(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第4318643号
(P4318643)

(45)発行日　平成21年8月26日(2009.8.26)　(24)登録日　平成21年6月5日(2009.6.5)

(51)Int.Cl.		F I		
G06F 13/00	(2006.01)	G06F	13/00	351M
G06F 11/22	(2006.01)	G06F	11/22	360E
G06F 11/34	(2006.01)	G06F	11/34	B

請求項の数 3　(全 35 頁)

(21)出願番号　特願2004-564435 (P2004-564435)
(86)(22)出願日　平成14年12月26日 (2002.12.26)
(86)国際出願番号　PCT/JP2002/013719
(87)国際公開番号　WO2004/061681
(87)国際公開日　平成16年7月22日 (2004.7.22)
審査請求日　平成16年11月9日 (2004.11.9)

(73)特許権者　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　毅巖
(72)発明者　高橋　大作
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　吉位　裕貴子
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】運用管理方法、運用管理装置および運用管理プログラム

(57)【特許請求の範囲】

【請求項１】

コンピュータでサーバの運用管理を行うための運用管理方法において、

収集手段が、前記サーバで実行される複数の機能を監視対象要素として、前記監視対象要素で発生したイベント情報を収集し、

生成手段が、複数の前記監視対象要素同士の関連性が定義されたデータベースを参照し、関連性のある前記監視対象要素から出力された前記イベント情報同士をグループ化してイベントグループを生成し、

照合手段が、同一の前記イベントグループ内の複数のイベント情報を組み合わせたイベント情報組み合わせを複数生成し、障害発生時に複数の前記監視対象要素それぞれから出力される前記イベント情報の発生パターンが定義された複数のパターン定義グループと複数の前記イベント情報組み合わせとを照合し、前記イベント情報組み合わせそれぞれに対して、前記イベント情報組み合わせを構成するすべての前記イベント情報を含むパターン定義グループを関連付け、

結果出力手段が、関連付けられた前記イベント情報組み合わせそれぞれに含まれる前記イベント情報の数の合計が多い前記パターン定義グループほど上位に配置した照合結果を出力する、

ことを特徴とする運用管理方法。

【請求項２】

サーバの運用管理を行うための運用管理装置において、

前記サーバで実行される複数の機能を監視対象要素として、前記監視対象要素で発生したイベント情報を収集する収集手段と、
複数の前記監視対象要素同士の関連性が定義されたデータベースを参照し、関連性のある前記監視対象要素から出力された前記イベント情報同士をグループ化してイベントグループを生成する生成手段と、
同一の前記イベントグループ内の複数のイベント情報を組み合わせたイベント情報組み合わせを複数生成し、障害発生時に複数の前記監視対象要素それぞれから出力される前記イベント情報の発生パターンが定義された複数のパターン定義グループと複数の前記イベント情報組み合わせとを照合し、前記イベント情報組み合わせそれぞれに対して、前記イベント情報組み合わせを構成するすべての前記イベント情報を含むパターン定義グループを関連付ける照合手段と、
関連付けられた前記イベント情報組み合わせそれぞれに含まれる前記イベント情報の数の合計が多い前記パターン定義グループほど上位に配置した照合結果を出力する結果出力手段と、
を有することを特徴とする運用管理装置。

【請求項３】

サーバの運用管理を行う処理をコンピュータに実行させる運用管理プログラムにおいて、
前記コンピュータに、
前記サーバで実行される複数の機能を監視対象要素として、前記監視対象要素で発生したイベント情報を収集し、
複数の前記監視対象要素同士の関連性が定義されたデータベースを参照し、関連性のある前記監視対象要素から出力された前記イベント情報同士をグループ化してイベントグループを生成し、
同一の前記イベントグループ内の複数のイベント情報を組み合わせたイベント情報組み合わせを複数生成し、障害発生時に複数の前記監視対象要素それぞれから出力される前記イベント情報の発生パターンが定義された複数のパターン定義グループと複数の前記イベント情報組み合わせとを照合し、前記イベント情報組み合わせそれぞれに対して、前記イベント情報組み合わせを構成するすべての前記イベント情報を含むパターン定義グループを関連付け、
関連付けられた前記イベント情報組み合わせそれぞれに含まれる前記イベント情報の数の合計が多い前記パターン定義グループほど上位に配置した照合結果を出力する、
処理を実行させることを特徴とする運用管理プログラム。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は監視対象装置の運用管理を行うための運用管理方法、運用管理装置および運用管理プログラムに関し、特に監視対象装置の障害対策に有効な運用管理方法、運用管理装置および運用管理プログラムに関する。

【背景技術】

【０００２】

インターネット環境が普及している昨今では、システムの信頼性強化策としてシステムの多重化が行われている。システムを多重化することにより、一部のシステムに障害が発生しても、他のシステムを利用して支障なく運用を継続することができる。

【０００３】

一般に、サーバで障害等のイベントが発生すると、そのサーバから他の装置（たとえば、運用管理サーバ）に対してメッセージが送信される。多重化されたシステムでは、１つの障害が発生すると、障害が発生した機能に関連する他の機能からもエラーメッセージが出力される。そのため、あるサーバで異常が発生した場合、エラーメッセージを出力するサーバは１つとは限らず、関連する他のサーバからもエラーメッセージが出力される。

【０００４】
このように障害箇所とは別のサーバからもエラーメッセージが出力されると、障害箇所の特定が困難となる。従来は、ネットワーク等に精通した技術者が、過去の経験と照らし合わせて、障害箇所を特定していた。そのため、技術者の能力によっては、障害の復旧に長い時間を要する場合があった。このとき、企業内のネットワークが停止すると、その企業の業務遂行に多大な影響を及ぼす。そのため、技術者の経験則に頼らずに、ネットワークを迅速に復旧できることが望まれていた。
【０００５】
そこで、ネットワークの障害箇所と、その障害の発生時に発行される障害通知情報の時系列パターンとの対応関係をデータベースに保持し、実際にネットワークから通知される障害通知情報とデータベースの内容とを照合することで障害箇所を特定する発明が考えられた。これにより、障害箇所を自動的に特定することができ、ネットワークを迅速に復旧させることができる（たとえば、特許文献１参照）。
【０００６】
しかし、特許文献１で対象としている障害は、ネットワーク上の通信障害であるため、サーバ上で実行されるアプリケーション等の障害については考慮されていない。したがって、サーバ上で実行されるアプリケーション、ミドルウェア、ＯＳ(Operating System)等が発生したエラーメッセージを利用した障害箇所の判定までは行われていない。
【０００７】
すなわち、１つの障害に関し、１つのサーバから複数のエラーメッセージが出力される場合、従来の技術では障害箇所を特定することができない。たとえば、サーバ上で実行されていたアプリケーションが停止した場合、アプリケーション自身がエラーメッセージを出力すると共に、そのアプリケーションに関連するミドルウェアやＯＳもエラーメッセージを出力する場合がある。しかも、複数のサーバが連携して動作している場合、障害の発生原因となったアプリケーションが実行されるサーバとは異なるサーバ上のアプリケーションからエラーメッセージが出されることもある。
【０００８】
このように、多機能のコンピュータシステム上で障害が発生すると、１つの障害に対して、複数のサーバ上の様々なアプリケーションからエラーメッセージが出力される。そのため、個別のメッセージを参照しただけでは、エラーの原因やエラーの発生場所を、サーバ内のソフトウェア単位で特定するのが困難である。
【０００９】
さらに、マルチタスク、マルチスレッド等のシステム環境では、メモリ管理等の問題により、個別のアプリケーションでは異常になっていないにも拘わらず、性能が低下したり、使用しているミドルウェアが原因不明でストップしたり等の障害が発生する。このような場合、障害が表面化したソフトウェアとは別の部分に原因が存在し、その原因を特定するのがさらに困難となっている。
【特許文献１】特開２００１－２５７６７７号公報（第１図）
【発明の開示】
【００１０】
本発明はこのような点に鑑みてなされたものであり、障害箇所をソフトウェア単位で特定できる運用管理方法、運用管理装置および運用管理プログラムを提供することを目的とする。
【００１１】
本発明では上記課題を解決するために、運用管理方法が提供される。本発明に係る運用管理方法は、運用管理を行うためのものである。運用管理方法では、以下の処理が行われる。まず、収集手段が、サーバで実行される複数の機能を監視対象要素として、監視対象要素で発生したイベント情報を収集する。次に、生成手段が、複数の監視対象要素同士の関連性が定義されたデータベースを参照し、関連性のある監視対象要素から出力されたイベント情報同士をグループ化してイベントグループを生成する。次に、照合手段が、同一

のイベントグループ内の複数のイベント情報を組み合わせたイベント情報組み合わせを複数生成し、障害発生時に複数の監視対象要素それぞれから出力されるイベント情報の発生パターンが定義された複数のパターン定義グループと複数のイベント情報組み合わせとを照合し、イベント情報組み合わせそれぞれに対して、イベント情報組み合わせを構成するすべてのイベント情報を含むパターン定義グループを関連付ける。そして、結果出力手段が、関連付けられたイベント情報組み合わせそれぞれに含まれるイベント情報の数の合計が多いパターン定義グループほど上位に配置した照合結果を出力する。

【００１３】

本発明の上記および他の目的、特徴および利点は本発明の例として好ましい実施の形態を表す添付の図面と関連した以下の説明により明らかになるであろう。

【発明を実施するための最良の形態】

【００１４】

以下、本発明の実施の形態を図面を参照して説明する。

まず、実施の形態に適用される発明の概要について説明し、その後、実施の形態の具体的な内容を説明する。

【００１５】

図１は、実施の形態に適用される発明の概念図である。図１では、運用管理サーバ１が、本発明に係る運用管理方法を実行する。運用管理サーバ１は、複数のサーバ２～４に接続されている。サーバ２では、アプリケーション２ａ、ミドルウェア２ｂ、およびＯＳ２ｃが実行されている。アプリケーション２ａ、ミドルウェア２ｂ、およびＯＳ２ｃが、運用管理サーバ１による監視対象要素である。他のサーバ３，４においても、同様のソフトウェアが実行されている。

【００１６】

また、運用管理サーバ１には、予めパターン定義グループデータベース（ＤＢ）１ａと障害対策情報データベース（ＤＢ）１ｂとが設けられている。パターン定義グループＤＢ１ａには、障害発生時に複数の監視対象要素それぞれから出力されるイベント情報の発生パターンが定義された複数のパターン定義グループが格納されている。障害対策情報ＤＢ１ｂには、パターン定義グループに予め関連付けられ、障害原因となる監視対象要素を示す障害対策情報が格納されている。障害対策情報には、障害箇所、障害原因、障害に対する対策方法が含まれる。ここで、運用管理サーバ１は、以下の処理を実行する。

【００１７】

運用管理サーバ１は、まず、サーバで実行される複数の機能を監視対象要素として、監視対象要素で発生したイベント情報を収集する（ステップＳＴ１）。次に、運用管理サーバ１は、収集した複数のイベント情報をグループ化してイベントグループを生成する（ステップＳＴ２）。さらに、運用管理サーバ１は、パターン定義グループＤＢ１ａ内の各パターン定義グループとイベントグループとの間でイベント情報の発生パターンを照合する（ステップＳＴ３）。そして、運用管理サーバ１は、イベントグループに対してイベント情報の発生パターンが類似するパターン定義グループに予め関連付けられ、障害原因となる監視対象要素を示す障害対策情報を、障害対策情報ＤＢ１ｂから抽出する（ステップＳＴ４）。ここで、イベント情報の発生パターンが類似する場合とは、たとえば、共通のイベント情報が少なくとも１つ存在する場合である。

【００１８】

このような運用管理方法によれば、監視対象要素で発生したイベント情報が収集、グループ化され、イベントグループが生成される。すると、複数のパターン定義グループとイベントグループとの間でイベント情報の発生パターンが照合される。次に、発生パターンが類似するパターン定義グループに予め関連付けられた障害対策情報が抽出される。

【００１９】

これにより、抽出された障害対策情報に基づいて、障害箇所、障害原因、障害に対する対策方法を、運用管理サーバの管理者が認識することができる。なお、該当するパターン定義グループが発見されなかったイベントグループに関しては、障害解決後、対応するパ

ターン定義グループをパターン定義グループＤＢ１ａに格納し、障害対策情報を障害対策情報ＤＢ１ｂに格納する。これにより、障害の対策方法等が、障害解決を行う作業者に適宜フィードバック（再利用）される。

【００２０】

このような運用管理方法を用いれば、作業者の能力に拘わらず、障害の解決時間を短縮することができる。しかも、過去の障害の内容をパターン定義グループや障害対策情報として蓄積することで、２度目以降の障害に対して、迅速に対応することができる。また、新規の障害を発見し、修復したときは、その内容をパターン定義グループや障害対策情報に反映させておけば、人づてに障害報告を行う必要が無くなる。その結果、人為的な不手際によるトラブルシューティングの長期化を防止することができる。

【００２１】

次に、本実施の形態における障害検出機能（トラブル情報回帰型解決方法）の概念を説明する。

図２は、トラブル情報回帰型解決方法の運用例を示す図である。図２に示す様に複数のサーバ１１，１２と運用管理サーバ２０とがネットワーク１３で接続されている。サーバ１１には、ハード監視、ソフト監視、性能監視、セキュリティ監視、および構成監視等の監視機能が組み込まれている。サーバ１２にも同様の監視機能が組み込まれている。

【００２２】

ハード監視機能は、サーバ内のハードウェアの動作（たとえば、ハードディスク装置の書き込みエラー発生率等）を監視する。ソフト監視機能は、各ソフトウェア（ＯＳ、ミドルウェア、アプリケーション）毎に設けられ、対応するソフトウェアの動作を監視する。性能監視機能は、動作状態（たとえば、メモリの空き容量）を監視する。セキュリティ監視機能は、サーバ１１に対する不正アクセス等を監視する。構成監視機能は、サーバ１１の構成変更等を監視する。

【００２３】

各監視機能は、エラーなどの所定の事象が発生すると、エラーイベントを発生させる。エラーイベントの内容を示すイベント情報がメッセージ３１，３２として、ネットワーク１３を介して運用管理サーバ２０に送られる。メッセージ３１，３２には、たとえば、サーバ１１，１２が発生するログ情報や、構成変更等の情報が含まれる。

【００２４】

運用管理サーバ２０は、イベント抽出テーブル２１、パターン定義グループ２２、対策情報テーブル２３、メッセージ正規化部２４、グループ化部２５、照合部２６、および対策出力部２７を有している。

【００２５】

イベント抽出テーブル２１は、受信したメッセージから必要なイベント情報を抽出するための規則が定義された情報テーブルである。

パターン定義グループ２２は、障害発生時に検出されるべきイベントの組み合わせを定義した情報である。

【００２６】

対策情報テーブル２３は、障害発生時のイベントパターンに応じた障害対策に関する情報が登録された情報テーブルである。

メッセージ正規化部２４は、サーバ１１，１２からのメッセージを受け取り、所定のデータ構造に正規化する。具体的には、メッセージ正規化部２４は、メッセージを受け取ると、イベント抽出テーブル２１を参照し、受信したメッセージに適用すべきイベント情報の抽出規則を取得する。そして、メッセージ正規化部２４は、取得した規則に従って、メッセージからイベント情報を抽出する。

【００２７】

グループ化部２５は、メッセージ正規化部２４で抽出されたイベント情報を、関連性のあるもの同士でグループ化する。たとえば、監視対象要素同士（装置、ＯＳ、ミドルウェア、アプリケーション等）の関連性が予め定義されており、関連する監視対象要素から短

時間（具体的な時間は、予め設定されている）に出力されたメッセージのイベント情報は、互いに関連するものと判断する。グループ化部２５がイベント情報のグループ化を行った結果、イベントグループ３３が生成される。イベントグループ３３は、関連のあるイベント情報の集合である。

【００２８】

照合部２６は、イベントグループ３３とパターン定義グループ２２とを照合し、イベントグループ３３と同一もしくは類似のパターン定義グループ２２を検出する。

対策出力部２７は、対策情報テーブル２３を参照し、検出されたパターン定義グループ２２に対応する対策情報を取得する。そして、対策出力部２７は、取得した対策情報を、運用管理サーバ２０に接続された端末装置等に表示させる。

【００２９】

このような構成のシステムにおいて、サーバ１１，１２から出力され運用管理サーバ２０に入力されたメッセージ３１，３２は、以下のように処理される。

まず、メッセージ３１，３２は、メッセージ正規化部２４により正規化され、イベント情報が生成される。

【００３０】

図３は、メッセージ正規化処理を示す図である。図３の例では、メッセージ３１，３２はシリアルＮｏ．、時刻、ホスト名、およびイベント内容の情報を含んでいる。シリアルＮｏ．は、メッセージを一意に識別するための識別番号である。時間は、イベントの発生時刻である。ホスト名は、メッセージを出力したサーバのネットワーク１３上での名称である。イベント内容は、サーバ内の監視機能が出力したイベントの内容である。

【００３１】

このようなメッセージ３１，３２をメッセージ正規化部２４が受け取ると、メッセージ正規化部２４は、イベント抽出テーブル２１を参照する。

イベント抽出テーブル２１には、ＨＷ（ハードウェア）またはＳＷ（ソフトウェア）名、キーワード、抽出方法の欄が設けられている。各欄の横方向に並べられた情報同士が互いに関連づけられて、メッセージ情報の抽出規則を構成している。ＨＷまたはＳＷ名は、イベントの発生原因となった要素の名称である。キーワードは、イベントの発生原因となる要素を特定するための情報である。抽出方法は、メッセージから必要なエラーコードを抽出するための規則である。

【００３２】

メッセージ正規化部２４は、イベント抽出テーブル２１を参照することで、メッセージ３１，３２からイベント情報３１ａ，３２ａを生成することができる。具体的には、メッセージ３１，３２を受け取ると、メッセージ正規化部２４は、イベント抽出テーブル２１のキーワード欄を参照し、イベント本文に含まれるキーワードを検索する。メッセージ正規化部２４は、検出されたキーワードに対応する抽出方法を参照し、その抽出方法に従って、イベント内容から必要なエラーコードを抽出する。

【００３３】

たとえば、メッセージ３１が入力されると、イベント抽出テーブル２１からキーワード「Kernel」が検出される。そこで、そのキーワードに対応する抽出方法「］の後ろのスペース以降”：”が入るまで」に従って、エラーコード「WARNING 999」が抽出される。

【００３４】

メッセージ正規化部２４は、エラーコードを含むイベント情報３１ａ，３２ａを生成する。イベント情報３１ａ，３２ａには、シリアルＮｏ．、時刻、ホスト名、ＨＷまたはＳＷ名およびエラーコードが含まれる。シリアルＮｏ．、時刻、およびホスト名については、メッセージ３１，３２から抽出される。ＨＷまたはＳＷ名およびエラーコードは、イベント抽出テーブル２１内の検出されたキーワードに関連付けられた情報である。

【００３５】

このようにして、正規化されたイベント情報３１ａ，３２ａが抽出される。

次に、グループ化部２５により、関連するイベント情報がグループ化される。たとえば

、グループ化部２５は、監視対象要素同士の関連性が予め定義された構成管理データを有する。

【００３６】

図４は、構成管理データの構造例を示す図である。図４に示す構成管理データでは、監視対象要素が階層構造で管理されている。上層からシステム５１，５２、ルータ５３，５４、サーバ５５～５９およびストレージ６０、ミドルウェア６１～６３、アプリケーション６４～６６の順となっている。

【００３７】

システム５１，５２は、たとえば、顧客毎のネットワークシステムを示す。ルータ５３，５４は、システム５１，５２に関連付けられており、システム５１，５２にアクセスするためのパケットを中継するルータの識別情報を示している。サーバ５５～５９は、ルータ５３，５４に関連付けられており、システム５１，５２内に構築されたサーバの識別情報を示している。ストレージ６０は、ルータ５４に関連付けられており、システム５２内のストレージデバイスの識別情報を示している。ミドルウェア６１～６３は、サーバ５５に関連付けられており、サーバ５５に実装されたミドルウェアの識別情報を示している。アプリケーション６４～６６は、ミドルウェア６１に関連付けられており、そのミドルウェア６１で管理されているアプリケーションソフトウェアの識別情報を示している。

【００３８】

このように、監視対象要素間の関連づけをグループ化部２５に定義しておくことで、グループ化部２５は、関連性のあるイベント情報を判別することができる。すなわち、グループ化部２５は、木構造の祖孫関係で関係づけられた監視対象要素から出力されたイベント情報同士が、互いに関連するものと判断する。

【００３９】

なお、図４に示した構成管理データを、保守作業中に関するイベント抑止に利用することもできる。すなわち、保守作業を行う場合、監視対象要素のイベント発生を抑止しておく必要がある。そのとき、保守作業対象である監視対象要素よりも下位に位置する監視対象要素からのイベント発生も抑止することが望まれる。

【００４０】

すなわち、上位の監視対象要素（たとえば、ミドルウェア）の保守作業中は、下位の監視対象要素（たとえば、アプリケーション）において障害が発生していなくても、エラーイベントが出力される場合がある。そこで、保守作業時には、作業対象よりも下位の監視対象要素におけるイベント発生を抑止しておくことで、保守作業中の無駄なイベント出力を停止させることができる。

【００４１】

たとえば、図４に示したルータ５３を保守する場合、ルータ５３の下位に属するサーバ５５～５７、ミドルウェア６１～６３、アプリケーション６４～６６が、保守時のエラー発生対象として想定される。したがって、これらの下位構造からのイベント出力を抑止する。なお、イベント抑止のために、グループ化部２５は、保守作業を行う監視対象要素が指定されると、構成管理データに基づいて抑止対象を選定する。そして、グループ化部２５は、抑止時間を指定して、イベント発生抑止の情報を、抑止対象となる監視対象要素に対して送信する。

【００４２】

また、保守作業時に、保守対象からイベントが発生することもある。このとき発生したイベントはグループ化され、保守対象異常時のパターン定義グループとして保持される。

このように、グループ化部２５は、構成管理データを利用して、イベント情報同士の関連性を判断し、イベント情報のグループ化を行う。

【００４３】

図５は、イベント情報のグループ化処理を示す概念図である。図５に示すように、メッセージ正規化部２４で正規化されたイベント情報７１～７３がグループ化部２５に入力されると、関連するイベント情報によりイベントグループ３３が生成される。イベントグル

ープ３３には、各イベントグループを一意に識別するためのＩＤ（識別子）が振られている。他の内容は、イベント情報と同じである。

【００４４】

生成されたイベントグループ３３は、照合部２６に渡される。照合部２６では、イベントグループ３３とパターン定義グループ２２との照合が行われる。

図６は、照合処理を示す模式図である。パターン定義グループ２２には、ハードウェア事象、ＯＳ事象、ミドルウェア、アプリケーションから、障害発生時に出力されるイベントの組み合わせが登録されている。照合部２６は、このパターン定義グループ２２とイベントグループ３３とを照合（マッチング）することで、イベントグループ３３の発生原因となった障害を判定する。

【００４５】

イベントグループ３３に対応するパターン定義グループが特定されると、そのパターン定義グループに対応する対策情報が抽出される。

図７は、イベントグループの照合および対策情報出力例を示す図である。図７に示すように、イベントグループ３３ａのＨＷまたはＳＷ名およびエラーコードの項目と、パターン定義グループ２２ａのＨＷまたはＳＷ名およびエラーコードの項目が一致している。そこで、対策出力部２７は、パターン定義グループ２２ａのＩＤ「PT0008」に対応する対策情報を、対策情報テーブル２３から検索する。

【００４６】

対策情報テーブル２３には、ＩＤ、原因、対策、緊急レベルの欄が設けられている。各欄の横方向に並べられた情報同士が互いに関連づけられている。ＩＤは、対策情報を一意に識別するための識別情報である。対策情報のＩＤは、対応するパターン定義グループのＩＤの下４桁が共通となっている。すなわち、ＩＤ「PT0008」のパターン定義グループには、ＩＤ「000008」の対策情報２３ａが対応する。原因は、障害の原因を示す情報であり、原因となる監視対象要素（たとえば、ディスク）に関する情報を含んでいる。対策は、障害に対する対策方法を示す情報である。緊急レベルは、障害の重要度を示しており、緊急に対処する必要がある障害ほど、緊急レベルの値が大きい。

【００４７】

図７の例では、パターン定義グループ２２ａに対応する対策情報が対策出力部２７によって抽出される。対策情報２３ａによれば、障害の原因は「ディスク破損によるクラスタ切り替えが発生した。」ことであり、対策方法は「システム復旧後ＤＢのリカバリを実行してください。」である。また、この対策情報の緊急レベルは「２」である。

【００４８】

なお、照合部２６は、イベントグループと完全に一致するパターン定義グループを抽出するだけではなく、類似するパターン定義グループも抽出する。この際、照合部２６は、抽出したパターン定義グループに一致数と端数との情報を付与する。一致数は、一致したイベント情報の数である。端数は、不一致のイベント情報の数である。パターン定義グループ内のイベント情報が不足している場合、端数の符号がマイナスとなる。また、パターン定義グループ内のイベント情報が過多である場合、端数の符号がプラスとなる。

【００４９】

対策出力部２７は、対策情報を表示する際に、対応するパターン定義グループとイベントグループとの一致数や端数に応じてソートして、各パターン定義グループの対策情報を表示する。

【００５０】

図８は、照合結果のソート手順を示す図である。図８では、イベントグループ８１と６つのパターン定義グループとを照合している。なお、パターン定義グループのＩＤを、それぞれＰＴ１、ＰＴ２、ＰＴ３、ＰＴ４、ＰＴ５、ＰＴ６とする。

【００５１】

［ＳＴ１］まず、照合部２６が、イベントグループ８１と各パターン定義グループとを照合する。図８の例では、イベントグループ８１には、イベント情報「ａ，ｂ，ｃ，ｄ」

が含まれる。ＩＤ「ＰＴ１」のパターン定義グループには、イベント情報「ａ，ｂ，ｃ」が含まれる。ＩＤ「ＰＴ２」のパターン定義グループには、イベント情報「ａ，ｃ」が含まれる。ＩＤ「ＰＴ３」のパターン定義グループには、イベント情報「ａ，ｘ，ｙ，ｚ，ｑ」が含まれる。ＩＤ「ＰＴ４」のパターン定義グループには、イベント情報「ａ，ｂ，ｃ，ｄ，ｙ」が含まれる。ＩＤ「ＰＴ５」のパターン定義グループには、イベント情報「ａ，ｂ，ｃ，ｄ」が含まれる。ＩＤ「ＰＴ６」のパターン定義グループには、イベント情報「ｂ」が含まれる。

【００５２】

各パターン定義グループの照合の結果、各パターン定義グループの一致数と端数とが算出される。ＩＤ「ＰＴ１」のパターン定義グループは、一致数「３」、端数「－１」である。ＩＤ「ＰＴ２」のパターン定義グループは、一致数「２」、端数「－２」である。ＩＤ「ＰＴ３」のパターン定義グループは、一致数「１」、端数「＋４」である。ＩＤ「ＰＴ４」のパターン定義グループは、一致数「４」、端数「＋１」である。ＩＤ「ＰＴ５」のパターン定義グループは、一致数「４」、端数「０」である。ＩＤ「ＰＴ６」のパターン定義グループは、一致数「１」、端数「－３」である。

【００５３】

［ＳＴ２］照合結果を受け取った対策出力部２７は、まず、一致数によりパターン定義グループをソートする。この場合、一致数が多いほど、上位に並べられる。すると、「ＰＴ４」、「ＰＴ５」、「ＰＴ１」、「ＰＴ２」、「ＰＴ３」、「ＰＴ６」の順となる。

【００５４】

［ＳＴ３］次に、対策出力部２７は、端数によりソートする。この場合、端数の絶対値が少ないほど、上位に並べられる。すると、「ＰＴ５」、「ＰＴ４」、「ＰＴ１」、「ＰＴ２」、「ＰＴ６」、「ＰＴ３」の順となる。

【００５５】

［ＳＴ４］最後に対策出力部２７は、各パターン定義グループに対応する対策情報における緊急レベルを参照する。そして、緊急レベルが所定値より高い（緊急レベルを示す値が大きい）対策情報を、強調表示対象とする。たとえば「ＰＴ２」と「ＰＴ５」とのパターン定義グループに関する対策情報の緊急レベルが高ければ、それらの対策情報が強調表示対象となる。

【００５６】

対策出力部２７は、監視対象システムで発生したイベントに対する対策情報を、ソート順に表示すると共に、強調表示対象となっている対策情報を強調表示する。強調表示手段としては、たとえば、他の対策情報とは異なる色で表示する。

【００５７】

図９は、照合部における処理手順を示すフローチャートである。以下、図９に示す処理をステップ番号に沿って説明する。

［ステップＳ１０１］照合部２６は、イベントグループ３３を取得する。このとき、取得したイベントグループ３３に含まれるイベント情報の数をＮ（Ｎは自然数）とする。

【００５８】

［ステップＳ１０２］照合部２６は、重複イベントフィルタリングを行い、複数のパターン定義グループ２２が格納されたデータベースから、仮想テーブルを作成する。仮想テーブルは、処理対象となるシステムに適用可能なパターン定義グループ２２のみを抽出したデータベースである。なお、仮想テーブルは、各システムに発生し得る共通のパターン定義グループが格納されたデータベース、顧客毎のシステム構成に応じて発生し得るパターン定義グループが格納されたデータベース、およびパターン定義グループに対応する障害情報が格納されたデータベース等で構成される。

【００５９】

［ステップＳ１０３］照合部２６は、変数Ｉに１を設定し（Ｉ＝１）、変数ＪにＮを設定する（Ｊ＝Ｎ）。その後、ステップＳ１０４～Ｓ１０６の処理が並列に実行される。

［ステップＳ１０４］照合部２６は、障害情報が格納されたデータベースから、Ｉ番目

のイベント情報に関連付けられた障害情報を検索する。

【００６０】

［ステップＳ１０５］照合部２６は、顧客毎のシステム構成に応じて発生し得るパターン定義グループが格納されたデータベースから、Ｉ番目のイベント情報に関連付けられたパターン定義グループを検索する。

【００６１】

［ステップＳ１０６］照合部２６は、各システムに発生し得る共通のパターン定義グループが格納されたデータベースから、Ｉ番目のイベント情報に関連付けられたパターン定義グループを検索する。

【００６２】

［ステップＳ１０７］照合部２６は、Ｊ＝Ｉか否かを判断する。Ｊ＝Ｉであれば処理がステップＳ１０９に進められ、そうでなければ処理がステップＳ１０８に進められる。

［ステップＳ１０８］照合部２６は、Ｉをインクリメントする（Ｉ＝Ｉ＋１）。その後、ステップＳ１０４～Ｓ１０６の処理に進められる。

【００６３】

［ステップＳ１０９］照合部２６は、Ｉに１を設定し（Ｉ＝１）、変数Ｊに検出されたパターン定義グループ数を設定する。

［ステップＳ１１０］照合部２６は、ステップＳ１０５，Ｓ１０６で検出されたＩ番目のパターン定義グループを選択し、イベントグループと重複するイベント情報を抽出し、その数を数える。

【００６４】

［ステップＳ１１１］照合部２６は、Ｉ＝Ｊか否かを判断する。Ｉ＝Ｊであれば処理がステップＳ１１３に進められ、そうでなければ処理がステップＳ１１２に進められる。

［ステップＳ１１２］照合部２６は、Ｉをインクリメントする（Ｉ＝Ｉ＋１）。その後、ステップＳ１１０の処理に進められる。

【００６５】

［ステップＳ１１３］照合部２６は、イベントをソートする。

［ステップＳ１１４］照合部２６は、過去の同一事象の有無を確認する。

［ステップＳ１１５］対策出力部２７は、イベント情報の少なくとも一部が一致したパターン定義グループの対策情報（障害原因や対策方法を含む）を表示する。

【００６６】

以上のようにして、システム内で発生したイベントの情報を集めてイベントグループに応じて、障害原因や対策方法を自動的に特定することができる。

このように、パターン定義グループを蓄積しておくことで、過去に発生したトラブルと同様のトラブルを早期に発見することが可能となる。しかも、原因や対処方法を関連付けておくことで、トラブルシューティングの時間が短縮される。

【００６７】

さらに、多数の顧客のパターン定義グループ等を一元管理して、運用管理サーバの管理者がパターン定義グループ等を適宜更新することで、各顧客は、他の顧客で発生したトラブルに関するパターン定義グループを利用し、障害を発見することができる。運用管理サーバの管理者は、システム運用者の運用形態や、ソフトウェア開発者から提供される情報等を元に、パターン定義グループを登録することで、高品質のサーバ運用監視サービスを提供することができる。

【００６８】

次に、以上のような障害検出機能を有する運用管理サーバを用いて、顧客システムの遠隔保守サービスを行う場合の具体例を詳細に説明する。

図１０は、障害検出機能を提供するためのシステム構成例を示す図である。図１０の例では、運用管理サーバ１００は、ネットワーク１４を介して中継装置２１０に接続されている。中継装置２１０には、複数のサーバ２２０，２３０，２４０が接続されている。なお、中継装置２１０とサーバ２２０，２３０，２４０とは、顧客のシステムである。運用

管理サーバ１００には、パターンマッチングエンジン１１０が設けられている。パターンマッチングエンジン１１０が、サーバ２２０，２３０，２４０からのメッセージを収集し、パターン定義グループとのパターンマッチングを行い、対策情報を出力する。

【００６９】

図１１は、本発明の実施の形態に用いる運用管理サーバのハードウェア構成例を示す図である。運用管理サーバ１００は、ＣＰＵ(Central Processing Unit)１０１によって装置全体が制御されている。ＣＰＵ１０１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。

【００７０】

ＲＡＭ１０２には、ＣＰＵ１０１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、ＲＡＭ１０２には、ＣＰＵ１０１による処理に必要な各種データが格納される。ＨＤＤ１０３には、ＯＳやアプリケーションプログラムが格納される。

【００７１】

グラフィック処理装置１０４には、モニタ１１１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１１１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１１２とマウス１１３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１１２やマウス１１３から送られてくる信号を、バス１０７を介してＣＰＵ１０１に送信する。

【００７２】

通信インタフェース１０６は、ネットワーク１４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１４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。

【００７３】

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１１には、運用管理サーバ１００のハードウェア構成について説明したが、中継装置２１０やサーバ２２０，２３０，２４０も同様のハードウェア構成で実現することができる。

【００７４】

図１２は、運用管理サーバの内部構成を示すブロック図である。図１２に示すように運用管理サーバ１００は、パターンマッチングエンジン１１０、イベントグループ１２１、顧客ＤＢ（データベース）１２２、各種構成情報ＤＢ１２３、事象パターンＤＢ１２４、仮想事象パターンＤＢ１２５、情報抽出部１３０、環境設定部１４０、当日事象ログ１５１、当月事象ログ１５２、当日イベントグループログ１５３、当月イベントグループログ１５４、および結果格納ファイル１５５を有している。

【００７５】

パターンマッチングエンジン１１０は、他の機能と連携してパターンマッチング処理全体を制御する。

イベントグループ１２１は、サーバ２２０，２３０，２４０から送られたメッセージによって構成されたイベントグループである。

【００７６】

顧客ＤＢ１２２には、顧客に関する情報が格納される。

各種構成情報ＤＢ１２３には、顧客のシステム構成（ハードウェアとソフトウェア）に関する情報が格納される。

【００７７】

事象パターンＤＢ１２４は、サポートしているシステムで想定される障害に関する事象パターン（パターン定義グループと対策情報）が格納されたデータベースである。

仮想事象パターンＤＢ１２５は、顧客のシステム構成に応じて事象パターンＤＢ１２４

から抽出された事象パターンである。

【００７８】

情報抽出部１３０は、事象パターンＤＢ１２４から、サービス対象の顧客のシステムに関係する事象パターンのみを抽出し、仮想事象パターンＤＢ１２５を構築する。

当日事象ログ１５１は、当日発生した障害等に応じた事象パターンの識別情報や照合結果等を格納する記憶領域である。

【００７９】

当月事象ログ１５２は、当月発生した障害等に応じた事象パターンの識別情報や照合結果等を格納する記憶領域である。

当日イベントグループログ１５３は、当日発生したイベントグループを格納する記憶領域である。

【００８０】

当月イベントグループログ１５４は、当月発生したイベントグループを格納する記憶領域である。

結果格納ファイル１５５は、イベントグループに応じて検出された対策情報のリストである。

【００８１】

なお、各種構成情報ＤＢ１２３には、ハードウェア構成情報、ソフトウェア構成情報、ネットワーク構成情報、およびシステム構成情報が含まれる。各構成情報は、相互リンクのための情報を有している。

【００８２】

図１３は、ハードウェア構成情報のデータ構造例を示す図である。ハードウェア構成情報１２３ａには、機器ＩＤ、システム名、ホスト名、製品名、ＣＰＵ数、メモリ（ＭＥＭ）容量、ディスク(Ｄｉｓｋ)容量、ＬＡＮ数の欄が設けられている。

【００８３】

機器ＩＤは、サーバとして動作しているコンピュータの識別情報である。この機器ＩＤは、システム内部でホスト名の長さに影響されないように割り当てられたコードである。システム名は、サーバが従属するシステム（サービスの種別）の名称である。ホスト名は、ネットワーク上、あるいは管理上決められたサーバの識別名である。製品名は、サーバとして動作しているコンピュータの製品名である。ＣＰＵ数は、サーバに実装されているＣＰＵの数である。メモリ（ＭＥＭ）容量は、サーバに実装されているメインメモリの記憶容量である。ディスク(Ｄｉｓｋ)容量は、サーバに接続されているハードディスク装置の記憶容量である。ＬＡＮ数は、サーバに実装されているネットワークインタフェースの数である。

【００８４】

図１４は、ソフトウェア構成情報のデータ構造例を示す図である。ソフトウェア構成情報１２３ｂには、機器ＩＤ、ソフトウェア（ソフト）種別、ソフトウェア名、版数、修正版数の欄が設けられている。

【００８５】

機器ＩＤは、ハードウェア構成情報１２３ａの機器ＩＤと同じ項目であり、この機器ＩＤよって互いの情報が関連付けられる。ソフトウェア種別は、ソフトウェアの種別（ＯＳ、ミドルウェア、アプリケーション等）を示している。ソフトウェア名は、ソフトウェアの名称である。版数は、ソフトウェアのバージョン番号である。修正版数は、ソフトウェアに対して適用した障害修正データ（パッチ）の版数である。

【００８６】

図１５は、ネットワーク構成情報のデータ構造例を示す図である。ネットワーク構成情報１２３ｃには、顧客コード、機器ＩＤ、ホスト名、ＩＰ、インタフェース名、ＩＰ種別の欄が設けられている。顧客コードは、システムを運用している顧客の識別情報である。機器ＩＤは、ハードウェア構成情報１２３ａやソフトウェア構成情報１２３ｂの機器ＩＤと同じ項目であり、この機器ＩＤよって互いの情報が関連付けられる。ホスト名は、ハー

ドウェア構成情報１２３ａのホスト名と同じ項目である。ＩＰは、システムのＩＰアドレスである。インタフェース名は、装置内で各ネットワークインタフェースに割り当てられた識別番号である。ＩＰ種別は、ＩＰアドレスの種別である。ＩＰ種別には、Ｖ(Virtual)とＲ(Real)とがある。Ｖ(Virtual)は、仮想ＩＰアドレスを示す。Ｒ(Real)は、実ＩＰアドレスを示す。

【００８７】

図１６は、システム構成情報のデータ構造例を示す図である。システム構成情報１２３ｄには、顧客コード、機器ＩＤ、関連ホスト名、関連種別、関連ソフトの欄が設けられている。顧客コードは、ネットワーク構成情報１２３ｃの顧客コードと同じ項目である。機器ＩＤは、ハードウェア構成情報１２３ａ、ソフトウェア構成情報１２３ｂおよびネットワーク構成情報１２３ｃの機器ＩＤと同じ項目であり、この機器ＩＤよって互いの情報が関連付けられる。関連ホスト名は、関連するコンピュータの機器ＩＤである。関連ホスト名にＰＰ９９９９が設定された場合、他の機器との関連性がないことが示される。

【００８８】

このような構成の運用管理サーバ１００において、以下の様な処理が行われる。

まず、メッセージグループ化処理について説明する。

図１７は、メッセージグループ化処理の手順を示すフローチャートである。以下、図１７に示す処理をステップ番号に沿って説明する。

【００８９】

［ステップＳ１］情報抽出部１３０は、メッセージを取得する。すなわち、エラーが発生したサーバからエラーイベントが発生すると、エラーイベントの内容を示すメッセージがサーバから運用管理サーバ１００へ送られる。運用管理サーバ１００では、そのメッセージがパターンマッチングエンジン１１０に入力される。

【００９０】

［ステップＳ２］情報抽出部１３０は、メッセージを整形（正規化）すると共に、各種構成情報ＤＢ１２３を参照し、管理対象要素（ハードウェア、ソフトウェア等）を特定する。

【００９１】

［ステップＳ３］情報抽出部１３０は、各種構成情報ＤＢ１２３を参照し、他の管理対象要素との関連性を調査する。

［ステップＳ４］情報抽出部１３０は、関連イベントを受信したか否かを判断する。受信した場合、処理がステップＳ５に進められる。受信していない場合、処理がステップＳ６に進められる。

【００９２】

［ステップＳ５］情報抽出部１３０は、関連イベントをグループ化する。

［ステップＳ６］情報抽出部１３０は、ステップＳ１においてメッセージを受信してから所定時間経過したか否かを判断する。所定時間経過した場合、イベントグループを確定し、処理をパターンマッチングエンジン１１０に受け渡す。所定時間経過していなければ、処理がステップＳ４に進められる。

【００９３】

図１８は、イベントグループ化例を示す図である。たとえば、「イベント＃１１」のメッセージが入力されると、そのイベント情報を有するイベントグループ４１１が生成される。その後、「イベント＃２１」のメッセージが入力されると、そのイベント情報を有するイベントグループ４２１が生成される。同様に、「イベント＃３１」のメッセージが入力されると、そのイベント情報を有するイベントグループ４３１が生成される。

【００９４】

ここで、「イベント＃１１」に関連する「イベント＃１２」のメッセージが入力されると、イベントグループ４１１に「イベント＃１２」のイベント情報が追加される。これにより、２つのイベント情報を含むイベントグループ４１２となる。さらに、「イベント＃１１」に関連する「イベント＃１３」のメッセージが入力されると、イベントグループ４

１２に「イベント＃１３」のイベント情報が追加される。これにより、３つのイベント情報を含むイベントグループ４１３となる。

【００９５】

このようにしてグループ化されたイベント情報に基づいて、運用管理サーバ１００内の他のデータベースが構築される。

図１９は、イベントグループのデータ構造例を示す図である。イベントグループ１２１には、イベントグループＩＤ、グループ明細通番、イベントＩＤ、顧客ＩＤ、機器ＩＤ、インタフェース名、発生時刻、発生種別、発生略称、フィルタリング、メッセージの欄が設けられている。

【００９６】

イベントグループＩＤは、イベントグループ単位に設定される識別情報である。グループ明細通番は、イベントグループＩＤ内で設定される各レコード（１つのイベント情報で１つのレコードを構成する）の通し番号である。イベントＩＤは、各イベントに設定される識別情報である。顧客ＩＤは、イベントを発生させたシステムを運用している顧客の識別情報である。機器ＩＤは、イベントを発生させたサーバの識別情報である。インタフェース名は、イベント情報を含むメッセージを出力したネットワークインタフェースの識別情報である。発生時刻は、運用管理サーバ１００にメッセージが到達した時刻である。発生種別は、イベントを発生させた監視対象要素の種別である。発生略称は、イベントを発生させた監視対象要素の略称である。フィルタリングは、フィルタリングの有無を示している。メッセージは、イベント情報のメッセージの内容を示している。

【００９７】

なお、図１９には、イベントグループ１２１のデータ構造を示したが、当日イベントグループログ１５３、当月イベントグループログ１５４も同様のデータ構造である。

図２０は、顧客ＤＢのデータ構造例を示す図である。顧客ＤＢ１２２には、顧客ＩＤ、顧客名、顧客窓口、連絡先の欄が設けられている。顧客ＩＤは、顧客の識別情報である。顧客名は、顧客の名称である。顧客窓口は、顧客側の窓口となる担当者の名称である。連絡先は、顧客の電話番号等の連絡先である。

【００９８】

図２１は、事象パターンＤＢのデータ構造例を示す図である。事象パターンＤＢ１２４には、事象パターンＩＤ、事象明細通番、対象種別、対象略称、エラーメッセージの欄が設けられている。事象パターンＩＤは、事象パターンＤＢで管理される一意の識別番号である。事象明細通番は、同一事象パターンＩＤ内の付与されるイベント情報の管理番号である。対象種別は、監視対象要素の種別である。エラーメッセージは、出力されたイベントの内容を示す情報である。

【００９９】

なお、図２１には、事象パターンＤＢ１２４のデータ構造例を示しているが、仮想事象パターンＤＢ１２５のデータ構造も同様である。

図２２は、当日事象ログのデータ構造例を示す図である。当日事象ログ１５１は、イベントグループＩＤ、マッチング時刻、事象パターンＩＤ、イベント数、ヒット数の欄が設けられている。イベントグループＩＤは、イベントグループの識別番号である。マッチング時刻は、パターンマッチングエンジン１１０において、パターンマッチング処理を行った時刻である。事象パターンＩＤは、イベントグループとの照合（マッチング）によって抽出された事象パターンの識別情報である。イベント数は、事象パターンＩＤで示される事象パターン内のイベント情報の数である。ヒット数は、事象パターンＩＤで示される事象パターン内のイベント情報のうち、照合対象のイベントグループにも同時に含まれるイベント情報の数である。

【０１００】

なお、当日事象ログ１５１のイベント数とヒット数とから、対応する事象パターンに関する重みを算出することができる。たとえば、ヒット数が大きいほど、重みの値を大きく（重要度を高く）する。このような重みを示す数値を予め計算し、当日事象ログ１５１に

設定しておいてもよい。

【０１０１】

図２３は、当日イベントグループログのデータ構造例を示す図である。当日イベントグループログ１５３には、イベントグループＩＤ、イベント明細通番、イベントＩＤ、顧客ＩＤ、機器ＩＤ、発生時刻、発生種別、エラー番号ｏｒメッセージ、事象ＩＤ、重複事象ＩＤ数、マッチング結果などの情報が含まれる。

【０１０２】

イベントグループＩＤは、イベントグループの識別情報である。イベント明細通番は、発生したイベントに付与される通番である。イベントＩＤは、各イベント情報の識別情報である。顧客ＩＤは、イベントを発行したシステムを運用している顧客のＩＤである。機器ＩＤは、イベントを発行したサーバの識別情報である。発生時刻は、イベントの発生時刻である。発生種別は、イベントの種別（エラー、ワーニング等）である。エラー番号ｏｒメッセージは、イベントの内容を示す識別情報である。事象ＩＤは、イベントグループによって特定された事象パターンの識別情報である。重複事象ＩＤ数は、複数の事象パターンが検出されたときの検出事象パターン数である。マッチング結果は、検出された事象パターンに設定された対策情報の内容である。

【０１０３】

次に、パターンマッチング処理について詳細に説明する。

図２４は、パターンマッチング処理の手順を示すフローチャートである。以下、図２４に示す処理をステップ番号に沿って説明する。

【０１０４】

［ステップＳ１１］パターンマッチングエンジン１１０は、前処理を行う。たとえば、パターンマッチングエンジン１１０は、ＲＡＭ１０２内にワークテーブルを作成する。

［ステップＳ１２］パターンマッチングエンジン１１０は、イベントグループ１２１からイベントグループを取り出し、重複イベントフィルタリングを行う。重複イベントフィルタリングとは、同一サーバから出力された同一イベント情報を検出して、１つのイベント情報を残し、他のイベント情報を削除する処理である。

【０１０５】

［ステップＳ１３］パターンマッチングエンジン１１０は、イベントグループに含まれるイベント情報が１件以下か否かを判断する。イベント情報が１件以下の場合、複数のイベント情報の発生パターンによる障害検出対象外であるため、処理がステップＳ１９に進められる。イベント情報が２件以上の場合、処理がステップＳ１４に進められる。

【０１０６】

［ステップＳ１４］パターンマッチングエンジン１１０は、仮想事象パターンＤＢ１２５から組み合わせ対象の事象パターンを抽出する。

［ステップＳ１５］パターンマッチングエンジン１１０は、事象パターンが少なくとも１件抽出できたか否かを判断する。１件も抽出できなかった場合（抽出０件）、処理がステップＳ１９に進められる。少なくとも１件の事象パターンが抽出された場合、処理がステップＳ１６に進められる。

【０１０７】

［ステップＳ１６］パターンマッチングエンジン１１０は、組み合わせマッチングを行う。具体的には、パターンマッチングエンジン１１０は、フィルタリングされたイベントグループ内のイベント情報と、仮想事象パターンＤＢ１２５から抽出された各事象パターン内のイベント情報とを、総当たりでマッチングを行う。

【０１０８】

［ステップＳ１７］パターンマッチングエンジン１１０は、マッチング処理の結果を示す情報を提供する。たとえば、パターンマッチングエンジン１１０は、マッチング結果を運用管理サーバ１００のモニタ等に表示させる。

【０１０９】

［ステップＳ１８］パターンマッチングエンジン１１０は、過去の事象確認を行う。具

体的には、パターンマッチングエンジン１１０は、当月事象ログ１５２を参照し、同一事象ログの有無を確認する。同一事象ログがある場合に、結果ファイルに、その事象を追記する。

【０１１０】

［ステップＳ１９］パターンマッチングエンジン１１０は、後処理を行う。具体的には、パターンマッチングエンジン１１０は、当日事象ログ１５１と当月事象ログ１５２へマッチング結果を格納する。

【０１１１】

以下、図２４の各ステップの処理の詳細を説明する。

図２５は、前処理の詳細を示すフローチャートである。以下、図２５に示す処理をステップ番号に沿って説明する。

【０１１２】

［ステップＳ２１］パターンマッチングエンジン１１０は、新規のワークテーブルを作成する。たとえば、当日イベントグループログ１５３や当日事象ログ１５１のワークテーブルが作成される。

【０１１３】

［ステップＳ２２］パターンマッチングエンジン１１０は、ワーク領域を初期化する。

［ステップＳ２３］パターンマッチングエンジン１１０は、引数のチェックを行う。具体的には、パターンマッチングエンジン１１０は、入力チェックとして、「イベントグループのＩＤ」、「出力先フォルダ」の有無を判断する。いずれかの項目が未入力の場合、エラーとなる。その後、図２４に示す主処理に復帰する。

【０１１４】

次に、重複イベントフィルタリング処理について説明する。

図２６は、重複イベントフィルタリング処理の手順を示すフローチャートである。以下、図２６に示す処理をステップ番号に沿って説明する。

【０１１５】

［ステップＳ３１］パターンマッチングエンジン１１０は、イベントグループ１２１内のレコード（イベント情報）を一件読み込む。読み込みは、たとえば、顧客ＩＤ、機器ＩＤ、エラー番号ｏｒメッセージの順で行われる。

【０１１６】

［ステップＳ３２］パターンマッチングエンジン１１０は、イベントグループ読み込みの正否を判断する。イベントグループを読み込んだ場合、処理がステップＳ３３に進められる。読み込むべきイベントグループが無い場合、図２４に示す主処理に復帰する。

【０１１７】

［ステップＳ３３］パターンマッチングエンジン１１０は、フィルタリングの要否を判断する。具体的には、一件前のレコードと同一機種、同一エラー番号ｏｒメッセージの場合、フィルタリングが必要と判断される。フィルタリングが必要な場合、処理がステップＳ３１に進められる。フィルタリングが不要な場合、処理がステップＳ３４に進められる。

【０１１８】

［ステップＳ３４］パターンマッチングエンジン１１０は、重複イベントフィルタリング後のイベントグループを、当日イベントグループログ（ワークテーブル）１５３に格納する。その後、処理がステップＳ３１に進められる。

【０１１９】

次に、事象パターン抽出処理について詳細に説明する。

図２７は、事象パターン抽出処理の手順を示すフローチャートである。以下、図２７に示す処理をステップ番号に沿って説明する。

【０１２０】

［ステップＳ４１］パターンマッチングエンジン１１０は、事象パターン抽出処理を、情報抽出部１３０に依頼する。すると、情報抽出部１３０は、当日イベントグループログ

１５３からイベントグループを読み込む。読み込み順番は、顧客ＩＤ、機器ＩＤ、エラー番号ｏｒメッセージの順である。正常に読み込めた場合、以降の処理を行う。もし、読み込みに失敗したとき（読み込むべきイベントグループが無いとき）は、図２４に示す主処理に復帰する。

【０１２１】

［ステップＳ４２］情報抽出部１３０は、各種構成情報ＤＢ１２３から、構成情報を読み込む。具体的には、パターンマッチングエンジン１１０は、ステップＳ４１で読み込んだレコードの顧客ＩＤ、機器ＩＤをキーに、各種構成情報ＤＢ１２３を検索し、検出された各構成情報の略称を読み込む。

【０１２２】

［ステップＳ４３］情報抽出部１３０は、事象パターンの抽出および追加を行う。具体的には、情報抽出部１３０は、ステップＳ４２で読み込んだ各略称をキーにして、事象パターンＤＢ１２４から事象パターンを抽出し、抽出したレコードを仮想事象パターンＤＢ１２５に追加する。その後、図２４に示す主処理に復帰する。

【０１２３】

図２８は、仮想事象パターンＤＢのデータ構造例を示す図である。仮想事象パターンＤＢ１２５には、事象パターンＩＤ、事象明細通番、エラー番号ｏｒメッセージ、対象種別、対象略称などの情報が含まれる。

【０１２４】

事象パターンＩＤは、事象パターンの識別番号である。事象明細通番は、事象パターンに設定された通し番号である。エラー番号ｏｒメッセージは、障害の内容を示す識別情報である。対象種別は、監視対象要素の識別情報である。対象略称は、監視対象要素の略称である。

【０１２５】

次に、組み合わせマッチング処理について詳細に説明する。

図２９は、組み合わせマッチング処理の手順を示すフローチャートである。以下、図２９に示す処理をステップ番号に沿って説明する。

【０１２６】

［ステップＳ５１］パターンマッチングエンジン１１０は、当日イベントグループログ１５３内の各イベントグループのメッセージ件数をカウントする。

［ステップＳ５２］パターンマッチングエンジン１１０は、未処理のイベントグループを、当日イベントグループログ１５３から読み込む。このとき、全てのイベントグループの処理が終了している場合、エントリ終了のメッセージが返される。

【０１２７】

［ステップＳ５３］パターンマッチングエンジン１１０は、イベントグループが読み込まれたか否かを判断する。イベントグループが読み込まれた場合、処理がステップＳ５４に進められる。エントリ終了の場合、処理がステップＳ６０に進められる。

【０１２８】

［ステップＳ５４］パターンマッチングエンジン１１０は、仮想事象パターンＤＢ１２５から事象パターン内のイベント情報を読み込む。このとき、全ての事象パターンの処理が終了している場合、エントリ終了のメッセージが返される。

【０１２９】

［ステップＳ５５］パターンマッチングエンジン１１０は、事象パターン内のイベント情報が読み込まれたか否かを判断する。事象パターン内のイベント情報が読み込まれた場合、処理がステップＳ５６に進められる。エントリ終了の場合、処理がステップＳ５９に進められる。

【０１３０】

［ステップＳ５６］パターンマッチングエンジン１１０は、ステップＳ５２で読み込んだイベントグループ内に、ステップＳ５４で読み込んだ事象パターン内の各イベント情報と同一のイベント情報が存在するか否かを、イベント情報毎に判断する。そして、判断対

象となったイベント情報と同一イベント情報がイベントグループ内にある場合、メモリ内に作成されている当日事象ログ１５１（ワーク）として、ステップＳ５４で読み込んだ事象パターンが存在するか否かを判断する。当日事象ログが存在しない場合、処理がステップＳ５７に進められる。当日事象ログが存在する場合、処理がステップＳ５８に進められる。

【０１３１】

［ステップＳ５７］パターンマッチングエンジン１１０は、ステップＳ５４で読み込んだ事象パターンに対応するレコードを、当日事象ログ１５１のワークとしてメモリに格納する。その後、処置がステップＳ５４に進められる。

【０１３２】

当日事象ログ１５１として格納されるレコードのうち、事象パターンＩＤと事象明細通番とは、ステップＳ５４で抽出された事象パターンから転記される。ヒット件数には、１が設定される。イベント数には、抽出された事象パターンに登録されているイベント情報の数が設定される。

【０１３３】

［ステップＳ５８］パターンマッチングエンジン１１０は、ステップＳ５４で読み込んだ事象パターンに対応する当日事象ログ１５１のヒット件数の値を更新（１加算）する。その後、処理がステップＳ５４に進められる。

【０１３４】

［ステップＳ５９］パターンマッチングエンジン１１０は、パターンマッチング結果の更新処理を行い、処理をステップＳ５２に進める。具体的には、パターンマッチングエンジン１１０は、当日イベントグループログ１５３のワークテーブルに対して、情報を設定する。イベントグループＩＤ、イベント明細通番、イベントＩＤ、顧客ＩＤ、機器ＩＤ、発生時刻、発生種別、エラー番号ｏｒメッセージについては、ステップＳ５２で抽出したイベントグループの情報が転記される。マッチング結果には、照合の結果、同一のイベント情報を有する事象パターンが検出されたか否かを示すフラグが設定される。

【０１３５】

［ステップＳ６０］パターンマッチングエンジン１１０は、当日事象ログ１５１として作成されたワークテーブルに、重みの値を設定する。具体的には、パターンマッチングエンジン１１０は、「イベントグループ内のメッセージ数＞ヒット件数」の場合、「イベントグループ内のメッセージ数－ヒット件数」を重みとする。「イベントグループ内のメッセージ数＜ヒット件数」の場合、「ヒット件数－イベントグループ内のメッセージ数」を重みとする。「イベントグループ内のメッセージ数＝ヒット件数」の場合、メッセージ数を重みとする。

【０１３６】

次に、情報提供処理について説明する。

図３０は、情報提供処理の手順を示すフローチャートである。以下、図３０に示す処理をステップ番号に沿って説明する。

【０１３７】

［ステップＳ７１］パターンマッチングエンジン１１０は、当日事象ログ１５１のワークテーブルの事象ログをソートする。ソートは、重みによるソートと同一イベント情報数（イベントグループに含まれるイベント情報のうち、各事象ログに含まれるイベント情報の数）によるソートとに分かれる。本実施の形態では、重みによるソートの方が優先度が高い。すなわち、パターンマッチングエンジン１１０は、重みにより事象ログをソートし、重みが同一の事象ログ同士で、同一イベント情報数によるソートが行われる。

【０１３８】

［ステップＳ７２］パターンマッチングエンジン１１０は、当日事象ログ１５１を、ソートされた配列の先頭から順に１件ずつ読み込む。読み込むべき事象ログが無い場合、レコード終了のメッセージが生成される。

【０１３９】

［ステップＳ７３］パターンマッチングエンジン１１０は、レコード終了か否かを判断する。レコード終了の場合、処理が図２４の主処理に復帰する。事象ログのレコードが読み込まれた場合、処理がステップＳ７４に進められる。

【０１４０】

［ステップＳ７４］パターンマッチングエンジン１１０は、事象ログのレコードに対応する対策情報を読み込む。

［ステップＳ７５］パターンマッチングエンジン１１０は、事象ログを編集する。具体的には、パターンマッチングエンジン１１０は、まず、適合率を算出する。適合率は、（ヒット件数÷イベントログ内のイベント情報数）×１００で算出される。次に、パターンマッチングエンジン１１０は、事象パターンＩＤをキーにして当日事象ログ１５１内のレコードを読み込む。

【０１４１】

［ステップＳ７６］パターンマッチングエンジン１１０は、適合率と結果とを結果格納ファイル１５５に出力する。

次に、過去の事象確認処理について詳細に説明する。

【０１４２】

図３１は、過去の事象確認処理の手順を示すフローチャートである。以下、図３１に示す処理をステップ番号に沿って説明する。

［ステップＳ８１］パターンマッチングエンジン１１０は、当日事象ログ１５１の事象ＩＤを読み込む。この際、読み込むべきレコードがなければ、レコード終了のメッセージが生成される。

【０１４３】

［ステップＳ８２］パターンマッチングエンジン１１０は、レコード終了か否かを判断する。レコード終了であれば、図２４に示す主処理に復帰する。レコードが読み込まれた場合、処理がステップＳ８３に進められる。

【０１４４】

［ステップＳ８３］パターンマッチングエンジン１１０は、過去の事象とのマッチング処理を行う。具体的には、当月事象ログ１５２から、同じレコードを検索する。

［ステップＳ８４］パターンマッチングエンジン１１０は、ステップＳ８３のマッチングの結果、一致するレコードがあったか否か判断する。一致するレコードがあった場合、処理がステップＳ８５に進められる。一致するレコードが無い場合、図２４に示す主処理に復帰する。

【０１４５】

［ステップＳ８５］パターンマッチングエンジン１１０は、一致したレコードが事象ログとして記録される元となったイベントグループＩＤを一件取得する。

［ステップＳ８６］パターンマッチングエンジン１１０は、ステップＳ８５で取得すべきレコードが終了したか否かを判断する。レコード終了の場合、処理が図２４に示す主処理に復帰する。レコードが取得できた場合、処理がステップＳ８７に進められる。

【０１４６】

［ステップＳ８７］パターンマッチングエンジン１１０は、ステップＳ８１で読み込まれた事象ＩＤに対応する事象パターン内の各イベント情報が、ステップＳ８５で取得されたイベントグループＩＤに対するイベントグループ内に存在するか否かを確認する。

【０１４７】

［ステップＳ８８］パターンマッチングエンジン１１０は、ステップＳ８７に処理において、全てのイベント情報が存在したか否かを判断する。全てのイベント情報が存在した場合、処理がステップＳ８９に進められる。そうでない場合、処理がステップＳ８５に進められる。

【０１４８】

［ステップＳ８９］パターンマッチングエンジン１１０は、ステップＳ８５で取得したイベントグループＩＤが生成された時刻を出力する。その後、処理がステップＳ８５に進

められる。

【０１４９】

次に、後処理について詳細に説明する。

図３２は、後処理の手順を示すフローチャートである。以下、図３２に示す処理をステップ番号に沿って説明する。

【０１５０】

［ステップＳ９１］パターンマッチングエンジン１１０は、当日事象ログ１５１を読み込む。

［ステップＳ９２］パターンマッチングエンジン１１０は、レコード終了か否かを判断する。レコード終了であれば、処理がステップＳ９４に進められる。レコード終了でなければ、処理がステップＳ９３に進められる。

【０１５１】

［ステップＳ９３］パターンマッチングエンジン１１０は、読み込んだ当日事象ログ１５１のワークテーブルをＨＤＤ等に蓄積すると共に、当月事象ログ１５２にも蓄積する。また、当日事象ログ１５１の抽出原因である当日イベントグループログ１５３のワークテーブルを、ＨＤＤ等に蓄積するとともに、当月イベントグループログ１５４にも蓄積する。その後、処理がステップＳ９１に進められる。

【０１５２】

［ステップＳ９４］パターンマッチングエンジン１１０は、蓄積処理が正常に終了したか否かを判断する。正常に終了した場合には、処理がステップＳ９５に進められる。そうでない場合には、処理がステップＳ９６に進められる。

【０１５３】

［ステップＳ９５］パターンマッチングエンジン１１０は、蓄積処理をコミットする。その後、処理が終了する。

［ステップＳ９６］パターンマッチングエンジン１１０は、ロールバック処理を行う。ロールバック処理とは、データベースに障害が発生したとき等に、記録してあるチェックポイントにまでデータを戻して、改めて処理を開始することである。その後、処理が終了する。

【０１５４】

図３３は、情報の流れを示す図である。図３３の例では、イベントグループ１２１に、４つのイベント情報が含まれている。ここで、イベントグループ１２１に含まれる各イベント情報のＩＤを、「ａ，ｂ，ｃ，ｄ」とする。

【０１５５】

仮想事象パターンＤＢ１２５には、複数の事象パターンが格納されている。ここで、各事象パターンの事象パターンＩＤを、それぞれ「Ａ，Ｂ，Ｃ，Ｄ，・・・」とする。事象パターンＩＤ「Ａ」の事象パターンには、イベント情報「ａ，ｂ，ｃ，・・・」が含まれる。事象パターンＩＤ「Ｂ」の事象パターンには、イベント情報「ａ，ｃ，・・・」が含まれる。事象パターンＩＤ「Ｃ」の事象パターンには、イベント情報「ａ，ｅ，・・・」が含まれる。事象パターンＩＤ「Ｄ」の事象パターンには、イベント情報「ｂ，ｃ，・・・」が含まれる。

【０１５６】

このようなイベントグループ１２１と仮想事象パターンＤＢ１２５内の各事象パターンとのマッチングが行われると、図３３に示すような組み合わせマッチング結果９１が得られる。なお、図３３の例では、簡単のため２つのイベント情報の組み合わせに対するマッチング結果のみを示している。

【０１５７】

組み合わせマッチング結果９１では、イベント情報組み合わせ「ａ，ｂ」に対応する事象パターンとして、事象パターンＩＤ「Ａ」が抽出されている。イベント情報組み合わせ「ａ，ｃ」に対応する事象パターンとして、事象パターンＩＤ「Ａ，Ｂ」が抽出されている。イベント情報組み合わせ「ａ，ｄ」に対応する事象パターンとして、事象パターンＩ

Ｄ「Ｃ」が抽出されている。イベント情報組み合わせ「ｂ，ｃ」に対応する事象パターンとして、事象パターンＩＤ「Ｄ」が抽出されている。イベント情報組み合わせ「ｂ，ｄ」に対応する事象パターンは抽出されていない。イベント情報組み合わせ「ｃ，ｄ」に対応する事象パターンは抽出されていない。

【０１５８】

このような組み合わせマッチング結果９１に基づいて、当日事象ログ１５１が生成される。イベント情報「ａ」に対しては、事象パターンＩＤ「Ａ，Ａ，Ｂ，Ｃ」が関連付けられている。イベント情報「ｂ」に対しては、事象パターンＩＤ「Ａ，Ｄ」が関連付けられている。イベント情報「ｃ」に対しては、事象パターンＩＤ「Ａ，Ｂ，Ｄ」が関連付けられている。イベント情報「ｄ」に対しては、事象パターンＩＤ「Ｃ」が関連付けられている。

【０１５９】

当日事象ログ１５１から結果格納ファイル１５５が生成される。図３３の例では、障害の原因である可能性の高い順に、上からヒット件数が示されている。図３３では、事象パターンＩＤ「Ａ」のヒット件数が「４」、事象パターンＩＤ「Ｂ」のヒット件数が「２」、事象パターンＩＤ「Ｃ」のヒット件数が「２」、事象パターンＩＤ「Ｄ」のヒット件数が「２」である。なお、図３３の例では、結果格納ファイル１５５にヒット件数を示しているが、適合率を計算して設定することもできる。

【０１６０】

以上のようにして、関連するイベント情報同士でイベントグループ１２１を作成し、事象パターンと照合することで、可能性の高いイベント発生原因を抽出することができる。運用管理サーバ１００の管理者は、結果格納ファイルを参照することで、障害の原因およびその対処方法を知ることができる。

【０１６１】

ところで、上記の説明では、システムからイベントに応じたメッセージを運用管理サーバ１００で直接取り込み、イベント情報を自動でグループ化して分析処理を行っているが、任意に指定したイベントグループの分析処理を行うこともできる。分析すべきイベントグループは、たとえば、イベントグループ検索画面から検索して抽出することができる。

【０１６２】

図３４は、イベントグループ検索画面の一例を示す図である。イベントグループ検索画面５１０は、照合対象のイベントグループを過去のログ情報（過去所定期間のイベントグループログが格納されたデータベース）から任意のイベントグループを検索するための検索画面である。

【０１６３】

イベントグループ検索画面５１０には、メッセージ指定情報入力部５１１と、事象内容指定情報入力部５１２とが設けられている。

メッセージ指定情報入力部５１１には、イベント情報を含むメッセージを出力したシステムの顧客名と、検索期間とを入力することができる。そして、メッセージ指定情報入力部５１１に入力された情報に適合するイベントグループが検索される。

【０１６４】

事象内容指定情報入力部５１２には、原因の入力領域と対処の入力領域とが設けられている。これらの入力領域にはチェックボックスが対応付けられており、チェックボックスが選択された入力領域が、検索キーとして有効となる。事象内容指定情報入力部５１２において入力された上に基づいてイベントグループログの検索が行われる。

【０１６５】

イベントグループ検索画面５１０で検索を行った結果、イベントグループが抽出され、画面表示される。このとき、イベントグループに含まれるイベント情報の中から、照合対応とするイベント情報を手動で選択し、選択されたイベント情報に基づいて、障害原因や対策の解析を行うこともできる。

【０１６６】

図３５は、イベント選択画面の例を示す図である。イベント選択画面５２０には、検索等により抽出されたイベントグループログに含まれるイベント情報のリストが表示されている。表示されているイベント情報の一部５２１を選択する。

【０１６７】

選択されたイベント情報を１つのイベントグループとして、障害原因等の分析が可能である。

各種手順で指定されたイベントグループの分析を行い、結果格納ファイル１５５が生成されると、その内容が分析結果表示画面に表示される。

【０１６８】

図３６は、分析結果表示画面の例を示す図である。分析結果表示画面５３０には、選択メッセージ表示欄５３１、検索条件表示欄５３２、および事象内メッセージ表示欄５３３が設けられている。選択メッセージ表示欄５３１には、選択されたイベントグループのイベント情報の内容が示されている。検索条件表示欄５３２には、関係ある事象パターンの原因と対処方法との一覧が表示されている。事象内メッセージ表示欄５３３には、事象パターンに定義されたイベント情報の内容が示されている。

【０１６９】

［応用例］

上記の実施の形態では、サーバが発生したエラーイベントに基づいて障害内容を解析しているが、操作入力に応じた障害内容の解析も可能である。言い換えると、上記の実施の形態は、障害発生の結果出力されるイベント情報に基づいて、障害箇所等を特定しているのに対し、障害発生原因となる操作入力に基づいて、障害箇所等を特定することができる。これにより、利用者の環境によっては、単独で実行可能な処理が、他のアプリケーションが動作しているために実行できない場合等に、それらの処理の実行命令の入力履歴から、障害の内容を自動で特定することができる。

【０１７０】

たとえば、データベースのバックアップ処理中は、そのデータベースに対するアクセスができない。この場合、従来では、データベースへのアクセスエラーが出力される。しかし、ユーザは、データベースにアクセスできない理由までは認識することができない。

【０１７１】

そこで、同時実行不可能な複数のコマンドのセットをパターン定義グループとして蓄積しておき、そのコマンドセットが入力された場合には、障害が発生する原因と対策方法（たとえば、原因となるアプリケーションの停止）を表示することで、ユーザの誤操作を正すことができる。

【０１７２】

しかも、複数のアプリケーションが同時に実行される場合、一方のアプリケーションの制約外の処理等が原因となり、エラーが発生する場合がある。このとき、一般的に、ユーザには、制約外の処理を行うアプリケーションプログラムの修正が許されていない。そこで、エラーを引き起こす重複実行を指示する操作入力が行われたときに、エラーを起こさせずに所望の処理結果を得るための代替の操作入力方法をユーザに提示することで、エラーを回避させることができる。

【０１７３】

このような処理は、図１１と同様の構成で実現することができる。ただし、サーバあるいは他の端末装置からは、操作入力内容を示すメッセージが運用管理サーバ１００に送られ、運用管理サーバ１００は、そのメッセージを履歴テーブルに格納する。また、運用管理サーバ１００には、障害が発生する操作パターンを格納する操作パターンＤＢが設けられる。

【０１７４】

図３７は、履歴テーブルのデータ構造例を示す図である。履歴テーブル６１０には、発生日時、コマンド名、サーバ名の欄が設けられている。発生日時は、操作入力の内容を示すメッセージを運用管理サーバ１００が受信した時刻である。コマンド名は、操作入力さ

れたコマンドの内容である。サーバ名は、そのメッセージを送信したサーバの名称である。なお、履歴テーブル６１０は、所定時間（たとえば、１３：３０から１０分間）内の操作入力内容を示している。

【０１７５】

図３８は、操作パターンＤＢのデータ構造例を示す図である。操作パターンＤＢ６２０には、操作パターンコード、操作パターン、現象、原因、対処の欄が設けられている。

操作パターンコードは、各操作パターンの識別番号である。操作パターンは、障害等を引き起こすコマンドの組である。現象は、障害の内容である。原因は、障害の原因である。対処は、障害を回避するための対処方法である。

【０１７６】

たとえば、図３７に示す履歴テーブル６１０から、vi（エディタの起動コマンド）でhosts（ファイル）を編集後に、mail（メーラの起動コマンド）を実行されている。このとき、エラーが発生すると、利用者は運用管理サーバ１００の解析結果を参照する。

【０１７７】

このような操作パターンは、操作パターンＤＢ６２０の操作パターンコード「s00000002」と一致する。従って、運用管理サーバ１００は、該当する操作パターンの現象、原因、対処の情報を利用者の端末装置に表示する。利用者は、現象を確認し、示されているメッセージが実際に起きた現象通りであれば、操作パターンコード「s00000002」で示された現象が発生したことを認識できる。そこで、利用者は、原因と対処を参照して、障害の発生原因を取り除くことができる。

【０１７８】

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、運用管理サーバ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 disc)などがある。

【０１７９】

プログラムを流通させる場合には、たとえば、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。

【０１８０】

プログラムを実行するコンピュータは、たとえば、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。

【０１８１】

以上説明したように本発明では、サーバ内の監視対象要素から出力されたイベント情報をグループ化し、グループ化されたイベントグループに類似するパターン定義グループに応じた障害対策情報を抽出するようにしたため、サーバ内の監視対象要素から障害原因となる監視対象要素を容易に特定することができる。

【０１８２】

上記については単に本発明の原理を示すものである。さらに、多数の変形、変更が当業者にとって可能であり、本発明は上記に示し、説明した正確な構成および応用例に限定されるものではなく、対応するすべての変形例および均等物は、添付の請求項およびその均等物による本発明の範囲とみなされる。

【図面の簡単な説明】

【０１８３】

【図１】実施の形態に適用される発明の概念図である。

【図２】トラブル情報回帰型解決方法の運用例を示す図である。

【図３】メッセージ正規化処理を示す図である。

【図４】構成管理データの構造例を示す図である。

【図５】イベント情報のグループ化処理を示す概念図である。

【図６】照合処理を示す模式図である。

【図７】イベントグループの照合および対策情報出力例を示す図である。

【図８】照合結果のソート手順を示す図である。

【図９】照合部における処理手順を示すフローチャートである。

【図１０】障害検出機能を提供するためのシステム構成例を示す図である。

【図１１】本発明の実施の形態に用いる運用管理サーバのハードウェア構成例を示す図である。

【図１２】運用管理サーバの内部構成を示すブロック図である。

【図１３】ハードウェア構成情報のデータ構造例を示す図である。

【図１４】ソフトウェア構成情報のデータ構造例を示す図である。

【図１５】ネットワーク構成情報のデータ構造例を示す図である。

【図１６】システム構成情報のデータ構造例を示す図である。

【図１７】メッセージグループ化処理の手順を示すフローチャートである。

【図１８】イベントグループ化例を示す図である。

【図１９】イベントグループのデータ構造例を示す図である。

【図２０】顧客ＤＢのデータ構造例を示す図である。

【図２１】事象パターンＤＢのデータ構造例を示す図である。

【図２２】当日事象ログのデータ構造例を示す図である。

【図２３】当日イベントグループログのデータ構造例を示す図である。

【図２４】パターンマッチング処理の手順を示すフローチャートである。

【図２５】前処理の詳細を示すフローチャートである。

【図２６】重複イベントフィルタリング処理の手順を示すフローチャートである。

【図２７】事象パターン抽出処理の手順を示すフローチャートである。

【図２８】仮想事象パターンＤＢのデータ構造例を示す図である。

【図２９】組み合わせマッチング処理の手順を示すフローチャートである。

【図３０】情報提供処理の手順を示すフローチャートである。

【図３１】過去の事象確認処理の手順を示すフローチャートである。

【図３２】後処理の手順を示すフローチャートである。

【図３３】情報の流れを示す図である。

【図３４】イベントグループ検索画面の一例を示す図である。

【図３５】イベント選択画面の例を示す図である。

【図３６】分析結果表示画面の例を示す図である。

【図３７】履歴テーブルのデータ構造例を示す図である。

【図３８】操作パターンＤＢのデータ構造例を示す図である。

【符号の説明】

【０１８４】

１　運用管理サーバ

１ａ　パターン定義グループＤＢ

１ｂ　障害対策情報ＤＢ

２～４　サーバ
１１，１２　サーバ
２０　運用管理サーバ
２１　イベント抽出テーブル
２２　パターン定義グループ
２３　対策情報テーブル
２４　メッセージ正規化部
２５　グループ化部
２６　照合部
３１，３２　メッセージ
３３　イベントグループ

【図１】

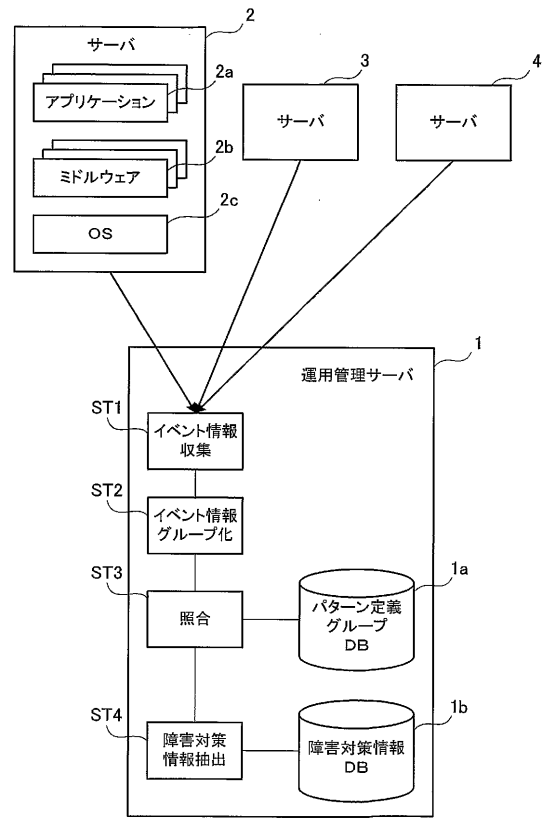


図1

【図２】

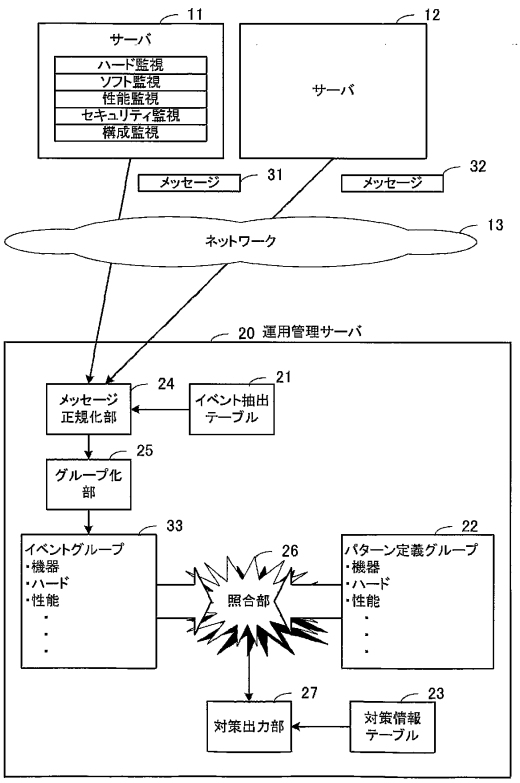
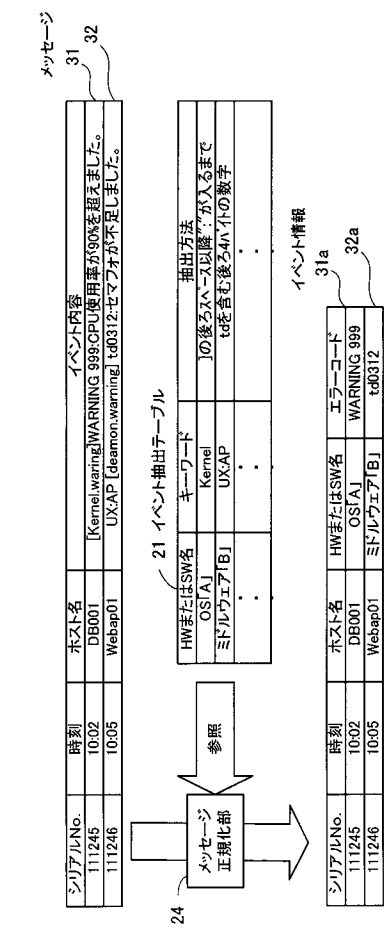


図2

【図３】



【図４】

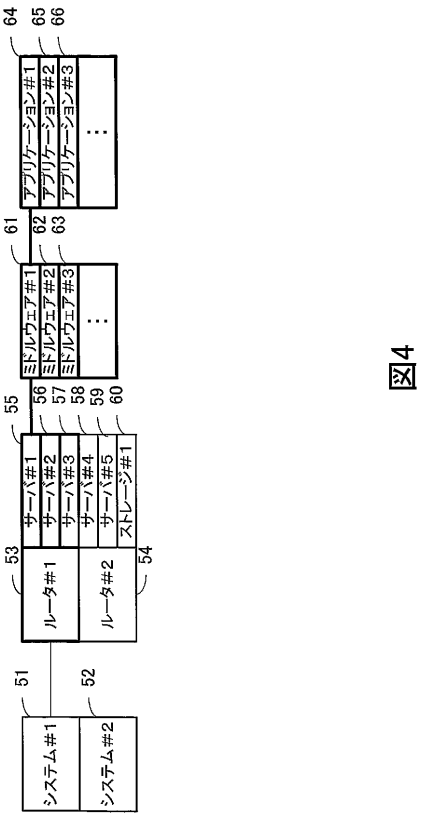


図4

【図５】

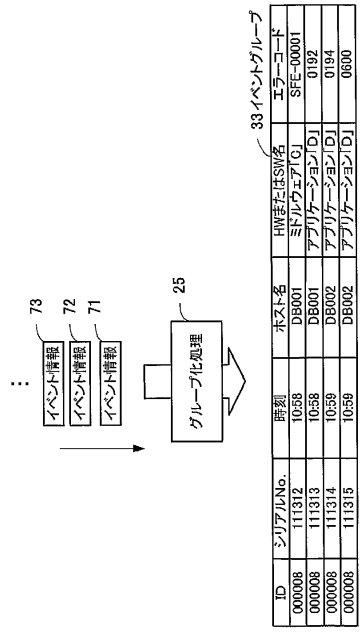


図5

【図６】

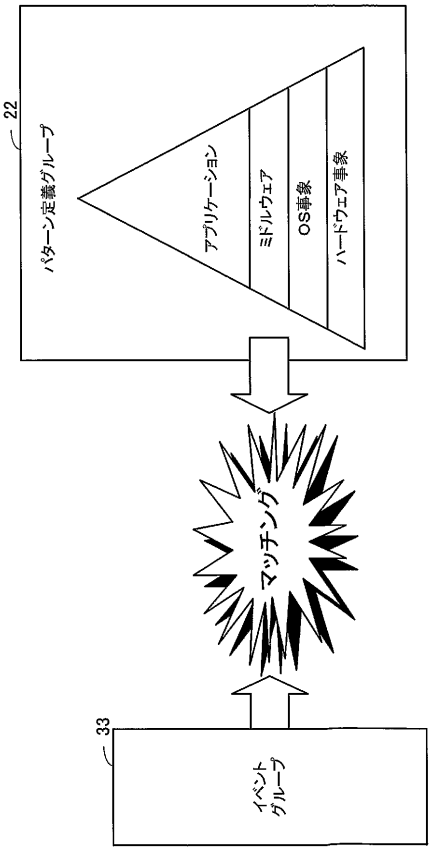


図6

【図７】

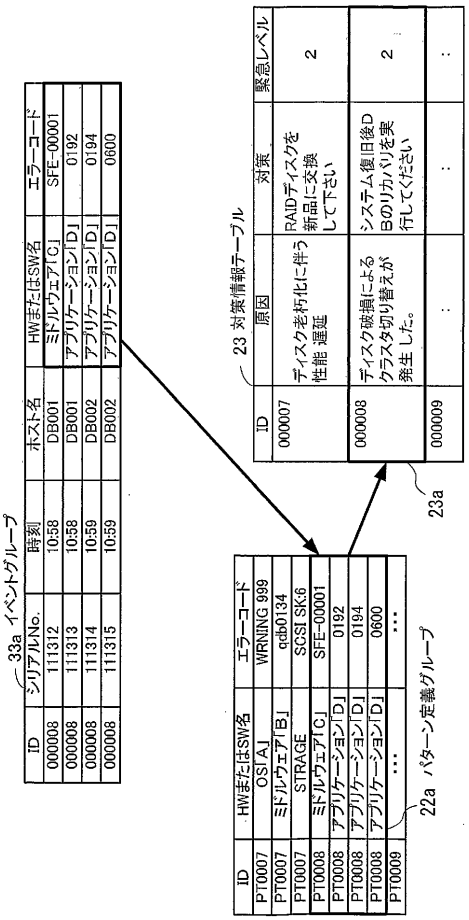


図7

【図８】

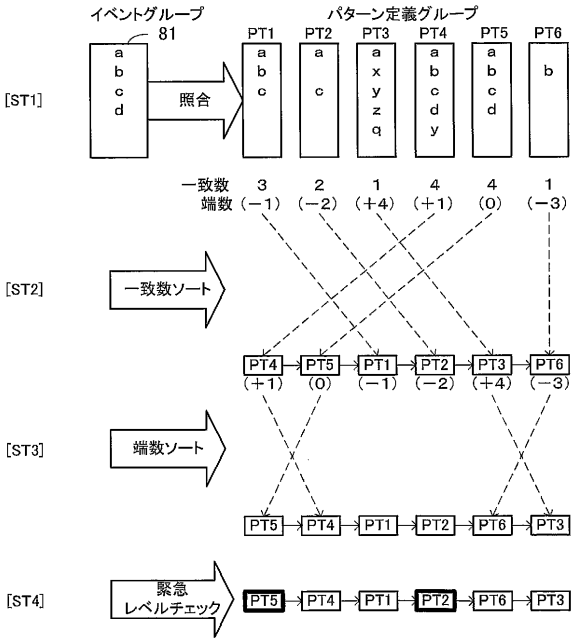


図8

【図９】

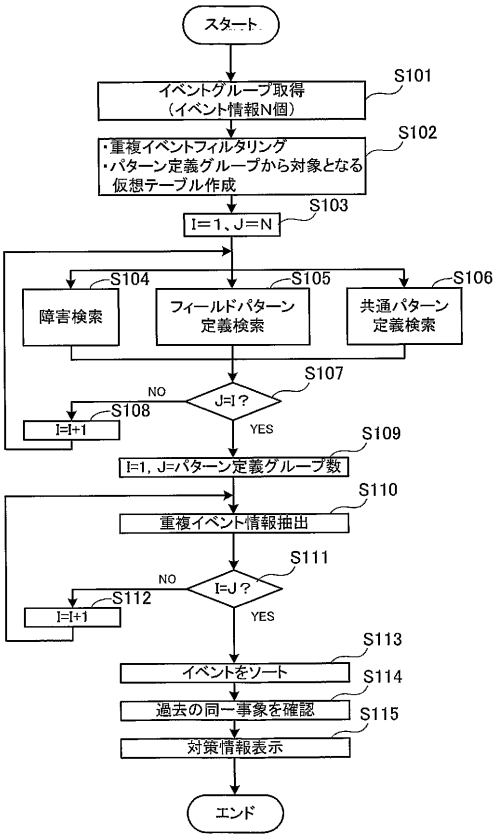


図9

【図１０】

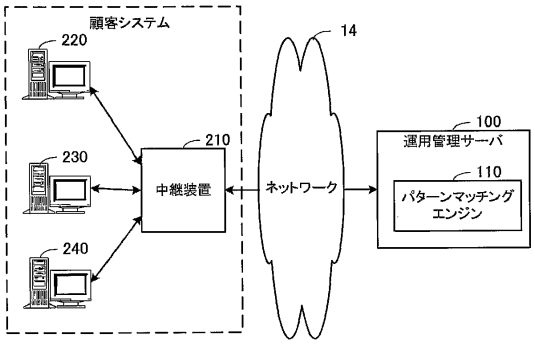


図10

【図１１】

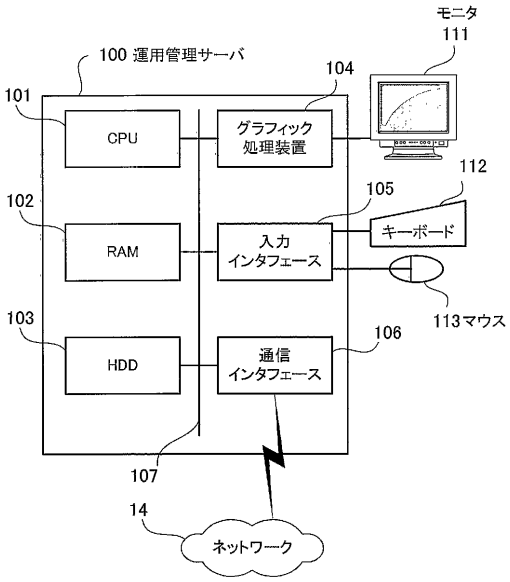


図11

【図１２】

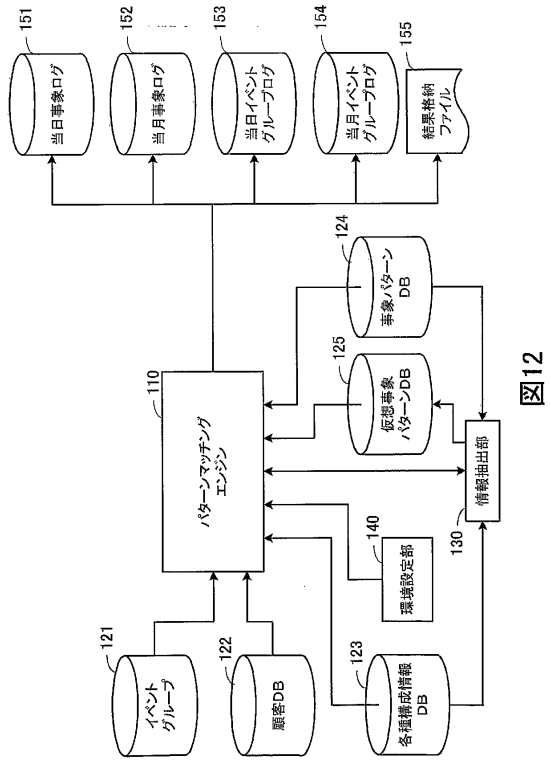


図12

【図１３】

123a ハードウェア構成情報

機器ID	システム名	ホスト名	製品名	CPU数	MEM容量	Disk容量	LAN数
PP0001	業務管理	test-nad01	サーバ#1	4	2048MB	40GB	3

図13

【図１４】

123b ソフトウェア構成情報

機器ID	ソフト種別	ソフトウェア名	版数	修正版数
PP0001	OS	UNIX	V8.0	1101版
PP0001	ミドルウェア	Iステージ	V5.1	0908版

図14

【図１５】

123c ネットワーク構成情報

顧客コード	機器ID	ホスト名	IP	インタフェース名	IP種別
U00001	PP0001	test-nad00	192.168.0.1	NIC01	V(Virtual)
U00001	PP0001	test-nad01	192.168.0.2	NIC02	R(Real)
U00001	PP0001	test-nad11	192.168.0.3	NIC03	R(Real)

【図１６】

123d システム構成情報

顧客コード	機器ID	関連ホスト名	関連種別	関連ソフト
U00001	PP0001	PP0002	CLUSTER	安全クラスタ
U00001	PP0001	PP0010	NAD-WWW	Iステージ
U00001	PP0001	PP0011	NAD-WWW	Iステージ
U00001	PP0001	PP0012	NAD-WWW	Iステージ
U00001	PP0001	PP1000	運用管理	S-walker
U00001	PP0001	PP9999	独立	STRAGE

図16

【図１７】

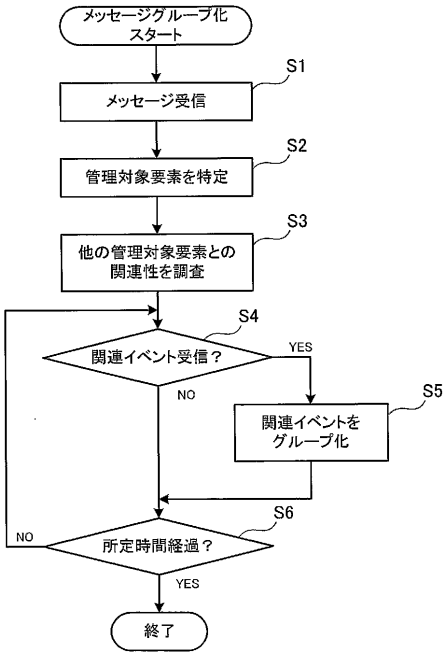


図17

【図１８】

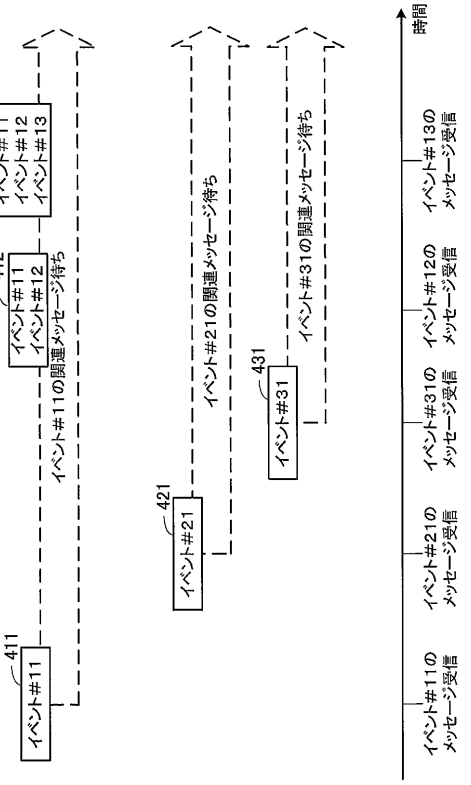


図18

【図１９】

121 イベントグループ

イベントグループID	グループ明細通番	イベントID	顧客ID	機器ID	インタフェース名	発生時刻
00000001	0001	6755132	U001	PP0001	NIC01	2002/10/31014:38:00
00000001	0002	6755133	U001	PP0001	NIC01	2002/10/31014:38:00
00000002	0001	6755134	U001	PP0002	NIC01	2002/10/31014:38:00
00000002	0002	6755135	U001	PP0002	NIC01	2002/10/31014:38:00
00000002	0003	6755136	U001	PP0002	NIC01	2002/10/31014:38:00

発生種別	発生略称	フィルタリング	メッセージ
ミドルウェア	S-walker	0	サービスレベル監視において、監視項目(Percentage of disk time)が、しきい値(9500)を上回りました。
OS	UNIX	0	ディスクアクセス率が高くなっています。
ミドルウェア	S-walker	0	bad community from localhost
ミドルウェア	S-walker	0	unable to process a pdu from localhost:32799
ミドルウェア	S-walker	0	deleted from the agent table

図19

【図２０】

122 顧客DB

顧客ID	顧客名	顧客窓口	連絡先
U000001	ABC電気株式会社	織田　信長	03-1111-1111
U000002	DEF運輸株式会社	豊臣　秀吉	03-1111-2222
U000003	XYZサービス株式会社	徳川　家康	03-1111-3333

図20

【図２１】

124 事象パターンDB

事象パターンID	事象明細通番	対象種別	対象略称	エラーメッセージ
PT000011	0001	ミドルウェア	S-walker	WARNING:999
PT000011	0002	OS	UNIX	ID:047568
PT000011	0001	ハードウェア	STRAGE	ASK:6 Disk Retry Over
PT000011	0002	OS	UNIX	ID:468756 disk error
PT000011	0003	ミドルウェア	STRAGE	td00136 Application timeout

【図２２】

151 当日事象ログ

イベントグループID	マッチング時刻	事象パターンID	イベント数	ヒット数
000000001	2002/10/31 14:49:13	PT000011	4	2
000000001	2002/10/31 14:49:14	PT000093	1	1
000000002	2002/10/31 14:45:02	PT000012	6	3
000000002	2002/10/31 14:46:40	PT000012	7	3

図22

【図２３】

153 当日イベントグループログ

イベントグループID
イベント明細通番
イベントID
顧客ID
機器ID
発生時刻
発生種別
エラー番号orメッセージ
事象ID
重複事象ID数
マッチング結果

図23

【図２４】

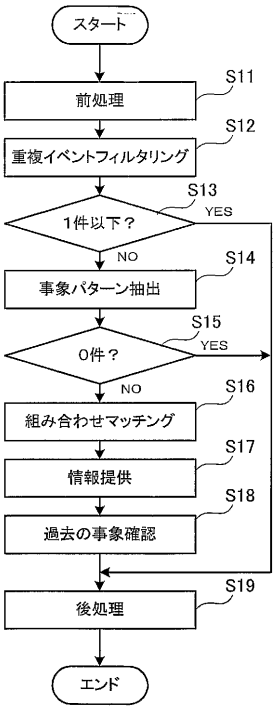


図24

【図２５】

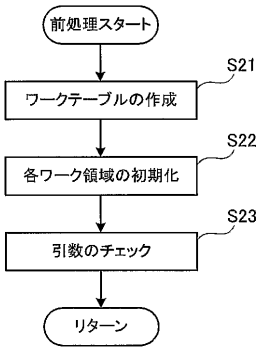


図25

【図２６】

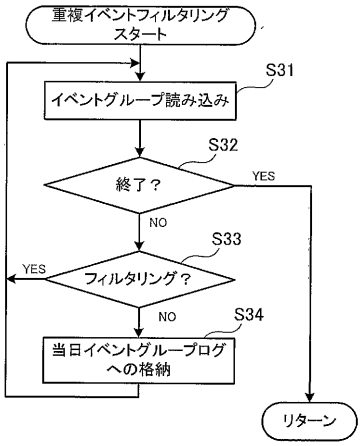


図26

【図２７】

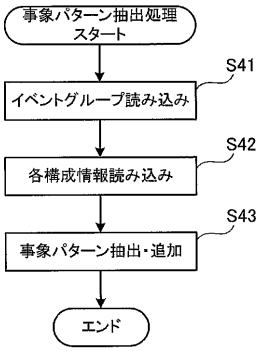


図27

【図２８】

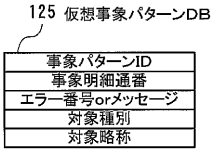


図28

【図２９】

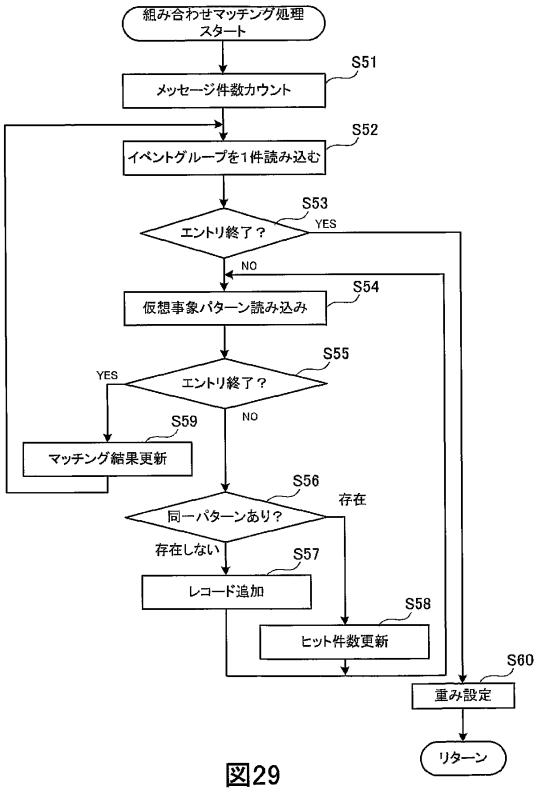


図29

【図３０】

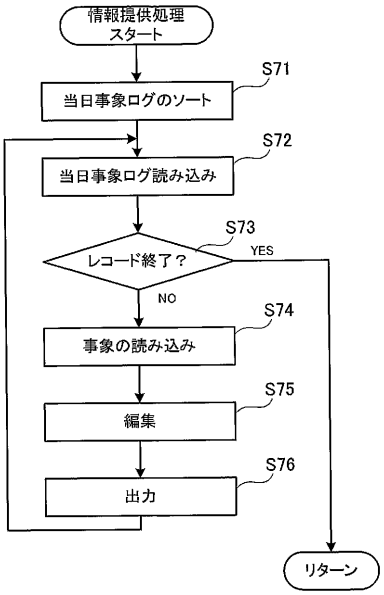


図30

【図３１】

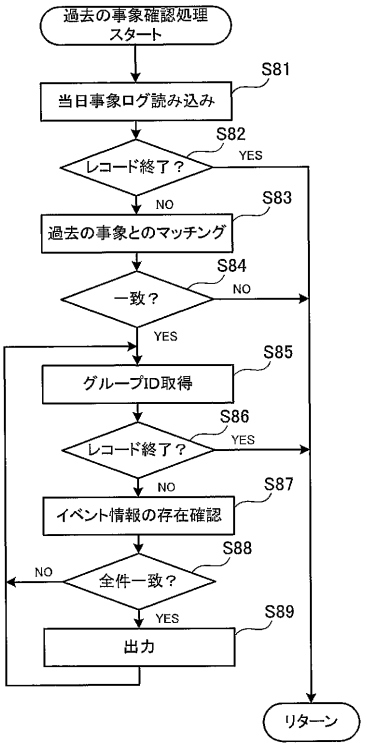
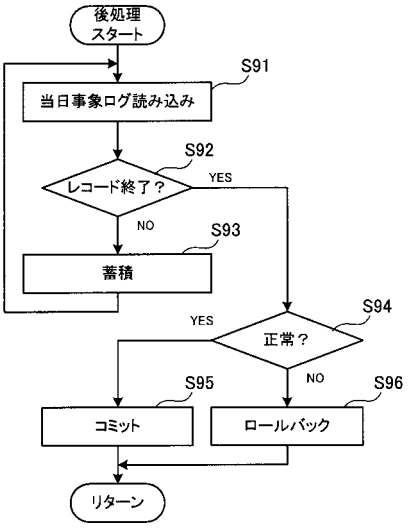


図31

【図３２】



【図３３】

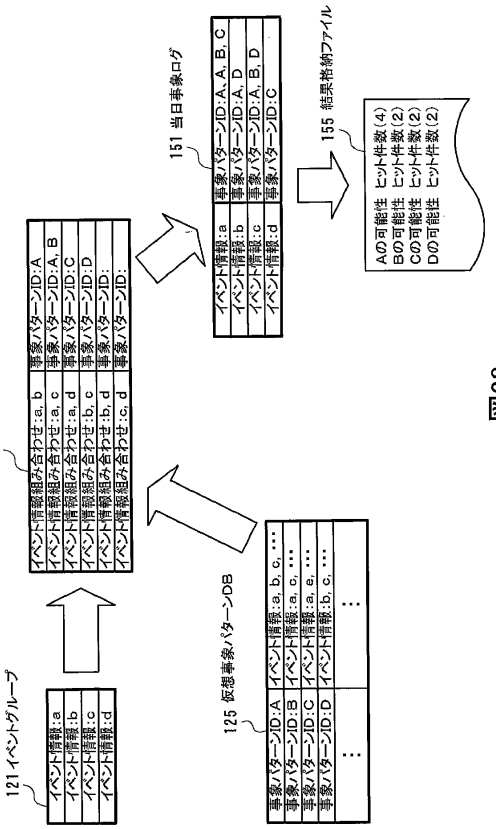


図33

【図３４】

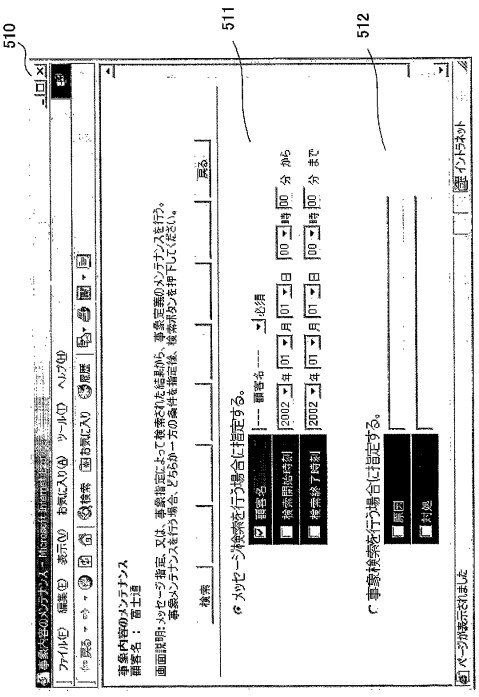


図34

【図３５】

図35

【図３６】

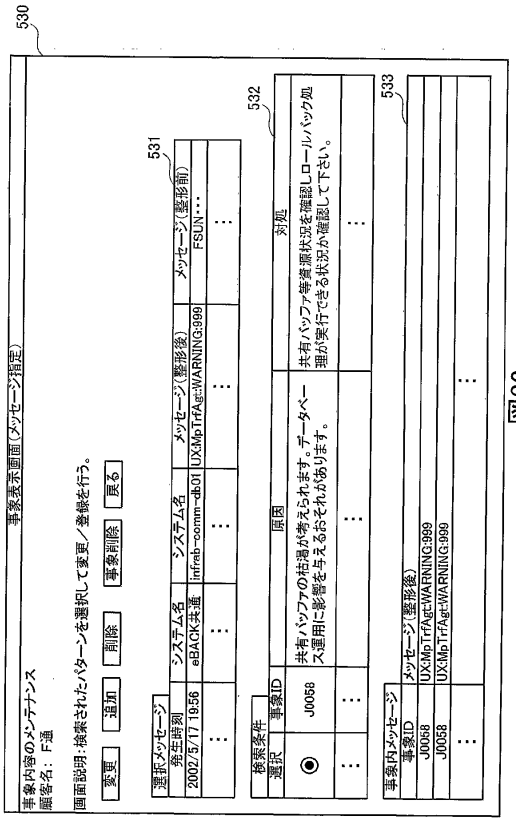


図36

【図３７】

610 履歴テーブル

発生日時	コマンド名	サーバ名
2002/11/2913:30:00	vi/etc/hosts	server1
2002/11/2913:36:00	mail	server1
2002/11/2913:37:00	kill −9 test	server1

図37

【図３８】

620 操作パターンDB

操作パターンコード	操作パターン	現象	原因	対処
s00000001	vi init.ora starup	DB管理プログラムが起動できない	DBブロックサイズの編集が間違えていた。	DBブロックサイズを適切な値に変更する
s00000002	vi /etc//hosts mail	mailコマンドでメールを送信したとき、 "My unqualified host name (cscats01) unknownsleeping for retry"が発生し、応答が遅い。	sendmailのバージョンが8.8.8以降から/etc/hostsに FQDN(省略記述を許さない完全なドメイン名)の設定が必要なため。	/etc/hostsの ホスト名の後にドメイン名を追加するよう、hostsファイルを修正する。

図38

フロントページの続き

(72)発明者　加来　義朗
北海道帯広市西７条北５丁目６番地　株式会社富士通東北海道システムエンジニアリング内
(72)発明者　小野　仁
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　鈴木　洋
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　河嶋　千晶
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

審査官　寺谷　大亮

(56)参考文献　特開平１１－３０８２２２（ＪＰ，Ａ）
特開平１１－２３２１４５（ＪＰ，Ａ）
特開２００２－３４２１８２（ＪＰ，Ａ）
特開平１０－１０５４４０（ＪＰ，Ａ）
特開２００１－２５６０３２（ＪＰ，Ａ）
特開２００１－２５７６７７（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
G06F　13/00
G06F　11/22
G06F　11/34