

(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2006-172050
(P2006-172050A)

(43) 公開日 平成18年6月29日(2006.6.29)

(51) Int. Cl. F I テーマコード (参考)
G06F 11/18 (2006.01) G06F 11/18 310G 5B034

審査請求 未請求 請求項の数 5 O L (全 7 頁)

(21) 出願番号	特願2004-362422 (P2004-362422)	(71) 出願人	399076312 安川情報システム株式会社 福岡県北九州市八幡西区東王子町5番15号
(22) 出願日	平成16年12月15日(2004.12.15)	(71) 出願人	000006622 株式会社安川電機 福岡県北九州市八幡西区黒崎城石2番1号
		(72) 発明者	石田 常竹 福岡県北九州市八幡西区東王子町5番15号 安川情報システム株式会社内
		Fターム(参考)	5B034 AA01 CC01 CC02 DD01 DD02 DD05 DD07

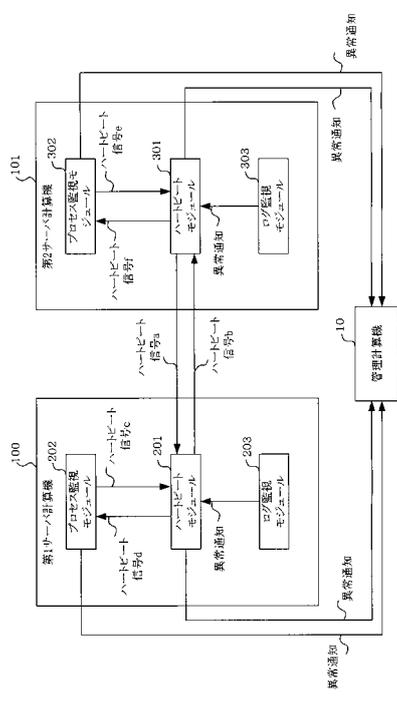
(54) 【発明の名称】 ホットスタンバイ式2重化システム

(57) 【要約】

【課題】 ホットスタンバイ式2重化システムを汎用性が有り安価で柔軟性に富むシステムに構築する。

【解決手段】 サーバ計算機100、101の各々に、他方のサーバ計算機の状態を監視すると共に、自サーバ計算機の状態を他方のサーバ計算機に通知するハートビートモジュール201、301と、自サーバ計算機内のプロセスを監視し、プロセスの異常を検出した場合はハートビートモジュール201、301にプロセス異常の検出を通知するプロセス監視モジュール202、302と、自サーバ計算機内のログを監視し、異常ログを検出した場合は前記ハートビートモジュール201、301に異常ログの検出を通知するログ監視モジュール203、303とを備える。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

ネットワークにより接続されている同一の動作をする 2 つのサーバ計算機を有するホットスタンバイ式 2 重化システムにおいて、

前記 2 つのサーバ計算機の各々に、

前記ネットワークに接続された他方のサーバ計算機が正常に動作しているかどうかの状態を監視すると共に、自サーバ計算機の状態を前記他方のサーバ計算機に通知するハートビートモジュールと、

前記自サーバ計算機内のプロセスを監視し、プロセスの異常を検出した場合は前記ハートビートモジュールにプロセス異常の検出を通知するプロセス監視モジュールと、

前記自サーバ計算機内のログを監視し、異常ログを検出した場合は前記ハートビートモジュールに異常ログの検出を通知するログ監視モジュールとを備えることを特徴とするホットスタンバイ 2 重化システム。

【請求項 2】

前記ハートビートモジュールは前記他方のサーバ計算機のハートビートモジュールが送信するハートビート信号を受信し、

該ハートビート信号を一定時間内に受信できなかった場合は前記他方のサーバ計算機が正常に動作していないと判断して前記自サーバ計算機をマスタとし、前記ハートビート信号を一定時間内に受信できた場合は前記他方のサーバ計算機の状態が正常に動作していると判断して前記自サーバ計算機をスレーブとすることを特徴とする請求項 1 記載のホットスタンバイ式 2 重化システム。

【請求項 3】

前記ハートビートモジュールは前記プロセス監視モジュールが送信するハートビート信号を受信し、該ハートビート信号を一定時間内に受信できなかった場合、

および前記ログ監視モジュールからの異常通知を受信した場合は前記ホットスタンバイ式 2 重化システム全体を管理する管理計算機に異常を通知することを特徴とする請求項 1 記載のホットスタンバイ式 2 重化システム。

【請求項 4】

前記プロセス監視モジュールは、前記ハートビートモジュールが送信するハートビート信号を受信し、該ハートビート信号を一定時間内に受信できなかった場合、

および監視対象のプロセスが異常かどうかチェックしプロセスの異常を検出した場合は前記ホットスタンバイ式 2 重化システム全体を管理する管理計算機に異常を通知することを特徴とする請求項 1 記載のホットスタンバイ式 2 重化システム。

【請求項 5】

前記ログ監視モジュールは、常時監視しているログに異常があった場合、前記ハートビートモジュールに異常を通知することを特徴とする請求項 1 に記載のホットスタンバイ式 2 重化システム。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、計算機システムのシステム構成装置であるサーバ計算機の運用切替方式に特徴を有するホットスタンバイ式 2 重化システムに関するものである。

【背景技術】**【0002】**

高信頼性が必要とされる計算機システムでは、システム構成装置のサーバ計算機 2 台を同じ構成にし、片方を作動させもう片方は同じ動作をしながら待機させておくことがある。このとき作動しているサーバ計算機をマスタといい、待機しているサーバ計算機をスレーブという。特にサーバ計算機 2 台を全く同一の動作をさせ、マスタに障害が発生した場合、即座にスレーブに動作が引き継がれるようにしたシステムをホットスタンバイ式 2 重化システムという。特許文献 1 では、他方のシステム構成装置の状態を監視するために専

10

20

30

40

50

用のハードウェアを準備し、システムの高信頼性を実現している。

【特許文献1】特開平5-233579号公報(第1図)

【発明の開示】

【発明が解決しようとする課題】

【0003】

従来のホットスタンバイ式2重化システムは、専用のハードウェアを準備する必要があるために、汎用性が低く、かつ非常に高価なシステムとなる。さらに、ハードウェアまたはソフトウェアを監視する回路等の障害に備え、回路自体を冗長化する必要がある。また、業務アプリケーションの動作の監視をする場合、業務アプリケーションプロセスの追加があると、ハードウェアの追加や変更または業務アプリケーションプロセス自体の改造が必要となり、柔軟性に欠けるという問題がある。

10

【0004】

本発明は、上記課題に鑑みてなされたものであり、ホットスタンバイ式2重化システムを汎用性が高く安価で柔軟性に富むシステムを構築することを目的とする。

【課題を解決するための手段】

【0005】

上記課題を解決するため、本発明は、ネットワークにより接続されている同一の動作をする2つのサーバ計算機を有するホットスタンバイ式2重化システムにおいて、前記2つのサーバ計算機の各々に、前記ネットワークに接続された他方のサーバ計算機が正常に動作しているかどうかの状態を監視すると共に、自サーバ計算機の状態を前記他方のサーバ計算機に通知するハートビートモジュールと、前記自サーバ計算機内のプロセスを監視し、プロセスの異常を検出した場合は前記ハートビートモジュールにプロセス異常の検出を通知するプロセス監視モジュールと、前記自サーバ計算機内のログを監視し、異常ログを検出した場合は前記ハートビートモジュールに異常ログの検出を通知するログ監視モジュールとを備えることを特徴としている。

20

また、前記ハートビートモジュールは、前記他方のサーバ計算機のハートビートモジュールが送信するハートビート信号を受信し、該ハートビート信号を一定時間内に受信できなかった場合は前記他方のサーバ計算機が正常に動作していないと判断して前記自サーバ計算機をマスタとし、前記ハートビート信号を一定時間内に受信できた場合は前記他方のサーバ計算機の状態が正常に動作していると判断して前記自サーバ計算機をスレーブとすることを特徴としている。

30

【0006】

また、前記ハートビートモジュールは、前記プロセス監視モジュールが送信するハートビート信号を受信し、該ハートビート信号を一定時間内に受信できなかった場合、

および前記ログ監視モジュールからの異常通知を受信した場合は前記ホットスタンバイ式2重化システム全体を管理する管理計算機に異常を通知することを特徴としている。

また、前記プロセス監視モジュールは、前記ハートビートモジュールが送信するハートビート信号を受信し、該ハートビート信号を一定時間内に受信できなかった場合、

および監視対象のプロセスが異常かどうかチェックしプロセスの異常を検出した場合は前記ホットスタンバイ式2重化システム全体を管理する管理計算機に異常を通知することを特徴としている。

40

また、前記ログ監視モジュールは、常時監視しているログに異常があった場合、前記ハートビートモジュールに異常を通知することを特徴としている。

【発明の効果】

【0007】

本発明によれば、ホットスタンバイ式2重化システムを、サーバ計算機が正常に動作しているかを監視するハートビートモジュールと、サーバ計算機内のプロセスを監視するプロセス監視モジュールと、サーバ計算機内のログを監視するログ監視モジュールで実現することにより汎用性が高く安価で柔軟なシステムを構築することができる。

【発明を実施するための最良の形態】

50

【 0 0 0 8 】

以下、図面を参照して、本発明の実施例の詳細を説明する。

【 実施例 1 】

【 0 0 0 9 】

図 1 は、本発明のホットスタンバイ式 2 重化システムの構成図である。図 1 において、10 はシステム全体を管理する管理計算機、100、101 はそれぞれ第 1 サーバ計算機、第 2 サーバ計算機である。それぞれのサーバ計算機 100、101 には、ハートビートモジュール 201、301、プロセス監視モジュール 202、302、ログ監視モジュール 203、303 がある。これら各モジュールはソフトウェアである。また、これら各モジュールの動作は同一であるので、第 1 サーバ計算機 100 を中心に説明する。

10

ハートビートモジュール 201 は、ハートビート信号 b を第 2 サーバ計算機 101 上のハートビートモジュール 301 に送信する。同様に第 2 サーバ計算機 101 上のハートビートモジュール 301 は、ハートビート信号 a を第 1 サーバ計算機 100 上のハートビートモジュール 201 へ送信する。

ここでハートビート信号とは、計算機やプロセスが正常に稼動していることを外部に知らせる信号で、ネットワークで送信可能なデータである。ハートビート信号を使用する複数の計算機やプロセスが相互に取り決めた一定時間の周期でハートビート信号を送信する。ハートビート信号を受信する計算機やプロセスは、取り決めた一定時間内に信号が受信できることで送信元の正常動作を確認する。

第 1 サーバ計算機 100 と第 2 サーバ計算機 101 はハートビート信号 a、b を使用し相互監視を行い、どちらがマスタ動作をするかを決定する。

20

また、ハートビートモジュール 201 とプロセス監視モジュール 202 はハートビート信号 c、d を使用し相互監視を行い、異常が発生すれば管理計算機 10 へ異常を通知し、ネットワークを遮断する。

【 0 0 1 0 】

プロセス監視モジュール 202 は、監視対象のプロセスが存在するかどうかと正常な状態であるかどうかを定期的に確認する。監視対象のプロセスが存在しない、または、消滅せずにシステム内に残っているのに動作しない、いわゆるゾンビプロセスのような異常な状態となった場合は、管理計算機 10 へ異常を通知し、ネットワークを遮断する。

なお、監視対象のプロセスはシステム管理者がシステムを起動する前にあらかじめ設定しておく。

30

プロセス監視モジュール 202 がハートビートモジュール 201 の異常を検出した場合は、管理計算機 10 へ異常を通知し、ネットワークを遮断する。

【 0 0 1 1 】

ログ監視モジュール 203 は、第 1 サーバ計算機内のオペレーティングシステムが出力するシステムに関するログとハートビートモジュール 201 とプロセス監視モジュール 202 が出力するログを監視する。監視しているログに事前に収集した異常ログに関する文字列が出現した場合、ハートビートモジュールに異常を通知する。

異常ログの文字列はシステム管理者がシステムを起動する前にあらかじめ設定しておく。どの文字列が異常ログであるかはオペレーティングシステムや監視するプロセスにより異なる。

40

【 0 0 1 2 】

図 2 はハートビートモジュール 201 の初期処理のフローチャートである。

ハートビート信号 a、b を使用して、第 1 サーバ計算機 100 と第 2 サーバ計算機 101 のどちらがマスタとして動作するかを決定する。

ステップ S 1 でハートビートモジュール 201 は第 2 サーバ計算機からのハートビート信号 a を受信する。ステップ S 2 で一定時間内に受信できたかどうかチェックし、一定時間に受信できなかった場合はステップ S 3 に進み第 1 サーバ計算機をマスタとする。すなわち、ステップ S 3 に進む場合は第 2 サーバ計算機 101 が正常に動作していないと判断する。ステップ S 2 でハートビート信号を一定時間内に受信できた場合は、第 2 サーバ計

50

算機 101 が正常に動作していると判断しステップ S4 に進み、第 1 サーバ計算機をスレーブとする。

このフローチャートはハートビートモジュール 201 について記載してあるが、ハートビートモジュール 301 も同様の処理を行う。

なお、第 1 サーバ計算機 100 と第 2 サーバ計算機 101 がほぼ同時に電源が投入され、ハートビートモジュール 201、301 がほぼ同時に初期処理を開始すると 2 つのサーバ計算機のどちらがマスタとなるか分からないため、最初にマスタとしたいサーバ計算機が決まっている場合は、タイマ等を利用してマスタとしたいサーバ計算機の電源を先に投入する。

【0013】

図 3 は、第 1 サーバ計算機 100 がマスタとして動作する場合のプロセス監視モジュールとハートビートモジュールとログ監視モジュールのフローチャートである。

ハートビートモジュール 201 はステップ S11 でプロセス監視モジュール 202 と第 2 サーバ計算機 101 上で動作するハートビートモジュール 301 にハートビート信号 d、b を送信する。また、プロセス監視モジュール 202 からのハートビート信号 c を受信する。第 2 サーバ計算機 101 からのハートビート信号 a は受信しない。これは、第 2 サーバ計算機がどのような状態でもマスタとして動き続けるためである。

ステップ S12 でプロセス監視モジュール 202 からのハートビート信号 c が一定時間内に受信できたかどうかチェックし、できなかった場合ステップ S15 に進み管理計算機 10 に異常を通知し、ステップ S16 でネットワークを遮断する。

ステップ S12 で一定時間内にプロセス監視モジュール 202 からのハートビート信号 c を受信した場合は、ステップ S13、S14 でログ監視モジュール 203 から異常通知が到着しているかどうか確認する。異常通知を受信した場合はステップ S15 に進み異常処理を実施する。異常通知を受信しなかった場合は、ステップ S11 へ戻る。

【0014】

プロセス監視モジュール 202 は、ステップ S21 でハートビートモジュール 201 との間でハートビート信号 c、d を送受信する。ステップ S22 でハートビートモジュール 201 からのハートビート信号 d をチェックし、信号が一定時間内に受信できなかった場合は、ステップ S24 に進み、管理計算機に異常を通知し、ステップ S25 でネットワークを遮断する。ハートビート信号 d を一定時間内に受信できた場合は、ステップ S23 で監視対象のプロセスが異常かどうかをチェックし、プロセスの異常を検出した場合はステップ S24 に進み異常処理を実施する。

ログ監視モジュール 203 は、常時監視しているログに異常ログが発生した場合、ステップ S31 で異常通知をハートビートモジュール 201 に送信する。

【0015】

図 4 は、第 1 サーバ計算機 100 がスレーブとして動作する場合のプロセス監視モジュールとハートビートモジュールとログ監視モジュールのフローチャートである。

図 3 のフローチャートと異なる点はハートビートモジュール 201 のステップ S51、S52、S53 だけである。

ハートビートモジュール 201 はステップ S51 でプロセス監視モジュール 202 と第 2 サーバ計算機 101 上で動作するハートビートモジュール 301 にハートビート信号 d、b を送信する。また、プロセス監視モジュール 202 からのハートビート信号 c と第 2 サーバ計算機 101 からのハートビート信号 a を受信する。

ステップ S52 でプロセス監視モジュール 202 からのハートビート信号 c、第 2 サーバ計算機 101 からのハートビート信号 a が一定時間内に受信できたかどうかチェックし、第 2 サーバ計算機 101 からのハートビート信号 a が受信できなかった場合はステップ S53 に進み、第 1 サーバ計算機をマスタに切り替える。

ステップ S52 でプロセス監視モジュール 202 からのハートビート信号 c が受信できなかった場合、ハートビート信号 c、a どちらも正常に受信できた場合の処理は図 3 と同一である。

10

20

30

40

50

なお、図3、図4共第1サーバ計算機100側の説明をしたが、第2サーバ計算機101の処理も同一である。

このように、ホットスタンバイ式2重化システムを、サーバ計算機が正常に動作しているかを監視するハートビートモジュールと、サーバ計算機内のプロセスを監視するプロセス監視モジュールと、サーバ計算機内のログを監視するログ監視モジュールで実現することにより汎用性が高く安価で柔軟なシステムを構築することができる。

【図面の簡単な説明】

【0016】

【図1】本発明のホットスタンバイ式2重化システムの構成図

【図2】ハートビートモジュールの初期処理のフローチャート

10

【図3】第1サーバ計算機がマスタとして動作する場合のプロセス監視モジュールとハートビートモジュールとログ監視モジュールのフローチャート

【図4】第1サーバ計算機100がスレーブとして動作する場合のプロセス監視モジュールとハートビートモジュールとログ監視モジュールのフローチャート

【符号の説明】

【0017】

10 管理計算機

100、101 第1、第2サーバ計算機

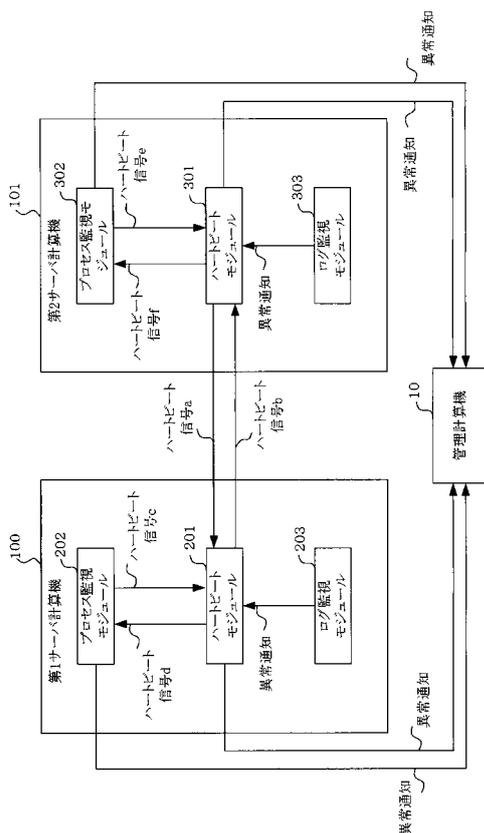
201、301 ハートビートモジュール

202、302 プロセス監視モジュール

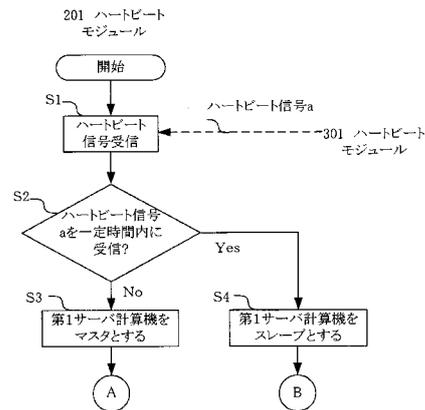
20

203、303 ログ監視モジュール

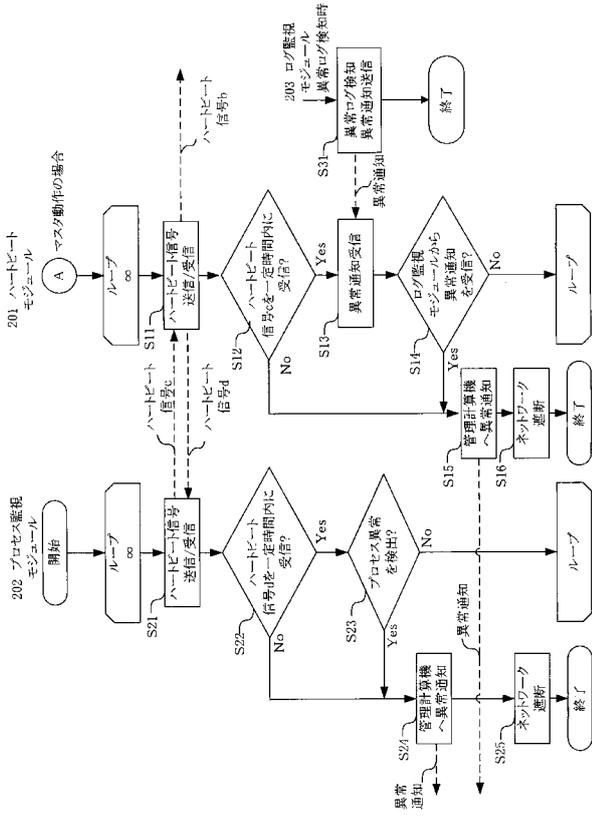
【図1】



【図2】



【 図 3 】



【 図 4 】

