(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4491482号
(P4491482)
(45)発行日　平成22年6月30日(2010.6.30)　(24)登録日　平成22年4月9日(2010.4.9)

(51)Int.Cl.　F I
G06F 11/20 (2006.01)　G06F 11/20 310F

請求項の数 11　(全 18 頁)

(21)出願番号　特願2007-307106(P2007-307106)
(22)出願日　平成19年11月28日(2007.11.28)
(65)公開番号　特開2009-129409(P2009-129409A)
(43)公開日　平成21年6月11日(2009.6.11)
審査請求日　平成21年8月6日(2009.8.6)

(73)特許権者　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
(74)代理人　100075513
弁理士　後藤　政喜
(74)代理人　100114236
弁理士　藤井　正弘
(74)代理人　100120260
弁理士　飯田　雅昭
(72)発明者　松本　洋和
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】障害回復方法、計算機、クラスタシステム、管理計算機及び障害回復プログラム

(57)【特許請求の範囲】
【請求項１】
業務処理を実行する第１の計算機と、前記第１の計算機によって処理されるデータの複製を保持する第２の計算機とを含むクラスタシステムにおいて、前記第１の計算機で発生した障害を回復する方法であって、
前記第１の計算機は、第１のプロセッサと、前記第１のプロセッサに接続される第１の記憶部と、前記第２の計算機に接続される第１のインタフェースとを備え、
前記第２の計算機は、第２のプロセッサと、前記第２のプロセッサに接続される第２の記憶部と、前記第１の計算機に接続される第２のインタフェースとを備え、
前記第１の記憶部は、前記業務処理で使用されるデータを記憶し、
前記クラスタシステムは、当該クラスタシステムの状態を含むシステム情報を保持し、
前記障害回復方法は、
前記第１の記憶部に記憶されたデータを、前記第２の計算機に送信し、
前記第１の計算機から送信されたデータを、前記第２の記憶部に記憶し、
前記第１の計算機に障害が発生した場合には、前記システム情報に基づいて、前記障害が発生した処理を前記第１の計算機で再開始するか、又は、前記障害が発生した処理を前記第２の計算機が実行するか、を判定し、
前記障害が発生した処理を前記第１の計算機で再開始する場合には、前記第２の記憶部に格納されたデータを前記第２の計算機から前記第１の計算機に送信し、前記第１の計算機に送信されたデータを前記第１の記憶部に記憶し、前記障害が発生した処理を再開始し

、
前記障害が発生した処理を前記第２の計算機が実行する場合には、前記第２の計算機が、前記障害が発生した処理を実行することを特徴とする障害回復方法。
【請求項２】
前記システム情報は、前記第２の計算機の数を含み、
前記障害回復方法は、前記第１の計算機の障害発生時に、前記第２の計算機の数が所定の閾値よりも小さい場合には、前記障害が発生した処理を前記第１の計算機で再開始することを特徴とする請求項１に記載の障害回復方法。
【請求項３】
前記システム情報は、前記第１の計算機で実行される処理を構成する各モジュールによって使用されるデータ量を含み、
前記障害回復方法は、
前記第１の計算機の障害発生時に、障害が発生したモジュールを特定し、
前記特定されたモジュールによって使用されるデータ量を前記システム情報から取得し、
前記取得されたデータ量が所定の閾値よりも小さい場合には、前記障害が発生した処理を前記第１の計算機で再開始することを特徴とする請求項１に記載の障害回復方法。
【請求項４】
前記システム情報は、前記第１の計算機及び前記第２の計算機の負荷情報を含み、
前記障害回復方法は、前記第１の計算機の障害発生時に、前記第１の計算機の負荷が所定の閾値よりも小さい場合には、前記障害が発生した処理を前記第１の計算機で再開始することを特徴とする請求項１に記載の障害回復方法。
【請求項５】
前記障害回復方法は、
前記第１の計算機の障害発生時に、前記第１の計算機の負荷が所定の閾値以上の場合には、最も負荷の少ない計算機を選択し、
前記選択された計算機が、前記障害が発生した処理を実行することを指示することを特徴とする請求項４に記載の障害回復方法。
【請求項６】
業務処理を実行する第１の計算機と、前記第１の計算機によって処理されるデータの複製を保持する第２の計算機とを含むクラスタシステムに含まれる第１の計算機であって、
プロセッサと、前記プロセッサに接続される記憶部と、前記第２の計算機に接続されるインタフェースとを備え、
前記記憶部は、
前記業務処理で使用されるデータを記憶し、
前記クラスタシステムの状態を含むシステム情報を記憶し、
前記プロセッサは、
前記記憶部に記憶されたデータを、前記第２の計算機に送信し、
前記第１の計算機に障害が発生した場合には、前記システム情報に基づいて、前記障害が発生した処理を再開始するか、又は、前記障害が発生した処理を前記第２の計算機が実行するか、を判定し、
前記障害が発生した処理を再開始する場合には、前記第２の計算機から前記第１の計算機によって処理されるデータの複製を取得し、前記取得されたデータを前記記憶部に記憶し、前記障害が発生した処理を再開始し、
前記障害が発生した処理を前記第２の計算機が実行する場合には、前記第２の計算機に前記障害が発生した処理を実行するように指示することを特徴とする計算機。
【請求項７】
前記システム情報は、前記第２の計算機の数を含み、
前記プロセッサは、前記第１の計算機の障害発生時に、前記第２の計算機の数が所定の閾値よりも小さい場合には、前記障害が発生した処理を再開始することを特徴とする請求

項６に記載の計算機。
【請求項８】
前記システム情報は、前記第１の計算機で実行される処理を構成する各モジュールによって使用されるデータ量を含み、
前記プロセッサは、
前記第１の計算機の障害発生時に、障害が発生したモジュールを特定し、
前記特定されたモジュールによって使用されるデータ量を前記システム情報から取得し、
前記取得されたデータ量が所定の閾値よりも小さい場合には、前記障害が発生した処理を再開始することを特徴とする請求項６に記載の計算機。
【請求項９】
前記システム情報は、前記第１の計算機及び前記第２の計算機の負荷情報を含み、
前記プロセッサは、前記第１の計算機の障害発生時に、前記第１の計算機の負荷が所定の閾値よりも小さい場合には、前記障害が発生した処理を再開始することを特徴とする請求項６に記載の計算機。
【請求項１０】
前記プロセッサは、
前記第１の計算機の障害発生時に、前記第１の計算機の負荷が所定の閾値以上の場合には、最も負荷の少ない計算機を選択し、
前記選択された計算機に、前記障害が発生した処理を実行するように指示することを特徴とする請求項９に記載の計算機。
【請求項１１】
業務処理を実行する第１の計算機と、前記第１の計算機によって処理されるデータの複製を保持する第２の計算機と、前記第１の計算機及び前記第２の計算機を管理する管理計算機とを含むクラスタシステムであって、
前記第１の計算機は、第１のプロセッサと、前記第１のプロセッサに接続される第１の記憶部と、前記第２の計算機に接続される第１のインタフェースとを備え、
前記第２の計算機は、第２のプロセッサと、前記第２のプロセッサに接続される第２の記憶部と、前記第１の計算機に接続される第２のインタフェースとを備え、
前記管理計算機は、第３のプロセッサと、前記第３のプロセッサに接続される第３の記憶部と、前記第１の計算機及び前記第２の計算機に接続される第３のインタフェースとを備え、
前記第１の記憶部は、前記業務処理で使用されるデータを記憶し、
前記第３の記憶部は、前記クラスタシステムの状態を含むシステム情報を記憶し、
前記第１の計算機は、前記第１の記憶部に記憶されたデータを前記第２の計算機に送信し、
前記第２の計算機は、前記第１の計算機から送信されたデータを、前記第２の記憶部に記憶し、
前記管理計算機は、
前記第１の計算機に障害が発生した場合には、前記システム情報に基づいて、前記障害が発生した処理を前記第１の計算機で再開始するか、又は、前記障害が発生した処理を前記第２の計算機が実行するか、を判定し、
前記障害が発生した処理を前記第１の計算機で再開始する場合には、前記第１の計算機に前記障害が発生した処理の再開始を指示し、
前記第１の計算機は、前記第２の記憶部に格納されたデータを前記第２の計算機から取得し、前記第２の計算機から取得したデータを前記第１の記憶部に記憶し、前記障害が発生した処理を再開始し、
前記管理計算機は、
前記障害が発生した処理を前記第２の計算機が実行する場合には、前記システム情報に基づいて、前記障害が発生した処理を継続する第２の計算機を選択し、

前記選択された第２の計算機に、前記障害が発生した処理を実行することを指示し、
前記選択された第２の計算機は、前記障害が発生した処理を実行することを特徴とするクラスタシステム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、現用系の計算機と待機系の計算機を含むクラスタシステムの障害を回復する技術に関する。
【背景技術】
【０００２】
従来、不揮発性の共有ディスクによって処理データを保持し、現用系の計算機と待機系の計算機とを含むクラスタシステムでは、現用系のプロセスに障害が発生した場合、プロセスの再開始又は待機系への系切り替えを実行することによって障害を回復させていた。
【０００３】
処理性能を向上させるために不揮発性の共有ディスクの代わりに揮発性メモリを用いるクラスタシステムでは、現用系にプロセス障害が発生すると、データが消失してしまうため、回復処理を実行することができない。そこで、現用系のプロセスに障害が発生した場合の回復手段として、別の計算機に再開始のために必要となるデータの複製を転送し、プロセスの再開始時には、別の計算機に複製したデータを利用して再開始を実行する技術が開示されている（特許文献１参照）。特許文献１に開示された技術では、データを複製するために、転送元となる計算機と転送先となる計算機が循環して配置され、すべての計算機でデータを二重化している。
【特許文献１】特開平９－１６８０１５号公報
【発明の開示】
【発明が解決しようとする課題】
【０００４】
しかし、特許文献１に開示された技術では、データを二重に保護しているのみであるため、プロセス再開始時完了までにデータ複製先に障害が発生すると、回復処理を実行することができなくなってしまう。
【０００５】
また、現用系のプロセス障害時には必ず同じ系によるプロセス再開始を実行し、別系からのデータ転送を試みるため、待機系への系切り替えと比較して処理時間が増大してしまう可能性があった。
【０００６】
本発明の目的は、処理データ消失の可能性を最大限に抑えつつ、回復処理の高速化を図るプロセス回復方法を提供する。
【課題を解決するための手段】
【０００７】
本発明の代表的な一形態では、業務処理を実行する第１の計算機と、前記第１の計算機によって処理されるデータの複製を保持する第２の計算機とを含むクラスタシステムにおいて、前記第１の計算機で発生した障害を回復する方法であって、前記第１の計算機は、第１のプロセッサと、前記第１のプロセッサに接続される第１の記憶部と、前記第２の計算機に接続される第１のインタフェースとを備え、前記第２の計算機は、第２のプロセッサと、前記第２のプロセッサに接続される第２の記憶部と、前記第１の計算機に接続される第２のインタフェースとを備え、前記第１の記憶部は、前記業務処理で使用されるデータを記憶し、前記クラスタシステムは、当該クラスタシステムの状態を含むシステム情報を保持し、前記障害回復方法は、前記第１の記憶部に記憶されたデータを、前記第２の計算機に送信し、前記第１の計算機から送信されたデータを、前記第２の記憶部に記憶し、前記第１の計算機に障害が発生した場合には、前記システム情報に基づいて、前記障害が発生した処理を前記第１の計算機で再開始するか、又は、前記障害が発生した処理を前記

第２の計算機が実行するか、を判定し、前記障害が発生した処理を前記第１の計算機で再開始する場合には、前記第２の記憶部に格納されたデータを前記第２の計算機から前記第１の計算機に送信し、前記第１の計算機に送信されたデータを前記第１の記憶部に記憶し、前記障害が発生した処理を再開始し、前記障害が発生した処理を前記第２の計算機が実行する場合には、前記障害が発生した処理を前記第２の計算機が実行する。

【発明の効果】

【０００８】

本発明の一形態によれば、システムの状態に基づいて、プロセスの再開始と系切り替えのいずれを実行するかを判定することによって、障害復旧処理の高速化及び高信頼化を実現することができる。

【発明を実施するための最良の形態】

【０００９】

以下、本発明の実施の形態を、図面を参照して説明する。

【００１０】

（第１の実施の形態）

図１は、本発明の第１の実施の形態のクラスタシステムの一例を示すシステム構成図である。

【００１１】

本発明の第１の実施の形態のクラスタシステムは、現用系の計算機１、及び、複数の待機系の計算機２～ｎを含む。

【００１２】

現用系及び待機系の各計算機は、処理データ管理部１０１、負荷情報管理部２０１、及びクラスタ情報管理部３０１を有する。クラスタシステムに含まれる現用系及び待機系の各計算機は、同じ構成である。また、現用系の計算機が系切り替えによって待機系の計算機に処理が引き継がれると、処理を引き継いだ待機系の計算機は、以降、現用系の計算機として稼働する。また、現用系として稼働していた計算機は、可能であれば、待機系の計算機として稼働させてもよい。

【００１３】

処理データ管理部１０１は、処理実行部１０２及び処理データ１０３を有する。処理実行部１０２は、要求された処理を実行する。処理データ１０３は、処理実行部１０２によって実行される処理に必要なデータである。また、処理データ１０３は、処理を高速化させるために揮発性のメモリに記憶されている。なお、処理データ１０３は、データベースに格納されていてもよい。

【００１４】

処理データ管理部１０１は、自系が現用系か待機系かを示すクラスタ状態３０４を回復判断テーブル３０３に通知する。処理実行部１０２は、処理管理部１００の各モジュールの回復に必要となるデータ量１０５を計算する。さらに、クラスタ情報管理部３０１にデータ量１０５を通知し、回復判断テーブル３０３に記録する。処理実行部１０２は、さらに、各モジュールの稼働状態を監視する機能と、障害が発生した場合にクラスタ情報管理部３０１にプロセス障害を通知する機能を備える。プロセス障害の通知には、障害が発生したモジュールの情報を含む。モジュールについては、図６にて詳細に説明する。

【００１５】

処理データ管理部１０１は、さらに、他系とデータを送受信するデータ転送部１０４を有する。データ転送部１０４は、処理実行部１０２によって処理された処理データ１０３を、他の計算機に転送し、又は、他の計算機から転送された処理データを受信する。なお、本発明の第１の実施の形態では、現用系の計算機のメモリに格納されている処理データ１０３は、すべての待機系の計算機に転送され、当該計算機のメモリに格納される。

【００１６】

データ転送部１０４による処理データの転送方法は、各計算機に個別にデータを送信するユニキャストであってもよいし、システム内のすべての計算機に対して同時にデータを

送信するマルチキャストであってもよい。マルチキャストによって、転送データ量の削減を図ることができる。

【００１７】

また、データ転送部１０４は、データ転送量に応じて、事前又は転送時にデータを圧縮することなどによって転送量を抑制したり、転送経路を複数使用することによって転送経路を他の処理よりも優先的に利用したりしてもよい。

【００１８】

さらに、本発明の第１の実施の形態では、データ転送部１０４によって他系に処理データを同期転送する。処理データを非同期で転送する場合には、障害発生時に一部の処理データが失われる可能性がある。したがって、データの再生が可能な場合など一部の処理データの欠損が許容されるシステムである場合、又は、さらに上位のシステムなどからデータの再送が可能であれば適用可能である。非同期転送の場合には、他系に処理データを転送後、処理データの格納の完了を待たずに自系の処理を継続できるため、処理性能を向上させることができる。

【００１９】

負荷情報管理部２０１は、負荷情報判断部２０２及び負荷情報転送部２０３を有する。負荷情報判断部２０２は、計算機の負荷情報を判断する。負荷情報転送部２０３は、負荷情報を他系に転送したり、他系から転送された負荷情報を受信したりする。さらに、負荷情報転送部２０３は、自系又は他系の負荷情報である負荷量２０４を回復判断テーブル３０３に一定間隔で通知する。なお、負荷量を一定間隔で他系に通知するのではなく、障害発生時に他系に負荷量を通知してもよい。この場合には、引き継ぎ先の系を他系が判断する構成としてもよい。

【００２０】

クラスタ情報管理部３０１は、クラスタ情報転送部３０２及び回復判断テーブル３０３を有する。クラスタ情報転送部３０２は、クラスタ情報を他系に転送したり、他系から転送されたクラスタ情報を受信したりする。回復判断テーブル３０３は、処理実行部１０２によって処理されたデータ量１０５、クラスタ状態３０４、及び、自系及び他系の負荷量２０４を格納する。

【００２１】

クラスタ情報管理部３０１は、処理データ管理部１０１を監視することによって、自系のプロセス障害を検出する。処理データ管理部１０１の監視は、データ量１０５の通信をハートビートとして利用する方法であってもよいし、負荷量２０４の通信によって負荷量を測定できたか否かを検出する方法であってもよい。また、他の通信によって直接的又は間接的に監視する方法であってもよい。

【００２２】

クラスタ情報管理部３０１は、自系のプロセス障害を検出すると、後述する判断基準に基づいて、プロセス再開始又は系切り替えのいずれかを実行するかを判断する。プロセス再開始を実行する場合には、処理データ管理部１０１にプロセスの再開始を指示する。処理データ管理部１０１は、プロセスを再開始する指示を受け付けると、データ転送部１０４を介して、他系に複製されているデータの転送を要求することによって、プロセス再開始に必要なデータを取得する。データ取得後、障害が発生した処理データ管理部１０１の全部又は一部のプロセスを再開始し、回復を完了する。

【００２３】

一方、クラスタ情報管理部３０１は、系切り替えを実行する場合には、クラスタ情報転送部３０２を介して、系切り替え先となる他系に引継ぎを指示する。引継ぎを指示された他系は、データ転送部１０４によって複製されているデータを取得し、プロセスを実行することによって系切り替えによる回復を完了する。

【００２４】

さらに、クラスタ情報管理部３０１は、クラスタ情報転送部３０２によって他系からのクラスタ情報を一定時間受信できなかった場合には、クラスタ情報を受信できなかった他

系に障害が発生したものと認識する。他系に障害が発生した場合には、複製されている処理データを利用して、プロセスを起動することによって、系切り替えを実行する機能を有する。ここで、他系の障害検出によって実行される系切り替え処理が、障害が発生した系のプロセス再開始又は系切り替えを指示する処理と重複して実行されないように制御する必要がある。例えば、障害が発生した系のプロセス再開始又は系切り替えが完了するために必要な時間だけ待機してもよいし、障害が発生した系で回復処理が実行されていないことを確認してから系切り替えを実行するようにしてもよい。さらに、同時に複数の計算機でプロセスが引き継がれないように、共有ディスク及びＩＰアドレスなどの共有されるリソースが排他制御される仕組みであってもよい。

【００２５】

図２は、本発明の第１の実施の形態のハードウェアの構成を示す図である。

【００２６】

現用系及び待機系の各計算機は、前述したように同じ構成である。各計算機は、ＣＰＵ２１、ディスプレイ装置２２、キーボード２３、マウス２４、ネットワークインタフェースカード（ＮＩＣ）２５、ハードディスク装置２６及びメモリ２７を備える。ＣＰＵ２１、ディスプレイ装置２２、キーボード２３、マウス２４、ネットワークインタフェースカード２５、ハードディスク装置２６及びメモリ２７は、バス２８によって接続される。

【００２７】

現用系及び待機系の各計算機は、ＮＩＣ２５を介してネットワークに接続し、他の計算機と相互に通信する。

【００２８】

ＣＰＵ２１は、メモリ２７に記憶されたプログラムを実行する。メモリ２７は、ＣＰＵ２１によって実行されるプログラム及び当該プログラムの実行に必要なデータを記憶する。メモリ２７は、処理管理部１００、オペレーティングシステム３０、処理データ管理部１０１、負荷情報管理部２０１、クラスタ情報管理部３０１、処理データ１０３、及び、回復判断テーブル３０３を記憶する。メモリ２７は、前述のように、揮発性のメモリである。

【００２９】

処理管理部１００は、オペレーティングシステム３０上で実行されるプログラムである。処理データ管理部１０１、負荷情報管理部２０１、及びクラスタ情報管理部３０１は、処理管理部１００によって呼び出されるプログラムである。処理データ管理部１０１、負荷情報管理部２０１及びクラスタ情報管理部３０１については、図１にて説明した処理を実行する。

【００３０】

処理データ１０３は、業務処理に必要なデータである。処理データ１０３は、前述したように、データベース管理システムによって管理されていてもよい。この場合、データベース管理システムは、メモリ２７に記憶される。回復判断テーブル３０３は、図１にて説明したように、現用系の計算機で発生した障害を回復させるために必要なクラスタ情報などの情報を格納する。

【００３１】

ディスプレイ装置２２は、業務処理の実行結果など各種情報を表示する。キーボード２３及びマウス２４は、利用者からの入力を受け付ける。ＮＩＣ２５は、ネットワークに接続する。ハードディスク装置２６は、メモリ２７に格納される処理データ、及び、メモリ２７にロードされるプログラムなどを格納する。

【００３２】

図３は、本発明の第１の実施の形態の回復判断テーブル３０３の構成を示す図である。

【００３３】

回復判断テーブル３０３は、クラスタ状態判断テーブル３３１、データ量判断テーブル３１１及び負荷状態判断テーブル３２１を含む。

【００３４】

クラスタ状態判断テーブル３３１は、各計算機のクラスタ状態３０４、及び、利用者又はシステムによって設定された残台数の閾値情報を含む。本発明の第１の実施の形態では、クラスタ状態には、「現用系」、「待機系」及びプロセスダウンを含む「ダウン」の三状態が定義されているが、さらに詳細なクラスタ状態を定義してもよい。例えば、待機系として起動中である状態を含んでもよい。この場合には、起動後は待機系としての役割を果たすことから待機系として扱ってもよいし、現在は待機系の役割を果たしていないことから待機系として扱わなくてもよい。

【００３５】

データ量判断テーブル３１１は、処理実行部１０２を構成するモジュールごとのデータ量、利用者又はシステムによって設定されたデータ量の閾値情報、及び、モジュール間の依存関係を表す情報を含む。依存関係は、例えば、図７に示すように、識別子の命名ルールによって表現してもよい。識別子の命名ルールによって依存関係を表す場合には、まず、メインモジュールから直接呼び出される下位モジュールである１段目の各モジュールは、英文字（Ａ－Ｚ）で示される識別子が付与される。さらに、１段目の各モジュールによって呼び出される２段目の各モジュールには、１段目のモジュールの識別子に数字（１－９）を付加した識別子を付与する。なお、モジュール間の依存関係を表す情報は、木構造などの他の手段によって表されてもよい。さらに、モジュール間の依存関係を表す情報は、データ量判断テーブル３１１とは別のテーブルに保持されてもよい。

【００３６】

負荷状態判断テーブル３２１は、各計算機の負荷量２０４を保持する。負荷状態判断テーブル３２１は、利用者又はシステムによって設定された負荷量の閾値情報、及び、各計算機の負荷量を含む。負荷量は、例えば、処理対象のデータ量又は処理の所要時間であってもよいし、データ量などの情報を変数とする計算式によって算出される値であってもよい。

【００３７】

図４は、本発明の第１の実施の形態のクラスタ情報管理部３０１によるクラスタ状態判断テーブル３３１に基づいて障害を回復させる処理の手順を示す図である。

【００３８】

図４に示す障害回復処理では、クラスタ情報管理部３０１によって、すべての計算機が障害などによって停止することによる処理データ１０３の消失を防ぐため、待機系の残り台数が閾値よりも少なくならないように制御する。

【００３９】

ＣＰＵ２１は、自系（現用系）で障害発生を検知した場合には（ステップ４０１）、クラスタ状態判断テーブル３３１を参照し、待機台数の合計を算出する（ステップ４０２）。さらに、クラスタ状態判断テーブル３３１から残台数閾値情報を取得する（ステップ４０３）。

【００４０】

ＣＰＵ２１は、待機台数が０か否かを判定する（ステップ４０４）。待機台数が０の場合には（ステップ４０４の結果が「Ｙ」）、回復に必要なデータが存在しないため、システム回復が不可能と判断し（ステップ４０５）、本処理を終了する。なお、ステップ４０５の処理では、終了以外に、処理データ１０３を不揮発性ディスクに複写するなどのデータ保護処理を実行してもよい。

【００４１】

ＣＰＵ２１は、待機台数が０よりも大きい場合には（ステップ４０４の結果が「Ｎ」）、待機台数の合計が残台数閾値情報以下であるか否かを判定する（ステップ４０６）。待機台数の合計が残台数閾値情報以下の場合には（ステップ４０６の結果が「Ｙ」）、プロセスの再開始を試みる（ステップ４０７）。さらに、プロセスの再開始が成功したか否かを判定する（ステップ４０８）。

【００４２】

一方、ＣＰＵ２１は、待機台数の合計が残台数閾値情報よりも大きい場合（ステップ４

０６）、又は、プロセスの再開始に失敗した場合には（ステップ４０８の結果が「Ｎ」）、待機系に系を切り替える（ステップ４０９）。待機系への系切り替えが完了、又は、プロセスの再開始が成功すると（ステップ４０８の結果が「Ｙ」）、システムを回復させることができる（ステップ４１０）。

【００４３】

図５Ａ及び図５Ｂは、本発明の第１の実施の形態のクラスタ状態判断テーブル３３１に基づいて障害を回復させる処理の一例を示す図である。

【００４４】

図５Ａに示すケース１では、計算機１は現用系、計算機２～４は待機系となっている。計算機１のクラスタ状態判断テーブル３３１には、各計算機のクラスタ状態が格納されている。

【００４５】

ここで、現用系の計算機１に障害が発生した場合には、残台数閾値情報と稼動中の待機系の台数とを比較する。ケース１では、待機系の残り台数は３台であり、閾値（２台）よりも大きいため、待機系への系切り替えを実行する。

【００４６】

図５Ｂに示すケース２では、計算機１は現用系、計算機２、３は待機系、計算機４は障害によるダウン中となっている。現用系の計算機１に障害が発生すると、待機系の残り台数２台であり、残台数閾値情報の値（２台）以下であるため、計算機１は待機系からデータを取得し、プロセスの再開始を試みる。

【００４７】

図６は、本発明の第１の実施の形態のクラスタ情報管理部３０１によるデータ量判断テーブル３１１に基づいて障害を回復させる処理の手順を示す図である。

【００４８】

本発明の第１の実施の形態では、処理実行部１０２は、機能ごとにモジュール単位で分割された構成となっている。処理実行時に最初に実行されるモジュールをメインモジュールとする。また、各モジュールは、機能ごとに階層構造となっており、上位のモジュールが下位のモジュールを作成し、さらに、下位のモジュールに障害が発生したか否かを監視する。処理実行部１０２は、障害が発生した場合には、クラスタ情報管理部３０１に障害の発生したモジュールを通知する。

【００４９】

次に、クラスタ情報管理部３０１は、データ量判断テーブル３１１を参照し、回復が必要となるモジュールを特定する。最下位のモジュールに障害が発生した場合は、当該モジュールを再作成することによって回復させる必要がある。また、下位モジュールを有するモジュールに障害が発生した場合には、すべての下位モジュールもあわせて回復させる必要がある。

【００５０】

各モジュールは、処理の実行時に処理データ１０３を必要とする。プロセスを再開始する場合には、障害が発生したモジュールごとに必要なデータを待機系から取得する必要がある。各モジュールが必要とするデータ量が大きい場合には、データ転送における処理時間が増大し、系切り替えと比較して回復処理に必要な時間が大きくなる場合がある。したがって、データ転送量が多い場合には、系切り替えを実行したほうが高速な回復が可能となる。本処理では、データ転送量に基づいてプロセスを再開始するか系切り替えを実行するかを判断し、システムを回復させる。

【００５１】

ＣＰＵ２１は、自系（現用系）で障害発生を検知した場合には（ステップ４０１）、データ量判断テーブル３１１を参照し、障害モジュール及び障害モジュールに依存関係を有する下位モジュールを特定し、全モジュールのデータ量の合計を算出する（ステップ４２１）。さらに、データ量判断テーブル３１１からデータ量閾値情報を取得する（ステップ４２２）。

【００５２】
ＣＰＵ２１は、データ量の合計がデータ量閾値情報の値よりも小さいか否かを判定する（ステップ４２３）。データ量の合計がデータ量閾値情報の値よりも小さい場合には（ステップ４２３の結果が「Ｙ」）、待機系の計算機から転送されるデータ量が小さいため、プロセスの再開始を試みる（ステップ４０７）。さらに、プロセスの再開始が成功したか否かを判定する（ステップ４０８）。
【００５３】
一方、ＣＰＵ２１は、データ量の合計がデータ量閾値情報の値以上の場合（ステップ４０６の結果が「Ｎ」）、又は、プロセスの再開始に失敗した場合には（ステップ４０８の結果が「Ｎ」）、待機系に系を切り替える（ステップ４０９）。待機系への系切り替えが完了、又は、プロセスの再開始が成功すると（ステップ４０８の結果が「Ｙ」）、システムを回復させることができる（ステップ４１０）。
【００５４】
図６では、障害が発生したモジュールに対してのみプロセスを再開始するか否かを判断する例を示したが、依存関係を有するより上位のモジュールを対象として、再帰的にモジュールを再開始させてもよい。例えば、系切り替えを実行するステップ４０９の処理の前に、上位モジュールの再開始を再帰的に実行するようにすればよい。また、このように再帰的にモジュールを再開始する場合には、データ量の合計を閾値と比較せずに、無条件にプロセスを再開始するようにしてもよい。
【００５５】
また、検知された障害が自プロセス内のメモリ資源枯渇による障害であった場合には、プロセスの再開始によるメモリ状態の初期化によって回復可能な場合がある。したがって、最初に、メインモジュール配下の全モジュールのデータ量を算出し、算出された値に基づいて、プロセスの再開始又は系切り替えのいずれを実行するかを判断する処理を追加してもよい。
【００５６】
図７は、本発明の第１の実施の形態のデータ量判断テーブル３１１に基づいて障害を回復させる処理の一例を示す図である。
【００５７】
図７では、障害（１）及び障害（２）が発生した場合について説明する。また、図７の説明において、プロセスを再開始するか否かを判定するために基準となるデータ量判断テーブル３１１は、図３に示した回復判断テーブル３０３のデータ量判断テーブル３１１を利用する。
【００５８】
障害（１）は、モジュールＢに障害が発生した場合を示している。この場合、まず、障害が発生したモジュールＢに下位モジュールが存在するか否かを、データ量判断テーブル３１１に含まれるモジュール間の依存関係に基づいて判断する。
【００５９】
データ量判断テーブル３１１を参照すると、モジュールＢには、下位モジュールとしてモジュールＢ１及びモジュールＢ２が存在し、当該モジュールの処理データを待機系から転送する必要があることがわかる。そして、モジュールＢ、モジュールＢ１及びモジュールＢ２の処理データ量の合計を算出すると、１５０（＝３０＋７０＋５０）となる。さらに、処理データ量の合計とデータ量判断テーブル３１１に格納された閾値と比較し、プロセスの再開始が必要であるか否かを判断する。障害（１）では、各モジュールのデータ量の合計（１５０）が閾値（１００）よりも大きいため、プロセスを再開始せずに系を切り替える。
【００６０】
一方、障害（２）は、モジュールＣに障害が発生した場合を示している。同様に、データ量判断テーブル３１１からモジュールＣ及び下位モジュールであるモジュールＣ１の処理データの合計値を算出し、閾値と比較する。障害（２）では、各モジュールのデータ量

の合計（３０）が閾値（１００）よりも小さいため、プロセスの再開始を実行する。

【００６１】

図８は、本発明の第１の実施の形態の負荷情報判断部２０２によって障害を回復させる処理の手順を示す図である。

【００６２】

プロセスの再開始又は系切り替えによって障害を回復させる場合に、処理を実行する計算機の負荷が高いと、回復処理に要する時間が増大する可能性が高く、さらに、正常に回復処理を実行できない可能性がある。そこで、図８に示す障害回復処理では、できるだけ負荷の低い計算機で処理が継続されるように回復処理を実行する。

【００６３】

各計算機の負荷量は、所定の基準に基づいて定められた方法によって決定される値である。例えば、負荷量は、一又は複数の情報に重み付けすることによって算出される。負荷量の基準としては、例えば、ＣＰＵ使用率、ネットワーク使用率、処理が完了していないデータ量などが挙げられる。また、重み付けの方法としては、前述した負荷量の基準と過去の実行時間に基づいて算出された値を利用して、事前に定義された算出式を用いる方法などがある。

【００６４】

負荷情報管理部２０１は、負荷量を一定間隔で算出し、負荷情報転送部２０３によって他系に転送する。他系からの負荷量が一定間隔に受信できなかった場合は、当該系の負荷量は高いと判断し、負荷量に最大値を設定する。また、負荷情報管理部２０１は、自系のクラスタ情報管理部３０１に算出された自系の負荷量及び受信した他系の負荷量を通知する。クラスタ情報管理部３０１は、通知された負荷量を回復判断テーブル３０３の負荷情報判断テーブル３２１に格納する。

【００６５】

ＣＰＵ２１は、自系（現用系）で障害発生を検知した場合には（ステップ４０１）、負荷状態判断テーブル３２１を参照し、各計算機の負荷量を取得する（ステップ４４１）。さらに、負荷状態判断テーブル３２１から負荷量閾値情報を取得する（ステップ４４２）。

【００６６】

ＣＰＵ２１は、自系の負荷量が負荷量閾値情報の値よりも小さいか否か、又は、自系の負荷量が最も低いか否かを判定する（ステップ４４３）。自系の負荷量が負荷量閾値情報の値よりも小さい場合、又は、自系の負荷量が最も低い場合には（ステップ４４３の結果が「Ｙ」）、プロセスの再開始を試みる（ステップ４０７）。さらに、プロセスの再開始が成功したか否かを判定する（ステップ４０８）。

【００６７】

一方、ＣＰＵ２１は、自系の負荷量が負荷量閾値情報の値以上の場合、かつ、自系の負荷量が最も低くない場合（ステップ４４３の結果が「Ｎ」）、又は、プロセスの再開始に失敗した場合には（ステップ４０８の結果が「Ｎ」）、最も負荷の低い待機系に系切り替えを実行する（ステップ４４４）。待機系への系切り替えが完了、又は、プロセスの再開始が成功すると（ステップ４０８の結果が「Ｙ」）、システムを回復させることができる（ステップ４１０）。

【００６８】

図９Ａ及び図９Ｂは、本発明の第１の実施の形態の負荷情報判断部２０２によって障害を回復させる処理の一例を示す図である。

【００６９】

負荷量は、基準となる負荷量を１００とした場合の相対的な値とし、値が大きいほど負荷が高いものとする。

【００７０】

図９Ａに示すケース１では、負荷量の高い計算機１に障害が発生した場合の例を示している。計算機１の負荷量（７０）は閾値（４０）よりも大きく、他の待機系の計算機のほ

うが計算機１よりも負荷量が小さいため、待機系に切り替える。この場合、系切り替え先は最も負荷量の低い計算機３となる。

【００７１】

図９Ｂに示すケース２では、負荷量の低い計算機１に障害が起こった場合の例を示している。計算機１の負荷量（２０）は閾値（４０）よりも小さいため、プロセスの再開始を実行する。なお、計算機１の負荷量が閾値以上の場合であっても、計算機１が最も負荷量の低い計算機であるため、プロセスの再開始を実行する。

【００７２】

図１０は、本発明の第１の実施の形態の現用系障害時の一連の回復処理の手順を示す図である。

【００７３】

図１０に示した回復処理は、図４、図６及び図８に示した手順を組み合わせたものである。各ステップの説明については、前述したとおりである。

【００７４】

本処理は、クラスタ情報管理部３０１によって、自系（現用系）の計算機の障害が検知された場合に実行される（ステップ４０１）。

【００７５】

ＣＰＵ２１は、まず、クラスタ状態判断テーブル３３１を参照し、待機系の計算機の台数及び残台数閾値情報の値を比較する（ステップ４０２～４０６）。処理データ１０３の消失を防ぐことを最優先とするため、現用系のデータを保持する待機系の計算機が一定台数以上稼働するように制御する。

【００７６】

続いて、ＣＰＵ２１は、データ量判断テーブル３１１を参照し、障害を回復するために待機系から取得するデータ量とデータ量閾値情報の値とを比較する（ステップ４２１～４２３）。そして、転送されるデータ量がデータ量閾値情報の値よりも少ない場合には、プロセスの再開始を試みる（ステップ４０７）。転送されるデータ量が少ないほど、プロセスを再開始するために必要な時間が短くなるからである。

【００７７】

最後に、ＣＰＵ２１は、負荷状態判断テーブル３２１を参照し、各計算機の負荷量と負荷量閾値情報の値とを比較する（ステップ４４１～４４３）。自系の計算機の負荷量が負荷量閾値情報の値よりも小さい場合、又は、自系の計算機の負荷がシステム内で最も低い場合には、プロセスの再開始を試みる。自系の負荷量が負荷量閾値情報の値以上の場合、かつ、自系の負荷量がシステム内で最も低くない場合、又は、プロセスの再開始を失敗した場合には（ステップ４４３の結果が「Ｎ」）、最も負荷の低い待機系を取得して系切り替えを実行する（ステップ４４４）。

【００７８】

本発明の第１の実施の形態によれば、現用系が回復するために必要なデータをすべての待機系が保持することによって、プロセス回復完了までに連続的に障害が発生した場合であってもデータ消失を防ぐことができる。

【００７９】

また、本発明の第１の実施の形態によれば、プロセスの障害回復手段として、システムの状態に基づいて、プロセスの再開始と系切り替えのいずれかを実施することによって、障害復旧処理の高速化及び高信頼化を実現することができる。

【００８０】

（第２の実施の形態）

本発明の第１の実施の形態では、回復判断テーブル３０３を各計算機が保持していたが、本発明の第２の実施の形態では、管理計算機が回復判断テーブル３０３を保持する。さらに、管理計算機によってプロセスの障害回復方法が決定され、各計算機に指示される。

【００８１】

図１１は、本発明の第２の実施の形態のクラスタシステムの一例を示すシステム構成図

である。

【００８２】

本発明の第２の実施の形態のクラスタシステムは、現用系及び待機系の計算機（１～ｎ）以外に管理計算機１１を含む。現用系及び待機系の計算機（１～ｎ）と管理計算機１１とは、ネットワークを介して接続される。

【００８３】

管理計算機１１は、クラスタ状態判断テーブル３３１及び負荷状態判断テーブル３２１を保持し、現用系に障害が発生した場合に、プロセスを再開始するか待機系に系切り替えを実行するかを判断する。また、待機系に切り替える場合には、処理を引き継ぐ計算機を選択する。

【００８４】

管理計算機１１のハードウェア構成は、図２に示した計算機のハードウェア構成と同様であって、ＣＰＵ、メモリ、ＮＩＣ及び入出力装置などを備える。なお、管理計算機１１は、仮想計算機上で実行されるプログラムによって実現されてもよい。

【００８５】

管理計算機１１は、回復判断テーブル３０３、データ量取得部１０８、クラスタ情報転送部３０２、負荷情報転送部２０３及び障害回復部１１０を含む。

【００８６】

回復判断テーブル３０３は、本発明の第１の実施の形態と同様に、データ量判断テーブル３１１、クラスタ状態判断テーブル３３１及び負荷状態判断テーブル３２１を含む。

【００８７】

なお、データ量判断テーブル３１１は、他の情報と比較して更新頻度が多く、管理計算機に格納されたデータ量判断テーブル３１１を随時更新すると、ネットワークトラフィックが増大し、処理効率が悪化するおそれがあるため、各計算機に格納されている。本発明の第２の実施の形態では、現用系の計算機に格納されたデータ量判断テーブル３１１の情報を定期的に管理計算機１１が取得することによって、ネットワークトラフィックの増大を抑制する。

【００８８】

データ量取得部１０８は、現用系の計算機に格納されたデータ量判断テーブル３１１から情報を取得し、管理計算機１１のデータ量判断テーブル３１１に情報を格納する。

【００８９】

クラスタ情報転送部３０２は、現用系及び待機系の計算機から送信されたクラスタ情報を受信し、管理計算機１１のクラスタ状態判断テーブル３３１に受信したクラスタ情報を格納する。

【００９０】

負荷情報転送部２０３は、現用系及び待機系の計算機から送信された負荷情報を受信し、管理計算機１１の負荷状態判断テーブル３２１に受信した負荷情報を格納する。

【００９１】

障害回復部１１０は、現用系の計算機に障害が発生すると、回復判断テーブル３０３に格納された情報に基づいて、システムを回復させる。なお、管理計算機１１で実行される回復処理は、図１０に示した本発明の第１の実施の形態の回復処理と同様である。

【００９２】

本発明の第２の実施の形態によれば、本発明の第１の実地の形態と同様に、現用系が回復するために必要なデータをすべての待機系が保持することによって、プロセス回復完了までに連続的に障害が発生した場合であってもデータ消失を防ぐことができる。

【００９３】

また、本発明の第２の実施の形態によれば、各計算機の情報を一元管理されるため、回復に必要な情報をすべての計算機で共有する必要がない。したがって、回復に必要な情報を転送するために必要なネットワークのトラフィックを軽減することができる。

【００９４】

さらに、本発明の第２の実施の形態によれば、各計算機がシステム内の他の計算機を監視する必要がなくなるため、各計算機の負荷を軽減することができる。

【図面の簡単な説明】

【００９５】

【図１】本発明の第１の実施の形態のクラスタシステムの一例を示すシステム構成図である。

【図２】本発明の第１の実施の形態のハードウェアの構成を示す図である。

【図３】本発明の第１の実施の形態の回復判断テーブルの構成を示す図である。

【図４】本発明の第１の実施の形態のクラスタ状態判断テーブルに基づいて障害を回復させる処理の手順を示す図である。

【図５Ａ】本発明の第１の実施の形態のクラスタ状態判断テーブルに基づいて障害を回復させる処理の一例を示す図である（プロセス再開始）。

【図５Ｂ】本発明の第１の実施の形態のクラスタ状態判断テーブルに基づいて障害を回復させる処理の一例を示す図である（系切り替え）。

【図６】本発明の第１の実施の形態のデータ量判断テーブルに基づいて障害を回復させる処理の手順を示す図である。

【図７】本発明の第１の実施の形態のデータ量判断テーブルに基づいて障害を回復させる処理の一例を示す図である。

【図８】本発明の第１の実施の形態の負荷情報判断部によって障害を回復させる処理の手順を示す図である。

【図９Ａ】本発明の第１の実施の形態の負荷情報判断部によって障害を回復させる処理の一例を示す図である（プロセス再開始）。

【図９Ｂ】本発明の第１の実施の形態の負荷情報判断部によって障害を回復させる処理の一例を示す図である（系切り替え）。

【図１０】本発明の第１の実施の形態の現用系障害時の一連の回復処理の手順を示す図である。

【図１１】本発明の第２の実施の形態のクラスタシステムの一例を示すシステム構成図である。

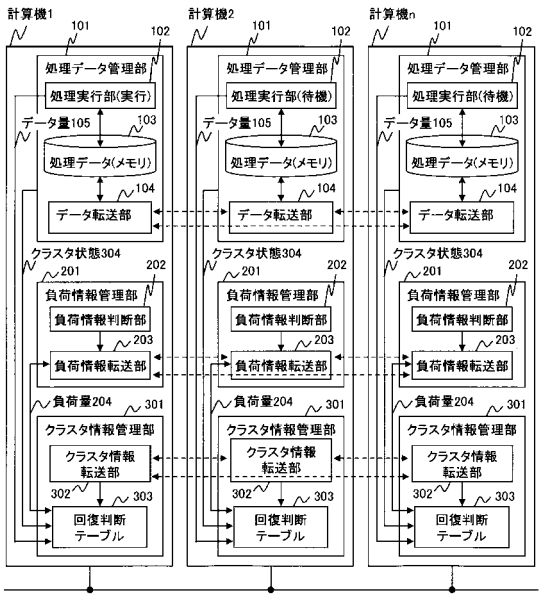
【符号の説明】

【００９６】

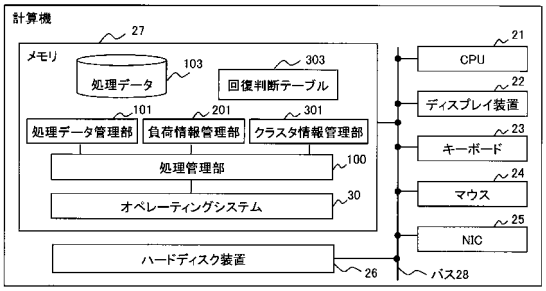
１～ｎ　計算機
１１　管理計算機
２１　ＣＰＵ
２２　ディスプレイ装置
２３　キーボード
２４　マウス
２５　ネットワークインタフェースカード
２６　ハードディスク装置
２７　メモリ
１００　処理管理部
１０１　処理データ管理部
１０２　処理実行部
１０３　処理データ
１０４　データ転送部
１０８　データ量取得部
１１０　障害回復部
２０１　負荷情報管理部
２０２　負荷情報判断部
２０３　負荷情報転送部
３０１　クラスタ情報管理部

３０２　　クラスタ情報転送部
３０３　　回復判断テーブル
３１１　　クラスタ状態判断テーブル
３２１　　負荷状態判断テーブル
３３１　　クラスタ状態判断テーブル

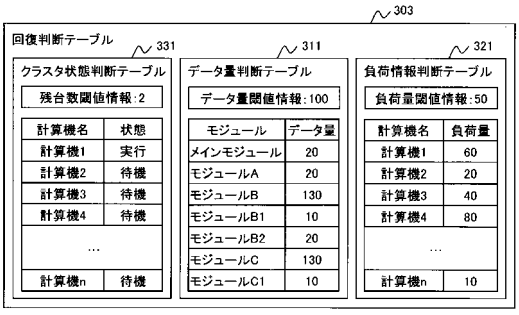
【図１】



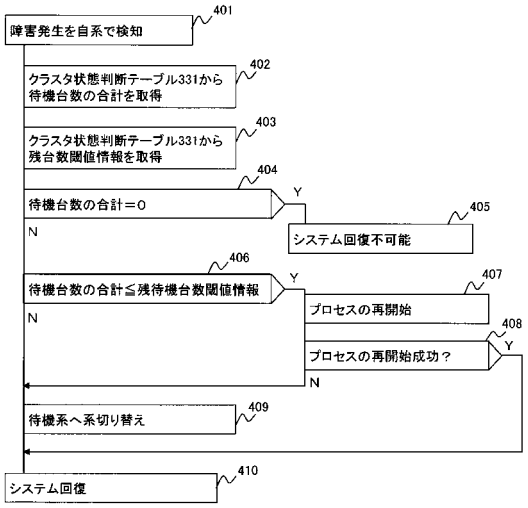
【図２】



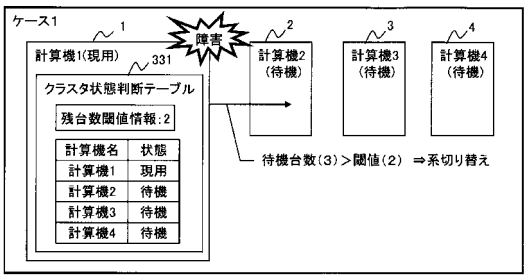
【図３】



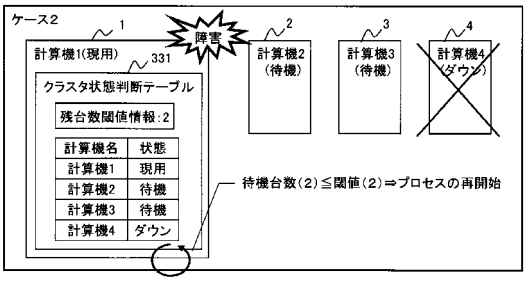
【図４】



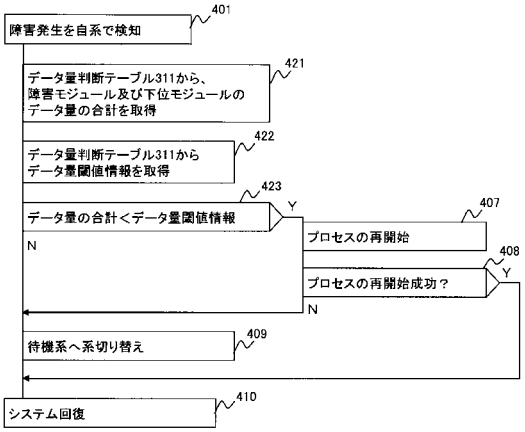
【図５Ａ】



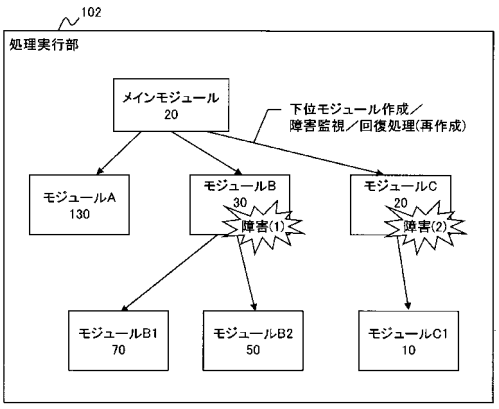
【図５Ｂ】



【図６】

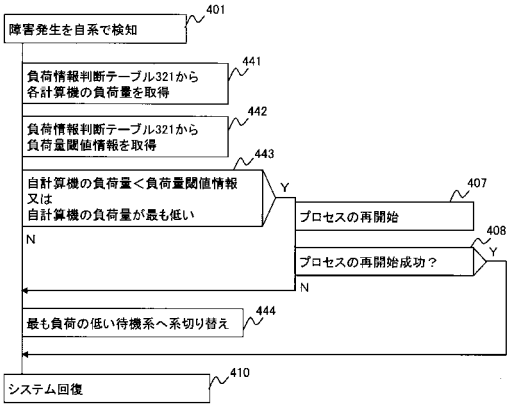


【図７】

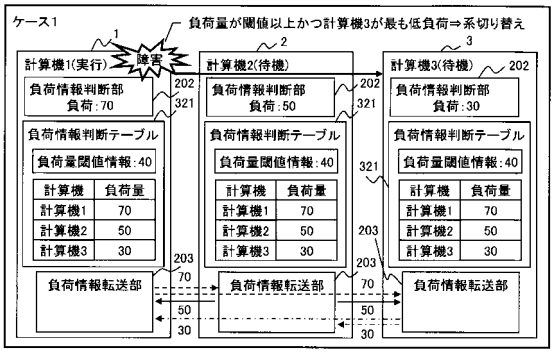


障害(1):30+70+50 ≧ 100(閾値) ⇒系切り替え
障害(2):20+10 ＜ 100(閾値) ⇒プロセスの再開始

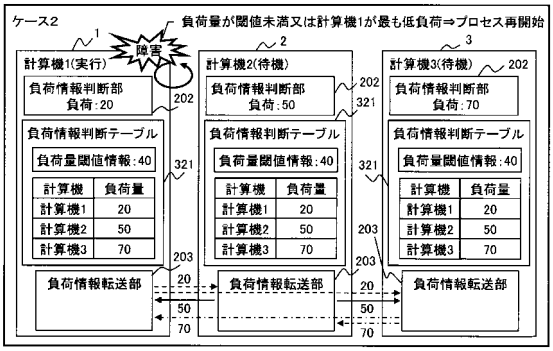
【図８】



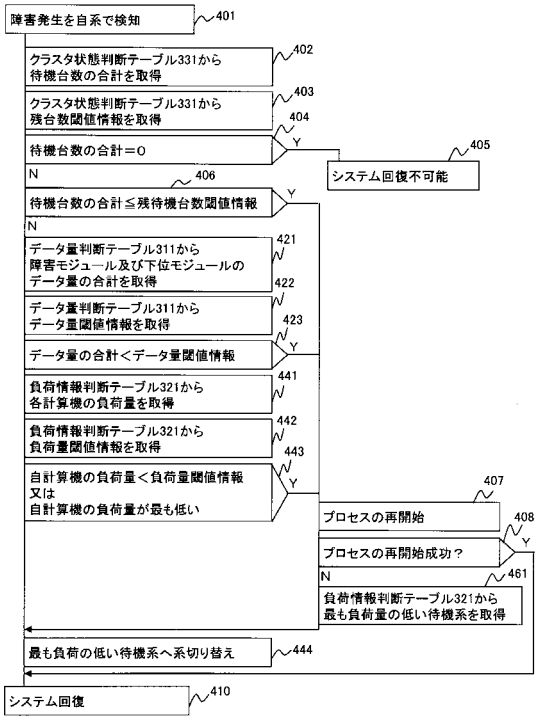
【図９Ａ】



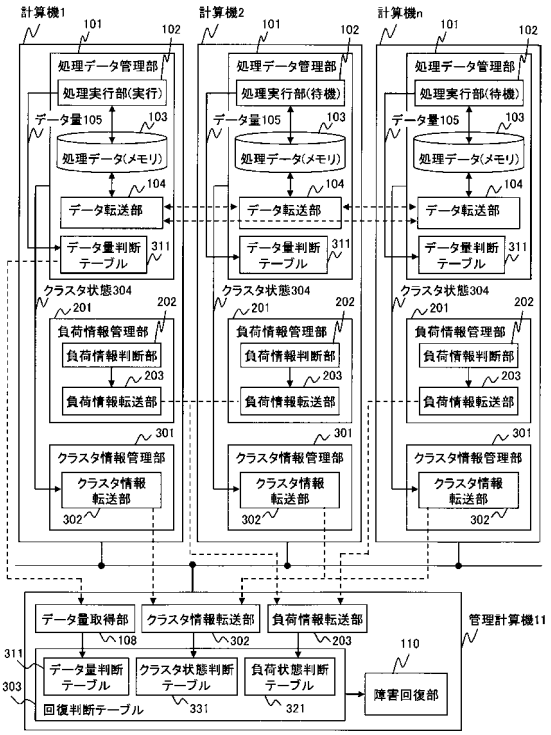
【図９Ｂ】



【図１０】



【図１１】



フロントページの続き

(72)発明者　馬場　恒彦
東京都国分寺市東恋ヶ窪一丁目２８０番地　株式会社日立製作所　中央研究所内
(72)発明者　浜田　真二
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
(72)発明者　市村　高志
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　高橋　則明
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

審査官　漆原　孝治

(56)参考文献　特開平０５－２５０１９７（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１１／２０