

(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2007-207173  
(P2007-207173A)

(43) 公開日 平成19年8月16日(2007.8.16)

(51) Int. Cl. F I テーマコード (参考)  
G06F 11/34 (2006.01) G06F 11/34 S 5B042

審査請求 未請求 請求項の数 10 O L (全 23 頁)

(21) 出願番号 特願2006-28517(P2006-28517)  
(22) 出願日 平成18年2月6日(2006.2.6)

(71) 出願人 000005223  
富士通株式会社  
神奈川県川崎市中原区上小田中4丁目1番1号  
(74) 代理人 100092152  
弁理士 服部 毅麿  
(72) 発明者 小野 美由紀  
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内  
(72) 発明者 山村 周史  
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内  
(72) 発明者 平井 聡  
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

最終頁に続く

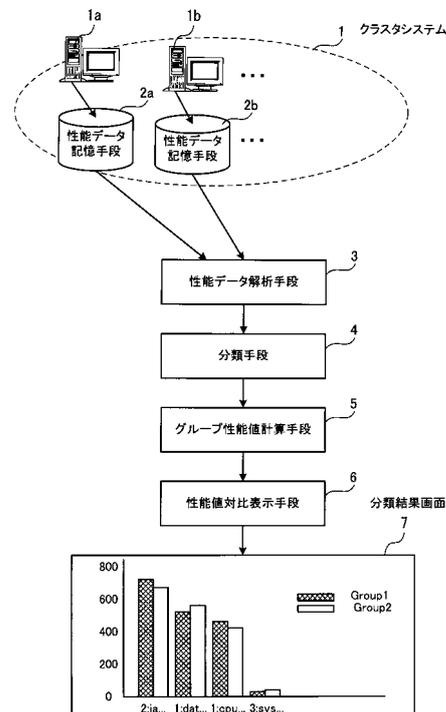
(54) 【発明の名称】 性能分析プログラム、性能分析方法、および性能分析装置

(57) 【要約】 (修正有)

【課題】 未知の問題を含め、クラスタシステムにおける性能面での特異なノードを効率よく調査できるようにする。

【解決手段】 性能データ解析手段3により、クラスタシステム1を構成する各ノード1a, 1b, ...の性能データが格納された性能データ記憶手段2a, 2b, ...から、各ノード1a, 1b, ...の性能データが収集される。次に、分類手段4により、所定の分類条件に従って、性能データ解析手段3が収集した性能データに基づいてノード1a, 1b, ...それぞれが複数のグループに分類される。さらに、グループ性能値計算手段5により、各グループに分類されたノードの性能データに基づいてグループ毎の性能データが統計処理され、各グループの性能データ種別毎の統計値が計算される。そして、性能値対比表示手段6により、各グループの統計値が性能データ種別毎にグループ間で対比して表示される。

【選択図】 図1



**【特許請求の範囲】****【請求項 1】**

クラスタシステムの性能を分析するための性能分析プログラムにおいて、  
コンピュータを、

前記クラスタシステムを構成する各ノードの複数種別の性能データが格納された性能データ記憶手段から前記ノードそれぞれの前記性能データを収集し、収集した前記性能データに基づいて前記ノードそれぞれの性能値を解析する性能データ解析手段、

所定の分類条件に従って、前記性能データ解析手段が収集した前記性能データを統計処理することによって前記ノードそれぞれを複数のグループに分類する分類手段、

前記グループに分類された前記ノードの前記性能データに基づいて前記グループ毎の前記性能データを統計処理し、前記グループそれぞれの性能データ種別毎の統計値を計算するグループ性能値計算手段、

前記グループそれぞれの前記統計値を性能データ種別毎に前記グループ間で対比して表示する性能データ対比表示手段、

として機能させることを特徴とする性能分析プログラム。

10

**【請求項 2】**

前記性能データ解析手段は、前記ノードそれぞれで実行される関数の実行時間を示すプロファイリングデータを前記性能データとして収集し、

前記分類手段は、関数の実行時間によって前記ノードを分類することを特徴とする請求項 1 記載の性能分析プログラム。

20

**【請求項 3】**

前記性能データ解析手段は、前記ノードそれぞれの CPU における命令の実行状況を示すデータを前記性能データとして収集し、

前記分類手段は、前記 CPU における命令の実行状況によって前記ノードを分類することを特徴とする請求項 1 記載の性能分析プログラム。

**【請求項 4】**

前記性能データ解析手段は、前記ノードそれぞれにおけるオペレーティングシステムの動作状況を示す前記性能データを収集し、

前記分類手段は、前記オペレーティングシステムの動作状況によって前記ノードを分類することを特徴とする請求項 1 記載の性能分析プログラム。

30

**【請求項 5】**

性能データ対比表示手段は、任意の前記グループの統計値を 1 としたときの、他の前記グループの統計値の値を前記グループ間で対比して表示することを特徴とする請求項 1 記載の性能分析プログラム。

**【請求項 6】**

前記性能データ対比表示手段は、前記グループそれぞれの棒グラフで表示した統計値に対して、前記グループに属するノード毎の性能データの分散状況を示すバーを表示することを特徴とする請求項 1 記載の性能分析プログラム。

**【請求項 7】**

コンピュータにより、クラスタシステムの性能を分析するための性能分析方法において

40

性能データ解析手段が、前記クラスタシステムを構成する各ノードの複数種別の性能データが格納された性能データ記憶手段から前記ノードそれぞれの前記性能データを収集し、収集した前記性能データに基づいて前記ノードそれぞれの性能値を解析し、

分類手段が、所定の分類条件に従って、前記性能データ解析手段が収集した前記性能データを統計処理することによって前記ノードそれぞれを複数のグループに分類し、

グループ性能値計算手段が、前記グループそれぞれに分類された前記ノードの前記性能データに基づいて前記グループ毎の前記性能データを統計処理し、前記グループそれぞれの前記性能データ種別毎の統計値を計算し、

性能データ対比表示手段が、前記グループそれぞれの前記統計値を前記性能データ種別

50

毎に前記グループ間では対比して表示する、  
ことを特徴とする性能分析方法。

【請求項 8】

前記性能データ解析手段は、前記ノードそれぞれで実行される関数の実行時間を示すプロファイリングデータを前記性能データとして収集し、

前記分類手段は、関数の実行時間によって前記ノードを分類することを特徴とする請求項 7 記載の性能分析方法。

【請求項 9】

クラスタシステムの性能を分析するための性能分析装置において、

前記クラスタシステムを構成する各ノードの複数種別の性能データが格納された性能データ記憶手段から前記ノードそれぞれの前記性能データを収集し、収集した前記性能データに基づいて前記ノードそれぞれの性能値を解析する性能データ解析手段と、

所定の分類条件に従って、前記性能データ解析手段が収集した前記性能データを統計処理することによって前記ノードそれぞれを複数のグループに分類する分類手段と、

前記グループそれぞれに分類された前記ノードの前記性能データに基づいて前記グループ毎の前記性能データを統計処理し、前記グループそれぞれの性能データ種別毎の統計値を計算するグループ性能値計算手段と、

前記グループそれぞれの前記統計値を前記性能データ種別毎に前記グループ間では対比して表示する性能データ対比表示手段と、

を有することを特徴とする性能分析装置。

【請求項 10】

前記性能データ解析手段は、前記ノードそれぞれで実行される関数の実行時間を示すプロファイリングデータを前記性能データとして収集し、

前記分類手段は、関数の実行時間によって前記ノードを分類することを特徴とする請求項 9 記載の性能分析装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明はクラスタシステムにおける性能分析プログラム、性能分析方法、および性能分析装置に関し、特に複数のノードから採取される性能データを統計処理することで性能分析を行う性能分析プログラム、性能分析方法、および性能分析装置に関する。

【背景技術】

【0002】

R & D (Research and Development)、HPC (High Performance Computing)、バイオインフォマティクスなどの分野において、複数の計算機をネットワークで接続し、1つの仮想計算機システムを構成して並列処理を行う“クラスタシステム”の利用が進んでいる。クラスタシステムは、複数の計算機(ノード)をネットワークで接続し、1つの仮想計算機システムとして機能させたものである。クラスタシステムでは、与えられた処理を、各ノードで並列に処理する。

【0003】

クラスタシステムは、低コストで高性能なシステムを構築できる反面、要求性能が高まるほど構成されるノードが増大するといった問題がある。このため、多数のノードの動作状況を把握するための技術が必要となっている。

【0004】

そこで、クラスタシステムを運用する際に、適宜クラスタシステムの性能分析を行うことが考えられる。例えば、複数の計算機上でのプロセスの動作性能に基づいて、プロセスのスケジューリングを行うことができる(特許文献 1 参照)。

【0005】

また、性能分析を行うことで、例えば、クラスタシステムを構成するノードの 1 つに何らかの障害が発生した場合に、その障害の発生を迅速に検出することが可能となる。性能

10

20

30

40

50

分析を行うシステムとして、例えば、クラスタシステムに関する各種統計情報を表示するシステムが考えられている（非特許文献1参照）。

【0006】

ただし、クラスタシステムを構成する個々のノード上では、独立してオペレーティングシステムおよびアプリケーションが動作している。このため、システム全体として評価しようとした場合に、ノード数分の様々な情報が採取されることになる。すると、大規模クラスタシステムにおいては処理すべき情報量が膨大となり、各ノードの動作状況を個別に判断し、その中から問題を有するノードを検出することは困難である。

【0007】

そのため、従来のクラスタシステムの評価手法は、代表的なノードの性能値の比較から動作を推測することが主となる。また、問題点の抽出においても、個々のノード上で採取するデータに閾値を設け、その閾値を超えたものを特定するといった手法が取られていた。また、各ノードのデータを統計処理して分類することにより、性能評価において重要な特徴を抽出しようとする試みも行われている（非特許文献2参照）。

【特許文献1】特開2003-6175号公報

【非特許文献1】Intel Trace Analyzer, [online]、インテル株式会社、[平成18年1月13日検索]、インターネット<URL:http://www.intel.com/cd/software/products/ijk/jpn/cluster/224160.htm>

【非特許文献2】Dong H. Ahn and Jeffrey S. Vetter, "Scalable Analysis Techniques for Microprocessor Performance Counter Metrics" [online]、2002年、[平成18年1月13日検索]、インターネット、<URL: http://citeseer.ist.psu.edu/ahn02scalable.html>

【発明の開示】

【発明が解決しようとする課題】

【0008】

しかし、従来の評価手法ではいずれの方法を用いたとしても、大規模なクラスタシステムを構成する多数のノードから、性能面において注目すべきノードを特定することが難しかった。

【0009】

例えば、閾値を利用する手法では既知の問題においては有効であるが、これまでと動作の異なる未知の問題には対処できない。すなわち、閾値を利用するには、どのような情報がどの程度の値になったときに異常と判断すべきかを、予め解析しておかなければならない。しかし、システムの障害等は、予期しない原因で発生する 경우가多々ある。ハードウェア性能が日々進歩し、セキュリティ対策等のシステム運用方法を適宜改善しなければならない現状において、全ての障害原因を予測するのは不可能である。

【0010】

なお、非特許文献1では、性能データを利用した自動的なグループ化機能は提供されていない。そのため、多数のノードから構成されるクラスタシステムの場合には膨大なデータをユーザが手探りで評価しなければならない。

【0011】

また、非特許文献2では、分類結果を単に開発者にフィードバックさせる、あるいは分類結果を他のシステムへ入力することを目的としている。そのため、分類したグループ間の情報を比較することまでは考慮していない。

【0012】

本発明はこのような点に鑑みてなされたものであり、未知の問題を含め、クラスタシステムにおける性能面での特異なノードを効率よく調査することができる性能分析プログラム、性能分析方法、および性能分析装置を提供することを目的とする。

【課題を解決するための手段】

【0013】

本発明では上記課題を解決するために、図1に示すような性能分析プログラムが提供さ

10

20

30

40

50

れる。本発明に係る性能分析プログラムは、クラスタシステム 1 の性能を分析するために、図 1 に示す機能をコンピュータに実行させることができる。

【0014】

性能データ解析手段 3 は、クラスタシステム 1 を構成する各ノード 1 a , 1 b , . . . の性能データが格納された性能データ記憶手段 2 a , 2 b , . . . から、各ノード 1 a , 1 b , . . . の複数種別の性能データを収集する。分類手段 4 は、所定の分類条件に従って、性能データ解析手段 3 が収集した性能データを統計処理することによってノード 1 a , 1 b , . . . それぞれを複数のグループに分類する。グループ性能値計算手段 5 は、グループそれぞれに分類されたノードの性能データに基づいてグループ毎の性能データを統計処理し、各グループの性能データ種別毎の統計値を計算する。性能値対比表示手段 6 は、グループそれぞれの統計値を性能データ種別毎にグループ間で対比して表示する。

10

【0015】

このような性能分析プログラムを実行するコンピュータでは、性能データ解析手段 3 により、クラスタシステム 1 を構成する各ノード 1 a , 1 b , . . . の性能データが格納された性能データ記憶手段 2 a , 2 b , . . . から、各ノード 1 a , 1 b , . . . の性能データが収集される。次に、分類手段 4 により、所定の分類条件に従って、ノード 1 a , 1 b , . . . それぞれが複数のグループに分類される。さらに、グループ性能値計算手段 5 により、各グループに分類されたノードの性能データに基づいてグループ毎の性能データが統計処理され、各グループの性能データ種別毎の統計値が計算される。そして、性能値対比表示手段 6 により、各グループの統計値が性能データ種別毎にグループ間で対比して表示される。

20

【発明の効果】

【0016】

本発明では、性能データに応じたノードを複数のグループに分類し、各グループの性能値を対比して表示するようにしたため、問題のあるノードがどのグループに属するのかを容易に判断することができる。その結果、未知の問題を含め、クラスタシステムにおける性能面での特異なノードを効率よく調査することができる。

【発明を実施するための最良の形態】

【0017】

以下、本発明の実施の形態を図面を参照して説明する。

30

図 1 は、本実施の形態の概略を示す図である。クラスタシステム 1 は、複数のノード 1 a , 1 b , . . . で構成される。各ノード 1 a , 1 b , . . . は、性能データ記憶手段 2 a , 2 b , . . . を有している。性能データ記憶手段 2 a , 2 b , . . . には、対応するノード 1 a , 1 b , . . . の性能に関するデータが格納される。

【0018】

そして、クラスタシステム 1 の個々のノード 1 a , 1 b , . . . が同一の動作をすると仮定し、分析を行うために、性能データ解析手段 3、分類手段 4、グループ性能値計算手段 5、および性能値対比表示手段 6 が設けられている。

【0019】

性能データ解析手段 3 は、クラスタシステム 1 を構成する各ノード 1 a , 1 b , . . . の性能データ（ノードから採取可能な性能に関するデータ）が格納された性能データ記憶手段 2 a , 2 b , . . . から、各ノード 1 a , 1 b , . . . の性能データを収集する。なお、性能データ解析手段 3 は、採取した性能データを解析し、性能データの種別に応じて性能データを加工することもできる。例えば、性能データ解析手段 3 は、採取時間内の合計値あるいは単位時間の平均値を性能値（性能データに基づく性能の解析結果として得られる数値）として算出する。

40

【0020】

分類手段 4 は、所定の分類条件に従って、性能データ解析手段 3 が収集した性能データを統計処理することによりノード 1 a , 1 b , . . . それぞれを複数のグループに分類する。グループ数は、例えば、初期値（デフォルト値）が決まっており、ユーザからの指定

50

が無い場合には、各ノードが初期値の数（例えば「2」）のグループに分類される。分類のグループ数がユーザによって指定された場合、各ノードが指定された数のグループに分類される。

#### 【0021】

グループ性能値計算手段5は、各グループに分類されたノードの性能データに基づいてグループ毎の性能データを統計処理し、各グループの性能データ種別毎の統計値を計算する。例えば、グループ性能値計算手段5は、性能データ種別毎に、各グループに属するノード群の平均値等をグループの性能値として算出する。

#### 【0022】

性能値対比表示手段6は、各グループの統計値を性能データ種別毎にグループ間で対比して表示する。例えば、性能値対比表示手段6は、グループ毎の性能値を棒グラフで示した分類結果画面7を表示する。この時、棒グラフは、性能データの種別毎に纏められる。性能データ種別毎の各グループの性能値が容易に比較できる。

10

#### 【0023】

このようなシステムによれば、性能データ解析手段3により、クラスタシステム1を構成する各ノード1a, 1b, ...の性能データが格納された性能データ記憶手段2a, 2b, ...から、各ノード1a, 1b, ...の性能データが収集される。次に、分類手段4により、所定の分類条件に従って、性能データ解析手段3が収集した性能データの解析処理が行われ、ノード1a, 1b, ...それぞれが複数のグループに分類される。さらに、グループ性能値計算手段5により、各グループに分類されたノードの性能データに基づいてグループ毎の性能データが統計処理され、各グループの性能データ種別毎の統計値が計算される。そして、性能値対比表示手段6により、各グループの統計値が性能データ種別毎にグループ間で対比して表示される。

20

#### 【0024】

その結果、実行時に採取した各ノードの性能データが統計処理され、ノードが任意の数のグループに分類され、個々のノードではなく分類されたグループ間の性能比較が行われる。グループに分けて性能比較を行うことにより、多数のノードを個別に性能評価するよりも、処理負荷が軽減される。しかも、グループ毎の性能値を対比表示することで、特異な性能値を有するグループを容易に特定することができる。そして、特定されたグループをさらに分類すれば、何らかの問題を有しているノードを容易に特定することができる。すなわち、ノードに発生する問題が、既知であるか未知であるかに関係なく、何らかの問題を有するノードが容易に特定できる。

30

#### 【0025】

次に、本実施の形態の詳細を説明する。

図2は、本実施の形態のシステム構成例を示す図である。クラスタシステム200は、複数のノード210, 220, 230, ...で構成されている。ノード210, 220, 230, ...には、ネットワーク10を介して管理サーバ100が接続されている。管理サーバ100は、クラスタシステム200から性能データを収集して、統計処理を行う。

#### 【0026】

図3は、本実施の形態に用いる管理サーバのハードウェア構成例を示す図である。管理サーバ100は、CPU (Central Processing Unit) 101によって装置全体が制御されている。CPU 101には、バス107を介してRAM (Random Access Memory) 102、ハードディスクドライブ (HDD: Hard Disk Drive) 103、グラフィック処理装置104、入力インタフェース105、および通信インタフェース106が接続されている。

40

#### 【0027】

RAM 102には、CPU 101に実行させるOS (Operating System) のプログラムやアプリケーションプログラムの少なくとも一部が一時的に格納される。また、RAM 102には、CPU 101による処理に必要な各種データが格納される。HDD 103には

50

、OSやアプリケーションプログラムが格納される。

【0028】

グラフィック処理装置104には、モニタ11が接続されている。グラフィック処理装置104は、CPU101からの命令に従って、画像をモニタ11の画面に表示させる。入力インタフェース105には、キーボード12とマウス13とが接続されている。入力インタフェース105は、キーボード12やマウス13から送られてくる信号を、バス107を介してCPU101に送信する。

【0029】

通信インタフェース106は、ネットワーク10に接続されている。通信インタフェース106は、ネットワーク10を介して、他のコンピュータとの間でデータの送受信を行う。

10

【0030】

以上のようなハードウェア構成によって、本実施の形態の処理機能を実現することができる。なお、図3には、管理サーバ100のハードウェア構成のみが示されているが、各ノード210, 220, 230, ...も同様のハードウェアで実現できる。

【0031】

図4は、性能分析を行うための機能を示すブロック図である。図4には、ノード210と管理サーバ100との機能が示されている。

ノード210は、マシン情報取得部211、性能データ取得部212、および性能データ記憶部213を有している。

20

【0032】

マシン情報取得部211は、OS等で提供されている機能を用いて、数値で表現可能なノード210のマシン構成情報(ハードウェア性能データ)を、性能データとして取得する。ハードウェア性能データとしては、CPU数、CPU動作周波数、キャッシュサイズなどがある。マシン情報取得部211は、採取したマシン情報を性能データ記憶部213に格納する。このようなマシン情報は、クラスタシステムが性能の異なるマシンから構成されている場合やクラスタシステム間の性能値を比較する場合に、分類対象として利用される。

【0033】

性能データ取得部212は、ノード210が実際に処理を実行することで測定できる性能データ(実行性能データ)を取得する。実行性能データとしては、例えば、IPC(Instruction Per Cycle:1サイクルあたりに実行可能な命令数)といったCPUレベルの実行性能を示すデータや、実行時間やキャッシュミスなどのイベントの発生回数を関数レベルで集計したデータ(プロファイリングデータ)がある。これらのデータは、プロファイリングツール等の各種システム管理ツールを使用することで収集可能である。性能データ取得部212は、採取した性能データを性能データ記憶部213に格納する。

30

【0034】

性能データ記憶部213は、ハードウェア性能データや実行性能データを、性能データとして格納する。

管理サーバ100は、クラスタ性能値計算部111、クラスタ性能値出力部112、性能データ解析部113、分類条件指定部114、分類対象選択部115、性能データ分類部116、クラスタ分散状況出力部117、グループ性能値計算部118、グラフ作成部119、および分類結果出力部120、グループ選択部121、およびグループ分散状況出力部122を有している。

40

【0035】

クラスタ性能値計算部111は、各ノード210, 220, 230, ...の性能データ記憶部から性能データを取得し、クラスタ全体の性能値を計算する。計算結果は、クラスタ性能値出力部112と性能データ解析部113とに渡される。

【0036】

クラスタ性能値出力部112は、クラスタ性能値計算部111から受け取ったクラスタ

50

システム 200 の性能値を、モニタ 11 等に出力する。

性能データ解析部 113 は、各ノード 210, 220, 230, … の性能データ記憶部から性能データを採取し、性能データを必要に応じて加工する。加工された性能データは、性能データ分類部 116 に渡される。

【0037】

分類条件指定部 114 は、ユーザからの操作に基づく分類条件の入力を受け付ける。分類条件指定部 114 は、入力された分類条件を、分類対象選択部 115 に渡す。

分類対象選択部 115 は、分類条件指定部 114 から渡された分類条件に基づいて、分類対象項目を選択する。選択した分類対象項目は、性能データ分類部 116 に渡される。

【0038】

性能データ分類部 116 は、階層的にグループを構成していく階層的グループ化の手法を用いてノードを分類する。ここで、階層的グループ化（階層的クラスタ分析）とは、統計解析において、大量のデータを入力とし、類似するデータ同士を、少数のグループにまとめる手法であり、階層的にグループを構成していくものである。分類結果は、クラスタ分散状況出力部 117 とグループ性能値計算部 118 とに渡される。

【0039】

クラスタ分散状況出力部 117 は、クラスタシステム 200 全体での各種性能データの分散状況を、モニタ 11 等に出力する。

グループ性能値計算部 118 は、分類されたグループ毎の性能値を計算する。計算された性能値は、グラフ作成部 119 とグループ選択部 121 に渡される。

【0040】

グラフ作成部 119 は、グループ毎の性能値を視覚的に分かりやすく比較するために、性能値を示すグラフを作成する。作成したグラフデータは、分類結果出力部 120 に渡される。

【0041】

分類結果出力部 120 は、グラフデータに基づいて、モニタ 11 にグラフを表示する。

グループ選択部 121 は、分類結果出力部 120 により出力された結果から 1 つのグループを選択する。

【0042】

グループ分散状況出力部 122 は、グループ選択部 121 により選択されたグループにおける性能値の分散状況を表すグラフを作成し出力する。

このような構成のシステムにおいて、クラスタシステム 200 の性能分析が行われる。その際、分類するグループ数や分類対象とする項目を変更しながらグループ間の性能比較を繰り返すことで、より確実に障害を有するノードを検出できる。例えば、クラスタシステムが設計通りの性能を発揮できない場合、以下のような手順で性能分析を行う。

【0043】

図 5 は、性能分析手順を示すフローチャートである。図 5 に示したのは、CPU レベルの性能データを用いた分類により異常ノード群と注目すべき性能項目を抽出し、さらにプロファイリングデータを用いた分類により異常ノード群と異常な関数群の特定を行う場合の例である。以下、図 5 に示す処理をステップ番号に沿って説明する。

【0044】

[ステップ S1] クラスタシステムの各ノードの性能データ取得部が、CPU レベルの性能データを採取し、性能データ記憶部に格納する。

[ステップ S2] 管理サーバ 100 の性能データ解析部 113 は、各ノードの性能データ記憶部から、性能データ取得部が採取した性能データを収集する。

【0045】

[ステップ S3] 性能データ分類部 116 は、性能データの統計処理結果に基づいて、ノードを複数のグループに分類する。グループ化においては、例えば、階層的なグループ化を行うことができる。

【0046】

10

20

30

40

50

[ステップS4] グループ性能値計算部118は、各グループの性能値を計算する。計算された性能値に基づいて、グラフ作成部119がグループ間の性能値を比較するグラフを作成し、分類結果出力部120が表示する。ユーザは、表示された分類結果に基づいて、異常な性能のグループ、または異常な性能項目があるか否かを判断する。異常なグループまたは性能項目が見つければ、処理がステップS6に進められる。異常な性能項目が見つからなければ、処理がステップS5に進められる。

【0047】

[ステップS5] ユーザは、分類条件指定部114または分類対象選択部115に対して、グループ数や対象性能項目を変更する操作入力を行う。すると、変更されたグループ数や対象性能項目が、分類条件指定部114または分類対象選択部115から性能データ分類部116に渡される。その後、処理がステップS3に進められ、ノードの再分類が行われる。

10

【0048】

このように、まず、CPUレベルの性能データを採取し、これを用いて分類して異常なノード群の抽出が試みられる。最初はデフォルトの分類条件(例えば、グループ数2、CPU毎の推奨性能項目群)で分類され、グループ内の分散状況とグループ間の性能差が確認される。

【0049】

グループ間の性能差が小さく、かつ各グループの分散が小さければ、分類は終了する。すなわち、異常ノード群はないと判断される。

20

グループ間の性能差が大きく、各グループの分散が小さければ分類は終了する。すなわち、性能が極端に悪いグループ内に、何らかの問題があると判断される。

【0050】

グループ内の分散が大きければ、グループ数を増やして、再分類が行われることとなる。また、グループ間の性能差が大きければ、性能の悪いグループに注目する。さらに、性能差の大きい性能項目に注目し、分類時に使用する測定データを性能差の大きな項目のみに絞ってみることも考えられる。

【0051】

このように、CPUの性能データに基づいて問題のグループを特定した後、処理がステップS6以降に進められる。

30

[ステップS6] クラスタシステムの各ノードの性能データ取得部が、問題のある性能項目に関するプロファイリングデータを採取し、性能データ記憶部に格納する。

【0052】

[ステップS7] 管理サーバ100の性能データ解析部113は、各ノードの性能データ記憶部から、性能データ取得部が採取したプロファイリングデータを収集する。

[ステップS8] 性能データ分類部116は、プロファイリングデータの統計処理結果に基づいて、ノードを複数のグループに分類する。グループ化においては、例えば、階層的なグループ化を行うことができる。

【0053】

[ステップS9] グループ性能値計算部118は、各グループの性能値を計算する。計算された性能値に基づいて、グラフ作成部119がグループ間の性能値を比較するグラフを作成し、分類結果出力部120が表示する。ユーザは、表示された分類結果に基づいて、異常な性能のグループ、または異常な関数があるか否かを判断する。異常なグループまたは関数が見つければ、処理が終了する。異常な関数が見つからなければ、処理がステップS10に進められる。

40

【0054】

[ステップS10] ユーザは、分類条件指定部114または分類対象選択部115に対して、グループ数や対象関数を変更する操作入力を行う。すると、変更されたグループ数や対象関数が、分類条件指定部114または分類対象選択部115から性能データ分類部116に渡される。その後、処理がステップS8に進められ、ノードの再分類が行われる

50

。

## 【0055】

このように、実行時間あるいは問題の性能項目（例えば、キャッシュミス回数等）について、プロファイリングデータを採取し、分類する。最初はデフォルトの分類条件（例えば、グループ数2、上位10関数の実行時間あるいは測定した性能項目の発生回数）で分類し、グループ内の分散状況とグループ間の性能差を確認する。確認の方法は、CPUレベルの性能データと同様である。再分類時に使用する関数の数や対象関数を変えることもできる。

## 【0056】

例えば、CPUレベルの分析で、他のグループと比較してキャッシュミス率が大きいグループが見つかったら、キャッシュミス回数のプロファイリングデータを採取する。関数毎のキャッシュミス回数による分類を行うことにより、キャッシュミスがどのノードのどの関数の実行時に多く発生しているのかを特定できる。

## 【0057】

代表的な性能指標であるCPI（1命令の実行に要するCPUクロックサイクル数）の値が悪いグループが見つかり、原因となりうる他の性能項目が見当たらなければ、実行時間のプロファイリングデータを採取する。関数毎の実行時間による分類を行うことにより、正常ノード群と比べて実行時間がかかっているノードと関数を特定できる。

## 【0058】

図6は、データ分類処理の流れを示す図である。性能データ解析部113が、クラスタシステムを構成する各ノードで必要な性能データ91, 92, ..., 9nを採取し、性能データテーブル301に纏める（ステップS21）。性能データ分類部116は、各ノードで採取されたデータをまとめ、単位の異なる性能データを比較するために正規化を行い、正規化データテーブル302を生成する（ステップS22）。なお、図6では最大値と最小値で正規化している（最大値が1、最小値が0となるように各データの値を変更する計算を行う）。性能データ分類部116は、統計処理ツールに正規化したデータを入力し、ノード間の距離行列を求め、距離行列303を生成する（ステップS23）。性能データ分類部116は、この距離行列と分類するグループ数をツールに入力し、階層的グループ化の分類結果304を得る（ステップS24）。

## 【0059】

分類の手法としては、グループを形成する核となる対象を設定してグループを構成していく、K-means法などの非階層的手法もあり、この手法を用いてもよい。K-means法による分類ツールを利用する場合には、距離行列とグループ数を入力として与える。

## 【0060】

このように分類された各グループの性能値を比較することで、障害のノードを含むグループを特定することができる。

以下、クラスタシステムを構成するノードから取得する性能データが、関数の実行時間を示すプロファイリングデータの場合、CPUの性能データの場合、およびOSから得られるシステムレベルの性能データの場合に分けて、グループ分けしたときの性能値の比較例を具体的に説明する。

## 【0061】

まず、プロファイリングデータを用いてノードを分類する例について説明する。ある一定期間内あるいはあるアプリケーションの実行時に、各ノードで実行された関数の内訳を調べることは、ユーザにとってわかりやすく、チューニングすべき箇所を特定しやすい。

## 【0062】

まず、性能データ解析部113において、各ノード210, 220, 230, ... から関数の実行時間などを採取する。

図7は、1ノード分のプロファイリングデータの例を示す図である。プロファイリングデータ21の1行目には、実行時間の種類別内訳とCPU内訳が示されている。「Total: 119788」は、プロファイリングデータ21を採取する間の総演算時間を示している。「OS

10

20

30

40

50

:72850」は、OSの関数の処理に要した時間を示している。「USER:46927」は、ユーザプロセスで実行された関数の処理に要した時間を示している。「CPU0:59889」、「CPU1:59888」は、2つ搭載されたCPUそれぞれの演算時間を示している。

#### 【0063】

2行目には、OSレベル関数(カーネル関数)とユーザ(USER)レベル関数(ユーザ定義関数)の実行比率が示されている。3行目以降には、関数情報が示されている。関数情報は、「Total」、「ratio」、「CPU0」、「CPU1」、および「関数名」で表される。「Total」は、対応する関数の処理に要した実行時間である。「ratio」は、対応する関数の処理に振り分けられた処理時間の割合である。「CPU0」、「CPU1」は、個々のCPUによって対応する関数の処理を行った時間である。「関数名」は、実行された関数の名称である。このようなプロファイリングデータ21がノード分収集される。 10

#### 【0064】

性能データ解析部113は、採取した性能データを解析し、全関数あるいはカーネル関数やユーザ定義関数等の関数タイプ毎に関数の実行時間でソートする。図7の例では、全関数がソートされた結果が出力されている。そこで、性能データ解析部113は、たとえば、性能データを、カーネル関数とユーザ定義関数に分けて集計する。

#### 【0065】

性能データ解析部113は、ソートの結果、上位から一定個数の関数のデータのみを性能データ分類部116への入力とする。通常、関数レベルではかなりの数の関数が実行されるが、そのすべてが均等に実行されるということではなく、一部の関数の実行に時間を要することが多い。そのため、本実施の形態では、実行時間に占める割合が多い関数のみを分類の対象とする。 20

#### 【0066】

クラスタ性能値計算部111では、クラスタシステムの性能値が算出される。クラスタシステムの性能値としては、各性能データの全ノードの平均値や全ノードの合計値が考えられる。クラスタ性能値計算部111で算出されたクラスタシステムの性能データは、クラスタ性能値出力部112で出力される。これにより、ユーザは、クラスタシステムの動作概要を把握することができる。

#### 【0067】

性能値を算出する対象性能データについては、分類に使用するデフォルト値を用いてもよいし、分類条件指定部114でユーザが指定したものを用いてもよい。 30

図8は、プロファイリングデータの概要の表示例を示す図である。プロファイリングデータ概要表示画面30には、8ノードで構成されているクラスタシステムのプロファイリングデータの概要として、各ノードの種類別実行時間比率、クラスタ全体の実行時間による関数ランキングなどを表示している。これにより、クラスタシステムの大まかな動作が把握できる。

#### 【0068】

分類条件指定部114では、性能データの正規化方法、ノードを分類する際のグループの数、および分類に使用する関数タイプと関数の数等に関し、ユーザからの指定入力を受け付ける。なお、予め注目すべき関数やノードがわかっている場合も考えられる。その場合、直接関数名やノード名で指定可能にしてもよい。 40

#### 【0069】

分類条件指定部114が受け付けた正規化方法に基づいて、性能データ分類部116が性能データの測定値を正規化する。例えば、各測定値を、クラスタシステムを構成するノード群の中の最大値/最小値あるいは平均値/標準偏差で正規化する。なお、関数の実行時間の場合には単位が同じなので、正規化は必ずしも必要ではない。

#### 【0070】

また、性能データに基づくノードの分類は、例えば、異常なノード群を発見することを目的として実行される。その場合、適当と思われるグループ数は2つである。すなわち、2グループに分類して、グループ間に性能差がなければ異常なノードは存在しないと考え 50

られる。

【0071】

ノードのグループ化では、性能の似たノードを同一グループに纏める。そのため、指定グループ数に分類した結果、グループ間に性能差があり、各グループ内の分散が大きくなければ、グループ数は適切と考えられる。

【0072】

グループ内の分散が大きい（グループ内のノードの性能に共通性がない）場合には、さらにグループ数を増やして分類する。グループ間にあまり性能差がない（近い性能のノードが異なるグループに属している）場合にはグループ数を減らして分類する。

【0073】

また、管理ノードと計算ノードに分かれている場合やマシンに性能の差があるノードにより構成されている場合など、予めノードの動作パターンがわかっている場合がある。この場合には、そのパターンに従って予想されるグループ数を指定するとよい。

【0074】

分類の結果、グループ分けが正しくない場合かつグループ内の分散が大きい場合には、さらにグループ数を増やして分類する。このように分類を繰り返すことにより、クラスタシステムの挙動が明らかになってくる。

【0075】

分類対象選択部115は、性能データ解析部113で解析された性能データのうち、分類条件指定部114を用いてユーザから指定された条件に合致する性能データのみを選択する。条件の指定がなければ、分類対象選択部115は、デフォルトで設定された値を使用する。デフォルトの値としては、例えば、グループ数は2、関数は全関数の上位10関数、ノードは全ノードが考えられる。

【0076】

性能データ分類部116は、階層的にグループを構成していく階層的グループ化の手法を用いてノードを分類する。分類手法を提供するツールはすでに存在しており、既存の分類ツールを使用する。

【0077】

具体的には、性能データ分類部116は、指定された性能データに対して指定された正規化を行い、正規化したデータ間の距離を計算し、距離行列を求める。性能データ分類部116は、この距離行列と分類するグループ数、クラスタ間の距離の定義の仕方を分類ツールの入力として与え、指定された数のグループにノードを分類する。クラスタ間の距離の定義方法としては、最短距離法、最長距離法、などがある。距離の定義方法についても、ユーザが指定できるようにしてもよい。

【0078】

グループ性能値計算部118は、分類された各グループの性能値を算出する。グループの性能値としては、グループに属するノード群の性能データの平均値、グループを代表するノードの値、グループに属する全ノードの合計値などが考えられる。代表ノードは各性能データにおいて平均的な値を持つノードとすることが考えられる。

【0079】

ノードのグループ分けとグループ性能値計算部118において算出された各グループの性能値は、分類結果出力部120で出力される。このとき、グラフ作成部119により、性能データ毎にグループ間の比較が可能なグラフを作成し、作成されたグラフを出力できる。グラフを出力すると、分類結果がわかりやすくなる。

【0080】

また、分類結果としては、性能データ毎に各グループの値を並べるだけでもよいが、ノード数が一番多いグループの性能値を基準として、その他のグループの性能値の割合をグラフで表してもよい。これにより、グループ間の比較が容易となる。

【0081】

図9は、分類結果の表示例を示す図である。図9の分類結果表示画面40には、図8で

10

20

30

40

50

使用したプロファイリングデータをもとに、平均値 / 標準偏差で正規化し、全関数の上位 10 関数の実行時間を対象として 2 グループに分類した結果が表示されている。

【 0 0 8 2 】

グループ表示部 4 0 a には、各グループのグループ名、各グループのノード数、および各グループに属するノード名が表示されている。図 9 の例では、7 つのノードと 1 つのノードに分類されている。

【 0 0 8 3 】

グラフ表示ボタン 4 0 b が押されることで、分散状態表示画面 5 0 ( 図 1 0 参照 ) が表示される。また、平行座標表示の色分け指示用のチェックボックス 4 0 d によって、グラフ内での色分けの基準を指定することができる。例えば、「グループ」が選択された場合、グループ毎に異なる色で表示される。

10

【 0 0 8 4 】

再表示ボタン 4 0 c が押下されると、グラフ 4 0 f の再表示が行われる。また、エラーバーの種別選択用のチェックボックス 4 0 e によって、エラーバー 4 0 g として、標準偏差を表示するのか、最大値・最小値を表示するのかを選択することができる。

【 0 0 8 5 】

図 9 に示すグラフ 4 0 f は各グループの性能値の平均値を示す棒グラフである。また、グラフ 4 0 f の、グループの分散状況として標準偏差範囲が黒のエラーバー 4 0 g で表されている。エラーバー 4 0 g は、棒フラグに重畳表示されている。なお、図 9 の例では、Group2 には、1 つのノードしか属していないため、標準偏差範囲は存在しない。

20

【 0 0 8 6 】

この例では、アイドルの状況 ( 1 : cpu\_\_idle ) が異なるが、それほど大きな違いはないということがわかる。

グループ選択部 1 2 1 は、ユーザらの操作入力に応じて、分類結果出力部 1 2 0 により出力された結果から 1 つのグループを選択する。グループ選択部 1 2 1 によりグループが選択されると、選択されたグループにおける性能値の分散状況を表すグラフがグループ分散状況出力部 1 2 2 で作成され、出力される。グループの分散状況を表すグラフとしては、グループに属するノード群の性能値を棒グラフ、ノード数が多い場合には度数分布表をヒストグラムで表すことも有効と考えられる。このグラフによりグループ内の分散状況を把握し、分散が大きい場合にはグループ数を増やして再度分類してみるとよい。

30

【 0 0 8 7 】

また、クラスタ分散状況出力部 1 1 7 により各ノードの性能値の分散状況を見ることもできる。クラスタ分散状況出力部 1 1 7 は、性能データ分類部 1 1 6 により分類されたグループ毎に色分けしたグラフを作成し出力する。グラフとしては、各性能値を正規化してグラフ化する平行座標表示、各性能データの分布を表す散布図が有効と考えられる。

【 0 0 8 8 】

図 1 0 は、分散状況の表示例を示す図である。分散状況表示画面 5 0 には、図 9 で分類したデータに対する平行座標表示例が示されている。図 1 0 において、0 が平均、± 1 が標準偏差範囲を表す。実行時間が大きい関数から順に表示している。例えば、グループ 1 に分類されたノードの折れ線 5 1 を見ると、他のノードと比較して、1 番目と 7 番目の関数の実行時間が少なく、4 番目から 6 番目、8 番目から 1 0 番目の関数の実行時間が多いことがわかる。

40

【 0 0 8 9 】

次に、CPU から得られる性能データを用いてノードを分類する例について説明する。性能データ取得部 2 1 2 では、実行命令数やキャッシュミス回数等の CPU から得られる性能データを採取する。

【 0 0 9 0 】

性能データ解析部 1 1 3 では、採取した性能データを解析し、実行命令数に占めるキャッシュミス回数の割合を示すキャッシュミス率等の性能値を算出する。

図 1 1 は、CPU の性能データの例を示す。性能データ 6 0 には、何らかのイベントを

50

カウントした実際の数だけでなく、割合を示す数値として得られるものもある。ノード毎のイベントの発生割合に関して、算出済みであれば再度計算する必要はない。ただし、グループ内での統計値を出す際には、各ノードの値を集計することが必要である。

**【0091】**

クラスタ性能値計算部111では、クラスタシステムの性能値として、例えば各性能データの全ノードの平均値や全ノードの合計値を計算する。なお、CPUから得られるデータには割合(%)で表されるものもあるため、その場合、平均値が使用される。

**【0092】**

クラスタ性能値出力部112では、CPUの性能を表す代表的な性能項目であるCPIやCPU使用率などの平均値を表示する。

分類条件指定部114では、性能データの正規化方法、ノードを分類するグループの数、分類に使用する性能項目をユーザが指定できるようにする。予め注目すべきノードがわかっている場合も考えられるので、対象ノードの指定を可能にしてもよい。測定値の正規化には、クラスタシステムを構成するノード群の中の最大値/最小値あるいは平均値/標準偏差での正規化がある。CPUから得られるデータでは性能項目によって値の単位やスケールが異なるため、正規化が必要である。

**【0093】**

分類対象選択部115では、分類条件指定部によりユーザから指定された条件に合致する性能データのみを選択する。指定がなければ、デフォルトの値を使用する。デフォルトの値としては、グループ数は2、ノードは全ノード、性能項目としては、CPI、CPU使用率、バス使用率、実行命令数に対する分岐命令数の割合を示す分岐率、分岐命令に対する分岐予測ミス率、命令数に対する命令TLB(I-TLB)ミス発生率、命令数に対するデータTLB(D-TLB)ミス発生率、キャッシュミス率、2次キャッシュミス率などが挙げられる。性能項目はCPUの種類によって採取可能なものが異なる場合もあり、デフォルト値も性能項目の異なるCPU毎に用意する。

**【0094】**

グループ性能値計算部118で算出するグループの性能値としては、一般的にはグループに属するノード群の性能データの平均値、グループを代表するノードの値、グループに属する全ノードの合計値などが考えられるが、CPUから得られるデータでは性能項目によって割合(%)で表すものもあるため、グループに属する全ノードの合計値は適さない。

**【0095】**

図12は、CPUの性能データに基づく分類結果表示画面を示す図である。この分類結果表示画面41は、8ノードから構成されるクラスタシステムで採取されたCPUの性能データのうち、11項目をもとに2つのグループに分類した結果の表示例である。

**【0096】**

この例では、4ノードずつに分けられ、Group2のCPU使用率がほとんど0であることからGroup2に属するノード群では何も実行されていなかったことがわかる。なお、分類結果表示画面41では、グループ分散状況を最大値/最小値の範囲を示すエラーバー41aで表現している。

**【0097】**

なお、図12の例では、D-TLBミス発生率(図中、「D-TLB」と表記)のグループ内分散が大きい(平均値0.02、最小値0.05、最大値0.57)が小さいため、気にすることはない。また、任意のグラフがマウスカーソル41bで指し示されたとき、グループの値の概要(平均値、最小値、最大値、標準偏差など)がツールチップ41cで表示される。これにより、詳細な内容がわかりやすくなる。

**【0098】**

図13は、CPUの性能データに基づいて3グループに分類したときの分類結果表示画面を示す図である。この例は、図12に示したデータを3グループに分類したものである。図13の分類結果表示画面42では、何も実行されていないグループから1つのノード

10

20

30

40

50

が分かれており、このノードがD-TLBミス発生率の分散を大きくしていたことがわかる。

#### 【0099】

図12、図13を比較すると、処理が実行されているノード群とそうでないノード群を識別するのであれば、2グループに分ければよいことがわかる。また、ある性能データの分散が大きい時に、その原因となるノードを知りたいときには、分類するグループ数を増やせばよいことがわかる。

#### 【0100】

図14は、散布状況を示す図である。この散布状況は、クラスタ分散状況出力部117により作成される。この例は、平均値/標準偏差で正規化した2つの性能項目の値から1つの散布図を作成し、分類に使用した各性能項目の散布図を、散布状況表示画面70内に並べたものである。散布図内では、例えば、グループ毎に色分けした点で各ノードの性能データをプロットする。これによって、グループの傾向を見ることができる。例えば、赤でプロットされた点がCPIの低い値に集中していれば、そのグループのCPIが小さいことがわかる。

#### 【0101】

次に、システムレベルの性能データ(オペレーティングシステムの動作状況を示す)を用いてノードを分類する場合の例について説明する。以下、最初に説明した例と異なる部分についてのみ説明する。

#### 【0102】

性能データ取得部212では、メモリ使用量、入出力されたデータ量等のシステムレベルの性能データを採取する。これらのデータはOSで提供されているコマンドや既存ツール等を使用すれば、収集可能である。

#### 【0103】

通常これらのデータは一定時間毎に採取されるので、性能データ解析部113では、採取した性能データを解析し、採取時間内の合計値あるいは単位時間の平均値を性能値として算出する。

#### 【0104】

図15は、性能データの例を示す図である。性能データ80の先頭行がヘッダ、2行目以降が各日時における採取データである。この例では1秒間隔にデータを採取している。

採取されている性能データは、ノード全体のCPU使用率内訳、ノードに搭載されたCPU毎のCPU使用率内訳、ディスク毎の入出力量、メモリ使用量など様々なデータを採取可能である。

#### 【0105】

クラスタ性能値計算部111では、クラスタシステムの性能値として各性能データの全ノードの平均値や全ノードの合計値が考えられるが、システムレベルのデータは割合(%)で表すものもあるため、平均値を使用する。

#### 【0106】

また、クラスタ性能値出力部112では、代表的な性能項目のクラスタの平均値を表示する。その際には、CPU、HDDなどのように1ノードに複数存在する資源については各資源の平均値と全体の平均値を表示し、ユーザが確認できるようにする。ディスクに対する入出力量のように合計値を求めることができるものはディスク全体の合計値とクラスタの合計値を表示してもよい。

#### 【0107】

この際、分類条件指定部114では、性能データの正規化方法、ノードを分類するグループの数、分類に使用する性能項目をユーザが指定できるようにする。予め注目すべきノードがわかっている場合も考えられるので、対象ノードの指定を可能にしてもよい。

#### 【0108】

測定値の正規化には、クラスタシステムを構成するノード群の中の最大値/最小値あるいは平均値/標準偏差での正規化がある。システムレベルのデータでは性能項目によって

10

20

30

40

50

値の単位やスケールが異なるため、正規化が必要である。

【0109】

分類対象選択部115では、分類条件指定部114によりユーザから指定された条件に合致する性能データのみを選択する。指定がなければ、デフォルトの値を使用する。デフォルトの値としては、グループ数は2、ノードは全ノード、性能項目としては、CPU使用率、スワップ量、入出力回数、入出力量、メモリ使用量、ネットワークの送受信量などが挙げられる。CPU使用率については、user、system、アイドル状態(idle)、io待ち(iowait)の実行割合を使用する。

【0110】

また、1ノードに複数のCPUが搭載されている場合には各CPUの値あるいはCPU合計の割合を使用する。入出力回数と量については、複数のディスクが接続されている場合には各ディスクの値、全ディスクの平均値、あるいは合計値を使用することができる。複数のネットワークカードが搭載されている場合も同様である。

【0111】

通常は採取時間全体が対象であるが、注目すべき時間がわかっている場合には、その時間を指定可能とする。各ノードでの採取開始時刻がわかっている場合には、時間指定において、開始からの相対時間だけでなく、時刻による絶対時間の指定を可能にすると、ノード間で測定開始時刻がずれている場合にも対処できる。

【0112】

グループ性能値計算部118で算出するグループの性能値としては、一般的にはグループに属するノード群の性能データの平均値、グループを代表するノードの値、グループに属する全ノードの合計値などが考えられる。ただし、システムレベルのデータでは性能項目によっては割合(%)で表すものもある。そのため、グループに属する全ノードの合計値は適さない。

【0113】

図16は、システムレベルの性能データによる分類結果の表示例を示す図である。この例では、CPUから得られるデータの例と同じクラスタシステムに対して同じアプリケーションを動かした際に採取した性能データを使用している。分類結果表示画面43では、2グループに分割されており、図12と同様にグループ分けされる。Group2はやはりUSERとSYSTEMの割合が低く、動作していないことがわかる。

【0114】

以上のように、本発明の実施の形態では、各ノードの動作をシステム情報、CPUから得られる情報、およびプロファイリング情報等を基に数値化し、これを各ノードの特徴と位置付け比較を行うことにより、様々な性能指標を使用して定量的に分析できる。

【0115】

例えば、性能データ分類部116において、実行時に採取した各ノードの性能データを統計処理してノードを任意の数のグループに分類し、分類されたグループ間の性能比較を行う。これにより、見るべき情報を大幅に削減でき、効率的な評価を行うことができる。

【0116】

また、クラスタシステム200を構成する各ノードが同一の動作をする場合には、分類したグループ間の性能差は小さいはずであり、性能差が大きい場合は異常な動作をするノード群が存在していることになる。また、各ノードの動作が予め予測できる場合には、予測できる数に分類し、グループ分けの結果を調べることにより、異常な挙動をするノード群を見つけることができる。

【0117】

数値で表現できる各ノードのマシン情報(CPU数、キャッシュサイズ、等)を取得し、実行時に測定した性能データだけでなく、マシン情報も分類に使用することにより、マシンの構成が異なることによる性能差を発見可能となる。

【0118】

さらに、クラスタ性能値計算部111により複数のクラスタシステムから採取された性

10

20

30

40

50

能データを対象として分析を行うことにより、クラスタ間の性能比較が行える。

以上のように、本発明によれば、クラスタの挙動の理解と性能分析が容易になり、異常な挙動をするノード群を自動的に抽出することができる。

【0119】

なお、上記の処理機能は、コンピュータによって実現することができる。その場合、管理サーバや各ノードが有すべき機能の処理内容を記述したプログラムが提供される。そのプログラムをコンピュータで実行することにより、上記処理機能がコンピュータ上で実現される。処理内容を記述したプログラムは、コンピュータで読み取り可能な記録媒体に記録しておくことができる。コンピュータで読み取り可能な記録媒体としては、磁気記録装置、光ディスク、光磁気記録媒体、半導体メモリなどがある。磁気記録装置には、ハードディスク装置（HDD）、フレキシブルディスク（FD）、磁気テープなどがある。光ディスクには、DVD（Digital Versatile Disc）、DVD-RAM（Random Access Memory）、CD-ROM（Compact Disc Read Only Memory）、CD-R（Recordable）/RW（ReWritable）などがある。光磁気記録媒体には、MO（Magneto-Optical disk）などがある。

10

【0120】

プログラムを流通させる場合には、例えば、そのプログラムが記録されたDVD、CD-ROMなどの可搬型記録媒体が販売される。また、プログラムをサーバコンピュータの記憶装置に格納しておき、ネットワークを介して、サーバコンピュータから他のコンピュータにそのプログラムを転送することもできる。

20

【0121】

プログラムを実行するコンピュータは、例えば、可搬型記録媒体に記録されたプログラムもしくはサーバコンピュータから転送されたプログラムを、自己の記憶装置に格納する。そして、コンピュータは、自己の記憶装置からプログラムを読み取り、プログラムに従った処理を実行する。なお、コンピュータは、可搬型記録媒体から直接プログラムを読み取り、そのプログラムに従った処理を実行することもできる。また、コンピュータは、サーバコンピュータからプログラムが転送される毎に、逐次、受け取ったプログラムに従った処理を実行することもできる。

【0122】

なお、本発明は、上述の実施の形態にのみ限定されるものではなく、本発明の要旨を逸脱しない範囲内において種々の変更を加えることができる。

30

【図面の簡単な説明】

【0123】

【図1】本実施の形態の概略を示す図である。

【図2】本実施の形態のシステム構成例を示す図である。

【図3】本実施の形態に用いる管理サーバのハードウェア構成例を示す図である。

【図4】性能分析を行うための機能を示すブロック図である。

【図5】性能分析手順を示すフローチャートである。

【図6】データ分類処理の流れを示す図である。

【図7】1ノード分のプロファイリングデータの例を示す図である。

40

【図8】プロファイリングデータの概要の表示例を示す図である。

【図9】分類結果の表示例を示す図である。

【図10】分散状況の表示例を示す図である。

【図11】CPUの性能データの例を示す図である。

【図12】CPUの性能データに基づく分類結果表示画面を示す図である。

【図13】CPUの性能データに基づいて3グループに分類したときの分類結果表示画面を示す図である。

【図14】散布状況を示す図である。

【図15】性能データの例を示す図である。

【図16】システムレベルの性能データによる分類結果の表示例を示す図である。

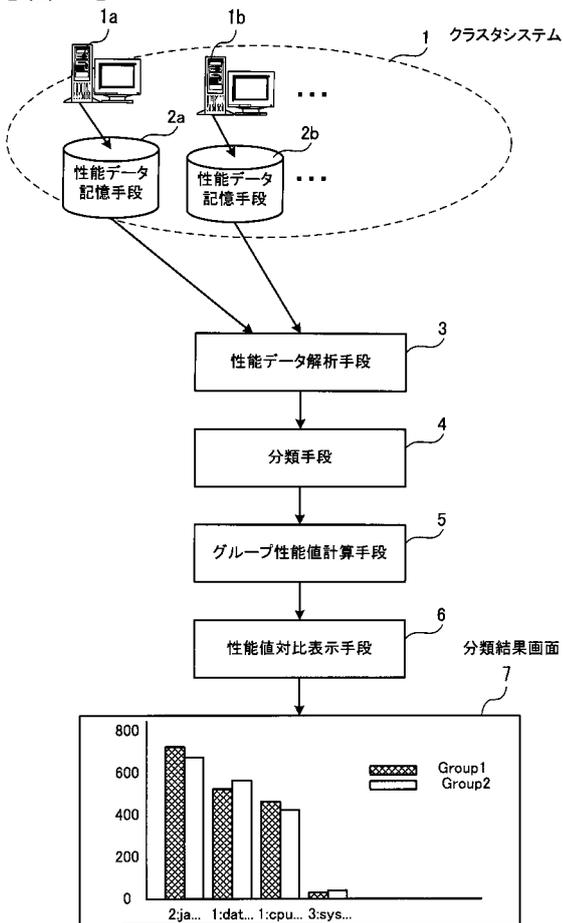
50

【符号の説明】

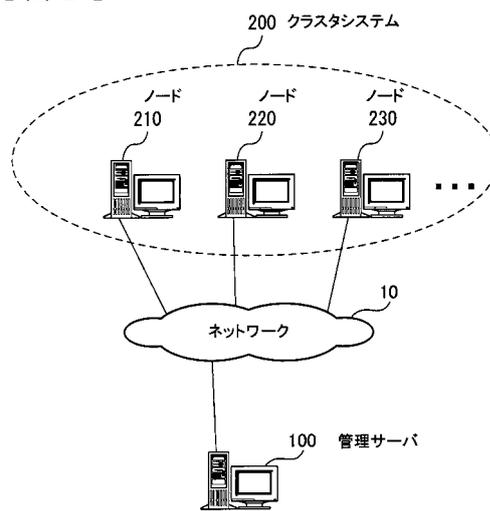
【0124】

- 1 クラスタシステム
- 1 a , 1 b , . . . ノード
- 2 a , 2 b , . . . 性能データ記憶手段
- 3 性能データ解析手段
- 4 分類手段
- 5 グループ性能値計算手段
- 6 性能値対比表示手段
- 7 性能値表示画面

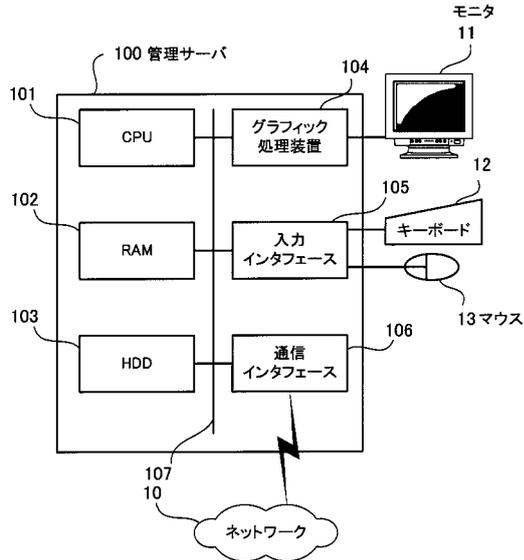
【図1】



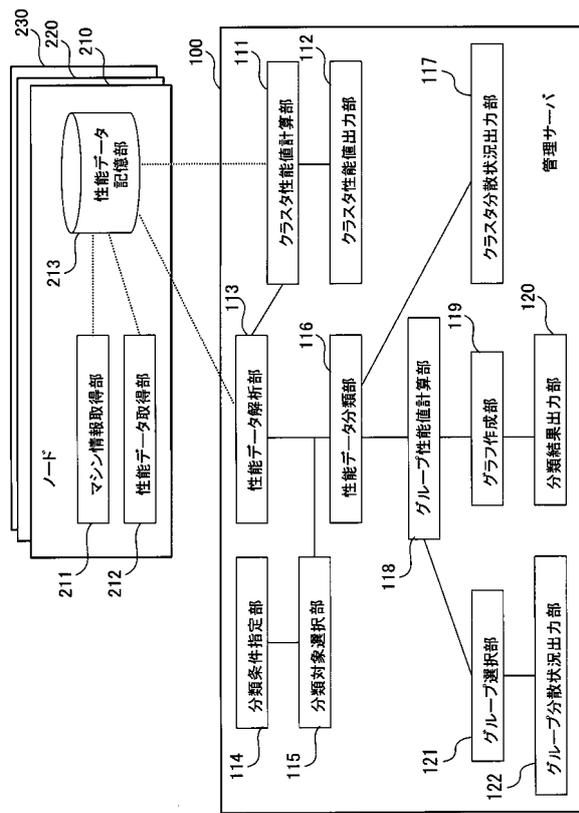
【図2】



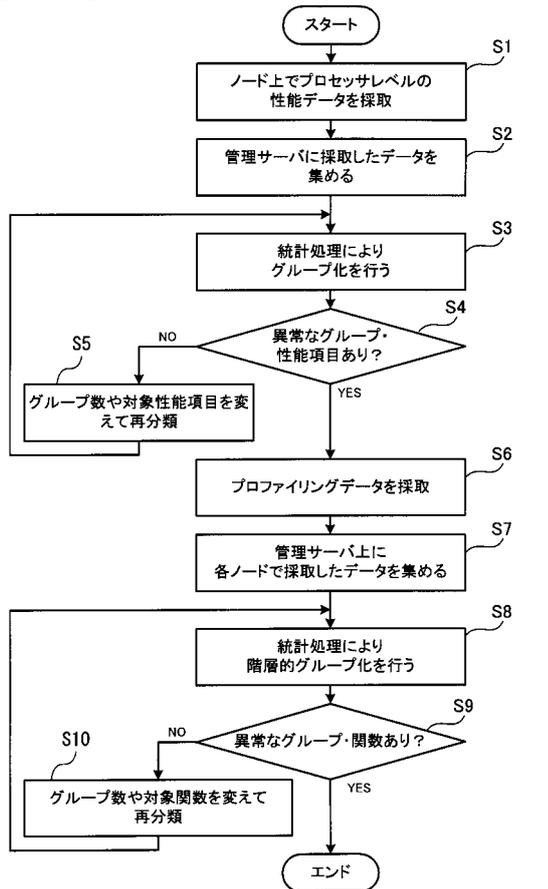
【図3】



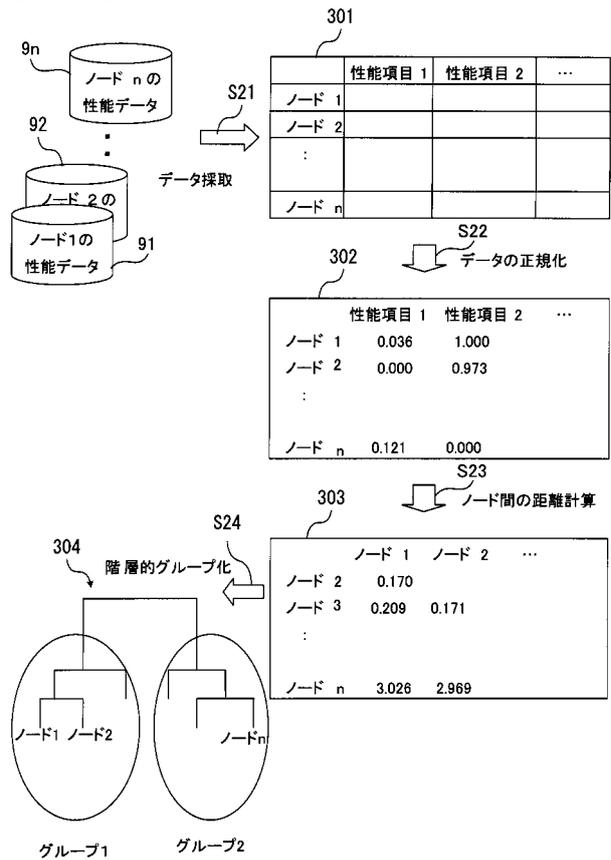
【図4】



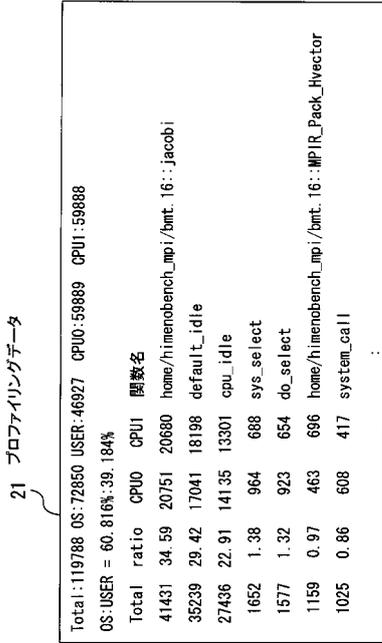
【図5】



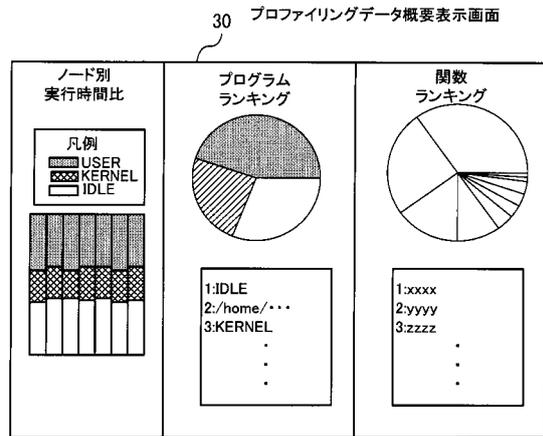
【図6】



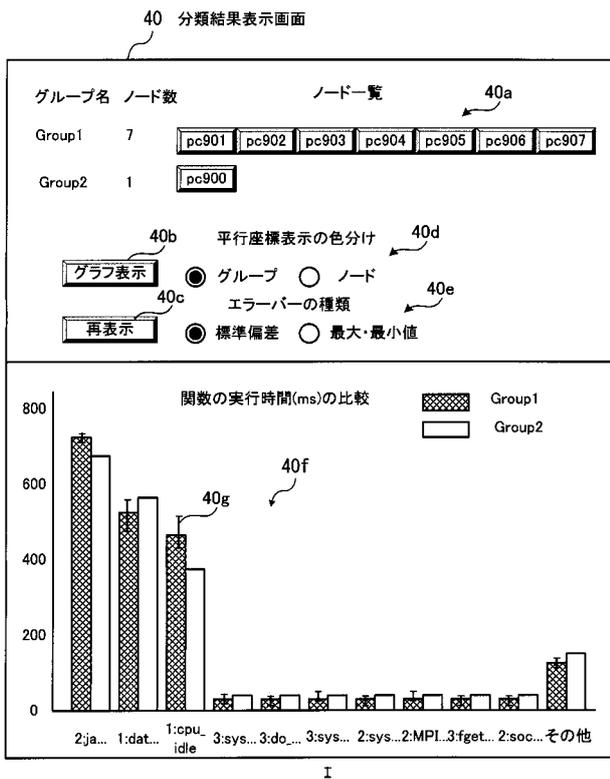
【 図 7 】



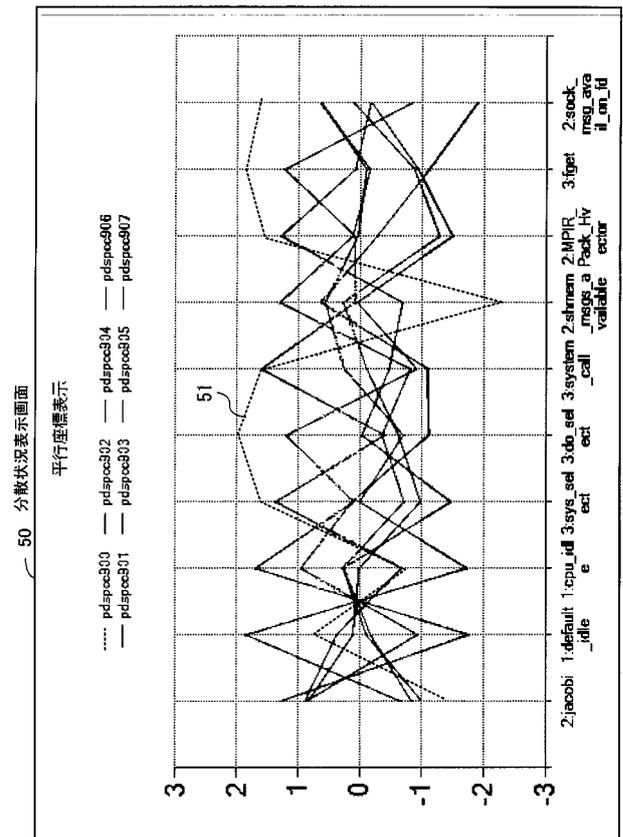
【 図 8 】



【 図 9 】



【 図 10 】



【 図 1 1 】

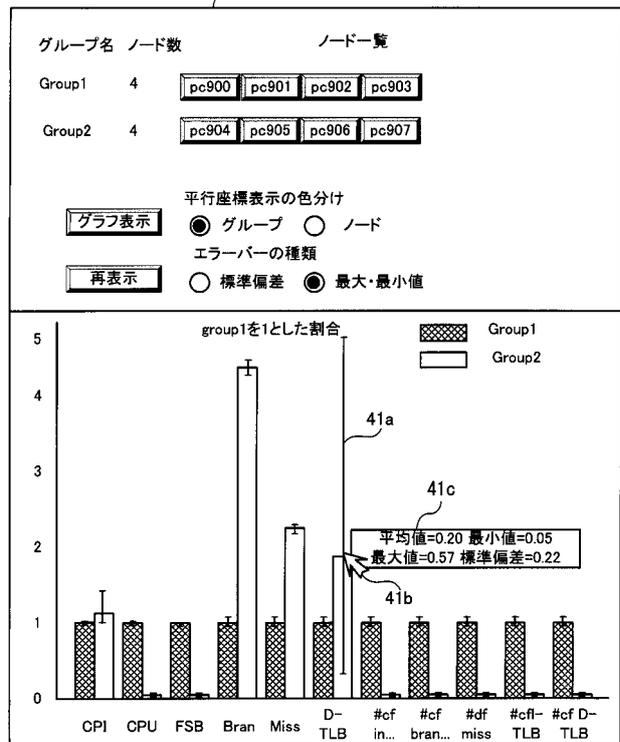
60

CPUID:total [Per tr]  
(Instruction)

	USER	OS	TOTAL
# of unhaltd clks	5,777,165,791	522,485,725	6,299,651,516
# of instructions	2,026,983,629	120,040,332	2,147,023,961
CPI	2.85	4.35	2.93
IPC	0.35	0.23	0.34
# of branch instructions	88,750,507	23,649,464	112,399,971
Bran. / Inst. [%]	4.38%	19.70%	5.24%
# of taken branches	85,802,666	13,559,384	99,362,050
Taken bran. / Insts. [%]	4.23%	11.30%	4.63%
Taken bran. / Bran. [%]	96.68%	57.33%	88.40%

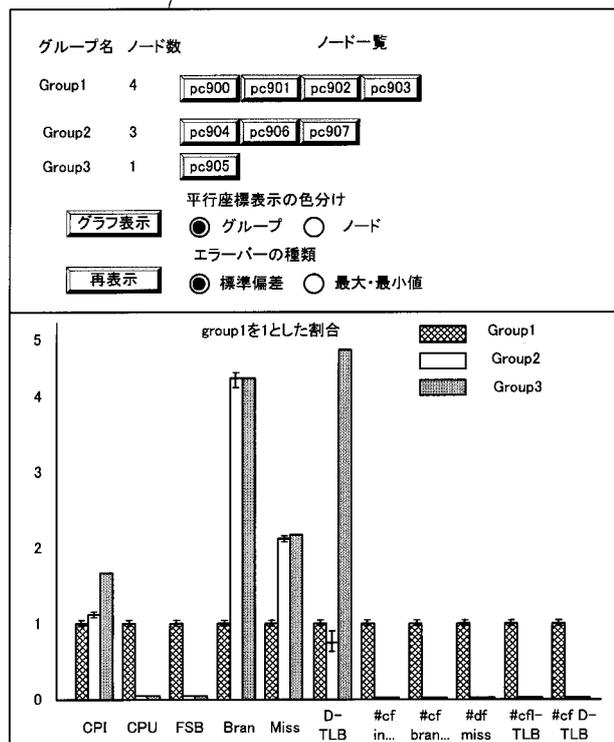
【 図 1 2 】

41 分類結果表示画面

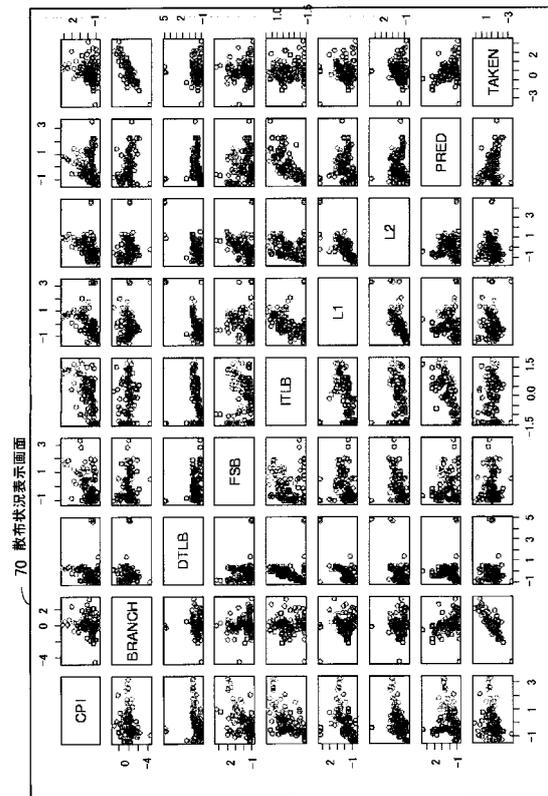


【 図 1 3 】

42 分類結果表示画面



【 図 1 4 】



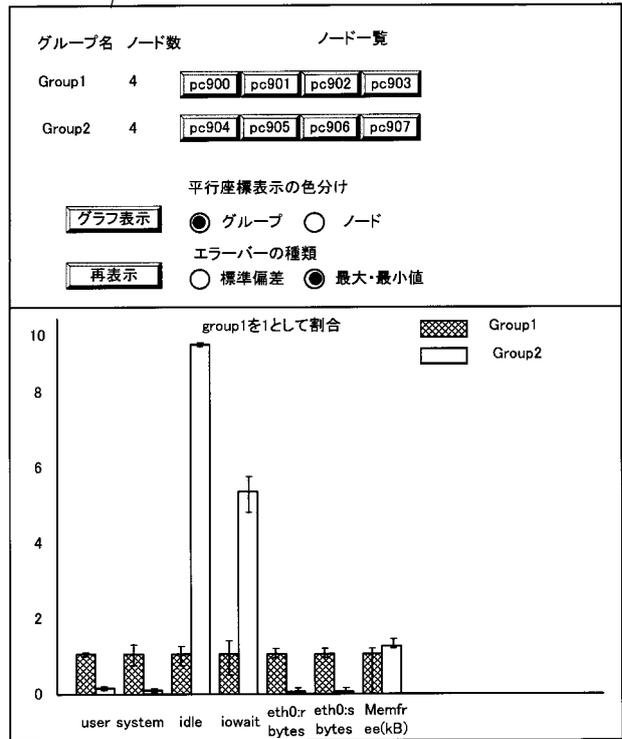
【 図 1 5 】

80 性能データ

date	user	nice	system	idle	iowait	Irq	■ ■ ■ ■
06:16.0	73	0	25	49	46	0	
06:17.0	177	0	5	11	0	1	
06:18.0	171	0	9	16	0	0	
06:19.0	176	0	8	16	0	0	
			⋮				

【 図 1 6 】

43 分類結果表示画面



---

フロントページの続き

(72)発明者 松本 和宏

神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 久門 耕一

神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

Fターム(参考) 5B042 GA11 HH20 MA08 MA14 MC23 MC29 NN04 NN09