

(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2007-128122  
(P2007-128122A)

(43) 公開日 平成19年5月24日(2007.5.24)

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

審査請求 未請求 請求項の数 7 O L (全 13 頁)

(21) 出願番号	特願2005-317867 (P2005-317867)	(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
(22) 出願日	平成17年11月1日(2005.11.1)	(74) 代理人	100100310 弁理士 井上 学
		(72) 発明者	半場 寛之 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア事業部 内
		(72) 発明者	東園 良二 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア事業部 内

最終頁に続く

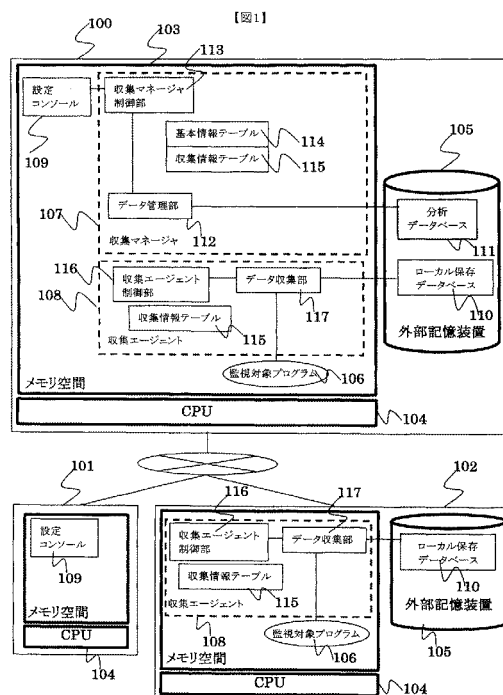
(54) 【発明の名称】 稼働性能データ収集開始時刻決定方法

(57) 【要約】

【課題】 稼働状況の異なる複数の計算機において、監視対象プログラムの稼働性能データを同じタイミングで取得するための、各計算機ごとの収集開始時刻決定方法を提供する。

【解決手段】 各計算機ごとに、監視の対象となる項目の稼働性能データ収集予定時刻と実際の収集時刻との時刻差を用いて、次回の収集時の収集開始時刻を、前記時刻差だけ前倒しする。また、収集予定時刻と実際の収集時刻との時刻差の算出や、収集開始時刻への反映などを、あらかじめ定めた見直し間隔にしたがって定期的に行う。

【選択図】 図1



## 【特許請求の範囲】

## 【請求項 1】

ネットワークで接続された複数の計算機上において、前記複数の計算機上で同一のタイミングで計算機やプログラムの稼働性能データを収集するために各計算機ごとの稼働性能データ収集開始時刻を決定する方法であって、過去の稼働性能データを収集した際の、あらかじめ設定した稼働性能データ収集時刻と実際の稼働性能データ収集時刻との時間差の平均値から、次回の稼働性能データ収集開始時刻を算出するステップを備えたことを特徴とする稼働性能データ収集開始時刻決定方法。

## 【請求項 2】

請求項 1 の稼働性能データ収集開始時刻決定方法において、各計算機上で複数の項目を連続して収集する場合に、全計算機で同一のタイミングで収集させる項目を、各計算機ごとに個別に設定した収集タイミング同期ポイントとして設定するステップを備えたことを特徴とする稼働性能データ収集開始時刻決定方法。

10

## 【請求項 3】

請求項 1 または 2 の稼働性能データ収集開始時刻決定方法において、あらかじめ定められた見直し間隔の度に、あらかじめ設定された分析データ保存期間の範囲に合致する過去の稼働性能データ収集時刻を元に稼働性能データ収集開始時刻を再計算することを特徴とする稼働性能データ収集開始時刻決定方法。

## 【請求項 4】

請求項 1 から 3 の稼働性能データ収集開始時刻決定方法において、あらかじめ定められた見直し間隔の度に、あらかじめ設定された分析データ保存期間の範囲に合致する過去の稼働性能データ収集時刻を元に稼働性能データ収集時刻を再計算する場合において、あらかじめ設定されている収集間隔の期間内に稼働性能データを収集することができない場合は、収集間隔に対してあらかじめ設定されている収集間隔増加幅だけ拡張し、全ての計算機において、稼働性能データを同じタイミングで収集できるように収集間隔を長く設定することを特徴とする稼働性能データ収集開始時刻決定方法。

20

## 【請求項 5】

請求項 1 から 4 の稼働性能データ収集開始時刻決定方法において、稼働性能データ収集開始時刻決定のために使用する過去の稼働性能データ収集時刻が、あらかじめ設定された見直し間隔が経過するたびに各計算機から収集され、かつ各計算機から収集された過去の稼働性能データおよび過去の稼働性能データ収集時刻のうち、あらかじめ設定された分析データ保存期間より古いものについては削除し、分析データ保存期間の範囲に合致する過去の稼働性能データおよび過去の稼働性能データ収集時刻のみを用いて稼働性能データ収集開始時刻を決定することを特徴とする稼働性能データ収集開始時刻決定方法。

30

## 【請求項 6】

請求項 1 から 5 の稼働性能データ収集開始時刻決定方法であって、あらかじめ設定された見直し間隔の度に、算出された各計算機ごとの稼働性能データ収集開始時刻を各計算機へ配信するステップを備えたことを特徴とする稼働性能データ収集開始時刻決定方法。

## 【請求項 7】

請求項 1 から 6 のデータ収集開始時刻決定方法であって、あらかじめ定められた見直し間隔のたびに配信される稼働性能データ収集開始時刻に従って、定期的に稼働性能データを収集し、過去の稼働性能データおよび過去の稼働性能データ収集時刻を各計算機上に蓄積するステップを備えたことを特徴とする稼働性能データ収集開始時刻決定方法。

40

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

本発明は、分散環境下の複数の計算機上において、ある同一のタイミングで稼働性能データを収集するための稼働性能データ収集開始時刻決定方法に関する。

## 【背景技術】

## 【0002】

50

近年、業務システムを構成する計算機は、Webサーバやアプリケーションサーバ、データベースサーバなど複数分散して存在し、これらはネットワークを介して接続されている。このようなシステムにおいては、分散する各サーバが個々に時刻管理を行っているため、複数のサーバの負荷状況やイベント発生時刻を単純に照らし合わせただけでは、システムに起こっている現象を正確に把握できない問題がある。例えば、サービスの性能劣化の原因調査を行う場合、ある同一のタイミングにおいて、各計算機上のアプリケーションがどのような状況であったのかを知ることが必要である。とくに、Webサービスにおいては、アプリケーションサーバでのトランザクション数やデータベースサーバの検索効率、サービスを利用しているユーザ数やそのときのネットワークトラフィックなど、性能に影響する要素が分散しており、統一した時刻のもとで稼働性能データを付き合わせる事が非常に困難である。また、負荷の高いサーバから負荷の低いサーバへ処理を振り分けるような場合においても、同じタイミングにおける各サーバの負荷を正確に知る事が出来れば、より最適な負荷分散が実現でき、システムリソースをより有効活用できる。

10

20

30

40

50

**【0003】**

分散する各サーバの稼働状況を知るための手段としては、稼働性能データを収集する技術(特許文献1)や、収集した稼働性能データの相関関係を分析する手法(特許文献2)等が用いられている。また、これらの技術に基づく問題の原因調査の精度を高めるためには、上記に述べたように業務システムを構成する各計算機で収集された稼働性能データが、同一のタイミングで収集されることが重要となる。このような課題に対する技術として、稼働性能データの収集開始時刻を一括して定義および配布し、収集する技術(特許文献3)が用いられている。

**【0004】**

上記のような従来技術においては、各サーバの負荷状況により稼働性能データ収集処理の遅延が生じてしまうため、同一のタイミングにおける稼働性能データが揃わずに分析精度が低下するという問題がある。ただし、全ての計算機で、ある同じタイミングの稼働性能データを収集するためには、各計算機上の特性を加味して個々に最適な稼働性能データ収集開始時刻を決定する必要があるが、各計算機上のアプリケーションの稼働状況などは時々刻々と変化していく。このため、稼働性能データ収集開始時刻は定期的な見直しによって最適なものを設定する必要がある。

**【0005】**

上記の一連の問題を解決するために、システムを構成する各サーバの負荷を考慮した上で、稼働性能データ収集開始時刻を周期的に決定する技術が求められる。

**【0006】**

【特許文献1】特開2001-27990号公報

【特許文献2】特開2004-348640号公報

【特許文献3】特開2004-355061号公報

【発明の開示】

【発明が解決しようとする課題】

**【0007】**

本発明の目的は、各計算機上のオペレーティングシステムやアプリケーションの過去の稼働性能データ収集時間をもとに、全ての計算機上で、ある同じタイミングで稼働性能データを取得するための、各計算機ごとの最適な稼働性能データ収集開始時刻を決定し、これを定期的に見直すことにより、時々刻々と変化していく環境下において、常に最適な収集時刻で稼働性能データを収集することを目的とする。

【課題を解決するための手段】

**【0008】**

上記目的を達成するため、本発明は、各計算機上の時計の時刻差だけでなく、各計算機上のオペレーティングシステムやアプリケーションの稼働性能データと、過去に稼働性能データを収集するのに要した時間の実績をもとに、あらかじめ与えられるデータ収集開始時刻を、各計算機上で最適な値に設定する。また、各計算機上のオペレーティングシステ

ムやアプリケーションの稼働性能データと、過去に稼働性能データを収集するのに要した時間の実績を定期的に更新し、分析データ保存期間より古い日付の稼働性能データを削除することにより、時々刻々と変化する環境下において、各計算機ごとの稼働性能データ収集開始時刻を常に最適な状態に保つ。

【発明の効果】

【0009】

本発明によれば、従来技術で考慮されていなかった各計算機ごとの稼働性能データに応じて、各計算機ごとに個別の最適な収集開始時刻を決定することが可能となる。これにより、システム中のすべての計算機でより正確なタイミングで稼働情報を取得でき、その結果、分散する複数サーバで構成されるシステムにおける障害調査や、負荷の高いサーバから処理を振りかえるためのサーバ決定など、システム全体の管理精度が向上するという利点がある。

10

【発明を実施するための最良の形態】

【0010】

以下、本発明を実施するための最良の形態を図面に基づいて詳細に説明する。

【0011】

図1は、本発明の実施例で用いる、ネットワークに接続された複数の計算機からなるシステム構成をしめしたものである。図1では、複数の計算機100、101、102がネットワークを介して接続されて、互いに通信可能な状態である。

【0012】

実施例で用いる計算機の構成について計算機1(100)を例に説明する。計算機1(100)は、プログラムや処理データを格納するメモリ空間(103)と、メモリ空間に記憶されたプログラムを実行するCPU(104)、参照頻度の低いデータを格納する外部記憶装置(105)とを有している。メモリ空間(103)には、オペレーティングシステムやアプリケーションなどの監視対象プログラム(106)と、システム中の各計算機の収集開始時刻を決定する収集マネージャ(107)と、収集マネージャ(107)が決定した収集開始時刻にもとづいて監視対象プログラム(106)から稼働性能データを収集する収集エージェント(108)と、収集エージェント(108)に収集させる項目や収集間隔などの基本情報を設定するための設定コンソール(109)が記憶されている。また、外部記憶装置(105)には、収集エージェントが取得した稼働性能データを蓄積

20

30

【0013】

収集マネージャ(107)は、分析データベース(111)に蓄積されているデータを管理するデータ管理部(112)と、各計算機ごとの収集開始時刻を決定し、収集エージェントへ配布する収集マネージャ制御部(113)で構成される。また、収集マネージャ(107)は、システム利用者が設定コンソール(109)から入力した情報を登録する基本情報テーブル(114)と、各計算機ごとの監視項目や収集開始時刻を登録して配布

40

【0014】

収集エージェント(108)は、収集マネージャ(107)と通信により情報をやり取りする収集エージェント制御部(116)と、収集マネージャ(107)から配布された各計算機ごとの収集情報テーブル(115)の内容に従って監視対象プログラム(106)から稼働情報を収集してローカル保存データベース(110)へ格納するデータ収集部(117)で構成される。

【0015】

収集マネージャ(107)、設定コンソール(109)、分析データベース(111)は、本発明により稼働情報の収集開始時刻を制御する複数の計算機のいずれか1台上に少

50

なくとも1必要である。収集エージェント(108)、ローカル保存データベース(110)は、本発明により稼働情報の収集開始時刻を制御する複数の計算機の全てに1ずつ必要である。つまり、システムを構成する計算機が1台のみである場合は、計算機1(100)のように構成し、システムを構成する計算機が2台以上存在する場合には、計算機2(101)のように設定コンソール(109)のみの構成や、計算機3(102)のように収集エージェント(108)とローカル保存データベース(110)のみの構成をとる計算機が存在してもよい。

#### 【0016】

図2は、基本情報テーブルである。これは、本発明により各計算機ごとの稼働性能データ収集開始時刻を決定するために最低限必要な複数の情報からなるテーブルであり、各値はシステム利用者によって登録される。収集開始タイミング(200)には、全計算機で同期させたい稼働性能データ収集のタイミングである。収集間隔(210)は、各計算機上で稼働性能データを収集する間隔である。収集タイミング見直し間隔(220)は、収集マネージャ(107)が決定する各計算機ごとの収集開始時刻を、過去の稼働性能データおよび過去の稼働性能データ収集時刻に基づいて定期的に見直しを行う間隔である。分析データ保存期間(230)は、収集タイミング見直し間隔(220)が経過した際に、分析データベース(111)に保存されている稼働性能データのうち、有効なものとして扱う期間の長さである。たとえば、分析データ保存期間(230)が30日に設定されている場合、分析データベース(111)に蓄積される稼働性能データが過去30日分に達するまではすべてを蓄積し、31日目の稼働性能データが収集された場合は、分析データベース(111)に格納されている1日目の稼働性能データが削除され、分析データベース(111)中には常に過去30日分の稼働性能データのみが蓄積される。収集間隔増加幅(240)には、収集開始タイミング(200)で指定した時刻ではどうしても収集できない計算機が存在する場合の対処方法として、収集間隔(210)を自動的に大きく設定する場合の増え幅である。例えば、収集間隔(210)が60秒、収集間隔増加幅(250)が60秒であった場合、最初の収集間隔(210)の60秒で稼働性能データが収集できないことが判明した場合、収集間隔増加幅(240)の60秒を足した次120秒を新たな収集間隔(210)として収集開始時刻を再計算する。

#### 【0017】

図3は、収集情報テーブル(115)の一例である。このテーブルは収集マネージャ(107)および収集エージェント(108)が存在する各計算機ごとに存在し、収集マネージャ(102)で決定した各計算機ごとの収集開始時刻などが登録されている。各計算機の収集エージェント(108)は収集マネージャ(107)より配布された収集情報テーブル(115)にしたがって、監視対象プログラム(108)の稼働性能データを収集する。収集項目(300)は、前記の収集情報テーブル(115)が存在する各計算機ごとに収集する稼働性能データのリストである。図3の例では、計算機1(100)について業務プログラムのCPU使用率(301)、ディスクI/O(302)、業務プログラムに接続中のユーザ数(303)が登録順に収集される。収集項目(300)の値はユーザによってあらかじめ入力されているものとする。収集タイミング同期ポイント(310)は、複数の項目を収集する場合に、どの項目の収集時刻で計算機間の同期をとるのかの基準となる項目である。たとえば、上記301~303の項目を収集させる場合、収集タイミング同期ポイント(310)をディスクI/O(302)としておくと、全計算機で同時にデータを収集するタイミングにおいて、計算機1(100)ではディスクI/O(302)が収集されるように調整を行う。この場合、CPU使用率(301)およびディスクI/O(302)の稼働性能データ収集にかかる時間分だけ収集開始時刻を前にずらすことで調整される。各計算機ごとの収集タイミング同期ポイント(310)の値はユーザによってあらかじめ入力されているものとし、入力がない場合は、収集項目(300)にある項目全ての収集が完了する時点を基準とする。収集開始時刻(320)は、収集マネージャ(107)によって決定した各サーバごとの収集開始時刻のリストである。リストされる収集開始時刻の数は、図2の基本情報テーブル(114)の収集タイミング見直し間隔(220)

の期間中に、収集間隔(210)で収集可能なタイミングの数になる。つまり、収集タイミング見直し間隔が24時間、収集間隔が60分の場合、収集開始時刻(320)には24項目がリストされる。

#### 【0018】

図3の例では、収集タイミング同期ポイント310で示したディスクI/Oのデータ収集後に収集タイミングをそろえるように調整されているが、収集タイミング同期ポイント310で示した項目のデータ収集開始時でそろえることも可能である。図3の計算機1(100)でディスクI/O(302)のデータ収集開始時にタイミングをそろえるように調整を行う場合、CPU使用率(301)の稼働性能データ収集にかかる時間分だけ収集開始時刻を前にずらすことで調整される。収集タイミングポイント310で示した項目のデータ収集開始前でタイミングをそろえるか収集開始後でタイミングをそろえるかは、図3には図示しないフラグ等で識別すればよい。

10

#### 【0019】

図4は、ローカル保存データベース(110)あるいは分析データベース(111)のテーブル構成の例である。計算機名(400)は、稼働性能データが収集された計算機名である。収集項目(410)は、収集情報テーブルの収集項目(300)で設定された稼働性能データの項目名である。収集開始時刻(420)は、収集項目(410)の各項目の収集が開始された時刻である。終了時刻(430)は、収集項目(410)の各項目の収集が終了した時刻である。性能値(440)は、収集項目(410)の各項目の収集された実際の値である。

20

#### 【0020】

図5は、本発明の一実施例に係る収集マネージャ制御部(113)の全体的な動作を示すフローチャートである。収集マネージャ制御部(113)は、前回に各計算機の収集開始時刻の再計算を行ってから経過した時間を定期的に確認し、基本情報テーブル(114)の収集タイミング見直し間隔(220)の時間だけ経過するまで待機する(500)。収集タイミング見直し間隔(220)の時間だけ経過すると、前回の収集開始時刻決定から現在までに、収集エージェント(108)が取得している稼働性能データの差分を各計算機の収集エージェント制御部(116)に要求し、分析データベース(111)に格納する(501)。稼働性能データの差分が分析データベース(111)に格納されたならば、分析データベース(111)の稼働性能データをもとに各計算機ごとの収集開始時刻を決定し(502)、更新された各計算機ごとの収集情報テーブル(115)を各計算機に配布する(503)。配布後は、500のステップへ帰って、収集タイミング見直し間隔の時刻になるまで待機する(504)。

30

#### 【0021】

図6は、収集マネージャ制御部(113)において、収集開始時刻の再計算を行うステップ(502)の動作を示すフローチャートである。収集開始時刻の再計算は、収集タイミング見直し間隔(220)中のすべての収集タイミングにおいて収集開始時刻が決定されるまで繰り返される(600)。また、600のステップ内では、収集対象である全計算機の収集開始時刻が決定されるまで繰り返される(601)。601のステップ内では、分析データベースに格納された過去の稼働性能データ収集時刻を、収集間隔ごとに集計して平均値を算出する(602)。この平均値と、あらかじめ設定されている収集開始タイミング200との時刻差を計算し(603)、あらかじめ設定されている収集開始タイミング200から、前記時刻差の分だけ前倒しした値を収集開始時刻とする(604)。

40

#### 【0022】

ここで、502のステップで求められた収集開始時刻が、それより以前の収集における収集時刻より過去の時間になっているかどうかを判定する(605)。判定の結果、今回の収集開始時刻がその前の収集間隔における収集時刻に影響する場合は、基本情報テーブルの収集間隔増加幅(250)に登録されている増加分を、上記影響がなくなる値まで加算し、収集間隔を大きくする(606)。収集間隔を大きくした場合は、これまで決定していた収集開始時刻をクリアして(607)、再度600から始まるステップを行う。9

50

01から905までのステップは、全計算機においてある収集間隔の収集開始時刻が確定するまで繰り返される(608)。また、900から905のステップは、全計算機において、収集タイミング見直し間隔(220)中のすべての収集間隔(210)の収集開始時刻が決定されるまで繰り返される(609)。最終的に各収集間隔において各計算機ごとに決定された収集開始時刻は、各計算機ごとの収集情報テーブルの収集開始時刻(320)に登録され(610)、収集マネージャ制御部(113)によって各計算機上の収集エージェント制御部(116)へと配信される(503)。

#### 【0023】

図7は、データ管理部(112)による分析データベース(111)の管理の動作を示すフローチャートである。データ管理部(112)は、収集マネージャ制御部(113)より通知を受け取ると、各計算機上の収集エージェント制御部(116)に対して、前回の要求から現時点まで新たに蓄積された稼働性能データの差分を送信するよう要求し(700)、収集エージェント(108)から前記差分が到着すると、それを分析データベース(111)へ格納する(701)。格納後に、格納されている稼働性能データのなかで最も古いものの日付と、基本情報テーブル(114)の分析データ保存期間(230)に登録されている情報とを比較し(702)、分析データ保存期間(230)より古い日付の情報については分析データベース(111)から削除する(703)。これによって、分析データベース(111)は常に分析データ保存期間(230)内の稼働性能データに更新されるため、収集開始時刻決定の際の精度が向上する。

10

#### 【0024】

図8は、収集エージェント制御部(116)の動作を示すフローチャートである。収集エージェント制御部(116)は、収集マネージャ制御部(113)からの通知を受け付けるまで待機している(800)。収集マネージャ制御部(113)から、収集情報テーブル(115)が配布された場合は(801)、収集エージェント制御部(116)がローカルにもっている収集情報テーブル(116)の内容を、配布された収集情報テーブル(115)の内容に更新する(802)。更新された場合は、その旨をデータ収集部(117)に通知する(803)。一方、収集マネージャ制御部(113)から取得済みの稼働性能データを要求された場合は(804)、ローカル保存データベース(110)に格納されている稼働性能データのうち、前回の要求から今回の要求の間に新たに蓄積された稼働性能データを抽出する(805)。抽出された稼働性能データは収集マネージャ(107)へ送信される(806)。

20

30

#### 【0025】

図9は、収集エージェントにおけるデータ収集部(117)の動作を示すフローチャートである。データ収集部(117)は、収集エージェント制御部(116)より収集情報テーブル更新の通知があると(900)、現在の収集をいったん停止し、更新された収集情報テーブル(115)より収集開始時刻などの情報を受けとる(901)。そして、収集情報テーブル(115)の収集開始時刻(320)に登録されている時刻かどうかを定期的に確認し(902)、収集開始時刻である場合には、監視対象プログラム(106)より稼働性能データを収集する(903)。収集した稼働性能データは、ローカル保存データベースへ(110)格納される(904)。格納後は、再び収集エージェントの通知を待ちながら、収集情報テーブル(115)の収集開始時刻に従って稼働情報を収集する(905)。収集開始時刻でない場合は、再び収集エージェント制御部(116)の通知を待つ(906)。

40

#### 【0026】

以上が、本発明の第一の実施例についての説明である。

#### 【0027】

次に、本発明の第二の実施例について説明する。上述した実施例においては、計算機上で取得する収集項目(300)は、全て同じ収集間隔(210)で収集していた。これについて、第二の実施例では、収集項目(300)のそれぞれについて異なる間隔で収集を行う場合を想定している。本実施例の説明においては、図10及び図11を新たに用いる

50

。図10は詳細収集情報テーブル、図11は本実施例における収集マネージャの動作を示すフローチャートである。なお、本実施例の説明においては、既に説明した実施例と異なる部分を中心に説明する。

#### 【0028】

図10は、詳細収集情報テーブル(1000)である。詳細収集情報テーブル(1000)は、本実施例において、収集情報テーブル(115)に代えて使用するテーブルである。たとえば、収集項目および収集間隔(1010)の欄にて、従来はすべての収集項目(300)に同じ収集間隔(210)を適用していたが、詳細収集情報テーブル(1000)では、業務プログラムのCPU使用率(1011)については60秒(1014)、ディスクI/O(1012)については300秒(1015)、業務プログラムに接続中のユーザ数(1013)については600秒(1016)といったように個別に設定が可能である。

10

#### 【0029】

図11は、収集マネージャ制御部(113)において、収集開始時刻の再計算を行うステップ(502)を詳細収集情報テーブル(1000)を用いて行う場合の動作を示すフローチャートである。本フローチャートは、601のステップまでは図6のフローチャートと同一である。601のステップ後、分析データベース(111)に格納されている過去の稼働性能データの収集実績から、各収集項目(1011、1012、1013)ごとに、収集に要した時間の平均値を算出する(1100)。次に、収集開始時刻(320)の基準値を設定する(1101)。ここで基準値とは、各計算機の収集実績を考慮する前の時刻である。ここより、1102から1105のステップを、すべての収集項目の数だけ繰り返す。すべての収集項目の数だけ処理されたかどうかを判定し(1102)、処理されていない項目がある場合はそのうち1つを取り出す(1103)。取り出した収集項目が、今回の収集タイミングにおいて収集の対象である場合(1104)は、収集開始時刻から、1100のステップで算出されている該当する収集項目の収集に要した時間の平均値だけ前倒しした時刻を、あらたな収集開始時刻として設定する(1105)。すべての収集項目について1102から1105のステップが実行されることで、必要な収集項目の負荷だけが考慮された、最適な収集開始時刻(320)を決定することができる。収集開始時刻(320)決定後の処理については、図6のフローチャートの605以降のステップと同様である。

20

30

#### 【0030】

なお、収集開始時刻を前倒しする時間が収集間隔より大きくなる場合は、実施例1と同様に収集間隔の幅を大きくする処理を行う。

#### 【0031】

以下では、本発明の第三の実施例について説明する。本実施例においては、計算機上の各収集項目ごとに収集に要した時間を管理することにより、全計算機上で、ある収集項目の収集開始時刻(320)を同一の揃える。本実施例においては、図12を使用する。

#### 【0032】

図12は、各計算機上のそれぞれの収集項目に対して、過去に収集に要した時間の平均値を算出したテーブルである。本発明では、図4にあるように、各計算機のそれぞれの収集項目に対して収集開始時刻(420)および終了時刻(430)を管理しているため、その時刻差から収集に要した時間を算出できる。これを、分析データベース(111)に保存されている稼働性能データについて集計し、平均値としたものが1221から1227の値である。たとえば、全計算機上で、ディスクI/Oの収集開始時刻を同一にしたい場合、計算機1(100)では、業務プログラムのCPU使用率に10秒の時間を要する(1221)ため、収集開始時間が10秒前倒しされる。この場合、ディスクI/O自体の収集時間(1222)や、その後の業務プログラムに接続中のユーザ数の収集時間(1223)については考慮する必要はない。計算機2(101)についても、業務プログラムのCPU使用率に要する時間(1224)だけ前倒しする。一方、計算機3(102)については、ディスクI/Oを最初に収集するため、前倒しは行われぬ。本実施例では

40

50

、収集開始時刻を同一にする例を説明したが、ある収集項目の収集完了時刻を全計算機上で同一させる場合についても、同様に算出可能である。

【0033】

上述した実施形態は、本発明の趣旨を逸脱しない範囲で適宜変更や組み合わせ可能である。

【図面の簡単な説明】

【0034】

【図1】本発明の実施例における、ネットワークに接続された計算機の構成を示す図である。

【図2】基本情報テーブルの一例を示す図である。

10

【図3】収集情報テーブルの一例を示す図である。

【図4】分析データベースあるいはローカル保存データベースのテーブル構造を示す図である。

【図5】収集マネージャ制御部の動作を示すフローチャートである。

【図6】収集マネージャ制御部における、収集開始時刻決定の動作を示すフローチャートである。

【図7】データ管理部の動作を示すフローチャートである。

【図8】収集エージェント制御部の動作を示すフローチャートである。

【図9】データ収集部の動作を示すフローチャートである。

【図10】詳細収集情報テーブルの一例を示す図である。

20

【図11】収集マネージャ制御部における、収集開始時刻決定を、詳細収集情報テーブルを用いて行う場合の動作を示すフローチャートである。

【図12】各計算機上の収集項目に対して、過去に収集に要した時間の平均値を算出した結果を保持するテーブルの一例を示す図である。

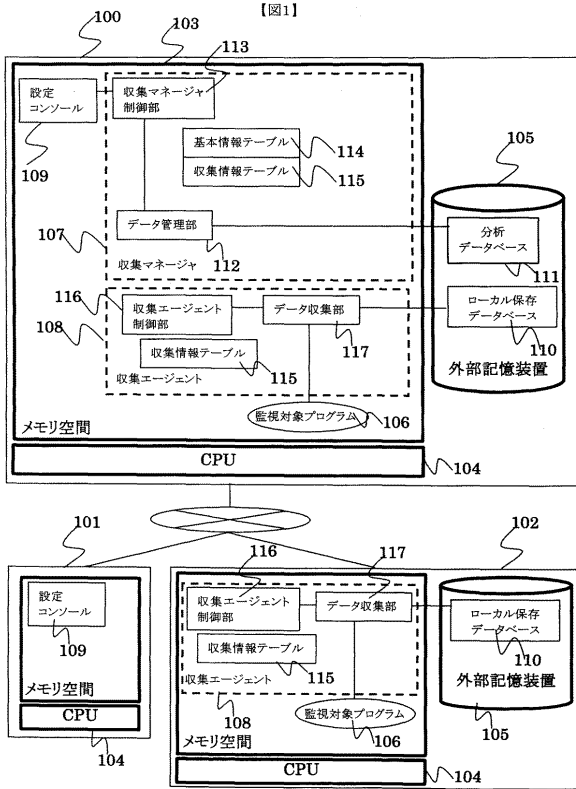
【符号の説明】

【0035】

100 ... 計算機1、101 ... 計算機2、102 ... 計算機3、103 ... メモリ空間、104 ... CPU、105 ... 外部記憶装置、106 ... 監視対象プログラム、107 ... 収集マネージャ、108 ... 収集エージェント、109 ... 設定コンソール、110 ... ローカル保存データベース、111 ... 分析データベース、112 ... データ管理部、113 ... 収集マネージャ制御部、114 ... 基本情報テーブル、115 ... 収集情報テーブル、116 ... 収集エージェント制御部、117 ... データ収集部。

30

【図1】



【図2】

【図2】

200	収集開始タイミング	毎分00秒
210	収集間隔	60秒
220	収集タイミング見直し間隔	24時間
230	分析データ保存期間	30日
240	収集間隔増加幅	60秒

【図3】

【図3】

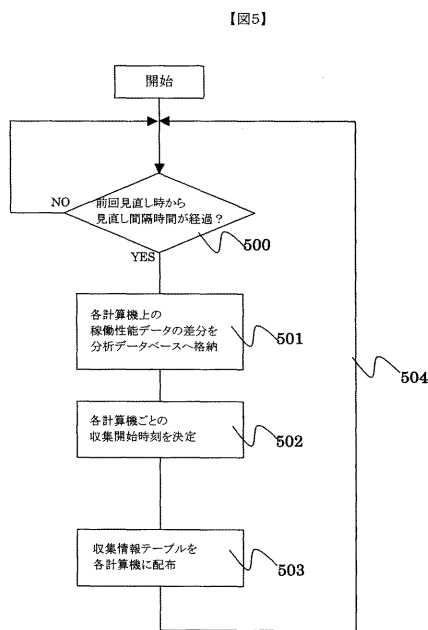
300	計算機1 収集項目	業務プログラムのCPU使用率	301	
		ディスクI/O		302
		業務プログラムに接続中のユーザ数		
310	収集タイミング同期ポイント	ディスクI/O	303	
320	収集開始時刻	23:59:30		
		0:58:50		
		1:59:10		
		⋮		

【図4】

【図4】

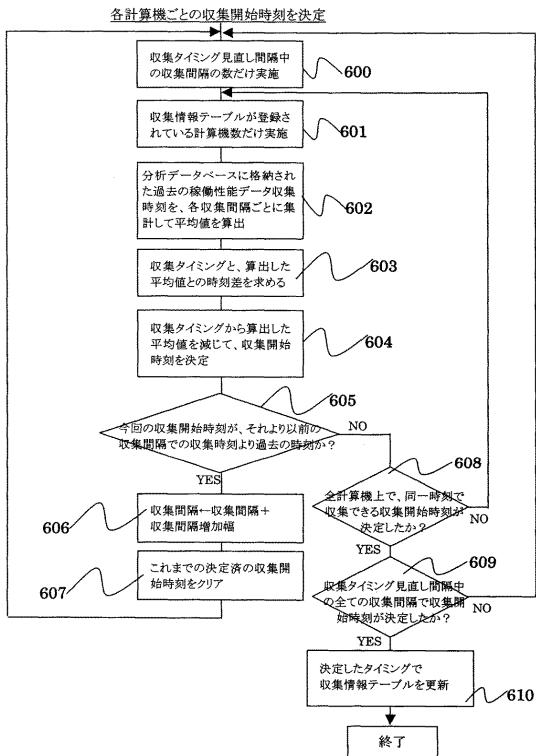
400	410	420	430	440
計算機名	収集項目	収集開始時刻	終了時刻	性能値
計算機1	業務プログラムのCPU使用率	09:59:20	09:59:50	3%
	ディスクI/O	09:59:50	10:00:00	10ms
	業務プログラムに接続中のユーザ数	10:00:00	10:00:20	3ユーザ
計算機1				
計算機2				

【図5】



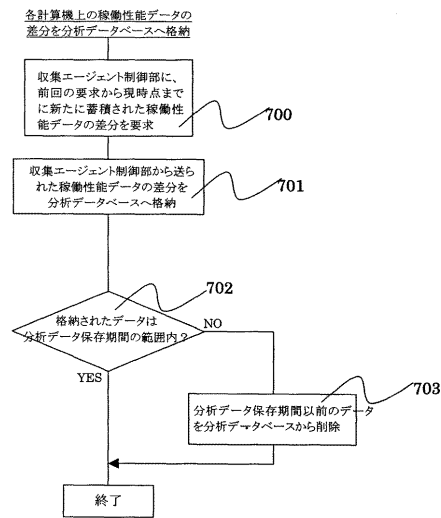
【 図 6 】

【図6】



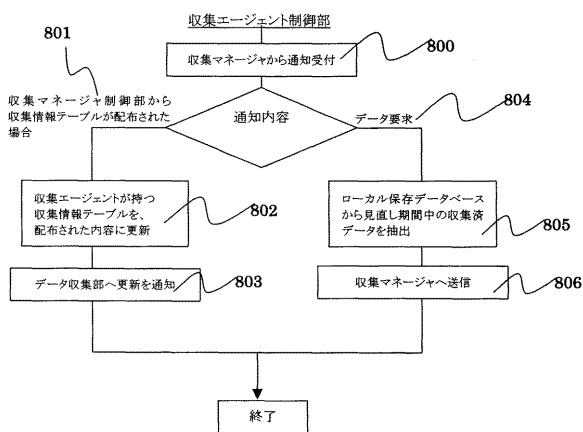
【 図 7 】

【図7】



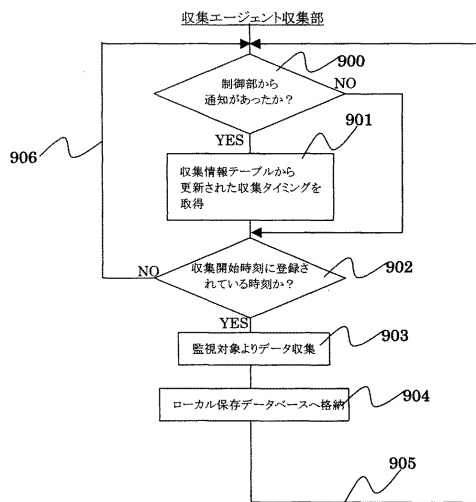
【 図 8 】

【図8】

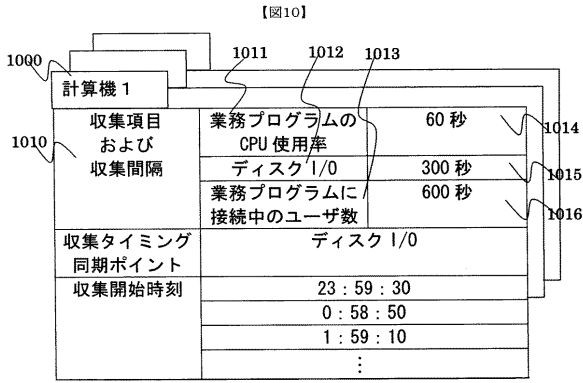


【 図 9 】

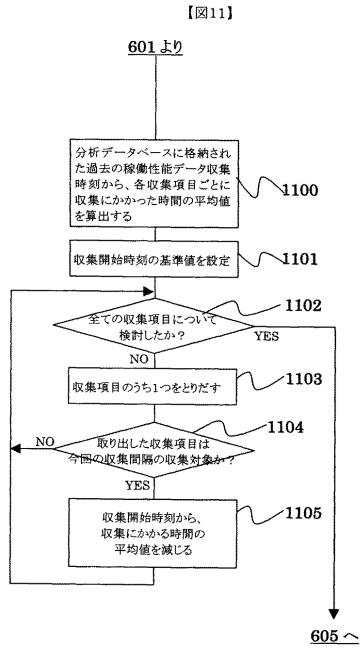
【図9】



【図10】



【図11】



【図12】

【図12】

計算機名	収集項目	平均収集時間
計算機 1	業務プログラムの CPU 使用率	10 秒
	ディスク I/O	15 秒
	業務プログラムに接続中のユーザ数	20 秒
計算機 2	業務プログラムの CPU 使用率	8 秒
計算機 3	ディスク I/O	10 秒
	ディスク I/O	25 秒
	業務プログラムに接続中のユーザ数	30 秒

1200 1210 1220 1221 1222 1223 1224 1225 1226 1227

---

フロントページの続き

(72)発明者 毛受 正夫

神奈川県横浜市戸塚区戸塚町5 0 3 0番地 株式会社日立製作所ソフトウェア事業部内

(72)発明者 渡邊 準

神奈川県横浜市戸塚区戸塚町5 0 3 0番地 株式会社日立製作所ソフトウェア事業部内

Fターム(参考) 5B042 GA12 HH20 JJ08 MA08 MA14 MB02 MC21 MC23 MC29 MC31

MC34 MC35