理装置 100 ごとに異なるイベント通知登録情報（サブスクリプション）を保持させることが可能であるのでセキュリティを確保しつつ各構成情報管理装置 100 が独自にリソースの識別子を管理することができる。

【0113】

なお、以上の実施形態の説明は、本発明の理解を容易にするためのものであり、本発明を限定するものではない。本発明はその趣旨を逸脱することなく、変更、改良され得ると共に本発明にはその等価物が含まれることは勿論である。

【図面の簡単な説明】

【0114】

【図1】本発明の実施の形態として説明する情報処理システム1の構成を示す図である。

10

【図2】本発明の実施の形態として説明する構成情報管理装置100、管理対象装置101、及び利用装置102として用いられる情報処理装置（コンピュータ）のハードウェア構成の一例を示す図である。

【図3】本発明の実施の形態として説明する構成情報管理装置100によって実現される機能、及び構成情報管理装置100が記憶しているデータである。

【図4】本発明の実施の形態として説明する管理対象装置101において実現される機能、及び管理対象装置101が記憶しているデータである。

【図5】本発明の実施の形態として説明する本発明の実施の形態として説明する管理対象装置101において実現される機能、及び管理対象装置101が記憶しているデータである。

20

【図6】本発明の実施の形態として説明する本発明の実施の形態として説明する管理対象装置101において実現される機能、及び管理対象装置101が記憶しているデータである。

【図7】本発明の実施の形態として説明する管理対象装置101が仮想マシンである場合における管理対象装置101の構成を示す図である。

【図8】本発明の実施の形態として説明する管理対象装置101（仮想マシン）のマイグレーションを示す図である。

【図9】本発明の実施の形態として説明するリソースアドレス発見部301の処理を示すフローチャートである。

【図10】本発明の実施の形態として説明するリソースアドレス登録部302の処理を示すフローチャートである。

30

【図11】本発明の実施の形態として説明するリソースアドレス通知イベント登録部の処理を示すフローチャートである。

【図12】本発明の実施の形態として説明するリソース情報収集更新部303の処理（Pull型）を示すフローチャートである。

【図13】本発明の実施の形態として説明するイベントメッセージ解析部305の処理を示すフローチャートである。

【図14】本発明の実施の形態として説明するリソースアドレス更新部307の処理を示すフローチャートである。

【図15】本発明の実施の形態として説明する管理エンドポイントテーブル310の一例である。

40

【図16】本発明の実施の形態として説明するWS-Addressingに基づくリソースアドレスの一例である。

【図17】本発明の実施の形態として説明するリソース一覧テーブル311の一例である。

【図18】本発明の実施の形態として説明するリソースアドレス通知イベント条件テーブル312の一例である。

【図19】本発明の実施の形態として説明するWS-Managementを用いる場合のイベント通知先アドレスの一例である。

【図20】本発明の実施の形態として説明するリソース情報テーブル313の一例である

50

。

【図 2 1】本発明の実施の形態として説明するリソース間関連テーブル 3 1 4 の一例である。

【図 2 2】本発明の実施の形態として説明するリソースアドレス通知イベントメッセージの一例である。

【図 2 3】本発明の実施の形態として説明するリソースアドレス発見部 3 0 1 の処理 (Push 型) の一例を示すフローチャートである。

【図 2 4】本発明の実施の形態として説明するイベントメッセージ解析処理を示すフローチャートである。

【図 2 5】本発明の実施の形態として説明するリソースアドレス発見処理 (Push 型) を説明するフローチャートである。

【図 2 6】本発明の実施の形態として説明するリソース情報収集更新部の処理 (Push 型) を示すフローチャートである。

【図 2 7】本発明の実施の形態として説明するイベントメッセージ解析部の処理を示すフローチャートである。

【図 2 8】本発明の実施の形態として説明するリソース情報収集更新部の処理 (Push 型) を示すフローチャートである。

【図 2 9】本発明の実施の形態として説明するイベント通知登録 (サブスクライブ) リクエストメッセージの一例を示す図である。

【図 3 0】本発明の実施の形態として説明するイベント通知登録リクエストメッセージの一例を示す図である。

【図 3 1】本発明の実施の形態として説明するリソースアドレス通知イベントメッセージの一例を示す図である。

【図 3 2】本発明の実施の形態として説明するイベント通知登録リクエストメッセージの一例を示す図である。

【符号の説明】

【0 1 1 5】

1 情報処理システム

1 0 0 構成情報管理装置

1 0 1 管理対象装置

1 0 2 利用装置

3 0 1 リソースアドレス発見部

3 0 2 リソースアドレス登録部

3 0 3 リソース情報収集更新部

3 0 4 リソースアドレス通知イベント登録部

3 0 5 イベントメッセージ解析部

3 0 6 リソースアクセス処理部

3 0 7 リソースアドレス更新部

3 0 8 イベント受信処理部

3 0 9 クライアントアクセス処理部

3 1 0 管理エンドポイントテーブル

3 1 1 リソース一覧テーブル

3 1 2 リソースアドレス通知イベント条件テーブル

3 1 3 リソース情報テーブル

3 1 4 リソース間関連テーブル

4 0 1 標準管理エージェント

4 1 1 オペレーティングシステム

4 0 3 クライアントアクセス処理部

4 0 4 オブジェクト管理部

4 0 5 実リソースアクセス部

10

20

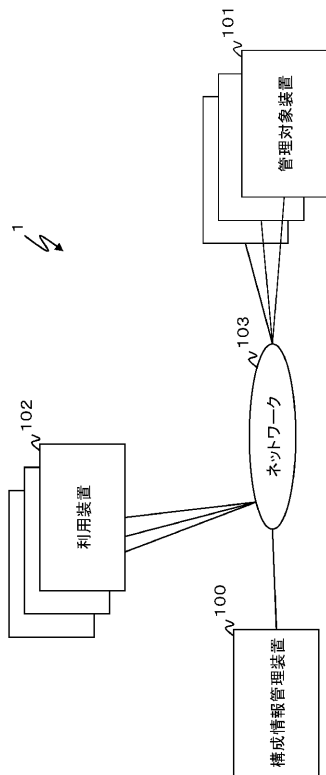
30

40

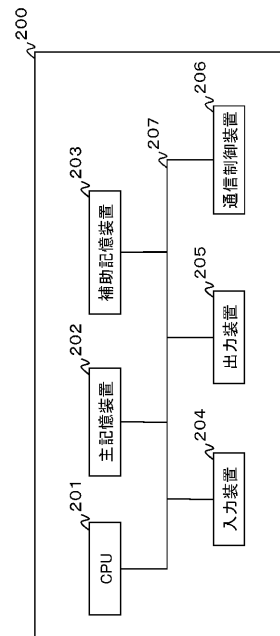
50

- 406 管理オブジェクト定義情報
- 407 管理オブジェクトインスタンス情報
- 700 物理マシン
- 701 仮想マシン
- 702 仮想化環境

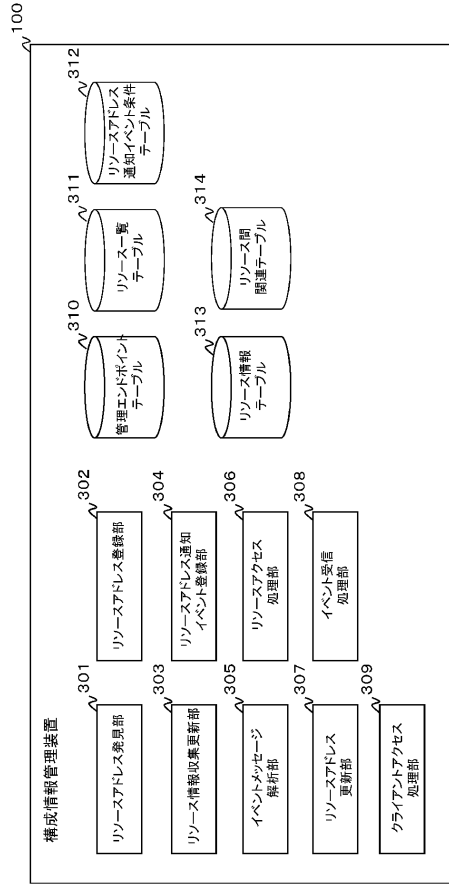
【 図 1 】



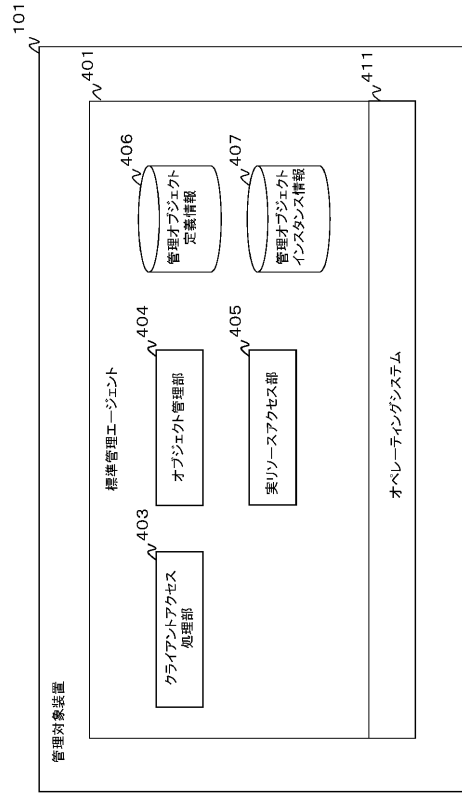
【 図 2 】



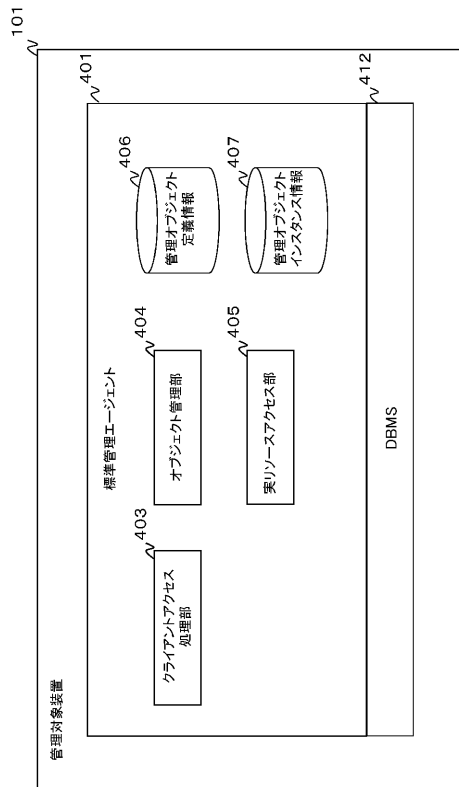
【 図 3 】



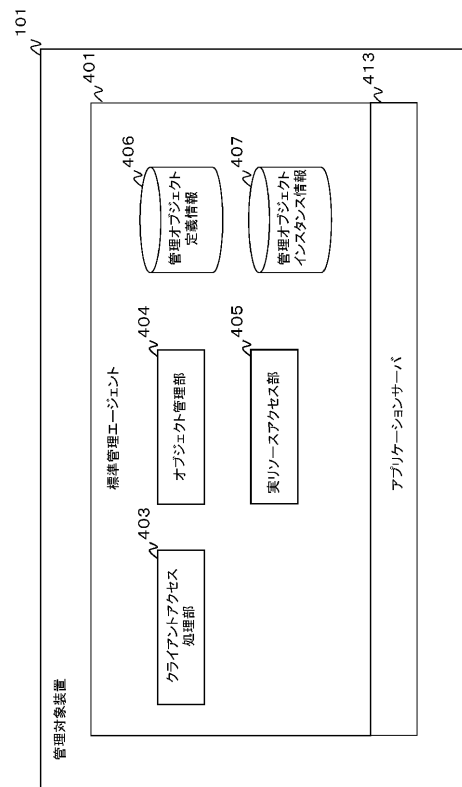
【 図 4 】



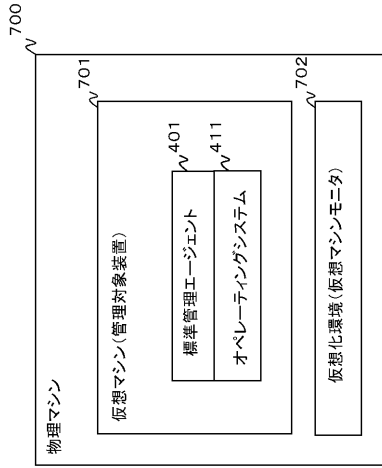
【 図 5 】



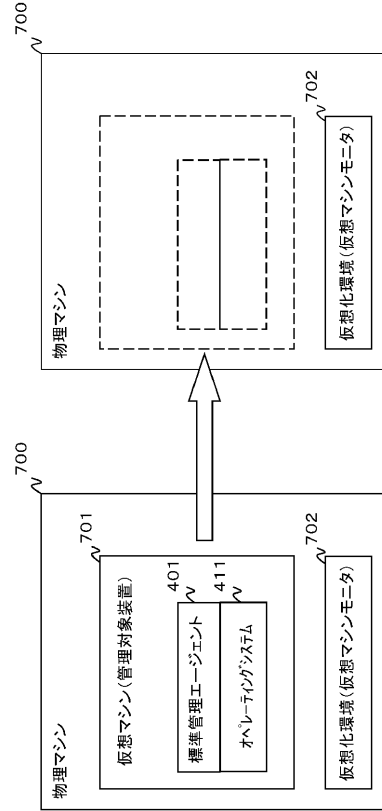
【 図 6 】



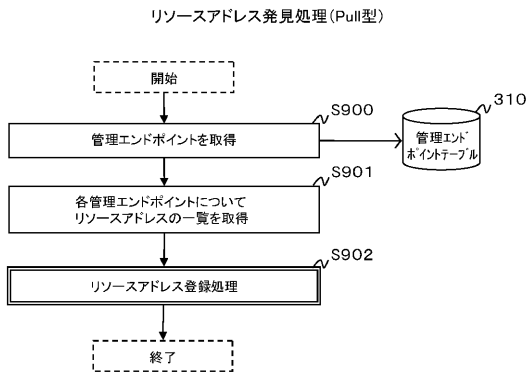
【 図 7 】



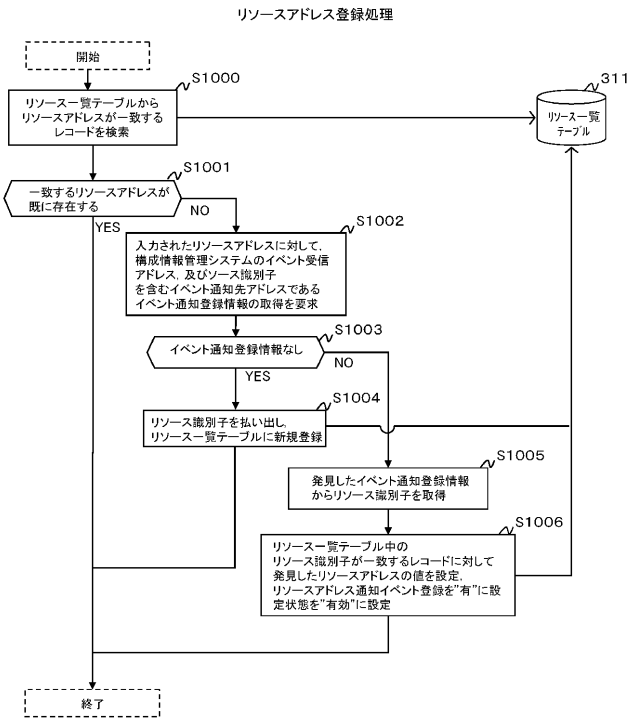
【 図 8 】



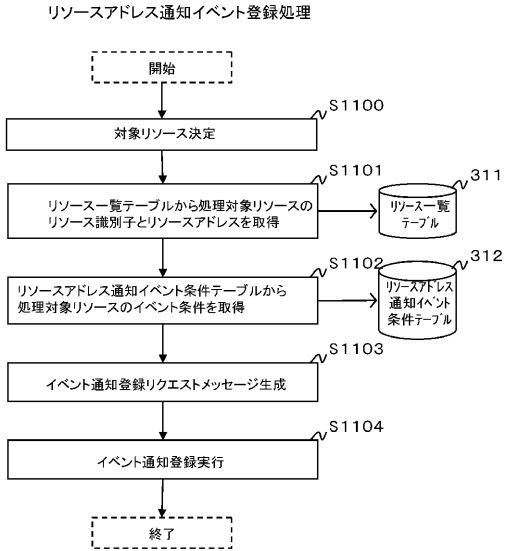
【 図 9 】



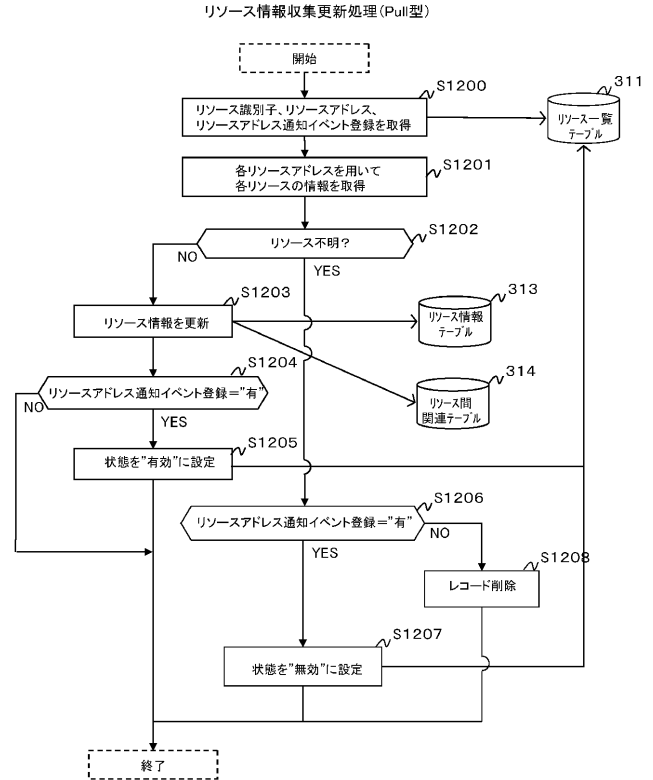
【 図 10 】



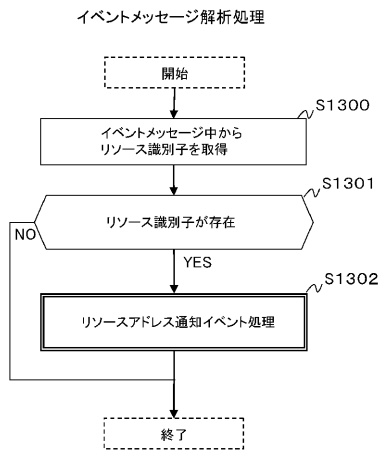
【 図 1 1 】



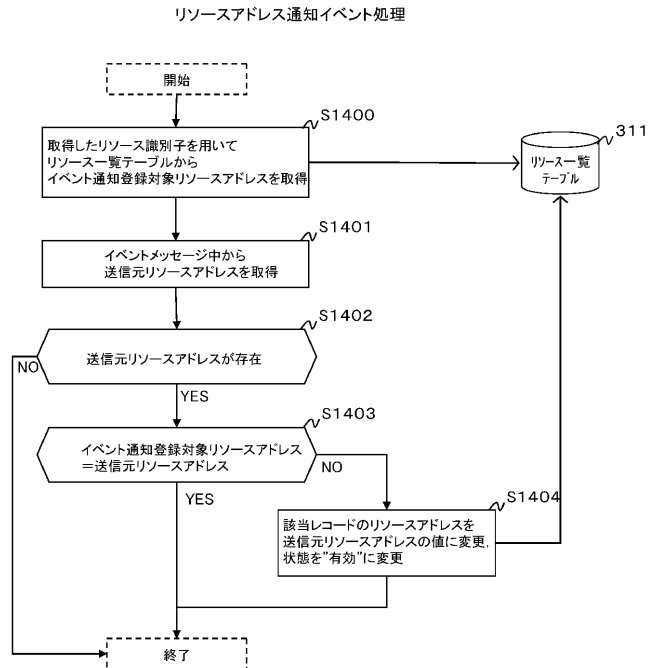
【 図 1 2 】



【 図 1 3 】



【 図 1 4 】



【 図 1 5 】

管理エンドポイントテーブル

管理エンドポイント
http://192.168.0.1:8080/wsman
http://192.168.0.2:8080/mep
.
.

【 図 1 6 】

WS-Addressingに基づくリソースアドレスの一例

```

<wsa:EndpointReference>
  <wsa:Address>http://192.168.0.101:8080/wsman</wsa:Address>
  <wsa:ReferenceParameters>
    <wsman:ResourceURI>
      http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_OperatingSystem
    </wsman:ResourceURI>
    <wsman:SelectorSet>
      <wsman:Selector Name="CSCreationClassName">CIM_ComputerSystem</wsman:Selector>
      <wsman:Selector Name="CSName">ABC</wsman:Selector>
      <wsman:Selector Name="CreationClassName">CIM_OperatingSystem</wsman:Selector>
      <wsman:Selector Name="Name">XXX_OS</wsman:Selector>
    </wsman:SelectorSet>
  </wsa:ReferenceParameters>
</wsa:EndpointReference>

```

【 図 1 9 】

イベント通知先アドレス

```

<wsa:EndpointReference>
  <wsa:Address>http://192.168.0.100:8090/EventConsumer</wsa:Address>
  <wsa:ReferenceParameters>
    <wsman:ResourceURI>http://schemas.foo/cmdb/1/EventConsumer.xsd</wsman:ResourceURI>
    <wsman:SelectorSet>
      <wsman:Selector Name="ResourceID">OS_1</wsman:Selector>
    </wsman:SelectorSet>
  </wsa:ReferenceParameters>
</wsa:EndpointReference>

```

【 図 2 0 】

リソース情報テーブル

リソース識別子	リソース種別	詳細情報
OS_1	OperatingSystem	Name="XYZ", Version="1.0", EnabledState="Enabled"
CS_1	ComputerSystem	Name="ABC"
CS_1	ComputerSystem	Name="DEF"
.		
.		

【 図 1 7 】

リソース一覧テーブル

リソース識別子	リソースアドレス	リソースアドレス通知イベント登録	状態
OS_1		有	有効
OS_1		無	-
OS_2		有	無効
CS_2		無	-
.	.	.	.
.	.	.	.

【 図 1 8 】

リソースアドレス通知イベント条件テーブル

リソース種別	イベント条件
OperatingSystem	EnabledState = "Enabled"
.	.
.	.

【 図 2 1 】

リソース間関係テーブル

関連識別子	関連種別	関連元リソース識別子	関連先リソース識別子
R_1	InstalledOS	OS_1	CS_1
R_2	RunningOS	OS_1	CS_1
R_3	HostedDependency	OS_1	CS_2
.			.
.			.

【 図 2 2 】

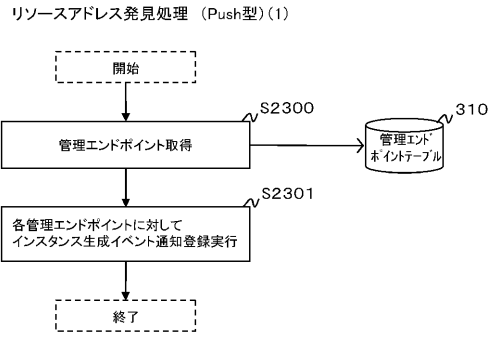
リソースアドレス通知イベントメッセージ(1)

```

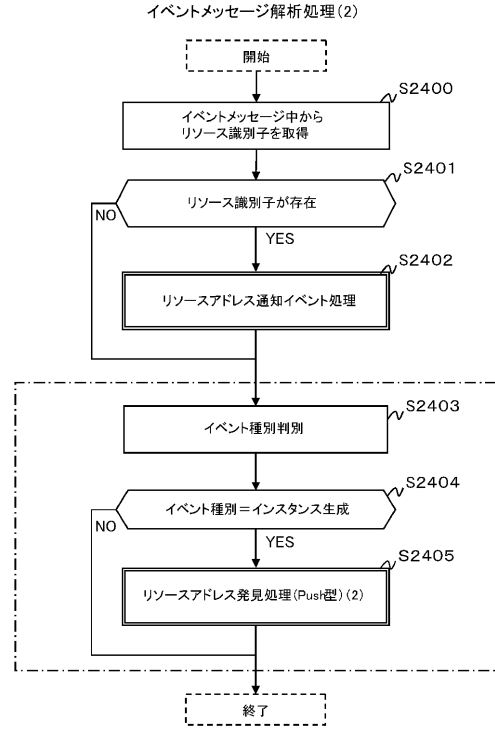
<?xml version="1.0" encoding="UTF-8" ?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <wsa:Action>...</wsa:Action>
    <wsa:To>http://192.168.0.100:8090/EventConsumer</wsa:To>
    <wsman:ResourceURI>http://schemas.foo/cmdb/1/EventConsumer.xsd</wsman:ResourceURI>
    <wsman:SelectorSet>
      <wsman:Selector Name="ResourceID">OS_1</wsman:Selector>
    </wsman:SelectorSet>
    <wsa:From>
      <wsa:Address>http://192.168.0.102:8080/wsman</wsa:Address>
      <wsa:ReferenceParameters>
        <wsman:ResourceURI>
          http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_OperatingSystem
        </wsman:ResourceURI>
        <wsman:SelectorSet>
          <wsman:Selector Name="CSCreationClassName">CIM_ComputerSystem</wsman:Selector>
          <wsman:Selector Name="CSName">XYZ</wsman:Selector>
          <wsman:Selector Name="CreationClassName">CIM_OperatingSystem</wsman:Selector>
          <wsman:Selector Name="Name">YYY_OS</wsman:Selector>
        </wsman:SelectorSet>
      </wsa:ReferenceParameters>
    </wsa:From>
  </soap:Header>
  <soap:Body>
    <im:CIM_InstModification>
      <im:IndicationIdentifier>...</im:IndicationIdentifier>
      <im:IndicationTime>...</im:IndicationTime>
      <im:PerceivedSeverity>...</im:PerceivedSeverity>
      <im:PreviousInstance>...</im:PreviousInstance>
      <im:SourceInstance>...</im:SourceInstance>
      <im:SourceInstanceHost>192.168.0.102</im:SourceInstanceHost>
      <im:SourceInstanceModelPath>...</im:SourceInstanceModelPath>
    </im:CIM_InstModification>
  </soap:Body>
</soap:Envelope>

```

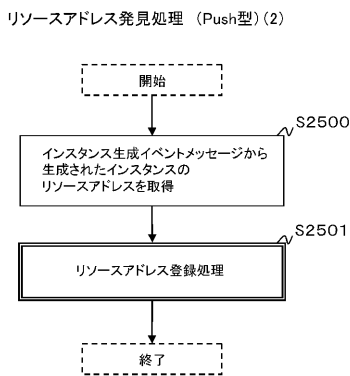
【図 2 3】



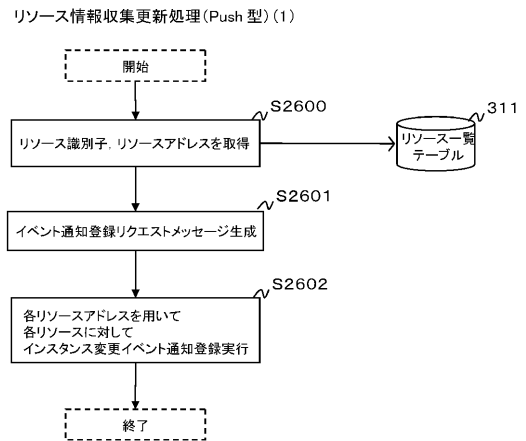
【図 2 4】



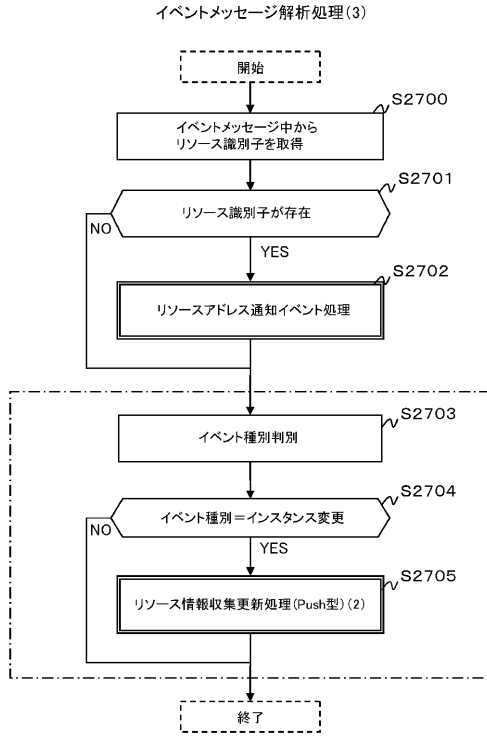
【図 2 5】



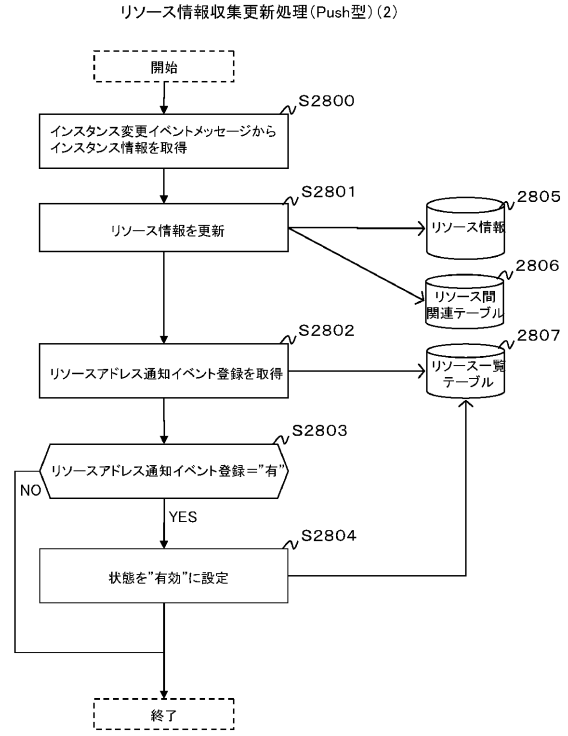
【図 2 6】



【 図 2 7 】



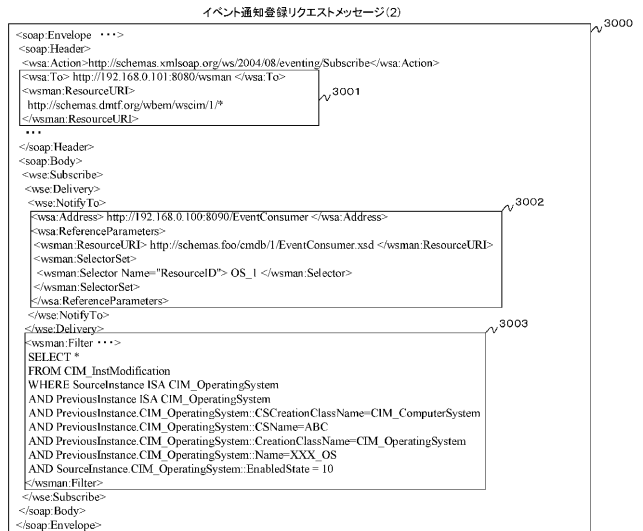
【 図 2 8 】



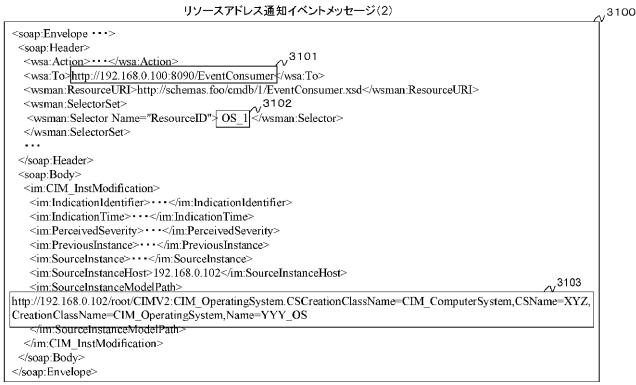
【 図 2 9 】



【 図 3 0 】



【 図 3 1 】



【 図 3 2 】

