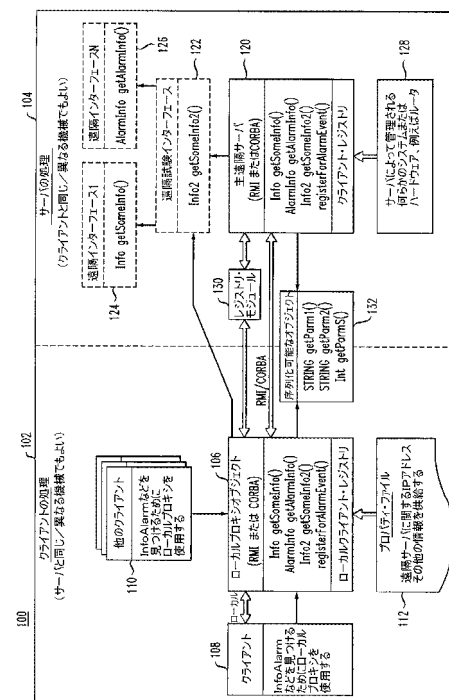




【特許請求の範囲】

【請求項 1】

クライアントとサーバとの間の通信処理をサポートするのに有用なシステムであって、
本システムは分散形オブジェクト指向システムであり、
オブジェクト指向サーバと、
前記オブジェクト指向サーバと通信するように動作する遠隔インターフェースと、
前記オブジェクト指向サーバに関連するレジストリ・モジュールと、
前記クライアントのために情報を処理するように動作するローカルプロキシオブジェクト
トであって、前記オブジェクト指向サーバ、前記遠隔インターフェースおよび前記レジス
トリ・モジュールと通信するように動作するローカルプロキシオブジェクトと、 10
前記オブジェクト指向サーバに関する情報を記憶するように動作する記憶ファイルとを
含むシステム。

【請求項 2】

前記オブジェクト指向サーバが遠隔メソッド呼び出し（RMI）サーバである、請求項
1 に記載のシステム。

【請求項 3】

前記レジストリ・モジュールが遠隔メソッド呼び出し（RMI）レジストリ・モジュ
ールである、請求項 1 に記載のシステム。

【請求項 4】

前記ローカルプロキシオブジェクトが遠隔メソッド呼び出し（RMI）プロトコルに準
拠している、請求項 1 に記載のシステム。 20

【請求項 5】

前記オブジェクト指向サーバが共通オブジェクト・リクエスト・ブローカー・アーキ
テクチャ（CORBA）サーバである、請求項 1 に記載のシステム。

【請求項 6】

前記レジストリ・モジュールが共通オブジェクト・リクエスト・ブローカー・アーキ
テクチャ（CORBA）ネームサービスである、請求項 1 に記載のシステム。

【請求項 7】

前記ローカルプロキシオブジェクトが共通オブジェクト・リクエスト・ブローカー・
アーキテクチャ（CORBA）プロトコルに準拠している、請求項 1 に記載のシステム。 30

【請求項 8】

分散形オブジェクト指向システムにおいてクライアントとサーバとの間の通信をサポ
ートするのに有用な方法であって、
主遠隔サーバを初期化する工程と、
前記主遠隔サーバをレジストリ・モジュールに登録する工程と、
前記主遠隔サーバにアクセスするためにクライアントによってローカルプロキシオブ
ジェクトのインスタンスを生成する工程と、
前記ローカルプロキシオブジェクトによって前記主遠隔サーバ上の情報を得る工程と、
前記ローカルプロキシオブジェクトによって前記主遠隔サーバにコンタクトする工程と
、 40
前記ローカルプロキシオブジェクトと前記主遠隔サーバとの間の接続を確立する工程と
、
前記クライアントによって、前記ローカルプロキシオブジェクトを通じて前記主遠隔サ
ーバからの情報を要求する工程と、
前記主遠隔サーバから前記ローカルプロキシオブジェクトへと前記情報を返送する工程
と、
前記ローカルプロキシオブジェクトから前記クライアントへと前記情報を返送する工程
とを含む方法。

【請求項 9】

分散形オブジェクト指向システムにおいてクライアントとサーバとの間の通信をサポ
ート 50

トするのに有用な方法であって、
主遠隔サーバを初期化する工程と、
前記主遠隔サーバをレジストリ・モジュールに登録する工程と、
前記主遠隔サーバにアクセスするために、クライアントによってローカルプロキシオブジェクトのインスタンスを生成する工程と、
前記ローカルプロキシオブジェクトによって前記主遠隔サーバ上の情報を得る工程と、
前記ローカルプロキシオブジェクトによって前記主遠隔サーバにコンタクトする工程と、
、
前記ローカルプロキシオブジェクトと前記主遠隔サーバとの間の接続を確立する工程と、
、
前記クライアントによって、前記ローカルプロキシオブジェクトを通じて前記主遠隔サーバからの情報を要求する工程と、
試験クライアントによって、前記ローカルプロキシオブジェクトを通じて警報事象を前記主遠隔サーバに登録する工程と、
登録に関するクライアントの情報を前記ローカルプロキシオブジェクトによってセーブする工程と、
サーバの故障および再スタートの場合に前記クライアントを自動的に再登録することを可能にするために、前記ローカルプロキシオブジェクトによって定期的に前記主遠隔主サーバにアクセスする工程とを含む方法。

10

20

30

40

50

【請求項 10】

分散形オブジェクト指向システムにおいてクライアントとサーバとの間の通信をサポートするのに有用な方法であって、
主遠隔サーバを初期化する工程と、
前記主遠隔サーバをレジストリ・モジュールに登録する工程と、
前記主遠隔サーバにアクセスするために、クライアントによってローカルプロキシオブジェクトのインスタンスを生成する工程と、
前記ローカルプロキシオブジェクトによって前記主遠隔サーバ上の情報を入手する工程と、
前記ローカルプロキシオブジェクトによって前記主遠隔サーバにコンタクトする工程と、
、
前記ローカルプロキシオブジェクトと前記主遠隔サーバとの間の接続を確立する工程とを含む方法。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、分散形オブジェクト指向システム用のクライアント側ローカルプロキシ（local proxy）オブジェクトを供給する方法および装置に関する。さらに特定すると本発明は、或る形式では、Java（登録商標） RMI プロトコルベースのシステム用のクライアント側ローカルプロキシオブジェクトの導入に関する。また別の形式では、本発明は Common Object Request Broker Architecture（CORBA）プロトコルに適合可能である。

【背景技術】**【0002】**

本発明は特に Java（登録商標） RMI に準拠するシステム内でローカルプロキシを供給する技術を指向し、したがって特にそれに関連して説明されるであろうが、本発明が他の分野および用途に有用性を有する可能性があることは理解されるであろう。例えば上述したように、本発明は CORBA に準拠するシステムのようなローカルプロキシが所望されるその他の状況で使用される可能性がある。

【0003】

背景として、分散形オブジェクト指向システムは多くのプロジェクト／プロダクトの共

通部分になってきた。Sun Microsystemsから出されたJava（登録商標）の遠隔メソッド呼び出し（RMI）、Object Management Group Organizationから出された共通オブジェクト・リクエスト・ブローカー・アーキテクチャ（Common Object Request Broker Architecture, CORBA）、World-wide Web Consortium（W3C）から出されたSimple Object Access Protocol（SOAP）はオブジェクト指向の分散形システムを導入するのに使用される開放された標準技術のいくつかである。

【発明の開示】

【発明が解決しようとする課題】

10

【0004】

RMI（またはCORBA）技術の使用は分散形オブジェクト指向技術の導入を簡素化する。これらの技術、ならびにこれらのタイプのシステムの構成要素のための付随するプログラミング言語と命令構造はよく知られている。それらは遠隔サーバ（remote server）によって供給される遠隔サービスの呼び出し、およびプロセスもしくは機械の境界を横切って送られるソフトウェア・オブジェクトの通過を可能にする。しかしながら、クライアント・アプリケーション側がRMIまたはCORBAに準拠したサーバ／システムを使用することにはまだ複雑さが存在する。

【0005】

例を挙げると、図1を参照すると既知のシステム10が示されている。システム10はクライアント12と遠隔サーバ14を含む。例示したように、サーバ14はRMIに準拠したサーバであって遠隔インターフェース16、RMIレジストリ・モジュール18、およびルータ20と通信する。

20

【0006】

そのようなシステムでは、導入に数多くの難題が存在する。例えば、クライアント12はRMIプロトコルを理解し、かつ遠隔インターフェース16を導入することを要求される。さらに、RMI準拠のサーバ14に対する参照を得るために、クライアント12はRMIレジストリ・モジュール18の位置を突きとめてアクセスすることを要求される。付け加えると、クライアント12はすべての遠隔異常、例えばサーバ・ダウンのメッセージを扱う能力を持たなければならない。最後に、システム10は、各々のクライアント・タイプがシステム内で機能できるように書かれており、かつそのように試験されたコードをすべてのクライアントが有することを要求するであろう。同様の難題がCORBA準拠のシステムに存在する。したがって、さらに効率的な技術が望まれる。

30

本発明は上記で引用した難題などを解決するような新規的で改良された取り組み方を検討する。

【課題を解決するための手段】

【0007】

分散形オブジェクト指向システム用のローカルプロキシオブジェクトを供給するための方法と装置が提供される。

本発明の一態様では、システムはオブジェクト指向サーバ、オブジェクト指向サーバと通信するように動作する遠隔インターフェース、オブジェクト指向サーバと結びついたレジストリ・モジュール、およびクライアントに関する情報を処理するように動作するローカルプロキシオブジェクトを含む。

40

【0008】

本発明のまた別の態様では、ローカルプロキシオブジェクトはオブジェクト指向サーバ、遠隔インターフェース、およびレジストリと通信するように動作する。

本発明のまた別の態様では、プロパティ・ファイルがサーバのIPアドレス、ポート番号などといった情報を含むオブジェクト指向サーバに関する情報を記憶するように動作する。

本発明のまた別の態様では、システムは遠隔メソッド呼び出し（RMI）に準拠する。

50

本発明のまた別の態様では、システムはCommon Object Request Broker Architecture (CORBA) に準拠する。

【0009】

本発明のまた別の態様では、方法は主遠隔サーバを初期化する工程と、主遠隔サーバをレジストリ・モジュールに登録する工程と、主遠隔サーバにアクセスするためにクライアントによってローカルプロキシオブジェクトのインスタンスを (i n s t a n t i a t e) 生成する工程と、ローカルプロキシオブジェクトによって主遠隔サーバ上の情報を得る工程と、ローカルプロキシオブジェクトによって主遠隔サーバと接触する工程と、ローカルプロキシオブジェクトと主遠隔サーバの間の接続を確立する工程と、ローカルプロキシオブジェクトを通じてクライアントによって主遠隔サーバから情報を要求する工程と、主遠隔サーバからローカルプロキシオブジェクトへと情報を返送する工程と、ローカルプロキシオブジェクトからクライアントへと情報を返送する工程とを含む。

【0010】

本発明のまた別の態様では、方法は主遠隔サーバを初期化する工程と、主遠隔サーバをレジストリ・モジュールに登録する工程と、主遠隔サーバにアクセスするためにクライアントによってローカルプロキシオブジェクトのインスタンスを生成する工程と、ローカルプロキシオブジェクトによって主遠隔サーバ上の情報を得る工程と、ローカルプロキシオブジェクトによって主遠隔サーバと接触する工程と、ローカルプロキシオブジェクトと主遠隔サーバの間の接続を確立する工程と、ローカルプロキシオブジェクトを通じてクライアントによって主遠隔サーバから情報を要求する工程と、ローカルプロキシオブジェクトを通じてクライアントによるいくつかの事象 (例えば警報事象) を主遠隔サーバに登録する工程と、登録に関する情報をローカルプロキシオブジェクトによってセーブする工程と、サーバの故障および再スタートの場合にクライアントの自動再登録を可能にするためにローカルプロキシオブジェクトによって主遠隔サーバに定期的にアクセスする工程とを含む。

【0011】

本発明のまた別の態様では、方法は主遠隔サーバを初期化する工程と、主遠隔サーバをレジストリ・モジュールに登録する工程と、主遠隔サーバにアクセスするためにクライアントによってローカルプロキシオブジェクトのインスタンスを生成する工程と、ローカルプロキシオブジェクトによって主遠隔サーバ上の情報を得る工程と、ローカルプロキシオブジェクトによって主遠隔サーバと接触する工程と、ローカルプロキシオブジェクトと主遠隔サーバの間の接続を確立する工程とを含む。

【0012】

本発明の応用性のさらなる範囲は下記で与えられる詳細な説明から明らかになるであろう。しかしながら、本発明の好ましい実施形態を示しているが、詳細な説明と特定の範例が単に具体的例示の方式で与えられており、それゆえに当業者にとって本発明の精神と範囲の中で様々な変更および改造が明白になるであろうことは理解されるべきである。

【0013】

本発明は装置の様々な部品類、および方法の複数工程の構造、配列、および組み合わせにあり、それにより、検討される複数の目的はこれ以降さらに充分に述べられ、特許請求項中で特に指摘され、かつ添付の図面類に例示されるように達成される。

【発明の効果】

【0014】

本発明により、分散形オブジェクト指向システム用のローカルプロキシオブジェクトを供給するための方法及び装置が提供される。

【発明を実施するための最良の形態】

【0015】

本発明によると、(例えばRMIもしくはCORBAに準拠する) 遠隔オブジェクト指向サーバにローカルプロキシオブジェクトを供給する技術はRMI / CORBAに準拠したシステムを使用してクライアントの視点から複雑さを隠すために導入される。この手法

の下で、RMI／CORBAサービスを使用するクライアントはRMI／CORBAプロトコル、およびRMIレジストリ（もしくはCORBA名前付けサービス）、遠隔サーバを突きとめ、遠隔インターフェースを導入し、遠隔の異常を捕捉することなどに含まれる複雑さを理解することを要求されない。クライアントは単純に遠隔サーバのローカルプロキシオブジェクトをローカルオブジェクトとして処理および使用し、サーバによって供給されるすべての遠隔サービスに、それらがローカルメソッド（local method）であるかのようにアクセスすることが可能である。遠隔サーバのローカルプロキシオブジェクトを使用すると、クライアントはRMI／CORBA技術の使用に気付くことすらない。とりわけ、オブジェクト指向の規格を使用する複雑さがクライアントによって認識もしくは処理される必要がなくなる。

10

【0016】

これはまた、サーバによって供給されるRMIまたはCORBAに準拠したサービスにアクセスするためにクライアント／ユーザが書き込むことを必要とするであろうコードの量も削減する。インターフェースを検査する努力が最小限にされる。ロードバランシング（そこでは多数のサーバのインスタンスが利用可能）もやはりローカルプロキシオブジェクトの内部に設けられることが可能である。

【0017】

ここで単に本発明の好ましい実施形態を例示する目的で示され、本発明を限定することを目的としない図面を参照すると、図2は本発明による範例のシステムの図を与えるものである。図示したように、RMIに準拠したクライアント・サーバのタイプのシステム100が例示されている。場合によっては、図示したようにシステム100はCORBA準拠であることも可能である。RMIとCORBAプロトコルの間で、コードと命令構造に僅かの違いがある可能性はあるが、それらのような代替のシステムは本発明を導入する目的に関して実質的に同じ動作をし、かつ実質的に同じ類似構造にされている。

20

【0018】

システム100では、クライアントの処理側102とサーバの処理側104が導入されている。クライアントの処理側102には、クライアント108と通信するローカルプロキシオブジェクト106が設けられる。他のクライアント110もやはりローカルプロキシオブジェクト106と通信する可能性がある。

【0019】

ローカルプロキシオブジェクト106はまた、遠隔サーバに関するIPアドレスおよびその他の情報を含むプロパティを記憶するためのデータベースのモジュール／ファイルを設けられる。ローカルプロキシオブジェクト106は、本発明に従ってローカルプロキシオブジェクトが遠隔サーバの処理側104と効率的に通信することを可能にするソフトウェアを含むかまたは有する。ローカルプロキシオブジェクトは、それ以外ではすべてのクライアント・タイプによってRMIもしくはCORBAレジストリに接触するのに使用される正確に同じ段階またはシステムのコードを使用し、それにより、サーバのIPアドレス／機械名、ポート番号などを使用して遠隔サーバの位置を突きとめる。これは標準的な方式であり、これによってRMIまたはCORBAに準拠したいかなるクライアントもレジストリ・モジュールと接触をして遠隔サーバの参照を入手する。さらに、ローカルプロキシオブジェクトが様々な遠隔メソッドを呼び出し、遠隔インターフェースを導入し、遠隔の異常を捕捉するのに使用するコードのシステムもやはり、いわゆるローカルプロキシオブジェクトが存在しないときにクライアントのソフトウェアが使用したであろうものと異なることはない。事実上、これは下にあるプロトコル（例えばRMI／CORBA）の知識を必要とするソフトウェア・コードのタイプであってそれはここではこのローカルプロキシオブジェクト内部の或る場所に包み込まれている。ローカルプロキシオブジェクトが多様な環境中に存在し、かつ多様な形式をとる可能性があることは理解されるべきである。しかしながら、或る形式では、それはパーソナル・コンピュータもしくはパーソナル・デジタル・アシスタントといったクライアント装置に存在するソフトウェア・コードの形式をとる。

30

40

50

【0020】

サーバの処理側104では、システムは遠隔インターフェース122、124および126と通信する主遠隔サーバ120を設けられる。主遠隔サーバ（もしくは遠隔サーバ）120もやはり主遠隔サーバによって管理されるシステムもしくはハードウェア、例えばルータ128と通信状態にある。サーバの処理側104はまた、この範例ではRMIレジストリのようなレジストリ・モジュール130を設けられる。もちろん、もしもシステムがRMIではなくてCORBAに準拠していれば、RMIレジストリ130はCORBA名前付けサービスで置き換えられるであろう。

【0021】

図2から分かり得るように、132で、順番に並べられたオブジェクトがクライアントのローカルプロキシオブジェクト106と主遠隔サーバ120の間で伝送されることが可能である。 10

【0022】

もちろん、図2に示したシステムが様々なハードウェアおよびソフトウェアの技術を使用して導入されることは理解されるべきである。このシステムは多様な形式をとる可能性があるが本発明の範囲内に入る。とりわけ、前に参照したように、このシステムはRMIもしくはCORBAに準拠した分散形システムあるいは同様の機能と構造を有する他のオブジェクト指向のシステムの両方の導入を可能にする形式をとることが可能である。

【0023】

図示した範例では、これは（複数）クライアント108、110と主遠隔サーバ120が同じ組織によって展開されるシステムであることが可能である。あるいはこれは主遠隔サーバ120が別の独立した製造製品であって異なる組織もしくは第三者の製造元によって設計されるケースのシステムであってもよい。例えば、このケースでは、主遠隔サーバ120はネットワーク・ルータ128のようなシステムまたはハードウェアを管理する別個の独立型製品である。これらの要素は株式市場のようなアクセス可能な情報を所有するエンティティを付随することが可能である。クライアント108もしくは複数クライアント110はローカルプロキシオブジェクト106を使用して主遠隔サーバ120からルータ128の情報を回収することを要求することが可能である。その目的のために、主遠隔サーバ120は管理されたルータ128のInfo1、Info2およびAlarmInfoを供給する。 20 30

【0024】

付け加えると、主遠隔サーバ120はルータ・ハードウェアをモニタおよび管理している事実上バックエンド型の遠隔サーバである。主遠隔サーバ120は多数のインターフェース（遠隔インターフェース1、2、N）124と126によって記述される様々なサービスを導入する。簡素化するためにこれら多数のインターフェースは遠隔試験インターフェース122と呼ばれる別のインターフェースに集められる。

【0025】

前に述べたように、本発明が無い場合に、これらのタスクを成し遂げるための技術はよく知られている。getSomeInfo()、getAlarmInfo()、getSomeInfo2()、およびregisterForAlarmEvent()といったコマンド／命令の使用が通常的である。 40

【0026】

本発明の導入の前は、クライアントはよく知られていて適切なコマンドおよび／またはプログラミング言語の使用を通じて適切な接続を確立するためにサーバ側の構成要素と通信することを要求される。しかしながら本発明によると、以前はクライアントによって成し遂げられるように要求された機能とタスクをローカルプロキシオブジェクトが実行する。この点で、ローカルプロキシオブジェクト106はクライアントの処理102に存在するいわゆるローカルプロキシオブジェクトであるが、しかし本発明によると、それはクライアントに対する主遠隔サーバ120の代理をする。同様に、ローカルプロキシオブジェクトはgetSomeInfo()、getAlarmInfo()、getSomeI 50

n f o 2 ()、および r e g i s t e r F o r A l a r m E v e n t () といったコマンドを発令することが可能である。ローカルプロキシオブジェクト内のソフトウェアはこの方式で遠隔サーバ120のそれを補完する。クライアントのコードがこのローカルプロキシオブジェクト106のインスタンスを生成すると、それは主遠隔サーバ120によって供給されるローカルプロパティ・ファイルに頼ってサーバの機械名／IPアドレス、ポート番号などといった主遠隔サーバ120の位置に関する情報を入手する。ローカルプロキシオブジェクト106はこの情報を使用してレジストリ・モジュール130を主遠隔サーバ120に無言で接触させ、かつ遠隔サーバの参照を入手する。主遠隔サーバ120を突きとめた後、ローカルプロキシオブジェクト106は主遠隔サーバ120との接続を無言で確立し、クライアントからメソッド呼び出しを受ける準備が整う。

10

【0027】

クライアントに単純な局所呼び出しを供給するために、ローカルプロキシオブジェクト106は主遠隔サーバ120によって導入されるすべての遠隔インターフェース122、124および126もやはり導入する。しかしこの導入は「ダミーの」導入であってここではローカルプロキシオブジェクト106は転送器としてはたらし、RMIプロトコルを使用してクライアントの呼び出しを主遠隔サーバ120へと転送する。それはバックエンド型サーバによって発される遠隔の異常を含むすべての応答を取り扱う。ローカルプロキシオブジェクト106は応答の中の受信情報をクライアント108へと転送し、遠隔の異常を変換してそれらを局所の異常としてクライアントへと転送する。

【0028】

20

この方式で、主遠隔サーバ120へのアクセスを試みるクライアントのアプリケーションは、RMIプロトコルを使用する主遠隔サーバ120との通信に必要とされる複雑なバックエンド活動に気付くことが全くなくなる。クライアントは、まさに処理部102の内部のいかなる他の局所オブジェクトとも同様に、単純にローカルプロキシオブジェクト106を使用する。クライアント108はRMI（もしくはCORBA）プロトコルを理解することを必要とせず、RMIレジストリ（もしくはCORBAネーミング（naming）モジュール）130に接触し、あるいは遠隔インターフェース122、124と126、および異常を取り扱う。実に、システムはRMIプロトコルからCORBAプロトコルへと変更されることが可能であり、その変更はクライアントに見えないものであろう。

【0029】

30

多数の遠隔サーバのインスタンスが存在して同じセットのサービスを提供するケースでは、ロードバランシング機構はローカルプロキシオブジェクト106の中で中央に導入され、そこでそれは利用可能なサーバ上のロード情報のトラックを保持し、使用されていないかまたはそれほど使用されていないサーバのインスタンスへと新たに呼び出しを送る。本発明の目的と合致するいかなるロードバランシングまたはトラフィック管理の技術も使用される可能性がある。多数サーバ・インスタンスの利用可能性に関する情報はプロパティ・ファイル112内に保持される。

【0030】

さらに、もしもクライアント108がサーバの処理側104上の何らかの特定の事象に関心を示し、そのような事象が生じるときに主遠隔サーバ120からの即時通知を必要とするならば、主遠隔サーバ120はクライアントがそのような事象を登録し、遠隔コールバックを受け取るように登録メカニズム（例えば r e g i s t e r F o r A l a r m E v e n t ）を供給する。ローカルプロキシオブジェクト106はこの事象および局所コールバックについて単純な登録メソッド（ r e g i s t e r F o r A l a r m E v e n t ）をクライアントに供給する。付け加えると、ローカルプロキシオブジェクト106は主遠隔サーバ120のトラックを保持しているので、それはサーバの故障を検出することが可能であり、主遠隔サーバ120が再スタートすると自動的にクライアントに代わって再登録を行うことが可能である。ローカルプロキシオブジェクト106はこの目的のためにクライアントのレジストリの局所リストを維持している。

40

【0031】

50

図3を参照すると、順序もしくは呼のフロー図が本発明による方法を具体的に示している。図示したように、主遠隔サーバ120が取り上げられ、(202で)初期化する。その後、主遠隔サーバ120はそれ自体をRMIレジストリ130に登録して(204で)そのサービスを公開する。

【0032】

ある程度の時間後に、クライアント108は主遠隔サーバ120のサービスを使用することを決定する。しかしRMIレジストリ130を突きとめ、遠隔サーバに接触するなどの困難を通して進むのではなく、クライアント108は単純にローカルプロキシオブジェクト106を使用する。同様に、クライアント108はローカルプロキシオブジェクト106を(206で)ローカルオブジェクトとしてインスタンス生成する(*i n s t a t i a t e*)。 10

【0033】

ここでローカルプロキシオブジェクト106は遠隔サーバのIPアドレス/機械名について(208で)プロパティ・ファイル112にアクセスする。もしも主遠隔サーバ120の多数のインスタンスが利用可能であれば、ローカルプロキシオブジェクト106はそれが全部と接触し、ロードバランシングに関するロード情報のトラックを保持するように全部のインスタンスのIPアドレスを入手する。

【0034】

次に、ローカルプロキシオブジェクト106はRMIレジストリ130に接触して(210で)主遠隔サーバ120の参照を判定する。ローカルプロキシオブジェクト106は(212で)主遠隔サーバ120に接触し、主遠隔サーバ120との直接接続を確立し、クライアント108からの要求のための準備が整う。 20

【0035】

クライアント108がサービス(例えば*g e t S o m e I n f o ()*)を要求する場合、それは(214で)適切なアドレス空間でローカルプロキシオブジェクト106上のローカルメソッドのように行う。ローカルプロキシオブジェクト106は(216で)RMIを使用して主遠隔サーバ120に遠隔呼び出しを行う。

【0036】

主遠隔サーバ120はクライアント108によって要求された情報を(218で)ローカルプロキシオブジェクト106へと返送する。ローカルプロキシオブジェクト106はその情報を(220で)クライアント108へと返送する。 30

【0037】

また別の状況では、クライアント108は将来の警報通知のために(222で)ローカルプロキシオブジェクト106に登録することを希望する可能性がある。そのときローカルプロキシオブジェクト106はクライアント108に代わって(224で)主遠隔サーバ120に登録する。ローカルプロキシオブジェクト106は(226で)クライアントの登録情報をセーブする。

【0038】

この処理では、ローカルプロキシオブジェクト106は(228で)定期的に「ピングング(*p i n g i n g*)」することによって主遠隔サーバ120のトラックを保持する。それはまた、もしも利用可能な多数のインスタンスが存在する場合、主遠隔サーバ120上のロード情報のトラックも保持することが可能である。もしも主遠隔サーバ120が破損して再スタートする場合、ローカルプロキシオブジェクト106は自動的かつ無言でクライアント108を主遠隔サーバ120に再登録する。 40

【0039】

本発明を導入すると、書き込むべき複雑なコードの量が削減される。さらに、RMIプロトコルに準拠した第三者のソフトウェアの一体化が極めて単純かつ簡単になる。その上さらに、本発明はローカルプロキシデベロッパに付随する検査時間もやはり削減する。本システムはまた、RMIサーバのアクセス・コード全部がここで集中化され、変更がクライアントに見えないものにされ得るので拡張可能かつ保守管理可能である。付け加える 50

と、サーバの多数のインスタンスのケースでは、ロードバランシングはローカルプロキシによって取り扱われることが可能である。

【0040】

以上の説明は単に本発明の特定の実施形態の開示を提供するものであり、それに限定する目的で意図されるものではない。同様に、本発明は上述の実施形態だけに限定されない。そうではなく、当業者が本発明の範囲内に入る別の選択肢となる実施形態を思いつく可能性があることが認識される。

【図面の簡単な説明】

【0041】

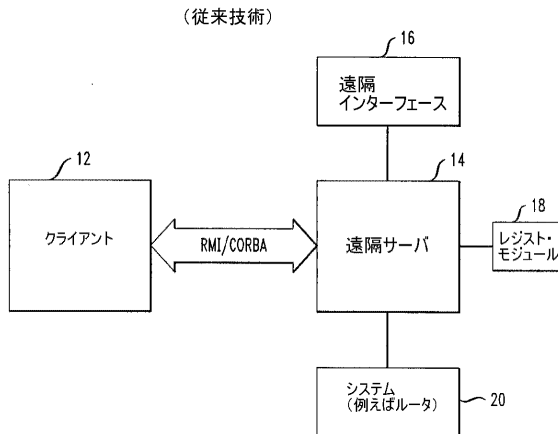
【図1】 従来のシステムの全体像を示す図である。

【図2】 本発明の実施形態を具体的に示す図である。

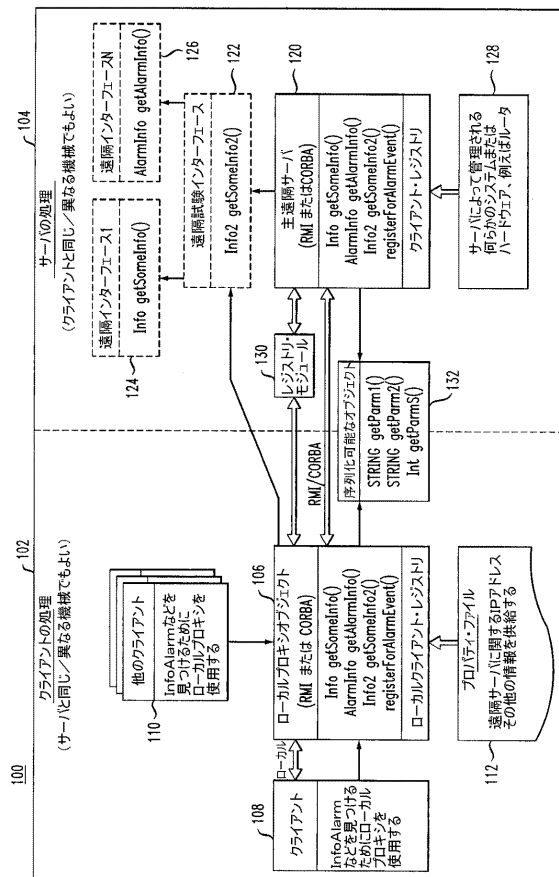
【図3】 本発明の動作の方法を例示する順序もしくは呼のフローを示す図である。

10

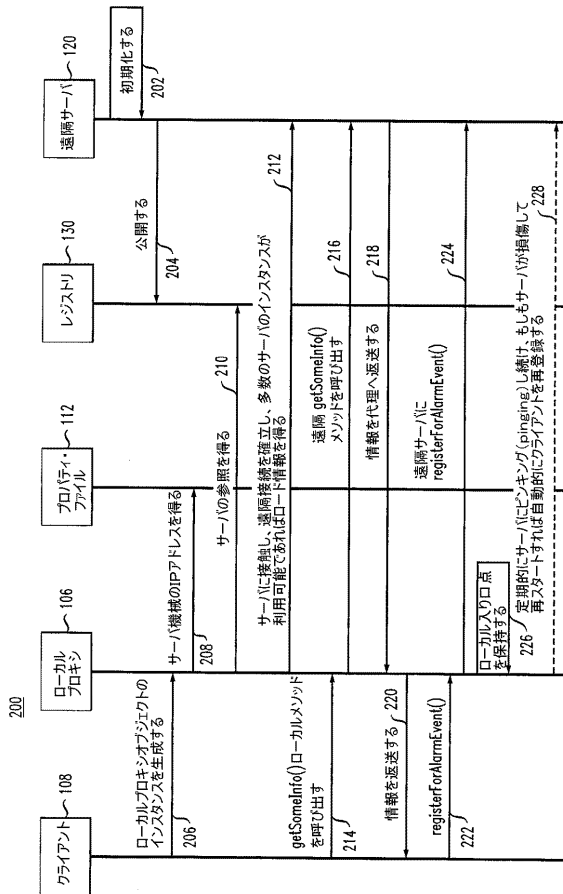
【図1】



【図2】



【図 3】



フロントページの続き

(74)代理人 100101498

弁理士 越智 隆夫

(74)代理人 100096688

弁理士 本宮 照久

(74)代理人 100104352

弁理士 朝日 伸光

(74)代理人 100128657

弁理士 三山 勝巳

(72)発明者 サミール デー. プラブー

アメリカ合衆国 6 0 5 6 3 イリノイス, ネイパーヴィル, エルダーベリー サークル 9 0 7
- 1 0 7

Fターム(参考) 5B045 BB28 BB49 GG01

5B082 HA05 HA08