

(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2008-3808

(P2008-3808A)

(43) 公開日 平成20年1月10日(2008.1.10)

(51) Int. Cl.

G06F 13/00 (2006.01)

F I

G06F 13/00 353B

テーマコード(参考)

5B089

審査請求 未請求 請求項の数 7 O L (全 21 頁)

(21) 出願番号 特願2006-171814 (P2006-171814)

(22) 出願日 平成18年6月21日(2006.6.21)

(特許庁注: 以下のものは登録商標)

1. フロッピー

(71) 出願人 504132272
国立大学法人京都大学
京都府京都市左京区吉田本町36番地1

(74) 代理人 110000338
特許業務法人原謙三国際特許事務所

(72) 発明者 荻野 博幸
京都府京都市左京区吉田本町 国立大学法
人京都大学大学院情報学研究所

Fターム(参考) 5B089 GB02 KB04

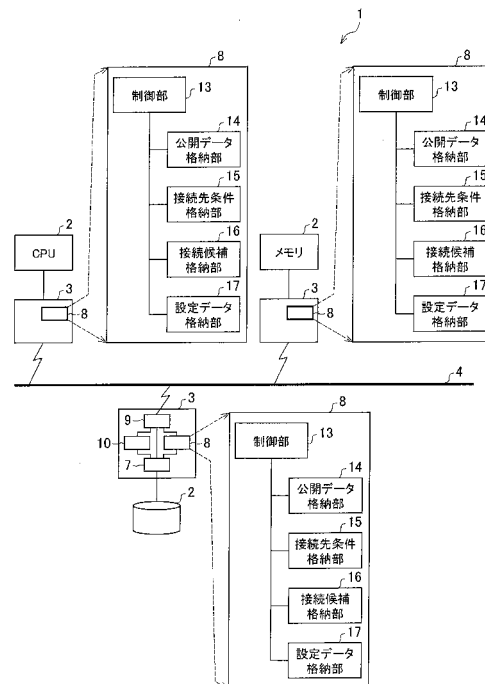
(54) 【発明の名称】 データ処理装置、計算機、及びデータ処理方法

(57) 【要約】

【課題】 計算機の構成要素である機能モジュールをより有効に利用する。

【解決手段】 本発明のデータ処理装置3は、機能モジュール接続制御部7を介して機能モジュール2と接続し、ネットワーク接続制御部9を介してネットワーク4に接続している。そして、自己紹介部8は、機能モジュール2の公開データをネットワーク4上に公開するので、計算機1を利用しようとするユーザは、ネットワーク4に接続されている全ての機能モジュール2の中から任意の機能モジュール2を選択することができる。そして、選択した各機能モジュール2の設定データ格納部17に、計算機1の設定データを書き込むことにより、選択した機能モジュールを計算機1として動作させることができるので、ネットワーク4に接続している機能モジュール2をより有効に利用することができる。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

機能要素とネットワークとに接続し、該機能要素とネットワークとの間でデータ処理を行うデータ処理装置であって、

データ処理装置と機能要素との間で行われるデータの送受信を制御する機能要素接続制御手段と、

データ処理装置と前記ネットワークとの間で行われるデータの送受信を制御するネットワーク接続制御手段と、

機能要素接続制御手段を介して接続している機能要素を特定するための公開データを前記ネットワーク上に公開すると共に、前記ネットワークを介して接続する接続先、及び該接続先と接続するときの接続設定条件を示す設定データを格納している自己紹介手段とを備えていることを特徴とするデータ処理装置。

10

【請求項 2】

複数の機能要素が、請求項 1 に記載のデータ処理装置とネットワークとを介して互いに接続されることによって構成されていることを特徴とする計算機。

【請求項 3】

上記データ処理装置は、前記計算機を構成する複数の機能要素間で行われるデータ送受信時に、データ送信側の機能要素から送信されるデータ形式を、データ受信側が処理可能なデータ形式に変換するデータ変換手段を備えていることを特徴とする請求項 2 に記載の計算機。

20

【請求項 4】

上記機能要素接続制御手段は、機能要素間で送受信されるデータを格納する送受信データ記憶手段を備えていること特徴とする請求項 2 または 3 に記載の計算機。

【請求項 5】

上記公開データを管理する組織構築手段を備え、

上記組織構築手段は、上記ネットワーク上に公開されている公開データを記憶する公開データ記憶手段と、

上記公開データ記憶手段に記憶されている公開データの中から、任意の機能要素を選択する選択手段とを備えていることを特徴とする請求項 2 乃至 4 の何れか 1 項に記載の計算機。

30

【請求項 6】

上記計算機を構成する複数の機能要素には、上記計算機における演算処理及び制御処理を行う処理装置、該処理装置の主記憶機能を担う主記憶装置、及び該主記憶装置の容量を補う補助記憶装置が含まれており、

上記処理装置と主記憶装置との間に仮想メモリ機構を構築するためのアドレス変換手段を有することを特徴とする請求項 2 乃至 5 の何れか 1 項に記載の計算機。

【請求項 7】

機能要素とネットワークとに接続し、該機能要素とネットワークとの間でデータ処理を行うデータ処理装置のデータ処理方法であって、

データ処理装置と機能要素との間で行われるデータの送受信を制御するステップと、

40

データ処理装置と前記ネットワークとの間で行われるデータの送受信を制御するステップと、

機能要素を特定するための公開データを前記ネットワーク上に公開するステップと、

前記ネットワークを介して接続する接続先、及び該接続先と接続するときの接続設定条件を示す設定データに基づいて接続先を決定するステップとを備えていることを特徴とするデータ処理方法。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明はデータ処理装置、計算機、及びデータ処理方法に関するものである。

50

【背景技術】

【0002】

従来の計算機は、計算機を構成する各機能要素がバスを介して結合され、各機能要素の固有の性能が最も有効に働くように構成されている。また、計算機をネットワークに接続して通信することによって、OS (Operating System) 上で計算機の資源を共有する技術が知られている。例えば、電子メール、FTP (File Transfer Protocol)、NFS (Network File System) 等を用いて計算機同士でデータの交換を行ったり、Webサーバ上に公開されているデータを閲覧・取得したりすることによって、ネットワークに接続されている複数の計算機の有する情報資源を共有することができる。

【0003】

さらに、計算機資源をより有効に活用するための計算機資源共有技術が生まれてきた。すなわち、グリッドコンピューティング、Peer to Peer (P2P)、NAS (Network Attached Storage) 等である。

【0004】

グリッドコンピューティングは、ネットワークに接続された複数の計算機を1つの仮想計算機として利用できるようにしたものである。すなわち、グリッドコンピューティングでは、複数の計算機のそれぞれにグリッドミドルウェアを搭載して仮想組織を構築し、該仮想組織のメンバーとして登録することにより、グリッドミドルウェアを登録している複数の計算機に計算処理を分散して実行させることができる。これにより、ネットワークに接続している遊休状態の計算機を利用して、大規模な計算処理を行うことができる。

【0005】

グリッドコンピューティングの例としては、非特許文献1に紹介されているように、メタコンピューティング、研究グリッド、アクセスグリッド、データグリッド、計算サービスクリッド、グリッドASP (Application Service Provider)、デスクトップグリッドコンピューティング、センサーグリッド等、多様な形態が知られている。

【0006】

P2Pは、基本的にサーバを介さず、計算機同士が直接にデータの交換を行うものである。P2Pを用いることによって、ネットワークに接続している計算機が持つデータを共有することができる。P2Pの例としては、非特許文献2に紹介されているように、クライアント・サーバP2P、スーパーノードハイブリッドP2P、ピュアP2P等が知られている。

【0007】

NASは、非特許文献3に示されているように、ネットワークにハードディスクを接続し、ネットワーク上でハードディスクを共有するものである。NASは、ハードディスク側にOSを搭載し、従来のサーバと同様にネットワーク上にデータを記憶させるものである。

【0008】

また、上記以外の計算機資源共有技術として、iSCSIと呼ばれるSCSI接続により、インターネット上でハードディスクを共有する技術がある(特許文献1参照)。

【0009】

さらに、他の計算機資源共有技術として、マルチプロセッサの環境でのHTTPキャッシュの管理をネットワーク上で行う方法がある(特許文献2参照)。その基礎となるメモリマネジメント技術については古くから知られていて、例えば、非特許文献4('Virtual Memory', pp.481-501) に詳しく述べられている。

【0010】

これらの技術を用いることによって、計算資源及び情報資源をネットワーク上で共有することができ、計算機資源を有効に活用することができる。

【非特許文献1】 巨理誠夫：グリッドの動向 - 次世代インターネット利用の中核技術になるか - 、文部科学省科学技術研究所科学技術動向、2002年9月

【非特許文献2】 R. Dingledine et.al. 'Peer-To-Peer: Harnessing the Power of Di

10

20

30

40

50

ruptive Technologies', O'reilly & Associates, March 15. 2001.

【非特許文献3】W. C. Perston: 'SANs & NAS', O'reilly & Associates, October. 2002.

【非特許文献4】D. A. Patterson, et. al 'Computer Organization & Design The Hardware / Software Interface' Morgan Kaufmann Publishers Inc., 1994.

【特許文献1】特表2005-502225(平成17年1月20日公表)

【特許文献2】米国特許第7,017,025号明細書

【発明の開示】

【発明が解決しようとする課題】

【0011】

しかしながら、上記従来技術では、ネットワーク全体として、計算機資源が完全に有効には活用されているとは言い難い。すなわち、上記従来技術では、ネットワークに接続した機器のうち、CPUを有するパソコンなどの機器は他のコンピュータからその存在を認識することができるが、そのパソコンに含まれるメモリやハードディスク、ディスプレイ等の機能要素は、他のコンピュータから、その存在を認識することができない。そのため、あるコンピュータが所有する資源の全てを他のコンピュータが利用することは困難であり、システム全体として資源が有効に使用されていない。

【0012】

本発明は、上記の問題点を鑑みてなされたものであって、その目的は、機能要素をより有効に利用することができるデータ処理装置、計算機、及びデータ処理方法を提供することにある。

【課題を解決するための手段】

【0013】

本発明のデータ処理装置は、上記の課題を解決するために、機能要素とネットワークとに接続し、該機能要素とネットワークとの間でデータ処理を行うデータ処理装置であって、データ処理装置と機能要素との間で行われるデータの送受信を制御する機能要素接続制御手段と、データ処理装置と前記ネットワークとの間で行われるデータの送受信を制御するネットワーク接続制御手段と、機能要素接続制御手段を介して接続している機能要素を特定するための公開データを前記ネットワーク上に公開すると共に、前記ネットワークを介して接続する接続先、及び該接続先と接続するときの接続設定条件を示す設定データを格納している自己紹介手段とを備えていることを特徴としている。

【0014】

また、本発明のデータ処理方法は、上記の課題を解決するために、機能要素とネットワークとに接続し、該機能要素とネットワークとの間でデータ処理を行うデータ処理装置のデータ処理方法であって、データ処理装置と機能要素との間で行われるデータの送受信を制御するステップと、データ処理装置と前記ネットワークとの間で行われるデータの送受信を制御するステップと、機能要素を特定するための公開データを前記ネットワーク上に公開するステップと、前記ネットワークを介して接続する接続先、及び該接続先と接続するときの接続設定条件を示す設定データに基づいて接続先を決定するステップとを備えていることを特徴としている。

【0015】

上記の構成によると、データ処理装置は、機能要素接続制御手段、及びネットワーク接続制御手段を備えているので、機能要素は、データ処理装置を介して、ネットワークに接続している他の機能要素とデータの送受信を行うことができる。

【0016】

なお、機能要素とは、計算機を構成する要素であり、機能的なまとまりを持つ要素を指す。例えば計算機を構成する機能モジュールは、機能要素に含まれる。

【0017】

また、自己紹介手段は、公開データをネットワーク上に公開するので、ネットワークにアクセスすれば、ネットワークに接続する全ての機能要素の公開データを閲覧することが

10

20

30

40

50

できる。したがって、前記ネットワークには、どれだけの機能要素、すなわち計算機資源が、ネットワークのどこに接続されているかを把握することができる。これにより、計算機資源を最適に配分し、有効に利用することができる。

【0018】

さらに、自己紹介手段は、前記ネットワークを介して接続する接続先、及び該接続先と接続するときの接続設定を示す設定データを格納しているため、データ処理装置は、この設定データに定義されている接続先に、設定データに定義されている接続設定条件で接続することができる。なお、接続設定条件は、通信速度や記憶容量等、接続先に要求する接続条件を指す。例えば、設定データにおける接続先をCPU、メモリ、及びハードディスク(HDD)の機能要素とし、接続設定条件としてメモリやHDDの使用容量等を定義した設定データを作成する。そして、この設定データをCPU、メモリ、及びHDDの機能要素がそれぞれ備えているデータ処理手段の設定データ格納手段に格納することにより、CPU、メモリ、及びHDDの機能要素を1つの計算機として機能させることが可能となる。

10

【0019】

また、本発明の計算機は、上記の課題を解決するために、複数の機能要素が、請求項1に記載のデータ処理装置とネットワークとを介して互いに接続されることによって構成されていることを特徴としている。

【0020】

上記の構成によると、上記計算機を利用しようとするユーザは、ネットワークに接続している機能要素について、各機能要素を特定する公開データを閲覧することができる。そして、公開データに基づいて、ネットワークに接続する機能要素の中から必要な機能要素を選択して計算機を構成することができる。したがって、ネットワーク接続と同様に、接続先となる機能要素を容易に追加、変更、または削除することができ、計算機を容易に再構築することができる。そして、このようにして構成された計算機は、ネットワークをあたかも自らの構成要素として動作する。

20

【0021】

従って、従来はOSの上でしか共有することができなかったCPU、メモリ等の機能要素を直接にネットワーク上で共有することができる。これにより、ネットワークに接続する機能要素を複数のユーザで最適に配分して利用することが可能となる。

30

【0022】

また、上記計算機は、複数の機能要素から、必要な機能要素を選択して使用することができるので、選択した機能要素に故障等のトラブルが発生した場合、容易に他の機能要素で代用することができる。また、予め同等の処理を行うことのできる複数の機能要素を計算機の構成要素として選択しておくこともできる。従って、上記計算機は、従来の計算機と比べて、故障に強く信頼性が高い計算機であるといえる。

【0023】

さらに、上記計算機は、機能要素をネットワークを介して接続して構成されているので、例えばネットワークとしてインターネットを用いたような場合には、世界中の機能要素を用いて計算機を構成することができる。従って、従来は、物理的、或いは経済的に困難であった構成の計算機を実現することができる。

40

【0024】

また、本発明の計算機は、上記の構成に加えて、上記データ処理装置は、前記計算機を構成する複数の機能要素間で行われるデータ送受信時に、データ送信側の機能要素から送信されるデータ形式を、データ受信側が処理可能なデータ形式に変換するデータ変換手段を備えていることが好ましい。

【0025】

計算機を動作させるときには、その計算機を構成する機能要素間でデータの送受信をしながら動作することになる。ここで、機能要素によっては、取り扱い可能なデータ形式が異なる等の理由でデータの送受信ができない場合がある。このような場合に、データ処理

50

装置がデータ変換手段を備えていれば、データ変換手段が機能要素間でデータ形式を変換するので、データの送受信が可能となる。なお、機能要素が送受信するデータには、計算機 1 を動作させるための制御データ、及びファイル等のデータを含む。

【0026】

また、本発明の計算機は、上記の構成に加えて、上記機能要素接続制御手段は、機能要素間で送受信されるデータを格納する送受信データ記憶手段を備えていることが好ましい。

【0027】

上述のように、本発明の計算機では、データ処理装置を介して複数の機能要素間でデータの送受信をしながら動作する。ここで、上記機能要素接続制御手段が送受信データ記憶手段を備えている場合、送受信データ記憶手段をキャッシュ・メモリのよう動作させることができる。例えば、送受信されるデータを送受信データ記憶手段に一時的に格納することによって、機能要素と、データ処理装置との間、またはデータ処理装置と他のデータ処理装置との間で行われるデータ送受信の回数や通信データ量を減らすことができ、データの送受信をよりスピーディに行うことができる。また、データ送受信の際にデータが損傷したような場合に、送受信データ記憶手段に記憶されているデータを用いてデータの修復を行うこともできる。

【0028】

また、本発明の計算機は、上記の構成に加えて、組織構築手段を備えることができ、その組織構築手段は、上記ネットワーク上に公開されている公開データを記憶する公開データ記憶手段と、公開データ記憶手段に記憶されている公開データの中から、任意の機能要素を選択する選択手段とを備えていることが好ましい。この場合、複数の機能要素を接続して計算機を構成するときに、組織構築手段は、ネットワークに接続する各機能要素の公開データを記憶しており、上記計算機のユーザは、組織構築手段の選択手段によって、所望する機能要素を速やかに選び出し、計算機を構成することができる。

【0029】

また、本発明の計算機は、上記の構成に加えて、上記計算機を構成する複数の機能要素は、上記計算機における演算処理及び制御処理を行う処理装置、該処理装置の主記憶機能を担う主記憶装置、及び該主記憶装置の容量を補う補助記憶装置を含み、上記処理装置と主記憶装置との間に仮想メモリ機構を構築するためのアドレス変換手段を有することが好ましい。

【0030】

上記の構成によると、処理装置、主記憶装置、及び補助記憶装置を用いて仮想メモリ機構を構築することができ、より少ない容量の主記憶装置で効率良く計算機を動作させることができる。

【発明の効果】

【0031】

本発明のデータ処理装置は、上述のように、機能要素とネットワークとに接続し、該機能要素とネットワークとの間でデータ処理を行うデータ処理装置であって、データ処理装置と機能要素との間で行われるデータの送受信を制御する機能要素接続制御手段と、データ処理装置と前記ネットワークとの間で行われるデータの送受信を制御するネットワーク接続制御手段と、機能要素接続制御手段を介して接続している機能要素を特定するための公開データを前記ネットワーク上に公開すると共に、前記ネットワークを介して接続する接続先、及び該接続先と接続するときの接続設定条件を示す設定データを格納している自己紹介手段とを備えている。従って、ネットワークに接続する機能要素を最適に配分し、有効に活用することができるという効果を奏する。

【発明を実施するための最良の形態】

【0032】

本発明の一実施形態について図 1 ないし図 11 に基づいて説明すれば、以下の通りである。

10

20

30

40

50

【0033】

図2は、従来の計算機100の概略構成を示す図である。図示のように、従来の計算機100は、計算機の構成要素である複数の機能モジュールが、バスと呼ばれる伝送路にハードウェア・ロジックのインターフェイスを介して結合することによって構成されている。

【0034】

なお、上記機能モジュールは、主演算処理や計算機全体の制御処理を行うCPUのような処理装置や、処理装置の主記憶機能を担うメモリのような主記憶装置や、主記憶装置の記憶容量を補ったり、データを保存したりするハードディスクドライブ(HDD)、フロッピーディスクドライブ、光ディスクドライブ等の補助記憶装置や、キーボードやマウス等の入力装置、ディスプレイやプリンタ等の出力装置等を指す。

10

【0035】

図3は、本発明の一実施形態の計算機1の概略構成を示す図である。図示のように、本実施の形態の計算機1は、複数の機能モジュール(機能要素)2を、データ処理装置3及びネットワーク4を介して接続することによって構成されている。すなわち、計算機1では、図2に示した計算機100のバスに相当する部分をネットワーク4が担っている。本実施の形態の計算機1は、データ処理装置3の働きにより、ネットワーク4に接続された機能モジュール2を任意に組み合わせる構成することができる。なお、計算機1を構成する機能モジュール2は、図3に示すものに限られない。計算機1は、図3に示す機能モジュール2を全て備えている必要は無いし、図3に示していない機能モジュール2で計算機1を構成することもできる。また、データ処理装置3の詳細については後述する。

20

【0036】

機能モジュール2は、従来の計算機100の機能モジュールと同様のものを使用することができる。また、図4に示すように、組織構築手段6を計算機1の構成に加えることもできる。

【0037】

組織構築手段6は、図示していない公開データ記憶部(公開データ記憶手段)と、機能モジュール選択処理部(選択手段)とを備えている。公開データ記憶部は、ネットワーク4に接続している機能モジュール2を特定するための公開データを記憶している。機能モジュール選択処理部は、公開データ記憶部が記憶している公開データの中から、計算機1を利用しようとするユーザが希望する機能モジュール2の選択を受け付ける。そして、選択に基づいて設定データを作成し、作成した設定データを、選択された機能モジュール2に接続しているデータ処理装置3の設定データ格納部に書き込む。例えば、計算機1の構成要素として、CPU1とメモリ1とHDD1とが選択された場合、計算機1がCPU1とメモリ1とHDD1とで構成される旨を示す設定データを作成し、CPU1、メモリ1、及びHDD1の機能モジュール2に接続しているデータ処理装置3の設定データ格納部に書き込む。なお、公開データ、設定データ、及び設定データ格納部については後述する。

30

【0038】

組織構築手段6を有する場合、計算機1を利用しようとするユーザは、組織構築手段6が記憶する公開データを参照し、機能モジュール選択処理部によって所望の機能モジュール2を選択して計算機1を構成することができる。また、組織構築手段6に、計算機1を構成する機能モジュール2の間で行われるデータの送受信を中継するデータ中継機能を持たせることもできる。データ中継機能を有する場合、機能モジュール2の間で行われるデータの送受信を最適化することができる。

40

【0039】

なお、図4では、機能モジュール2の一つとして組織構築手段6を導入した例を示したが、組織構築手段6の導入方法はこれに限られない。すなわち、組織構築手段6を必要な各所に設ける方法や、階層化する方法、一部の機能モジュール2が組織構築手段6を兼ねるようにする方法等、様々な方法が可能である。

50

【0040】

ネットワーク4としては、例えば、LAN、Bluetooth（登録商標）、イーサネット（登録商標）、USB、IEEE1394、IEEE-488、RS-232C等、有線、無線に関わらず様々なネットワークを使用することができる。なお、本実施の形態では、ネットワーク4がインターネットである場合を想定している。

【0041】

また、図3に参照番号5で示しているのは、Webブラウザ上の表示画面の一例である表示画面5である。本実施の形態の計算機1は、ネットワーク4に接続する複数の機能モジュール2が協調して一台の計算機として動作する。例えば、図中に示すマウスを移動させる操作を行った場合、マウスから出力される操作データが該マウスのデータ処理装置3及びネットワーク4を介してCPUの機能モジュール2に伝達され、CPUの機能モジュール2は、操作データの受信を受けてメモリやディスプレイの機能モジュール2に命令を送る。これにより、表示画面5のカーソルが移動することになる。

10

【0042】

次に、データ処理装置3の構成について、図5に基づいて説明する。図5は、データ処理装置3の概略構成を示す図である。図示のように、データ処理装置3は、機能モジュール接続制御部（機能要素接続制御手段）7と、自己紹介部（自己紹介手段）8と、ネットワーク接続制御部（ネットワーク接続制御手段）9と、データ変換部（データ変換手段）10とを備えている。これらの機能について、図5に基づいて説明する。なお、図5では、機能モジュール接続制御部7と、ネットワーク接続制御部9を別個の部材として記載しているが、これらを一体に構成しても良い。

20

【0043】

機能モジュール接続制御部7は、機能モジュール2とデータ処理装置3との接続を制御する。なお、図示していないが、機能モジュール接続制御部7に、送受信データ記憶部（送受信データ記憶手段）を設け、機能モジュール接続制御部7を介して送受信されるデータのうち、頻繁に使用されるデータを送受信データ記憶部に一時的に格納しても良い。この場合、機能モジュール接続制御部7と機能モジュール2との間や、機能モジュール接続制御部7と、他の機能モジュール2と接続しているデータ処理装置3との間で行われるデータ送受信の回数や通信データの量を減らすことができ、データの送受信をよりスピーディに行うことができる。また、データ送受信の際にデータが損傷したような場合に、送受信データ記憶部に記憶されているデータを用いてデータの修復を行うこともできる。さらに、機能モジュール接続制御部7に、データの圧縮、解凍を行う機能を付加した場合、よりスムーズにデータの送受信を行うことができる。また、機能モジュール2と他の機能モジュール2を接続するときに、機能モジュール接続制御部7に各機能モジュール2のインターフェイスのハードウェアに代わって働く機能をさせることもできる。

30

【0044】

自己紹介部8は、ネットワーク4を介して機能モジュール2の公開データを送信する。また、自己紹介部8は、ネットワーク4を介して接続する接続先のネットワークID、及び上記接続先と接続するときの接続設定条件を示す設定データを格納している。ここで、上記接続先とは、ネットワーク4に接続している他の機能モジュール2や、ネットワーク4に接続している従来の計算機等を指し、接続設定条件とは、通信速度や記憶容量等、接続先に要求する条件を指している。自己紹介部8については、後に詳しく説明する。

40

【0045】

ネットワーク接続制御部9は、データ処理装置3とネットワーク4との間で行われるデータの送受信を制御する。具体的には、ネットワーク4を介して他の機能モジュール2や、ネットワーク4に接続している他の計算機との間で行われるデータの送受信を制御する。

【0046】

また、ネットワーク接続制御部9は、計算機1の動作時に機能モジュール2の間で行われるデータ送受信を制御する。具体的には、ネットワーク接続制御部9は、通信プロトコ

50

ルとしてTCP/IPを備え、ネットワーク4を介して接続する他の機能モジュール2との間で行われるデータ送受信の制御を行う。ここで、機能モジュール2の間で処理可能なデータ形式が異なる場合には、データ変換部10を介してデータの送受信を行う。なお、ネットワーク接続制御部9が送受信するデータには、計算機1を動作させるための制御データ、及びファイル等のデータを含む。

【0047】

データ変換部10は、機能モジュール2の間におけるデータの送受信時にデータ形式の変換を行う。例えば、計算機1を構成するCPUの機能モジュール2とメモリの機能モジュール2とで処理可能なデータ形式が異なる場合等も考えられる。このような場合に、CPUからメモリへとデータが送信されるとき、データ変換部10は、CPUからのデータをメモリが処理可能なデータ形式に変換する。なお、この場合、CPU側、またはメモリ側の何れか一方のデータ変換部10が変換を行えばよい。すなわち、データ変換部10は、全ての機能モジュール2に組み込む必要は無い。

10

【0048】

なお、データ変換部10を備えていない場合でも、本実施の形態の計算機1を動作させることは可能である。しかしながら、データ変換部10を備えている場合、処理可能なデータ形式が異なる機能モジュール2の間でのデータの送受信を可能にすることができるので、データ処理装置3は、データ変換部10を備えていることが好ましい。

【0049】

本実施の形態の計算機1は、上述のように、データ処理装置3を介して機能モジュール2を接続することから、OS等の基本ソフトウェアを従来よりも簡略化することができる。計算機1で使用されるソフトウェアについて、図6に基づいて説明する。

20

【0050】

図6は、従来の計算機で使用されているソフトウェアを示す図である。従来の計算機で使用されているソフトウェアは、図示のように、アプリケーション・ソフトウェア、ミドルウェア、OS、基本ネットワーク機能、及び通信機能を備えている。なお、基本ネットワーク機能及び通信機能は、OSの機能である。

【0051】

これに対し、本実施の形態の計算機1では、基本ネットワーク機能及び通信機能は、各機能モジュール2のデータ処理装置3が担うので、OSを簡略化することができる。具体的には、ネットワーク接続制御手段9が従来のソフトウェアにおける基本ネットワーク機能の役割を果たし、データ変換部10が通信機能の役割を果たす。すなわち、図6に示す基本ネットワーク機能及び通信機能の部分だけOSを簡略化することができる。

30

【0052】

なお、計算機1は、ミドルウェアを搭載することもできるので、上記〔背景技術〕で紹介したグリッドや、P2P等のミドルウェアによる資源の共有化機能を計算機1上で動作させることもできる。

【0053】

図7は、本実施の形態の計算機1におけるデータ伝達の一例を示している。同図に小円で示すノード12は、機能モジュール2である。図7の左下、及び右下の $n \times n$ の平面は、それぞれノード群を表しており、各ノード群は、 n^2 個のノード12を含む。また、楕円で示すスーパーノード11は、従来の計算機である。スーパーノード11は、図7の上部中央に記載されている最上位のスーパーノード11から、図示していない最下位のスーパーノード11まで、計 n 個存在する。スーパーノード11は、上記ノード群とは別の階層に位置し、各スーパーノード11もそれぞれ別の階層に位置する。なお、本実施の形態では、ネットワーク4がインターネットであることを想定しているので、ノード群とスーパーノード11とは、IPアドレスによってその階層が区別される。なお、スーパーノード11として、本発明の計算機を用いることもできる。

40

【0054】

まず、ノード群におけるデータ伝達の方法について説明する。本実施の形態の計算機1

50

では、機能モジュール2の間、すなわちノード12の間でデータが伝達される際に、該データの伝達回数を示す濃度レベルを、該データに添付して伝達する。なお、濃度レベルは、データの伝達回数の増加に従って濃度が薄くなる（レベル値が大きくなる）ように更新される。例えば、濃度レベルがレベル0のノード12から、他のノード12にデータを送ると、該データを受け取ったノード12では、そのデータの濃度レベルをレベル1に更新して、また他のノード12へとデータを送る。そしてそのデータを受け取ったノード12では、濃度レベルをレベル2に更新する。データを他のノード12に送った後も、該データはノード12にキャッシュとして記憶されているので、データの伝達中にデータが損傷を受けたような場合には、損傷を受けたデータに添付されている濃度レベルを参照し、より小さいレベル値のデータを用いてデータの修復をすることができる。

10

【0055】

ここで、ノード群の中の、ある1つのノード12に対してデータを伝達する場合を考える。この、データを伝達する対象のノード12を目標ノードと称する。目標ノードにデータを伝達するためには、 n^2 個のノード12の1つ1つにデータを伝達して行くという試行を目標ノードに到達するまで繰り返す必要がある。すなわち、目標ノードに到達するまでには、最大で n^2 回の試行が必要となる。このように、ノード群内でデータ伝達を行う場合、目標ノードに到達するまでに多くの試行が必要になる。なお、これは同時に、多くのノード12を経由してデータ伝達が行われることを意味するので、多くのノード12にデータがキャッシュとして残ることになり、これをキャッシュ・メモリとして利用することにより、伝送速度を向上することができる。

20

【0056】

次に、スーパーノード11を用いたデータ伝達の方法について説明する。 n 個のスーパーノード11で、ノード群に含まれる n^2 個のノード12についての全てのデータを記憶している。ノード群内のデータは、 n 個のスーパーノード11の少なくとも1つに記憶されているので、最大でも n 回の試行で目標ノードに到達できる。すなわち、スーパーノード11を用いてデータ伝達を行う場合、より短時間で目標ノードに到達することができる。

【0057】

また、図7に示すように、複数のノード群をスーパーノード11で接続した場合、ノード群内のデータが複数の階層のスーパーノード11やノード12に分散されたキャッシュとして存在することになるので、データの損傷に強い信頼性の高いシステムとなる。

30

【0058】

なお、データ伝達の形態は、図7に示すようなスーパーノード11を用いる方法に限られず、他の多様な形態が可能である。

【0059】

〔本実施形態の計算機の構成方法〕

本実施の形態の計算機1は、従来の計算機とは異なり、独立した機能モジュール2を任意に組み合わせて構成されるので、ネットワーク4に接続している機能モジュール2を複数の計算機で共有し、有効に利用することが可能となる。機能モジュール2は、主に自己紹介部8の働きによって計算機1として構成される。計算機1を構成する方法について、

40

【0060】

図8は、計算機1を構成するための初期設定の方法を示している。上述のように、機能モジュール2はデータ処理装置3を介してネットワーク4に接続しているが、この状態では計算機1として動作しない。そこで、ネットワーク4に従来の計算機を接続し、計算機1の初期設定を行う。

【0061】

まず、計算機1を利用しようとするユーザは、従来の計算機でネットワーク4にアクセスし、自己紹介部8が公開する公開データを参照し、計算機1を構成する機能モジュール2を選択する。なお、上記公開データは、後述する公開データ格納部に格納されている。

50

選択する機能モジュール 2 は、ネットワーク 4 に接続されている機能モジュール 2 であればどのようなものを選択しても良いが、ここでは、CPU、メモリ、HDD の機能モジュール 2 を選択するものとする。

【0062】

続いて、選択したそれぞれの機能モジュール 2 の自己紹介部 8 に設定データを書き込む。ここでは、設定データの種別の項目に、選択した各機能モジュール 2、すなわち CPU、メモリ、及び HDD が書き込まれ、接続設定条件の項目に、メモリ及び HDD の使用記憶容量や通信速度等が書き込まれ、ID の項目に、CPU、メモリ、及び HDD のネットワーク ID が書き込まれる（図 9 参照）。これにより、CPU、メモリ、HDD の機能モジュール 2 が 1 つの計算機 1 として動作するための関係が構築される。

10

【0063】

上記構成では、CPU、メモリ、及び HDD は互いに接続されている必要があるので、設定データには、CPU、メモリ、及び HDD の全てが書き込まれる必要がある。しかしながら、設定データには、機能モジュール 2 の接続先のみが書き込まれていても良い。例えば、上記計算機 1 の構成要素としてキーボードの機能モジュール 2 を追加する場合、キーボードの機能モジュール 2 は、CPU のみと接続していれば良く、メモリや HDD の機能モジュール 2 とは接続する必要はない。この場合、キーボードの機能モジュール 2 の設定データには、上記 CPU のネットワーク ID、及び上記 CPU との接続設定条件のみが書き込まれていれば良い。設定データに機能モジュール 2 の接続先のみが書き込まれている場合、データ処理装置 3 の格納するデータ量を減らすことができ、データ処理装置を簡略化することができる。

20

【0064】

次に、計算機 1 に OS をインストールする手順を説明する。上述のようにして計算機 1 を構成する全ての機能モジュール 2 について設定データを書き込んだ後、CPU の機能モジュール 2 の ROM (Read Only Memory) に BIOS (Basic Input Output System) を書き込む。BIOS には、CPU が起動した際に最初に行う命令が含まれている。BIOS の書き込み終了後、計算機 1 を構成する全ての機能モジュール 2 の初期化を行う。初期化は、計算機 1 を構成する全ての機能モジュール 2 の電源を切り、再度電源を入れる、または計算機 1 を構成する全ての機能モジュール 2 をリセットする等で行う。

【0065】

初期化を行うと、CPU の機能モジュール 2 で、上記 BIOS の作用により、HDD、の機能モジュール 2 に格納されている OS のインストールが行われる。なお、OS は、ネットワーク 4 を介して計算機 1 を構成していない他の機能モジュール 2 や、ネットワーク 4 に接続している他の記録装置や記録媒体等から供給されるようにしても良い。OS のインストール後、CPU の機能モジュール 2 の ROM を、インストールした OS が起動するように書き換える。以上の手順により、計算機 1 には OS がインストールされる。

30

【0066】

OS のインストールが完了した後、計算機 1 を再起動すると、起動時に OS が動作するようになり、以降は従来の計算機と全く同様に使用することができる。すなわち、計算機 1 では、図 2 に示したような従来の計算機におけるバスの機能を、データ処理手段 3 及びネットワーク 4 が代替することにより、ネットワーク 4 上に分散した機能モジュール 2 を 1 つの計算機 1 として動作させることができる。

40

【0067】

なお、補足として、機能モジュール 2 及び計算機 1 で行われるデータ送受信を、OS I 参照モデル用いて説明すると以下ようになる。OS I 参照モデルとは、計算機等で使用される通信プロトコルを物理層 (Layer1)、データリンク層 (Layer2)、ネットワーク層 (Layer3)、トランスポート層 (Layer4)、セッション層 (Layer5)、プレゼンテーション層 (Layer6)、アプリケーション層 (Layer7) の 7 層に分け、各階層について機能と役割を定めたものである。OS I 参照モデルに準拠した通信プロトコルを使用することにより、異なる種類の機器の接続が容易になる。本実施の形態の計算機 1 においても、OS I

50

参照モデルに準拠した通信プロトコルを使用しているが、これに限られるものではない。

【0068】

ネットワーク4に接続している機能モジュール2のうち、計算機1を構成していないものは、上記公開データをネットワーク4に公開し、上記設定データの書き込みを受け付ける。このとき、上記機能モジュールのデータ処理装置3で使用する階層は、アプリケーション層(Layer7)である。また、計算機1として構成された機能モジュール2のデータ処理装置3は、ネットワーク層(Layer3)及びトランスポート層(Layer4)で通信し、機能モジュール2間でのデータの送受信を中継する。

【0069】

次に、計算機1の構成方法について、図1に基づいてより詳しく説明する。図1は、図8に示した計算機1の自己紹介部8をより詳細に示す図である。図示のように、CPU、メモリ、及びHDDの機能モジュール2は、それぞれデータ処理装置3を備えており、データ処理装置3には、機能モジュール接続制御部7、自己紹介部8、ネットワーク接続制御部9、及びデータ変換部10を備えている。なお、図1では、HDDの機能モジュール2に接続したデータ処理装置3のみ、これらの構成要素を図示しているが、他の機能モジュール2についても同様の構成要素を備えている。

10

【0070】

自己紹介部8は、図1に示すように、制御部13、公開データ格納部14、接続先条件格納部15、接続候補格納部16、及び設定データ格納部17を備えている。

【0071】

制御部13は、自己紹介部8全体の機能を統括的に制御する。制御部13は、図示していないが、CPU及び記憶部を備えており、記憶部に記憶されたプログラムをCPUが実行することによって機能する。

20

【0072】

公開データ格納部14は、ネットワーク4上に公開する公開データを格納する。具体的には、図9に示すように、機能モジュール2の種別、仕様、及びネットワークID等、機能モジュール2を特定するためのデータを格納する。なお、図9では、一例として、CPUの機能モジュール2の自己紹介部8を示しているので、公開データの種別の項目はCPUとなっている。制御部13は、公開データ格納部14に格納されているこれらの公開データを、ネットワーク接続制御部9を介してネットワーク4上に公開する。

30

【0073】

設定データ格納部17は、設定データとして、ネットワーク4を介して接続する接続先の種別、ID、及び該接続先と接続するときの接続設定条件を格納している。図9では、種別の項目には、メモリやHDD等が、IDの項目には、種別の項目に格納されている各機能モジュール2に対応するネットワークIDが格納されている。そして、接続設定条件の項目は、メモリについては実メモリ使用容量及び仮想メモリ使用容量等が、HDDについては使用領域、仮想メモリ記憶容量、物理記憶領域等が格納されている。なお、図9には示していないが、接続設定条件には通信速度等も含まれている。なお、1つの種別について複数のIDを対応付けることもできる。例えば、計算機1で複数のHDDを使用した場合には、複数のHDDのネットワークIDをIDの項目に格納することができる。

40

【0074】

自己紹介部8は、公開データ格納部14と設定データ格納部17とを備えているので、計算機1を利用しようとするユーザは、計算機1の初期設定を行うことができ、複数の機能モジュール2を1つの計算機1として動作させることができる。

【0075】

本実施の形態の計算機1は、仮想メモリ機構を構築してデータの送受信を行う機能をさらに備えている。接続先条件格納部15、及び接続候補格納部16は、仮想メモリ機構を構築するための構成である。なお、仮想メモリ機構については後に詳しく説明する。

【0076】

接続先条件格納部15は、接続先の条件を格納する。ここでは、機能モジュール2がC

50

P Uであるから、図9に示すように、接続先のメモリに対して要求する条件、例えば、読み書き速度や揮発/不揮発性等の条件を格納している。制御部13が、ネットワーク4上に公開されている機能モジュール2の公開データを取得し、その中から上記接続先条件を満たすものを選択して接続候補格納部16に格納する。

【0077】

なお、接続先候補となる機能モジュール2を絞り込む必要がない場合は、計算機1のユーザが、接続候補格納部16に格納する接続候補を任意に選択すればよい。この場合、自己紹介部8は、接続先条件格納部15を備えていなくてもよい。但し、ネットワーク4に接続する機能モジュール2の数が多数になるような場合、接続先条件格納部15を備えていると、必要な機能モジュール2の選択が容易になるので、自己紹介部8は、接続先条件格納部15を備えていることが好ましい。

10

【0078】

また、計算機1は、設定データを変更することにより、計算機1を構成する機能モジュール2間の接続関係、例えば通信速度や記憶容量等を変更したり、計算機1を構成する機能モジュール2を変更、追加、または削除したりすることによって計算機1を再構築することができる。

【0079】

例えば、計算機1のユーザが、各機能モジュール2の設定データを直接変更することによって計算機1を再構築することができるし、組織構築手段6によって各機能モジュール2の設定データを更新することによっても計算機1を再構築することができる。

20

【0080】

ここでは、計算機1を再構築する例として、設定データにおける接続設定条件(図9参照)が変更された場合に、制御部13が計算機1の再構築を行う例を説明する。

【0081】

まず、計算機1のユーザは、計算機1を構成する任意の機能モジュール2に接続しているデータ処理装置3の設定データ格納部17における接続設定条件を所望の条件に変更する。接続設定条件が変更されたデータ処理装置3の制御部13は、設定データ格納部17から、変更された接続設定条件を読み出す。ここで、制御部13は、変更された接続設定条件が、接続先の変更を伴うか否かを判断する。接続先の変更を伴う場合、制御部13は、接続候補格納部16から接続先となる機能モジュール2を選択し、選択した機能モジュール2のネットワークIDを設定データに反映させる。そして、制御部13は、該設定データを、計算機1を構成する各機能モジュール2に送り、各機能モジュール2の設定データを更新する。

30

【0082】

一方、接続先の変更を伴わない場合、制御部13は、計算機1を構成する各機能モジュール2に接続設定条件が変更された設定データを送り、各機能モジュール2の設定データを更新する。

【0083】

このように、制御部13が各機能モジュール2の設定データを更新する場合、計算機1のユーザは、所望の接続設定条件を書き込むだけで計算機1を再構築することができる。すなわち、計算機1のユーザは、計算機1を構成する各機能モジュール2のデータ処理装置3の1つ1つについて設定データを変更する必要が無く、計算機1を容易に再構築することができる。また、接続先のネットワークID書き込む必要も無い。

40

【0084】

例えば、計算機1におけるHDDの物理記憶容量を100メガバイトから200メガバイトに変更する場合、計算機1を構成する任意の機能モジュール2(例えばCPUの機能モジュール2)に接続しているデータ処理装置3の設定データ格納部17における接続設定条件として、HDDの物理記憶容量が100メガバイトと定義されている所を、200メガバイトに書き替える。

【0085】

50

CPUに接続しているデータ処理装置3の制御部13は、接続設定条件の書き替えを受け、現在の接続先HDDに200メガバイトの記憶容量が確保できるか否かを判定する。確保できる場合、制御部13は、HDDの接続設定条件が200メガバイトに変更された設定データを、計算機1を構成する各機能モジュール2に送り、各機能モジュール2の設定データに反映させる。

【0086】

一方、現在の接続先HDDに200メガバイトの記憶容量が確保できない場合、制御部13は、接続候補格納部16から、不足分の記憶容量を補うことのできる1つまたは複数の、記憶機能を持つ機能モジュール2(HDD等)を選択し、そのネットワークIDを設定データに反映させる。制御部13は、新たに追加された記憶機能を持つ機能モジュール2を含めた、計算機1を構成する各機能モジュール2に、上記設定データを送り、各機能モジュール2の設定データを更新する。

10

【0087】

なお、制御部13は、計算機1を構成する各機能モジュール2の中から、設定データの更新が必要な機能モジュール2を選択し、選択した機能モジュール2のみに対して設定データの更新を行っても良い。例えば、キーボードの機能モジュール2を追加した場合、キーボードの機能モジュール2は、CPUの機能モジュール2とのみ接続するので、CPUの機能モジュール2の設定データのみを更新することで計算機1を再構築することができる。この場合、計算機1のメモリやHDD等の機能モジュール2の設定データを更新する必要はない。

20

【0088】

〔本実施の形態の計算機の動作例〕

次に、計算機1の動作例について、図10及び図11に基づいて説明する。まず、比較のために、従来 of 計算機の動作について簡単に説明する。

【0089】

一般に、CPUは、命令を実行するときにメモリから必要なデータを読み出し、命令実行後に演算結果をメモリに書き戻す動作を行っている。本実施の形態の計算機1においても、CPUの機能モジュール2は、上記と同様の方法で命令の実行を行う。ここでは、計算機1の動作例として、CPUの機能モジュール2がメモリからデータを読み出す動作について、図10に基づいて説明する。図10は、CPU1(処理装置)、メモリ2(主記憶装置)、メモリ3(主記憶装置)、HDD1(補助記憶装置)、及びHDD2(補助記憶装置)の機能モジュール2間におけるデータの読み出し動作の一例を示す図である。

30

【0090】

なお、図10は、計算機1がCPU1、メモリ1、メモリ2、HDD1、及びHDD2の機能モジュール2を備えている状態を示しているが、計算機1を構成する機能モジュール2はこの組み合わせに限られない。図10に示す機能モジュール2の全てを備えていない場合でも計算機1を動作させることは可能であるし、図示していない機能モジュール2が計算機1の構成要素として接続されている場合でも計算機1を動作させることができる。

40

【0091】

まず、CPU1は、読み出したいデータの物理アドレスをデータ処理装置3に送る。データ処理装置3のネットワーク接続制御部9(図5参照)は、受け取った物理アドレス、すなわちメモリ1の物理アドレスに対して、ネットワーク4を介してデータ送信要求を送る。なお、ここでは、CPU1が読み出したいデータがメモリ1に物理的に格納されているものとする。従って、CPU1のネットワーク接続制御部9は、メモリ1の物理アドレスにデータ送信要求を送り、メモリ1の物理アドレスに送られたデータ送信要求は、メモリ1のネットワーク接続制御部9を介してメモリ1に渡される。データ送信要求を受け取ったメモリ1は、要求された物理アドレスに格納されているデータをCPU1のアドレスに送信する。該データは、メモリ1のネットワーク接続制御部

50

9、ネットワーク4、及びCPU 1のネットワーク接続制御部9を介してCPU 1に渡される。計算機1は、以上のようにしてデータの送受信を行いながら動作する。

【0092】

このように、メモリの物理アドレスにのみデータを格納する方法でも計算機1を動作させることができるが、容量が限られており、揮発してしまう性質を持つメモリをより有効に活用することができれば好ましい。従来の計算機では、限られたメモリ容量を有効に利用するために、仮想メモリを使用する方法が利用されている。仮想メモリを使用すると、仮想メモリ領域の一部のみに物理メモリが割り当てられることになるので、少ない物理メモリで大容量のデータやプログラムを処理することができる。

【0093】

仮想メモリを使用するためには、MMU (Memory Management Unit) というハードウェアが使用されている。MMUは、仮想メモリと物理メモリとのアドレス変換を行う。また、MMUは、物理メモリを仮想メモリ空間に割り当てる。この機能によって、メモリが足りなくなった場合、HDD等の2次記憶装置で代用することができるので、物理メモリよりも大きい仮想メモリをソフトウェアに提供することができる。

10

【0094】

本実施の形態の計算機1においても、仮想メモリを使用することができる。以下では、計算機1のより複雑な動作例として、仮想メモリを使用して動作する例を説明する。仮想メモリを使用するためには、上述のように、CPUとメモリとの間にMMUを備える必要がある。

20

【0095】

計算機1において、仮想メモリを用いたデータの送受信方法について、図10及び図11に基づいて説明する。図10に示すCPU 1、メモリ1、及びメモリ2の機能モジュール2は、データ処理装置3と機能モジュール2との間にMMU (アドレス変換手段) 18を備えている。

【0096】

なお、計算機1は、従来の計算機と異なり、機能モジュール2をネットワーク4で接続して構成されているので、MMU 18の構成も従来のMMUとは異なる部分がある。MMU 18は、基本的には従来のMMUと同様に、仮想メモリと物理メモリとのアドレス変換を行い、また、物理メモリをページに分割して管理し、これを仮想メモリ空間に割り当てる。これらの機能に加えて、MMU 18は、接続候補格納部16を参照し、接続候補格納部16に格納されているネットワークIDに従って他の記憶装置にデータ要求を転送することができる。

30

【0097】

続いて、計算機1における仮想メモリを用いたデータの送受信の具体的なデータの流れを図11に基づいて説明する。まず、計算機1の通常の動作状態におけるデータの送受信について説明する。

【0098】

CPU 1は、CPU 1と接続しているMMU 18に、CPU 1が要求するデータの仮想アドレスを送る。MMU 18は、仮想アドレスを仮想ページに変換し、データ処理装置3に渡す。データ処理装置3のネットワーク接続制御部9は、受け取った仮想ページにメモリ1のネットワークIDを対応付けて、ネットワーク・ページIDとしてネットワーク4に送り出す。

40

【0099】

ネットワーク・ページIDを受け取ったメモリ1のデータ処理装置3は、該ネットワーク・ページIDからネットワークIDを外して仮想ページに戻し、メモリ1のMMU 18に渡す。メモリ1のMMU 18は、仮想ページを物理アドレスに変換し、メモリ1に物理アドレスを渡す。メモリ1は、その物理アドレスに対応するデータをネットワーク接続制御部9に渡し、ネットワーク接続制御部9は、受け取ったデータにCPU 1のネットワークIDを付してネットワーク4に送り出す。

50

【0100】

なお、ここでは、CPU 1がメモリ 1にデータ送信要求を送る例を示したが、データ送信要求の送り先は、メモリ 1に限られず、計算機1を構成する記憶装置の機能モジュール2であれば任意のものが可能である。また、データ送信要求を複数の機能モジュール2に対して送ることもできる。ただし、HDDのような記憶装置にデータ送信要求を送る場合、メモリにデータ送信要求を送る場合と比較して、データの読み出しに時間がかかることがある。例えば、図11の破線矢印で示されるように、CPU 1のデータ処理装置3から、HDD 1にデータ送信要求を送る場合、HDD 1のデータ処理装置3が受け取ったネットワーク・ページIDを仮想ページに戻してHDD 1に渡す。そして、HDD 1は、受け取った仮想ページをディスクページに変換し、このディスクページに基づいて、HDD 1の物理ページに格納されているデータを読み出す。この、HDDの物理ページに格納されているデータを読み出す動作には、少なくとも数ミリ秒の時間を要するために、HDDからのデータの読み出しはメモリからの読み出しと比較して時間がかかることになる。

10

【0101】

また、それぞれのメモリにキャッシュ・メモリを設けることもできる。キャッシュ・メモリを設けた場合、CPUが行うデータ読み出しは、HDDの物理記憶にまでアクセスが及ばないので、効率のよい読み出しが可能となる。

【0102】

このようにして、CPU 1は要求したデータを受け取ることができる。なお、CPU 1がデータの書き込みを行う場合、CPU 1がデータ書き込みの指示を出すと、ネットワーク接続制御部9がメモリ 1のネットワークIDをデータに付してネットワーク4に送り出し、メモリ 1に記録する。続いて、メモリ 1が備えるデータ処理装置3のネットワーク接続制御部9は、上記データにHDD 1のネットワークIDを付してネットワーク4に送り出し、データは最終的にHDD 1に記録される。なお、最終的にデータが書き込まれる機能モジュール2は、HDD 1に限られず、HDD 2でもよいし、HDD 1及びHDD 2の両方でもよい。すなわち、不揮発性の記憶装置でありさえすれば、計算機1に接続するどの機能モジュール2に記憶しても良い。

20

【0103】

上述のように、CPU 1が書き込み指示を出した後は、メモリ 1、HDD 1等の記憶装置がCPU 1の指示とは独立に動作して書き込み処理を行う。従って、CPU 1は、指示を出すだけで書き込み処理が行われることになり、指示を出した直後から別の処理を行うことができる。

30

【0104】

次に、初期状態やページフォルトが発生した場合等における、仮想メモリを用いたデータの送受信方法について説明する。上述した、通常のデータ送受信の例では、CPU 1が要求したデータがメモリ 1に存在していたが、初期状態や、ページフォルトが発生した場合、揮発性のメモリが記憶していたデータは消えてしまうので、HDD等の不揮発性の記憶装置からデータの読み込みを行う必要がある。ここでは、CPU 1の要求するデータがメモリ 1及びメモリ 2には格納されておらず、HDD 1に格納されているものとする。

40

【0105】

初期状態やページフォルトが発生した場合等におけるデータの送受信も、通常のデータ送受信と同様に、CPU 1は、まず、メモリ 1にデータ送信要求を送る。しかしながら、初期状態やページフォルトが発生した後であるから、メモリ 1には要求されたデータが存在しない。

【0106】

この場合、メモリ 1のMMU 18は、接続候補格納部16を参照し、他の接続候補にデータ送信要求を転送する。なお、ここでは、メモリ 2、HDD 1、及びHDD 2が接続候補格納部16に格納されているものとする。初期状態やページフォルトが発生

50

した場合等におけるデータの送受信の、具体的なデータの流れを図11に基づいて説明する。なお、CPU 1がデータ送信要求を送ってからメモリ 1のMMU 18がメモリ 1に物理アドレスを送るまでの処理は、上記と同様であるから、説明を省略する。

【0107】

メモリ 1のMMU 18は、メモリ 1に物理アドレスを送るが、ここでは、メモリ 1には、メモリ 1のMMU 18から送られてくる物理アドレスに対応するデータが格納されていないので、データを返すことができない。この場合、メモリ#1のMMU 18は、接続候補格納部16に格納されているメモリ 2、HDD 1、及びHDD 2にデータ送信要求を転送する。このとき、仮想ページに転送先のネットワークIDを付加してネットワーク・ページIDとし、ネットワーク接続制御部9を介してネットワーク4に送り出す。このようにして、CPU 1が要求するデータを探してゆく。なお、図11には、HDD 3にネットワーク・ページIDを送る様子は示していない。

10

【0108】

ここでは、HDD 1にCPU 1が要求するデータが格納されていることを想定しているので、メモリ 1のMMU 18から転送されたデータ送信要求を受け取ったHDD 1は、上記の通常のデータ送受信時と同様にして要求されたデータを読み出し、メモリ 1に読み出したデータを送る。メモリ#1はそのデータを受け取ると、該データを自らに書き込むとともに、CPU 1に送る。このように、HDD 1から読み出されたデータは、メモリ 1に書き込まれるので、これ以降は、メモリ 1から上記データを読み出すことができる。

20

【0109】

初期状態やページフォールトが発生した場合等には、以上のようにしてCPU 1が要求するデータを読み出す。なお、図10及び図11では、CPU及びメモリの双方がMMU 18を備えている例を示したが、MMU 18は、CPUとメモリとの間に少なくとも1つ備えられていれば良い。

【0110】

なお、MMU 18をメモリ、HDD等といった記憶装置の機能モジュール2のそれぞれに設けた場合、記憶装置の機能モジュール2がそれぞれ仮想メモリ機能を備えることもできる。この場合、CPUのデータ送信要求を受けた記憶装置は、該データ送信要求の要求先アドレスをさらに別の記憶装置のアドレスに変換することができる。なお、仮想メモリは、最終的に物理的な記憶として1台、または複数台の記憶装置上に記憶させる。従って、CPUの要求したデータが複数の記憶装置上に分散して存在することとなる。すなわち、複数の記憶装置がキャッシュ・メモリのような働きをすることにより、計算機1の動作をより高速化することができる。

30

【0111】

上述の実施形態では、1つの機能モジュール2が、1つの計算機1のみで使用される態様を示した。しかしながら、設定データに複数の接続先を書き込むことにより、1つの機能モジュール2を複数の計算機1で共有して利用することもできる。なお、1つの機能モジュール2を複数の計算機1で共有する場合、各計算機で矛盾が生じないようにする必要がある。例えば、HDDを2台の計算機（仮に、計算機1A及び計算機1Bとする）で共有する場合、計算機1Aが書き込んだデータを計算機1Bから消去できるというようなことが無いようにする必要がある。この方法の一例としては、HDDのパーティションを設定データとして書き込んでおく、または〔背景技術〕で説明したNASのシステムを利用する等の方法が可能である。

40

【0112】

無論、複数の計算機1で共有して利用することができる機能モジュール2は、HDDに限られず、任意の機能モジュール2を共有して利用することができる。なお、CPUを共有することもできるが、CPUを共有する場合、BIOSを書き込んだCPUは避けることが望ましい。従って、計算機1が占有するCPUを少なくとも1つ設けることが望ましい。

50

【 0 1 1 3 】

〔 変形例 〕

上述の説明では、データ処理装置 3 の自己紹介部 8 が機能モジュール 2 の公開データをネットワーク 4 上に公開するので、計算機 1 を利用しようとするユーザは、ネットワーク 4 に接続して公開データを閲覧し、必要な機能モジュール 2 を選択することによって、計算機 1 が構成される形態を示した。ここでは、変形例として、自己紹介部 8 が公開データを格納していない場合に計算機 1 を構成方法について説明する。

【 0 1 1 4 】

計算機 1 を構成するためには、ネットワーク 4 に接続している機能モジュール 2の中から、計算機 1 を構成するために必要な機能モジュール 2 を選択する必要がある。上述の実施の形態では、自己紹介部 8 の公開データ格納部 1 4 が格納する公開データを、自己紹介部 8 がネットワーク 4 に送信することによって、ユーザは、公開データを閲覧し、必要な機能モジュール 2 を選択することができる。

【 0 1 1 5 】

ここで、例えば、図 4 に示した組織構築手段 6 に公開データを格納しておけば、計算機 1 のユーザは、組織構築手段 6 にアクセスすることによって公開データを閲覧し、計算機 1 を構成する機能モジュール 2 を選択することができる。

【 0 1 1 6 】

また、例えば、計算機 1 を利用しようとするユーザが、計算機 1 を構成する各機能モジュール 2 のネットワーク ID を知っていれば、従来の計算機をネットワーク 4 に接続して計算機 1 を構成する各機能モジュール 2 に直接アクセスし、設定データを書き込むことによって計算機 1 を構成することができる。

【 0 1 1 7 】

この変形例のように、自己紹介部 8 は、公開データ格納部 1 4 を備えていない場合でも、計算機 1 を利用しようとするユーザが、ネットワーク 4 に接続している機能モジュール 2の中から、計算機 1 を構成する機能モジュール 2 を選択し、選択した機能モジュール 2 のデータ処理装置 3 に設定データを書き込むことができさえすれば、計算機 1 を構成することができる。この場合、自己紹介部 8 は、公開データ格納部 1 4 を備えていなくても良いし、公開データをネットワーク 4 に送信する必要もない。従って、自己紹介部 8 を大幅に簡略化することができる。

【 0 1 1 8 】

本発明は、上記の実施の形態に限定されるものではなく、本発明の範囲内で種々の変更が可能である。

【 産業上の利用可能性 〕

【 0 1 1 9 】

本発明は、メモリ、HDD、CPU、ディスプレイ等の計算機を構成する機能モジュールを、ネットワーク上で共有することを可能にするものであり、上記以外にも種々の機能モジュールに適用することができる。

【 図面の簡単な説明 〕

【 0 1 2 0 】

【 図 1 】 自己紹介部によって機能モジュールを接続して計算機を構成する例を示す図である。

【 図 2 】 従来の計算機の概略構成を示す図である。

【 図 3 】 本発明の計算機の概略構成を示す図である。

【 図 4 】 組織構築手段を備えた本発明の計算機の概略構成を示す図である。

【 図 5 】 本発明のデータ処理装置の概略構成を示す図である。

【 図 6 】 従来の計算機で使用されるソフトウェアを示す図である。

【 図 7 】 本発明の計算機におけるデータ伝達の一例を示す図である。

【 図 8 】 本発明の計算機における初期設定の方法を示す図である。

【 図 9 】 本発明のデータ処理装置における自己紹介部の詳細を示す図である。

10

20

30

40

50

【図10】本発明の計算機の動作例を示すものであり、CPUの要求するデータをメモリまたはHDDから読み出す動作を示す図である。

【図11】図10に示すデータ読出し動作時におけるデータの流れを示す図である。

【符号の説明】

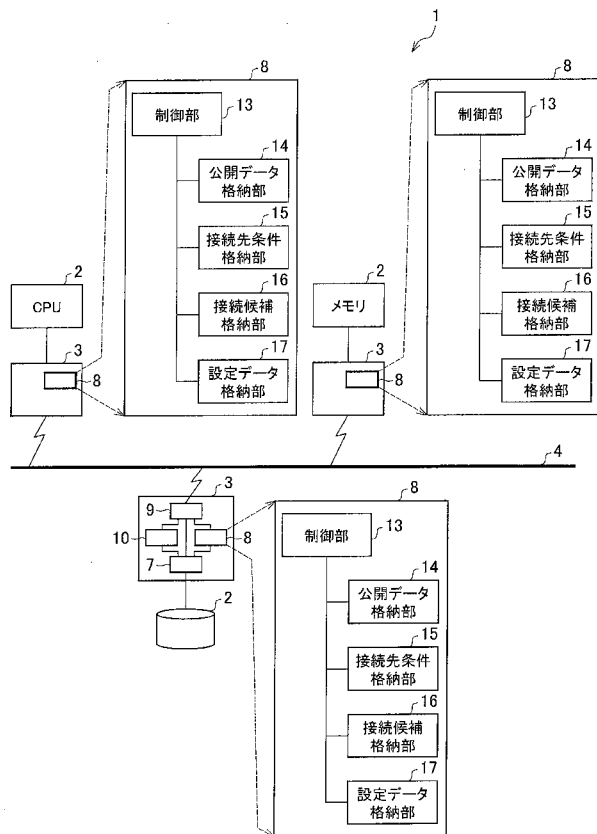
【0121】

- 1 計算機
- 2 機能モジュール（機能要素）
- 3 データ処理装置
- 4 ネットワーク
- 5 表示画面
- 6 組織構築手段
- 7 機能モジュール接続制御部（機能要素接続制御手段）
- 8 自己紹介部（自己紹介手段）
- 9 ネットワーク接続制御部（ネットワーク接続制御手段）
- 10 データ変換部（データ変換手段）
- 11 スーパーノード
- 12 ノード
- 13 制御部
- 14 公開データ格納部
- 15 接続先条件格納部
- 16 接続候補格納部
- 17 設定データ格納部
- 18 MMU（アドレス変換手段）

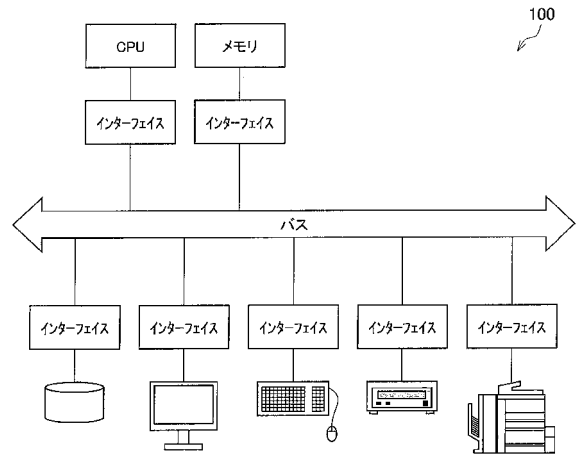
10

20

【図1】

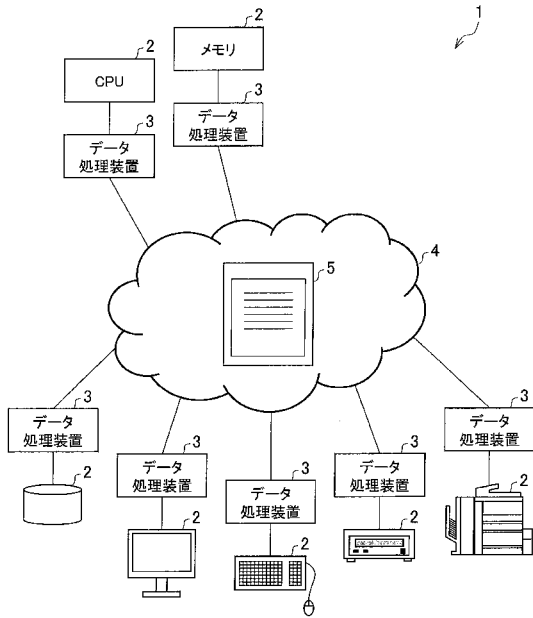


【図2】

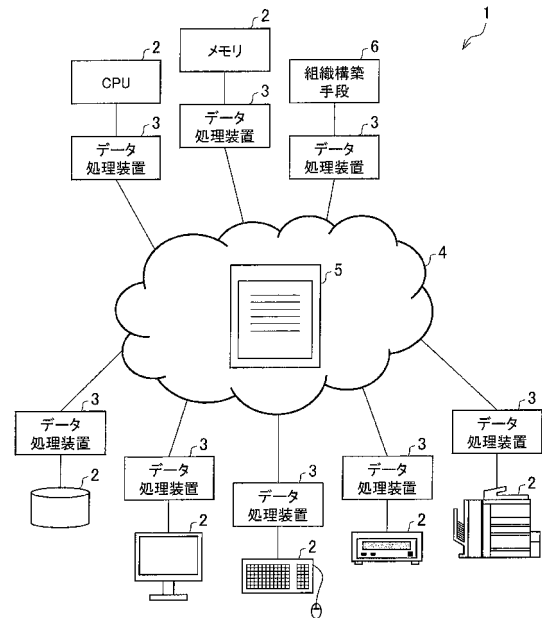


100

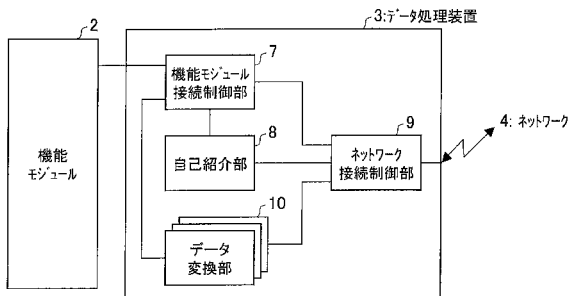
【 図 3 】



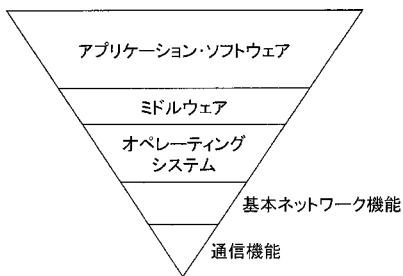
【 図 4 】



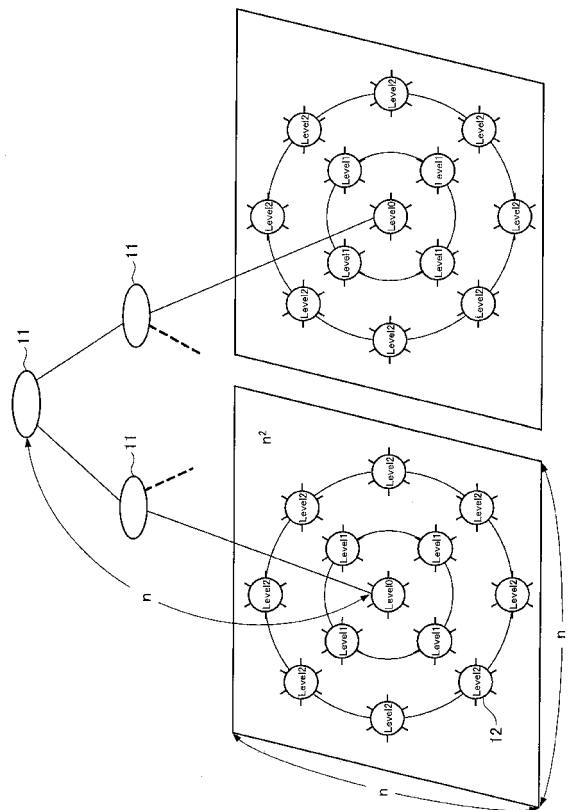
【 図 5 】



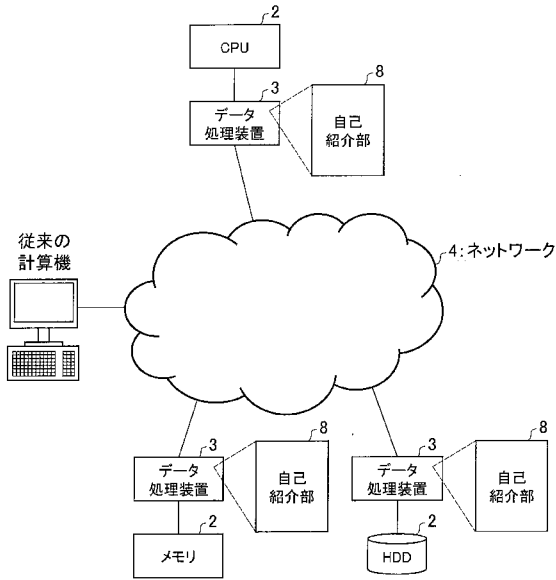
【 図 6 】



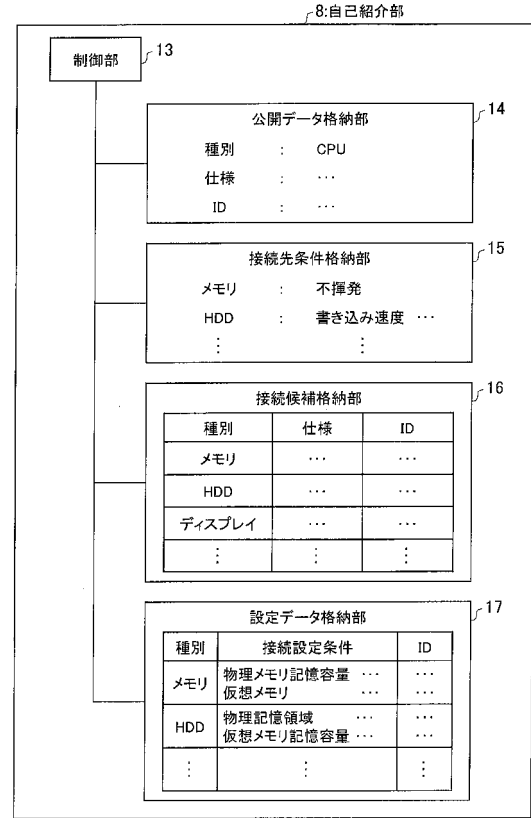
【 図 7 】



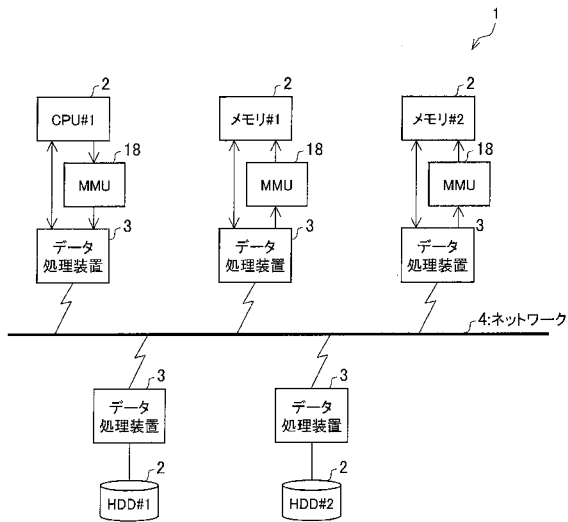
【 図 8 】



【 図 9 】



【 図 10 】



【 図 11 】

