

(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2005-50347  
(P2005-50347A)

(43) 公開日 平成17年2月24日(2005.2.24)

(51) Int.Cl.<sup>7</sup>  
G06F 12/00

F I  
G06F 12/00 520P

テーマコード (参考)  
5B082

審査請求 未請求 請求項の数 17 O L (全 10 頁)

(21) 出願番号	特願2004-222289 (P2004-222289)	(71) 出願人	500046438
(22) 出願日	平成16年7月29日 (2004.7.29)		マイクロソフト コーポレーション
(31) 優先権主張番号	10/630, 130		アメリカ合衆国 ワシントン州 9805
(32) 優先日	平成15年7月29日 (2003.7.29)		2-6399 レッドモンド ワン マイ
(33) 優先権主張国	米国 (US)		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	スティア デビッド キャパーズ
			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 27
			ストリート 9844

最終頁に続く

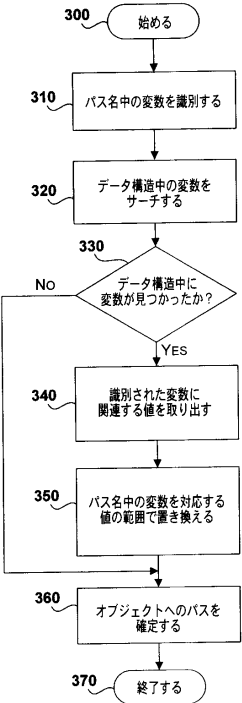
(54) 【発明の名称】 実行時バインディング／動的パス名解決の方法および装置

(57) 【要約】

【課題】 実行時バインディング／動的パス名解決の方法および装置を提供すること。

【解決手段】 実行時バインディング／動的パス名解決は、ユーザモードまたはカーネルモードで動作する変数識別子、カーネルモードで動作するパス名エンジンおよびデータ構造によって実施され得る。変数識別子はパス名中の変数を識別する。パス名エンジンは、オブジェクトファイルへのパスが確定されるように変数を文字列に変換することによって変数を評価する。パス名エンジンは、変数をデータ構造に記憶された対応する値にマップすることによって変数の可能な文字列値の範囲を決定する。使用可能な値がデータ構造での記憶のために選択された値だけであるから、各変数は明確な値の範囲を有する。値を確定する機構は拡張可能であり、いつでも変数／値対をデータ構造中に実装することができる。

【選択図】 図3



## 【特許請求の範囲】

## 【請求項 1】

コンピュータ実行可能コンポーネントを有するコンピュータ可読媒体であって、  
変数 / 値マッピングを含むデータ構造と、  
パス名中の変数を識別し、前記識別された変数に関連する値を前記データ構造から取り出し、前記変数が前記データ構造からの前記値で置き換えられるように前記パス名を変更するように構成されたパス名リゾルバと  
を含むことを特徴とするコンピュータ可読媒体。

## 【請求項 2】

前記パス名リゾルバが、前記パス名中の前記変数を識別するように構成された変数識別子を含むことを特徴とする請求項 1 に記載のコンピュータ可読媒体。 10

## 【請求項 3】

前記パス名リゾルバが、  
前記データ構造中の前記変数をサーチするステップと、  
前記対応する値にアクセスするステップと、  
パスがオブジェクトの場所を識別するように前記パス名中の前記変数の代わりに前記値を挿入するステップと  
を実施することによって前記変数を識別するように構成されたパス名エンジンを含むことを特徴とする請求項 1 に記載のコンピュータ可読媒体。

## 【請求項 4】

前記データ構造がオペレーティングシステムのカーネルモードに実装されることを特徴とする請求項 1 に記載のコンピュータ可読媒体。 20

## 【請求項 5】

変数 / 値対がユーザによって前記データ構造に定義されることを特徴とする請求項 1 に記載のコンピュータ可読媒体。

## 【請求項 6】

変数 / 値対が、前記コンピュータ可読媒体が動作する状況によって前記データ構造に定義されることを特徴とする請求項 1 に記載のコンピュータ可読媒体。

## 【請求項 7】

コンピュータ実行可能命令を有するコンピュータ可読媒体であって、  
変数を含むパス名を受け取ること、  
前記変数をデータ構造中の対応する値にマップすることによって前記パス名を解決すること、  
前記解決されたパス名によって指し示されるオブジェクトにハンドルを返すこと、および  
前記データ構造に変数 / 値対を付加することによって前記データ構造を拡張すること  
を含むことを特徴とするコンピュータ可読媒体。 30

## 【請求項 8】

前記パス名中の前記変数を識別することをさらに含むことを特徴とする請求項 7 に記載のコンピュータ可読媒体。 40

## 【請求項 9】

オペレーティングシステムのカーネルに実装されたデータ構造中の前記変数をサーチすることをさらに含むことを特徴とする請求項 8 に記載のコンピュータ可読媒体。

## 【請求項 10】

前記パス名中の変数の代わりに前記変数に対応する値を挿入することをさらに含むことを特徴とする請求項 9 に記載のコンピュータ可読媒体。

## 【請求項 11】

拡張可能なデータ構造を用いて符号化されたコンピュータ可読媒体であって、  
パス名中に含まれる変数の識別を含む第 1 のフィールドと、  
値を含む前記パス名がオブジェクトを指し示すように動作する前記変数の値を含む第 2 50

のフィールドと

を含むことを特徴とするコンピュータ可読媒体。

【請求項 1 2】

前記データ構造がオペレーティングシステムのカーネルに記憶されることを特徴とする請求項 1 1 に記載のコンピュータ可読媒体。

【請求項 1 3】

前記データ構造が変数 / 値対を付加することによって拡張可能であることを特徴とする請求項 1 1 に記載のコンピュータ可読媒体。

【請求項 1 4】

コンピュータで実施される方法であって、

オブジェクトへのアクセスを要求するコンポーネントによって提供されたパス名中の変数を識別するステップと、

前記変数をデータ構造中の対応する値にマップするステップと、

解決されたパス名が前記オブジェクトを指し示すパスを作成するように、前記パス名中の変数を前記対応する値で置き換えることによって前記パス名を変更するステップと、

前記要求元コンポーネントに前記変更されたパス名を返すステップと

を含むことを特徴とする方法。

【請求項 1 5】

変数を含むパス名を受け取るステップをさらに含むことを特徴とする請求項 1 4 に記載の方法。

【請求項 1 6】

オペレーティングシステムのカーネルに実装された前記データ構造中の変数をサーチするステップをさらに含むことを特徴とする請求項 1 4 に記載の方法。

【請求項 1 7】

請求項 1 4 に記載の方法を実施するコンピュータ実行可能命令を含むことを特徴とするコンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、計算処理システムに関し、より詳細には、計算処理システム上に記憶されたリソースの場所を識別するパス名に関する。

【背景技術】

【0002】

コンピュータオペレーティングシステムでは、パス名は、ファイルまたはディレクトリの場所を識別する一続きの英数字である。最も単純な形式のパス名がファイル自体の名前である。オペレーティングシステムは、パス名がファイル名で指定されると、現在の作業ディレクトリ中のファイルを探す。そのファイルが異なるディレクトリにある場合には、オペレーティングシステムがそのファイルを探し出せるようにパスが指定される。パス名をオペレーティングシステム言語に変換するプロセスはパス名解決 (path name resolution) と呼ばれる。

【発明の開示】

【発明が解決しようとする課題】

【0003】

既存のシステムの 1 つの問題は、パス名が本質的に静的なことである。現在のところ、カスタマイズ可能な、またはパスのユーザの状況で評価されるパス名コンポーネントのための機構はない。パス名が作成されると、どのユーザがログオンするか、どのマシン上でそのパス名解決が行われるかなどに関わりなく、その同じパス名が同じデータに解決されることになる。この限界が、かねてより、コンピュータシステムのユーザおよび設計者を悩ませている。

【0004】

10

20

30

40

50

動的パス名解決の機構は、現在に至るまで、当業者によって成し遂げられていない。

【課題を解決するための手段】

【0005】

本発明の一態様によれば、コンピュータ実行可能コンポーネントを有するコンピュータ可読媒体は、変数／値マッピングおよびパス名リゾルバを含む。このパス名リゾルバは、パス名中の変数を識別し、データ構造から識別された変数に関連する値を取り出し、その変数がそのデータ構造からの値で置き換えられるようにパス名を変更するように構成される。

【0006】

本発明の別の態様によれば、コンピュータ可読媒体はコンピュータ実行可能命令を有する。このコンピュータ実行可能命令は、変数を含むパス名を受け取ること、その変数をデータ構造中の対応する値にマップすることによってパス名を解決すること、解決されたパス名によって指し示されたオブジェクトにハンドルを返すこと、およびデータ構造に変数／値対を付加することによってデータ構造を拡張することを含む。

【0007】

本発明の別の態様によれば、コンピュータ可読媒体は、拡張可能なデータ構造を用いて符号化される。この拡張可能なデータ構造は、第1のフィールドおよび第2のフィールドを含む。第1のフィールドはパス名に含まれる変数の識別を含む。第2のフィールドはその変数の値を含む。パス名はオブジェクトを指し示すように動作する値を含む。

【0008】

本発明の別の態様によれば、コンピュータで実施される方法は、オブジェクトへのアクセスを要求するコンポーネントによって提供されるパス名中の変数を識別するステップ、その変数をデータ構造中の対応する値にマップするステップ、解決されたパス名がそのオブジェクトを指し示すパスを作成するようにパス名中の変数を対応する値で置き換えることによってパス名を変更するステップ、および要求元コンポーネントに変更されたパス名を返すステップを含む。

【発明を実施するための最良の形態】

【0009】

手短に言うと、本発明は、実行時バインディング／動的パス名解決の方法およびシステムに関する。このパス名解決は、変数識別子、パス名エンジン、およびデータ構造によって実施され得る。パス名によるオブジェクトへのアクセス要求時に、変数識別子はパス名中の変数を識別する。パス名エンジンは、変数／値マッピングを持つデータ構造を参照することによってその変数を評価する。データ構造は、現ユーザの状況で記憶することができる。パス名エンジンは、パス名中の変数をデータ構造からのそれに対応する値で置き換えることによってパス名を変更し、変更されたパス名を返す。

【0010】

以下では、まず、本発明の実施形態を実装することのできる例示的計算処理環境の一例を参照して本発明を説明する。次に、本発明の具体的な一実装形態の詳細例を説明する。また、この具体的実装形態の若干の詳細に関連する代替実装形態も含まれ得る。本発明の実施形態は本明細書で説明するものに限らないことが理解されるであろう。

【0011】

例示的動作環境

図1を参照すると、本発明を実装するためのシステムの一例は、計算処理装置100などの計算処理装置を含む。ごく基本的な構成では、計算処理装置100は、通常、少なくとも1つの処理装置102およびシステムメモリ104を含む。計算処理装置の正確な構成および種類に応じて、システムメモリ104は、揮発性(RAMなど)とすることも、不揮発性(ROM、フラッシュメモリなど)とすることも、それら2つの何らかの組み合わせとすることもできる。システムメモリ104は、通常、オペレーティングシステム105、1つまたは複数のプログラムモジュール106を含み、プログラムデータ107も含み得る。オペレーティングシステム105は、本発明によるパス名リゾルバ120をさ

らに含む。この基本的構成を、図1の破線108内のコンポーネントによって示す。

#### 【0012】

計算処理装置100は、その他の機構または機能も持ち得る。例えば、計算処理装置100は、磁気ディスク、光ディスク、またはテープなどの付加的（取り外し可能および/または取り外し不能）データ記憶装置を含むこともできる。そうした付加的記憶機構を図1に取り外し可能記憶装置109および取り外し不能記憶装置110で示す。コンピュータ記憶媒体には、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、またはその他のデータなどの情報を記憶するための任意の方法または技術で実装された、揮発性および不揮発性、取り外し可能および取り外し不能の媒体が含まれ得る。システムメモリ104、取り外し可能記憶装置109および取り外し不能記憶装置110は、すべて、コンピュータ記憶媒体の例である。コンピュータ記憶媒体には、それだけに限らないが、RAM、ROM、EEPROM、フラッシュメモリまたはその他のメモリ技術、CD-ROM、デジタル多用途ディスク(DVD)またはその他の光記憶機構、磁気カセット、磁気テープ、磁気ディスク記憶装置またはその他の磁気記憶装置、あるいは所望の情報を記憶するのに使用することができ、計算処理装置100からアクセス可能な他の任意の媒体が含まれる。そうしたどんなコンピュータ記憶媒体も計算処理装置100の一部とすることができる。また、計算処理装置100は、キーボード、マウス、ペン、音声入力装置、タッチ入力装置などの入力装置112も備えることができる。ディスプレイ、スピーカ、印刷装置などの出力装置114も含まれ得る。これらの装置は当分野で公知であり、ここで詳細に論じる必要はない。

10

20

#### 【0013】

計算処理装置100は、この装置がネットワークなどを介して他の計算処理装置118とやりとりすることを可能にする通信接続116も含み得る。通信接続116は、通信媒体の一例である。通信媒体は、通常、搬送波やその他の搬送機構などの変調されたデータ信号に、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、またはその他のデータによって実施することができ、任意の情報送達媒体を含む。「変調されたデータ信号」という用語は、その特性セットのうちの1つまたは複数を持つ信号、あるいは信号中に情報を符号化するような方式で変更された信号を意味する。例をあげると、それだけに限らないが、通信媒体には、有線ネットワークや直接配線接続などの有線媒体、および音響、無線周波数、赤外線、その他の無線媒体などの無線媒体が含まれる。本明細書で使用するコンピュータ可読媒体という用語は、記憶媒体と通信媒体の両方を含むものである。

30

#### 【0014】

##### パス名リゾルバの構造

図2を参照すると、本発明を実施するための環境の一例を示す構成図が示されている。図2に示す例示的環境は、本発明によるパス名リゾルバ120を含むファイルシステムのためのバス/リンクサブシステム環境200である。この環境は、要求元コンポーネント205を含み、これは、パス名206によってオブジェクト（オブジェクト212など）へのアクセスを要求する任意のコンポーネントとすることができる。要求元コンポーネント205は、例えば、電子メールメッセージ中の埋め込みパスまたはユニフォームリソースロケータ(URL)、スクリプト中に含まれる変数を持つパスのうちの任意の1つまたは複数とすることができる。

40

#### 【0015】

この実施形態では、オブジェクト212を参照するのに使用されるパス名206は変数を含む。例えば、そうしたパス名の一例として「\\server\share\@user\phonenumber.doc」が考えられる。この例では、「user」という用語が、「@」文字の使用によって変数であると識別される。他の文字を使用することもできる。

#### 【0016】

パス名リゾルバ120は、変数識別子210、パス名エンジン220、およびデータ構造222を含む。この具体的実装形態では、変数識別子210はユーザモードで動作し、

50

パス名エンジン 2 2 0 およびデータ構造はカーネルモードにある。パス名エンジン 2 2 0 は、要求元コンポーネント 2 0 5 に応答し、変数識別子 2 1 0 に変数評価を要求する。あるいは、要求元コンポーネント 2 0 5 は、変数評価を要求するために変数識別子 2 1 0 と直接やりとりする。

#### 【 0 0 1 7 】

変数識別子 2 1 0 はパス名 2 0 6 中の変数を識別する。この具体的実装形態では、パス名 2 0 6 中の一意の接頭部、この例では「@」文字を見つけ出すことによって変数が識別される。一意の接頭部は、1 つまたは複数の文字の任意の組み合わせとすることもできる。変数識別子 2 1 0 は、識別された変数をパス名エンジン 2 2 0 に返す。

#### 【 0 0 1 8 】

パス名エンジン 2 2 0 は、データ構造 2 2 2 を参照することによって変数を評価する。データ構造 2 2 2 は、対応する値にマップされた多数の変数名を記憶する。それらの値は、文字列とすることも、文字列を生じさせる任意の実行可能コードへの参照点とすることもできる。パス名エンジン 2 2 0 は、データ構造 2 2 2 中の変数をサーチする。対応する値は、値がデータ構造 2 2 2 中に記憶されている場合に限り見つかる。したがって、使用可能な値はデータ構造 2 2 2 に記憶された値だけであるので、各変数は明確な値の範囲を有する。データ構造 2 2 2 中にその変数が存在しない場合には、パス名は、変数を含むパス名ではなく、文字列であると識別される。これにより、変数を識別する接頭部として適正な文字を使用することが可能になる。代替として、不正な文字を使用して変数を識別することも可能であり、その場合は、変数がデータ構造 2 2 2 中に存在しなかった場合には、エラーが返されるはずである。

#### 【 0 0 1 9 】

変数に適した値がデータ構造 2 2 2 中に存在する場合には、その値が変数識別子 2 1 0 に返される。次いで、変数識別子 2 1 0 は、変数を変数の値で置き換えることによって、要求元コンポーネント 2 0 5 が提供したパス名 2 0 6 を変更することができる。このようにして、変更されたパス名はオブジェクト 2 1 2 を指し示す。次いで、変数識別子 2 1 0 は、変更されたパス名をそのオブジェクト 2 1 2 のハンドルに解決するように要求し、そのハンドルを要求元コンポーネント 2 0 5 に返すことができる。

#### 【 0 0 2 0 】

一実施形態では、異なるユーザが異なる 1 組の変数 / 値マッピングを持つように、データ構造 2 2 2 を各ユーザに一意とすることができる。これは、どのユーザがログオンするかに基づいて、テキスト上同一のパス名 ( 変数を含む ) を異なるオブジェクトにマップできるようにする。例えば、「@ u s e r n a m e 」などの変数が動的パス名に使用される場合には、この変数は、どのユーザがその計算処理システムにログオンしたかに基づいて異なる値に評価されるはずである。別の例では、「@ L 1 4 6 」などの変数がオフィスの場所を指すことができる。その変数を、そのオフィスの現在の占有者に基づく名前を持つ、そのオフィス内の印刷機を識別するパス名中で使用することができる。したがって、データ構造 2 2 2 は、オフィスの場所の変数をそのオフィスの現在の占有者のユーザ名にマップする表を含み得る。他の多くの代替手段も、当業者には明らかになるであろう。

#### 【 0 0 2 1 】

この実装形態では、任意の特権のあるアプリケーションがデータ構造 2 2 2 に値および / または変数を書き込み、したがって、この機構を拡張可能にすることができる。あるいは、恣意的に値を付加することができないように、有限数の値を用いてデータ構造 2 2 2 のサイズを固定することもできる。ファイルシステム 2 0 0 の要件によって決定される可能な値の範囲を持つデータ構造 2 2 2 で一意の変数を定義することも可能である。

#### 【 0 0 2 2 】

変数に対応する値は様々に定義することができる。例えば、値は、ユーザ定義とすることもでき、コンテキスト特有とすることもできる ( 例えば、値は、オペレーティングシステムまたは環境の状態によって定義され得る )。「u s e r」と定義される変数に対応する値など、広範に使用されるいくつかの基本的値を提供することもできる。

10

20

30

40

50

## 【0023】

実行時バインディング／動的パス名解決のプロセス

図3を参照すると、本発明に従って実施されるパス名解決プロセスの例示的概要の論理的流れ図が示されている。プロセスは、要求元コンポーネントがパス名によって識別されるオブジェクトへのアクセスを要求している開始ブロック300から始まる。続いてブロック310に進む。

## 【0024】

ブロック310では、パス名中の変数を識別する。変数を、パス名をその構成部分に構文解析し、一意の接頭文字、「@」などをサーチすることによって識別し得る。変数が見つかった場合には、ブロック320に進む。

## 【0025】

ブロック320では、変数／値マッピングを含む表などのデータ構造でその変数のサーチを実施する。判断ブロック330で、データ構造中にその変数のエントリが見つかったかどうか判定する。データ構造中に変数が見つかった場合には、ブロック340に進む。データ構造中に変数が見つからなかった場合には、ブロック360に進み、その変数を、解決を要求しない文字列として処理する。

## 【0026】

ブロック340で、データ構造から変数に関連する値を取り出し、ブロック350に進む。ブロック350で、パス名中の変数を対応する値で置き換えて完全なパス名を形成し、ブロック360に進む。ブロック360で、要求元コンポーネントに完全なパス名を返す。プロセスはブロック370に進み、パス名解決を完了する。

## 【0027】

データ構造に変数／値対を付加するプロセス

図4を参照すると、本発明に従って実施される、データ構造を拡張するプロセスの一例の論理的流れ図が示されている。データ構造は、新規の変数／値対を付加することによって拡張することができる。

## 【0028】

プロセスはブロック400から開始し、そこでシステム管理者は、より多くの変数および／または対応する値をデータ構造に含めることを決定する。判断ブロック410で、オペレーティングシステムが実行時バインディング／動的パス名をサポートしているかどうか判定する。オペレーティングシステムが実行時バインディング／動的パス名をサポートしている場合には、ブロック430に進む。オペレーティングシステムが実行時バインディング／動的パス名をサポートしていない場合には、ブロック420に進む。ブロック420で、システム管理者に、オペレーティングシステムが実行時バインディング／動的パス名をサポートしていないことを知らせるエラーメッセージを発行し、ブロック460に進む。

## 【0029】

ブロック430で、システム管理者に、変数／値対を入力するよう促す。ブロック440に進む。ブロック440で、システム管理者がシステムに変数／値対を入力する。ブロック450に進む。ブロック450で、データ構造に変数／値対を付加する。プロセスはブロック460で終了する。

## 【0030】

本発明によるデータ構造への変数／値対の付加は、ユーザログオン時などのスクリプトによっても実施され得ることが理解されるであろう。また、事前定義の変数／値マッピングを含むポリシーを設定することもできる。同様に、保護モードでデータ構造を変更するのに十分な許可を含むソフトウェアコンポーネントが、データ構造中の変数／値マッピングにアクセスし、それを変更することも可能である。このように、個々のユーザ、そうしたいくつかのコンポーネントが実行される個々のハードウェアまたはソフトウェアの状況で、あるいは他の何らかの基準に基づいて、変数／値マッピングを作成することができる。

10

20

30

40

50

## 【 0 0 3 1 】

以上の明細書、例およびデータは、本発明の構成の製造および使用についての完全な説明を提供するものである。本発明の精神および範囲を逸脱することなく本発明の多くの実施形態が実施され得るので、本発明は、添付の特許請求の範囲に存するものである。

## 【 図面の簡単な説明 】

## 【 0 0 3 2 】

【 図 1 】 本発明の例示的实施形態で使用するのことができる計算処理装置の一例を示す図である。

【 図 2 】 本発明を実施するための環境の一例を示す構成図である。

【 図 3 】 本発明に従って実施されるパス名解決プロセスの概要の一例を示す論理的流れ図である。 10

【 図 4 】 本発明に従って実施されるデータ構造拡張プロセスの概要の一例を示す論理的流れ図である。

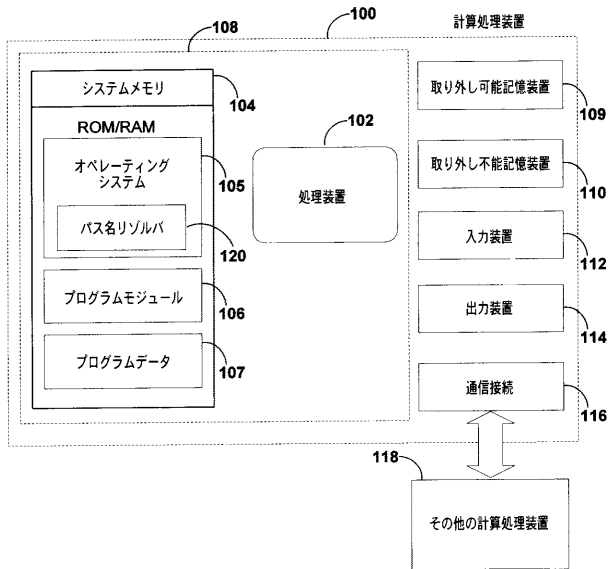
## 【 符号の説明 】

## 【 0 0 3 3 】

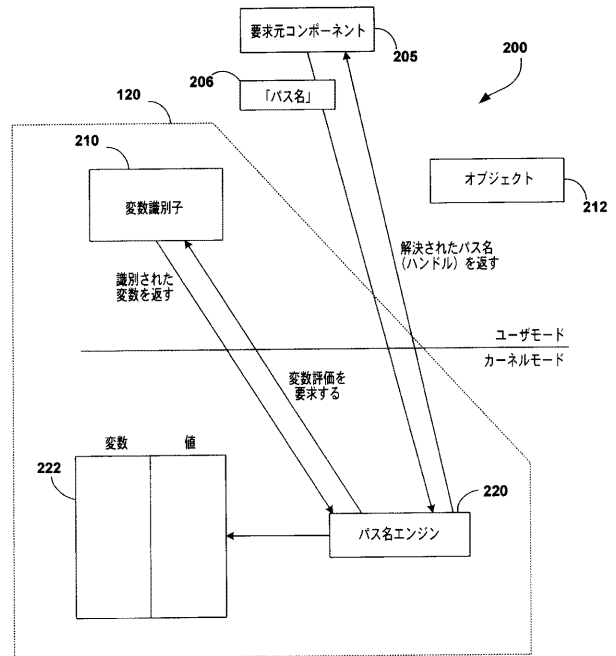
1 0 0	計算処理装置	
1 0 2	処理装置	
1 0 4	システムメモリ	
1 0 5	オペレーティングシステム	
1 0 6	プログラムモジュール	20
1 0 7	プログラムデータ	
1 0 9	取り外し可能記憶装置	
1 1 0	取り外し不能記憶装置	
1 1 2	入力装置	
1 1 4	出力装置	
1 1 6	通信接続	
1 1 8	その他の計算処理装置	
1 2 0	パス名リゾルバ	
2 0 5	要求元コンポーネント	
2 0 6	「パス名」	30
2 1 0	変数識別子	
2 1 2	オブジェクト	
2 2 0	パス名エンジン	
2 2 2	変数 / 値	



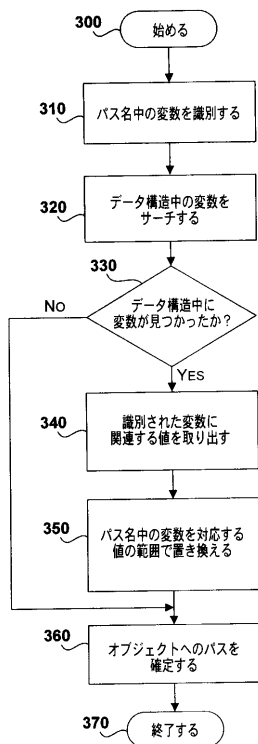
【図 1】



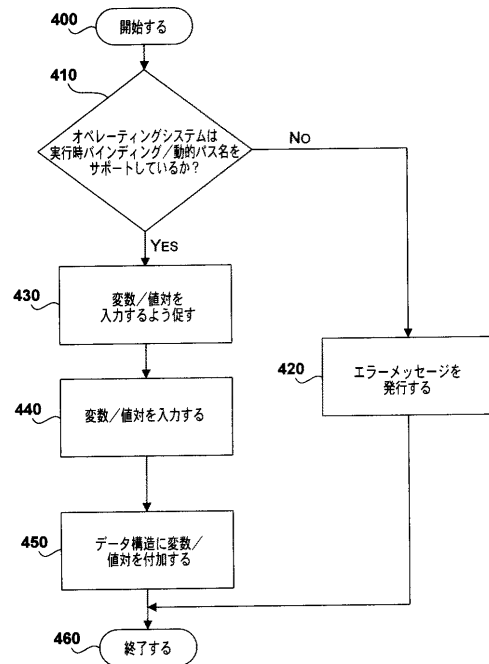
【図 2】



【図 3】



【図 4】



---

フロントページの続き

(72)発明者 デューイ ブライアン

アメリカ合衆国 9 8 1 0 5 ワシントン州 シアトル 2 9 アベニュー ノースイースト 5  
7 3 2

(72)発明者 ナーガル ラジェブ

アメリカ合衆国 9 8 0 7 4 ワシントン州 サマミッシュ 2 0 5 プレイス ノースイースト  
1 9 0 1

(72)発明者 ゴータ ピシャル ブイ .

アメリカ合衆国 9 8 0 0 7 ワシントン州 ベルビュー ノースイースト 4 3 プレイス 1  
4 6 2 3 ナンバー 1 5 0 7

F ターム(参考) 5B082 EA01